

IBM WebSphere DataPower XC10 アプライアンス バージョン 2.5 資料

IBM® WebSphere® DataPower® XC10 アプライアンス 資料へようこそ。ここでは、IBM WebSphere DataPower XC10 アプライアンス をインストール、保守、および使用する方法に関する情報を見つけることができます。

始めに

[IBM WebSphere DataPower XC10 アプライアンス の概要](#)

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

2.5+ [チュートリアル: 単純データ・グリッド・アプリケーションの入門](#)

[バージョン 2.5 の新機能](#)

[IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成](#)

[WebSphere DataPower XC10 アプライアンス の更新](#)

一般的なタスク

[アプライアンスの構成](#)

[集合およびゾーンの構成](#)

[データ・グリッド の構成](#)

[クライアントの構成](#)

[データ・グリッドの管理](#)

[単純データ・グリッドにアクセスするアプリケーションの開発](#)

[モニター](#)

[セキュリティ](#)

トラブルシューティングおよび その他の情報 サポート

[トラブルシューティング](#)

[サポート・ポータル](#)

[Fix central](#)

[フォーラム](#)

[リリース情報](#)

[IBM ソフトウェア・サポート・ホーム・ページ](#)

[IBM WebSphere DataPower XC10 アプライアンス 研修](#)

[記事](#)

[Redbooks](#)

[IBM Elastic Caching Community](#)

IBM WebSphere DataPower XC10 アプライアンス の概要

IBM® WebSphere® DataPower® XC10 アプライアンス ライセンス・プロダクトは、WebSphere アプリケーションのための、コスト効率がよく高速なドロップイン・キャッシングを実現するよう最適化された専用アプライアンスです。

利点

DataPower XC10 アプライアンス を使用すると、以下の利点が得られます。

容易なスケールアウト

DataPower XC10 アプライアンス には、ビジネスに不可欠なアプリケーション用のデータをホスティングするために使用できる、240 GB のサイズ調整が可能なデータ・グリッド (data grid)が含まれています。データ・グリッド (data grid)にさらにメモリーを追加するには、ご使用の構成にもう 1 つアプライアンスを追加して、データをホスティングするアプライアンスの集合を作成します。

コード変更なしのドロップインの使用

アプリケーションのコードを大幅に変更することなく DataPower XC10 アプライアンス を使用できます。アプライアンスは以下のドロップイン・シナリオで使用できます。

- HTTP 要求のセッション管理
- WebSphere Application Server動的キャッシュのサポート

フォールト・トレランス

データ・グリッド内のデータが自動的に複製され、データ損失のリスクを低減します。

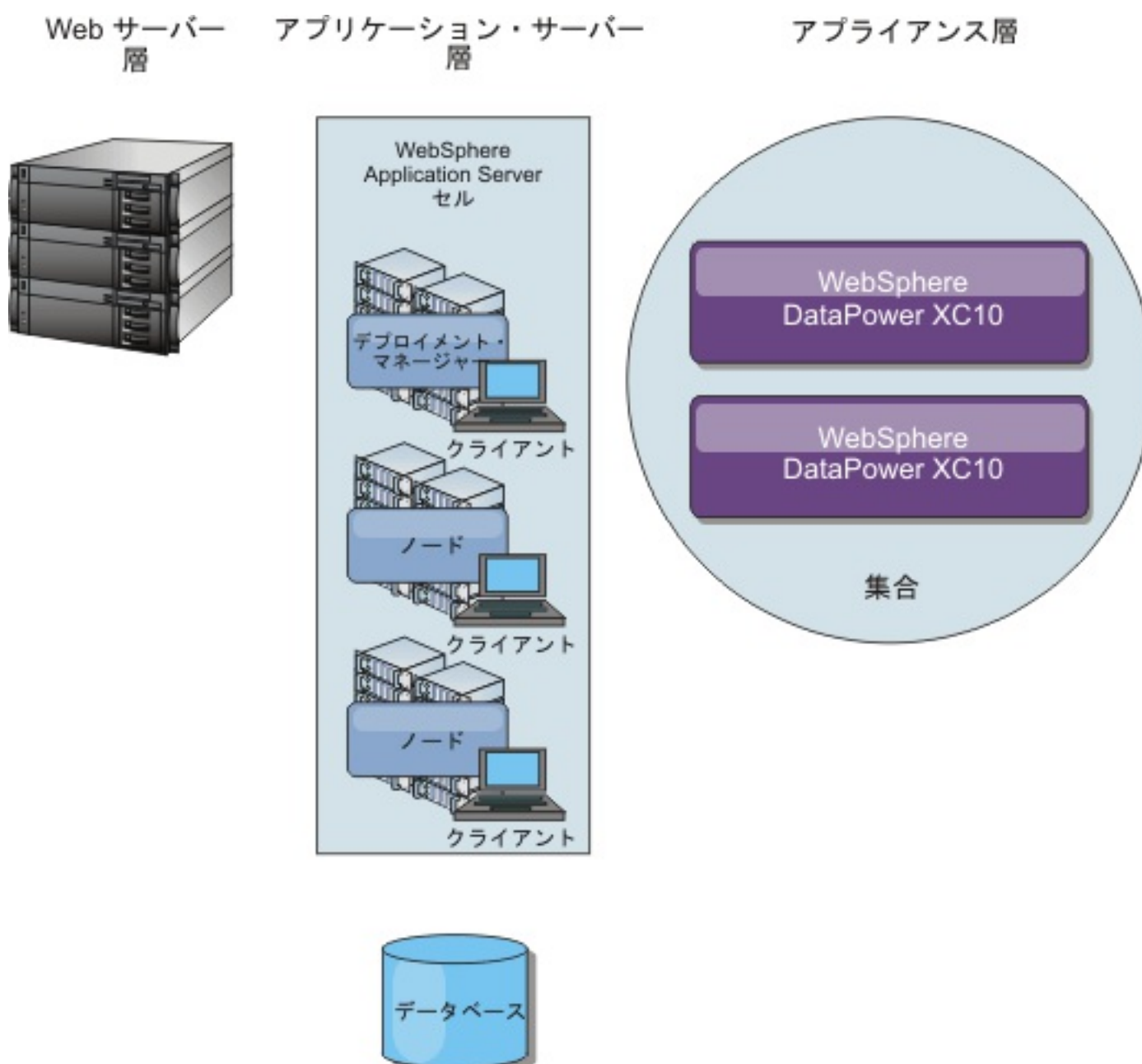
柔軟でシンプルなユーザー管理

ユーザー・インターフェースにより、ご使用のアプライアンスでユーザーおよびユーザー・グループの管理が容易に行えます。また、そのユーザー・インターフェースを使用して、ご使用のデータ・グリッドの作成、管理、およびモニターを行うことができます。

ハイレベル・トポロジー

DataPower XC10 アプライアンス がご使用のアプリケーション・サーバー層の後ろに控えています。アプリケーション・サーバー層では、デプロイメント・マネージャーを含めて、ノードごとに WebSphere eXtreme Scale クライアントをインストールします。このクライアントにより、アプリケーション・サーバー層とアプライアンス層との通信を可能にします。アプライアンス層では、アプライアンス同士を関連付けて集合を形成します。

図1. ハイレベル・トポロジー



バージョン 2.5 の新機能

バージョン 2.5 では、エンタープライズ・データ・グリッド、.NET アプリケーション、セキュリティー機能拡

張、連続照会、モニター向上、およびニア・キャッシュ無効化がサポートされるようになりました。

[リリース情報](#)

製品のサポート Web サイト、製品資料、および製品の最新の更新、制限、および既知の問題へのリンクが提供されています。

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

データ・グリッドは、特定のアプリケーションまたはアプリケーション群で使用するオブジェクトを保持するために作成されるストレージ・ユニットです。集合は、スケーラビリティおよび管理の目的でアプライアンスをグループ化します。ゾーンは、アプライアンスの物理的な場所を定義し、キャッシュ内でのデータの配置を決定するために使用されます。

2.5+ [エンタープライズ・データ・グリッドの概要](#)

エンタープライズ・データ・グリッドは eXtremeIO トランスポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトランスポートおよびシリアライゼーション形式により、Java™ クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。

[トランザクション処理の概要](#)

WebSphere eXtreme Scale クライアントは、データとの相互作用のメカニズムとしてトランザクションを使用します。

[特記事項](#)

[プライバシー・ポリシーに関する考慮事項](#)

バージョン 2.5 の新機能

バージョン 2.5 では、エンタープライズ・データ・グリッド、.NET アプリケーション、セキュリティー機能拡張、連続照会、モニター向上、およびニア・キャッシュ無効化がサポートされるようになりました。

WebSphere eXtreme Scale Client for .NET

WebSphere® eXtreme Scale Client for .NET をインストールすることによって、データ・グリッドにアクセスする .NET アプリケーションをデプロイすることができます。 [詳細...](#)

アプライアンス始動状況

アプライアンスの始動状況をユーザー・インターフェースで確認できるようになりました。 [詳細...](#)

ASP.NET セッション状態ストア・プロバイダー

データ・グリッドにセッション状態を保管するように ASP.NET アプリケーションを構成することができます。 [詳細...](#)

連続照会

データ・グリッドと対話するクライアント・アプリケーションを開発する際、新しいエントリーが挿入されたり更新されたりしたときに自動的なリアルタイムの結果を得る照会が必要になることがあります。連続照会を使用すれば、データ・グリッドでデータの挿入や更新が行われたとき、クライアント Java™ 仮想マシン (JVM) で通知されるようにすることができます。このフィーチャーは、開発者、管理者、またはその両者がグリッドおよびデータの管理をより容易に行えるようにします。 [詳細...](#)

タスクの削除

すべてのタスクをユーザー・インターフェースで削除することができます。 [詳細...](#)

.NET 用クライアント・プロパティの動的更新

クライアント・プロパティ・ファイル内のプロパティ値に対して行われた手動変更を動的に有効にするように WebSphere eXtreme Scale Client for .NET を構成することができます。データ・グリッドへの既存の接続を再始動しなくても変更が有効になります。 [詳細...](#)

FIPS を使用可能に設定

すべての暗号化ネットワーク通信で連邦情報処理標準 (FIPS) 140-2 を使用するようにアプライアンス集合を構成することができます。この標準により、通信で送信されるデータの優れた保護が確保されます。 [詳細...](#)

クライアント・プロパティ・ファイル内の credentialGeneratorProps プロパティを .NET 用にエンコード

FilePasswordEncoder ユーティリティを使用して Client.Net.properties ファイル内の credentialGeneratorProps プロパティの値をエンコードすることができます。 [詳細...](#)

エンタープライズ・データ・グリッド

エンタープライズ・データ・グリッドは eXtremeIO トランスポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトランスポートおよびシリアライゼーション形式により、Java クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。 [詳細...](#)

eXtreme Data Format (XDF)

XDF は MapSerializerPlugin プラグインに基づいており、現在は、IBM® eXtremeIO (XIO) の実行時に使用されるデフォルトのシリアライゼーション・テクノロジーです。

入門用サンプル

入門用サンプルを使用して、アプライアンス上のデータ・グリッドにアクセスする Java および .NET クライアント・アプリケーションを実行することができます。 [詳細...](#)

グリッド別名

REST ゲートウェイを使用して、グリッド別名を作成し、管理することができます。グリッド別名は、複数のグリッドを同時に取り込む必要がある場合に使用します。 [詳細...](#)

ホスト名管理

publishHost プロパティを使用して、公開されたホスト名を構成したりオーバーライドしたりすることができます。このプロパティを使用することで、クライアントとサーバー間での情報のやりとりをよりきめ細かく制御することが可能となり、そのため、データ・グリッド内のノード相互間の通信を最適化することができます。 [詳細...](#)

LDAP over SSL

Lightweight Directory Access Protocol (LDAP) ディレクトリーを確保して、アプライアンスにアクセスするユーザーの認証を行います。この LDAP ディレクトリーを確保するには、SSL 接続経由で LDAP の認証を行うようにアプライアンスを構成します。 [詳細...](#)

ニア・キャッシュの無効化

できるだけ速やかにニア・キャッシュから失効データが削除されるようにニア・キャッシュの無効化を構成することが

できます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化がトリガーされます。 [詳細...](#)

新しい xscmd ユーティリティー・コマンドとパラメーター

- **xscmd -c getNotificationFilter** コマンド: このコマンドを実行すると、メッセージ・センターからの新しい通知のための現行フィルターが表示されます。 [詳細...](#)
- **xscmd -c listenForNotifications** コマンド: このコマンドを実行すると、メッセージ・センターからの新しい通知が listen されます。 [詳細...](#)
- **xscmd -c setNotificationFilter** コマンド: このコマンドを実行すると、メッセージ・センターからの新しい通知のためのフィルターが作成されます。 [詳細...](#)
- **xscmd -c showLinkedDomains** コマンド: このコマンドを実行すると、ご使用のローカル・カタログ・サービス・ドメインにリンクされているカタログ・サービス・ドメインが表示されます。 [詳細...](#)
- **xscmd -c showNotificationHistory** コマンド: このコマンドを実行すると、イベント通知履歴の出力が表形式で表示されます。
- **xscmd -c showSessionSize** コマンド: このコマンドを実行すると、指定したセッションのサイズが表示されます。 [詳細...](#)
- **-to** または **--timeout** パラメーター: ネットワーク・ブラウン・アウトまたはシステム損失時にオペレーティング・システム・タイムアウトや他のネットワーク・タイムアウトが起こるまで待たないようにするためにタイムアウト値を減らすには、このパラメーターを指定します。 [詳細...](#)
- **-hc** または **--linkHealthCheck** パラメーター: プライマリー断片の持つ集合リンクの数が適切であるかどうかを確認するには、このパラメーターを **xscmd -c showLinkedPrimaries** コマンドで使用します。
- **xscmd -c listDisabledForPlacement** コマンド: 断片配置に対して使用不可にされた断片コンテナのリストを表示するには、このコマンドを実行します。
- **xscmd -c listIndoubts** コマンド: 未確定トランザクションのリストを表示するには、このコマンドを実行します。ある区画で起こった可能性があるロック・タイムアウト例外を解決したいときには、このコマンドを実行してください。 [詳細...](#)
- **xscmd -c showReplicationState** および **xscmd -c showDomainReplicationState** コマンド: ご使用のすべてのカタログ・サーバーまたはカタログ・サービス・ドメインについて改訂の状態を調べるには、これらのコマンドを実行します。 [詳細...](#)
- **xscmd -c showTransport** コマンド: カatalog・サービス・ドメインのトランスポート・タイプを表示するには、このコマンドを実行します。 [詳細...](#)

動的キャッシュ・データ・グリッドにおける **requestRetryTimeout** プロパティのオーバーライド
クライアント・プロパティ・ファイル内の **requestRetryTimeout** プロパティをオーバーライドしたり、タイムアウトまでの要求を実行できる時間 (ミリ秒) を指定したりするには、**com.ibm.websphere.xs.dynacache.request_retry_timeout_override** 動的キャッシュ・プロパティを使用します。 [詳細...](#)

アプライアンス・コマンド行インターフェース内からの xscmd ユーティリティーの実行

アプライアンス・コマンド行インターフェース内から **xscmd** ユーティリティーを実行することができます。アプライアンス・コマンド行インターフェース内からこのユーティリティーを呼び出す際に、セキュリティー・サーバー接続およびカタログ・サーバー接続についてはコマンド行引数を指定する必要はありません。 [詳細...](#)

秘密鍵生成

Web コンソールで固有の秘密鍵を指定するには、出荷時のデフォルト認証秘密鍵をオーバーライドします。 [詳細...](#)

更新された Simple Network Monitoring Protocol (SNMP) 統計

SNMP クライアントが使用できる SNMP データを指定するために WebSphere DataPower XC10 アプライアンス で提供される管理情報ベース (MIB) は、新しい統計で更新されるようになりました。新しい統計には、**applianceUsedBytes**、**applianceCapacity**、**gridUsedBytes**、および **gridCapacity** が含まれています。SNMP モニターを使用可能にし、この MIB ファイルをダウンロードする方法については、[アプライアンスでの SNMP モニターの使用可能化](#)を参照してください。

タイムアウト値および要求タイムアウト値の指定

eXtremeIO (XIO) トランスポートを使用するかオブジェクト・リクエスト・ブローカー・トランスポートを使用するかにかかわらず、接続あるいは要求が処理される時間を指定することができます。例えば、XIO の場合は、**xioTimeout** プロパティを使用してアウトバウンド・ソケット接続試行のタイムアウト値を指定でき、さらに、**xioRequestTimeout** プロパティを使用して要求がタイムアウトになるまでの要求の待ち時間 (秒数) を指定することができます。**requestRetryTimeout** プロパティは、XIO トランスポートまたは ORB トランスポートのいずれにも適用することができます。このプロパティを使用して、例外が発生した後いつまで要求の処理を続行するかを指定します。 [詳細...](#)

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

リリース情報

製品のサポート Web サイト、製品資料、および製品の最新の更新、制限、および既知の問題へのリンクが提供されています。

- [最新の更新、制限、および既知の問題へのアクセス](#)
- [システムのアクセスおよびソフトウェア要件](#)
- [製品資料へのアクセス](#)
- [製品のサポート Web サイトへのアクセス](#)
- [IBM ソフトウェア・サポートへの連絡](#)

最新の更新、制限、および既知の問題へのアクセス

リリース情報は、製品のサポート・サイトの技術情報から入手できます。WebSphere® DataPower® XC10 アプライアンスのすべての技術情報のリストを参照するには、[サポート Web ページ](#)にアクセスしてください。ここで提供されているリンクをクリックすると、サポート Web ページで関連リリース・ノートが検索されます。関連リリース・ノートはリストとして返されます。

- リリース・ノートのリストを表示するには、[サポート Web ページ](#)にアクセスしてください。

システムのアクセスおよびソフトウェア要件

ハードウェアおよびソフトウェア要件は以下のページに記載されています。

- [システム要件](#)

製品資料へのアクセス

情報セット全体に関しては、[ライブラリー・ページ](#)にアクセスしてください。

製品のサポート Web サイトへのアクセス

最新の技術情報、ダウンロード、フィックス、およびその他のサポート関連情報を検索するには、[サポート・ページ](#)にアクセスしてください。

IBM ソフトウェア・サポートへの連絡

この製品で問題が発生した場合には、最初に以下のアクションを試行してください。

- 製品資料に記載されているステップを実行します。
- 関連資料をオンライン・ヘルプで検索します。
- エラー・メッセージをメッセージ解説書で検索します。

上記の方法で問題を解決できない場合、IBM® テクニカル・サポートに連絡してください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

親トピック: [トラブルシューティング](#)

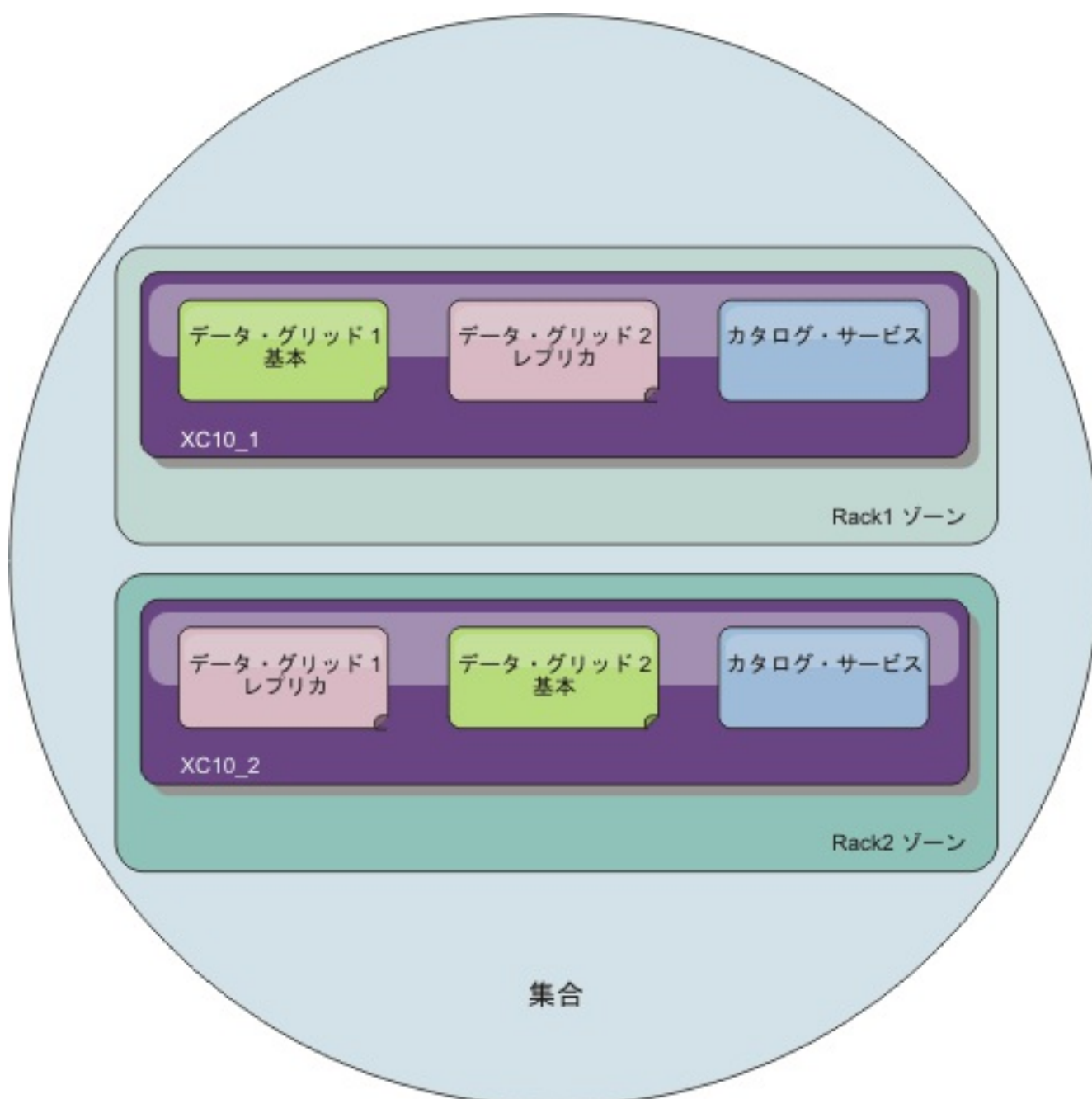
アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド

データ・グリッドは、特定のアプリケーションまたはアプリケーション群で使用するオブジェクトを保持するために作成されるストレージ・ユニットです。集合は、スケーラビリティおよび管理の目的でアプライアンスをグループ化します。ゾーンは、アプライアンスの物理的な場所を定義し、キャッシュ内でのデータの配置を決定するために使用されます。

アプライアンスのトポロジー

集合およびゾーンは、いずれも 1 つ以上の WebSphere® DataPower® XC10 アプライアンス に関連付けられています。各アプライアンスは 1 つの集合および 1 つのゾーンのメンバーとなり得ます。各アプライアンスは、キャッシュ・データを保持する複数のデータ・グリッドをホスティングします。

図 1. 集合およびゾーンのトポロジー



重要: データ・グリッドを高可用性にするために、2 つのアプライアンスが必要です。

集合とマルチマスター複製

マルチマスター複製は、複数のデプロイメント環境全体で連続可用性を確保するための技法です。マルチマスターのトポロジーは、複数の集合を作成して、それらをリンクで接続することで WebSphere DataPower XC10 アプライアンスに実装できます。集合を定義すると、次の情報が集合内のアプライアンス間で共有されます: データ・グリッド、モニター情報、集合およびゾーンのメンバー、およびユーザー。これらの情報のいずれかを更新すると、行われた変更が集合内の他のすべてのアプライアンスで保持されます。カタログ・サービスはアプライアンス間の通信を可能にします。カタログ・サービスはカタログ・サーバーのグループです。集合内の各アプライアンスは、集合ごとに 3 つのカタログ・サーバーを限度に、カタログ・サーバーを実行します。集合内に 4 つ以上のアプライアンスがある場合、カタログ・サービスは集合に追加された最初の 3 つのアプライアンス上で実行されます。カタログ・サーバーを持つアプライアンスを集合から除去する場合や、カタログ・サーバーを持つアプライアンスが使用できなくなった場合は、次に集合に追加されたアプライアンスがカタログ・サーバーを実行します。カタログ・サーバーは他のアプライアンスにフェイルオーバーしません。

アプライアンスを集合に追加するには、別のアプライアンスの集合構成パネルにそのアプライアンスのホスト名および共通鍵を追加します。この構成は集合内のどのアプライアンスでも作成できますが、それは集合のメンバーシップが集合メンバー間で保持されるためです。

アプライアンスは 1 つの集合内にのみ存在できます。既にある集合に存在するアプライアンスを別の集合に追加することはできません。2 つの集合を結合して 1 つの集合にすることもできません。別々の集合のアプライアンスを結合するには、各アプライアンスをそれぞれの集合から除去し、各アプライアンスをスタンドアロンにする必要があります。そうすれば、すべてのアプライアンスを含む新しい集合を作成できます。

集合を使用してほとんどの構成の変更が行える一方で、「アプライアンス」 > 「アプライアンスの設定」および「アプ

ライセンス」 > 「トラブルシューティング」の各パネルで設定を変更するためには、所定のアプライアンスにログインしなければなりません。

ゾーン

ゾーンは 例えば市区町村や研究所内のラックの場所など、アプライアンスの 物理的な場所と関連付けられます。ゾーンは、カタログ・サービスでご利用の データ・グリッド内のデータの保管場所を定義するのに 役立ちます。例えば、ある 1 つのゾーンにデータ・グリッド (data grid) の基本情報が保管されている場合に、レプリカのデータは別のゾーンにあるアプライアンスに保管されます。この構成では、データ・グリッド (data grid) の 基本を保持するアプライアンスに障害が起ると、基本からレプリカに フェイルオーバーが発生します。

データ・グリッド

データ・グリッドはアプリケーション用のオブジェクトを保持します。オブジェクトをキャッシングすることで、アプリケーションのパフォーマンスが向上します。 データ・グリッドには 3 つのタイプがあります。

単純データ・グリッド (data grid)

単純データ・グリッドは、データをキー値のペアで保持します。例えば、データベース照会の結果を単純 データ・グリッド (data grid) に保管できます。単純データ・グリッド (data grid) を実装するには、ObjectMap API を使用します。ObjectMap API は、Java™ Maps と同じように動作します。

セッションデータ・グリッド (data grid)

WebSphere Application Server セッションを使用する場合は、セッション管理データ用のアプライアンス上でセッションデータ・グリッド (data grid) を使用するようにアプリケーションを構成できます。新規のアプリケーションをインストールする際に、セッション データ・グリッド (data grid) を使用するようにアプリケーションを構成できます。既存のアプリケーションまたはサーバーの設定を更新して、アプライアンス上でセッションデータ・グリッド (data grid) を使用することもできます。

動的キャッシュデータ・グリッド (data grid)

WebSphere Application Server の動的キャッシュから取り出したデータを保管するために、アプライアンス上の動的キャッシュデータ・グリッド (data grid) を使用できます。動的キャッシュ API を使用して作成されたアプリケーションや、サーブレットのようなコンテナ・レベル・キャッシングを使用するアプリケーションを使用可能にして、アプライアンスをキャッシュ・プロバイダーとして使用することができます。その結果として、アプリケーション・サーバーによって使用されるメモリーが少なくなります。すべてのキャッシュ・データが アプライアンスにオフロードされ、アプリケーション・サーバーのメモリーにはもう保管されません。

データ・グリッド・レプリカ

特定のデータ・グリッド (data grid) のレプリカのターゲット数を定義できます。集合に少なくとも 2 つのアプライアンスがあるときに、レプリカは作成されます。アプライアンスが 1 つの場合、レプリカは作成されません。集合内のアプライアンス数が n の場合、1 つのアプライアンスが基本データ・グリッドをホスティングするため、レプリカの最大数は $n-1$ になります。レプリカのターゲット数が現行の $n-1$ よりも大きい場合は、アプライアンスを集合に追加するときに、追加のレプリカを作成できます。レプリカの数、今後必要となる可能性があるレプリカの最大数に設定しておくことを検討してください。レプリカ設定を編集すると、データ・グリッドの内容が消去されます。今後のレプリカの数、考慮してこの値を設定してください。新しいアプライアンスが集合に追加されると、追加のレプリカが作成されます。基本およびレプリカのデータ・グリッドは、集合内のすべてのアプライアンス間で均一に分散またはストライピングされます。新しいアプライアンスが集合に追加されると、基本およびレプリカのデータ・グリッドを分散するための再バランシングが行われます。

レプリカは、同期レプリカまたは 非同期レプリカのいずれかになります。同期レプリカは、基本データ・グリッド (data grid) 上の トランザクションの一部として更新を受け取ります。非同期レプリカは、基本データ・グリッド (data grid) 上の トランザクションがコミットされた後に 更新されます。同期レプリカはデータの一貫性を保証しますが、非同期レプリカに比べると要求への応答時間が長くなります。非同期レプリカは データの一貫性においては同様の保証ができませんが、トランザクションを より速く完了させることができます。データ・グリッド (data grid) にはデフォルトで 1 つの非同期レプリカがあります。配置アルゴリズムによって、レプリカの配置場所が制御されます。

マップ

マップは、キー値のペアにあるデータ・グリッドのデータを含むデータ構造です。1 つのデータ・グリッドは、データ・グリッドおよびデータ・グリッド・レプリカに存在する複数のマップを持つことができます。

ご利用のクライアント・アプリケーションを特定の名前の付いたマップに接続させることで、データ・グリッド内に追加のマップを作成できます。動的マップが自動的に作成されます。

集合リンク

障害の誤検出が起こる可能性があるため、単一の集合が信頼できないネットワーク上に広がってはなりません。しかし、信頼できないネットワークへの接続を持つアプライアンス間で、データ・グリッドのデータを複製する必要が生じる場合もあります。 このタイプのトポロジーを使用することになる一般的なシナリオには以下のようなものがあります。

- 1 つの集合がアクティブでもう 1 つの集合がバックアップとして使用されているデータ・センター間での災害復旧
- 地理的に近いクライアントのために、すべての集合がアクティブになっている地理的に分散されたデータ・センター

—。

2つの集合を接続すると、同じ名前を持つすべてのデータ・グリッドが集合間で非同期的に複製されます。これらのデータ・グリッドは、各集合内に同じ数のレプリカを持ち、同じ動的マップ構成を持っていない限りなりません。

[マルチマスター複製を実装する集合のリンクのトポロジー](#)

複数の集合を組み込んだデプロイメントのトポロジーを選択する際、いくつかの異なるオプションがあります。マルチマスター複製のトポロジーは、複数の集合を作成して、それらをリンクで接続することで DataPower XC10 アプライアンス に実装できます。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

関連タスク:

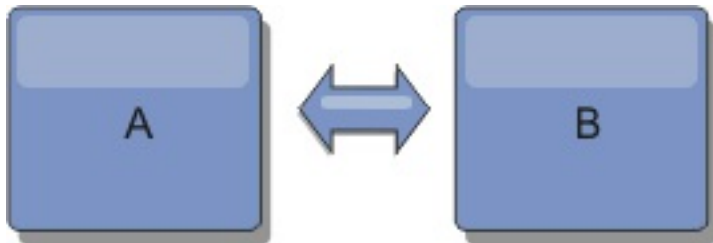
[集合間のマルチマスター複製の構成](#)

マルチマスター複製を実装する集合のリンクのトポロジー

複数の集合を組み込んだデプロイメントのトポロジーを選択する際、いくつかの異なるオプションがあります。マルチマスター複製のトポロジーは、複数の集合を作成して、それらをリンクで接続することで DataPower® XC10 アプライアンス に実装できます。

集合 を接続するリンク

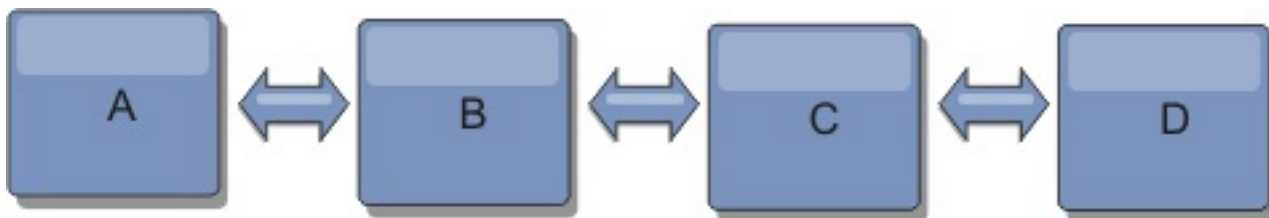
複製データ・グリッドのインフラストラクチャーは、集合間を双方向のリンクで接続したカタログ・サービス・ドメインのグラフです。リンクを使用して、2つの集合はデータ変更内容をやりとりできます。例えば、最も単純なトポロジーは、カタログ・サービス・ドメイン間に単一のリンクを持つ1対の集合です。集合は、左から A、B、C というようにアルファベット順で指定されています。リンクは、遠距離にわたる広域ネットワーク (WAN) を経由する場合があります。リンクが遮断されたとしても、いずれかの集合 (collective) でまだデータを変更できます。トポロジーは、リンクが集合と再接続したときに変更を調整します。ネットワーク接続が中断されると、リンクは自動的に再接続しようとします。



リンクをセットアップすると、この製品はまず、すべての集合 (collective) を同一にしようと試みます。次に、いずれかの集合 (collective) で変更が発生すると、eXtreme Scale は同一の状態を維持するよう試みます。目標は、各集合 (collective) がリンクで接続されたすべての他の集合 (collective) の正確なミラーになることです。集合間の複製リンクは、1つの集合 (collective) で行われたすべての変更を確実に他の集合にコピーするのに役立ちます。

ライン・トポロジー

ライン・トポロジーはこのような単純なデプロイメントですが、かなりのリンク品質を実証します。まず、変更を受け取るために、集合 (collective) は直接すべての他の集合 (collective) に接続する必要がありません。集合 (collective) B は集合 (collective) A から変更をプルします。集合 (collective) C は、集合 A と C を接続する集合 (collective) B を介して集合 (collective) A から変更を受信します。同様に、集合 (collective) D は集合 (collective) C を介して別の集合から変更を受信します。この機能により、変更配布の負荷が変更のソースから離れた場所に分散できます。



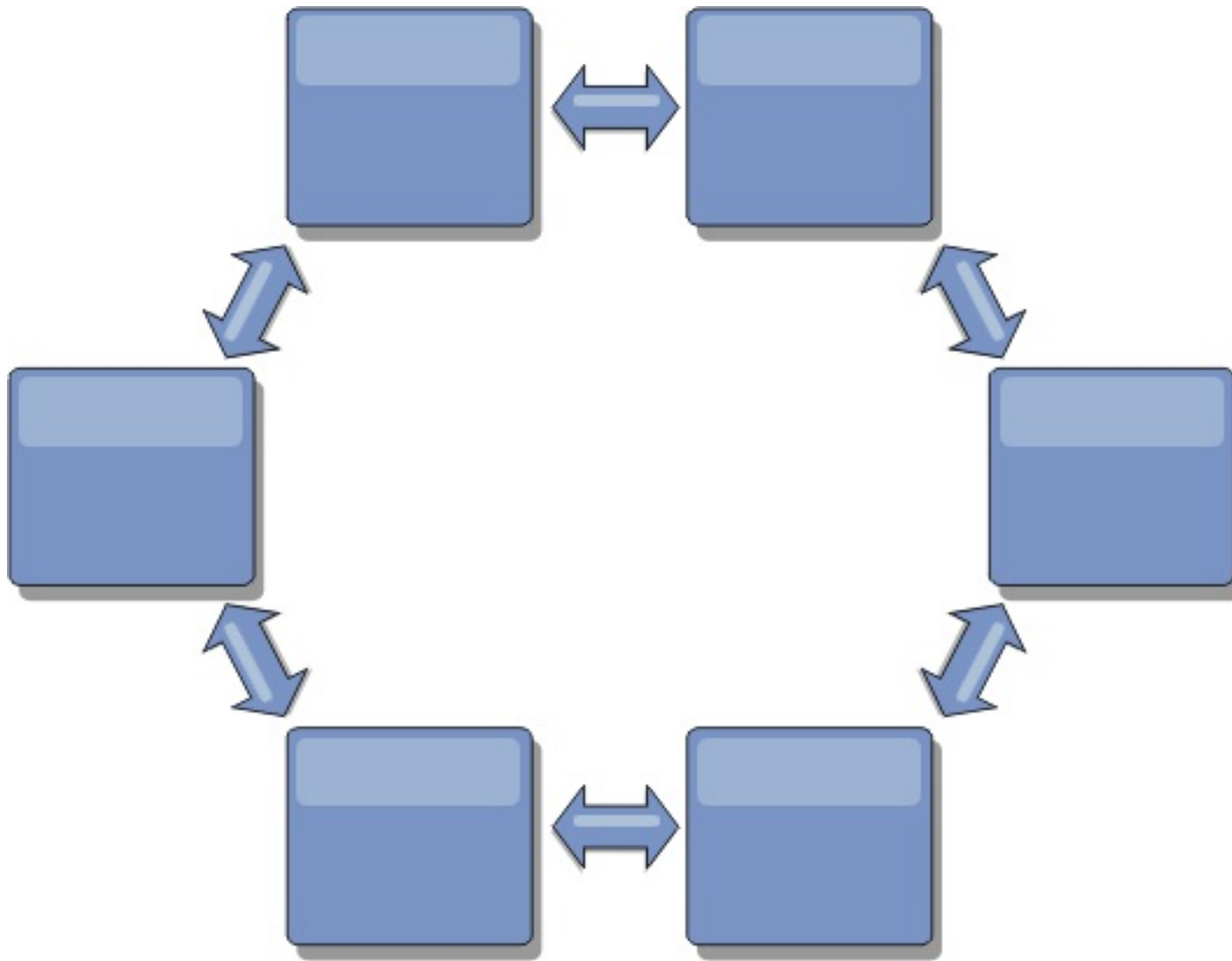
集合 (collective) C に障害が起こった場合、以下のアクションの発生が考えられることに注意してください。

1. 集合 (collective) D は、集合 (collective) C が再開されるまで孤立します。
2. 集合 (collective) C は、集合 (collective) A のコピーである集合 (collective) B と自分自身を同期させます。
3. 集合 (collective) D は、集合 (collective) C を使用して、集合 (collective) A と B で発生した変更と自分自身を同期させます。これらの変更は最初は、集合 (collective) D が孤立していた間 (集合 (collective) C がダウンしていた間) に発生しました。

最終的に、集合 A、B、C、および D はすべて、互いのドメインと再び同一になります。

リング・トポロジー

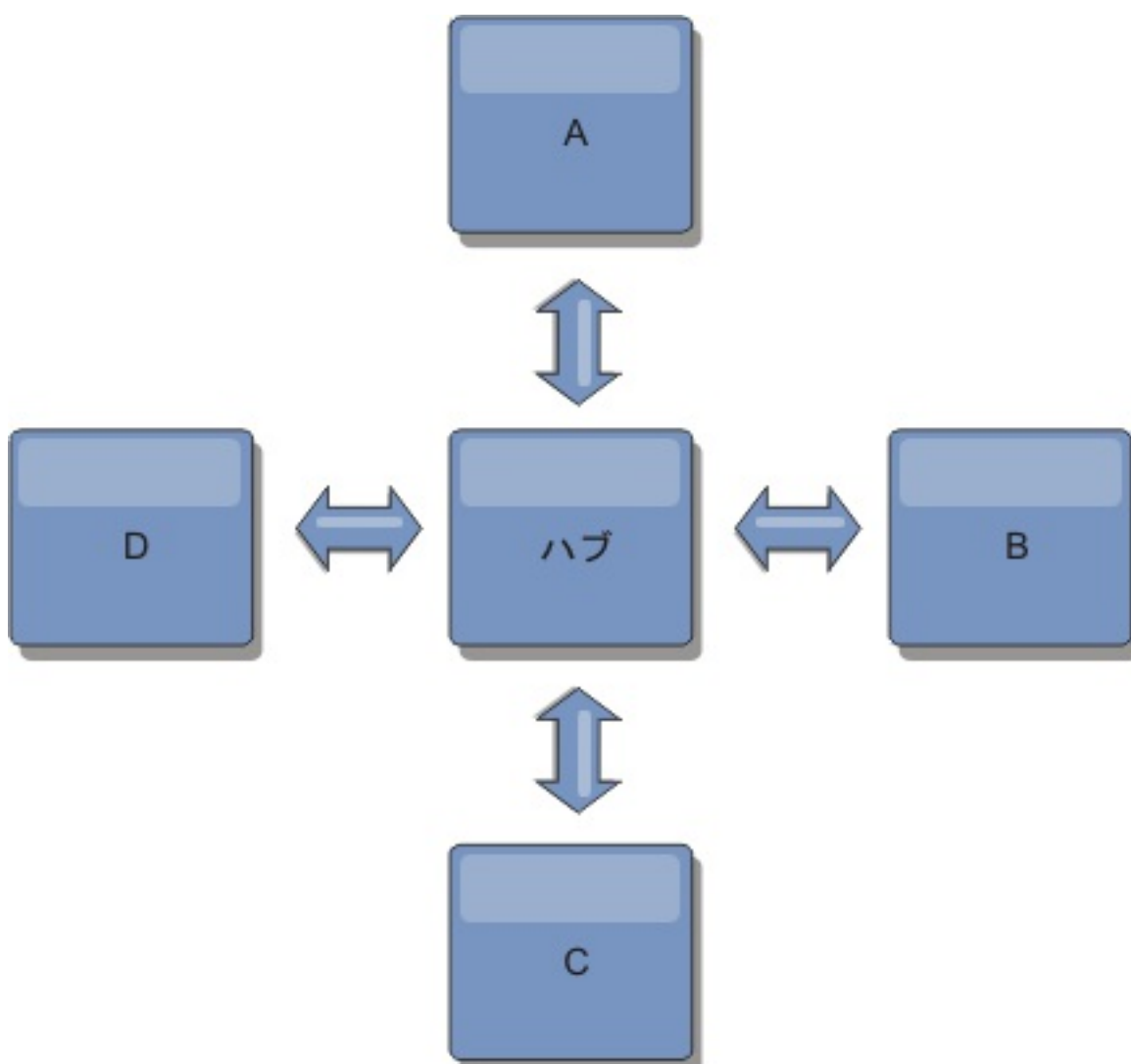
リング・トポロジーは、より回復力のあるトポロジーの例です。集合 (collective) または単一リンクに障害が起こった場合でも、残った集合がまだ変更を取得できます。その集合は、障害から離れて、リングの周りを回ります。リング・トポロジーの大きさには関係なく、各集合 (collective) は他の集合とのリンクを最大2つ持ちます。変更を伝搬するための待ち時間は長くなる場合があります。特定の集合 (collective) での変更は、すべての集合にその変更が反映されるまで、複数のリンクを経由して伝搬する必要がある場合があります。ライン・トポロジーにも同じ特性があります。



リングの中心に置いたルート・集合 (collective) を使用した、より洗練されたリング・トポロジーをデプロイすることも可能です。ルート・集合 (collective) は、調整の中心点として機能します。他の集合は、ルート・集合 (collective) で生じた変更に対する調整のリモート・ポイントとして働きます。ルート・集合 (collective) は、集合間の変更をアービトレーションすることができます。ルート・集合 (collective) を囲む複数のリングがリング・トポロジーに含まれている場合、集合 (collective) は最も内側にあるリング間の変更のみをアービトレーションすることができます。ただし、アービトレーションの結果は他のリングの集合にも広がります。

ハブ・アンド・スポーク・トポロジー

ハブ・アンド・スポーク・トポロジーでは、ハブ・集合 (collective) を経由して変更が伝搬します。ハブは指定される唯一の中間集合 (collective) であるため、ハブ・アンド・スポーク・トポロジーでは待ち時間が短縮されます。ハブ・集合 (collective) は、リンク経由ですべてのスポーク・集合 (collective) に接続されています。ハブは、集合間の変更を配布します。ハブは、衝突に対して調整のポイントとして機能します。更新頻度の高い環境では、同期を保つために、スポークよりも多くのハードウェア上でハブを稼働する必要がある場合があります。WebSphere® DataPower XC10 アプライアンスは、直線的に拡大するように設計されています。つまり、問題なく、必要に応じてハブをさらに大きくすることができます。ただし、ハブに障害が起こった場合は、変更はハブが再始動するまで配布されません。スポーク・集合上の変更は、ハブが再接続された後に配布されます。



また、完全に複製したクライアントを使用したストラテジー、すなわち、ハブとして稼働しているサーバーのペアを使用

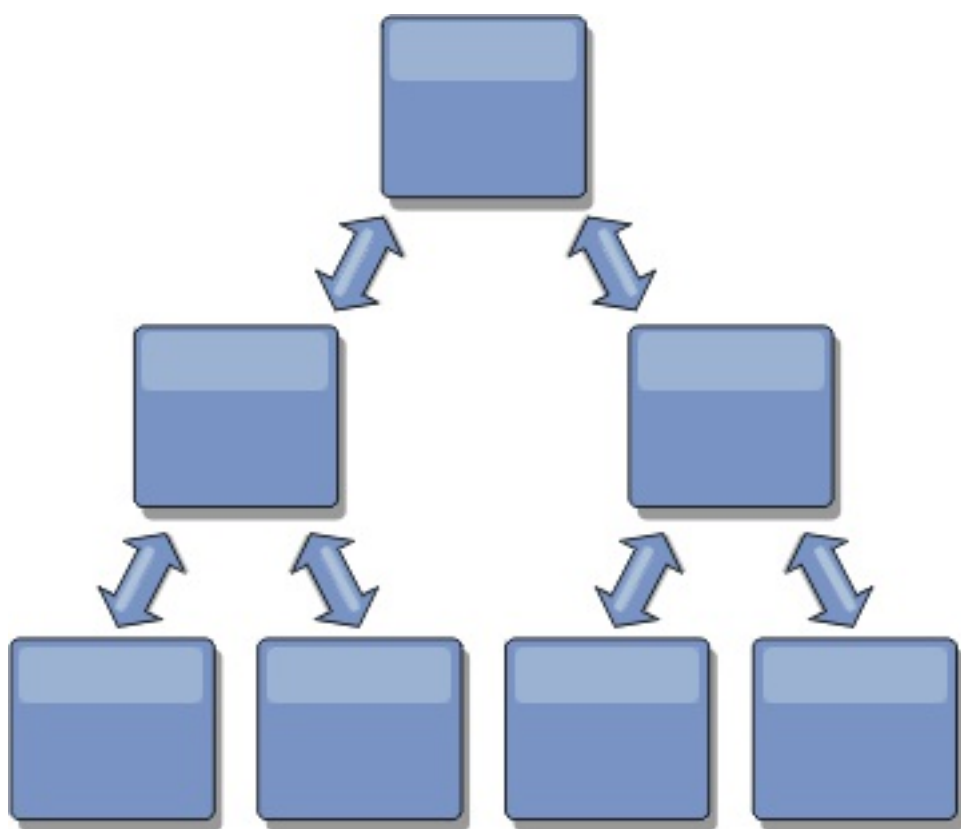
するトポロジーのバリエーションを使用することもできます。各クライアントは、クライアント JVM 内に、必要なものを完備した単一コンテナ・データ・グリッドとカタログを作成します。クライアントは、そのデータ・グリッドを使用してハブ・カタログに接続します。この接続により、クライアントはハブへの接続を取得すると、すぐにハブと同期するようになります。

クライアントによって行われた変更は、クライアントに対してローカルで、非同期でハブに複製されます。ハブはアービトレーション・集合 (collective) として機能し、すべての接続されたクライアントに変更を配布します。完全複製クライアントのトポロジーは、OpenJPA などのオブジェクト・リレーショナル・マッパーに信頼性の高い L2 キャッシュを提供します。変更はハブを介してクライアント JVM 間に迅速に配布されます。キャッシュ・サイズを使用可能なヒープ・スペース内に含むことができる場合、このトポロジーは L2 のこのスタイルにとって信頼できるアーキテクチャーです。

必要であれば、複数の区画を使用して、複数の JVM 上にハブ・集合 (collective) を拡張します。すべてのデータはまだ単一のクライアント JVM に収まらなければならないため、複数の区画を使用してハブの容量を増加させ、変更の配布とアービトレーションを行います。ただし、複数の区画を使用しても、単一集合 (collective) の容量は変更されません。

ツリー・トポロジー

非循環有向ツリーを使用することもできます。非循環ツリーには循環やループはなく、有向セットアップにより、リンクの存在は親と子の間にのみ制限されます。この構成は、多くの集合を含むトポロジーで役立ちます。これらのトポロジーでは、すべての接続可能なスポークに接続されている中央ハブを使用することは実用的ではありません。また、このタイプのトポロジーは、ルート・集合 (collective) を更新することなく子集合を追加する必要がある場合にも便利です。



ツリー・トポロジーでもまだ、ルート・集合 (collective) に調整の中心点を置くことができます。第 2 レベルはまだ、それらの下の集合 (collective) で生じた変更に対する調整のリモート・ポイントとして機能します。ルート・集合 (collective) は、第 2 レベルにある集合間の変更のみをアービトレーションすることができます。それぞれが各レベルで N 個の子を持つ、n 進ツリーを使用することもできます。それぞれの集合 (collective) は、n 個のリンクに接続します。

完全複製クライアント

このトポロジー変化には、ハブとして稼働する 1 対のサーバーが含まれます。各クライアントは、クライアント JVM 内に、必要なものを完備した単一コンテナ・データ・グリッドとカタログを作成します。クライアントは、そのデータ・グリッドを使用してハブ・カタログに接続します。これにより、クライアントはハブへの接続を取得すると、すぐにハブと同期するようになります。

クライアントによって行われた変更は、クライアントに対してローカルで、非同期でハブに複製されます。ハブはアービトレーション・集合 (collective) として機能し、すべての接続されたクライアントに変更を配布します。完全複製クライアントのトポロジーは、OpenJPA などのオブジェクト・リレーショナル・マッパーに適した L2 キャッシュを提供します。変更はハブを介してクライアント JVM 間に迅速に配布されます。キャッシュ・サイズをクライアントの使用可能なヒープ・スペース内に含むことができる限り、このトポロジーは L2 のこのスタイルに適したアーキテクチャーです。

必要であれば、複数の区画を使用して、複数の JVM 上にハブ・集合 (collective) を拡張します。すべてのデータはまだ単一のクライアント JVM に収まらなければならないため、複数の区画を使用してハブの容量を増加させ、変更の配布とアービトレーションを行います。ただし、単一集合 (collective) の容量は変更しません。

親トピック: [アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連タスク:

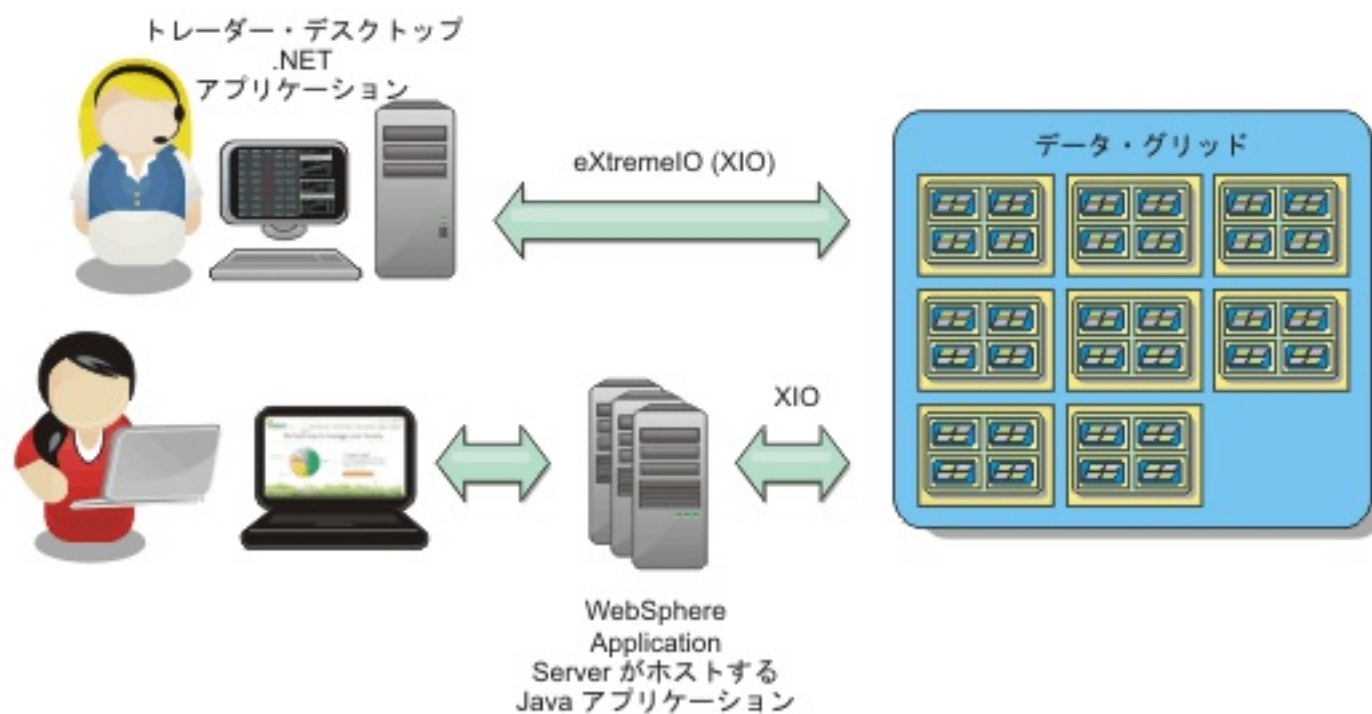
[集合間のマルチマスター複製の構成](#)

エンタープライズ・データ・グリッドの概要

エンタープライズ・データ・グリッドは eXtremeIO トラnsポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトラnsポートおよびシリアライゼーション形式により、Java™ クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。

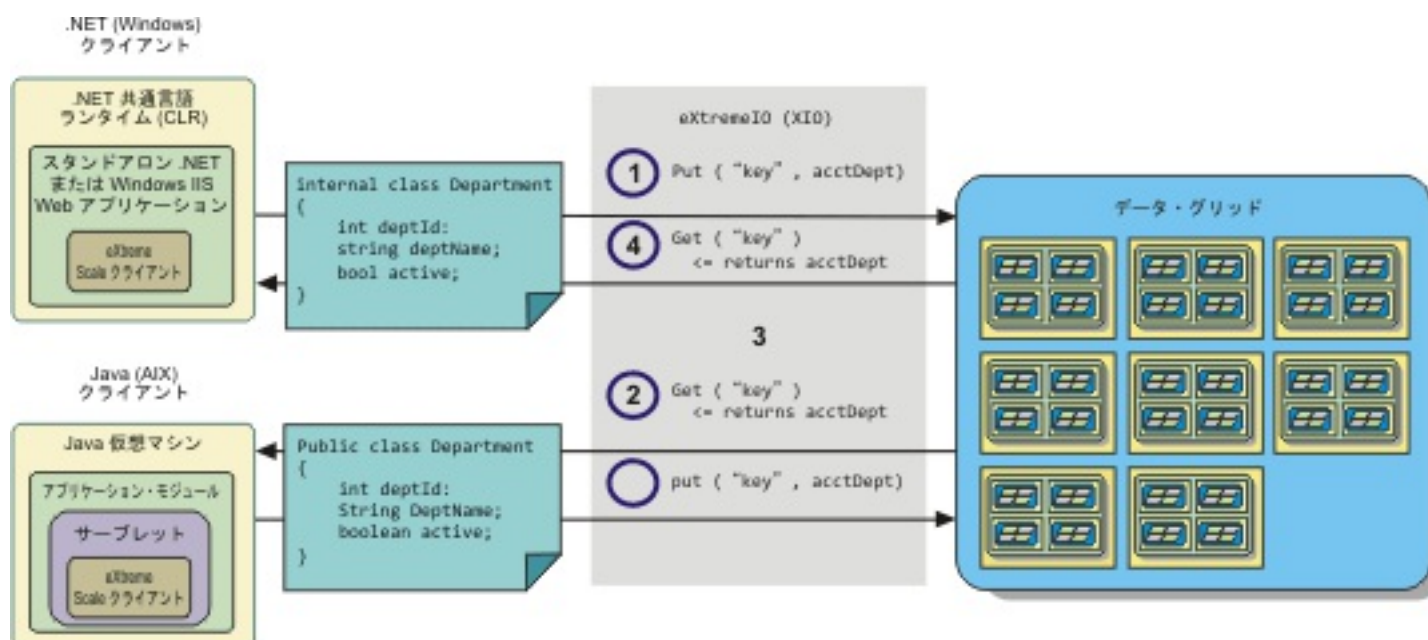
エンタープライズ・データ・グリッドにより、異なるプログラミング言語で書かれた、さまざまなタイプのアプリケーションを作成して、データ・グリッド内の同じオブジェクトにアクセスすることができます。これまでのリリースでは、データ・グリッド・アプリケーションは Java プログラミング言語でしか書くことができませんでした。エンタープライズ・データ・グリッド機能を使用すれば、Java アプリケーションと同じデータ・グリッドでオブジェクトの作成、検索、更新、および削除を行える .NET アプリケーションを書くことができます。

図1. エンタープライズ・データ・グリッドの概要



異なるアプリケーションにまたがるオブジェクト更新

図2. エンタープライズ・データ・グリッド・オブジェクト更新フロー



1. .NET クライアントが .NET 形式のデータをデータ・グリッドに保存します。
2. データは汎用形式で保管されるので、Java クライアントからこのデータが要求されたとき、このデータを Java 形式に変換することができます。
3. Java クライアントがデータを更新して再保存します。
4. .NET クライアントが更新されたデータにアクセスします。そのとき、データは .NET 形式に変換されます。

トラnsポート・メカニズム

eXtremeIO (XIO) はクロスプラットフォーム・トラnsポート・プロトコルです。XIO は Java の制約を受けるオブジェクト・リクエスト・ブローカー (ORB) に取って代わるものです。ORB の場合は、WebSphere® DataPower® XC10 アプライアンスは Java ネイティブ・クライアント・アプリケーションと密接に結びついています。XIO は、特にデータ・キャッシングを対象とし、異なるプログラミング言語で書かれたクライアント・アプリケーションがデータ・グリッドに接続できるようにする、カスタマイズされたトラnsポート・メカニズムです。

シリアライゼーション形式

eXtreme データ形式 (XDF) はクロスプラットフォームのシリアライゼーション形式です。XDF は、ObjectGrid 記述子 XML ファイルで copyMode 属性値が COPY_TO_BYTES であるマップでの Java シリアライゼーションに取って代わります。XDF を使用すれば、パフォーマンスが向上し、データがよりコンパクトになります。さらに、XDF の導入により、異なるプログラミング言語で書かれたクライアント・アプリケーションが同じデータ・グリッドに接続できるようになります。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

親トピック: [シナリオ: エンタープライズ・データ・グリッドの構成](#)

次のトピック: [IBM eXtremeIO \(XIO\) の構成](#)

トランザクション処理の概要

WebSphere® eXtreme Scale クライアント は、データとの相互作用のメカニズムとしてトランザクションを使用します。

Java

Java™ アプリケーションにおけるトランザクション処理

データとの相互作用のために、アプリケーション内のスレッドは、独自のセッションを必要とします。アプリケーションがスレッド上で ObjectGrid を使用する必要がある場合、ObjectGrid.getSession メソッドの 1 つを呼び出してセッションを取得します。このセッションを使用すると、アプリケーションは ObjectGrid マップに保管されているデータの処理を行うことができます。

アプリケーションが Session オブジェクトを使用する場合、そのセッションはトランザクションのコンテキスト内に必要があります。Session オブジェクトに対する begin メソッド、commit メソッド、および rollback メソッドにより、トランザクションは、開始してコミット、あるいは開始してロールバックを行います。また、アプリケーションは自動コミット・モードで稼働することもできます。このモードでは、マップに対して操作が実行されるたびに、Session は自動的にトランザクションを開始してコミットします。自動コミット・モードでは、複数の操作を単一のトランザクションにグループ化することができません。複数の操作のバッチを作成して単一のトランザクションにした場合、自動コミット・モードは低速のオプションとなります。ただし、単一の操作しか含まないトランザクションの場合は、自動コミット・モードの方が速いオプションとなります。

アプリケーションがセッションを終了したときは、オプションの Session.close() メソッドを使用してセッションを閉じます。セッションを閉じると、セッションがヒープから解放され、getSession() メソッドの後続の呼び出しの再利用が可能となるため、パフォーマンスが向上します。

.NET

.NET アプリケーションにおけるトランザクション処理

データとの相互作用のために、アプリケーション内の各スレッドは、独自のトランザクション・オブジェクトを必要とします。アプリケーションでスレッドに対して IGrid インターフェースを使用するには、次のいずれかのメソッドを呼び出します。

- IGrid.GetGridMapPessimisticAutoTx
- IGrid.GetGridMapPessimisticTx

これらのメソッドを呼び出すと、固有のトランザクション・オブジェクトを持つ IGridMap オブジェクトを取得します。この IGridMap オブジェクトを使用すると、アプリケーションは IGrid マップに保管されているデータの処理を行うことができます。アプリケーションが IGridMapPessimisticTx オブジェクトを使用する場合、そのデータ・グリッド操作はトランザクションのコンテキスト内に必要があります。アプリケーションは、IGridTransaction オブジェクトに対する begin、commit、および rollback メソッドを使用して、トランザクションを開始してコミットしたり、開始してロールバックしたりします。また、アプリケーションは自動コミット・モードで稼働することもできます。このモードでは、マップに対して操作が実行されるたびに、IGridMapPessimisticAutoTx は自動的にトランザクションを開始してコミットします。自動コミット・モードでは、複数の操作を単一のトランザクションにグループ化することができません。複数の操作のバッチを作成して単一のトランザクションにした場合、自動コミット・モードは低速のオプションとなります。ただし、単一の操作しか含まないトランザクションの場合は、自動コミット・モードの方が速いオプションとなります。

アプリケーションが IGridMap インスタンスを終了したときは、IGridMap オブジェクトを破棄してください。このオブジェクトを破棄すると、関連するトランザクション・オブジェクトが閉じられます。その結果、GetGridMapPessimisticAutoTx メソッドおよび GetGridMapPessimisticTx メソッドの後続の呼び出しで既存の空きトランザクション・オブジェクトを再利用でき、これによってパフォーマンスが向上します。

トランザクション

トランザクションには、データ保管および操作に関して多くの利点があります。トランザクションを使用すれば、同時変更からデータ・グリッドを保護したり、複数の変更を 1 つの並行ユニットとして適用したり、データを複製したり、変更に対するロックのライフサイクルを実装したりすることができます。

ロック・ストラテジー

ロック・ストラテジーには、ペシミスティック、オプティミスティック、およびロックなしがあります。

ロック・タイプ

ペシミスティック・ロックおよびオプティミスティック・ロックを使用する場合、整合性を保つために、共有 (S) ロック、アップグレード可能 (U) ロック、および排他的 (X) ロックが使用されます。ペシミスティック・ロックを有効にしている場合、ロックとその動作について理解することが重要です。オプティミスティック・ロックではロックを保持しません。ロックにはライフサイクルがあります。いろいろなタイプのロックがさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

デッドロック

2つのトランザクションが同じキャッシュ・エントリを更新しようとするするとデッドロックが起こることがあります。

データ・アクセスおよびトランザクション

WebSphere eXtreme Scale クライアント はトランザクションを使用します。アプリケーションがデータ・グリッドへの接続を獲得したら、ユーザーはデータ・グリッド内のデータにアクセスし、データを操作することができます。

トランザクション分離

3つのトランザクション分離レベル (反復可能読み取り、読み取りコミット済み、および読み取りアンコミット) のうち 1つを使用して、各キャッシュ・マップ内の整合性を保つロック・セマンティクスを調整することができます。

Java

2 フェーズ・コミットとエラー・リカバリー

2 フェーズ・コミット・プロトコルは、分散トランザクションに関与するすべての区画を、そのトランザクションをコミットするかロールバックするかに基づいて調整します。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

トランザクション

トランザクションには、データ保管および操作に関して多くの利点があります。トランザクションを使用すれば、同時変更からデータ・グリッドを保護したり、複数の変更を1つの並行ユニットとして適用したり、データを複製したり、変更に対するロックのライフサイクルを実装したりすることができます。

トランザクションが開始すると、WebSphere® eXtreme Scale クライアントは、そのトランザクションが使用するキーと値のペアの現在の変更またはコピーを保持する特別な差分マップを割り振ります。通常、キーと値のペアにアクセスすると、アプリケーションがその値を受け取る前に、値のコピーが作成されます。Java™ アプリケーションでは、この差分マップが挿入、更新、取得、除去などの操作によるすべての変更を追跡します。.NET アプリケーションでは、この差分マップが追加操作、置換操作、取得操作、および除去操作による変更を追跡します。キーは不変のものに見なされているため、コピーされません。トランザクションがロールバックされると、差分マップ情報が破棄され、エントリーに対するロックが解除されます。トランザクションをコミットすると、変更がマップに適用され、ロックが解除されま

Java Java アプリケーションで ObjectTransformer オブジェクトが指定されると、このオブジェクトが値をコピーするために使用されます。トランザクションがオプティミスティック・ロックを使用している場合は、トランザクションのコミット時に、以前の値のイメージも比較のために追跡されます。

Java Java アプリケーションでオプティミスティック・ロックが使用されている場合は、WebSphere eXtreme Scale クライアントが値の変更前イメージ・バージョンとマップ内の値を比較します。トランザクションをコミットするには、これらの値が一致している必要があります。こうした比較によって複数バージョンのロック体系が可能になりますが、トランザクションがそのエントリーにアクセスすると、代わりに2つのコピーが作成されます。すべての値が再度コピーされ、新しいコピーがマップに保管されます。WebSphere eXtreme Scale クライアントは、コミット後に値へのアプリケーション参照を変更するアプリケーションから自身を保護するために、このコピーを実行します。

情報の複数のコピーを使用しないようにできます。アプリケーションは、並行性を制限する代償としてオプティミスティック・ロックの代わりにペシミスティック・ロックを使用することで、コピーを節約できます。コミット後に値を変更しないことにアプリケーションが同意すれば、コミット時の値のコピーも回避することができます。

.NET 注: .NET アプリケーションはペシミスティック・ロックのみをサポートします。

トランザクションの利点

トランザクションを使用するのは、以下の理由からです。

トランザクションを使用して、以下の操作を行うことができます。

- 例外が発生した場合や、ビジネス・ロジックにより状態変更を元に戻す必要がある場合に、変更をロールバックします。
- コミット時に複数の変更をアトミック単位で適用する
- データに対するロックの保持および解除を行い、コミット時に複数の変更をアトミック単位で適用します。
- 同時変更からスレッドを保護します。
- 変更に対するロックのライフサイクルを実装します。
- アトミック単位のレプリカ生成をします。

トランザクション・サイズ

トランザクションは、特にレプリカ生成の場合には、大きいほど効果的です。ただし、トランザクションが大きくなると、エントリーのロックの保持時間が長くなるため、並行性に悪影響を及ぼすことがあります。大きなトランザクションを使用すると、レプリカ生成のパフォーマンスが向上する場合があります。このパフォーマンスの向上は、マップを事前にロードする場合には重要です。さまざまなバッチ・サイズで実験を行い、使用するシナリオに最適なサイズを判別してください。

自動コミット・モード

Java アクティブに始動されたトランザクションがない場合は、アプリケーションが ObjectMap オブジェクトとの対話を行うと、アプリケーションの代わりに自動的に開始およびコミット操作が行われます。この自動的な開始およびコミット操作は役に立ちますが、ロールバックおよびロックが有効に機能する妨げとなります。トランザクションのサイズが小さすぎると、同期レプリカ生成スピードに影響します。エンティティ・マネージャー・アプリケーションを使用している場合は、自動コミット・モードは使用しないでください。その理由は、EntityManager.find メソッドで検索されたオブジェクトが、そのメソッドが戻されると同時に管理不能となり、使用不可となるためです。

.NET .NET アプリケーションでは、GridMapPessimisticAutoTx マップ・インターフェースが等価の自動開始およびコミット操作を提供します。制約は同じです。つまり、ロールバックとロックが正しく機能せず、同期レプリカ生成スピードが低下します。

Java

Java EE トランザクション統合

WebSphere eXtreme Scale クライアントは、リモート・データ・グリッドへのクライアント接続とローカル・トランザクション管理の両方をサポートする Java Connector Architecture (JCA) 1.5 準拠リソース・アダプターを含みます。サーブレットなどの Java Platform、Enterprise Edition (Java EE) アプリケーション、JavaServer Pages (JSP) フ

イル、および Enterprise JavaBeans (EJB) コンポーネントは、標準 javax.resource.cci.LocalTransaction インターフェースまたは WebSphere eXtreme Scale クライアント セッション・インターフェースを使用して、WebSphere eXtreme Scale クライアント トランザクションを区分的ことができます。

WebSphere Application Server で最終参加者サポートが有効になっている状態でアプリケーションを実行しているときには、WebSphere eXtreme Scale クライアント トランザクションを、他の 2 フェーズ・コミット・トランザクション・リソースと共にグローバル・トランザクションに登録することができます。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[.NET アプリケーションにおけるトランザクションのためのプログラミング](#)

ロック・ストラテジー

ロック・ストラテジーには、ペシミスティック、オプティミスティック、およびロックなしがあります。

ロックはトランザクションに 束縛されます。以下のロック設定を指定することができます。

Java ロックなし

ロック設定を使用しないと、実行は最速になります。読み取り専用データを使用していれば、ロックは必要ない場合があります。

制約事項: ロックなしストラテジーを使用するように構成された BackingMap は、複数区画トランザクションに関与できません。

Java .NET ペシミスティック・ロック

エントリーに対するロックを取得し、コミット時までそのロックを保持します。このロック戦略は、スループットを低下させる代わりに、優れた一貫性を提供します。

Java オプティミスティック・ロック

トランザクションがタッチするすべてのレコードの以前のイメージを取得して、トランザクションのコミット時に、そのイメージと現在のエントリーの値を比較します。エントリーの値が変更された場合、そのトランザクションはロールバックします。コミット時までロックは保持されません。このロック戦略は、ペシミスティック戦略よりも並行性において優れていますが、トランザクション・ロールバックのリスクがあり、エントリーのコピーを作成するためにメモリーを消費します。

重要: WebSphere® eXtreme Scale Client for .NET でクライアント・アプリケーションを使用している場合は、ペシミスティック・ロックのみがサポートされます。

ロック・マネージャー

ロック・ストラテジーに OPTIMISTIC または PESSIMISTIC が使用されている場合は、BackingMap に対してロック・マネージャーが作成されます。ロック・マネージャーは、ハッシュ・マップを使用して、1 つ以上のトランザクションによってロックされるエントリーを追跡します。ハッシュ・マップに多くのマップ・エントリーが存在する場合、ロック・バケットが多いほど、パフォーマンスが良好になる可能性が高くなります。バケット数が増えるにつれて、Java™ 同期の衝突のリスクは下がります。またロック・バケットを増やすことが、並行性の増大につながります。前の例では、特定の BackingMap インスタンスに使用するロック・バケットの数をアプリケーションでどのように設定できるかを示しています。

Java .NET

ペシミスティック・ロック

ペシミスティック・ロック・ストラテジーは、キャッシュ・エントリーに対してロックを取得するため、データが頻繁に変更される場合に使用するようにしてください。キャッシュ・エントリーが読み取られる場合は、必ずロックが取得され、トランザクションが完了するまでロックが条件付きで保持されます。ロックによっては、セッションのトランザクション分離レベルを使用して、その期間を調整することができます。

ほかのロック・ストラテジーが可能でない場合は、マップの読み書きにペシミスティック・ロック・ストラテジーを使用します。ObjectGrid マップがペシミスティック・ロック・ストラテジーを使用するように構成されている場合、トランザクションが最初に BackingMap からのエントリーを取得すると、マップ・エントリーのペシミスティック・トランザクション・ロックが取得されます。ペシミスティック・ロックは、アプリケーションがトランザクションを完了するまでは保留されます。通常の場合、ペシミスティック・ロック・ストラテジーは、以下の状態で使用されます。

- BackingMap がで構成され、バージョン管理情報が使用可能でない場合。

制約事項: ローダー・プラグインで構成された BackingMap は、マルチ区画トランザクションのマップを読み取ることはできますが、書き込むことはできません。

- BackingMap が、並行処理制御について WebSphere eXtreme Scale クライアント からの支援を必要とするアプリケーションによって直接使用されている場合。
- バージョン管理情報は使用できるが、更新トランザクションがバックキング・エントリー上で頻繁に衝突し、その結果、オプティミスティック更新が失敗する場合。

ペシミスティック・ロック・ストラテジーは、パフォーマンスおよびスケーラビリティに最大の影響があります。したがって、このストラテジーは、他のロック・ストラテジーを使用できない場合に、マップの読み書きにのみ使用するようにしてください。例えば、こうした状態には、オプティミスティック更新の失敗が頻繁に発生する場合や、オプティミスティック障害からのリカバリーをアプリケーションが処理するには難しい場合が含まれます。

ペシミスティック・ロックを使用しているときには、ロック・メソッドを使用して、データ値を返すことなくデータすなわちキーをロックすることができます。メソッドのリストと、それらのメソッドが獲得するロックの種類については、[ロック・タイプ](#)を参照してください。

Java

オプティミスティック・ロック

デフォルトのロック・ストラテジーは、OPTIMISTIC です。データの変更が頻繁でない場合は、このオプティミスティック・ロックを使用します。データがキャッシュから読み取られ、トランザクションにコピーされる間、ロックは短期間だけ保持されます。トランザクション・キャッシュがメイン・キャッシュと同期されると、更新されたあらゆるキャッシュ・オブジェクトが元のバージョンに対してチェックされます。チェックが失敗すると、トランザクションはロールバックされ、OptimisticCollisionException 例外となります。

オプティミスティック・ロック・ストラテジーでは、並行して実行中である 2 つのトランザクションが同じマップ・エントリーの更新を試みることはないということが前提となっています。複数のトランザクションが同一のマップ・エントリーを同時に更新することはないと考えられるので、トランザクションのライフサイクルでロックは保持されません。オプティミスティック・ロック・ストラテジーは通常、以下の場合に使用されます。

- BackingMap が構成され、バージョン管理情報が使用可能である場合。

制約事項: ロダー・プラグインで構成された BackingMap は、マルチ区画トランザクションのマップを読み取ることにはできますが、書き込むことはできません。

- BackingMap のほとんどのトランザクションが読み取り操作である場合。BackingMap に対するエントリーの挿入、更新、または除去操作は、あまり行われません。
- BackingMap は、読み取りと比べてより頻繁に挿入、更新、または除去されるが、トランザクションは同じマップ・エントリー上でほとんど衝突しない場合。

ペシミスティック・ロック・ストラテジーと同様に、ObjectMap インターフェース上のメソッドは、WebSphere eXtreme Scale クライアント が、アクセス中のマップ・エントリーのロック・モードを自動的に取得する方法を決定します。ただし、ペシミスティック・ストラテジーとオプティミスティック・ストラテジーの間には、以下のような違いがあります。

- ペシミスティック・ロック・ストラテジーと同様に、メソッドの呼び出しの際、get メソッドおよび getAll メソッドによって S ロック・モードが取得されます。しかし、オプティミスティック・ロックを使用すると、S ロック・モードはトランザクションが完了するまで保留されません。代わりに、S ロック・モードはメソッドがアプリケーションに戻す前に保留解除されます。ロック・モードの取得の目的は、WebSphere eXtreme Scale クライアント が、その他のトランザクションからのコミット済みデータのみが現行トランザクションに可視となるように保証できるようにすることです。WebSphere eXtreme Scale クライアント がそのデータがコミット済みであることを確認した後で、S ロック・モードは保留解除されます。コミット時に、オプティミスティック・バージョン管理チェックが実行され、現行トランザクションがその S ロック・モードを保留解除した後で、マップ・エントリーを変更したトランザクションが他にないことが確認されます。更新、無効化、または削除される前にマップからエントリーがフェッチされない場合、WebSphere eXtreme Scale クライアント ランタイムによって、暗黙的にマップからエントリーがフェッチされます。この暗黙的な get 操作は、エントリーの変更が要求された時点における現行値を取得するために実行されます。
- ペシミスティック・ロック・ストラテジーとは異なり、getForUpdate メソッドと getAllForUpdate メソッドは、オプティミスティック・ロック・ストラテジーが使用された場合には、get メソッドと getAll メソッドと同様に処理されます。つまり、S ロック・モードはメソッドの開始時に取得され、S ロック・モードはアプリケーションに戻る前に保留解除されます。

他のすべての ObjectMap メソッドは、ペシミスティック・ロック・ストラテジーの場合と同様に扱われます。commit メソッドが呼び出されると、挿入、更新、除去、タッチ、または無効化の対象となるマップ・エントリーについては X ロック・モードが獲得されます。X ロック・モードは、トランザクションがコミット処理を完了するまで保持されません。

オプティミスティック・ロック・ストラテジーでは、並行して実行中のトランザクションが同じマップ・エントリーを更新することはないと想定します。この想定から、トランザクションの存続期間中、ロック・モードを保留する必要はありません。これは、複数のトランザクションがマップ・エントリーを並行して更新するとは考えられないためです。しかし、ロック・モードが保留されなかったため、現行トランザクションがその S ロック・モードを保留解除した後で、別の並行トランザクションがマップ・エントリーを更新する可能性があります。

この可能性に対処するため、WebSphere eXtreme Scale クライアント はコミット時に X ロックを取得し、オプティミスティック・バージョン管理チェックを行って、現行トランザクションが BackingMap からマップ・エントリーを読み取って以降、他にマップ・エントリーを変更したトランザクションがないことを確認します。別のトランザクションがマップ・エントリーを変更した場合、バージョン・チェックは失敗し、OptimisticCollisionException 例外が発生します。この例外により、現行トランザクションが強制的にロールバックされ、トランザクション全体がアプリケーションによって再試行されることとなります。オプティミスティック・ロック・ストラテジーは、マップがほとんど既読で、同じマップ・エントリーに対する更新が起こる可能性が低い場合に便利です。

Java

ロックなし

データがまったく更新されないか、静止期間のみに更新されるため、ロックが必要ない場合は、NONE ロック・ストラテジーを使用すれば、ロックを使用不可にすることができます。このストラテジーは、ロック・マネージャーを必要としないため、非常に高速です。NONE ロック・ストラテジーは、ルックアップ表または読み取り専用のマップの場合に理想的です。

BackingMap がロックなしストラテジーを使用するよう構成されている場合、マップ・エントリーのトランザクション・ロックは獲得されません。

制約事項: ロックなしストラテジーを使用するよう構成された BackingMap は、複数区画トランザクションに関与できません。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[ロック・ストラテジーの構成](#)

[Java](#) [Java アプリケーションにおけるロックの構成と実装](#)

[.NET](#) [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

[.NET](#) [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

例: [flush メソッドのロック順序](#)

ロック・タイプ

ペシミスティック・ロックおよびオプティミスティック・ロックを使用する場合、整合性を保つために、共有 (S) ロック、アップグレード可能 (U) ロック、および排他的 (X) ロックが使用されます。ペシミスティック・ロックを有効にしている場合、ロックとその動作について理解することが重要です。オプティミスティック・ロックではロックを保持しません。ロックにはライフサイクルがあります。いろいろなタイプのロックがさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

共有ロック、アップグレード可能ロック、および排他的ロック

アプリケーションがマップ・プログラミング・インターフェースのメソッドをどれか呼び出すと、WebSphere® eXtreme Scale クライアント は自動的に、アクセス対象のマップ・エンタリーに対するロックを獲得しようと試みます。

Java Java アプリケーションでは、アプリケーションが索引に対して検索メソッドを使用したときや照会を行ったときにも、ロックが獲得されます。

ペシミスティック・ロックを使用している場合、ロック・メソッドを使用して、データ値を返すことなくキーをロックすることができます。ロック・メソッドにより、データ・グリッド内のキーをロックするか、またはキーをロックして値がデータ・グリッド内に存在するかどうかを確認することができます。

LockMode は列挙型で以下に示す値を取ることができ、ここでロックしたいキーを指定できます。

- **Java** SHARED、UPGRADABLE、および EXCLUSIVE
- **.NET** Shared、Upgradable、Exclusive

WebSphere eXtreme Scale クライアント は、マップ・プログラミング・インターフェースでアプリケーションが呼び出すメソッドに基づいて以下のロック・モードを使用します。

S ロック

マップ・エンタリーのキーに対する共有ロック・モード。S ロックが保持されている期間は、使用されるトランザクション分離レベルによります。S ロック・モードでは、同一キーに対して S ロック・モードまたはアップグレード可能ロック (U ロック) モードを取得しようとする トランザクション間での並行処理が許されますが、同一キーに対して排他的ロック (X ロック) モードを取得しようとする他の トランザクションはブロックされます。

U ロック

マップ・エンタリーのキーに対するアップグレード可能ロック・モード。U ロックは、トランザクションが完了するまで保留になります。U ロック・モードでは、同一キーに対して S ロック・モードを取得する トランザクション間での並行処理が許されますが、同一キーに対して U ロック・モードまたは X ロック・モードを取得しようとする他の トランザクションはブロックされます。

X ロック

マップ・エンタリーのキーに対する排他的ロック・モード。X ロックは、トランザクションが完了するまで保留になります。X ロック・モードでは、1 つのトランザクションのみが所定のキー値のマップ・エンタリーを挿入、更新、または除去することになります。X ロックは、同一キーに対する S、U、または X ロック・モードを取得しようとする他のすべてのトランザクションをブロックします。

S ロック・モードは U ロック・モードより弱いロック・モードです。つまり、複数のトランザクションが同じマップ・エンタリーにアクセスするとき、S ロック・モードのほうがより多くのトランザクションを並行して実行できます。U ロック・モードは、S ロック・モードよりも少し強力です。それは、U ロック・モードまたは X ロック・モードのどちらかを要求している他のトランザクションをブロックするためです。S ロック・モードは、X ロック・モードを要求しているその他のトランザクションのみをブロックします。この小さな差が、一部のデッドロックの発生を防止するには重要です。X ロック・モードは、最強のロック・モードです。これは、同一のマップ・エンタリーに対して S、U、または X ロックのモードを取得しようとしているその他すべてのトランザクションをブロックするためにです。X ロック・モードは、1 つのトランザクションのみがマップ・エンタリーを挿入、更新、または除去できるようにするとともに、複数のトランザクションが同一のマップ・エンタリーの更新を試みている場合に更新が失われないようにします。

次の表を参照して、これらのロック・モード値と既存のメソッドの振る舞いとの関係を理解してください。

表 1. LockMode 値と既存の等価メソッド

ロック・モード (Java / .NET)	等価の Java メソッド	等価の .NET メソッド
SHARED / Shared	ObjectMap インターフェース上の get メソッドおよび getAll メソッド、索引メソッド、および照会	Get() メソッド、GetAndLock(Key, LockMode.Shared) メソッド、GetAndLockAll(KeyList, LockMode.Shared) メソッド、GetAll メソッド
UPGRADABLE /Upgradable	getForUpdate()、getAllForUpdate()	GetAndLock(Key, LockMode.Updgradable)、GetAndLockAll(KeyList, LockMode.Upgradable)

EXCLUSIVE / Exclusive	<p>getNextKey() メソッド、commit() メソッド、put メソッド、putAll メソッド、remove メソッド、removeAll メソッド、insert メソッド、update メソッド、および touch メソッド、グローバル invalidate メソッドおよびグローバル invalidateAll メソッド。(ローカルの invalidate メソッドおよび invalidateAll メソッドの場合、ロックは獲得されません。これは、どの BackingMap エントリーもローカル invalidate メソッド呼び出しによっては無効化されないからです。)</p> <p>Java  注: upsert メソッドおよび upsertAll メソッドが ObjectMap の put メソッドおよび putAll メソッドに取って代わります。データ・グリッド内のエントリーがキーと値をグリッドに配置しなければならないことを BackingMap とローダーに知らせるには、upsert メソッドを使用します。BackingMap とローダーは、insert または update のいずれかを行って、値をデータ・グリッドおよびローダーに配置します。アプリケーション内で upsert API を実行すると、ローダーは UPSERT LogElement タイプを取得します。これにより、ローダーは、insert や update を使用する代わりにデータベースの merge 呼び出しまたは upsert 呼び出しを行うことができます。</p>	<p>Commit()、Add、AddAll、Put、PutAll、Remove、RemoveAll、Replace、ReplaceAll、Touch、TouchAll、Invalidate、InvalidateAll</p> <p>注: Put メソッドおよび PutAll メソッドは Java の upsert および upsertAll メソッドと等価です。</p>
-----------------------	--	--

次表は、ロック・モードの互換性マトリックスです。前述のロック・モードをまとめたもので、互いに互換性のあるロック・モードはいずれかを調べる場合に使用してください。このマトリックスを読み取る場合、マトリックスの行は既に認可されているロック・モードを表します。列は、別のトランザクションによって要求されたロック・モードを表します。列に「あり」と表示されている場合は、別のトランザクションによって要求されたロック・モードは認可されています。これは、既に認可されているロック・モードと互換性があるためです。「なし」は、ロック・モードの互換性がないことを表します。その他のトランザクションは、最初のトランザクションが保持しているロックを解放するのを待たなければなりません。

表 2. ロック・モードの互換性マトリックス

ロック	ロック・タイプ S (共有)	ロック・タイプ U (アップグレード可能)	ロック・タイプ X (排他的)	強さ
S (共有)	はい	はい	いいえ	最弱
U (アップグレード可能)	はい	いいえ	いいえ	通常
X (排他的)	いいえ	いいえ	いいえ	最強

親トピック: [トランザクション処理の概要](#)

関連タスク:

[ロック・ストラテジーの構成](#)

Java [Java アプリケーションにおけるロックの構成と実装](#)

.NET [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

.NET [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

例: [flush メソッドのロック順序](#)

デッドロック

2つのトランザクションが同じキャッシュ・エントリを更新しようとするするとデッドロックが起こることがあります。

古典的なデッドロックの例

ロック・モード要求の以下のシーケンスについて検討します。

1. Xロックは、トランザクション 1 の key1 に対して認可されています。
2. Xロックは、トランザクション 2 の key2 に対して認可されています。
3. トランザクション 1 によって要求された、key2 に対する X ロック (トランザクション 1 はブロックされ、トランザクション 2 が所有するロックを待機しています。)
4. トランザクション 2 によって要求された、key1 に対する X ロック (トランザクション 2 はブロックされ、トランザクション 1 が所有するロックを待機しています。)

上記のシーケンスは、2つのトランザクションからなる古典的なデッドロックの例です。2つのトランザクションが複数のロックを取得しようとし、各トランザクションは異なる順序でロック取得します。このデッドロックを防止するには、各トランザクションが複数ロックを同じ順序で獲得しなければなりません。

Java

オプティミスティック・ロックによるデッドロック防止

オプティミスティック・ロック・ストラテジーが使用され、ObjectMap インターフェースの flush メソッドがアプリケーションによって絶対に使用されない場合は、ロック・モードがトランザクションによって要求されるのはコミット・サイクル中のみです。コミット・サイクル中、WebSphere® eXtreme Scale クライアントは、ロックする必要があるマップ・エントリのキーを決定し、キー・シーケンスのロック・モードを要求します (決定論的振る舞い)。WebSphere eXtreme Scale クライアントは、この方法を使用して多くの古典的デッドロックを防止します。

しかし、eXtreme Scale がすべてのデッドロック・シナリオを防止するわけでも、防止できるわけでもありません。アプリケーションが考慮する必要があるシナリオがいくつか存在します。以下は、アプリケーションが注意し、予防アクションを取らなければならないシナリオです。

1つのシナリオは、ロック待ちタイムアウトが発生するのを待たなくとも WebSphere eXtreme Scale クライアントがデッドロックを検出できる場合です。このシナリオが起こると、com.ibm.websphere.objectgrid.LockDeadlockExceptionException が発生します。次のコード例をご覧ください。

```
Session sess = ...;
ObjectMap person = sess.getMap("PERSON");
sess.begin();
Person p = (IPerson)person.get("Lynn");
// Lynn had a birthday; so make her 1 year older.
p.setAge( p.getAge() + 1 );
person.put( "Lynn", p );
sess.commit();
```

同じシナリオで、次のコード例のように upsert メソッドを使用することができます。

```
Session sess = ...;
ObjectMap person = sess.getMap("PERSON");
sess.begin();
Person p = (IPerson)person.get("Lynn");
// Lynn had a birthday; so make her 1 year older.
p.setAge( p.getAge() + 1 );
person.upsert( "Lynn", p );
sess.commit();
```

この場合、2つのトランザクションが Lynn person オブジェクトの年齢を更新しようとしています。この状態では、person.get("Lynn") メソッド呼び出しの結果として両方のトランザクションが PERSON マップの Lynn エントリに対して S ロック・モードを保持します。person.put ("Lynn", p) メソッド呼び出しの結果として、両方のトランザクションは S ロック・モードを X ロック・モードに格上げしようとしています。両方のトランザクションがブロックされ、そのどちらのトランザクションも、もう一方のトランザクションがその所有する S ロック・モードを解放するのを待っています。その結果、2つのトランザクションの間に循環待ち状態が存在することになってデッドロックが発生します。循環待ち状態は、複数のトランザクションが同一のマップ・エントリに対して持っているロックを弱いモードからより強いモードに昇格させようとしたときに発生します。このシナリオでは、LockTimeoutException 例外ではなく、LockDeadlockException 例外になります。

Java アプリケーションでは、ペシミスティック・ロック・ストラテジーではなくオプティミスティック・ロック・ストラテジーを使用することによって、上記の例の LockDeadlockException 例外を防止することができます。オプティミスティック・ロック・ストラテジーの使用は、マップが主として読み取りで、マップの更新がまれにしか行われない場合、推奨される解決策です。

ペシミスティック・ロックによるデッドロック防止

.NET 重要: .NET アプリケーションはペシミスティック・ロックのみをサポートします。次のセクションでは、Java メソッド名について説明します。ただし、.NET メソッド名にも当てはまります。これらのメソッドは、Get、GetAndLock、GetAndLockAll、Put、Add、Replace、および Remove です。

ペシミスティック・ロック・ストラテジーを使用する場合にデッドロックを防止するには、次のようにします。

- READ_COMMITTED トランザクション分離レベルを使用します。READ_COMMITTED トランザクション分離レベルは、get メソッドによって獲得された S ロックがトランザクション完了まで保持されることがないようにします。キーがトランザクション・キャッシュで無効化されない場合、反復可能読み取りは引き続き保証されます。
- get メソッドではなく代替の get メソッドを使用します。
 - **Java** getForUpdate メソッドを使用します。
 - **.NET** GetAndLock または GetAndLockAll メソッドを使用します。

最初のトランザクションは getForUpdate メソッドを呼び出して、S ロックではなく U ロック・モードを獲得します。このロック・モードにより、2 番目のトランザクションは、getForUpdate メソッドを呼び出したときにブロックされます。U ロック・モードが付与されるトランザクションは 1 つです。2 番目のトランザクションはブロックされるので、マップ・エントリーに対するロック・モードを何も所有しません。最初のトランザクションは、最初のトランザクションからの put メソッド呼び出しの結果として、U ロック・モードから X ロック・モードへの格上げをしようとしたときに、ブロックしません。この働きは、U ロック・モードがアップグレード可能ロック・モードと呼ばれる理由を説明しています。最初のトランザクションが完了すると、2 番目のトランザクションのブロックが解除され、U ロック・モードが付与されます。ペシミスティック・ロック・ストラテジーが使用される場合は、アプリケーションは、get メソッドの代替として getForUpdate メソッドを使用することによりロック昇格デッドロック・シナリオを回避することができます。

重要: この解決策は、読み取り専用トランザクションがマップ・エントリーを読み取るのを妨げません。読み取り専用トランザクションは get メソッドを呼び出します。読み取り専用トランザクションが put、insert、update、または remove メソッドを呼び出すことはありません。並行性は通常の get メソッドが使用されたときと同程度です。唯一、並行性が低減するのは、複数のトランザクションによって同一のマップ・エントリーに対して getForUpdate メソッドが呼び出されることです。

あるトランザクションが複数のマップ・エントリーに対して getForUpdate メソッドを呼び出す場合、各トランザクションによって確実に U ロックが同じ順序で取得されるように注意しなければなりません。例えば、最初のトランザクションがこのメソッドを 2 回 (キー 1、次いでキー 2 に対して) 呼び出し、別の並行トランザクションもこのメソッドを同じ 2 つのキーに対して (ただし逆順で) 呼び出すとします。このシーケンスにより、古典的なデッドロックが発生します。複数ロックが異なるトランザクションによって異なる順序で獲得されるためです。アプリケーションでは引き続き、複数のマップ・エントリーにアクセスするどのトランザクションもキー・シーケンスに従い、デッドロックが発生しないようにする必要があります。U ロックはコミット時ではなく、getForUpdate メソッドが呼び出される時に獲得されるので、WebSphere eXtreme Scale クライアントは、コミット・サイクル中に行われるようにロック要求を順序付けることはできません。アプリケーションは、このケースではロックの順序付けを制御する必要があります。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[ロック・ストラテジーの構成](#)

Java [Java アプリケーションにおけるロックの構成と実装](#)

.NET [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

.NET [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

[例: flush メソッドのロック順序](#)

データ・アクセスおよびトランザクション

WebSphere® eXtreme Scale クライアント はトランザクションを使用します。アプリケーションがデータ・グリッドへの接続を獲得したら、ユーザーはデータ・グリッド内のデータにアクセスし、データを操作することができます。

Java

Java アプリケーションにおけるトランザクション

Java アプリケーションを使用して、分散インスタンスへのクライアント接続を確立することができます。

アプリケーションがセッションとの対話を行う場合、そのセッションはトランザクションのコンテキストの中にある必要があります。トランザクションはセッション・オブジェクトの `Session.begin`、`Session.commit`、および `Session.rollback` メソッドを使用して、開始されたり、コミットまたはロールバックされます。アプリケーションは、自動コミット・モードで作業を行うこともできます。このモードの場合、アプリケーションがマップとの対話を行うたび、セッションが自動的に開始し、トランザクションをコミットします。ただし、自動コミット・モードは低速です。

Java アプリケーション内のスレッドには、独自のセッションが必要です。アプリケーションがスレッド上で `ObjectGrid` を使用するようにしたい場合は、`getSession` メソッドの 1 つを呼び出してセッションを取得します。アプリケーションがそのセッションを終了した後、`Session.close()` メソッドを呼び出します。このメソッドはセッションを閉じて、そのセッションをプールに返し、そのセッションのリソースを解放します。セッションを閉じるのはオプションですが、そうすると、`getSession()` メソッドに対する後続の呼び出しのパフォーマンスが向上します。アプリケーションが、Spring のような依存性注入フレームワークを使用する場合、必要なときにセッションをアプリケーション Bean に注入することができます。

セッションを取得した後、アプリケーションは `ObjectGrid` 内のマップに保管されたデータにアクセスできます。マップ・ベースの API は `Session.getMap` メソッドを使用して取得されます。

.NET

.NET アプリケーションにおけるトランザクション

.NET アプリケーションでは、スレッドごとに別々の `IGridMapPessimisticTx` オブジェクトまたは `IGridMapPessimisticAutoTx` オブジェクトを持つ必要があります。 `IGridMapPessimisticTx` オブジェクトでは、`Transaction` プロパティを使用してトランザクションの開始、コミット、またはロールバックが明示的に行われます。 `IGridMapPessimisticAutoTx` オブジェクトでは、トランザクションの開始、コミット、およびロールバック操作が自動的に行われます。これらのオブジェクトの 1 つを取得すると、アプリケーションはデータ・グリッド内の保管データにアクセスできるようになります。

トランザクション使用のロジック

トランザクションは遅く感じられるかもしれません。以下の理由からトランザクションを使用する必要があります。

1. 例外が発生した場合や、状態変更を元に戻すことをビジネス・ロジックが必要とする場合に、変更のロールバックが可能であること。
2. 1 つのトランザクションの存続時間中にデータに対するロックの保持と解除を行うことで、一連の変更がアトミックに行われる、つまり、データに対してすべての変更を行うか、何も変更しないかにできること。
3. 複製のアトミックな単位を生成できること。
4. **Java** 複数の区画を更新できること。

どの程度のトランザクション・サポートが必要であるかをカスタマイズすることができます。アプリケーションはロールバック・サポートおよびロックをオフにできますが、これはアプリケーションに犠牲を強いることになります。アプリケーションはこれらのフィーチャーの欠如に対処する必要があります。アプリケーションがトランザクション・サポートをどのように管理できるかの例を以下に示します。

- **Java** 動的マップ・ロック・ストラテジーを `NONE` に構成すると、アプリケーションはロックをオフにできます。詳しくは、[Java API を使用した動的マップの作成](#)を参照してください。このストラテジーは高速ですが、並行トランザクションが互いに保護されずに、同じデータを変更できるようになります。 `NONE` を使用する場合は、そのアプリケーションが、すべてのロックおよびデータの整合性に対する責任を持つことになります。 `PESSIMISTIC` ロック・ストラテジーしかサポートしていない `WebSphere eXtreme Scale Client for .NET` アプリケーションの場合、このオプションは無効です。

Java `NONE CopyMode` 値を使用して取得されたオブジェクトがアプリケーションによって変更されると、そのオブジェクトのコミット済みのコピーが直接変更されます。このモードでは、トランザクションのロールバックは意味がありません。 `ObjectGrids` での唯一のコピーが変更されます。 `NONE CopyMode` を使用すると処理は速くなりますが、その影響に注意する必要があります。 `NONE CopyMode` を使用するアプリケーションは、トランザクションを決してロールバックしてはなりません。もしアプリケーションがトランザクションをロールバックした場合、索引に変更を反映する更新は行われず、かつ、レプリカ生成がオンにされていても変更は複製されません。

デフォルト値を使用するほうが簡単で、誤りの可能性も低くなります。データ信頼性を犠牲にしてもパフォーマンスを上げたい場合は、意図しない問題を回避するために、アプリケーションは実行内容をよく認識する必要があります。

トランザクションと区画

Java アプリケーションにおけるトランザクションは単一の区画でも複数の区画でも更新できますが、単一区画の更新がデフォルトの動作です。.NET アプリケーションは単一区画の更新しか行えません。

WebSphere eXtreme Scale クライアントに対する複数区画トランザクション・サポートを使用可能にするには、TxCommitProtocol セッション API を使用します。次の 2 つのオプションを使用することができます。

- TxCommitProtocol.ONEPHASE (デフォルト): クライアントからのトランザクションは、複数の区画から読み取ることができますが、単一区画の更新しか行えません。複数区画の更新を試みると失敗します。
- TxCommitProtocol.TWOPHASE: クライアントからのトランザクションは複数の区画を読み取って更新することができます。このトランザクションは、2 フェーズ・コミット・プロトコルを使用して、これらの区画に書き込まれたデータが自動的にコミットまたはロールバックされるようにします。このトランザクションが単一の区画にのみ書き込む場合は、1 フェーズ・コミットメント・プロトコルが使用されます。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[Java](#) [スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクションに書き込むアプリケーションの開発](#)

[Java](#) [Java アプリケーションのためのトランザクション内のデータとの対話](#)

[.NET](#) [.NET アプリケーションのためのトランザクション内のデータとの対話](#)

トランザクション分離

3つのトランザクション分離レベル(反復可能読み取り、読み取りコミット済み、および読み取りアンコミット)のうち1つを使用して、各キャッシュ・マップ内の整合性を保つロック・セマンティクスを調整することができます。

トランザクション分離の概説

トランザクション分離は、1つの操作で行われた変更がどのように他の並行操作に可視になるのかを定義します。

WebSphere® eXtreme Scale クライアントが各キャッシュ・マップ内の整合性を保つために使用するロック・セマンティクスを調整するためのトランザクション分離レベル(反復可能読み取り、読み取りコミット済み、および読み取りアンコミット)を定義することができます。

次のいずれかの方法でトランザクション分離レベルを設定することができます。

- **Java** Session インターフェイスで `setTransactionIsolation` メソッドを使用する。トランザクション分離は、現在進行中のトランザクションがなければ、セッションの存続期間中いつでも変更できます。
- **.NET** IGridTransaction インターフェイスで `TransactionIsolationLevel` プロパティを使用する。

この製品では、共有 (S) ロックが要求および保持される方法を調整することによって、さまざまなトランザクション分離セマンティクスが施行されます。トランザクション分離は、オプティミスティック・ロックを使用するように構成されているマップや、ロックを使用しないように構成されているマップに対しては何の影響もありません。また、アップグレード可能 (U) ロックが獲得されるときにも影響を及ぼしません。

ペシミスティック・ロックでの 反復可能読み取り

反復可能読み取りは、デフォルトのトランザクション分離レベルです。この分離レベルは、ダーティー読み取りおよび反復不能読み取りを防止しますが、幻償読み取りは防止しません。ダーティー読み取りとは、あるトランザクションによって変更されたが、コミットされていないという状態のデータに対して発生する読み取り操作のことです。反復不能読み取りは、読み取り操作実行時に読み取りロックが取得されていない場合に発生する可能性があります。幻償読み取りは、2つの同一の読み取り操作が実行されたが、操作と操作との間にデータに対する更新があったために2つの異なる結果セットが戻される場合に、発生する可能性があります。Java アプリケーションでは、照会または索引を使用しているときに幻償読み取りが発生する可能性があります。これは、ロックがデータの範囲に対して獲得されず、索引または照会基準を満たすキャッシュ・エントリーに対してのみ獲得されるからです。この製品は、すべての S ロックを、ロックを所有するトランザクションが完了するまで保持し続けることによって、反復可能読み取りを実現します。X ロックは、すべての S ロックが解放されるまで認可されないため、S ロックを保持するすべてのトランザクションは、再読み取り時に同じ値を参照することが保証されます。

ペシミスティック・ロックでの読み取りコミット済み

読み取りコミット済みのトランザクション分離レベルを WebSphere eXtreme Scale クライアントで使用できます。この分離レベルは、ダーティー読み取りを防止しますが、反復不能読み取りまたは幻償読み取りを防止しないため、WebSphere eXtreme Scale クライアントは S ロックを引き続き使用してキャッシュ・マップからデータを読み取りますが、すぐにロックを解放します。

ペシミスティック・ロックでの読み取りアンコミット

読み取りアンコミットのトランザクション分離レベルを WebSphere eXtreme Scale クライアントで使用できます。この分離レベルは、ダーティー読み取り、反復不能読み取り、および幻償読み取りを許容します。

親トピック: [トランザクション処理の概要](#)

関連資料:

Java [Java のトランザクション分離の例](#)

2 フェーズ・コミットとエラー・リカバリー

2.5+ 2 フェーズ・コミット・プロトコルは、分散トランザクションに関与するすべての区画を、そのトランザクションをコミットするかロールバックするかに基づいて調整します。

分散データ・グリッドでは、区画は複数の Java™ 仮想マシン (JVM) に分散されます。これらの JVM は複数のシステムに存在することができます。複数の区画に書き込むトランザクションは、複数のシステムに影響を及ぼすトランザクション決定を伴うことがあります。このようなトランザクションが 2 フェーズ・コミット・プロトコルを使用してコミットされると、このコミット・プロセスは、トランザクション全体が保持されるか、またはトランザクションがまったく保持されないようにします。2 フェーズ・コミット・プロセスは、区画、システム、または通信に障害が起こった場合でも、この結果を保証します。第 2 フェーズで障害が起こると、そのエラーが手動で介入できる一定基準を満たしているとき以外は、WebSphere® eXtreme Scale クライアントが障害を自動的に解決しようと試みます。

複数の区画への書き込みが有効になっているトランザクションは 2 フェーズ・コミット・プロトコルを使用します。2 フェーズ・コミット・プロトコルは、すべての区画およびシステムでコミット・プロセスが整合性を保てるようにします。WebSphere eXtreme Scale は、2 フェーズ・コミット・プロセスを制御するコーディネーターとなります。このようなトランザクションに関係する区画を参加者あるいはリソース・マネージャー (RM) と呼びます。コーディネーターは、コミット・プロトコルの第 2 フェーズで、区画の 1 つがトランザクション・マネージャー (TM) として動作するように委任します。TM は、各トランザクションの決定の追跡と、障害が発生した場合のトランザクションのリカバリーを行います。

第 1 フェーズ:

アプリケーションがトランザクションをコミットすると、WebSphere eXtreme Scale クライアントはまず、RM として特定された各区画にコミット準備要求を送ることによって第 1 フェーズを開始します。各区画はトランザクション変更をバックアップ・マップに適用し、データ安全性を確保するためにすべてのロックを保持します。RM は WebSphere eXtreme Scale クライアントに通知します。RM として特定されたすべての区画が成功で応答すると、WebSphere eXtreme Scale クライアントはコミット・プロトコルの第 2 フェーズを開始します。

第 2 フェーズ:

第 1 フェーズで少なくとも 1 つの区画が障害を起こすと、コーディネーターは第 2 フェーズですべての区画をロールバックします。すべての RM 区画が成功で応答すると、WebSphere eXtreme Scale クライアントは区画の 1 つを TM 区画となるように任命します。WebSphere eXtreme Scale は、コーディネーターとして、トランザクションに関係するすべての区画にコミット要求またはロールバック要求を送ることによってコミット・プロトコルの第 2 フェーズを開始します。そうすると、RM として特定された各区画が変更をバックアップ・マップに適用するかまたはロールバックし、すべてのロックを解除します。その後、RM は WebSphere eXtreme Scale クライアントに通知します。第 2 フェーズで少なくとも 1 つの区画が障害を起こした場合は、委任された TM 区画が自動的にトランザクションをリカバリーします。自動リカバリーにより、トランザクションに関係するすべての区画の整合性が確保されます。

未確定フェーズ:

未確定フェーズとは、RM 区画が第 1 フェーズを正常に処理し終えてから第 2 フェーズを開始するまでの待機期間を指します。未確定期間中、RM 区画はトランザクションをコミットするかロールバックするかを知りません。RM 区画はロックを保持し続けます。ロックが保持されていると、他のトランザクションでロック競合が増加する場合があります。

2 フェーズ・コミット中のエラー・リカバリー

第 1 フェーズで障害が発生すると、WebSphere eXtreme Scale クライアントはトランザクションをロールバックします。区画の 1 つがトランザクションのコミットに失敗すると、TM は、トランザクションのコミットを周期的に試みることによってトランザクションがコミットされるようにします。このシナリオで生成されるログ・メッセージの例を次に示します。

```
00000099 TransactionLog I CW0BJ8705I: Automatic resolution of transaction WXS-40000139-DF01-216D-E002-1CB456931719 at RM:TestGrid:TestSet2:20 is still waiting for a decision. Another attempt to resolve the transaction will occur in 30 seconds.
```

WebSphere eXtreme Scale クライアントがトランザクションを解決できるようにしてください。手動介入を試みるのは、トランザクションが 1 分以内にリカバリーされない場合や、未確定トランザクションとなりアプリケーションで頻繁にロック競合が長発生する場合のみにしてください。トランザクションの手動リカバリーについて詳しくは、[マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)を参照してください。

親トピック: [トランザクション処理の概要](#)

親トピック: **2.5+** [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

関連概念:

[ロック・ストラテジー](#)

関連タスク:

[マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について 実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒103-8510

東京都中央区日本橋箱崎町19番21号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス渉外

以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムと その他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM® Corporation
Mail Station P300
522 South Road
Poughkeepsie, NY 12601-5400
USA
Attention: Information Requests

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性がありますが、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

表示されている IBM の価格は IBM が小売り価格として提示しているもので、現行価格であり、通知なしに変更されるものです。卸価格は、異なる場合があります。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。これらのサンプル・プログラムは特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。

© Copyright IBM Corp. _年を入れる_. All rights reserved.

プログラミング・インターフェース情報

本書の情報は、WebSphere® eXtreme Scale のプログラミング・インターフェースとして使用されることを意図して記述されたものではありません。本書には、プログラムを作成するユーザーが WebSphere eXtreme Scale のサービスを使用するためのプログラミング・インターフェースが記述されています。この情報は、章またはセクションの始まりの文によって、あるいは下記の表示によって、その出現箇所を識別できます: プログラミング・インターフェース情報。

商標

IBM、IBM ロゴ、および ibm.com は、世界の多くの国で登録された International Business Machines Corp. の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、www.ibm.com/legal/copytrade.shtml の [Copyright and trademark information](#) をご覧ください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

プライバシー・ポリシーに関する考慮事項

サービス・ソリューションとしてのソフトウェアも含めた IBM® ソフトウェア製品（「ソフトウェア・オファリング」）では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie はじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的事項を確認ください。

この「ソフトウェア・オファリング」は、Cookie もしくはその他のテクノロジーを使用して個人情報を収集することはありません。

この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人情報を収集する機能を提供する場合、お客様は、個人情報を収集するにあたって適用される法律、ガイドライン等を遵守する必要があります。これには、エンド・ユーザーへの通知や同意取得の要求も含まれますがそれらには限りません。

このような目的での Cookie を含むさまざまなテクノロジーの使用について詳しくは、『IBM プライバシー・ポリシー』（<http://www.ibm.com/privacy>）および『IBM オンラインでのプライバシー・ステートメント』（<http://www.ibm.com/privacy/details/us/en>）の『クッキー、Web ビーコン、その他のテクノロジー』と『ソフトウェア製品と Software-as-a Service』のセクションを参照してください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

シナリオ

全体像を描くために、シナリオには実在の情報が含まれています。シナリオは、新しい概念を理解したり、一般的なタスクを実行するのに役立ちます。

[シナリオ: エンタープライズ・データ・グリッドの構成](#)

Java™ アプリケーションと .NET アプリケーションの両方が同じデータ・グリッドに接続できるようにしたいときは、エンタープライズ・データ・グリッドを構成してください。

2.5+ [WebSphere DataPower XC10 アプライアンスの保護](#)

アプライアンス内のデータ・グリッドは、保護する必要がある機密情報を保管します。

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

以前に設定したメモリー間複製セッションまたはデータベース・セッションをマイグレーションして、WebSphere® eXtreme Scale セッション管理を使用できます。

シナリオ: エンタープライズ・データ・グリッドの構成

Java™ アプリケーションと .NET アプリケーションの両方が同じデータ・グリッドに接続するようにしたいときは、エンタープライズ・データ・グリッドを構成してください。

始める前に

- 製品をインストールします。クライアントの場合は、Java クライアントと .NET クライアントの両方を使用することができます。詳しくは、[WebSphere DataPower XC10 アプライアンスのインストール](#)を参照してください。
- 前のリリースからアップグレードする場合は、すべてのコンテナ・サーバーおよびカタログ・サーバーが同じリリース・レベルでなければなりません。詳しくは、[WebSphere DataPower XC10 アプライアンスの更新](#)を参照してください。

1. [エンタープライズ・データ・グリッドの概要](#)

エンタープライズ・データ・グリッドは eXtremeIO トラnsポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトラnsポートおよびシリアライゼーション形式により、Java クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。

2. [IBM eXtremeIO \(XIO\) の構成](#)

IBM® eXtremeIO (XIO) は、オブジェクト・リクエスト・ブローカー (ORB) を置き換えるトラnsポート・メカニズムです。

3. [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

IBM eXtremeIO を構成した後、エンタープライズ・データ・グリッドにアクセスするアプリケーションを作成することができます。

親トピック: [2.5+ シナリオ](#)

IBM eXtremeIO (XIO) の構成

2.5+ IBM® eXtremeIO (XIO) は、オブジェクト・リクエスト・ブローカー (ORB) を置き換えるトランスポート・メカニズムです。

始める前に

- **2.5** XIO を構成するためには、WebSphere® eXtreme Scale クライアント バージョン 8.6.0.2 以降が必要です。

カタログ・サーバーで XIO を使用可能にすることで、カタログ・サービス・ドメイン内のすべてのコンテナ・サーバーに対して XIO を構成できます。コンテナ・サーバーはカタログ・サーバーのトランスポート・タイプをディスカバーし、そのトランスポート・タイプを使用します。

2.5

このタスクについて

バージョン 2.5 以降を使用して新規構成を作成している場合は、XIO が、デフォルトのトランスポート・メカニズムです。前のファームウェア・バージョンからアップグレードする場合は、トランスポート設定は ORB を使用するように設定されます。

手順

集合で XIO トランスポート・メカニズムを使用可能に設定します。ユーザー・インターフェースで、「集合」 > 「設定」 > 「通信サービス」をクリックします。XIO トランスポート・メカニズムを選択します。トランスポートの設定は、集合全体での設定です。したがって、トランスポート設定を更新すると、集合内のすべてのアプライアンスを再始動する必要があります。

タスクの結果

構成したサーバーは、XIO トランスポートを使用します。構成が正しいことを確認するには、[カタログ・サービス・ドメインのトランスポート・タイプの表示](#)を参照してください。

集合が、XIO トランスポート・メカニズムを使用します。XIO を使用可能に設定すると、以下も使用可能になります。

eXtreme Data Format (XDF)

XDF は、キーおよび値を、言語に依存しない形式でシリアライズし、データ・グリッドに保管します。XDF を使用すると、Java アプリケーションと .NET アプリケーションの両方が同じデータ・グリッド・オブジェクトにアクセスできます。

IBM eXtremeMemory

eXtremeMemory を使用すると、ガーベッジ・コレクションの一時停止を回避することができ、パフォーマンスがより安定し、応答時間が予測可能になります。

2.5+ [カタログ・サービス・ドメインのトランスポート・タイプの表示](#)

カタログ・サービス・ドメインで現在使用されているトランスポート・タイプを表示できます。

親トピック: [シナリオ: エンタープライズ・データ・グリッドの構成](#)

前のトピック: [エンタープライズ・データ・グリッドの概要](#)

次のトピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

親トピック: [集合およびゾーンの構成](#)

前のトピック: [集合間のマルチマスター複製の構成](#)

エンタープライズ・データ・グリッド・アプリケーションの開発

IBM eXtremeIO を構成した後、エンタープライズ・データ・グリッドにアクセスするアプリケーションを作成することができます。

始める前に

- データ・グリッドにアクセスする既存の Java または .NET アプリケーションがなければなりません。アプリケーション作成入門について詳しくは、[入門チュートリアル・モジュール 2: クライアント・アプリケーションの作成](#)を参照してください。

[クラスの進化](#)

eXtreme data format (XDF) ではクラスの進化が可能です。クラスの進化を活用することで、旧バージョンのクラスを使用している古いアプリケーションに影響を及ぼすことなく、データ・グリッドで使用されるクラス定義を進化させることができます。これらの古いクラスは、新しいアプリケーションと同じマップ内のデータにアクセスします。

[Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

ClassAlias および FieldAlias 注釈を使用して、Java と .NET クラス間でのデータ・グリッド・データの共有を使用可能にします。

[PartitionKey 注釈を使用したキーから区画へのマップ](#)

PartitionKey 別名を使用して、データが保存される区画を判別するためにハッシュ・コード計算を実行するフィールドまたは属性を識別します。PartitionKey 注釈は、キー属性でのみ有効です。

[Java および C# の等価データ型](#)

エンタープライズ・データ・グリッド・アプリケーションを開発するときは、Java アプリケーションと C# アプリケーションの間でデータ型に互換性がなければなりません。

親トピック: [シナリオ: エンタープライズ・データ・グリッドの構成](#)

前のトピック: [IBM eXtremeIO \(XIO\) の構成](#)

クラスの進化

eXtreme data format (XDF) ではクラスの進化が可能です。クラスの進化を活用することで、旧バージョンのクラスを使用している古いアプリケーションに影響を及ぼすことなく、データ・グリッドで使用されるクラス定義を進化させることができます。これらの古いクラスは、新しいアプリケーションと同じマップ内のデータにアクセスします。

概要

クラスの進化とは、2つのタイプが共に機能できるだけの十分な互換性を備えているかどうかの決定にかかわる、クラスとフィールドの識別のさらなる拡張です。一方のクラスが他方のクラスよりフィールド数が少ないとき、これらのクラスは共に機能することができます。以下のユーザー・シナリオは XDF の実装を意図したものです。

同じオブジェクト・クラスに複数のバージョンがある場合

このシナリオには、販売アプリケーションで顧客の追跡に使用されるマップがあります。このマップには2つの異なるインターフェースがあります。1つ目は Web での購入用のインターフェースです。2つ目は電話での購入用のインターフェースです。この販売アプリケーションのバージョン2では、Web での買い物客に対して、それぞれの購買習慣に基づいて割引を行うことにします。この割引は「顧客」オブジェクトと一緒に保管されます。電話で受注をする販売員はまだこのアプリケーションのバージョン1を使用しています。このバージョン1では、Web バージョンにある新しい割引フィールドが認識されていません。アプリケーションのバージョン2にある「顧客」オブジェクトが、バージョン1アプリケーションで作成された「顧客」オブジェクトと連動するようにします(逆の場合も同じです)。

別のオブジェクト・クラスに複数のバージョンがある場合

このシナリオには、Java™ で作成され、「顧客」オブジェクトのマップを保持する販売アプリケーションがあります。また、C# で作成されているもう1つのアプリケーションがあり、このアプリケーションは倉庫内の在庫を管理し、商品を顧客に出荷するために使用されます。現在、これらのクラスは、クラス、フィールド、およびタイプの名前に基づき互換性があります。Java の販売アプリケーションでは、販売員を顧客アカウントと関連付けるオプションを「顧客」レコードに追加することにします。ただし、このフィールドは倉庫アプリケーションでは不要であるため、このフィールドを保管するように倉庫アプリケーションを更新することはしません。

同じクラスに複数の非互換バージョンがある場合

このシナリオでは、販売アプリケーションと在庫アプリケーションの両方に「顧客」オブジェクトが含まれています。在庫アプリケーションはストリングの ID フィールドを使用し、販売アプリケーションは整数の ID フィールドを使用します。これらのタイプには互換性がありません。そのため、これらのオブジェクトはおそらく同じマップに保管されません。これらのオブジェクトは XDF シリアライゼーションによって処理され、かつ2つの異なるタイプとみなされるようにする必要があります。このシナリオは実際にはクラスの進化ではありませんが、アプリケーション設計全体の一部として考慮しなければならない事項です。

進化の判別

XDF では、クラス名が一致し、かつフィールド名のタイプが矛盾していないとき、クラスの進化が試みられます。C# アプリケーションと Java アプリケーションの間で、クラス名やフィールド名が若干異なるクラスを一致させようと試みる時は、ClassAlias および FieldAlias 注釈が役立ちます。これらの注釈は、Java アプリケーションまたは C# アプリケーションのいずれか、あるいはその両方に入れることができます。ただし、Java アプリケーションでクラスを検索することは、C# アプリケーションで ClassAlias を定義するよりも効率が悪いことがあります。ClassAlias および FieldAlias 注釈について詳しくは、[ClassAlias および FieldAlias 注釈](#)を参照してください。

シリアライズされたデータ内の欠落フィールドの影響

クラスのコンストラクターはデシリアライゼーション時に呼び出されないため、欠落フィールドは、言語に基づいてそれに割り当てられているデフォルト値を持つこととなります。新しいフィールドを追加するアプリケーションは、旧バージョンのクラスが取得されたとき、欠落フィールドを検出して対処できなければなりません。

古いアプリケーションが新しいフィールドを保持するようにするための唯一の方法であるデータの更新

アプリケーションがフェッチ操作を実行し、クライアントからのシリアライズされた値に含まれている一部のフィールドが欠落している旧バージョンのクラスでマップを更新することがあります。その後、サーバーはサーバー上で値をマージし、元のバージョンのいずれかのフィールドが新しいレコードにマージされるかどうかを判別します。アプリケーションがフェッチ操作を実行した後、エントリーを削除して挿入した場合は、元の値にあったフィールドが失われます。

マージ機能

配列やコレクション内のオブジェクトは XDF によってマージされません。配列やコレクションに対する更新がその配列のエレメントやタイプを変更するように意図されているかどうかは必ずしも明らかとは限りません。マージが位置付けに基づいて行われた場合は、配列内のエントリーが移動されると、XDF では、関連付けるように意図されていないフィールドがマージされることがあります。結果として、XDF では配列やコレクションの内容のマージが試行されないこととなります。ただし、新しいバージョンのクラス定義で配列を追加した場合は、その配列が旧バージョンのクラスにマージし直されます。

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義

ClassAlias および FieldAlias 注釈を使用して、Java™ と .NET クラス間でのデータ・グリッド・データの共有を使用可能にします。

始める前に

- IBM® eXtremeIO が構成されている必要があります。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

このタスクについて

既存の Java クラスがあり、対応する C# クラスを作成する場合は、ClassAlias 注釈と FieldAlias 注釈の使用を検討することができます。このシナリオでは、Java クラス名を含む注釈を C# クラスに追加できます。ClassAlias および FieldAlias 注釈について詳しくは、[ClassAlias および FieldAlias 注釈](#)を参照してください。

手順

ClassAlias および FieldAlias 注釈を使用して、Java クラスと C# クラス間でオブジェクトを相関付けます。 Java

図 1. ClassAlias および FieldAlias 注釈を使用した Java の例

```
@ClassAlias("Employee")
class com.company.department.Employee {

    @FieldAlias("id")
    int myId;

    String name;
}
```

.NET

図 2. ClassAlias および FieldAlias 属性を使用した .NET の例

```
[ ClassAlias( "Employee" ) ]
class Com.MyCompany.Employee {

    [ FieldAlias("id") ]
    int identifier;

    string name;
}
```

[ClassAlias および FieldAlias 注釈](#)

ClassAlias および FieldAlias 注釈を使用して、クラス間でのデータ・グリッドのデータの共有を可能にします。2 つの Java クラス間または Java クラスと .NET クラス間でデータを共有することができます。

.NET [ClassAlias および FieldAlias 注釈](#)

ClassAlias および FieldAlias 注釈を使用して、クラス間でのデータ・グリッドのデータの共有を可能にします。2 つの Java クラス間または Java クラスと .NET クラス間でデータを共有することができます。

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

親トピック: .NET **2.5+** [.NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[ClassAlias および FieldAlias 注釈](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

関連情報:

[レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成](#)

ClassAlias および FieldAlias 注釈

ClassAlias および FieldAlias 注釈を使用して、クラス間でのデータ・グリッドのデータの共有を可能にします。2 つの Java™ クラス間または Java クラスと .NET クラス間でデータを共有することができます。

2 つのクラスを同じ名前およびフィールドで定義した場合は、データ・グリッドのデータは自動的にクラス間で共有されます。例えば、Customer1 クラスが Java アプリケーションにあり、同じフィールドを持つ Customer1 クラスが .NET アプリケーションにある場合は、データはこれらのクラス間で共有されます。この例では、クラス名にクラス修飾子も含まれていて、そのクラス修飾子が Java のパッケージ名でもあり、C# の名前空間名でもあることが前提になっています。名前空間名とパッケージ名は一致するため、これらの名前は自動的に共有されます。次の例を参照してください。これらの名前は両方とも大/小文字を区別しません。

```
Java:
package com.mycompany.app
public class SampleClass {
    int field1;
    String field2;
}
```


```
C#
namespace Com.MyCompany.App
public class SampleClass {
    int field1;
    string field2;
}
```

ただし、異なる名前を持つクラス間でデータを相関することもできます。データ・グリッドに保管するデータを異なる名前の異なるクラス間で相関するには、ClassAlias 注釈または FieldAlias 注釈を使用します。

2 つの Java アプリケーション間: 異なる名前を持つ 2 つの異なるクラスを別々の Java アプリケーション環境で定義することができます。同じ ClassAlias 注釈を持つクラスにマークを付けることによって、この 2 つのクラス間ですべてのフィールドおよびフィールド・タイプが突き合わされます。これらのクラスは、異なるクラス名を持っている場合でも、同じクラス・タイプ ID で相関されます。そのため、異なる Java アプリケーション・ランタイムでは、同じクラス・タイプ ID とメタデータがこれらのクラス間で再使用されます。

Java アプリケーションと .NET アプリケーション間: C# アプリケーション内で類似の注釈を使用して、C# クラスを Java クラスと相関させることができます。クラス C# に対して定義されている ClassAlias 属性およびフィールドが、同じ ClassAlias 注釈を持つ Java クラスに突き合わされます。

親トピック: [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

親トピック:  [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

関連タスク:

[Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

関連情報:

[レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成](#)

PartitionKey 注釈を使用したキーから区画へのマップ

PartitionKey 別名を使用して、データが保存される区画を判別するためにハッシュ・コード計算を実行するフィールドまたは属性を識別します。PartitionKey 注釈は、キー属性でのみ有効です。

始める前に

eXtreme Data Format (XDF) を使用している必要があります。IBM eXtremeIO を使用している場合、XDF は使用可能に設定されています。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

このタスクについて

PartitionKey 別名を設定して、複数のクラスでデータが同じ区画に保存されるようにすることができます。例えば、PartitionKey 値を departmentID キーに設定した場合は、従業員レコードは、同じ区画に配置されることになります。

PartitionableKey インターフェースは既存の Java インターフェースであり、C# の PartitionableKey 注釈よりも優先されます。

手順

- **Java** Java アプリケーションのフィールドに PartitionKey 注釈を定義します。

```
class Employee {
    int empId;

    @PartitionKey(order = 0)
    int deptId;
}
```

複数のキーで PartitionKey 注釈を設定できます。また、クラスで PartitionKey 別名を設定できます。Java アプリケーションで PartitionKey 注釈を設定する方法のさらなる例については、[Java API 資料: 注釈タイプ PartitionKeys](#)を参照してください。

- **.NET** .NET アプリケーションのフィールドで PartitionKey 属性を定義します。

```
class Employee {
    int empId;

    [PartitionKey]
    int deptId;
}
```

.NET クラスでも PartitionKey 属性を設定できます。詳しくは、[『.NET API documentation: PartitionKeyAttribute Class』](#)を参照してください。

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

親トピック: **.NET 2.5+** [.NET API を使用したデータ・グリッド・アプリケーションの開発](#)

Java および C# の等価データ型

エンタープライズ・データ・グリッド・アプリケーションを開発するときは、Java アプリケーションと C# アプリケーションの間でデータ型に互換性がなければなりません。

表 1. Java および C# 間での等価データ型

Java 型	C# 型
boolean	bool
java.lang.Boolean	bool?
byte	sbyte または byte
java.lang.Byte	sbyte?
short	short?、ushort
java.lang.Short	short?、ushort?
int	int、uint、ushort
java.lang.Integer	int?、uint?
long	long、ulong、uint
java.lang.Long	long?、ulong?、uint?
short または int	ushort
java.lang.Short または java.lang.Integer	ushort?
int または long	uint
java.lang.Integer または java.lang.Long	uint?
long または BigInteger	ulong
java.lang.Long または java.lang.BigInteger	ulong?
char, java.lang.Character	char
java.lang.Character	char?
float または java.lang.Float	float
java.lang.Float	float?
double	double
java.lang.Double	double?
java.math.BigDecimal	decimal または decimal?
java.math.BigInteger	decimal、long または ulong?
java.lang.String	ストリング (string)
java.util.Date、java.util.Calendar	System.DateTime
java.util.Date(rounding)、 java.util.Calendar(rounding)	System.DateTime
java.util.ArrayList	System.Collections.ArrayList, System.Collections.Generic.List
java.util.HashMap	System.Collections.Generic.Dictionary、 System.Collections.Hashtable
java.util.LinkedList	System.Collections.Generic.LinkedList
java.util.ArrayList、java.util.Vector	System.Collections.Generic.List
java.util.Stack	System.Collections.Generic.Stack
java.util.Vector	System.Collections.ArrayList、 System.Collections.Generic.List

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

WebSphere DataPower XC10 アプライアンスの保護

アプライアンス内のデータ・グリッドは、保護する必要がある機密情報を保管します。

始める前に

このシナリオのいくつかの要素、例えば、連邦情報処理標準 140-2 (FIPS) を使用可能な場合などは、集合内のすべてのメンバーが最新レベルでなければなりません。保護するアプライアンスが集合のメンバーである場合は、その集合のすべてのメンバーにアップグレード済みのファームウェアがなければなりません。そうでないと、このシナリオのタスクを完了できません。

このタスクについて

WebSphere® DataPower® XC10 アプライアンスには、総合的なセキュリティー管理が含まれています。デフォルト構成には、ユーザーが変更する必要があるデフォルトのパスワード、SSL 鍵、および認証の秘密が含まれています。以下のシナリオを完了して構成を変更し、その後アプライアンスのデプロイメントをセキュアにします。

セキュア・デプロイメントでは、セキュリティーを最適化するために、複数の層の保護を使用します。保護の最初のエレメントは、ネットワークをセグメント化するファイアウォールの使用です。Web アプリケーションの標準階層モデルは、Web クライアント、HTTP サーバーのプレゼンテーション層、アプリケーション・サーバーから成るアプリケーション層、データ層、およびストレージ層で構成されます。

WebSphere DataPower XC10 アプライアンスアプライアンスは、データ層の一部としてデプロイされます。標準プラクティスでは、1つのファイアウォールによって保護される非武装地帯 (DMZ) にプレゼンテーション層のサーバーを配置し、複数のファイアウォールによって保護されるネットワーク・セグメントにアプリケーション、データ、およびストレージの各層を配置します。アプライアンスを DMZ にデプロイしないでください。業界の標準プラクティスに従って、データ層の他のすべてのエレメントと同様にアプライアンスも保護する必要があります。

ただし、セキュリティー上の脅威に対する最適な保護のために、いくつかの追加手段によってアプライアンス操作やデータ・グリッドに保管されたデータを保護する徹底した防御メカニズムを使用してください。これらの追加手段は、外部からの脅威に対する保護に役立つだけでなく、アプライアンスが存在するネットワーク・セグメントにアクセスできる可能性がある従業員や請負業者による無許可データ・アクセスも防止します。

このシナリオのステップは、WebSphere DataPower XC10 アプライアンスの Web コンソールで実行されます。これらの各ステップは、プログラムから HTTP コマンド行インターフェースを呼び出すことによって自動化することもできます。HTTP コマンド行インターフェースの詳細については、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

1. **2.5+** [データ・グリッドへのセキュアなアクセスの構成](#)

管理アクセスを構成し、アプライアンス集合の設定を指定して、データ・グリッドへのセキュアな認証と許可を構成します。

2. **2.5+** [データ・グリッドのセキュリティーを使用可能に設定する](#)

データ・グリッドの作成後、データ・グリッド (data grid)のセキュリティーはデフォルトで使用不可になります。データ・グリッド (data grid)のセキュリティー設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

3. **2.5+** [クライアント・セキュリティーの構成](#)

アプライアンスのデータ・グリッドを保護したら、クライアントがその保護されているデータ・グリッドに接続できるようにクライアントを構成する必要があります。

親トピック: **2.5+** [シナリオ](#)

データ・グリッドへのセキュアなアクセスの構成

管理アクセスを構成し、アプライアンス集合の設定を指定して、データ・グリッドへのセキュアな認証と許可を構成します。

このタスクについて

セキュリティ上の脅威に対する最適な保護のために、以下のステップを実行してください。これらのステップは、外的脅威を防止したり、アプライアンスがクライアントとサーバーの間の通信を行うネットワーク・セグメントにアクセスできる可能性がある従業員や請負業者による無許可データ・アクセスを防止したりするのに役立ちます。

重要: 管理コンソールでサーバー間通信やネットワーク上のデータの保護のための集合設定を指定するときに、セキュリティを制御する設定を構成することができます。これらの設定を変更した場合、集合全体を再始動する必要があります。これらの設定を指定するには、「集合」 > 「設定」とクリックします。効率化のために、すべての変更を同時に行ってから、それらの変更を実行依頼するようにしてください。そうすれば、集合の再始動は一度だけで済みます。

手順

1. 管理アクセスを保護します。

管理ユーザー ID (xcadmin) とデフォルトの管理パスワード (xcadmin) を使用してアプライアンスを構成します。このユーザー ID とパスワードにより、デバイスと集合のすべての管理機能およびデータに対する全アクセス権限が付与されます。推測するのが難しいルート管理パスワードを構成する必要があります。このアクションを実行するには、Web コンソールで、「集合」 > 「ユーザー」とクリックします。「管理者」ユーザーを選択し、パスワードを編集してください。

2. SSL トランスポート・セキュリティを構成します。

アプライアンスは、デフォルトの鍵ストアとトラストストアが構成されて出荷されます。デフォルトのトラストストアには、デフォルトの鍵ストアの署名者証明書が含まれます。この証明書はどのアプライアンスにも同梱されているので、SSL 構成をセキュアにするためには、これを取り換える必要があります。証明書の取り換えには、いくつかのステップが必要です。

- a. 新規に証明書と秘密鍵を作成します。この証明書は、自己署名のもの、あるいは、ローカルの認証局が発行するもの (推奨) のいずれかを使用します。
- b. トラストストアを作成または変更します。1つの方法は、デフォルトのトラストストアをアプライアンスからダウンロードして、「**IBM WebSphere DataPower XC10**」というラベルの付いたトラストストアの署名者証明書を削除して、新しい鍵ストアでトラストを確立する証明書を追加することです。新しい鍵ストアに自己署名の証明書がある場合は、その証明書をトラストストアにインポートする必要があります。新しい鍵ストアにローカルの認証局により発行された証明書がある場合は、その認証局のルート証明書をトラストストアに追加する必要があります。
- c. 新しい鍵ストアおよびトラストストアをアップロードします。
- d. 鍵ストアの証明書の名前と一致するように証明書別名の設定を変更します。すべてのデータ・グリッド・クライアント用のトラストストア・ファイルも、アプライアンスの証明書とのトラストを確立するために更新する必要があります。

Java ランタイム環境 (JRE) に付属している keytool コマンドを使用して、鍵ストアとトラストストアを管理します。他の鍵管理ツールも同様に使用できます。鍵ストア・ファイルとトラストストア・ファイルを管理するために実行する以下の管理用タスクを参照してください。

- 自己署名証明書を生成します。

```
keytool -genkey -alias ogsample -keystore key.jks -storetype JKS -keyalg  
rsa -dname "CN=ogsample, OU=OGSample, O=acme, L=Your City, S=Your State,  
C=Your Country" -storepass ogpass -keypass ogpass -validity 3650
```

- アプライアンスからダウンロードするトラストストアの中から、アプライアンスのデフォルトの署名者証明書を取得します。

```
keytool -delete -alias "ibm websphere datapower xc10" -keystore trust.jks  
-storetype JKS -keypass xc10pass
```

- 鍵ストアから新規証明書をエクスポートします。

```
keytool -export -alias ogsample -keystore key.jks -file temp.key -  
storepass ogpass
```

- トラストストアにその証明書をインポートします。

```
keytool -import -noprompt -alias ogsamplepublic -keystore trust.jks -file temp.key -storepass xc10pass
```

このインポートのステップは、鍵ストアからエクスポートされた証明書をトラストするようトラストストアを構成するために実行する必要があります。このステップは、いくつかのコンテキストで使用されます。クライアントのトラストストアは、アプライアンスの鍵ストアの証明書をトラストするように更新される必要があります。構成によっては、アプライアンスのトラストストアが、クライアントの鍵ストアの証明書をトラストするように更新される必要があります。クライアントの鍵ストアとアプライアンスの鍵ストアの両方の証明書が、ローカルの認証局によって発行されるのが理想的です。この場合、認証局のルート署名者証明書をインポートすることでトラストが確立されます。

- 鍵ストアとトラストストア上のパスワードを変更します。

```
keytool -storepasswd -new newpass -keystore trust.jks -storepass xc10pass
```

- 新規鍵ストアとトラストストアをアプライアンス集合にアップロードします。

- e. 次のオプションの中から SSL モードを指定します: **TCP/IP** (この場合、SSL はデータ・グリッド通信には使用されません)、**TLS サポート**、および **TLS 必須**。

セキュリティのためには、「**TLS 必須**」を推奨します。TLS が使用されない場合、パスワードおよび機密グリッド・データが、グリッド・クライアントとアプライアンスを接続するネットワーク・リンク上を暗号化されないまま受け渡されます。

注: TLS 通信が必要な場合、アプライアンスのトラストストアは、クライアントの鍵ストアの証明書をトラストするように構成されている必要があります。クライアントのトラストストアは、サーバー鍵ストアの証明書をトラストするように構成されている必要があります。[クライアント・セキュリティの構成](#)を参照してください。

- f. 連邦情報処理標準 (FIPS) を使用可能に設定します。

すべての暗号化ネットワーク通信で FIPS 140-2 を使用するようにアプライアンス集合を構成することができます。この標準により、通信で送信されるデータの優れた保護が確保されます。「**FIPS 140-2 暗号方式を使用可能に設定**」を選択して FIPS を使用可能にします。

一部の Web ブラウザー・バージョンは、FIPS 対応サーバーで動作しません。Mozilla Firefox、Microsoft Internet Explorer、および Google Chrome などのほとんどのブラウザの現行バージョンは、FIPS 対応サーバーとの通信をサポートしています。FIPS モードでは SSLv3 がサポートされないため、ブラウザを構成して、TLS を使用できるようにする必要がある可能性があります。詳しくは、ご使用のブラウザの資料を参照してください。

- g. クライアント証明書認証を有効にします。

これを有効にする場合、アプライアンスと通信する各クライアントおよびブラウザの鍵ストアを、アプライアンス・トラストストアによってトラストされている証明書を持つように構成する必要があります。クライアント証明書認証は、ORB トランスポートでは使用されません。これは、HTTPS と XIO トランスポートの場合でのみ使用されます。クライアント証明書認証を有効にしなくてもセキュアな構成は保持されます。

3. サーバー間の認証を構成します。

集合内のアプライアンス間、およびリンクされたアプライアンス・ドメイン間のデータ・グリッド通信は、共有秘密鍵を使用して認証されます。アプライアンスは、各アプライアンスにハードコーディングされているデフォルトの秘密鍵で既に構成されます。セキュアな構成を保持するためには秘密鍵を変更する必要があります。「**出荷時のデフォルト認証秘密鍵をオーバーライド**」を選択して固有の秘密鍵を指定してください。

秘密鍵は、推測するのが難しい長いパスフレーズでなければなりません。秘密鍵を書き留めて、安全な場所に保管してください。リンク・ドメイン内で集合が結合される際、各集合は同じ秘密鍵を使用して構成される必要があります。

4. すべてのデータ・グリッド要求に認証を要求します。

データ・グリッドへのクライアント要求ごとに認証を構成することができます。デフォルトでは、この認証は必須ではありません。しかし、個々のグリッドを保護することによってデータ・グリッドへのアクセスを保護することができます。また、セキュア構成の場合は、すべてのグリッド要求について認証が必要になります。これが設定されると、アプライアンス集合によって認識されるユーザー ID とパスワードを使用して各クライアントを構成する必要があります。また、LDAP 認証の場合は、ユーザー ID とパスワードが LDAP に登録されていなければなりません。LDAP 認証なしにログインできるのは、ルート管理者 (xcadmin ユーザー ID) のみです。

ステップ 2 から 4 が完了したら、変更を実行依頼して、集合を再始動してください。これらの設定は、集合に同化されるアプライアンスに自動的に伝搬されます。FIPS を使用可能に設定した場合は、アプライアンスが集合に同化される前に各アプライアンスで FIPS が使用可能になります。

5. Telnet アクセスを使用不可に設定します。

アプライアンスのデフォルト構成には、アクティブな Telnet サーバーが含まれています。アプライアンスの Telnet 通信は、SSL をサポートしません。セキュアな構成では、Telnet サーバーを使用不可に設定してください。Telnet を使用不可にするには、管理者のユーザー ID とパスワードを使用してアプライアンスへの SSH セッションを確立し、コマンド **platform service telnet disable** を実行します。これは集合全体に対する設定ではありません。つまり、このコマンドをアプライアンスごとに実行する必要があります。コマンド **platform service telnet disable** は、Telnet が使用不可になっている場合に Telnet を開始します。この手順は手動であり、自動化することはできません。

6. LDAP 認証を構成します。

ブラウザー、REST、およびアプライアンスに対するデータ・グリッド・アクセスの認証は、次の 2 つの方法のいずれかで行われます。認証済み ID をアプライアンス集合または LDAP に保管することができます。セキュア構成ではどちらの方法も使用できます。

ヒント: ルート管理者の認証では、LDAP ではなく集合によって検証されたパスワードが常に使用されます。

LDAP 認証を使用する場合、パスワードが暗号化されないままネットワーク上を通過しないようにするために、SSL との LDAP 接続を保護してください。LDAP 接続に対して SSL を使用可能にするには、LDAPS URL (例えば、`ldaps://ldapserverserver.company.com:636` など) を指定します。ここで、LDAP サーバーとのトラストを確立して SSL 通信を行うために、証明書を使用してアプライアンス・トラストストアを構成する必要があります。LDAP 認証の構成の詳細については、[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)を参照してください。

親トピック: [2.5+ WebSphere DataPower XC10 アプライアンスの保護](#)

次のトピック: [2.5+ データ・グリッドのセキュリティーを使用可能に設定する](#)

関連情報:

↳ [連邦情報処理標準 Java Secure Socket Extension ファイルの構成](#)

データ・グリッドのセキュリティーを使用可能に設定する

データ・グリッドの作成後、データ・グリッド (data grid)のセキュリティーはデフォルトで使用不可になります。データ・グリッド (data grid)のセキュリティー設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

このタスクについて

データ・グリッドが集合の一部である場合、そのデータ・グリッドに対してセキュリティーおよびアクセス制御を構成することができます。セキュリティー設定は、そのデータ・グリッドにアクセスするクライアントが確実に認証されるようにします。すべてのデータ・グリッド・アクセスに対して認証が必要なように集合が構成されている場合、当該データ・グリッドへの認証が常に必要となります。認証を必要とするということは、グリッドへのアクセスが十分に制限されない可能性があるということです。例えば、LDAP 認証の場合は、どの LDAP ユーザーも特定のグリッドにアクセスできます。アクセスを制限するには、そのグリッドに対する許可を使用可能に設定します。

重要: データ・グリッドのセキュリティー設定を変更すると、データ・グリッドは自動的に再始動します。データ・グリッドが再始動すると、データ・グリッド内のすべてのデータが失われます。データ・グリッドのデータの保存を開始する前に、データ・グリッドのセキュリティーを構成します。

REST ゲートウェイを介した通信は、データ・グリッドでセキュリティーを有効にしていなくても、常に安全です。詳しくは、[REST ゲートウェイ: セキュリティー構成](#)を参照してください。

手順

1. ユーザー・インターフェースで、データ・グリッド (data grid)の設定にナビゲートします。「データ・グリッド」>「**data_grid_type**」をクリックします。編集する **data_grid_name** をクリックします。
2. データ・グリッド (data grid)に関するセキュリティーまたは許可を有効にします。「セキュリティーを有効にする」をクリックして、ユーザー・インターフェースにアクセス権を持つすべてのユーザーがデータ・グリッド (data grid)にアクセスできるようにします。さらにアクセス権を制限する場合は、「許可を使用可能に設定」をクリックします。

注: セッション・データ・グリッドの許可を使用可能にすることはできません。

許可を使用可能にすると、「アクセス認可」リストにユーザーまたはユーザー・グループのリストを指定できます。「許可を使用可能に設定」が選択された場合、このアクセス・リストに登録されたユーザーのみがデータ・グリッド (data grid)データにアクセスできます。ユーザー・インターフェースに表示されるデフォルトのアクセス・タイプの名前をクリックすることで、ユーザーまたはユーザー・グループに対して以下のアクセス権を割り当てることができます。

- 「**読み取り**」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドのデータの読み取りまたは照会をすることができます。
- 「**書き込み**」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドに対するデータの読み取り、照会、および書き込みができます。
- 「**作成**」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドの読み取り、照会、書き込み、挿入、および動的マップの作成ができます。
- 「**すべて**」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドの読み取り、照会、書き込み、挿入、動的マップの作成、除去、およびデータの無効化を行うことができます。アプライアンス管理者には、デフォルトで「すべて」の許可があります。

セキュリティーおよび許可の設定を変更する場合、5 分のタイムアウト値があります。

- **認証タイムアウト:** 既にデータ・グリッドに対して認証されたユーザーのユーザー・パスワードを変更した場合、元の資格情報は最大 5 分間有効なままです。
- **許可タイムアウト:** ユーザーの許可を除去すると、そのユーザーは、最大 5 分間、その許可を所有し続けます。このタイムアウトは、除去された許可に対してのみ適用されます。ユーザーに許可を付与すると、そのユーザーは、直ちに許可を取得します。

親トピック: [2.5+ WebSphere DataPower XC10 アプライアンスの保護](#)

前のトピック: [2.5+ データ・グリッドへのセキュアなアクセスの構成](#)

次のトピック: [2.5+ クライアント・セキュリティーの構成](#)

親トピック: [データ・グリッドの構成](#)

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadmin/パスワード](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

Java

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

Java [クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)
[データ・グリッド・アプリケーション用の TLS の構成](#)
[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

クライアント・セキュリティーの構成

アプライアンスのデータ・グリッドを保護したら、クライアントがその保護されているデータ・グリッドに接続できるようにクライアントを構成する必要があります。

このタスクについて

アプライアンスが TLS を必要とするように構成されている場合、クライアントは SSL トラnsポート用に構成されている必要があります。さらにクライアントに適切な鍵ストアとトラストストアが必要です。認証が必要な場合は、クライアントは、ユーザー ID とパスワードを使用して構成されている必要もあります。具体的な手順は、クライアントがスタンドアロンのインストール済み環境であるか、WebSphere® Application Server プロセスで稼働するインストール済み環境であるかによって異なります。

スタンドアロン環境の場合は、データ・グリッド・アプリケーションに渡される設定を含む `client.properties` ファイルを構成する必要があります。ご使用の環境に WebSphere Application Server がある場合は、WebSphere Application Server ツールを使用してクライアント・セキュリティーを構成してください。

手順

1. クライアントがスタンドアロンのインストール済み環境である場合は、クライアント・プロパティー・ファイルを使用して、クライアントとサーバーの間の通信およびクライアント・セキュリティーを構成します。このファイルのロケーションとフォーマットについては、[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)を参照してください。

Java™ クライアントまたは .NET クライアントのために構成できるセキュリティー・プロパティーについては、[クライアント・プロパティー・ファイル](#)を参照してください。

2. クライアントが WebSphere Application Server プロセスで稼働している場合は、WebSphere Application Server ツールを使用して、鍵ストアやトラストストアなどの SSL 構成値を構成します。

証明書をアプライアンス鍵ストアから WebSphere Application Server トラストストアにインポートするユーティリティーが提供されています。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

アプライアンス鍵ストアからの証明書を信頼するように、その WebSphere Application Server トラストストアを構成する必要があります。アプライアンスがクライアント証明書認証用に構成されている場合、またはオブジェクト・リクエスト・ブローカー (ORB) 通信が使用されている場合は、WebSphere Application Server 鍵ストアからの証明書も信頼するようにアプライアンス・トラストストアを構成する必要があります。

認証を必要とするようにアプライアンスが構成されている場合は、WebSphere Application Server のもとで実行されるアプライアンス・クライアントも、認証資格情報を使用して構成する必要があります。Java スタンドアロン・クライアントの場合と同様、プロパティー・ファイルを使用してクライアントを構成することができます。ただし、アプライアンス・クライアントが WebSphere Application Server にインストールされている場合、WebSphere Application Server 管理コンソールが拡張され、この管理コンソールで認証資格情報を指定できるようになります。この拡張により、カタログ・サービス・ドメインを構成したり、クライアント認証プロパティーを指定したりできます。

3. サーバーに連邦情報処理標準 (FIPS) が設定されている環境でクライアントが稼働している場合、クライアントは TLS ハンドシェーク・プロトコルを使用してアプライアンスと通信する必要があります。SSL ハンドシェーク・プロトコルは使用できません。TLS プロトコルは、クライアント・プロパティー・ファイル内の設定 `protocol=TLS` で構成されます。
 - XIO が使用され、クライアント・プロパティー・ファイルにプロトコル設定がない場合のデフォルト設定は SSL および TLS です。この設定は、アプライアンスで FIPS が使用可能になっている場合に機能します。
 - スタンドアロン・クライアントで ORB 通信が使用される場合、FIPS モードでアプライアンスと通信するためには、クライアント・プロパティー・ファイルでプロトコル設定を TLS にしなければなりません。
 - ORB 通信が使用され、クライアントが WebSphere Application Server と共に稼働している場合、ハンドシェーク・プロトコルを設定するために WebSphere セキュリティー設定が使用されます。デフォルトでは、この設定は SSL_TLS です。これは、アプライアンスで FIPS が使用可能になっている場合に機能します。

WebSphere Application Server 管理コンソールを使用して、この設定を TLS または SSL_TLS に構成することができます。「セキュリティー」 > 「SSL 証明書および鍵管理」 > 「エンドポイント・セキュリティー構成の管理」をクリックします。セル名を選択し、「SSL 構成」 > 「CellDefaultSSLSettings」 > 「保護品質 (QOP) 設定」をクリックします。プロトコル・フィールドが TLS または SSL_TLS に設定されていることを確認してください。

親トピック: [2.5+ WebSphere DataPower XC10 アプライアンスの保護](#)

前のトピック: [2.5+ データ・グリッドのセキュリティーを使用可能に設定する](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション

以前に設定したメモリー間複製セッションまたはデータベース・セッションをマイグレーションして、WebSphere® eXtreme Scale セッション管理を使用できます。

始める前に

- WebSphere Application Server において実行されているクライアント・アプリケーション (クラスター) の用のセッション・サポートでは、WebSphere eXtreme Scale を、WebSphere Application Server ノード・デプロイメント (デプロイメント・マネージャー・ノードを含む) 上にインストールする必要があります。[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。

このタスクについて

このシナリオの手順は、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

WebSphere DataPower XC10 アプライアンス セッションへのマイグレーションの一部として、WebSphere Application Server 管理コンソールの前の構成設定をメモする必要があります。WebSphere DataPower XC10 アプライアンス セッションにマイグレーションする際に、構成設定で、データベース・セッションおよびメモリー間セッション用に既に構成されている内容を反映する必要があります。

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

データ・グリッド内の HTTP セッションへのマイグレーションの一部として、WebSphere Application Server 管理コンソールでカタログ・サービス・ドメインを作成する必要があります。

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

WebSphere Application Server 管理コンソール内のメモした以前の構成設定を使用して、アプリケーションまたはアプリケーション・サーバーのいずれかを WebSphere eXtreme Scale セッション管理に関連付ける必要があります。

親トピック: [2.5+ シナリオ](#)

親トピック: [セッション・データ・グリッドの作成](#)

関連タスク:

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

WebSphere Application Server 管理コンソールの前の構成設定のメモ

WebSphere® DataPower® XC10 アプライアンス セッションへのマイグレーションの一部として、WebSphere Application Server 管理コンソールの前の構成設定をメモする必要があります。WebSphere DataPower XC10 アプライアンス セッションにマイグレーションする際に、構成設定で、データベース・セッションおよびメモリー間セッション用に既に構成されている内容を反映する必要があります。

このタスクについて

WebSphere Application Server 管理コンソール内にメモする必要がある特定の設定があります。これらの値は、`splicer.properties` ファイルの更新時に必要になります。この手順のステップは、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale クライアント バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

手順

- WebSphere Application Server 管理コンソールを開始します。
 - 以前にサーバー・レベルで設定を構成したことがある場合は、以下にアクセスします。
 - 「サーバー」 > 「サーバー・タイプ」 > 「**WebSphere Application Server**」
 - 「アプリケーション・サーバー」領域で、ご使用のサーバー名を選択します。
 - 「コンテナ設定」領域で、「セッション管理」をクリックします。
 - 以前にアプリケーション・レベルで設定を構成したことがある場合は、以下にアクセスします。
 - 「アプリケーション」 > 「すべてのアプリケーション」。
 - 「アプリケーション・サーバー」領域で、ご使用のアプリケーション名を選択します。
 - 「**Web** モジュール・プロパティ」領域で、「セッション管理」をクリックします。
- 「一般プロパティ」で、「オーバーフローの許可」チェック・ボックスを選択します。
- 「一般プロパティ」領域で、WebSphere Application Server 設定をメモします。これらの値は後から、`splicer.properties` ファイルのプロパティを更新する際に必要になります。

表 1. `splicer.properties` ファイルを更新するための構成設定

WebSphere Application Server 管理コンソールでの設定	<code>splicer.properties</code> ファイルで更新するプロパティ
Cookie 使用可能 (Enable cookies)	<code>useCookies</code>
URL 再書き込みを有効にする	<code>useURLEncoding</code>
メモリー内の最大セッション・カウント	<code>sessionTableSize</code>

- 「一般プロパティ」領域で、「**Cookie 使用可能 (Enable cookies)**」チェック・ボックスが選択されている場合は、それをクリックし、WebSphere Application Server 設定をメモします。これらの値は後から、`splicer.properties` ファイルのプロパティを更新する際に必要になります。

表 2. `splicer.properties` ファイルのプロパティの構成設定

WebSphere Application Server 管理コンソールでの設定	<code>splicer.properties</code> ファイルで更新するプロパティ
Cookie ドメイン	<code>cookieDomain</code>
Cookie パス	<code>cookiePath</code>

- 「セッション管理」をクリックし、「追加プロパティ」領域で、「分散環境設定」をクリックします。
- 「分散セッション」領域で、以前のデータベース構成またはメモリー間複製構成を「なし」に変更します。
- 「カスタム・チューニング・プロパティ (**Custom Tuning Properties**)」をクリックし、WebSphere Application Server 設定をメモします。これらの値は後から、`splicer.properties` ファイルのプロパティを更新する際に必要になります。

表 3. `splicer.properties` ファイルのプロパティの構成設定

WebSphere Application Server 管理コンソールでの設定	<code>splicer.properties</code> ファイルで更新するプロパティ
書き込み頻度	<code>replicationInterval</code>
書き込みの内容	<code>fragmentedSession</code>

次のタスク

次に、WebSphere DataPower XC10 アプライアンス セッション用のカタログ・サービス・ドメインを作成します。

親トピック: [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成

データ・グリッド内の HTTP セッションへのマイグレーションの一部として、WebSphere® Application Server 管理コンソールでカタログ・サービス・ドメインを作成する必要があります。

このタスクについて

この手順のステップは、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale クライアント バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

WebSphere Application Server 管理コンソールで、データ・グリッド用のカタログ・サービス・ドメインを作成します。詳しくは、[WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)を参照してください。

手順

1. WebSphere Application Server 管理コンソールを開始します。
2. 上部メニューで、「システム管理」>「**WebSphere eXtreme Scale**」>「カタログ・サービス・ドメイン」をクリックします。
注: WebSphere eXtreme Scale が表示されない場合は、WebSphere Application Server プロファイルがデータ・グリッド用に拡張されていません。を参照してください。
3. 「新規」をクリックします。
4. 「名前」ボックスにカタログ・サービスの名前を指定します。
5. 「カタログ・サーバー」領域で、「リモート・サーバー」を選択し、ボックス内にリモート・サーバーの場所または名前を指定します。
6. 「リスナー・ポート」ボックスにポート番号を指定します。
7. 「適用」または「**OK**」をクリックし、構成を保存します。

次のタスク

次に、WebSphere Application Server 管理コンソール内のメモした以前の構成設定を使用して、アプリケーションまたはアプリケーション・サーバーのいずれかを WebSphere eXtreme Scale セッション管理に関連付けます。

親トピック: [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

以前の構成設定を使用するための **WebSphere DataPower XC10** アプライアンス の構成

WebSphere® Application Server 管理コンソール内のメモした以前の構成設定を使用して、アプリケーションまたはアプリケーション・サーバーのいずれかを WebSphere eXtreme Scale セッション管理に関連付ける必要があります。

このタスクについて

この手順のステップは、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale クライアント バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

手順

- WebSphere eXtreme Scale セッション管理に関連付けられるようにアプリケーションを構成する場合は、以下の手順に従います。
 1. WebSphere Application Server 管理コンソールを開始します。
 2. 上部メニューで、「アプリケーション」>「すべてのアプリケーション」をクリックします。
 3. 「**WebSphere** エンタープライズ・アプリケーション」領域で、「アプリケーション名」を選択します。
 4. 「**Web** モジュール・プロパティ」領域で、「セッション管理」をクリックします
 5. 「**eXtreme Scale** セッション管理設定」をクリックします。
 6. WebSphere eXtreme Scale が表示されない場合は、WebSphere Application Server プロファイルが WebSphere eXtreme Scale 用に拡張されていません。詳しくは、を参照してください。
 7. WebSphere eXtreme Scale クライアント 用にアプリケーションを構成するには、以下の手順に従います。
 - a. 「セッション・パーシスタンスの管理」リストで、「**IBM WebSphere DataPower XC10** アプライアンス」を選択します
 - b. リストから、作成したカタログ・サービス・ドメインの IP またはホスト名を指定します。
 - c. ユーザー名とパスワードを指定し、接続をテストします。
 - d. 新規データ・グリッドを指定するか、既存のものを選択します。
 8. 「適用」または「**OK**」をクリックし、構成を保存します。
 9. このアプリケーション用に新規 `splicer.properties` ファイルが作成されます。 `splicer.properties` ファイルの場所は、新規プロパティ `{application name}, com.ibm.websphere.xs.sessionFilterProps` の値です。このカスタム・プロパティを見つけるには、「システム管理」>「セル」にアクセスし、「カスタム・プロパティ」をクリックします。
 10. [WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)で取得した値を使用して、`splicer.properties` ファイルを更新します。
 11. アプリケーション・サーバー・プロセスを再始動します。

注: デプロイメント・マネージャー・レベルで `splicer.properties` を変更して、プロパティがノード・エージェントに同期されるようにします。ノード・レベルで `splicer.properties` を更新した場合は、次回同期時に、デプロイメント・マネージャーによって `splicer.properties` ファイルが上書きされます。

注: データベース・セッション管理に戻ってから WebSphere eXtreme Scale セッション管理に戻ると、`splicer.properties` ファイルが再作成されるため、行った変更がすべてオーバーライドされます。デプロイメント・マネージャーからノードへのファイル同期プロセスおよび変更される内容については、[「System Management File Synchronization」](#)を参照してください。

- WebSphere eXtreme Scale セッション管理に関連付けられるようにアプリケーション・サーバーを構成する場合は、以下の手順に従います。
 1. WebSphere Application Server 管理コンソールを開始します。
 2. 上部メニューで、「サーバー」>「サーバー・タイプ」>「**WebSphere Application Server**」をクリックします。
 3. 「アプリケーション・サーバー」領域で、ご使用のサーバー名を選択します。
 4. 「コンテナ設定」領域で、「セッション管理」をクリックします。
 5. 「**eXtreme Scale** セッション管理設定」をクリックします。

注: WebSphere eXtreme Scale が表示されない場合は、WebSphere Application Server プロファイルが WebSphere eXtreme Scale 用に拡張されていません。詳しくは、を参照してください。

 6. WebSphere eXtreme Scale クライアント 用にアプリケーションを構成するには、以下の手順に従います。
 - a. 「セッション・パーシスタンスの管理」リストで、「**IBM WebSphere DataPower XC10** アプライアンス」を選択します
 - b. リストから、作成したカタログ・サービス・ドメインの IP またはホスト名を指定します。
 - c. ユーザー名とパスワードを指定し、接続をテストします。

- d. 新規データ・グリッドを指定するか、既存のものを選択します。
7. 「適用」または「OK」をクリックし、構成を保存します。
8. このアプリケーション用に新規 splicer.properties ファイルが作成されます。 splicer.properties ファイルの場所は、新規プロパティ com.ibm.websphere.xs.sessionFilterProps の値です。このカスタム・プロパティを見つけるには、「サーバー」>「サーバー・タイプ」>「**WebSphere Application Server**」にアクセスします。
9. 「アプリケーション・サーバー」領域で、ご使用のサーバー名を選択します。
10. 「サーバー・インフラストラクチャー」領域で、「カスタム・プロパティ」を選択します。
11. [WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)で取得した値を使用して、splicer.properties ファイルを更新します。
12. アプリケーション・サーバー・プロセスを再始動します。

注: デプロイメント・マネージャー・レベルで splicer.properties を変更して、プロパティがノード・エージェントに同期されるようにします。ノード・レベルで splicer.properties を更新した場合は、次回同期時に、デプロイメント・マネージャーによって splicer.properties ファイルが上書きされます。

注: データベース・セッション管理に戻ってから WebSphere eXtreme Scale セッション管理に戻ると、splicer.properties ファイルが再作成されるため、行った変更がすべてオーバーライドされます。デプロイメント・マネージャーからノードへのファイル同期プロセスおよび変更される内容については、[「System Management File Synchronization」](#)を参照してください。

タスクの結果

これで、WebSphere eXtreme Scale セッション管理で、メモリー間複製セッションまたはデータベース・セッション管理用に以前の構成設定が変更されました。

親トピック: [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

[| 次へ >](#)

チュートリアル: 単純データ・グリッド・アプリケーションの入門

入門用サンプル・アプリケーションを使用して、クライアントのインストール済み環境とアプライアンスの間の接続を検証できます。入門用サンプル・アプリケーションは、エンタープライズ・データ・グリッドの概要を紹介しています。

学習目標

- ユーザー・インターフェースでデータ・グリッドを作成する方法を学習する。
- クライアント・アプリケーションの開発について学習する (Java または .NET プログラミング言語での開発)。プログラミング言語間の相互運用方法およびエンタープライズ・データ・グリッドの作成について学習する。
- クライアント・アプリケーションを実行して、データをデータ・グリッドに挿入する。
- web コンソールでデータ・グリッドをモニターする。

所要時間

60 分

[| 次へ >](#)

[< 前へ | 次へ >](#)

入門チュートリアル・レッスン 1.1: データ・グリッドの定義

データ・グリッドは、ユーザー・インターフェースで定義できます。このチュートリアルでは、`my_simple_data_grid` という名前のデータ・グリッドを作成してください。

関連タスク:

[単純データ・グリッドの作成](#)

単純データ・グリッドの作成

単純データ・グリッド (data grid) を使用して、作成、取得、更新、および削除の各操作を実行することができます。

- アプライアンスを初期化して、構成する必要があります。詳しくは、[WebSphere DataPower XC10 アプライアンスのインストール](#) を参照してください。

単純データ・グリッド (data grid) を作成します。ユーザー・インターフェースで、「データ・グリッド」 > 「単純データ・グリッド」をクリックします。追加アイコン (+) をクリックし、作成する単純データ・グリッド (data grid) の名前を指定します。このチュートリアルでは、`my_simple_data_grid` という名前のデータ・グリッドを作成します。

レッスンのチェックポイント

このレッスンでは、以下を学習しました。

- ユーザー・インターフェースで単純データ・グリッドを作成する方法

[< 前へ | 次へ >](#)

[< 前へ | 次へ >](#)

入門チュートリアル・モジュール 2: クライアント・アプリケーションの作成

データ・グリッドでデータを挿入、更新、削除、および取得するクライアント・アプリケーションを作成します。サンプル・アプリケーションを使用して、ご使用の環境でアプリケーションを作成する方法が学習できます。

学習目標

このモジュールのレッスンを完了すると、以下のタスクの実行方法が理解できるようになります。

- [Java クライアント・アプリケーションの開発](#)
- [.NET クライアント・アプリケーションの開発](#)
- [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

このモジュールのレッスン

[入門チュートリアル・レッスン 2.1: Java クライアント・アプリケーションの作成](#)

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる Java クライアント・アプリケーションが組み込まれています。

[入門チュートリアル・レッスン 2.2: .NET クライアント・アプリケーションの作成](#)

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる .NET クライアント・アプリケーションが組み込まれています。

[レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成](#)

Java クライアントと .NET クライアントの両方が同じデータ・グリッドを更新できるエンタープライズ・データ・グリッド・アプリケーションを作成するには、ご使用のクラスが互換性を持つようにする必要があります。入門用サンプル・アプリケーションでは、.NET サンプル・アプリケーションが Java デフォルトと一致する別名を持っています。

[< 前へ | 次へ >](#)

入門チュートリアル・レッスン 2.1: Java クライアント・アプリケーションの作成

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる Java クライアント・アプリケーションが組み込まれています。

[wxs_install_root/ObjectGrid/gettingstarted/client/src/](#) ディレクトリーにある `Client.java` ファイルは、カタログ・サーバーへの接続方法、ObjectGrid インスタンスの取得方法、および ObjectMap API の使用法を示したクライアント・プログラムです。ObjectMap API は、データをキー値ペアとして保管し、リレーションシップを持たないオブジェクトのキャッシングに適しています。以下のステップでは、`Client.java` ファイルの内容を検討します。

1. `ClientClusterContext` インスタンスを取得することで、カタログ・サービスに接続します。

カタログ・サーバーに接続するには、`ObjectGridManager` API の `connect` メソッドを使用します。以下のコード・スニペットは、カタログ・サーバーへの接続方法と `ClientClusterContext` インスタンスの取得方法を示します。

```
ClientClusterContext ccc =
ObjectGridManagerFactory.getObjectGridManager().connect(cep, null, null);
```

`connect` メソッドは、接続が成功するまでリスト内の各アプライアンスに接続しようとします。他のアプライアンスの 1 つが応答しなければ、自動フェイルオーバーが提供されます。

カタログ・サーバーへの接続が成功すれば、`connect` メソッドは `ClientClusterContext` インスタンスを戻します。この `ClientClusterContext` インスタンスは、`ObjectGridManager` API から `ObjectGrid` を取得するのに必要です。

2. `ObjectGrid` インスタンスを取得します。

`ObjectGrid` インスタンスを取得するには、`ObjectGridManager` API の `getObjectGrid` メソッドを使用します。`getObjectGrid` メソッドは、`ClientClusterContext` インスタンスと、データ・グリッド・インスタンスの名前との両方を必要とします。`ClientClusterContext` インスタンスは、カタログ・サーバーへの接続中に取得されます。データ・グリッド・インスタンスの名前は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。以下のコード・スニペットは、`ObjectGridManager` API の `getObjectGrid` メソッドを呼び出すことによってデータ・グリッドを取得する方法を示します。

```
ObjectGrid grid = ObjectGridManagerFactory.getObjectGridManager().getObjectGrid(ccc,
"my_simple_data_grid");
```

3. 必要なセキュリティー資格情報を設定します。

クライアント・セキュリティー構成を作成するほか、アプリケーションに提供するユーザー名とパスワードを使用して資格情報生成プログラムを作成します。使用するユーザー名とパスワードには、アプライアンス上のデータ・グリッドにアクセスするための権限が必要です。権限があるユーザーの作成については、[ユーザーおよびグループの管理](#)を参照してください。

```
// Creates a ClientSecurityConfiguration object using the specified
file
ClientSecurityConfiguration clientSC =
ClientSecurityConfigurationFactory.getClientSecurityConfiguration();
clientSC.setSecurityEnabled(true);
// Creates a CredentialGenerator using the passed-in user and
password.
CredentialGenerator credGen = new
UserPasswordCredentialGenerator(username,password);
clientSC.setCredentialGenerator(credGen);
return clientSC;
```

4. `Session` インスタンスを取得します。

取得した `ObjectGrid` インスタンスから、`Session` を取得することができます。`Session` インスタンスは、`ObjectMap` インスタンスの取得とトランザクション区分の実行のために必要です。以下のコード・スニペットは、`ObjectGrid` API の `getSession` メソッドを呼び出すことによって `Session` インスタンスを取得する方法を示します。

```
Session sess = grid.getSession();
```

5. ObjectMap インスタンスを取得します。

Session を取得した後、Session API の `getMap` メソッドを呼び出すことによって、Session インスタンスから ObjectMap インスタンスを取得することができます。`getMap` メソッドに渡す マップ・インスタンス名は、ユーザー・インターフェースで作成したデータ・グリッドと同じ名前です。以下のコード・スニペットは、Session API の `getMap` メソッドを呼び出すことによって ObjectMap を取得する方法を示します。

```
ObjectMap map1 = sess.getMap("my_simple_data_grid");
```

前の例は、データ・グリッドにちなんで名前が付けられるデフォルト・マップ・インスタンスを使用しています。次の例のように、新規マップ名を指定することもできます。

```
ObjectMap map2 = sess.getMap("my_simple_data_grid.CT.P");  
ObjectMap map3 = sess.getMap("my_new_map.NONE");
```

`my_simple_data_grid.CT.P` マップは作成時除去と ペシミスティック・ロックを使用するマップです。`my_new_map.NONE` マップには、除去もロック設定もありません。詳しくは、[動的マップの構成オプション](#)を参照してください。

6. ObjectMap メソッドを使用します。

ObjectMap インスタンスを取得した後、ObjectMap API を使用できます。ObjectMap インターフェースはトランザクション・マップであり、Session API の `begin` メソッドおよび `commit` メソッドを使用したトランザクション区分を必要とすることに注意してください。アプリケーションに明示的なトランザクション区分がない場合、ObjectMap 操作は自動コミット・トランザクションで実行します。

使用する鍵は、`java.lang.String` または `Integer` といった既存の Java 型にできます。値は、任意の直列化可能オブジェクト・タイプから構成できます。

- 以下のコード・スニペットは、自動コミット・トランザクションでの ObjectMap API の使用方法を示しています。

```
map1.insert(key1, value1);
```

- 一度に 1 つの区画でトランザクションを実行することもできれば、複数の区画でトランザクションを実行することもできます。単一の区画でトランザクションを実行するには、1 フェーズ・コミット・トランザクションを使用します。

```
sess.setTxCommitProtocol(TxCommitProtocol.ONEPHASE);  
sess.begin();  
map1.insert(k, v);  
sess.commit();
```

複数の区画でトランザクションを実行するには、2 フェーズ・コミット・トランザクションを使用します。

```
sess.setTxCommitProtocol(TxCommitProtocol.TWOPHASE);  
sess.begin();  
map1.insert(k, v);  
sess.commit();
```

7. オプション: セッションを閉じます。 Session 操作および ObjectMap 操作がすべて完了したら、`Session.close()` メソッドを使用してセッションを閉じます。このメソッドを実行すると、セッションで使用されていたリソースが返されます。

```
sess.close();
```

結果として、後続の `getSession()` メソッド呼び出しはより早く戻り、ヒープでの Session オブジェクトが少なくなります。

関連概念:

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連タスク:

[Java API 資料へのアクセス](#)

関連情報:

[API 資料](#)

レッスンのチェックポイント

このレッスンでは、データ・グリッドの操作を実行するシンプルなクライアント・アプリケーションを作成する方法について学習しました。

[< 前へ](#) | [次へ >](#)

入門チュートリアル・レッスン 2.2: .NET クライアント・アプリケーションの作成

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる .NET クライアント・アプリケーションが組み込まれています。

- WebSphere® eXtreme Scale クライアント for .NET がインストールされていなければなりません。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストール](#)を参照してください。
- このサンプルのプロジェクト・ファイルは Microsoft Visual Studio 2010 以降と連動します。これより前のバージョンの Microsoft Visual Studio を使用している場合は、独自のプロジェクト・ファイルを作成する必要があります。

以下の目的で .NET 入門用サンプル・アプリケーションを使用することができます。

- WebSphere eXtreme Scale クライアント for .NET が正しくインストールされていることを確認するため。
- データ・グリッドと通信する .NET クライアント用アプリケーションを作成する方法を学んでカスタム・アプリケーションを作成できるようになるため。このサンプルでは、リモート・カタログ・サーバー上のデータ・グリッドに接続する方法を示します。対話モードは、GridMapPessimisticTx マップを使用して手動トランザクションを実行する方法を示します。コマンド行モードは、GridMapPessimisticAutoTx マップを使用した自動コミット・トランザクションを示します。
- Java™ 入門用サンプルと相互運用する方法を学習するため。これらのサンプル・アプリケーションは両方とも、項目を TestKey/TestValue のペアでデータ・グリッドに保管します。 .NET サンプルには、シリアライゼーションおよびデシリアライゼーション用の固有の ID を作成する ClassAlias および FieldAlias 属性があります。Java クライアント・アプリケーションからキー挿入操作が実行されると、.NET クライアントは挿入されたキーに対して取得操作を実行することによって値を取得することができます。

.NET 入門用サンプル・アプリケーションには以下の制約があります。

- ペシミスティック・ロックのみがサポートされます。
- 2 フェーズ・コミット操作はサポートされていません。操作をコミットできる区画は 1 つのみです。複数の区画を伴うコミットを実行すると、MultiplePartitionWriteException 例外が発生します。
- サンプルでは、ヌル値はサポートされません。 .NET API はヌル値を許可しますが、ヌル可能タイプを使用する必要があります。

SimpleClient.csproj プロジェクト・ファイルが [net_client_home/sample/SimpleClient](#) ディレクトリーにあります。このプロジェクト・ファイルは、カタログ・サーバーに接続し、ObjectGrid インスタンスを取得し、ObjectMap API を使用する方法を示すクライアント・プログラムです。ObjectMap API は、データをキー値ペアとして保管し、リレーションシップを持たないオブジェクトのキャッシングに適しています。以下のステップには、SimpleClient.csproj ファイルの主な内容に関する情報が含まれています。Microsoft Visual Studio でこのプロジェクト・ファイルをさらに詳細に調べることができます。

このチュートリアルは、アプリケーションが対話モードで実行されるときに使用される手動トランザクション・マップである IGridMapPessimisticTx の使用を示します。アプリケーションをコマンド行モードで使用した場合は、IGridMapPessimisticAutoTx マップが使用されます。

1. IClientConnectionContext インスタンスを取得することで、カタログ・サービスに接続します。

カタログ・サーバーに接続するには、IGridManager API の connect メソッドを使用します。

```
IGridManager gm = GridManagerFactory.GetGridManager( );
ICatalogDomainInfo cdi = gm.CatalogDomainManager.CreateCatalogDomainInfo( endpoint
);
ccc = gm.Connect( cdi, "SimpleClient.properties" );
```

カタログ・サーバーへの接続が成功すれば、Connect メソッドは IClientConnectionContext インスタンスを戻します。IGridManager API からデータ・グリッドを取得するためには IClientConnectionContext インスタンスが必要です。

2. ObjectGrid インスタンスを取得します。

ObjectGrid インスタンスを取得するには、IGridManager API の GetGrid メソッドを使用します。GetGrid メソッドは、IClientConnectionContext インスタンスと、データ・グリッド・インスタンスの名前との両方を必要とします。IClientConnectionContext インスタンスは、カタログ・サーバーへの接続中に取得されます。データ・グリッド・インスタンスの名前は、objectgrid.xml ファイル内で指定されるグリッドです。

```
grid = gm.GetGrid( ccc, gridName );
```

3. map インスタンスを取得します。

IGrid API の GetGridMapPessimisticTx メソッドを呼び出すことによってマップ・インスタンスを取得することができます。マップの名前をパラメーターとして GetGridMapPessimisticTx メソッドに渡すことでマップ・インスタンスを取得します。

```
peessMap = grid.GetGridMapPessimisticTx<Object, Object>( mapName );
```

4. IGridMapPessimisticTx メソッドを使用します。

マップ・インスタンスが取得された後で、IGridMapPessimisticTx API を使用することができます。

次のコード・スニペットは、IGridMapPessimisticTx API の使用法を示しています。

- IGridMapPessimisticTx API でトランザクションを開始するには、map.Transaction.Begin() メソッドを呼び出す必要があります。このメソッドは、操作を実行できる新しいトランザクションを開始します。

```
case "begin":
    map.Transaction.Begin( );
    return 0;
```

- add メソッドは、新しいキー/値のペアを挿入します。キーが現在存在する場合は、例外がスローされません。

```
case "a":
    if( key == null ) throw new MissingParameterException( "key" );
    if( value == null ) throw new MissingParameterException( "value" );
    map.Add( key, value );
    Console.WriteLine( "SUCCESS: Added key '{0}' with value '{1}',
        partitionId={2}", key, value, partitionId );
    return 0;
```

- put メソッドは、キー/値のペアを挿入または更新します。

```
case "p":
    if( key == null ) throw new MissingParameterException( "key" );
    if( value == null ) throw new MissingParameterException( "value" );
    map.Put( key, value );
    Console.WriteLine( "SUCCESS: Put key '{0}' with value '{1}',
        partitionId={2}", key, value, partitionId );
    return 0;
```

- replace メソッドは、既存のキー/値のペアを置き換えます。項目が存在しない場合には、例外がスローされます。

```
case "r":
    if( key == null ) throw new MissingParameterException( "key" );
    if( value == null ) throw new MissingParameterException( "value" );
    map.Replace( key, value );
    Console.WriteLine( "SUCCESS: Replaced key '{0}' with value '{1}',
        partitionId={2}", key, value, partitionId );
    return 0;
```

- remove メソッドは、キー/値のペアを削除します。

```
case "d":
    if( key == null ) throw new MissingParameterException( "key" );
    map.Remove( key );
    Console.WriteLine( "SUCCESS: Deleted value with key '{0}',
        partitionId={1}", key, partitionId );
    return 0;
```

- get メソッドは、与えられたキーの値を取得します。

```
case "g":
    if( key == null ) throw new MissingParameterException( "key" );
    value = ( TestValue )map.Get( key );
    if( value != null )
    {
        Console.WriteLine( "SUCCESS: Value is '{0}',
```

```
partitionId={1}", value, partitionId );
}
else
{
    Console.WriteLine( "FAILED: Key not found" );
}
return 0;
```

- コミットする前の操作で実行した操作を取り消したい場合は、rollback メソッドを使用します。

```
case "rollback":
    map.Transaction.Rollback( );
    return 0;
```

- commit メソッドは、トランザクションで完了した操作をコミットします。

```
case "commit":
    map.Transaction.Commit( );
    return 0;
```

関連タスク:

[.NET](#) [.NET 開発環境の設定](#)

[.NET](#) [WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)

レッスンのチェックポイント

このレッスンでは、データ・グリッド操作を実行する簡単な .NET クライアント・アプリケーションを作成する方法について学習しました。

[< 前へ](#) | [次へ >](#)

[< 前へ | 次へ >](#)

レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成

Java™ クライアントと .NET クライアントの両方が同じデータ・グリッドを更新できるエンタープライズ・データ・グリッド・アプリケーションを作成するには、ご使用のクラスが互換性を持つようにする必要があります。入門用サンプル・アプリケーションでは、.NET サンプル・アプリケーションが Java デフォルトと一致する別名を持っています。

クラス別名属性とフィールド別名属性を .NET アプリケーションに追加します。クラス別名を .NET アプリケーション、Java アプリケーション、またはその両方に追加することができます。 .NET サンプルには Java デフォルトと一致する別名があるため、Java アプリケーションは別名を必要としません。 TestKey.cs および TestValue.cs ファイルが [net_client_home/sample/SimpleClient](#) ディレクトリーにあります。

図 1. TestKey.cs ファイル内のクラス別名属性

```
[ClassAlias( "com.ibm.websphere.xs.sample.gettingstarted.model.TestKey" )]
```

図 2. TestValue.cs ファイル内のクラス別名属性

```
[ClassAlias( "com.ibm.websphere.xs.sample.gettingstarted.model.TestValue" )]
```

関連概念:

[ClassAlias および FieldAlias 注釈](#)

関連タスク:

[Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

レッスンのチェックポイント

クラス属性を .NET 入門用アプリケーションに追加しました。その結果、Java 入門用アプリケーションと相互運用してエンタープライズ・データ・グリッドを作成できるようになりました。

[< 前へ | 次へ >](#)

[< 前へ | 次へ >](#)

モジュール 3: データ・グリッド内でのサンプル・アプリケーションの実行

アプライアンス上のデータ・グリッドに対してサンプル・クライアント・アプリケーションを実行することができます。

クライアント・アプリケーションをどのように実行するかは、Java サンプルを実行しているか、または .NET サンプルを実行しているかによって異なります。

学習目標

このモジュールのレッスンを完了すると、以下のタスクの実行方法が理解できるようになります。

- **Java** [Java 入門用サンプル・クライアント・アプリケーションの実行](#)
- **.NET** [.NET サンプル・クライアント・アプリケーションの実行](#)

Java サンプル・アプリケーションと .NET サンプル・アプリケーションを別々に実行するほかに、これらを同じデータ・グリッド上で同時に実行することもできます。例えば、.NET アプリケーションを使用して値をデータ・グリッドに挿入し、次いで Java アプリケーションを使用してその値を取得することができます。このシナリオでは、エンタープライズ・データ・グリッドを実行します。

このモジュールのレッスン

[入門チュートリアル・レッスン 3.1: Java 入門用サンプル・クライアント・アプリケーションの実行](#)

以下のステップを使用して、データ・グリッドと対話する Java クライアントを実行します。

[入門チュートリアル・レッスン 3.2: .NET サンプル・クライアント・アプリケーションの実行](#)

以下のステップを使用して、データ・グリッドと対話する WebSphere eXtreme Scale Client for .NET アプリケーションを実行します。この例では、カタログ・サーバー、コンテナー・サーバー、およびクライアントがすべて単一のサーバー上で実行されます。

[< 前へ | 次へ >](#)

[< 前へ](#) | [次へ >](#)

入門チュートリアル・レッスン 3.1: Java 入門用サンプル・クライアント・アプリケーションの実行

以下のステップを使用して、データ・グリッドと対話する Java クライアントを実行します。

[wxs_install_root/ObjectGrid/gettingstarted/env.bat|sh](#) ファイルを編集します。このファイルはクライアントによって自動的に呼び出されます。ファイルには、以下の情報が含まれています。

```
SET CATALOGSERVER_HOST=<xc10_hostname>
SET CATALOGSERVER_PORT=2809
SET GRID_NAME=my_simple_data_grid
SET MAP_NAME=my_map.P
```

アプライアンス上のデータ・グリッドに接続するには、**CATALOGSERVER_HOST** および **CATALOGSERVER_PORT** 変数を更新する必要があります。指定すべきカタログ・サーバーが、作成した単純データ・グリッドのユーザー・インターフェースのページに表示されます。「データ・グリッド」 > 「単純データ・グリッド」 > 「**my_simple_data_grid**」をクリックし、「カタログ・サービス」フィールドにある値を使用します。**GRID_NAME** プロパティの値は、作成したデータ・グリッドの名前と一致しなければなりません。**MAP_NAME** プロパティは、存続時間 (TTL) 設定およびペシミスティック・ロックのないマップを作成します。動的マップの命名については、[動的マップの構成オプション](#)を参照してください。

- クライアントを対話モードで実行します。コマンド行ウィンドウから、次のいずれかのコマンドを実行します。

- **UNIX** | **Linux** `./runclient.sh`
- **Windows** `runclient.bat`

1. トランザクションを開始します。トランザクションに対して 1 フェーズ・コミット操作または 2 フェーズ・コミット操作を使用することができます。1 フェーズ・コミットの場合は、トランザクションが単一の区画に書き込まなければなりません。異なる区画に配置される複数のキーをトランザクション時に挿入すると、コミット時にトランザクションが失敗します。2 フェーズ・コミットを使用すれば、単一のトランザクションで複数の区画に書き込まれるようにすることができます。

- 1 フェーズ・コミット・トランザクションを開始します。

```
begin
```

- 2 フェーズ・コミット・トランザクションを開始します。

```
begin2pc
```

2. 値を挿入します。

```
> i key1 helloWorld
SUCCESS: Inserted TestValue [value=helloWorld] with key TestKey [key=key1], partitionId=6
```

3. 挿入した値を取得します。

```
> g key1
Value is TestValue [value=helloWorld], partitionId=6
```

4. 値を更新します。

```
> u key1 goodbyeWorld
SUCCESS: Updated key TestKey [key=key1] with value TestValue [value=goodbyeWorld], partitionId=6
```

5. トランザクションをロールバックします。トランザクションをロールバックすると、このトランザクションに関連するすべての操作がキャンセルされます。

```
> rollback
```

6. ロールバック操作をテストするには、キーの取得を再び試みます。トランザクションをロールバックしたので、キーが存在していません。

```
> g key1
```

7. 値を挿入します。

```
> i key1 helloWorld  
SUCCESS: Inserted TestValue [value=helloWorld] with key TestKey [key=key1], p  
art  
itionId=6
```

8. 値をコミットします。トランザクションをコミットした後から変更をロールバックすることはできません。

```
> commit
```

9. 挿入した値を削除します。

```
> d key1  
SUCCESS: Deleted value with key TestKey [key=key1], partitionId=6
```

10. テスト項目をいくつか挿入します。例えば、0 から 999 までの番号が付けられた 1000 個のキーおよび値を挿入するには、次のコマンドを使用します。

```
> n 1000
```

- クライアントをコマンド行モードで実行します。クライアント・アプリケーションを実行するスクリプトを作成したい場合は、コマンド行モードが役立ちます。対話モードで実行するコマンドと同じコマンドを実行することができます。コマンド行モードの構文の例を以下に示します。

- **UNIX | Linux**

```
./runclient.sh i "key1" "helloWorld"
```

- **Windows**

```
runclient.bat i "key1" "helloWorld"
```

レッスンのチェックポイント

学習した内容

このレッスンでは、以下を学習しました。

- データ・グリッドに対してデータの挿入、取得、更新、および削除を行う Java サンプル・クライアント・アプリケーションを実行する方法。

[< 前へ](#) | [次へ >](#)

入門チュートリアル・レッスン 3.2: .NET サンプル・クライアント・アプリケーションの実行

以下のステップを使用して、データ・グリッドと対話する WebSphere® eXtreme Scale Client for .NET アプリケーションを実行します。この例では、カタログ・サーバー、コンテナ・サーバー、およびクライアントがすべて単一のサーバー上で実行されます。

WebSphere eXtreme Scale Client for .NET は 1 フェーズ・コミットのみをサポートします。したがって、同じトランザクションで複数の値を挿入しようと試みた場合は、それらの値が異なる区画に送られるため例外が発生します。サンプルを実行したときこのような例外が発生しないようにするため、1つの区画を使用するようにデプロイメント・ポリシー記述子 XML ファイルを変更することができます。区画数の更新について詳しくは、[入門チュートリアル・レッスン 1.1: データ・グリッドの定義](#)を参照してください。

サンプル・アプリケーションは対話モードまたはコマンド行モードで実行することができます。対話モードでは、アプリケーションは IGridMapPessimisticTx API を使用して手動データ・グリッド・トランザクションを実行します。コマンド行モードでは、IGridMapPessimisticAutoTx API を使用して自動データ・グリッド・トランザクションを実行します。

サンプルは対話モードまたはコマンド行モードで実行することができます。

- サンプル・クライアント・アプリケーションを対話モードで実行します。
 1. シンプルなクライアント・アプリケーションを実行します。ファイルは [net_client_home](#) ¥gettingstarted¥bin¥ ディレクトリーにあります。サンプルを対話モードで実行するには、次のコマンドを実行します。

```
SimpleClient.exe -i [-h <hostname:port>] [-g <gridname>] [-m <mapname>]
```

-h <hostname:port>

接続先のカタログ・サーバーのホスト名とポートを指定します。指定すべきカタログ・サーバーが、作成した単純データ・グリッドのユーザー・インターフェースのページに表示されます。「データ・グリッド」 > 「単純データ・グリッド」 > 「**my_simple_data_grid**」をクリックし、「カタログ・サービス」フィールドにある値を使用します。

-g <gridname>

使用するデータ・グリッドの名前を指定します。ペシミスティック・ロック・ストラテジーでマップを使用する必要があります。このチュートリアルでは、my_simple_data_grid、またはアプライアンス・ユーザー・インターフェースでデータ・グリッドを作成したときに指定した名前を使用してください。値を指定しなかった場合は、Grid データ・グリッドが使用されます。

-m <mapname>

使用するマップの名前を指定します。このチュートリアルでは、my_map.NONE.P を指定してください。値を指定しなかった場合は、エラーが発生します。マップ名について詳しくは、[動的マップの構成オプション](#)を参照してください。

パラメーターなしでアプリケーションを実行した場合は、アプリケーションのヘルプが表示されます。

2. 使用可能なコマンドのリストを表示します。

```
Enter a command: help
This program executes simple CRUD operations on a map.
  a - Adds a value with the specified key. If the key already exists,
      DuplicateKeyException is thrown
  p - Adds a value with the specified key, replacing the entry if it
      already exists
  r - Replaces the value of the specified key. If the key does not exist,
      a CacheKeyNotFound exception is thrown
  g - Retrieve and display the value of the specified key
  d - Deletes the key
  gp - Gets the partition id for the key
  ck - Checks if the map contains the key
  h - Display help
begin - Begin manual transaction
commit - Commit transactions
rollback - Rollback transactions
exit - Exit program
```

3. トランザクションを開始します。データ・グリッドでコマンドを実行するためにはトランザクションを開始する必要があります。トランザクションを開始しなかった場合は、NoActiveTransactionException 例外が発生します。

```
Enter a command: begin
```

4. データ・グリッドにデータを追加します。

```
Enter a command: a key1 value1
SUCCESS: Added 'TestKey [key=key1]' with value 'TestValue [value=value1]',
partitionId=6
```

5. 値を検索して表示します。

```
Enter a command: g key1
SUCCESS: Value is 'TestValue [value=value1]', partitionId=6
```

この例では、value1 が戻されます。

6. キーを更新します。put コマンドを使用します。このコマンドは、指定されたキーを持つ値を追加するもので、既存の値が存在する場合はそれを置き換えます。

```
Enter a command: p key1 value2
SUCCESS: Put key 'TestKey [key=key1]' with value 'TestValue [value=value2]',
partitionId=6
Enter a command: g key1
SUCCESS: Value is 'TestValue [value=value2]', partitionId=6
```

7. キーを置き換えます。replace コマンドは、指定されたキーの値を置き換えます。キーが存在しない場合は、CacheKeyException 例外が発生します。

```
Enter a command: r key1 value3
SUCCESS: Replaced key 'TestKey [key=key1]' with value 'TestValue [value=value
3]'
, partitionId=6
```

8. トランザクションをロールバックし、キーと値の再表示を試みます。トランザクションは、コミットする前であればいつでもロールバックすることができます。

```
Enter a command: rollback
Enter a command: begin
Enter a command: g key1
FAILED: Key not found
```

get コマンドを実行すると、キーが見つからなかったことを示すエラーが出されます。

9. キーと値をデータ・グリッドにコミットします。

```
Enter a command: begin
Enter a command: a key2 value2
SUCCESS: Added 'TestKey [key=key2]' with value 'TestValue [value=value2]',
partitionId=7
Enter a command: commit
```

10. キーの区画 ID を取得します。

```
Enter a command: begin
Enter a command: gp key2
SUCCESS: partitionId=7
```

11. キーのマップを確認します。

```
Enter a command: ck key2
SUCCESS: The map contains key 'TestKey [key=key2]'
Enter a command: ck key3
SUCCESS: The map does NOT contain key 'TestKey [key=key3]'
```

12. キーを削除し、終了します。

```
Enter a command: begin
Enter a command: d key2
SUCCESS: Deleted value with key 'TestKey [key=key2]', partitionId=7
Enter a command: commit
Enter a command: exit
```

- クライアントをコマンド行モードで実行します。コマンド行モードでは、IGridMapPessimisticAutoTx API を使用して自動データ・グリッド・トランザクションを実行します。このモードを使用するには、コマンド行でアクションを渡します。クライアント・アプリケーションを実行するスクリプトを作成したい場合は、コマンド行モードが役立ちます。対話モードで実行するコマンドと同じコマンドを実行することができます。コマンド行モードの構文の例を以下に示します。

```
SimpleClient [-h <host:port>] [-g <grid_name>] [-m <mapname>] <a | p | r | g | d>
<key> [<value>]
```

関連タスク:

[.NET .NET API を使用したデータ・グリッド・アプリケーションの開発](#)
[WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)

レッスンのチェックポイント

このレッスンでは、以下を学習しました。

- データ・グリッドに対してオブジェクトの挿入、取得、更新、および削除を行う .NET サンプル・クライアント・アプリケーションを実行する方法。

[< 前へ | 次へ >](#)

[< 前へ](#)

入門チュートリアル・レッスン 4: 環境のモニター

xscmd ユーティリティおよび Web コンソールのツールを使用して、データ・グリッド環境をモニターできます。

関連タスク:

[ユーザー・インターフェースでのデータ・グリッドのモニター](#)

ユーザー・インターフェースでのデータ・グリッドのモニター

ユーザー・インターフェースで、環境内のデータ・グリッドの全体的なパフォーマンスを表示することができます。ユーザー・インターフェースのモニター・セクションには、アプライアンス上のすべてのデータ・グリッドの概要、個々のデータ・グリッドの概要、および個々のデータ・グリッドの詳細レポートがあります。

ユーザー・インターフェースでのデータ・グリッドのモニターについて詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

xscmd ユーティリティによるモニター

1. オプション: クライアント認証が使用可能な場合: コマンド行ウィンドウをクライアントのインストール済み環境で開きます。コマンド行で、適切な環境変数を設定します。
2. [wxs_home](#)/bin ディレクトリーに移動します。

```
cd wxs\_home/bin
```

3. 各種コマンドを実行して、環境に関する情報を表示します。
 - Grid データ・グリッドと mapSet マップ・セットのすべてのオンライン・コンテナー・サーバーを表示します。

```
xscmd -c showPlacement -g Grid -ms mapSet
```

- データ・グリッドのルーティング情報を表示します。

```
xscmd -c routetable -g Grid
```

- データ・グリッド内のマップ・エントリーの数を表示します。

```
xscmd -c showMapSizes -g Grid -ms mapSet
```

レッスンのチェックポイント

このレッスンでは、以下を学習しました。

- データ・グリッドおよびサーバーの統計をモニターする方法。

[< 前へ](#)

DataPower XC10 アプライアンス環境の計画

DataPower® XC10 アプライアンスをトポロジーに組み込む前に、環境が次の前提条件およびソフトウェア要件を満たしている必要があります。

システム要件

ご使用の環境に WebSphere® DataPower XC10 アプライアンスを組み込むには、以下の要件がインストールされている必要があります。

ディレクトリー規則

`wxs_install_root` や `wxs_home` など、参照が必要な特別のディレクトリーに対して、資料全体で、次のディレクトリー規則が使用されます。インストール中、およびコマンド行ツールの使用時も含めて、さまざまなシナリオで、これらのディレクトリーにアクセスします。

ネットワーク・ポート

ファイアウォール内で WebSphere DataPower XC10 アプライアンスを使用している場合、次のポートを経由した通信を使用可能に設定する必要があります。

IBM WebSphere DataPower XC10 アプライアンス インストールの前提条件

IBM® WebSphere DataPower XC10 アプライアンスをインストールおよび構成するには、特定のハードウェア、ソフトウェア、ラック、およびツールの要件を満たす必要があります。この前提条件一式を使用して、IBM WebSphere DataPower XC10 アプライアンスのインストール、構成、および使用の計画を立てます。

アプライアンスの仕様およびフィーチャー

WebSphere DataPower XC10 アプライアンス バージョン 2.5 は Type 7199-92x ハードウェア上でサポートされます。ハードウェア Type 9235-92x にはもともとバージョン 1.0 が付属しており、このハードウェアではバージョン 2.1 までしかサポートされません。

製品のインターオペラビリティ

WebSphere DataPower XC10 アプライアンスは、他の IBM 製品とのインターオペラビリティについてテスト済みです。

.NET 2.5+ Microsoft .NET に関する考慮事項

WebSphere eXtreme Scale には 2 つの .NET 環境 (開発環境とランタイム環境) が存在します。これらの環境はそれぞれ固有の要件を持っています。

システム要件

ご使用の環境に WebSphere® DataPower® XC10 アプライアンス を組み込むには、以下の要件がインストールされている必要があります。

ハードウェア要件

初期構成を実行するには、シリアル接続を使用する必要があります。ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC から、 WebSphere DataPower XC10 アプライアンス アプライアンスのシリアル・ポートに対するシリアル接続が必要です。シリアル接続を行う場合は、付属の RJ45 から DB-9 のシリアル・ヌル・モデム・ケーブルを使用してください。

注: ターミナルにシリアル・ポートが装備されていない場合は、付属の USB シリアル変換ケーブルを使用してください。

無人インストールまたはリモート・インストールは、ネットワーク接続された端末サーバー経由で行うことができます。

ソフトウェア要件

WebSphere DataPower XC10 アプライアンスは、IBM 提供のソフトウェアに付属しています。アプライアンスに他のソフトウェアをインストールすることはできません。

Web ブラウザー要件

ユーザー・インターフェースでは、以下の Web ブラウザーがサポートされます。

- Mozilla Firefox、バージョン 3.5.x 以降
- Microsoft Internet Explorer バージョン 7 以降

サポートされるソフトウェア

次のソフトウェアは、WebSphere DataPower XC10 アプライアンスと連携して使用できます。

- WebSphere eXtreme Scale クライアント 8.5
- WebSphere eXtreme Scale クライアント バージョン 8.6
- WebSphere Application Server バージョン 6.1.0.41 以降
- WebSphere Application Server バージョン 7.0.0.21 以降
- WebSphere Application Server バージョン 8.0.0.2 以降
- WebSphere Application Server バージョン 8.5.0.0 以降
- WebSphere Application Server バージョン 8.5.5.0

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

ディレクトリー規則

`wxs_install_root` や `wxs_home` など、参照が必要な特別のディレクトリーに対して、資料全体で、次のディレクトリー規則が使用されます。インストール中、およびコマンド行ツールの使用時も含めて、さまざまなシナリオで、これらのディレクトリーにアクセスします。

`wxs_install_root`

`wxs_install_root` ディレクトリーは、WebSphere® eXtreme Scale 製品ファイルがインストールされているルート・ディレクトリーです。`wxs_install_root` ディレクトリーは、試用版のアーカイブが解凍されたディレクトリー、または WebSphere eXtreme Scale 製品がインストールされているディレクトリーの可能性があります。

- 試用版を解凍した場合の例:

例: `/opt/IBM/WebSphere/eXtremeScale`

- WebSphere eXtreme Scale がスタンドアロン・ディレクトリーにインストールされている場合の例:

UNIX 例: `/opt/IBM/eXtremeScale`

Windows 例: `C:\Program Files\IBM\WebSphere\eXtremeScale`

- WebSphere eXtreme Scale が WebSphere Application Server に統合されている場合の例:

例: `/opt/IBM/WebSphere/AppServer`

`wxs_home`

`wxs_home` ディレクトリーは、WebSphere eXtreme Scale 製品ライブラリー、サンプル、およびコンポーネントのルート・ディレクトリーです。このディレクトリーは、試用版を解凍した場合は、`wxs_install_root` ディレクトリーと同じです。スタンドアロンのインストール済み環境の場合、`wxs_home` ディレクトリーは、`wxs_install_root` ディレクトリー内の `ObjectGrid` サブディレクトリーです。WebSphere Application Server に統合されているインストール済み環境の場合、このディレクトリーは、`wxs_install_root` ディレクトリー内の `optionalLibraries/ObjectGrid` ディレクトリーです。

- 試用版を解凍した場合の例:

例: `/opt/IBM/WebSphere/eXtremeScale`

- WebSphere eXtreme Scale がスタンドアロン・ディレクトリーにインストールされている場合の例:

UNIX 例: `/opt/IBM/eXtremeScale/ObjectGrid`

Windows 例: `wxs_install_root\ObjectGrid`

- WebSphere eXtreme Scale が WebSphere Application Server に統合されている場合の例:

例: `/opt/IBM/WebSphere/AppServer/optionalLibraries/ObjectGrid`

`was_root`

`was_root` ディレクトリーは、WebSphere Application Server インストール済み環境のルート・ディレクトリーです。

例: `/opt/IBM/WebSphere/AppServer`

.NET `net_client_home`

`net_client_home` ディレクトリーは、.NET クライアント・インストール済み環境のルート・ディレクトリーです。

例: `C:\Program Files\IBM\WebSphere\eXtreme Scale .NET Client`

`java_home`

`java_home` は、Java™ Runtime Environment (JRE) インストール済み環境のルート・ディレクトリーです。

UNIX 例: `/opt/IBM/WebSphere/eXtremeScale/java`

Windows 例: `wxs_install_root\java`

`samples_home`

`samples_home` は、チュートリアルに使用するサンプル・ファイルを解凍したディレクトリーです。

UNIX 例: `wxs_home/samples`

Windows 例: `wxs_home\samples`

dvd_root

dvd_root ディレクトリーは、製品が含まれた DVD のルート・ディレクトリーです。

例: *dvd_root/docs/*

user_home

user_home ディレクトリーは、ユーザー・ファイル (セキュリティー・プロファイルなど) が保管されている場所です。

Windows c:¥Documents and Settings¥*user_name*

UNIX /home/*user_name*

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

ネットワーク・ポート

ファイアウォール内で WebSphere® DataPower® XC10 アプライアンスを使用している場合、次のポートを経由した通信を使用可能に設定する必要があります。

カタログ・サーバー

下記のポートはカタログ・サーバーによって使用されるものです。集合に追加される最初の 3 つのアプライアンスは、それぞれ 1 つのカタログ・サーバーを実行します。

ピア・ポート: カタログ・サーバー間の通信に使用されます。このポートは 6601 を使用するように設定されます。

クライアント・ポート: カタログ・サーバー間の通信に使用されます。このポートは 6602 を使用するように設定されます。

JMX サービス・ポート: 非 SSL JMX 接続のために使用されます (xscmd ユーティリティーなど)。このポートは 1099 を使用するように設定されます。

JMX コネクター・ポート: SSL JMX 接続のために使用されます (xscmd ユーティリティーなど)。このポートは 1100 を使用するように設定され、7100 から 7116 までの間のポートを選択します。

ORB リスナー・ポート: クライアントとデータ・グリッド・サーバー間の通信に使用されます。このポートは 2809 を使用するように設定されます。

CSiv2 リスナー・ポート: クライアントとデータ・グリッド・サーバー間のセキュア通信に使用されます。このポートは 7499 を使用するように設定されます。

SNMP ポート: SNMP モニターのために使用されます。このポートは 161 を使用するように設定されます。

注: WebSphere DataPower XC10 アプライアンスは、SNMP バージョン v2c をサポートします。

以下のポートは、コンテナ・サーバー間の通信に必要なものです。

DCS コア・グループ・ポート: 内部 WebSphere DataPower XC10 アプライアンス DCS 通信に使用されます。このポートは、6700 から 6716 の範囲内のポートを使用します。

ORB リスナー・ポート: クライアントとデータ・グリッド・サーバー間の通信に使用されます。このポートは、6800 から 6816 の範囲内のポートを使用します。

CSiv2 リスナー・ポート: クライアントとデータ・グリッド・サーバー間のセキュア通信に使用されます。このポートは、7500 から 7516 の範囲内のポートを使用します。

注: アプライアンスが集合内にあり、かつさまざまなファイアウォール側にある場合は、双方向通信のために、コンテナ・ポート DCS、ORB、および CSiv2 を、カタログ・ポート 6601 および 6602 と共に開く必要があります。

ブラウザー・ポート

アプライアンス・コンソールまたは REST サービスを使用する場合は、ポート 80 および 443 を開きます。集合内のすべてのアプライアンスでこれらのポートが開いていなければ、一部の構成オプションは失敗する可能性があります。

クライアント・サーバー

WebSphere eXtreme Scale クライアントは、リスナー・ポートを使用します。このポートは、`wxs_client_root` ¥properties ディレクトリー内のクライアント・プロパティー・ファイルに指定されています。

問題判別アプリケーション

問題判別アプリケーションにポート 9060 を開きます。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

IBM WebSphere DataPower XC10 アプライアンス インストールの前提条件

IBM® WebSphere® DataPower® XC10 アプライアンスをインストールおよび構成するには、特定のハードウェア、ソフトウェア、ラック、およびツールの要件を満たす必要があります。この前提条件一式を使用して、IBM WebSphere DataPower XC10 アプライアンス のインストール、構成、および使用の計画を立てます。

必要なスキル

アプライアンスをセットアップおよび管理するには、ネットワーク管理のスキルが必要です。

必要な情報

DataPower XC10 アプライアンス の基本構成を定義するためには、以下のデータを収集する必要があります。

- データ・グリッドに 1 ギガビット・イーサネット・ポートを使用しているか、10 ギガビット・イーサネット・ポートを使用しているか。使用するポートはすべて 1 ギガビット・ポート、またはすべて 10 ギガビット・ポートでなければなりません。MGMT0 の管理ポートに接続します。
- アプライアンス管理アクセス用のイーサネット・インターフェース (MGMT) の IP アドレスおよびサブネット・マスク。
- イーサネット・インターフェースのサブネットをサポートするデフォルト・ゲートウェイ (ルーター) の IP アドレス。
- ドメイン・ネーム・システム (DNS) サーバーの IP アドレス。フォワード・ルックアップとリバース・ルックアップの両方が可能となるように、DNS サーバーを構成してください。
- シリアル・インターフェースの通信パラメーター: 9600.8.n.1 (9600 ボー、8 データ・ビット、パリティなし、1 ストップ・ビット)。
- メール・サーバー情報 (E メール通知のセットアップ用)。
- (オプション) アプライアンス・サービス・アクセス用のイーサネット・インターフェース (ETH0、ETH1、および ETH2) の IP アドレスおよびサブネット・マスク。

ラック要件

DataPower XC10 アプライアンスをインストールするには、以下の基準を満たす、最小で 25 インチ (63.5 cm) の奥行きがある標準の 19 インチ (48.26 cm) ラックが必要です。

- 背面に取り付け用のコラムがある
- 正面取り付けと背面取り付けの両方がサポートされている
- 背面に取り付け用のコラムがある
- 正面取り付けと背面取り付けの両方がサポートされている

ラックの内部と周囲に以下の必要なスペースが確保されている状態にしてください。

- ラックの背後に 30 インチ (76.20 cm) 以上の空き領域がある
- アプライアンスの上下が 2 インチ (5.1 cm) 以上空いている
- イーサネット・ケーブルおよびシリアル・コンソール・ケーブル用に正面に十分な空きスペースがある

アプライアンスを取り付ける環境内の気温は、104°F (40°C) を超えてはいけません。

ツールおよびサプライ

DataPower XC10 アプライアンス をインストールするために、以下のサプライを揃えておいてください。

- 中型のプラス・ドライバー
- 2 本の標準ラックねじ (アプライアンスに同梱)
- 1 本から 4 本のネットワーク・ケーブル
- シリアル・ケーブルまたは PL-2303 USB シリアル・ケーブル

重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。

注: アプライアンスのインストール後も、ケーブルを 廃棄しないでください。今後、問題判別や保守に必要となる場合があります。

- オス DB9 コネクター付きシリアル・コンソール。このコンソールには、VT100 スタイルのコンソールなどの専用ハードウェア・デバイス、またはハイパーターミナルや Minicom などのエミュレーターが稼働する PC が考えられます。

ユーザー・インターフェースの要件

Web ユーザー・インターフェースを使用するには、以下のブラウザーのいずれかが使用できます。

- Mozilla Firefox、バージョン 3.5.x 以降
- Microsoft Internet Explorer バージョン 7 以降

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

アプライアンスの仕様およびフィーチャー

WebSphere® DataPower® XC10 アプライアンス バージョン 2.5 は Type 7199-92x ハードウェア上でサポートされます。ハードウェア Type 9235-92x にはもともとバージョン 1.0 が付属しており、このハードウェアではバージョン 2.1 までしかサポートされません。

アプライアンスのタイプを判別するには、「アプライアンス」 > 「設定」 > 「ファームウェア」とクリックします。このパネルには、アプライアンス・モデル・タイプとアプライアンスのシリアル番号が表示できます。

[アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

仕様とフィーチャーを参考に、アプライアンスを設置するために必要な物理的環境を 決定してください。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

アプライアンス Type 7199-92x の仕様およびフィーチャー

仕様とフィーチャーを参考に、アプライアンスを設置するために必要な物理的環境を決定してください。

Type 7199-92X アプライアンスについて

Type 7199-92X WebSphere® DataPower® XC10 アプライアンスには、WebSphere DataPower XC10 アプライアンスバージョン 2.0 ソフトウェアが付属しています。この新しいハードウェア・リリースでは、以前のリリースと比べて、より高速なプロセッサが搭載され、ネットワーク・ポート数とキャッシュ容量が増えました。

アプライアンスのタイプを判別するには、「アプライアンス」>「設定」>「ファームウェア」とクリックします。このパネルには、アプライアンス・モデル・タイプとアプライアンスのシリアル番号が表示できます。

仕様

表 1. Type 7199-92x アプライアンスの仕様. Type 7199 シャーシの仕様を要約します。

サイズ/重量:	
	7199
高さ	8.89 cm (3.5 インチ)
幅	42.8 cm (17.25 インチ)
奥行き	58.4 cm (23 インチ)
重量	最大: 21 kg (46.2 ポンド)
電気入力:	
正弦波	50 - 60 Hz (単相) が必要
110 ボルト AC	最小: 100 V _{RMS} 最大: 127 V _{RMS}
220 ボルト AC	最小: 200 V _{RMS} 最大: 240 V _{RMS}
電力使用	10 A (110 V AC) 5 A (220 V AC) Type 7199 アプライアンスには、720 ワット 電源モジュールが 2 つあります。両方の電源機構モジュールは同じ給電部に接続し、2 つの電源モジュール間で接地電圧に差が生じないようにしなければなりません。
環境	
気温	電源オン時: • 高度 0 m から 914.4 m (3000 フィート) 50°F から 95°F (10°C から 35°C) • 高度: 914.4 m (3000 フィート) から 2133.6 m (7000 フィート) 50°F から 89.6°F (10°C から 32°C) 最大高度: 2133.6 m (7000 フィート) 電源オフ時: 50°F から 109.4°F (10°C から 43°C) 出荷時: -40°F から 140°F (-40°C から 60°C)
湿度	8% から 80%

フィーチャー

表 2. データ・ストレージ・オプション

フィーチャー	説明
ローカル側容量	ローカル・ファイル・システム上に 16 GB のストレージ
ハード・ディスク・アレイ	2 基の 1 TB シンプル・スワップ・シリアル接続 SCSI (SAS) ハード・ディスク 容量: 2 TB
データ・グリッド	320 GB Fusion DUO Drive (240 GB キャッシュ搭載)

[Type 7199-92x 正面図](#)

正面図では、Type 7199 アプライアンスのコントロール、LED、およびコネクタを表示します。イーサネット・モジュールおよびハード・ディスク・モジュールは、Type 7199-92x アプライアンスのフロント・パネルから取り付けることができます。

[Type 7199-92x 背面図](#)

背面図では、アプライアンスの背面にある構成装置と LED を表示します。ファン・モジュールおよび電源モジュール

ールは、アプライアンスの背面に 取り付けられます。

イーサネット・ネットワーク構成

イーサネット・モジュールは、ネットワーク接続オプションを拡張します。アプライアンスには、それぞれ 2 つのイーサネット・モジュールがあります。イーサネット・モジュールは左から右に 番号が振られます。ただし、モジュールのポート数が 8 個より少ない場合、そのモジュールは 範囲内で一番小さいポート番号を使用します。

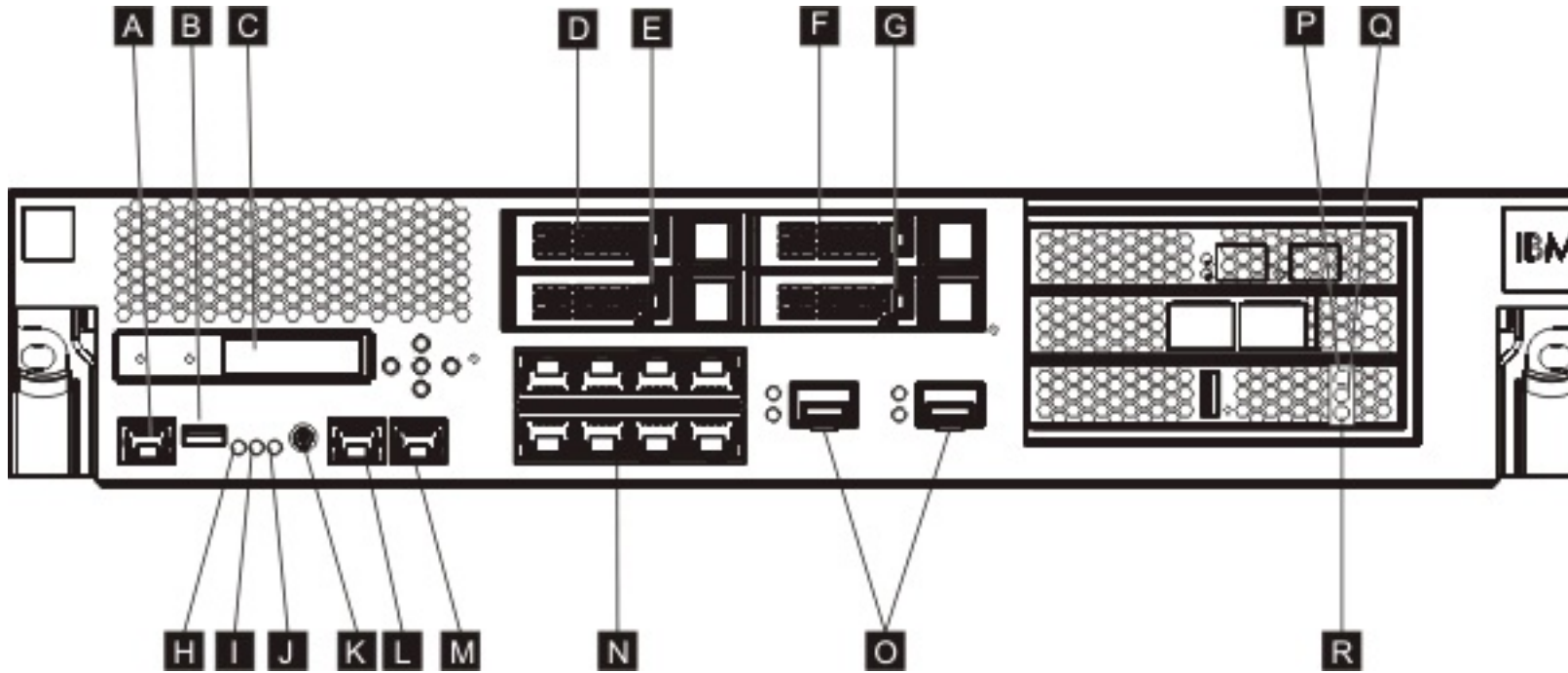
親トピック: [アプライアンスの仕様およびフィーチャー](#)

Type 7199-92x 正面図

正面図では、Type 7199 アプライアンスのコントロール、LED、およびコネクタを表示します。イーサネット・モジュールおよびハード・ディスク・モジュールは、Type 7199-92x アプライアンスのフロント・パネルから取り付けることができます。

正面図

図 1. Type 7199-92x 正面図



上図のラベルは、Type 7199-92x アプライアンスのフロント・パネルにある以下の構成装置に対応しています。

- A** コンソール・コネクタ
- B** USB ポート
- C** LCM モジュール
- D** ハード・ディスク・モジュール 2
- E** ハード・ディスク・モジュール 0
- F** ハード・ディスク・モジュール 3
- G** ハード・ディスク・モジュール 1
- H** 障害 LED
- I** 場所の特定 LED
- J** 電源 LED
- K** 電源ボタン
- L** MGT0 イーサネット・コネクタ
- M** MGT1 イーサネット・コネクタ
- N** 左方イーサネット・モジュール:
 - eth0
 - eth1
 - eth2

- eth3
- eth4
- eth5
- eth6
- eth7

O

右方イーサネット・モジュール:

- eth8
- eth9

P

琥珀色。キャッシュのエラーを示す LED 表示ライト。

Q

黄色。キャッシュの書き込みを示す LED 表示ライト。

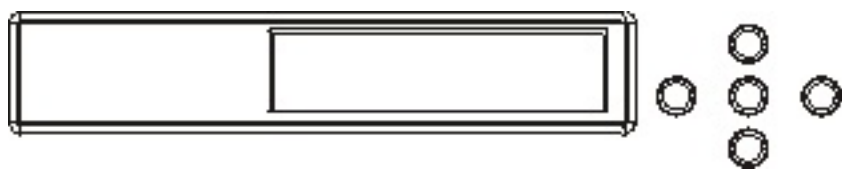
R

緑色。キャッシュの読み取りを示す LED 表示ライト。

LCD モジュール

アプライアンスのフロント・パネルには、液晶ディスプレイ (LCD) と 5 つのメニュー・ボタンからなる LCD モジュールがあります。LCD はアプライアンスのモデル・タイプとビルド番号に関する情報を示しますが、メニュー・ボタンは機能しません。

図 2. LCD モジュール



コンソール・コネクタ

すべての Type 7199 アプライアンスのフロント・パネルにコンソール・コネクタがあります。初期構成では、アプライアンスに付属の RJ45 (ISO 8877) - DB-9 (DE-9 または EIA/TIA-562 と呼ばれます) シリアル・ヌル・モデム・ケーブルを使用して、ASCII 端末 ¹ から、または 端末エミュレーション・ソフトウェアを実行している PC からアプライアンスに接続します。ケーブルの一方の端には RJ45 接続があり、ケーブルのもう一方の端には DB-9 シリアル・ヌル・モデム接続があります。RJ45 側の終端はアプライアンスに接続し、DB-9 シリアル・ヌル・モデム・ケーブル側の終端は ASCII 端末またはパーソナル・コンピュータに接続します。ケーブルをパーソナル・コンピュータに接続する場合は、USB シリアル・コンバーター・ケーブルを使用してください。

注: 初期構成の場合、アプライアンスに付属の RJ45 シリアル接続ケーブルを使用するか、下の表にリストするケーブル・ピン配列仕様に基づいてケーブルを作成することができます。シリアル・コンソール・ポートをイーサネット・ネットワークに接続するとき、イーサネット・ケーブルは使用しないでください。

表 1. シリアル・ポートのピン配列. コンソール・コネクタのシリアル・ポートのピン配列を記載します。

RJ45		DB9	
ピン番号	シグナル	ピン番号	シグナル
1	RTS	8	CTS
2	DTR	6	DSR
3	TXD	2	RXD
4	GND	5	GND
5	GND	5	GND
6	RXD	3	TXD
7	DSR	4	DTR
8	CTS	7	RTS

USB ポート

アプライアンスのフロント・パネルには、USB 2.0 デバイスに規格適合した USB インターフェースがあります。この USB コネクタは使用可能に設定されていないため、どのような接続も提供しません。

LED

すべての Type 7199 アプライアンスのフロント・パネルに、3 つのスタンドアロン LED があります。

障害 LED

クリティカル・イベントが検出されると、オレンジ色の障害 LED が点灯します。

場所の特定 LED

場所の特定 LED は青色で、ファームウェアによってアクティブにされると点灯します。この LED を点灯するかどうかは、コマンド行から制御できます。この LED は非アクティブにされるまで点灯し続けます。コマンド行インターフェースで次の **locate-led** コマンドを使用します。

- アクティブにする場合は、次のコマンドを入力します。

```
locate-led on
```

- 非アクティブにする場合は、次のコマンドを入力します。

```
locate-led off
```

電源 LED

緑色の電源 LED は、アプライアンスを給電部に接続し、アプライアンスの電源を入れると点灯します。

- 緑色の電源 LED は、アプライアンスの電源が入っていて、完全に機能しているときに点灯します。
- この LED が点灯していない場合、アプライアンスは電源が切られています。

電源ボタン

電源ボタンは、アプライアンスのフロント・パネルにあります。次の場合、電源ボタンを押します。

- アプライアンスの電源をオンにします。
- 安全なシャットダウンを開始する (既にアプライアンスの電源が入っている場合)。

電源ボタンを 5 秒間押し続けるとハードウェアの即時シャットダウンが実行されます。

注: 電源ボタンを押してアプライアンスの電源を切っても、デバイスには引き続き電流が供給されます。アプライアンスへの全電力を完全に停止するには、すべての電源コードを抜いてください。

ネットワーク・コネクタ

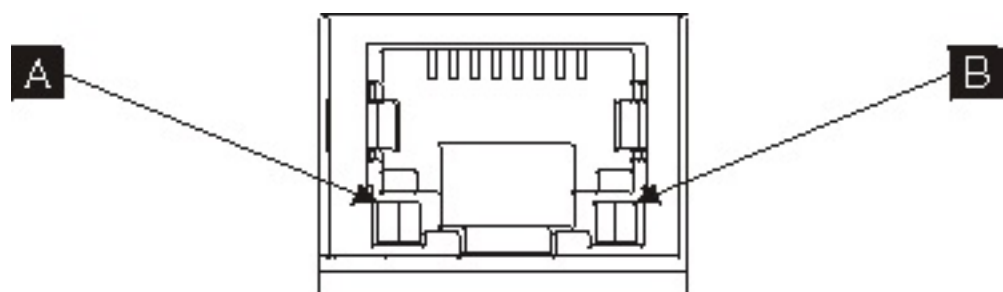
各アプライアンスのフロント・パネルには、2 つの LAN 管理イーサネット・ポートと 2 つのイーサネット・モジュールがあります。イーサネットの命名規則に関する説明については、[イーサネット・ネットワーク構成](#)を参照してください。

LAN 管理イーサネット・ポート

2 つのシステム管理イーサネット・ポートが LAN への接続を提供します。これらのポートは、デバイスへのリモート管理アクセスを可能にするものであり、データ・ポートとしては使用できません。残りのイーサネット・インターフェースは、各種 DataPower® サービスとの間のデータ・トラフィックやロギング機能进行处理できます。

ベスト・プラクティス: 着信 SNMP、SSH のネットワーク・トラフィックを処理するシステム全体の管理機能、およびイントラネット上でのユーザー・インターフェース機能には、MGT0 または MGT1 イーサネット・インターフェースを使用してください。

図 3. イーサネット・ポート LED



MGT0 イーサネット・コネクタ

このイーサネット・インターフェースで、アプライアンス上のすべてのトランザクション・データを管理できます。MGT0 イーサネット・コネクタは、IPMI over LAN (Serial over LAN を含む) もサポートします。MGT0 には関連の速度 LED とアクティビティ LED があります。

速度 LED (A)

- 緑色の LED は 1 Gbps 接続を示します。
- オレンジ色の LED は 10 Mbps または 100 Mbps 接続を示します。

アクティビティ LED (B)

- 緑色の LED は、ポートがリンクされていることを示します。
- 明滅する緑色の LED は、ポートがアクティブなことを示します。

MGT1 イーサネット・コネクタ

このイーサネット・インターフェースで、アプライアンス上のすべてのトランザクション・データを管理できま

す。MGT1 には関連のアクティビティ LED ライトと 速度 LED ライトがあります。

速度 LED (A)

- 緑色の LED は 1 Gbps 接続を示します。
- オレンジ色の LED は 10 Mbps または 100 Mbps 接続を示します。

アクティビティ LED (B)

- 緑色の LED は、ポートがリンクされていることを示します。
- 明滅する緑色の LED は、ポートがアクティブなことを示します。

イーサネット・モジュール

DataPower アプライアンスには、イーサネット接続用の 2 つのイーサネット・モジュールがあります。左方イーサネット・モジュールには 8 つの RJ45 ポートがあり、右方イーサネット・モジュールには 2 つの 10 ギガバイト Small Form Factor Pluggable (SFP+) ポートがあります。イーサネット・インターフェースの名前は、モジュール構成に依存します。

1 ギガビット・モジュールは、シールドなしの対より線を使用するイーサネットをサポートし、インターフェース規格には以下が含まれます。

- 10BASE-T
- 100BASE-TX
- 1000BASE-T

10 ギガビット・モジュールは、インターフェース・モジュールとパッチ・ケーブルを使用する Small Form Factor Pluggable (SFP+) ポートをサポートします。オートネゴシエーションは常に使用可能です。

10GBASE-SR
10GBASE-LR

左方イーサネット・モジュール

シールドなしの対より線 (RJ45) イーサネット・ポートが 8 つあります。イーサネット番号は ETH0 から ETH7 までで、それぞれ使用可能なポートの番号に関連付けられます。

右方イーサネット・モジュール

10 ギガビットの SFP ポートが 2 つあります。イーサネット番号は ETH8 から ETH9 までで、それぞれ使用可能なポートの番号に関連付けられます。

イーサネットの番号付けの説明については、[イーサネット・ネットワーク構成](#)を参照してください。

注: イーサネット・モジュールはホット・スワップ可能ではありません。このモジュールをホット・スワップすると、システムが破壊され、アプライアンスにも損傷を与える恐れがあります。

ハード・ディスク・モジュール

アプライアンスのフロント・パネルには、2.5 インチ・ハード・ディスク・モジュールが 4 つあります。アプライアンスは SAS ハード・ディスクをサポートし、各ハード・ディスク・モジュールに 2 つの LED があります。左側の LED はハード・ディスク・アクティビティをモニターし、右側の LED は潜在的な問題を示します。

- 明滅する緑色の LED は、ハード・ディスクがアクセス中であることを示します。
- 明滅するオレンジ色の LED は、ハード・ディスクに障害があったことを示します。
- LED が点灯していない場合は、ハード・ディスクがアクティブでないことを示します。

注: ハード・ディスク・モジュールはホット・スワップ可能ではありません。このモジュールをホット・スワップすると、システムが破壊される恐れがあります。

親トピック: [アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

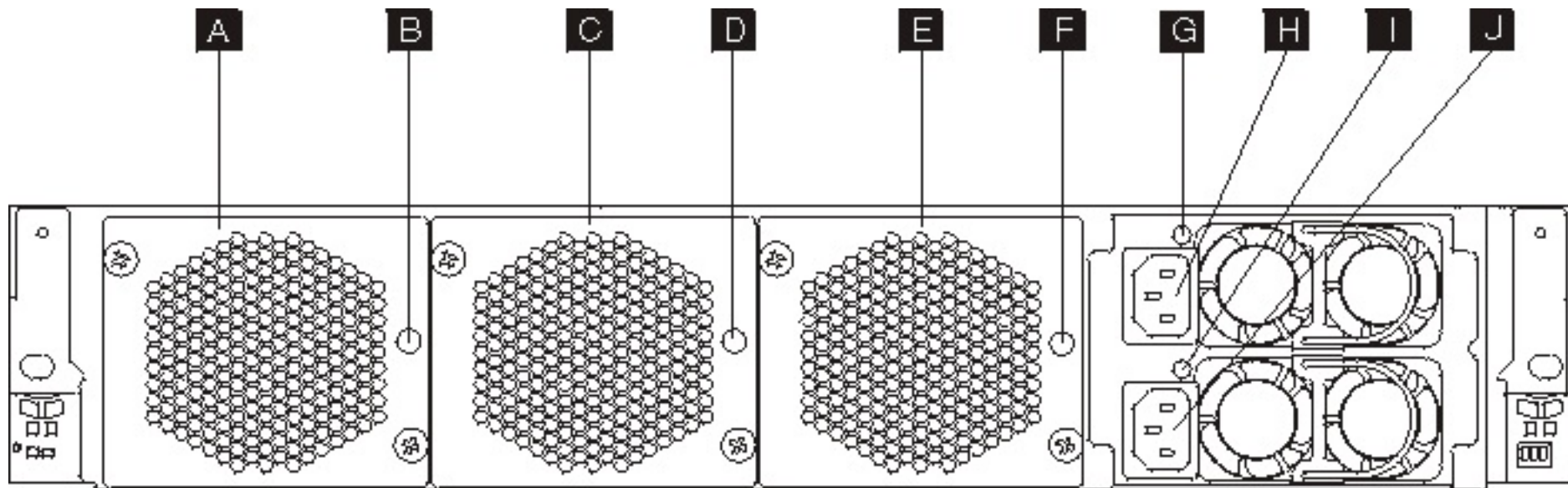
¹ ASCII データを送信 (入力) および受信 (出力) するシンプルなデバイス。

Type 7199-92x 背面図

背面図では、アプライアンスの背面にある構成装置と LED を表示します。ファン・モジュールおよび電源モジュールは、アプライアンスの背面に取り付けられます。

背面図

図 1. Type 7199-92x 背面図



上図のラベルは、Type 7199-92x アプライアンスの背面パネルにある以下の構成装置に対応しています。

- A**
ファン・モジュール 1
- B**
ファン・モジュール 1 LED
- C**
ファン・モジュール 2
- D**
ファン・モジュール 2 LED
- E**
ファン・モジュール 3
- F**
ファン・モジュール 3 LED
- G**
電源機構モジュール 1 LED
- H**
電源機構モジュール 1
- I**
電源機構モジュール 2 LED
- J**
電源機構モジュール 2

ファン・モジュール

アプライアンスには 3 つのファン・モジュールがあります。各ファン・モジュールには個別の冷却ファンがあり、ファン・モジュールごとに LED も付いています。

- オレンジ色の LED が点灯した場合は、ファン・モジュールに問題があります。
- オレンジ色の LED が点灯していない場合、ファンは正常に作動しています。

ファンの速さは、アプライアンスの温度によって変わります。温度が上がると、ファンの速さを増して、アプライアンスの温度バランスを保つようにします。

電源機構モジュール

アプライアンスには、予備電源を含む 2 つの電源機構モジュールにより電力が供給されます。1 つの電源機構モジュール

ルで、アプライアンスの動作をサポートするために十分な電力を供給可能です。電源機構モジュールはホット・スワップ可能であるため、アプライアンスを電源遮断せずに、一方の電源機構モジュールを交換できます。電源機構モジュールにはそれぞれ LED が付いています。

- オレンジ色の電源機構 LED が点灯した場合は、電源機構にエラーがあります。
- LED が点灯していない場合、電源機構は想定どおり作動しています。

注: 電源ボタンを押してアプライアンスの電源を切っても、デバイスには引き続き電流が供給されます。アプライアンスへの全電力を完全に停止するには、すべての電源コードを抜いてください。

親トピック: [アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

イーサネット・ネットワーク構成

イーサネット・モジュールは、ネットワーク接続オプションを拡張します。アプライアンスには、それぞれ 2 つのイーサネット・モジュールがあります。イーサネット・モジュールは左から右に番号が振られます。ただし、モジュールのポート数が 8 個より少ない場合、そのモジュールは範囲内で一番小さいポート番号を使用します。

番号付け規則

イーサネット・インターフェースを構成し、ネットワーク・ケーブルを取り付ける場合の番号付け規則は、次のとおりです。

- 左方モジュールの範囲は ETH0 から ETH7。
- 右方モジュールの範囲は ETH8 から ETH9。

Type 7199 接続

各イーサネット・モジュールは、以下の構成のいずれかを備えています。

- 左方イーサネット・モジュールには、1 ギガビット・イーサネット・ポートが 8 つあり、それらは RJ45 コネクターです。
- 右方イーサネット・モジュールには、10 ギガビット・イーサネット・ポートが 2 つあり、それらは Small Form Factor Pluggable (SFP+) トランシーバーです。

アプライアンスには 10 のイーサネット接続が装備されています。イーサネット・インターフェースの名前は、ETH0 から ETH7、ETH8、および ETH9 です。

図 1. 8x2 イーサネット接続



親トピック: [アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

製品のインターオペラビリティ

WebSphere® DataPower® XC10 アプライアンスは、他の IBM® 製品との インターオペラビリティについてテスト済みです。

WebSphere Application Server

WebSphere Application Server は、HTTP セッションおよび 動的キャッシュ・データ・グリッド・シナリオを実装するために必須です。具体的に必要な WebSphere Application Server の リリースの詳細については、[システム要件](#)を参照してください。

WebSphere DataPower Integration アプライアンス XI50

WebSphere DataPower XC10 アプライアンスは、WebSphere DataPower Integration アプライアンス XI50 の サイド・キャッシュとして使用できます。REST ゲートウェイにより、非 Java ベースのクライアントが、HTTP ベースの一連の操作を使用して単純データ・グリッドにアクセスできるようになり、この統合が可能になります。REST ゲートウェイの詳細については、[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。WebSphere DataPower XC10 アプライアンスと WebSphere DataPower Integration アプライアンス XI50 間の統合を示すサンプル・アプリケーションの情報については、「[Using the WebSphere DataPower XC10 Appliance as a side cache for the WebSphere DataPower XI50 Integration Appliance \(WebSphere DataPower XI50 統合アプライアンスのサイド・キャッシュとしての、WebSphere DataPower XC10 の使用\)](#)」を参照してください。

WebSphere Portal

WebSphere Portal の HTTP セッションをアプライアンスのデータ・グリッドに保持できます。この構成の作成について詳しくは、[WebSphere Portal での HTTP セッション・マネージャーの構成](#)を参照してください。さらに、IBM WebSphere Portal の IBM Web Content Manager は、動的キャッシュ・インスタンスを使用して、拡張キャッシングが使用可能のときに Web Content Manager から取得されるレンダリング内容を保管することができます。WebSphere DataPower XC10 アプライアンス キャッシュ・アプライアンスは、デフォルトの動的キャッシュ実装を使用する代わりに、キャッシュ内容を柔軟性のあるデータ・グリッドに保管する、動的キャッシュの実装を提供します。

WebSphere Commerce

WebSphere Commerce バージョン 7.0.0.1 は、WebSphere eXtreme Scale クライアント バージョン 7.1 の使用をサポートするようになりました。WebSphere DataPower XC10 アプライアンス を使用して、WebSphere Commerce の動的キャッシュ・データをキャッシュに入れることができます。詳しくは、[動的キャッシュの作成データ・グリッド](#)を参照してください。

Tivoli monitoring

WebSphere DataPower XC10 アプライアンス の SNMP モニターを有効にすると、IBM Tivoli® Monitoring バージョン 6.2.2 フィックスパック 2 以降にそのデータをインポートすることができます。Tivoli Monitoring を使用して、複数のオペレーティング・システム上のシステムとネットワークのアプリケーションをモニターおよび管理し、企業内のあらゆる部分の可用性とパフォーマンスを追跡し、傾向を追跡するレポートを作成し、問題のトラブルシューティングをすることができます。詳しくは、[WebSphere DataPower XC10 アプライアンスから Tivoli Monitoring への SNMP データのインポート](#)を参照してください。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

関連情報:

☞ [WebSphere eXtreme Scale を使用した WebSphere Portal および IBM Web Content Manager パフォーマンスの向上](#)

Microsoft .NET に関する考慮事項

WebSphere® eXtreme Scale には 2 つの .NET 環境 (開発環境とランタイム環境) が存在します。これらの環境はそれぞれ固有の要件を持っています。

開発環境の要件

Microsoft .NET バージョン

.NET 3.5 以降のバージョンがサポートされます。

Microsoft Visual Studio

次のいずれかの Visual Studio バージョンを使用することができます。

- Visual Studio 2008 SP1
- Visual Studio 2010 SP1
- Visual Studio 2012

Windows

ご使用の Visual Studio のリリースでサポートされる任意の Windows バージョンがサポートされます。Visual Studio の Windows 要件については、次のリンクを参照してください。

- [Visual Studio 2008 システム要件](#)
- [Visual Studio 2010 Professional システム要件](#)
- [Visual Studio 2012 Professional システム要件](#)

メモリー

- 1 GB (32 ビットおよび 64 ビットのいずれのインストール済み環境にも適用されます)

ディスク・スペース

WebSphere eXtreme Scale は、Visual Studio 要件のほかに、さらに 50 MB の使用可能ディスク・スペースを必要とします。

ランタイム環境

Microsoft .NET バージョン

.NET 3.5 以降のバージョン (.NET 4.0 専用環境での実行を含む) がサポートされます。

Windows

上記の Microsoft .NET バージョン要件を満たす任意の Windows 環境。

メモリー

WebSphere eXtreme Scale サーバーに保管されているデータにアクセスするプロセスあたり 65 MB。

ディスク・スペース

WebSphere eXtreme Scale は 35 MB の使用可能ディスク・スペースを必要とします。トレースを使用可能に設定しているときは、最大 2.5 GB の追加のディスク・スペースが必要です。

WebSphere eXtreme Scaleruntime

.NET クライアント・アプリケーションを使用しているときには、eXtremeIO トランスポート・メカニズムを使用していなければなりません。eXtremeIO については、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

2.5+ ASP.NET セッション状態ストア・プロバイダーの要件

- ASP.NET セッション・ストア・プロバイダーでは、以下の IIS サーバー・バージョンのいずれかが必要です。
 - IIS 6.0 (Windows Server 2003 に同梱)
 - IIS 7.0 (Windows Server 2008 に同梱)
 - IIS 7.5 (Windows Server 2008 R2 に同梱)
 - IIS 8.0 (Windows Server 2012 に同梱)
- メモリー: プロセスごとに 120 MB の追加 (合計メモリー 185 MB)。
- セキュリティー: ASP.NET アプリケーション・プール ID が管理者特権を有する必要があります。
- セキュリティー: ASP.NET アプリケーションのトラスト・レベルを full に設定する必要があります。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

WebSphere DataPower XC10 アプライアンス のインストール

ご使用の既存の環境に WebSphere® DataPower® XC10 アプライアンス をインストールするには、最初にアプライアンス・ハードウェアをインストールする必要があります。次に、アプリケーション環境に WebSphere eXtreme Scale クライアントをインストールします。

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

アプライアンスの使用を開始する前に、アプライアンスのラックへのインストール、アクセスのためのアプライアンスの構成、ユーザー・インターフェースの開始、およびアプライアンスの 操作状態の確認を行う必要があります。

[IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成](#)

DataPower XC10 アプライアンス ハードウェアを取り付け、接続すると、アプライアンスを初期設定および 構成する準備は完了します。

[WebSphere eXtreme Scale クライアント のインストール](#)

IBM® WebSphere DataPower XC10 アプライアンス を WebSphere Application Server または、スタンドアロン環境のクライアントと連携させるには、WebSphere eXtreme Scale クライアント をインストールするか、クライアントの Java™ アーカイブ (JAR) ファイルをアプリケーションとともにパッケージ化する必要があります。

[Liberty プロファイルのインストール](#)

Installation Manager を使用するか、Java アーカイブ (JAR) ファイルを実行することで、Liberty プロファイル・アプリケーション・サービス環境をインストールします。

[製品インストールのトラブルシューティング](#)

IBM Installation Manager は多くの IBM ソフトウェア製品のための共通インストーラーです。このバージョンの WebSphere eXtreme Scale をインストールするには、このインストーラーを使用します。

クイック・スタート: アプライアンス・ハードウェアのインストール

アプライアンスの使用を開始する前に、アプライアンスのラックへのインストール、アクセスのためのアプライアンスの構成、ユーザー・インターフェースの開始、およびアプライアンスの操作状態の確認を行う必要があります。

始める前に

- MGMT0 の管理ポート用に 1 ギガビット・ネットワーク接続が必要です。
- データ・グリッドに 1 ギガビット・イーサネット・ポートを使用するか、10 ギガビット・イーサネット・ポートを使用するかを決定します。左のイーサネット・モジュール (1 ギガビット・ポート) または右のイーサネット・モジュール (10 ギガビット・ポート) のいずれかを使用する必要があります。
- 管理アクセス用およびデータ・グリッド用のイーサネット・インターフェースの IP アドレス。
- イーサネット・インターフェースのサブネットをサポートするデフォルト・ゲートウェイ (ルーター) の IP アドレス。
- ネットワーク・サービス (SSH、Telnet、など) の IP アドレス。

手順

1. アプライアンスを物理的にラックに取り付けます。

重要: アプライアンスのケースは開けないでください。ケースを開けると、セキュリティ障害が発生したとみなされ、アプライアンスは作動しなくなります。その場合、再設定のためにアプライアンスを IBM® に返却しなければなりません。

- a. アプライアンスを箱から丁寧に取り出します。付属の電源コード、シリアル・ケーブル、およびレールをすべて確認します。
- b. アプライアンスを取り付けるラックの位置を決め、アプライアンス保守のため、アプライアンスの上側および下側に、空気の流れを確保する十分なスペースがあることを確認します。
- c. マウント・レールを取り付けます。
- d. アプライアンスをレールに取り付け、アプライアンスを所定の位置までスライドします。
- e. アプライアンスの正面から、ネットワーク・ケーブルを使用してアプライアンスをネットワークに接続します。イーサネット・ケーブルは付属していません。
 - MGMT0 管理ポートを 1 ギガビット・ネットワークに接続します。
 - データ・グリッドの非管理トラフィックでは、左のイーサネット・モジュール (1 ギガビット・ポート) または右のイーサネット・モジュール (10 ギガビット・ポート) のいずれかを使用します。
- f. 付属の電源コードを使用して、電源機構を電源コンセントに接続します。
- g. シリアル・コンソールを、デバイス正面の CONSOLE コネクタに接続します。接続構成は、9600 ボー、8N1 (キヤラクター長 8 ビット、パリティなし、1 ストップ・ビット)、およびフロー制御なしです。提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して接続します。PL-2303 USB シリアル・ケーブルを使用する場合は、ケーブル用のドライバーをダウンロードしてインストールしてください。シリアル・コンソールに推奨される端末エミュレーションは、VT100 です。
- h. 電源ボタンを押します。緑色の電源 LED ライトが点灯し、シリアル・コンソールが表示されます。
- i. ログイン・プロンプトが表示されます。初期アプライアンス構成では、ユーザー ID とパスワード `xcadmin/xcadmin` を使用してログインします。

重要: `xcadmin` ユーザー ID とパスワードを失くさないよう注意してください。これらの情報を失くすと、アプライアンスに再度ログインすることができなくなり、再設定のためにアプライアンスを IBM に返送していただく必要があります。この場合、アプライアンスにあるデータはすべて消去されます。`xcadmin` ユーザー ID とパスワードに確実にアクセスできるようにするため、SMTP サーバーと E メール・アドレスを構成し、`xcadmin` パスワードを再設定できるようにすることをお勧めします。

2. シリアル・コンソールでのアクセス用にアプライアンスを構成します。ご使用条件に同意し、イーサネット・ポートを構成する一連のプロセスをウィザードが支援します。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。
3. ご使用のアプライアンス・ファームウェアを最新のバージョンに更新してください。詳しくは、[ファームウェアの更新](#)を参照してください。
4. ファームウェアのアップグレード後にアプライアンスの始動状況を追跡します。ユーザー・インターフェースでアプライアンスの始動状況を追跡することができます。Web ブラウザーのアドレス・バーから `https://<appliance_hostname>:9443/` にアクセスしてください。アプライアンスの初期化に使用している端末から `start-progress` コマンドを実行して進行状況をモニターすることもできます。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。
5. ユーザー・インターフェースを開始します。
 - a. Web ブラウザーのアドレス・バーに、デバイスの初期設定時に定義した URL およびポート情報を入力します。定義した IP アドレスまたはその IP アドレスに対応するホスト名 (例え

ば、<https://myXC10.ibm.com>) を使用できます。セキュア HTTP プロトコル (HTTPS) を使用してください。

- b. 「ユーザー」フィールドに `xcadmin` を入力します。
- c. 「パスワード」フィールドに対応するパスワードを入力します。このパスワードの初期設定は `xcadmin` です。
- d. 「ログイン」をクリックします。ログアウトするには、「ログアウト」をクリックします。

6. アプライアンスの動作状態を確認します。

- アプライアンス正面の緑色の電源 LED が点灯している。
- アプライアンス正面のオレンジ色の障害 LED が点灯していない。
- アプライアンス正面の緑色のキャッシング LED が点灯している。
- アプライアンス正面の琥珀色のキャッシング・エラー LED が点灯していない。
- アプライアンス正面の液晶ディスプレイ (LCD) に製品のリリースおよびビルド・バージョンが表示されている。

これらの各 LED のアプライアンス上での場所を示した図については、[アプライアンスの仕様およびフィーチャー](#) を参照してください。

ご不明な点がある場合は、IBM サポートにお問い合わせください。http://www-947.ibm.com/support/entry/portal/overview/software/websphere/websphere_datapower_xc10_appliance からアクセスできます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

関連タスク:

[ファームウェアの更新](#)

[IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成](#)

IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成

DataPower® XC10 アプライアンス ハードウェアを取り付け、接続すると、アプライアンスを初期設定および 構成する準備は完了します。

始める前に

初期構成では、シリアル接続を使用する必要があります。ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC を、アプライアンスのシリアル・ポートにシリアル接続する必要があります。シリアル・コンソールとして PC を使用している場合は、Windows 用または Linux 用の PC ベースのシリアル通信プログラムを使用する必要があります。VT100 スタイルのコンソールなどの専用ハードウェア・デバイス、またはハイパーターミナルや Minicom などのエミュレーターが稼働する PC が使用できます。提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して、アプライアンスに接続します。

重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。

基本構成を定義する前に、以下の情報を確認してください。

- MGMT0 の管理ポート用に 1 ギガビット・ネットワーク接続が必要です。
- データ・グリッドに 1 ギガビット・イーサネット・ポートを使用するか、10 ギガビット・イーサネット・ポートを使用するかを決定します。左のイーサネット・モジュール (1 ギガビット・ポート) または右のイーサネット・モジュール (10 ギガビット・ポート) のいずれかを使用する必要があります。
- アプライアンス管理アクセス用のイーサネット・インターフェース (MGMT) の IP アドレスおよびサブネット・マスク。
- イーサネット・インターフェースのサブネットをサポートするデフォルト・ゲートウェイ (ルーター) の IP アドレス。
- ドメイン・ネーム・システム (DNS) サーバーの IP アドレス。
- メール・サーバー情報 (E メール通知のセットアップ用)。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)を参照してください。
- 集合内のアプライアンス間の通信を可能にするための、Network Time Protocol (NTP) サーバー情報。

このタスクについて

初めて DataPower XC10 アプライアンス の初期設定を行うには、このタスクを使用します。アプライアンスに初めてサインオンする手順は、2 回目以降のサインオン手順と若干異なります。

注意:

1. [xcadminパスワード](#)の **xcadmin** ユーザー ID および パスワードの保存に関する重要な情報を参照してください。
2. アプライアンスのケースは開けないでください。ケースを開けると、セキュリティー障害が発生したとみなされ、アプライアンスは作動しなくなります。アプライアンスを復元するには、**device clear-intrusion** コマンドを実行する必要があります。
3. アプライアンスに付属されていたシリアル・ケーブルを紛失しないでください。アプライアンスの初期セットアップにこのケーブルは必要ですが、問題判別のために後から必要になることもあります。このケーブルは、このアプライアンスで機能することを目的として特別に設計されました。他のシリアル・ケーブルでは、このアプライアンスで機能しない場合があります。

初期の基本ファームウェア構成を実行してください。この構成は、WebSphere® DataPower XC10 アプライアンス をユーザー環境に追加する最小構成です。

手順

1. アプライアンスを初期設定します。以下の手順に従ってください。
 - a. アプライアンスに同梱されているシリアル・ケーブルまたは USB シリアル・ケーブルを使用して、シリアル・コンソールをアプライアンスに接続します。デバイスの前面にある CONSOLE コネクタにケーブルを接続し、ASCII 端末または PC で端末エミュレーション・ソフトウェアを実行させておく必要があります。アプライアンスが始動すると、この接続を使用して、アプライアンスからのメッセージを表示することができます。9600 ボー 8N1 (キャラクター長 8 ビット、パリティなし、1 ストップ・ビット)、およびフロー制御なしで、エミュレーション・ソフトウェアを構成します。シリアル・コンソールに推奨される端末エミュレーションは、VT100 です。
 - b. アプライアンスの電源がオンになっていることを確認してください。アプライアンスの電源がオンになっていない場合は、電源ボタンを切り替えてください。電源ボタンは、アプライアンスの正面にあります。アプライアンスが始動されるまで数秒間待ちます。アプライアンスの電源をオンにすると、以下の状態になります。
 - アプライアンス正面の緑色の電源 LED ライトが点灯し、ファンが起動します。
 - シリアル・コンソールにログイン・プロンプトが表示されます。初期アプライアンス構成では、ユーザー ID とデフォルト・パスワード **xcadmin/xcadmin** を使用してログインします。

- c. WebSphere DataPower XC10 アプライアンス 使用許諾契約書に同意します。先に進む前に、初回の場合はライセンスに同意する必要があります。各ライセンス・プロンプトで、「Accept」、「Reject」、または「StartOver」と入力します。
 - d. MGMT イーサネット・ポートを構成します。クラスレス・ドメイン間ルーティング (CIDR) 形式で IP アドレスを指定します。
 - e. MGMT イーサネット・ポートのデフォルト・ゲートウェイを構成します。
 - f. データ・グリッドのイーサネット・ポートを構成します。使用するポートがすべて 1 ギガビット・ポートであるか、またはすべて 10 ギガビット・ポートであるかを指定します。適用可能なポートの CIDR アドレスを構成します。
 - g. ドメイン・ネーム・システム (DNS) サーバーを構成します。DNS サーバーの有効な IP アドレスを指定します。
2. アプライアンス・ファームウェアを更新します。WebSphere DataPower XC10 アプライアンスのファームウェアをダウンロードまたは更新するには、アプライアンス管理許可が必要です。ファームウェア更新を取得するために、アプライアンスを外部インターネットに接続する必要はありません。ファームウェアのダウンロードまたは更新については詳しくは、『[ファームウェアの更新](#)』を参照してください。
 3. 新規アプライアンスにファームウェアをはじめてインストールする場合には、アプライアンスで **clear-all** コマンドを実行する必要があります。ファームウェア更新からの再始動が終了した後に、以下の手順を実行します。
 - a. xcadmin ユーザーを使用して、アプライアンスとの接続を確立します。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
 - b. **clear-all** コマンドを実行します。

```
Console> clear-all
Force Stopped all XC-10 processes
Deleting configuration data and logs
Deleting grid data
```

注意:

clear-all コマンドを実行する前に、他のコマンドを実行しないでください。他のコマンドを実行すると、アプライアンス構成で問題が生じる可能性があります。

注意:

clear-all コマンドの実行中に、他のコマンドを実行しないでください。 **start-progress** コマンドまたはアプライアンス始動状況ユーザー・インターフェース・ページを使用して進行状況をモニターするだけにしてください。

- c. **clear-all** コマンドの実行後に、アプライアンスの始動の進行状況を追跡することができます。以下のいずれかのオプションを使用します。
 - **2.5+** アプライアンスの始動状況をユーザー・インターフェースで追跡することができます。Web ブラウザーのアドレス・バーで、次の URL およびポート情報を入力してください。 https://<appliance_hostname>:9443/。詳しくは、[アプライアンスの始動状況のモニタ](#)を参照してください。
 - **start-progress** コマンドを実行して、進行をモニターします。このコマンドが STARTED を返す場合、アプライアンスは使用可能です。
4. 構成を保護するために、xcadmin ユーザーのパスワードを変更します。デフォルト・パスワードは xcadmin です。以下のコマンドでパスワードを変更できます。

```
user password <old_password> <new_password>
```

5. 構成を検証します。Web ブラウザーのユーザー・インターフェースを使用して、構成を検証します。

重要: 検証手順では、イーサネット・インターフェースのホスト名を myXC10.ibm.com と想定しています。

ブラウザーからユーザー・インターフェースにアクセスするには、以下の手順を使用します。

- a. Web ブラウザーを開きます。ネットワークに接続されたコンピューターで、Web ブラウザーを開きます。
- b. URL を入力します。アドレス・バーに、デバイスの初期設定時に定義した URL を入力します。例えば、<https://myXC10.ibm.com> と入力します。

注: http ではなく https プロトコルを使用してください。

- c. アプライアンスにログオンします。ローカルの XCADMIN のアカウントおよびパスワードを使用して、アプライアンスにログオンします。入力したパスワードは平文のため、表示されません。

d. 「ログイン」をクリックします。

「ようこそ」ページが表示されたら、ローカル XCADMIN アカウントの認証は成功です。

6. 構成を完了します。

タスクの結果

以上で、WebSphere DataPower XC10 アプライアンス の初期設定は完了です。

次のタスク

ユーザー・インターフェースでアプライアンス設定を構成します。これらの設定には、ユーザーとユーザー・グループ、セキュリティー、Eメール送達、ドメイン・ネーム・システム (DNS) サーバー、および日時の各設定が含まれます。詳しくは、[アプライアンスの構成](#)を参照してください。

[xcadmin/パスワード](#)

管理 (xcadmin) ユーザー ID とパスワードを設定した後は、その情報を信頼のおける場所に保管してください。Eメール送達を構成して、ユーザー・インターフェースでのパスワードの再設定機能を使用可能にすることができます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

関連タスク:

[ファームウェアの更新](#)

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

xcadminパスワード

管理 (xcadmin) ユーザー ID とパスワードを設定した後は、その情報を信頼のおける場所に保管してください。Eメール送達を構成して、ユーザー・インターフェースでのパスワードの再設定機能を使用可能にすることができます。

ユーザー ID およびパスワードの保存

xcadmin ユーザー ID とパスワードはアプライアンスにログオンするために必要です。そのため、ユーザー ID とパスワードを変更した後は、それらを安全な場所に保管してください。それらを忘れてしまい、復旧する手段を用意していない場合には、再設定のためにアプライアンスを IBM® に返却する必要があります。

Eメールでのパスワードの再設定

Eメール送達を構成すると、すべてのユーザーが、ユーザー・インターフェースログイン画面の「パスワードを忘れた場合」リンクをクリックすることで、パスワードを復元できます。新たに生成されたパスワードが含まれたEメールがユーザーに送達されます。

注意:

ユーザー・インターフェースのログイン画面の「パスワードを忘れた場合」リンクが、**xcadmin** パスワードを再設定するための唯一の方法です。Eメール送達を構成していない場合にパスワードを忘れた場合は、**device RESET** コマンドを使用してアプライアンスを再設定する必要があります。このコマンドでは、アプライアンスのすべての設定が再設定されます。**device RESET** コマンドについて詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。

xcadmin パスワードの変更

ユーザー・インターフェースまたはアプライアンスのコマンド行インターフェースで、xcadmin のパスワードを編集できます。

ユーザー・インターフェースでパスワードを変更する場合は、ユーザーを編集します。詳しくは、[ユーザーの管理](#)を参照してください。

アプライアンスのコマンド行インターフェースでパスワードを変更する場合は、**user password** コマンドを使用できます。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)
[ユーザー許可](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)
[ユーザーおよびグループの管理](#)
[データ・グリッドのセキュリティーを使用可能に設定する](#)
[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)
[IBM WebSphere DataPower XC10 アプライアンスのEメール送達の管理](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

WebSphere eXtreme Scale クライアント のインストール

IBM® WebSphere® DataPower® XC10 アプライアンス を WebSphere Application Server または、スタンドアロン環境のクライアントと連携させるには、 WebSphere eXtreme Scale クライアント をインストールするか、クライアントの Java™ アーカイブ (JAR) ファイルをアプリケーションとともにパッケージ化する必要があります。

このタスクについて

クライアント・アプリケーションが IBM WebSphere DataPower XC10 アプライアンスと通信するには、 WebSphere eXtreme Scale クライアントをインストールする必要があります。 クライアントをインストールする前に、スタンドアロン環境でクライアントをインストールするのか、あるいは、 WebSphere Application Server 環境にインストールするのかを認識している必要があります。

アプリケーションで WebSphere Application Server 動的キャッシュまたは HTTP セッションを使用する場合は、 WebSphere eXtreme Scale クライアントを WebSphere Application Server 環境にインストールする必要があります。

単純なデータ・グリッドは、組み込み WebSphere Application Server 環境でも、スタンドアロン環境でも使用できます。 クライアントをスタンドアロン環境にインストールすると、インストール済み環境は WebSphere Application Server を使用しません。

データ・グリッドのタイプの詳細については、 [アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#) を参照してください。

手順

1. IBM Installation Manager をダウンロードおよびインストールして、必要な製品リポジリーが確実にインストールされていることを確認してください。 [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#) を参照してください。
2. サポート・サイトから WebSphere eXtreme Scale クライアントをダウンロードします。 クライアントをダウンロードする場所の詳細については、 [サポート・ポータル](#) を参照してください。
3. Installation Manager を使用して、インストールを実行します。 正しい製品オフリングを選択してください。 製品オフリングは、Installation Manager 内のインストールの設定に追加したリポジリーに応じて使用可能になります。 WebSphere eXtreme Scale クライアント に使用可能な製品オフリングには、以下が含まれます。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 6
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

以下の中から、計画した構成に適切なインストールを使用します。

- **組み込みクライアント・インストール:** 動的キャッシュまたは HTTP セッションを使用する場合は、組み込みインストールを使用する必要があります。 また、単純なデータ・グリッドで組み込みインストールを使用することもできます。 詳しくは、 [WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- **スタンドアロン・クライアント・インストール:** スタンドアロン環境のクライアント・インストールは、単純なデータ・グリッドでのみ使用できます。 詳しくは、 [GUI の使用による WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- **サイレント・インストール:** スタンドアロン環境、あるいは、組み込み WebSphere Application Server 環境で、応答ファイルを使用したクライアントのインストールもできます。 詳しくは、 [応答ファイルの使用による WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。

次のタスク

アプライアンスを構成します。 詳しくは、 [アプライアンスの構成](#) を参照してください。

[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングは、製品リポジリーで使用可能です。 これらのリポジリーにアクセスするには、まず IBM Installation Manager をインストールする必要があります。

[IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

IBM Installation Manager を使用して WebSphere eXtreme Scale クライアント 製品オフリングをアンインストールします。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

WebSphere eXtreme Scale Client for .NET は、ランタイム環境でインストールすることもできれば、ランタイム環境と開発環境の両方でインストールすることもできます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール

WebSphere® eXtreme Scale クライアント 製品オフリングは、製品リポジトリで使用可能です。これらのリポジトリにアクセスするには、まず IBM® Installation Manager をインストールする必要があります。

Installation Manager は、製品メディア内のファイル、パスポート・アドバンテージのサイトから入手したファイル、もしくは [IBM Installation Manager ダウンロード Web サイト](#) から入手したファイルのいずれかを使用してインストールできます。ファイルはインストール・イメージが収められた圧縮ファイルです。

注:

Installation Manager は、32 ビット版または 64 ビット版でダウンロードすることができます。どちらの版の Installation Manager を使用しても、WebSphere eXtreme Scale をインストールできます。

Installation Manager を使用すると、必要な製品リポジトリにアクセスできます。WebSphere eXtreme Scale 製品オフリングをインストールするには、これらのリポジトリにアクセスする必要があります。

製品リポジトリにアクセスするには、2 つのオプションがあります。

オプション 1: 物理メディア上の製品リポジトリにアクセスして、ローカル・インストールを行う

1. システムに Installation Manager をインストールします。
2. Installation Manager を使用して、メディア上の製品リポジトリから製品オフリングをインストールします。

オプション 2: パスポート・アドバンテージから製品リポジトリをダウンロードして、ローカル・インストールを行う

1. パスポート・アドバンテージのサイトからリポジトリをダウンロードします。

注: IBM パスポート・アドバンテージ・オンライン Web サイトからダウンロード可能な IBM WebSphere eXtreme Scale のインストール・イメージのリストとその他の情報については、[サポートされるソフトウェア](#)を参照してください。

2. システムに Installation Manager をインストールします。
3. Installation Manager を使用して、ダウンロードした製品リポジトリから製品をインストールします。

[WebSphere eXtreme Scale クライアント 製品のオフリング ID](#)

製品の更新をインストールする場合、あるいはフィックスをロールバックする場合、コマンド行でオフリング ID を指定する必要があります。製品オフリングは、下の表を使用して特定してください。

[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)

WebSphere eXtreme Scale クライアントを、WebSphere Application Server または WebSphere Application Server Network Deployment がインストールされている環境にインストールできます。WebSphere Application Server または WebSphere Application Server Network Deployment の既存のフィーチャーを使用して、WebSphere DataPower XC10 アプライアンス アプリケーションを拡張できます。

[GUI の使用による IBM Installation Manager のインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM Installation Manager をインストールする必要があります。Installation Manager はウィザード・コンソールを使用してインストールできます。

[コマンド行の使用による IBM Installation Manager のインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM Installation Manager をインストールする必要があります。Installation Manager はコマンド行からインストールできます。

[応答ファイルの使用による IBM Installation Manager のインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM Installation Manager をインストールする必要があります。Installation Manager は応答ファイルを使用してインストールできます。

親トピック: [WebSphere eXtreme Scale クライアント のインストール](#)

WebSphere eXtreme Scale クライアント 製品のオフリング ID

製品の更新をインストールする場合、あるいはフィックスをロールバックする場合、コマンド行でオフリング ID を指定する必要があります。製品オフリングは、下の表を使用して特定してください。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール

WebSphere® eXtreme Scale クライアントを、WebSphere Application Server または WebSphere Application Server Network Deployment がインストールされている環境にインストールできます。WebSphere Application Server または WebSphere Application Server Network Deployment の既存のフィーチャーを使用して、WebSphere DataPower® XC10 アプライアンス アプリケーションを拡張できます。

始める前に

- ターゲット・インストール・ディレクトリーに WebSphere eXtreme Scale クライアントの既存のインストール済み環境が含まれていないことを確認します。
- WebSphere Application Server または WebSphere Application Server Network Deployment 環境で実行中のすべてのプロセスを停止します。**stopManager**、**stopNode**、および **stopServer** コマンドについて詳しくは、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。

注意:

すべての実行中のプロセスが停止していることを確認します。実行中のプロセスが停止していない場合でもインストールは続行しますが、予測不能な結果が生じます。一部のプラットフォームでは、インストールが不確定な状態のままになることがあります。

重要: WebSphere eXtreme Scale クライアント をインストールする際には、WebSphere Application Server をインストールしたのと同じディレクトリーにインストールする必要があります。例えば、WebSphere Application Server を C:¥[was_root](#) にインストールした場合は、C:¥[was_root](#) を、WebSphere eXtreme Scale クライアント のインストールのターゲット・ディレクトリーとしても選択する必要があります。

このタスクについて

eXtreme Scale を WebSphere Application Server または WebSphere Application Server Network Deployment に統合して、eXtreme Scale の機能をご使用の Java™ Platform, Enterprise Edition アプリケーションに適用します。Java EE アプリケーションは、クライアント接続を使用してそのデータ・グリッドにアクセスします。

手順

- WebSphere eXtreme Scale クライアント を WebSphere Application Server バージョン 8 環境内にインストールする場合、以下のステップを実行します。
 1. IBM Installation Manager をインストールします。詳しくは、[GUI の使用による IBM Installation Manager のインストール](#)を参照してください。
 2. Installation Manager を使用して、適切な eXtreme Scale 製品オフリングをインストールします。
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。
 3. パスポート・アドバンテージのサイトから必要な WebSphere Application Server バージョン 8 のリポジトリーをダウンロードします。詳しくは、「[How to download WebSphere Application Server - Express V8.5 from Passport Advantage](#)」を参照してください。
 4. WebSphere Application Server バージョン 8 をインストールします。詳しくは、[GUI を使用した、分散オペレーティング・システムでの製品のインストール](#)を参照してください。
- WebSphere eXtreme Scale クライアント を WebSphere Application Server バージョン 7 環境内にインストールする必要がある場合、以下のステップを実行します。
 1. IBM Installation Manager をインストールします。詳しくは、[GUI の使用による IBM Installation Manager のインストール](#)を参照してください。
 2. InstallShield MultiPlatform (ISMP) インストーラーを使用して、WebSphere Application Server バージョン 7 をインストールします。詳しくは、[アプリケーション・サービス提供環境のインストール](#)を参照してください。
 3. インストール後、インストールを完了するために WebSphere Application Server バージョン 7 を Installation Manager にインポートする必要があります。WebSphere Application Server バージョン 7 を Installation Manager にインポートすると、製品のフィックスパックの管理とインストールが 1 つのロケーションから行えます。フィックスパックおよび更新にアクセスできるよう、必要なリポジトリーが Installation Manager 内にセットアップされていることを確認してください。WebSphere Application Server 7 の既存のインストール済み環境を Installation Manager にインポートする方法については、[IBM Installation Manager レジストリーへの IBM WebSphere Application Server 製品情報のインポート \(Importing IBM WebSphere Application Server product information into the IBM Installation Manager registry\)](#) を参照してください。
 4. Installation Manager を使用して、適切な eXtreme Scale 製品オフリングをインストールします。

- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフラインのインストール](#)

GUI の使用による IBM Installation Manager のインストール

WebSphere® eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM® Installation Manager をインストールする必要があります。Installation Manager はウィザード・コンソールを使用してインストールできます。

始める前に

IBM Installation Manager をインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

手順

1. Installation Manager のインストール・ファイルが含まれているロケーションから、次のいずれかのコマンドを実行します。

管理インストール:

- **Windows** install.exe
- **UNIX Linux** ./install

非管理インストール:

- **Windows** userinst.exe
- **UNIX Linux** ./userinst

管理インストールおよび非管理インストールについて詳しくは、[管理者、非管理者、またはグループとしてインストール](#)を参照してください。

グループ・モードでのインストール:

- **UNIX Linux** ./groupinst

グループ・モードに関する注記:

- グループ・モードでは、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理することができます。
- グループ・モードでは、IBM Installation Manager の単一インスタンスを 2 人のユーザーが同時に使用することはできません。
- **Windows** グループ・モードは、Windows オペレーティング・システムでは使用できません。
- グループ・モードを使用せずに Installation Manager をインストールする場合、後からこの Installation Manager インスタンスを使用してインストールする製品を管理するためにグループ・モードを使用することはできません。
- 現行ユーザーのインストール・ロケーションをデフォルト・ロケーションから、グループ内のすべてのユーザーがアクセスできるロケーションに変更します。
- グループ・モードでインストールする前に、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップの説明のとおり、グループ、権限、および環境変数を設定してください。
- グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。

2. Installation Manager のパッケージが選択されていることを確認して、「次へ」をクリックします。
3. 使用条件の条項に同意し、「次へ」をクリックします。
4. 「次へ」をクリックします。
5. 要約情報を検討して、「インストール」をクリックします。正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。
6. 製品リポジトリを Installation Manager 設定に追加します。
 - a. Installation Manager を開始します。
 - b. トップ・メニューで、「ファイル」>「設定」をクリックします。
 - c. 「リポジトリ」を選択します。
 - d. 「リポジトリの追加」をクリックします。
 - e. リポジトリ・ファイルを含むロケーション内の repository.config ファイルのパスを入力します。以下に例を示します。

- **Windows** C:¥repositories¥product_name¥local-repositories

- **UNIX** | **Linux** /var/repositories/product_name/local-repositories

- f. 「**OK**」をクリックします。
7. 「リポジトリ」ウィンドウにリストされているロケーションのうち使用していないものをクリアします。
8. 「適用」をクリックします。
9. 「**OK**」をクリックします。
10. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

次のタスク

Installation Manager のインストールとリポジトリのセットアップが正常に完了したら、引き続き製品オフリング用の WebSphere eXtreme Scale クライアント、または for をインストールすることができます。詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

ウィザード・コンソールから Installation Manager を使用し、WebSphere eXtreme Scale クライアント 製品 オフリングをインストールします。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

GUI の使用による WebSphere eXtreme Scale クライアント のインストール

ウィザード・コンソールから Installation Manager を使用し、WebSphere® eXtreme Scale クライアント 製品オフアリングをインストールします。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフアリングのインストール](#)を参照してください。

手順

1. Installation Manager を開始します。

UNIX | **Linux** ヒント: `./IBMIM` コマンドを使用して、Installation Manager をグループ・モードで開始することができます。

- グループ・モードを使用すると、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理できるようになります。
- グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。

2. 「インストール」をクリックします。

注: 認証のプロンプトが出されたら、プログラムの Web サイトで登録した IBM ID とパスワードを使用してください。

Installation Manager は、定義済みリポジトリ内にある使用可能なパッケージを検索します。

3. 以下のいずれかの製品オフアリングと適切なバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

ご使用のシステムに既に当該製品がインストールされている場合は、その製品が既にインストールされていることを示すメッセージが表示されます。別のロケーションに製品のインストール済み環境を別途作成するには、「続行」をクリックします。

ヒント: 「インストール・マネージャー・リポジトリ設定」ページで「インストール中および更新中にサービス・リポジトリの検索」オプションが選択されていて、インターネットに接続している場合、「他のバージョンと拡張機能の確認」をクリックできます。これにより、選択したパッケージのデフォルト更新リポジトリで更新を検索することができます。この場合、「インストール・マネージャー・リポジトリ設定」ページに特定のサービス・リポジトリ URL を追加する必要はありません。

- a. インストールするフィックスを選択します。

デフォルトでは、任意の推奨フィックスが選択されます。

推奨フィックスがある場合、推奨フィックスのみを表示し、非推奨フィックスを非表示にするオプションを選択できます。

- b. 「次へ」をクリックします。

注: Installation Manager は、リポジトリに接続する際に、最新レベルの Installation Manager への更新を要求するプロンプトを出す場合があります。プロンプトが出されたら、新しいバージョンに更新してから続行します。自動更新については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

4. 使用条件の条項に同意し、「次へ」をクリックします。
5. 製品のインストール・ルート・ディレクトリを指定します。

パネルには共有リソース・ディレクトリおよびディスク・スペース情報も指定されます。

注: 初めて Installation Manager を使用してパッケージをインストールする場合、共有リソース・ディレクトリを指定します。共有リソース・ディレクトリは、1 つ以上のパッケージ・グループで使用できるインストール成果物が置かれるディレクトリです。このインストールには、最も大きいドライブを使用してください。すべてのパッケージをアンインストールし終わるまで、このディレクトリ・ロケーションは変更できません。

制約事項:

- デフォルト・ターゲット・ロケーションを削除し、インストール・ディレクトリー・フィールドを空のままにすると、続行できなくなります。
- 宛先ディレクトリーとしてシンボリック・リンクを使用しないでください。
シンボリック・リンクはサポートされていません。
- ディレクトリー名にセミコロンを使用しないでください。
ターゲット・ディレクトリーにセミコロンが含まれている場合、WebSphere eXtreme Scale は想定どおりにインストールされません。

Windows セミコロンは、Windows システムでは、クラスパスを構成するために使用する文字です。

- **Windows** Windows Server 2008、Windows Vista、および Windows 7 オペレーティング・システムでの最大パス長は 60 文字です。

6. 「次へ」をクリックします。
7. インストールするコンテンツの翻訳言語を選択します。

常に英語が選択されています。

8. 「次へ」をクリックします。
9. インストールするフィーチャーを選択します。

選択した製品オフリングに応じて、次のフィーチャーから選択できます。

- コンソール

すべての WebSphere eXtreme Scale 製品オフリングで使用可能です。 モニター・コンソールのインストールを選択することができます。 Web コンソールでは、現在と過去の統計をグラフにできます。このコンソールには、概要を表示するようにグラフがいくつか用意されているほか、使用可能な統計からグラフを作成できるカスタム・レポート・ページもあります。 WebSphere eXtreme Scale のモニター・コンソールのグラフ機能を使用して、環境内のデータ・グリッドの全体的なパフォーマンスを表示できます。

- サンプル

すべての WebSphere eXtreme Scale 製品オフリングで使用可能です。

10. 「次へ」をクリックします。
11. 要約情報を検討して、「インストール」をクリックします。

- 正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。

注: さらに、プログラムによって、重要なポストインストール指示も指定されることがあります。

- インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。

12. このインストールの完了後に開始するツールを選択します。
 - 実稼働環境に適した設定のアプリケーション・サーバー・プロファイルを新規に作成する場合は、「**プロファイルを作成するためのプロファイル管理ツール**」を選択します。
 - 開発環境に適した設定のアプリケーション・サーバー・プロファイルを作成する場合は、「**開発環境のアプリケーション・サーバー・プロファイルを作成するためのプロファイル管理ツール (Profile Management Tool to create an application server profile for a development environment)**」を選択します。

注: development 設定は、アプリケーションの更新が頻繁に行われ、システム・リソースが最小限である開発環境に適しています。実動サーバーには development 設定を使用しないでください。

- このインストールの完了後に、新しいプロファイルを作成しない場合は、「なし」を選択します。

制約事項: プロファイル管理ツールを起動するオプションは、プロファイル管理ツールを含むバージョンの WebSphere Application Server がインストールされている場合にのみ使用可能です。

13. 「終了」をクリックします。
14. 「ファイル」 > 「終了」をクリックして、Installation Manager を閉じます。

親トピック: [GUI の使用による IBM Installation Manager のインストール](#)

関連タスク:

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

コマンド行の使用による IBM Installation Manager のインストール

WebSphere® eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM® Installation Manager をインストールする必要があります。Installation Manager はコマンド行からインストールできます。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

手順

1. Installation Manager のインストール・ファイルが含まれているロケーションに移動して、以下のコマンドのいずれかを実行します。

管理インストール:

- **Windows** `installc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./installc -acceptLicense -log log_file_path_and_name`

非管理インストール:

- **Windows** `userinstc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./userinstc -acceptLicense -log log_file_path_and_name`

グループ・モードでのインストール:

```
UNIX Linux ./groupinstc -acceptLicense -dataLocation application_data_location
-log log_file_path_and_name -installationDirectory Installation_Manager_home
```

グループ・モードに関する注記:

- グループ・モードを使用すると、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理できるようになります。
 - **Windows** グループ・モードは、Windows オペレーティング・システムでは使用できません。
 - グループ・モードを使用せずに Installation Manager をインストールする場合、この Installation Manager を使用して今後インストールする製品はグループ・モードを使用して管理できません。
 - インストール・ロケーションは、必ず現在のユーザーのホーム・ディレクトリ内のデフォルトのロケーションから、グループ内のすべてのユーザーがアクセス可能なロケーションに変更してください。
 - グループ・モードでインストールする前に、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップの説明のとおり、グループ、権限、および環境変数を設定してください。
 - グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。
2. オプション: リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、`imutilsc` コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

次のタスク

Installation Manager のインストールとリポジトリのセットアップが正常に完了したら、引き続き WebSphere eXtreme Scale WebSphere eXtreme Scale クライアント、または WebSphere eXtreme Scale for WebSphere Application Server 製品オフリングをインストールできます。詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

[コマンド行の使用による WebSphere eXtreme Scale クライアント のインストール](#)

コマンド行から Installation Manager を使用し、WebSphere eXtreme Scale クライアント 製品オフリングをインストールします。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

コマンド行の使用による **WebSphere eXtreme Scale** クライアント のインストール

コマンド行から Installation Manager を使用し、WebSphere® eXtreme Scale クライアント 製品オフアリングをインストールします。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフアリングのインストール](#)を参照してください。

手順

1. システムにログオンします。
2. Installation Manager をインストールしたディレクトリーの eclipse/tools サブディレクトリーに移動します。
3. 製品リポジトリが使用可能であることを確認してください。

Windows

```
imcl.exe listAvailablePackages -repositories source_repository
```

UNIX Linux

```
./imcl listAvailablePackages -repositories source_repository
```

オフアリングの 1 つ以上のレベルが表示されるはずですが。

4. **imcl** コマンドを使用して、この製品をインストールします。

Windows

```
imcl.exe install com.ibm.websphere.v85_offering_version,optional_feature_ID  
-repositories source_repository  
-installationDirectory installation_directory  
-sharedResourcesDirectory shared_directory  
-accessRights access_mode  
-preferences preference_key=value  
-properties property_key=value  
-keyring keyring_file -password password  
-acceptLicense
```

UNIX Linux

```
./imcl install com.ibm.websphere.offering_version,optional_feature_ID  
-repositories source_repository  
-installationDirectory installation_directory  
-sharedResourcesDirectory shared_directory  
-accessRights access_mode  
-preferences preference_key=value  
-properties property_key=value  
-keyring keyring_file -password password  
-acceptLicense
```

ヒント:

- *offering_ID* は、[WebSphere eXtreme Scale クライアント 製品のオフアリング ID](#) にリストされるオフアリング ID です。
- *offering_version* は、アンダースコアを使用してオフアリング ID にオプションで付加することができます。これはインストール対象のオフアリングの特定バージョンです (例: 8.6.0.20110503_0200)。
 - *offering_version* が指定されていない場合、オフアリングの最新バージョンと、そのバージョンのすべてのインテリム・フィックスがインストールされます。
 - *offering_version* が指定されている場合、オフアリングの指定バージョンがインストールされ、そのバージョンのインテリム・フィックスはインストールされません。

オフアリングのバージョンは、リポジトリに対して次のコマンドを実行すると、オフアリング ID の最後にアンダースコアを使用して付加されている形で見つかります。

```
imcl listAvailablePackages -repositories source_repository
```

- また、`-installFixes` 引数に `none`、`recommended`、または `all` を指定することで、どのインテリム・フィックスをオフアリングと一緒にインストールするのかが示すこともできます。
 - オフアリングのバージョンが指定されていない場合、`-installFixes` オプションはデフォルトで `all` になります。
 - オフアリングのバージョンが指定されている場合、`-installFixes` オプションはデフォルトで `none` になります。
- コンマで区切られたフィーチャーのリストを追加できます。以下に例を示します。

```
imcl -acceptLicense install  
com.ibm.websphere.WXS.v85,xs.console.feature,xs.samples.feature
```

```
imcl -acceptLicense install  
com.ibm.websphere.WXS.v86,xs.console.feature,xs.samples.feature
```

- すべての製品オフアリングで使用可能です。xs.console.feature モニター・コンソールのインストールを選択することができます。Web コンソールでは、現在と過去の統計をグラフにできます。このコンソールには、概要を表示するように事前構成されたグラフがいくつか用意されているほか、使用可能な統計からグラフを作成できるカスタム・レポート・ページもあります。WebSphere eXtreme Scale のモニター・コンソールのグラフ機能を使用して、環境内のデータ・グリッドの全体的なパフォーマンスを表示できます。
- xs.samples.feature すべての製品オフアリングで使用可能です。サンプルのインストールを選択できます。

注:

- 以前に Installation Manager をインストールするモードを指定した場合、`-accessRights` パラメーターは必要ありません
- 後で問題が生じた場合のために、Installation Manager はパッケージの以前のバージョンをロールバックの対象として保存することができます。Installation Manager が以前のバージョンにパッケージをロールバックする際は、現行バージョンのファイルはアンインストールされて、以前のバージョンが再インストールされます。ロールバック用にファイルを保存しないことを選択する場合は、次の設定を指定することによりファイルが保存されないようにできます。

```
-preference  
com.ibm.cic.common.core.preferences.preserveDownloadedArtifacts=False
```

Installation Manager 設定の設定方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: ロールバック用にファイルを保存しないよう選択した場合でも、リポジトリからロールバック用の製品ファイルにアクセスすることは可能です。

- プログラムが重要なポストインストール指示を標準出力に書き込むことがあります。

imcl コマンドを使用して製品をインストールする方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [コマンド行の使用による IBM Installation Manager のインストール](#)

関連タスク:

[コマンド行の使用による WebSphere eXtreme Scale クライアントのアンインストール](#)

応答ファイルの使用による IBM Installation Manager のインストール

WebSphere® eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM® Installation Manager をインストールする必要があります。Installation Manager は応答ファイルを使用してインストールできます。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

手順

Installation Manager のインストール・ファイルが含まれているロケーションに移動して、以下のコマンドのいずれかを実行して Installation Manager をインストールします。

管理インストール:

- **Windows** `installc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./installc -acceptLicense -log log_file_path_and_name`

非管理インストール:

- **Windows** `userinstc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./userinstc -acceptLicense -log log_file_path_and_name`

グループ・モードでのインストール:

```
UNIX Linux ./groupinstc -acceptLicense -dataLocation application_data_location -log log_file_path_and_name -installationDirectory Installation_Manager_home
```

グループ・モードに関する注記:

- グループ・モードを使用すると、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理できるようになります。

グループ・モードは、IBM Installation Manager の単一インスタンスを 2 人のユーザーが同時に使用できるという意味ではありません。

- **Windows** グループ・モードは、Windows オペレーティング・システムでは使用できません。
- グループ・モードを使用せずに Installation Manager をインストールする場合、この Installation Manager を使用して今後インストールする製品はグループ・モードを使用して管理できません。
- インストール・ロケーションは、必ず現在のユーザーのホーム・ディレクトリ内のデフォルトのロケーションから、グループ内のすべてのユーザーがアクセス可能なロケーションに変更してください。
- グループ・モードでインストールする前に、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップの説明のとおり、グループ、権限、および環境変数を設定してください。
- グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。

次のタスク

Installation Manager のインストールとリポジトリのセットアップが正常に完了したら、引き続き製品オフリング用の WebSphere eXtreme Scale WebSphere eXtreme Scale クライアント、または WebSphere eXtreme Scale for WebSphere Application Server をインストールできます。詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

[応答ファイルの使用による WebSphere eXtreme Scale クライアント のインストール](#)

応答ファイルで Installation Manager を使用し、WebSphere eXtreme Scale クライアント 製品オフリングをインストールします。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

応答ファイルの使用によるWebSphere eXtreme Scale クライアント のインストール

応答ファイルで Installation Manager を使用し、WebSphere® eXtreme Scale クライアント 製品オフリングをインストールします。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

このタスクについて

Installation Manager を使用して、応答ファイル进行处理してさまざまな方法で製品のインストールを行うことができます。GUI を使用して応答ファイルを記録できます。

手順

1. コマンド行で、Installation Manager をインストールしたディレクトリー の Eclipse サブディレクトリーに移動します。
2. -record オプションを使用して、コマンド行から Installation Manager を開始します。

以下に例を示します。

- **Windows** 管理者または非管理者:

```
IBMIM.exe -skipInstall "C:¥temp¥imRegistry"  
-record C:¥temp¥install_response_file.xml
```

- **UNIX** | **Linux** 管理者:

```
./IBMIM -skipInstall /var/temp/imRegistry  
-record /var/temp/install_response_file.xml
```

- **UNIX** | **Linux** 非管理者:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry  
-record user_home/var/temp/install_response_file.xml
```

ヒント: 新しい応答ファイルを記録する際に、-skipInstall パラメーターを指定することができます。このパラメーターには、以下の利点があります。

- ファイルをインストールしないので、記録の速度が速くなります。
- -skipInstall パラメーターを指定して一時データ・ロケーションを使用すると、Installation Manager は、記録する際に、インストール・レジストリーを指定されたデータ・ロケーションに書き込みます。-skipInstall パラメーターを指定せずに Installation Manager を再度開始すると、応答ファイルを使用して実際のインストール・レジストリーにインストールすることができます。

-skipInstall の操作は、Installation Manager が使用する実際のエージェント・データ・ロケーションで使用することはできません。この操作はサポートされていません。クリーンで書き込み可能なロケーションを使用し、今後の記録セッションの際にはこのロケーションを再利用してください。

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

3. 適切なリポジトリを Installation Manager 設定に追加します。
 - a. トップ・メニューで、「ファイル」>「設定」をクリックします。
 - b. 「リポジトリ」を選択します。
 - c. 各リポジトリに対して以下のアクションを実行します。
 - i. 「リポジトリの追加」をクリックします。
 - ii.
 - iii. Web ベースのリモート・リポジトリ内、またはリポジトリ・ファイルを解凍したローカル・ディレクトリー内の、repository.config ファイルのパスを入力します。

例:

- リモート・リポジトリ:

https://downloads.mycorp.com:8080/WXS_85_repository

- ローカル・リポジトリ:
 - **Windows** C:\%repositories%\wxs85\local-repositories
 - **UNIX | Linux** /var/repositories/wxs85/local-repositories

- iv. 「**OK**」をクリックします。
- v. 「**適用**」をクリックします。
- vi. 「**OK**」をクリックします。

- d. 「**インストール**」をクリックします。

注: 認証のプロンプトが出されたら、プログラムの Web サイトで登録した IBM ID とパスワードを使用してください。

Installation Manager は、定義済みリポジトリ内にある使用可能なパッケージを検索します。

- 4. 以下のいずれかの製品オフラインと適切なバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

ご使用のシステムに既に当該製品がインストールされている場合は、その製品が既にインストールされていることを示すメッセージが表示されます。別のロケーションに製品のインストール済み環境を別途作成するには、「**続行**」をクリックします。

ヒント: 「インストール・マネージャー・リポジトリ設定」ページで「インストール中および更新中にサービス・リポジトリの検索」オプションが選択されていて、インターネットに接続している場合、「**他のバージョンと拡張機能の確認**」をクリックできます。これにより、選択したパッケージのデフォルト更新リポジトリで更新を検索することができます。この場合、「インストール・マネージャー・リポジトリ設定」ページに特定のサービス・リポジトリ URL を追加する必要はありません。

- 5. インストールするフィックスを選択します。

デフォルトでは、任意の推奨フィックスが選択されます。

推奨フィックスがある場合、推奨フィックスのみを表示し、非推奨フィックスを非表示にするオプションを選択できます。

- 6. 「**次へ**」をクリックします。
- 7. 使用条件の条項に同意し、「**次へ**」をクリックします。
- 8. 製品のインストール・ルート・ディレクトリを指定します。

パネルには共有リソース・ディレクトリおよびディスク・スペース情報も指定されます。

注: 初めて Installation Manager を使用してパッケージをインストールする場合、共有リソース・ディレクトリを指定します。共有リソース・ディレクトリは、1 つ以上のパッケージ・グループで使用できるインストール成果物が置かれるディレクトリです。このインストールには、最も大きいドライブを使用してください。すべてのパッケージをアンインストールし終わるまで、このディレクトリ・ロケーションは変更できません。

制約事項:

- デフォルト・ターゲット・ロケーションを削除し、インストール・ディレクトリ・フィールドを空のままにすると、続行できなくなります。
- 宛先ディレクトリとしてシンボリック・リンクを使用しないでください。

シンボリック・リンクはサポートされていません。

- **Windows** Windows Server 2008、Windows Vista、および Windows 7 オペレーティング・システムでの最大パス長は 60 文字です。

- 9. 「**次へ**」をクリックします。
- 10. インストールするコンテンツの翻訳言語を選択します。

常に英語が選択されています。

- 11. 「**次へ**」をクリックします。
- 12. インストールするフィーチャーを選択します。
 - コンソール

すべての WebSphere eXtreme Scale 製品オファリングで使用可能です。モニター・コンソールのインストールを選択することができます。Web コンソールでは、現在と過去の統計をグラフにできます。このコンソールには、概要を表示するようにグラフがいくつか用意されているほか、使用可能な統計からグラフを作成できるカスタム・レポート・ページもあります。WebSphere eXtreme Scale のモニター・コンソールのグラフ機能を使用して、環境内のデータ・グリッドの全体的なパフォーマンスを表示できます。

- サンプル

すべての WebSphere eXtreme Scale 製品オファリングで使用可能です。

13. 「次へ」をクリックします。

14. 要約情報を検討して、「インストール」をクリックします。

- 正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。

注: さらに、プログラムによって、重要なポストインストール指示も指定されることがあります。

- インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。

15. 「終了」をクリックします。

16. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

鍵リングの作成

WebSphere eXtreme Scale WebSphere eXtreme Scale クライアント 製品オファリングをインストールするための応答ファイルを、Installation Manager を使用して記録した後、鍵リング・ファイルの作成を選択することができます。認証を必要とするリモート・リポジトリを使用している場合、インストールのための鍵リングを作成できます。

親トピック: [応答ファイルの使用による IBM Installation Manager のインストール](#)

関連タスク:

[応答ファイルを使用した WebSphere eXtreme Scale クライアントのアンインストール](#)

鍵リングの作成

WebSphere® eXtreme Scale WebSphere eXtreme Scale クライアント 製品オフリングをインストールするための応答ファイルを、Installation Manager を使用して記録した後、鍵リング・ファイルの作成を選択することができます。認証を必要とするリモート・リポジトリを使用している場合、インストールのための鍵リングを作成できます。

始める前に

応答ファイルを記録する必要があります。詳しくは、[応答ファイルの使用によるWebSphere eXtreme Scale クライアントのインストール](#)を参照してください。

手順

1. コマンド行で、Installation Manager をインストールしたディレクトリー の Eclipse サブディレクトリーに移動します。
2. -record オプションを使用して、コマンド行から Installation Manager を開始します。

例:

- **Windows** 管理者または非管理者:

```
IBMIM.exe -skipInstall "C:¥temp¥imRegistry"  
-keyring C:¥IM¥im.keyring  
-record C:¥temp¥keyring_response_file.xml
```

- **UNIX** | **Linux** 管理者:

```
./IBMIM -skipInstall /var/temp/imRegistry  
-keyring /var/IM/im.keyring  
-record /var/temp/keyring_response_file.xml
```

- **UNIX** | **Linux** 非管理者:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry  
-keyring user_home/var/IM/im.keyring  
-record user_home/var/temp/keyring_response_file.xml
```

3. 認証済みリモート・リポジトリを使用するための資格情報を要求するウィンドウが開いたら、正しい資格情報を入力して、その情報を保存します。
4. 「ファイル」 > 「終了」をクリックして、Installation Manager を閉じます。

詳しくは、[IBM® Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [応答ファイルの使用によるWebSphere eXtreme Scale クライアントのインストール](#)

IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール

IBM® Installation Manager を使用して WebSphere® eXtreme Scale クライアント 製品オフリングをアンインストールします。

[GUI の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

IBM Installation Manager のウィザード・コンソールを使用して WebSphere eXtreme Scale クライアント をアンインストールできます。

[コマンド行の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

コマンド行から IBM Installation Manager を使用して WebSphere eXtreme Scale クライアント をアンインストールできます。

[応答ファイルを使用した WebSphere eXtreme Scale クライアントのアンインストール](#)

IBM Installation Manager で応答ファイルを使用して WebSphere eXtreme Scale クライアント をアンインストールできます。

親トピック: [WebSphere eXtreme Scale クライアント のインストール](#)

GUI の使用による WebSphere eXtreme Scale クライアント のアンインストール

IBM® Installation Manager のウィザード・コンソールを使用して WebSphere® eXtreme Scale クライアント をアンインストールできます。

始める前に

WebSphere eXtreme Scale をアンインストールする前に、すべての WebSphere Application Server プロファイルから WebSphere eXtreme Scale 拡張を削除する必要があります。WebSphere eXtreme Scale をアンインストールすると、拡張解除を実行できなくなります。manageprofiles コマンドを使用して、WebSphere eXtreme Scale 環境内にある既存のプロファイルを拡張解除します。

手順

1. 製品をアンインストールします。
 - a. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティ \(Command-line utilities\)](#) を参照してください。
 - b. Installation Manager を開始します。
 - c. 「アンインストール」をクリックします。
 - d. 「パッケージのアンインストール」ウィンドウで、以下のアクションを実行します。
 - i. 以下のいずれかと適切なバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 6
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
 - ii. 「次へ」をクリックします。
 - e. アンインストール・ウィザードで拡張された WebSphere Application Server プロファイルのリストが表示された場合は、アンインストールを続行するために、それらのプロファイルを拡張解除する必要があります。
 - f. サマリー情報を検討します。
 - g. 「アンインストール」をクリックします。
 - アンインストールが正常に行われた場合には、正常に終了したことを示すメッセージがプログラムによって表示されます。
 - アンインストールが正常に行われなかった場合には、「ログの表示」をクリックして、問題をトラブルシューティングします。
 - h. 「終了」をクリックします。
 - i. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。
2. オプション: IBM Installation Manager をアンインストールします。

重要: IBM Installation Manager をアンインストールする前に、Installation Manager によってインストールされたすべてのパッケージをアンインストールする必要があります。

この手順の実行については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

関連タスク:

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

コマンド行の使用による **WebSphere eXtreme Scale** クライアント のアンインストール

コマンド行から IBM® Installation Manager を使用して WebSphere® eXtreme Scale クライアント をアンインストールできます。

始める前に

WebSphere eXtreme Scale をアンインストールする前に、すべての WebSphere Application Server プロファイルから WebSphere eXtreme Scale 拡張を削除する必要があります。WebSphere eXtreme Scale をアンインストールすると、拡張解除を実行できなくなります。manageprofiles コマンドを使用して、WebSphere eXtreme Scale 環境内にある既存のプロファイルを拡張解除します。

手順

1. システムにログオンします。
2. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
3. Installation Manager をインストールしたディレクトリーの eclipse/tools サブディレクトリーに移動します。
4. **imcl** コマンドを使用して、この製品をアンインストールします。

Windows

```
imcl.exe uninstall com.ibm.websphere.v85,optional_feature_ID  
-installationDirectory installation_directory
```

UNIX Linux

```
./imcl uninstall com.ibm.websphere.v85,optional_feature_ID  
-installationDirectory installation_directory
```

ヒント:

- *offering_ID* は、[WebSphere eXtreme Scale クライアント 製品のオファリング ID](#) にリストされるオファリング ID です。
- コンマとフィーチャー ID で区切られたフィーチャーのリストを削除できます。以下に例を示します。

```
imcl uninstall com.ibm.websphere.WXS.v85,xs.console.feature,xs.samples.feature
```

- client スタンドアロン・クライアントのフィーチャーを示します
- server スタンドアロン・サーバーのフィーチャーを示します
- console Web ベースのモニター・コンソールを示します
- samples はサンプルを示します。

- フィーチャーのリストが指定されていない場合は、製品全体がアンインストールされます。

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

5. アンインストール・プロセスで拡張された WebSphere Application Server プロファイルのリストが表示された場合は、アンインストールを続行するために、それらのプロファイルを拡張解除する必要があります。
6. オプション: IBM Installation Manager をアンインストールします。

重要: IBM Installation Manager をアンインストールする前に、Installation Manager によってインストールされたすべてのパッケージをアンインストールする必要があります。

アンインストール・スクリプトを使用してこの手順を実行する方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

関連タスク:

[コマンド行の使用による WebSphere eXtreme Scale クライアント のインストール](#)

応答ファイルを使用した **WebSphere eXtreme Scale** クライアントのアンインストール

IBM® Installation Manager で応答ファイルを使用して WebSphere® eXtreme Scale クライアント をアンインストールできます。

始める前に

WebSphere eXtreme Scale をアンインストールする前に、すべての WebSphere Application Server プロファイルから WebSphere eXtreme Scale 拡張を削除する必要があります。WebSphere eXtreme Scale をアンインストールすると、拡張解除を実行できなくなります。manageprofiles コマンドを使用して、WebSphere eXtreme Scale 環境内にある既存のプロファイルを拡張解除します。

オプション: Installation Manager のインストールおよび製品のインストールを実行して (あるいは記録のみを行って)、いずれかのシステムの一時インストール・レジストリーに記録します。こうすることにより、Installation Manager がインストールされている標準レジストリーを使用せずに、この一時レジストリーを使用して、アンインストールを記録することができます。

このタスクについて

Installation Manager を使用して、応答ファイルを処理してさまざまな方法で製品のアンインストールを行うことができます。次の手順で説明するように GUI を使用して応答ファイルを記録することができます。あるいは、手動で、またはサンプルを入手して変更することによって、新しい応答ファイルを生成することもできます。

手順

- ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
- オプション: **製品をアンインストールする応答ファイルの記録:** いずれかのシステムで、以下のアクションを実行して、製品をアンインストールする応答ファイルを記録します。
 - コマンド行で、Installation Manager をインストールしたディレクトリーの Eclipse サブディレクトリーに移動します。
 - record オプションを使用して、コマンド行から Installation Manager を開始します。

例:

- **Windows** 管理者または非管理者:

```
IBMIM.exe -skipInstall "C:¥temp¥imRegistry"  
-record C:¥temp¥uninstall_response_file.xml
```

- **UNIX** | **Linux** 管理者:

```
./IBMIM -skipInstall /var/temp/imRegistry  
-record /var/temp/uninstall_response_file.xml
```

- **UNIX** | **Linux** 非管理者:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry  
-record user_home/var/temp/uninstall_response_file.xml
```

ヒント: 「始める前に」の説明に従って作成した一時インストール・レジストリーで -skipInstall パラメーターの使用を選択すると、Installation Manager は応答ファイルを記録する際に一時インストール・レジストリーを使用します。-skipInstall パラメーターが指定されていると、いずれの製品パッケージもインストールまたはアンインストールされないことにご注意ください。Installation Manager で行うすべてのアクションは、指定の一時レジストリーに保管されているインストール・データを更新するのみです。応答ファイルの生成後に、この応答ファイルを使用して、製品ファイルを削除して標準インストール・レジストリーを更新し、製品をアンインストールできます。

-skipInstall の操作は、Installation Manager が使用する実際のエージェント・データ・ロケーションで使用することはできません。これはサポートされていません。空の書き込み可能なロケーションを使用し、今後の記録セッションにはそのロケーションを再使用してください。

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

- c. 「アンインストール」をクリックします。
- d. 「パッケージのアンインストール」ウィンドウで、以下のアクションを実行します。
 - i. 以下のいずれかと適切なバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 6
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
 - ii. 「次へ」をクリックします。
 - iii. 「次へ」をクリックします。
- e. サマリー情報を検討します。
- f. 「アンインストール」をクリックします。
 - アンインストールが正常に行われた場合には、正常に終了したことを示すメッセージがプログラムによって表示されます。
 - アンインストールが正常に行われなかった場合には、「ログの表示」をクリックして、問題をトラブルシューティングします。
- g. 「終了」をクリックします。
- h. 「ファイル」 > 「終了」をクリックして、Installation Manager を閉じます。

3. 応答ファイルを使用した製品のアンインストール: 製品をアンインストールする各システムのコマンド行で、Installation Manager をインストールしたディレクトリーの eclipse/tools サブディレクトリーに移動し、作成した応答ファイルを使用して製品をアンインストールします。

例:

- **Windows** 管理者または非管理者:

```
imcl.exe
input C:\temp\uninstall_response_file.xml
-log C:\temp\uninstall_log.xml
```

- **UNIX Linux** 管理者:

```
./imcl
input /var/temp/uninstall_response_file.xml
-log /var/temp/uninstall_log.xml
```

- **UNIX Linux** 非管理者:

```
./imcl
input user_home/var/temp/uninstall_response_file.xml
-log user_home/var/temp/uninstall_log.xml
```

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

4. オプション: すべてのインストール済みパッケージをリストして、アンインストールを検証します。

UNIX Linux

```
./imcl listInstalledPackages
```

Windows

```
imcl listInstalledPackages
```

- 5. アンインストール・プロセスで拡張された WebSphere Application Server プロファイルのリストが表示された場合は、アンインストールを続行するために、それらのプロファイルを拡張解除する必要があります。
- 6. オプション: IBM Installation Manager をアンインストールします。

重要: IBM Installation Manager をアンインストールする前に、Installation Manager によってインストールされたすべてのパッケージをアンインストールする必要があります。

アンインストール・スクリプトを使用してこの手順を実行する方法については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

関連タスク:

[応答ファイルの使用による WebSphere eXtreme Scale クライアント のインストール](#)

WebSphere eXtreme Scale Client for .NET のインストールの概要

WebSphere® eXtreme Scale Client for .NET は、ランタイム環境でインストールすることもできれば、ランタイム環境と開発環境の両方でインストールすることもできます。

.NET アプリケーションをビルドしてテストするには、WebSphere eXtreme Scale Client for .NET を開発環境でインストールします。開発環境のインストールには、必ずランタイム環境のインストールが含まれます。ランタイム・アセンブリーは、ディスクおよびグローバル・アセンブリー・キャッシュ (GAC) にインストールされます。開発環境のインストールでは、さらに、サンプル・コード、Visual Studio IntelliSense 機能の統合 (吹き出しクラスとメソッドの説明用)、および API の資料がインストールされます。サンプルの WebSphere eXtreme Scale Client for .NET ソース・コードおよび Visual Studio プロジェクトは [net_client_home](#)\sample ディレクトリーにインストールされます。API 資料は [net_client_home](#)\doc ディレクトリーにインストールされます。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET のインストール](#)

このフレームワークで実行されるアプリケーションがある場合は、WebSphere eXtreme Scale クライアントを .NET 環境でインストールすることができます。

.NET 2.5+ [サイレント・モードでの WebSphere eXtreme Scale クライアント for .NET のインストール](#)

WebSphere eXtreme Scale クライアントを .NET 環境でサイレント・モードでインストールすることができます。この方法は、インストールを自動で行いたい場合や、製品を複数のマシンにインストールする必要がある場合に使用されます。サイレント・モードのインストールでは、まず最初に、応答ファイルを記録し、このファイルにパラメーターを渡す必要があります。

.NET 2.5+ [WebSphere eXtreme Scale クライアント for .NET のアンインストール](#)

WebSphere eXtreme Scale クライアント for .NET を環境から削除するには、Windows コントロールパネルからこれをアンインストールすることができます。また、サイレント・モードでアンインストールする場合は応答ファイルを記録することができます。複数の WebSphere eXtreme Scale クライアントをインストールしており、それらのインストール済み環境を速やかに削除する必要がある場合は、応答ファイルを記録しておくことが最適な方法です。

.NET 2.5+ [インストール・プログラムによらない WebSphere eXtreme Scale Client for .NET のインストール](#)

setup.exe ファイルを実行する許可がない場合は、既存のインストール済み環境から別の Windows システムにファイルをコピーすることができます。

親トピック: [WebSphere eXtreme Scale クライアントのインストール](#)

WebSphere eXtreme Scale Client for .NET のインストール

このフレームワークで実行されるアプリケーションがある場合は、WebSphere® eXtreme Scale クライアントを .NET 環境でインストールすることができます。

始める前に

- DVD から WebSphere eXtreme Scale Client for .NET を取得します。ルート・ディレクトリー内で setup.exe ファイル、すなわち、 /ClientForDotNet/setup.exe を見付けるか、または、 [サポート・サイト](#) からダウンロードすることができます。
- WebSphere eXtreme Scale Client for .NET を開発環境にインストールする場合は、ハードウェア要件およびソフトウェア要件を満たす Windows システムを使用している必要があります。詳しくは、 [Microsoft .NET に関する考慮事項](#) を参照してください。

このタスクについて

WebSphere eXtreme Scale Client for .NET は、ランタイム環境でインストールするか、またはランタイム環境と開発環境の両方でインストールすることができます。

手順

1. ウィザードを使用して、DVD からクライアントをインストールします。ルート・ディレクトリー内で setup.exe、すなわち、 /ClientForDotNet/setup.exe を見付けるか、または、 [サポート・サイト](#) からダウンロードすることができます。
2. setup.exe ファイルを実行します。
3. ウィザードのプロンプトに従って進み、「次へ」をクリックします。「セットアップ・タイプ」ページが表示されます。
4. WebSphere eXtreme Scale Client for .NET をランタイム環境にインストールすることに決めていた場合は、「ランタイム」をクリックします。次のステップを実行します。

「インストール」をクリックしてインストーラーを実行し、「終了」をクリックします。デフォルトのインストール・ディレクトリーは C:\Program Files (x86)\IBM\WebSphere\eXtreme Scale .NET Client です。
5. WebSphere eXtreme Scale クライアント をランタイム環境と開発環境の両方にインストールすることに決めていた場合は、「カスタム」を選択し、以下のステップを実行します。
 - a. WebSphere eXtreme Scale クライアント をデフォルトのインストール・ディレクトリーにインストールするか、またはインストール・ディレクトリーを選択し、「次へ」をクリックします。
 - b. デフォルトでは、ランタイム環境と開発環境の両方がフィーチャーとして選択されます。両方をインストールする場合は、十分なディスク・スペースがあることを確認してください。「次へ」をクリックします。
 - c. ログ・ファイルの場所を選択し、「次へ」をクリックします。
 - d. 「インストール」をクリックしてインストーラーを実行し、「終了」をクリックします。

次のタスク

- 入門アプリケーションを実行して、ご使用の WebSphere eXtreme Scale Client for .NET をテストします。詳しくは、 [チュートリアル: 単純データ・グリッド・アプリケーションの入門](#) を参照してください。
- WebSphere eXtreme Scale Client for .NET を構成します。詳しくは、 [WebSphere eXtreme Scale Client for .NET の構成](#) を参照してください。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

[2.5+ WebSphere eXtreme Scale クライアント for .NET のアンインストール](#)

[.NET 2.5+ インストール・プログラムによらない WebSphere eXtreme Scale Client for .NET のインストール](#)

サイレント・モードでの WebSphere eXtreme Scale クライアント for .NET のインストール

WebSphere® eXtreme Scale クライアント を .NET 環境でサイレント・モードでインストールすることができます。この方法は、インストールを自動で行いたい場合や、製品を複数のマシンにインストールする必要がある場合に使用されます。サイレント・モードのインストールでは、まず最初に、応答ファイルを記録し、このファイルにパラメーターを渡す必要があります。

始める前に

- DVD から WebSphere eXtreme Scale クライアント を取得します。ルート・ディレクトリー内で `setup.exe`、すなわち、`/net_client_home/setup.exe` を見付けるか、または、[サポート・サイト](#) からダウンロードすることができます。
- WebSphere eXtreme Scale クライアント for .NET を開発環境にインストールする場合は、Microsoft .NET の考慮事項に記載されているハードウェア要件およびソフトウェア要件を満たす Windows システムを使用している必要があります。詳しくは、[Microsoft .NET に関する考慮事項](#) を参照してください。

このタスクについて

WebSphere eXtreme Scale クライアント for .NET は、ランタイム環境でインストールするか、またはランタイム環境と開発環境の両方でインストールすることができます。

手順

1. コマンド・プロンプトを開き、次のスクリプトを実行します: `setup.exe /r /f1"<Response_Files_Directory>%Setup.iss"` <Response_Files_Directory> は応答ファイルを作成する場所です。
2. ウィザードのプロンプトに従って進み、「次へ」をクリックします。「セットアップ・タイプ」ページが表示されます。
3. 選択するオプションに応じて、Setup.iss 応答ファイルを作成するために次の値を渡すことができます。
 - WebSphere eXtreme Scale クライアント をランタイム環境でインストールすることを選択するか、またはカスタム・インストールを選択します。カスタム・インストールでは、製品を両方の環境でインストールすることができます。
 - WebSphere eXtreme Scale クライアント をランタイム環境にインストールすることに決めていた場合は、「ランタイム」をクリックし、以下のことを行います。
 - a. 「インストール」をクリックして、「終了」をクリックします。デフォルトのインストール・ディレクトリーは `C:%Program Files (x86)%IBM%WebSphere%eXtreme Scale .NET Client` です。
 - WebSphere eXtreme Scale クライアント をランタイム環境と開発環境の両方にインストールすることに決めていた場合は、「カスタム」を選択し、以下のことを行います。
 - a. WebSphere eXtreme Scale クライアント をデフォルトのインストール・ディレクトリーにインストールするか、またはインストール・ディレクトリーを選択します。「次へ」をクリックします。
 - b. ランタイム環境と開発環境の両方を選択します。両環境でインストールしたい場合は、十分なディスク・スペースがあることを確認してください。「次へ」をクリックします。
 - c. ログ・ファイルの場所を選択し、「次へ」をクリックします。
4. 「インストール」をクリックして、「終了」をクリックします。
5. コマンド・プロンプトを開き、スクリプト `setup.exe /s /f1"<Response_Files_Directory>%Setup.iss"` を実行して、サイレント・モードで WebSphere eXtreme Scale クライアント をインストールします。ここで、<Response_Files_Directory> は、応答ファイルが置かれる場所です。

次のタスク

有効な eXtreme Scale Client for .NET API を試みるように SimpleClient を更新または変更することができます。インストール・ディレクトリーにある `<installation_directory>%sample%SimpleClient` で、SimpleClient を見付けて、このファイルを Visual Studio にロードすると、単純な作成、検索、更新、および削除操作を使用するサンプル・アプリケーションを表示することができます。SimpleClient をデータ・グリッドへのアクセス・ガイドとして使用します。このアプリケーションを変更することもできれば、eXtreme Scale for .NET クライアントがサポートする API セットを使用する新しいアプリケーションを作成することもできます。詳しくは、[チュートリアル: 単純データ・グリッド・アプリケーションの入門](#) を参照してください。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

[2.5+ WebSphere eXtreme Scale クライアント for .NET のアンインストール](#)

WebSphere eXtreme Scale クライアント for .NET のアンインストール

WebSphere® eXtreme Scale クライアント for .NET を環境から削除するには、Windows コントロール パネルからこれをアンインストールすることができます。また、サイレント・モードでアンインストールする場合は応答ファイルを記録することができます。複数の WebSphere eXtreme Scale クライアント をインストールしており、それらのインストール済み環境を速やかに削除する必要がある場合は、応答ファイルを記録しておくことが最適な方法です。

始める前に

開発環境から製品をアンインストールする場合は、必ず Visual Studio をシャットダウンしてください。

重要: アンインストーラーは、すべてのバイナリー・ファイルと、フィックスパックやインテリム・フィックスなどのすべての保守を同時に削除します。

手順

1. すべての .NET eXtreme Scale プロセスを停止します。
2. 以下のいずれかの方法を使用して、WebSphere eXtreme Scale クライアント for .NET をアンインストールすることができます。
 - Windows の「コントロール パネル」から「プログラムの追加と削除」をクリックして、「IBM WebSphere eXtreme Scale クライアント for .NET」を選択してアンインストールします。
 - 応答ファイルを記録する必要がある場合は、コマンド・プロンプトを開き、次のスクリプトを実行します。

```
setup.exe /uninst /r /f1"<Response_Files_Directory>%Setup.iss"
```

- a. アンインストール・ウィザードが開き、確認ウィンドウが表示されます。WebSphere eXtreme Scale クライアント for .NET とそのすべてのフィーチャーを削除することを確認してください。「**OK**」をクリックします。
 - b. アンインストール・プロセスが完了したら、「終了」をクリックします。
3. オプション: 応答ファイルを使用して WebSphere eXtreme Scale クライアント の既存のインストール済み環境をアンインストールする場合は、次のようにして、記録しておいた応答ファイルを実行します。
 - a. コマンド・プロンプトを開き、次のスクリプトを実行して WebSphere eXtreme Scale クライアント for .NET をサイレント・モードでアンインストールします。

```
setup.exe /uninst /s /f1"<Response_Files_Directory>%Setup.iss"
```

次のタスク

Windows Explorer で、すべてのフォルダーがインストール・ディレクトリーから削除されたことを確認します。また、Windows コントロール パネルでも、製品がリストされていないことを確認してください。インストーラーは、インストール後に生成されたファイル (例えば、ログ・ファイル、カスタム構成ファイル、SimpleClient サンプルのビルドで作成された任意の成果物) を含むフォルダーを削除しません。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

[.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストール](#)

[.NET 2.5+ サイレント・モードでの WebSphere eXtreme Scale クライアント for .NET のインストール](#)

インストール・プログラムによらない **WebSphere eXtreme Scale Client for .NET** のインストール

2.5+ setup.exe ファイルを実行する許可がない場合は、既存のインストール済み環境から別の Windows システムにファイルをコピーすることができます。

始める前に

ファイルのコピー元となる WebSphere® eXtreme Scale Client for .NET の完全なインストール済み環境がなければなりません。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストールの概要](#)を参照してください。

このタスクについて

ご使用のクライアント・サーバー上で setup.exe ファイルを実行する許可がない場合は、既存の WebSphere eXtreme Scale Client for .NET からご使用の .NET アプリケーションのランタイム・ディレクトリーにファイルをコピーすることができます。

警告: 自動アップグレードおよび自動マイグレーションは、WebSphere eXtreme Scale Client for .NET インストール・プログラムを使用した場合にのみ使用可能です。以下の手順を使用して製品をインストールする場合は、手動アップグレードおよび手動マイグレーションが必要です。

手順

- Windows** ご使用の WebSphere eXtreme Scale Client for .NET インストール済み環境から、以下のファイルをターゲットの Windows システムにコピーします。ターゲットの Windows システム・ディレクトリーは、ご使用のデータ・グリッドの .NET アプリケーションを実行するプロセスのランタイム・ディレクトリーでなければなりません。このプロセスは、ターゲット・ディレクトリーに対する読み取りおよび書き込み権限を持っていないければなりません。ログ・ファイルは、このディレクトリーの下にある logs フォルダーに生成されます。
 - [net_client_home](#)\bin\IBM.WebSphere.Caching.dll
 - [net_client_home](#)\bin\IBM.WebSphere.Caching.CredentialGenerator.dll (提供された資格情報生成プログラムを使用するように WebSphere eXtreme Scale Client for .NET アプリケーションが構成された場合にのみ必要です。)
 - [net_client_home](#)\config\Client.Net.properties
 - [net_client_home](#)\config\Client.Net.Log.config
- オプション: 手動で IBM.WebSphere.Caching.dll アセンブリーをグローバル・アセンブリー・キャッシュ (GAC) にインストールします。GAC をインストールする正確な手順は、Windows 環境によって異なります。

次のタスク

WebSphere eXtreme Scale Client for .NET を構成します。詳しくは、[WebSphere eXtreme Scale Client for .NET の構成](#)を参照してください。

親トピック: **.NET** **2.5+** [WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

Liberty プロファイル のインストール

Installation Manager を使用するか、Java™ アーカイブ (JAR) ファイルを実行することで、Liberty プロファイル・アプリケーション・サービス環境をインストールします。

このタスクについて

Liberty プロファイルをインストールするには、WebSphere® eXtreme Scale クライアント for WebSphere Application Server バージョン 8.5 以上をインストールすることができます。または、提供された JAR ファイルを実行することによって Liberty プロファイルをインストールすることができます。

z/OS® オペレーティング・システムでは、Liberty プロファイル は運用環境を提供します。この環境に関するネイティブでの作業は、MVS™ コンソールを使用して行うことができます。アプリケーション開発の場合は、別の分散システム、Mac OS、または z/OS 上の Linux シェルで Eclipse ベースの開発者ツールを使用することを検討してください。

手順

- 以下のいずれかのオプションを使用して、WebSphere Application Server Liberty プロファイル をインストールします。
 - WebSphere DataPower® XC10 アプライアンス では、WebSphere eXtreme Scale クライアント をインストールする必要があります。このトポロジーで Liberty プロファイル を実行するためには、[Java アーカイブ \(JAR\) ファイルを実行することによって Liberty プロファイル・アプリケーション・サービス環境をインストール](#)する必要があります。
- WebSphere eXtreme Scale クライアント バージョン 8.5 以上をインストールします。このオプションを使用して製品をインストールすると、自動的に Liberty プロファイルがインストールされます。
- ご使用の Liberty プロファイル環境をマイグレーションします。Liberty プロファイルをあるメジャー・リリースからより高いメジャー・リリースにマイグレーションするときは、Liberty プロファイルの server.xml ファイルでフィーチャー・バージョン番号を変更する必要があります。

例えば、製品がサポートしていた Liberty プロファイル の初期バージョンでは、フィーチャー・バージョン番号は 1.0 レベルでした。WebSphere eXtreme Scale クライアント バージョン 8.6 以上では、フィーチャーバージョン番号は 1.1 レベルです。

注: WebSphere Application Server V8.5.5 にアップグレードすると、Liberty プロファイルは削除されます。WebSphere eXtreme Scale はこのフィーチャーの削除をサポートします。ただし、WebSphere Application Server V8.5 にロールバックすると、再び Liberty プロファイルが追加されます。これは、WebSphere eXtreme Scale で問題を引き起こします。

[データ・キャッシングと Liberty プロファイル](#)

データ・キャッシング製品では、WebSphere Application Server Liberty プロファイルを使用して、HTTP セッション (REST ゲートウェイを介したクライアント/サーバー接続) を開発したり、その他のキャッシュ統合シナリオを管理したりすることができます。

[JAR ファイルの実行による Liberty プロファイルのアプリケーションのサービス環境のインストール](#)

配布イメージが入った Java アーカイブ (JAR) ファイルを実行することで、アプリケーション・サービス環境をインストールし、Liberty サーバーを作成する準備ができます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

関連タスク:

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

[コマンド行の使用による WebSphere eXtreme Scale クライアント のインストール](#)

[応答ファイルの使用による WebSphere eXtreme Scale クライアント のインストール](#)

データ・キャッシングと Liberty プロファイル

Java データ・キャッシング製品では、WebSphere® Application Server Liberty プロファイルを使用して、HTTP セッション (REST ゲートウェイを介したクライアント/サーバー接続) を開発したり、その他のキャッシュ統合シナリオを管理したりすることができます。

WebSphere DataPower® XC10 アプライアンスでは、Liberty プロファイルを使用してアプライアンス内のデータ・グリッドに接続することができます。例えば、WebSphere eXtreme Scale クライアントを Liberty プロファイルと共にインストールすると、HTTP セッション・アプリケーション、Java クライアント・アプリケーション、および REST クライアント・アプリケーション (これらのアプリケーションは Liberty プロファイルにインストールされます) を管理するために使用できるフィーチャーへのアクセスが可能になります。

以下のフィーチャーには、主要な使用可能フィーチャーに関する情報が記載されています。構成にフィーチャーを組み込むと、さらに 1 つ以上のフィーチャーが自動的にロードされることがあります。各フィーチャーには、要旨とそのフィーチャーの宣言例が含まれています。


クライアント・フィーチャー

クライアント・フィーチャーには、eXtreme Scale のプログラミング・モデルの大部分が含まれています。Liberty プロファイルで実行され、eXtreme Scale API を使用するアプリケーションがあるときは、クライアント・フィーチャーを追加してください。

`wlp_install_root/usr/server/wlp_install_root/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale Client">
  <featureManager>
    <feature>eXtremeScale.client-1.1</feature>
  </featureManager>
</server>
```

Web フィーチャー

 Web フィーチャーは推奨されません。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを使用してください。

Web フィーチャーには、Liberty プロファイル Web アプリケーションを拡張する機能が含まれています。フォールト・トレランスのために HTTP セッション・データを複製するときは、Web フィーチャーを追加してください。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">
  <featureManager>
    <feature>eXtremeScale.web-1.1</feature>
  </featureManager>
  <xsWebAppV85/>
</server>
```

WebApp フィーチャー

webApp フィーチャーには、Liberty プロファイル Web アプリケーションを拡張する機能が含まれています。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを追加してください。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">
  <featureManager>
    <feature>eXtremeScale.webApp-1.1</feature>
  </featureManager>
  <xsWebApp/>
</server>
```

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

<featureManager>
<feature>eXtremeScale.webGrid-1.1</feature>
</featureManager>

<xsWebGrid/>
</server>
```

REST フィーチャー

Representational State Transfer (REST) ゲートウェイを使用して、Liberty プロファイル内の集合がホストする単純データ・グリッドにアクセスします。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

<featureManager>
<feature>eXtremeScale.rest-1.1</feature>
</featureManager>

<xsRest/>
</server>
```

動的キャッシュ・フィーチャー

Liberty プロファイル・サーバーは、動的キャッシュが使用可能になっているアプリケーションのデータをキャッシュするデータ・グリッドをホストすることができます。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

  <featureManager>
    <feature>eXtremeScale.dynacacheGrid-1.1</feature>
  </featureManager>

  <xsDynacacheGrid/>
</server>
```

2.5+

動的キャッシュ・アプリケーション・フィーチャー

Liberty プロファイル・サーバーは、このフィーチャーを使用して動的キャッシュ・プロバイダーとして構成できる WebSphere eXtreme Scale をホストすることができます。

Liberty プロファイルにおけるデフォルト・キャッシュ・プロバイダーの使用については次の例を参照してください。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

<featureManager>
<feature>eXtremeScale.dynacacheapp-1.1</feature>
</featureManager>

<xsClientDomain default="production">
<endpointConfig> production;localhost:2809 </endpointConfig>
</xsClientDomain>

<distributedMap id="baseCache" libraryRef="idgenerator" cacheProviderName="WebSphere
eXtreme Scale">
<xsDynacacheApp remoteDomain="production" />
</distributedMap>

<distributedMap id="cache01" jndiName="cache01" memorySizeInEntries="2000"
createCacheAtServerStartup="true" cacheProviderName="WebSphere eXtreme Scale">
```

```
<xsDynacacheApp remoteDomain="production" />  
</distributedMap>  
</server>
```

親トピック: [Liberty プロファイルのインストール](#)

JAR ファイルの実行による Liberty プロファイルのアプリケーションのサービス環境のインストール

配布イメージが入った Java アーカイブ (JAR) ファイルを実行することで、アプリケーション・サービス環境をインストールし、Liberty サーバーを作成する準備ができます。

このタスクについて

このトピックで説明しているように JAR ファイルを実行するか、Installation Manager を使用することで、Liberty プロファイル・アプリケーション・サービス環境をインストールできます。

WebSphere® Application Server 用の JAR ファイルを実行して Liberty プロファイルをインストールする場合は、まずその JAR ファイルを解凍する必要があります。次に、WebSphere eXtreme Scale 用の Liberty プロファイル JAR ファイルを解凍します。IBM Installation Manager を使用して WebSphere Application Server バージョン 8.5 をインストールし、Liberty プロファイルを取得する場合は、Installation Manager を使用して WebSphere eXtreme Scale をインストールする必要もあります。

このタスクは以下のエディションをサポートします。

- WebSphere Application Server Liberty Core
- WebSphere Application Server, Base および Developer エディション
- WebSphere Application Server, Network Deployment
- WebSphere Application Server for z/OS

Liberty プロファイル・アプリケーション・サービス環境およびデータ・キャッシング環境のダウンロードについては、[WASdev コミュニティーのダウンロード・ページ](#)を参照してください。

手順

1. WebSphere Application Server Liberty プロファイル 配布イメージを任意のディレクトリーに解凍します。

このイメージは JAR ファイルとしてパッケージされています。例: `wlp-edition-8.6.0.0.jar`。次のいずれかのアクションを使用してこの JAR ファイルを解凍します。

- ウィザードを使用して配布イメージを解凍するには、`java -jar wlp-edition-8.6.0.0.jar` を実行します。
- ライセンス条件をサイレントに受け入れて配布イメージを解凍するには、`java -jar wlp-edition-8.6.0.0.jar -acceptLicense` を実行します。
- 使用可能なすべてのオプションを表示するには、`java -jar wlp-edition-8.6.0.0.jar -help` を実行します。

アプリケーション・サーバー・ファイルはすべて、`wlp` ディレクトリーのサブディレクトリーに格納されています。

2. オプション: ご使用の環境に `JAVA_HOME` プロパティを設定します。

Liberty プロファイルを使用するには、その実行場所となる JRE が必要です。WebSphere Application Server の完全プロファイルが使用する JDK または JRE を共有しません。『[Liberty プロファイル環境のカスタマイズ](#)』で説明されているように、`server.env` ファイル内の `JAVA_HOME` プロパティを使用して JDK または JRE の場所を指定できます。Linux システムまたは UNIX システムでは、代わりに、ユーザーの `.bashrc` ファイル内の `JAVA_HOME` を設定するか、`PATH` 環境変数に JDK または JRE のパスを追加することができます。Windows システムでは、代わりに、`JAVA_HOME` をシステム環境変数として設定するか、`PATH` システム変数に JDK または JRE のパスを追加することができます。

Windows 例えば、Windows システムでは、以下のコマンドを使用して、`JAVA_HOME` プロパティを設定し、パスに `Java™ /bin` ディレクトリーを追加することができます。

```
set JAVA_HOME=C:\Program\Java\jdk-16
set PATH=%JAVA_HOME%\bin;%PATH%
```

注:

- Liberty プロファイルのランタイム環境は、`JAVA_HOME` プロパティ、`JRE_HOME` プロパティ、システム `PATH` プロパティの順に `java` コマンドを検索します。
- サポートされる Java 環境およびその取得場所について詳しくは、『[Liberty プロファイル: ランタイム環境での既知の制約事項](#)』の『[Java の最小サポート・レベル](#)』を参照してください。

3. WebSphere eXtreme Scale 配布イメージを、`wlp-edition-8.6.0.0.jar` ファイルを解凍したディレクトリーに解凍します。

このイメージは、V8.6 で `wxs-wlp_8.6.0.0.jar` と呼ばれ、V8.6 Fix Pack 2 で `wxs-wlp_8.6.0.2.jar` と呼ばれる JAR ファイルとしてパッケージされています。配布イメージを解凍するには、JAR ファイルを実行します。例えば、eXtreme Scale のバージョンに応じて次のコマンドを実行します。


```
java -jar wxs-wlp_8.6.0.0.jar -acceptLicense
```

```
java -jar wxs-wlp_8.6.0.2.jar -acceptLicense
```

タスクの結果

ファイル `wxs-wlp_8.6.0.0.jar` ファイル (V8.6) および `wxs-wlp_8.6.0.2.jar` ファイル (V8.6 Fix Pack 2) を解凍する場合、両方の JAR ファイルを同じディレクトリーに解凍すると、eXtreme Scale は WebSphere Application Server Liberty プロファイル上にインストールされます。

親トピック: [Liberty プロファイルのインストール](#)

製品インストールのトラブルシューティング

IBM® Installation Manager は多くの IBM ソフトウェア製品のための共通インストーラーです。このバージョンの WebSphere® eXtreme Scale をインストールするには、このインストーラーを使用します。

タスクの結果

ロギングおよびトレースに関する注記:

- Installation Manager を開いて「ファイル」>「ログの表示」をクリックすると、ログを簡単に表示できます。表内のログ・ファイルを個々に選択して、「ログ・ファイルを開く」アイコンをクリックすることにより、個々のログ・ファイルを開くことができます。
- ログは、Installation Manager のアプリケーション・データ・ロケーションの logs ディレクトリーにあります。次に例を示します。

- **Windows** 管理インストール:

```
C:\Documents and Settings\All Users\Application Data\IBM\Installation Manager
```

- **Windows** 非管理インストール:

```
C:\Documents and Settings\user_name\Application Data\IBM\Installation Manager
```

- **UNIX** | **Linux** 管理インストール:

```
/var/IBM/InstallationManager
```

- **UNIX** | **Linux** 非管理インストール:

```
user_home/var/ibm/InstallationManager
```

- メイン・ログ・ファイルは、タイム・スタンプが記されている XML ファイルで、logs ディレクトリーにあります。これらのログ・ファイルは、任意の標準 Web ブラウザーを使用して表示できます。
- logs ディレクトリー内の log.properties ファイルは、Installation Manager が使用するロギングまたはトレースのレベルを指定します。WebSphere eXtreme Scale プラグインのトレースをオンにするには、例えば、以下のような内容の log.properties ファイルを作成します。

```
com.ibm.ws=DEBUG
com.ibm.cic.agent.core.Engine=DEBUG
global=DEBUG
```

必要に応じて Installation Manager を再始動します。Installation Manager によって、WebSphere eXtreme Scale プラグインのトレースが出力されます。

トラブルシューティングに関する注記:

- **UNIX** | **Linux** デフォルトで、一部の HP-UX システムは、ホスト名の解決に DNS を使用しないように構成されます。その場合、Installation Manager は外部のリポジトリーに接続できない場合があります。

リポジトリーを ping することはできますが、nslookup は何も返しません。

システム管理者と相談して、DNS を使用するようにご使用のマシンを構成するか、またはリポジトリーの IP アドレスを使用してください。

- 場合によっては、Installation Manager の既存のチェック・メカニズムを迂回する必要がある場合もあります。
 - 一部のネットワーク・ファイル・システムでは、ディスク・スペースが正しく報告されない場合があります。この場合、ディスク・スペース・チェックを迂回して、インストールを続行する必要があります。

ディスク・スペース・チェックを使用不可にするには、`IM_install_root/eclipse/configuration` の `config.ini` ファイルで以下のシステム・プロパティーを指定して、Installation Manager を再始動します。

```
cic.override.disk.space=sizeunit
```

ここで `size` は正整数です。`unit` は、バイトの場合は空白、キロの場合は `k`、メガバイトの場合は `m`、ギガバイトの場合は `g` です。例:

```
cic.override.disk.space=120 (120 バイト)
cic.override.disk.space=130k (130 キロバイト)
cic.override.disk.space=140m (140 メガバイト)
cic.override.disk.space=150g (150 ギガバイト)
cic.override.disk.space=true
```

Installation Manager は Long.MAX_VALUE のディスク・スペース・サイズを報告します。使用可能なディスク・スペースが非常に大容量の場合は、表示されずに N/A が表示されます。

- オペレーティング・システムの前条件チェックを迂回するには、`disableOSPrereqChecking=true` を `IM_install_root/eclipse/configuration` の `config.ini` ファイルに追加して、Installation Manager を再始動します。

これらの迂回メソッドのいずれかを使用する必要がある場合は、Installation Manager チェック・メカニズムを迂回しないソリューションの開発における支援を受けるため、IBM サポートにお問い合わせください。

- Installation Manager の使用について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

Installation Manager の最新バージョンの詳細については、リリース情報を参照してください。リリース情報にアクセスするには、以下のタスクを実行します。

- **Windows** 「スタート」 > 「プログラム」 > 「IBM Installation Manager」 > 「リリース情報」をクリックします。
- **UNIX | Linux** Installation Manager がインストールされているディレクトリーにある文書サブディレクトリーに移動し、`readme.html` ファイルを開きます。
- 製品をインストールしようとしている時に致命的エラーが生じた場合は、以下の手順に従ってください。
 - IBM サポートが後で確認する必要がある場合があるので、現在の製品インストール・ディレクトリーのバックアップ・コピーを取ってください。
 - Installation Manager を使用して、製品のインストール・ロケーション (パッケージ・グループ) にインストールしたものをすべてアンインストールします。このときエラーが生じる可能性もありますが、無視しても支障ありません。
 - 製品のインストール・ディレクトリーに残っているものをすべて削除します。
 - Installation Manager を使用し、同じロケーションか新しいロケーションに製品を再インストールします。

バージョンおよび履歴情報に関する注記: `versionInfo` コマンドおよび `historyInfo` コマンドは、システム上で実行されるインストール、アンインストール、更新、および、ロールバックのすべてのアクティビティーに基づいて、バージョンと履歴情報を戻します。

親トピック: [WebSphere DataPower XC10 アプライアンスのインストール](#)

WebSphere DataPower XC10 アプライアンス の更新

ご使用の WebSphere® DataPower® XC10 アプライアンス 環境に保守を適用するには、まずファームウェア更新を適用してください。その後、ご使用の WebSphere eXtreme Scale クライアント インストール済み環境に更新を適用することができます。

1. [ファームウェアの更新](#)

IBM® WebSphere DataPower XC10 アプライアンス ソフトウェアのアップグレードは、ファームウェア更新を使用して行うことができます。Fix Central Web サイトから新しいファームウェア・バージョンをダウンロードして、アプライアンス・ソフトウェアを更新します。

2. [WebSphere Application Server 上の WebSphere eXtreme Scale クライアント の更新](#)

WebSphere Application Server を新しいバージョンにマイグレーションするとき、WebSphere eXtreme Scale クライアント の構成を新しい WebSphere Application Server のインストール済み環境にマイグレーションすることもできます。

3. [IBM Installation Manager の使用によるフィックスパックのインストール](#)

IBM Installation Manager を使用して、WebSphere eXtreme Scale クライアント 製品オフリングに使用可能なフィックスパックでこの製品を更新することができます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してインストールできます。

4. [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

IBM Installation Manager を使用して WebSphere eXtreme Scale クライアント 製品オフリングを前のバージョンにロールバックできます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してアンインストールできます。

5. **2.5+** [WebSphere eXtreme Scale Client for .NET のアップグレード](#)

インストール・プログラムを実行して、既存の WebSphere eXtreme Scale Client for .NET インストール済み環境をアップグレードすることができます。プログラムは既存のインストール済み環境を検出し、該当するファイルを置き換えます。

ファームウェアの更新

IBM® WebSphere® DataPower® XC10 アプライアンス ソフトウェアのアップグレードは、ファームウェア更新を使用して行うことができます。Fix Central Web サイトから新しいファームウェア・バージョンをダウンロードして、アプライアンス・ソフトウェアを更新します。

始める前に

アプライアンス管理権限が割り当てられている必要があります。

ファームウェア更新プロセスを開始する前に、すべてのアクティブ・ジョブが完了していることを確認してください。ファームウェア更新プロセスによって、実行中のアクティブ・ジョブが中断されます。これにより、データの不整合が発生する可能性があります。更新プロセスを開始する前に、必ずこの動作を想定した対応策を立てておいてください。

重要: 新しいアプライアンスに初めてファームウェアをインストールする場合は、ファームウェア更新が完了してから **clear-all** コマンドをアプライアンスで実行してください。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。

このタスクについて

IBM から新しいファームウェアの更新をダウンロードし、アプライアンスを新しいファームウェアで更新することによって、アプライアンスのファームウェアの更新を行うことができます。ユーザー・インターフェース、またはコマンド行でファームウェア更新を実行することができます。

注意:

2.0 以前のバージョンから、バージョン **2.5** にアップグレードするときは、現在ワークロードにサービスを提供している集合でファームウェア・アップグレードを実行することはできません。集合をアップグレードすると、データ・グリッドにロードされたデータがすべて失われます。集合内のアプライアンスでファームウェア・アップグレードを実行している場合、集合内のすべてのアプライアンスがアップグレードされるまで、アップグレードしたアプライアンスは、完全には再始動しません。まだアップグレードされていない残りのアプライアンスが要求を処理します。集合内では **1** つのアプライアンスでファームウェア・アップグレード・プロセスを完了してから、別のアプライアンスでプロセスを開始する必要があります。すべてのアプライアンスが同じファームウェア・レベルになるまで構成変更は行わないでください。

手順

- ファームウェア・アップグレードをダウンロードします。

[IBM Fix Central](#) にアクセスし、ご使用のローカル・ファイル・システムにファームウェア更新をダウンロードします。「Fix Central」ページの「製品グループ」で「**WebSphere**」を選択し、製品リストから「**WebSphere DataPower XC10 アプライアンス**」を選択します。ご使用のアプライアンス・ハードウェアのタイプに適したファームウェア・アップグレードをダウンロードします。

- アプライアンス・タイプ 7199-92X のアプライアンス・ファームウェアの更新は、.scrypt3 という拡張子の付いた単一のファイルになります。

この scrypt ファイルには、実行する更新の安全性を保証するために署名が付けられています。ファームウェア・アップグレードの実行にユーザー・インターフェースを使用している場合は、ユーザー・インターフェースへのアクセスに使用するコンピューターに、このファイルを保存します。ファームウェア・アップグレードの実行にコマンド行インターフェースを使用している場合、アプライアンスからアクセス可能なサーバーにこのファイルを保存します。

- ユーザー・インターフェースでファームウェア更新を実行するには、以下を行います。

1. ユーザー・インターフェースにログオンします。
2. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースで、「アプライアンス」 > 「設定」をクリックします。
 - 「ホーム」パネルで、「ステップ **1**: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
3. 「ファームウェア」を展開します。アプライアンスにインストールされているファームウェアのレベルが表示されます。
4. 新しいファームウェア更新をインストールします。
 - a. 「参照...」をクリックし、新しいファームウェア更新ファイルを選択します。
 - b. 新しいファームウェア更新ファイルを選択して、「**OK**」をクリックします。
 - c. 「アップグレード」をクリックします。ファームウェア更新のアップロードに要する時間は、接続速度によって異なります。アップロードが完了し、ファームウェア更新が開始されると、メッセージが表示されます。更新が開始されると、アプライアンスが再始動します。ユーザー・インターフェースでは、画面に進行状況は示されず、変更も表示されません。変更が表示されないのは、アプライアンスの再始動時にはセッションが終了し、アップグレード・プロセス中にユーザー・インターフェースが使用できないためです。

重要: アプライアンスは、ファームウェア・アップグレードのプロセス中に数回再始動します。ファームウェア・アップグレード中に、プロセスを中断したり手動でアプライアンスを再始動したりしないでください。

ファームウェア更新が完了したことを確認するには、更新が完了したと予想される頃に、もう一度ログインする必要があります。実際のファームウェア更新にかかる時間は、開始後、平均して約 10 分から 15 分ですが、これより長くかかる場合もあります。ファームウェア更新の進行状況をモニターしたい場合は、アプライアンス始動状況ユーザー・インターフェース・パネルを使用するか、またはコマンド行インターフェースで **start-progress** コマンドを使用してください。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。

- コマンド行インターフェースでファームウェア更新を実行するには、以下を行います。

1. コマンド行を使用して、またはシリアル・コンソールからアプライアンスに接続します。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
2. .script ファイルをアプライアンスに置きます。コマンド行インターフェースから、次のコマンドを実行します。

```
file get <url_to_script_file> <firmware_file>
```

ここで、*url_to_script_file* は Fix Central から .script ファイルを保存したローカル・サーバーで、*firmware_file* はアプライアンスで使用する .script ファイルの名前です。

3. ファームウェア更新をインストールします。以下のコマンドを実行します。

```
firmware upgrade <firmware_file>
```

重要: ファームウェア・アップグレードのプロセス中、アプライアンスは数回再始動します。ファームウェア・アップグレード中に、プロセスを中断したり手動でアプライアンスを再始動したりしないでください。

4. アプライアンス始動の進行状況をモニターします。

2.5+ 以下のいずれかのオプションを使用します。

- **2.5+** アプライアンスの始動状況をユーザー・インターフェースで追跡することができます。Web ブラウザーのアドレス・バーで、次の URL およびポート情報を入力してください。https://<appliance_hostname>:9443/。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。
- **start-progress** コマンドを実行して、進行をモニターします。このコマンドが STARTED を返す場合、アプライアンスは使用可能です。

タスクの結果

アプライアンスには更新されたファームウェアのバージョンがあります。ファームウェアのアップグレードが正常に行われたことを確認するために、「アプライアンス」 > 「設定」パネルの「ファームウェア」セクションで現行ファームウェア・レベルを確認できます。このパネルには、アプライアンス・モデル・タイプとアプライアンスのシリアル番号も表示されます。集合内の他のアプライアンスに対して、この手順を繰り返します。

次のタスク

新しいアプライアンスに初めてファームウェアをインストールする場合は、ファームウェア更新が完了してから **clear-all** コマンドをアプライアンスで実行してください。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。

親トピック: [WebSphere DataPower XC10 アプライアンスの更新](#)

次のトピック: [WebSphere Application Server 上の WebSphere eXtreme Scale クライアントの更新](#)

関連タスク:

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)

WebSphere Application Server 上の WebSphere eXtreme Scale クライアントの更新

WebSphere® Application Server を新しいバージョンにマイグレーションするとき、WebSphere eXtreme Scale クライアントの構成を新しい WebSphere Application Server のインストール済み環境にマイグレーションすることもできます。

始める前に

- WebSphere eXtreme Scale バージョン 7 と WebSphere eXtreme Scale バージョン 8 をいずれも同じサーバー上にインストールすることを前提とします。
- WebSphere Application Server バージョン 7 を WebSphere Application Server バージョン 8 にマイグレーションします。詳しくは、[製品構成のマイグレーション](#)を参照してください。
- WebSphere eXtreme Scale クライアント バージョン 8 をお使いの WebSphere Application Server バージョン 8 のインストール済み環境にインストールします。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。WebSphere eXtreme Scale クライアントのマイグレーション・スクリプトはすべて、WebSphere eXtreme Scale クライアント バージョン 8.5 以降から実行する必要があります。例えば、バージョン 7.x からバージョン 8 にマイグレーションする場合、マイグレーション用のスクリプトを <WXS_v8_install_root>/bin ディレクトリーから実行します。

このタスクについて

WebSphere eXtreme Scale を統合している WebSphere Application Server の新しいバージョンをインストールするときは、まず、通常のプロセスで WebSphere Application Server をアップグレードします。次に、その新しいインストール済み環境に WebSphere eXtreme Scale クライアントの新しいバージョンをインストールします。その後、**xsmigration** スクリプトを使用して WebSphere eXtreme Scale 構成情報を新しい WebSphere Application Server インストール済み環境に移動できます。

手順

1. デプロイメント・マネージャー関連の構成をバージョン 7 からバージョン 8 にマイグレーションします。
 - a. WebSphere Application Server バックアップ・スクリプトを実行します。詳しくは、[WASPreUpgrade コマンド](#)を参照してください。
 - b. デプロイメント・マネージャーを停止します。
 - c. WebSphere eXtreme Scale クライアント 構成内のデプロイメント・マネージャー・サーバーにアクセスして、マイグレーション・スクリプトを実行します。
 - i. ディレクトリーを <WXS_v8_install_root>/bin に移動します。
 - ii. 以下のコマンドを実行します。

```
xsmigration.bat|sh -targetwashome <WAS8x_HOME>  
-sourcwashome <WAS7x_HOME> -targetprofilepath <WAS8x_DmgrProfile>  
-sourceprofilepath <WAS7x_DmgrProfile>
```

各部の意味は、次のとおりです。

- <WAS8x_HOME> は、WebSphere Application Server バージョン 8.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere8
- <WAS7x_HOME> は、WebSphere Application Server バージョン 7.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere7
- <WAS8x_DmgrProfile> は、WebSphere Application Server バージョン 8.x のデプロイメント・マネージャーのプロファイルが置かれているロケーションです。例: /opt/IBM/WebSphere8/profiles/DMgr01
- <WAS7x_DmgrProfile> は、WebSphere Application Server バージョン 7.x のデプロイメント・マネージャーのプロファイルが置かれているロケーションです。例: /opt/IBM/WebSphere7/profiles/DMgr01

2. アプリケーション・サーバー関連の構成をバージョン 7 からバージョン 8 にマイグレーションします。
 - a. ディレクトリーを <WXS_v8_install_root>/bin に移動します。
 - b. 以下のコマンドを実行します。

```
xsmigration.bat|sh -targetwashome <WAS8x_HOME>  
-sourcwashome <WAS7x_HOME> -targetprofilepath <WAS8x_AppServerProfile>  
-sourceprofilepath <WAS7x_AppServerProfile>
```

各部の意味は、次のとおりです。

- <WAS8x_HOME> は、WebSphere Application Server バージョン 8.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere8

- <WAS7x_HOME> は、WebSphere Application Server バージョン 7.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere7
- <WAS8x_AppServerProfile> は、WebSphere Application Server バージョン 8.x のアプリケーション・サーバー・プロファイルが置かれているロケーションです。例:
/opt/IBM/WebSphere8/profiles/AppServer01
- <WAS7x_AppServerProfile> は、WebSphere Application Server バージョン 7.x のアプリケーション・サーバー・プロファイルが置かれているロケーションです。例:
/opt/IBM/WebSphere7/profiles/AppServer01

3. WebSphere Application Server バージョン 8 のデプロイメント・マネージャーを再始動して、すべての管理対象ノードを同期化します。

親トピック: [WebSphere DataPower XC10 アプライアンスの更新](#)

前のトピック: [ファームウェアの更新](#)

次のトピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

IBM Installation Manager の使用によるフィックスパックのインストール

IBM® Installation Manager を使用して、WebSphere® eXtreme Scale クライアント 製品オフリングに使用可能なフィックスパックでこの製品を更新することができます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してインストールできます。

[GUI の使用によるフィックスパックのインストール](#)

IBM Installation Manager ウィザードを使用して、WebSphere eXtreme Scale クライアント を新しいバージョンに更新できます。

[コマンド行の使用によるフィックスパックのインストール](#)

IBM Installation Manager をコマンド行から使用して、WebSphere eXtreme Scale 製品オフリングに使用可能なフィックスパックで製品を更新することができます。

[応答ファイルの使用によるフィックスパックのインストール](#)

IBM Installation Manager で応答ファイルを使用して、WebSphere eXtreme Scale クライアント を新しいバージョンに更新できます。

親トピック: [WebSphere DataPower XC10 アプライアンス の更新](#)

前のトピック: [WebSphere Application Server 上の WebSphere eXtreme Scale クライアント の更新](#)

次のトピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

GUI の使用によるフィックスパックのインストール

IBM® Installation Manager ウィザードを使用して、WebSphere® eXtreme Scale クライアント を新しいバージョンに更新できます。

始める前に

WebSphere eXtreme Scale クライアントを更新する前に、アプライアンス・ファームウェアをアップグレードします。詳しくは、[ファームウェアの更新](#)を参照してください。WebSphere eXtreme Scale スタンドアロンまたは WebSphere eXtreme Scale for WebSphere Application Server 製品オフリング用のアップグレードに関する情報については、IBM ソフトウェア・サポート・センターにお問い合わせください。最新の情報は、IBM ソフトウェア・サポート・センターおよび [Fix Central](#) から入手できます。

IBM Installation Manager を使用して、以下の製品オフリングに製品保守を適用します。

- WebSphere eXtreme Scale クライアント (スタンドアロン環境)
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

Web ベースのサービス・リポジトリまたはローカル・サービス・リポジトリのロケーションがリストされ、チェックされているかどうか、あるいは Installation Manager 設定の「リポジトリ」パネルで「インストール中および更新中にサービス・リポジトリの検索」オプションが選択されているかどうかを確認してください。Installation Manager でのサービス・リポジトリの使用について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

このタスクについて

制約事項: Installation Manager を使用して、インストール済み環境をアップグレードしたり、完全な WebSphere Application Server プロファイルのフィーチャーを追加または削除したりすることはできません。

手順

1. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
2. システムにログオンします。
3. 更新される WebSphere Application Server インストール済み環境で、すべてのサーバーおよびアプリケーションを停止します。
4. Installation Manager を開始します。
5. 「更新」をクリックします。

注: 認証のプロンプトが出されたら、保護されている IBM ソフトウェア Web サイトへのアクセスに使用する IBM ID とパスワードを使用してください。

6. 更新するパッケージ・グループを選択します。

ヒント: 「すべて更新」を選択すると、Installation Manager は過去にインストールしたすべてのパッケージ・グループに対する更新の追加リポジトリおよび定義済みリポジトリをすべて検索します。このフィーチャーは、ターゲット・リポジトリ内に含まれるフィックスについて完全に制御できる場合のみ使用してください。インストールしたいフィックスのみを含んだ一連のカスタム・リポジトリを作成して指定しているのであれば、このフィーチャーを使用しても問題はありません。サービス・リポジトリの検索を使用可能にする場合や、他の Web ベースのライブ・リポジトリから直接フィックスをインストールする場合は、このオプションの選択は推奨されません。未選択であれば、後続のパネルで各オフリングに対してインストールするフィックスのみを選択できます。

7. 「次へ」をクリックします。
8. 更新後のバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
9. インストールするフィックスを選択します。

デフォルトでは、任意の推奨フィックスが選択されます。

推奨フィックスがある場合、推奨フィックスのみを表示し、非推奨フィックスを非表示にするオプションを選択できます。

10. 「次へ」をクリックします。

11. 使用条件の条項に同意し、「次へ」をクリックします。
12. 更新後のインストール済み環境に必要なオプション・フィーチャーを選択します。
13. 要約情報を検討して、「更新」をクリックします。
 - 正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。
 - インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。
14. 「終了」をクリックします。
15. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

親トピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

関連タスク:

[GUI の使用によるフィックスパックのアンインストール](#)

コマンド行の使用によるフィックスパックのインストール

IBM® Installation Manager をコマンド行から使用して、WebSphere® eXtreme Scale 製品オフラインに使用可能なフィックスパックで製品を更新することができます。

始める前に

WebSphere eXtreme Scale クライアントを更新する前に、アプライアンス・ファームウェアをアップグレードします。詳しくは、[ファームウェアの更新](#)を参照してください。WebSphere eXtreme Scale スタンドアロンまたは WebSphere eXtreme Scale for WebSphere Application Server 製品オフラインのアップグレードに関する情報については、IBM ソフトウェア・サポート・センターにお問い合わせください。最新の情報は、IBM ソフトウェア・サポート・センターおよび [Fix Central](#) から入手できます。

IBM Installation Manager を使用して、以下の製品オフラインに製品保守を適用します。

- WebSphere eXtreme Scale クライアント (スタンドアロン環境)
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

このタスクについて

制約事項: Installation Manager を使用して、インストール済み環境をアップグレードしたり、完全な WebSphere Application Server プロファイルの機能を追加または削除したりすることはできません。

手順

1. WebSphere eXtreme Scale 8.5 に有効なインテリム・フィックスとフィックスパックのリスト、および各フィックスに関する具体的な情報を入手するには、次のアクションを実行します。
 - a. [Fix Central](#) に移動します。
 - b. **WebSphere** を製品グループとして選択します。
 - c. WebSphere eXtreme Scale を製品として選択します。
 - d. インストール・バージョンとして **8.5** を選択します。
 - e. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - f. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - g. 各フィックスの下の「**詳細**」をクリックすると、そのフィックスの情報が表示されます。
 - h. **推奨:** インストールしたいフィックスパックの名前をメモしておいてください。
2. 次の手順を使用して、フィックスパックで WebSphere eXtreme Scale バージョン 8.5 を更新してください。
 - フィックスパックを含むファイルを Fix Central からダウンロードし、ローカル更新を行います。

フィックスパックを含む圧縮ファイルは Fix Central からダウンロードできます。フィックスパックの各圧縮ファイルには、そのフィックスパックの Installation Manager リポジトリが含まれていて、通常は .zip という拡張子が付いています。フィックスパック・ファイルをダウンロードして解凍した後、Installation Manager を使用してフィックスパックで WebSphere Application Server バージョン 8.x を更新します。

- a. フィックスパックをダウンロードするには、以下の手順を行います。
 - i. [Fix Central](#) に移動します。
 - ii. **WebSphere** を製品グループとして選択します。
 - iii. **WebSphere eXtreme Scale** を製品として選択します。
 - iv. インストール・バージョンとして **8.5** を選択します。
 - v. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - vi. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - vii. ダウンロードするフィックスパックを選択して、「**続行**」をクリックします。
 - viii. ダウンロード・オプションを選択して、「**続行**」をクリックします。
 - ix. ご使用条件に同意する場合は「**同意します**」をクリックします。
 - x. 「**今すぐダウンロード**」をクリックしてフィックスパックをダウンロードします。
 - xi. 圧縮ファイルをバイナリー形式でインストール先のシステムに転送します。
 - xii. 圧縮されたリポジトリ・ファイルをシステム上のディレクトリに解凍します。
- b. ダウンロードしたファイルからフィックスパックをインストールするには、次のアクションを実行します。

- i. システムにログオンします。
- ii. ご使用の環境で実行中のすべてのプロセスを停止します。WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
- iii. `Installation_Manager_binaries/eclipse/tools` ディレクトリーに移動します。ここでは、`Installation_Manager_binaries` が Installation Manager のインストール・ルート・ディレクトリーです。
- iv. フィックスパックをインストールします。

UNIX | Linux

```
./imcl install offering_ID offering_version,optional_feature_ID  
-installationDirectory product_installation_location  
-repositories location_of_expanded_files  
-acceptLicense
```

Windows

```
imcl.exe install offering_ID offering_version,optional_feature_ID  
-installationDirectory product_installation_location  
-repositories location_of_expanded_files  
-acceptLicense
```

ヒント:

- `offering_version` は、アンダースコアを使用してオファリング ID にオプションで付加することができます。これはインストール対象のオファリングの特定バージョンです (例: 8.5.0.20110503_0200)。
 - `offering_version` が指定されていない場合、オファリングの最新バージョンと、そのバージョンのすべてのインテリム・フィックスがインストールされます。
 - `offering_version` が指定されている場合、オファリングの指定バージョンがインストールされ、そのバージョンのインテリム・フィックスはインストールされません。

オファリングのバージョンは、リポジトリーに対して次のコマンドを実行すると、オファリング ID の最後にアンダースコアを使用して付加されている形で見つかります。

```
imcl listAvailablePackages -repositories source_repository
```

- また、`-installFixes` 引数に `none`、`recommended`、または `all` を指定することで、どのインテリム・フィックスをオファリングと一緒にインストールするのかを示すこともできます。
 - オファリングのバージョンが指定されていない場合、`-installFixes` オプションはデフォルトで `all` になります。
 - オファリングのバージョンが指定されている場合、`-installFixes` オプションはデフォルトで `none` になります。
 - コンマで区切られたフィーチャーのリストを追加できます。フィーチャーのリストが指定されていない場合は、デフォルトのフィーチャーがインストールされます。
- v. オプション: すべてのインストール済みパッケージをリストして、インストールを検証します。

UNIX | Linux

```
./imcl listInstalledPackages -long
```

Windows

```
imcl.exe listInstalledPackages -long
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

関連タスク:

[コマンド行の使用によるフィックスパックのアンインストール](#)

応答ファイルの使用によるフィックスパックのインストール

IBM® Installation Manager で応答ファイルを使用して、WebSphere® eXtreme Scale クライアント を新しいバージョンに更新できます。

始める前に

ヒント: 本書で説明した手順に代わる手段として、Installation Manager で、応答ファイルまたはコマンド行で **updateAll** コマンドを使用してすべてのインストール済みパッケージの検索と更新を行うことができます。このコマンドは、ターゲット・リポジトリ内にどのフィックスを含めるかについて、完全に制御できる場合のみ使用してください。インストールしたいフィックスのみを含んだ一連のカスタム・リポジトリを作成して指定しているのであれば、このコマンドを使用しても問題はありません。サービス・リポジトリの検索を使用可能にする場合や、他の Web ベースのライブ・リポジトリから直接フィックスをインストールする場合は、このオプションの選択は推奨されません。未選択であれば、コマンド行から **install** コマンドと共に **-installFixes** オプションを使用するか、応答ファイルで **installFixes** 属性を使用して、インストールするフィックスのみを選択できます。

手順

1. WebSphere eXtreme Scale クライアント に有効なインテリム・フィックスとフィックスパックのリスト、および各フィックスに関する具体的な情報を入手するには、次のアクションを実行します。
 - a. [Fix Central](#) に移動します。
 - b. **WebSphere** を製品グループとして選択します。
 - c. WebSphere eXtreme Scale クライアントを製品として選択します。
 - d. インストール・バージョンとして **8.x** を選択します。
 - e. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - f. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - g. 各フィックスの下の「**詳細**」をクリックすると、そのフィックスの情報が表示されます。
 - h. **推奨:** インストールしたいフィックスパックの名前をメモしておいてください。
2. 次の手順を使用して、フィックスパックで WebSphere eXtreme Scale クライアント を更新してください。
 - フィックスパックを含むファイルを Fix Central からダウンロードし、ローカル更新を行います。

フィックスパックを含む圧縮ファイルは Fix Central からダウンロードできます。フィックスパックの各圧縮ファイルには、そのフィックスパックの Installation Manager リポジトリが含まれていて、通常は .zip という拡張子が付いています。フィックスパック・ファイルをダウンロードして解凍した後、Installation Manager を使用してフィックスパックで WebSphere eXtreme Scale クライアント を更新します。
 - a. フィックスパックをダウンロードするには、以下の手順を行います。
 - i. [Fix Central](#) に移動します。
 - ii. **WebSphere** を製品グループとして選択します。
 - iii. **WebSphere eXtreme Scale** クライアントを製品として選択します。
 - iv. インストール・バージョンとして **8.6** を選択します。
 - v. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - vi. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - vii. ダウンロードするフィックスパックを選択して、「**続行**」をクリックします。
 - viii. ダウンロード・オプションを選択して、「**続行**」をクリックします。
 - ix. ご使用条件に同意する場合は「**同意します**」をクリックします。
 - x. 「**今すぐダウンロード**」をクリックしてフィックスパックをダウンロードします。
 - xi. 圧縮ファイルをバイナリー形式でインストール先のシステムに転送します。
 - xii. 圧縮されたリポジトリ・ファイルをシステム上のディレクトリに解凍します。

b. 次のアクションを実行します。

- i. システムにログオンします。
- ii. リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、**imutilsc** コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

- iii. `Installation_Manager_binaries/eclipse/tools` ディレクトリーに移動します。ここでは、`Installation_Manager_binaries` が Installation Manager のインストール・ルート・ディレクトリーです。
- iv. 応答ファイルを使用してフィックスパックをインストールします。

例:

- **Windows** 管理者または非管理者:

```
imcl.exe -acceptLicense
input C:¥temp¥update_response_file.xml
-log C:¥temp¥update_log.xml
-keyring C:¥IM¥im.keyring
```

- **UNIX** | **Linux** 管理者:

```
./imcl -acceptLicense
input /var/temp/update_response_file.xml
-log /var/temp/update_log.xml
-keyring /var/IM/im.keyring
```

- **UNIX** | **Linux** 非管理者:

```
./imcl -acceptLicense
input user_home/var/temp/update_response_file.xml
-log user_home/var/temp/update_log.xml
-keyring user_home/var/IM/im.keyring
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

関連タスク:

[応答ファイルを使用したフィックスパックのアンインストール](#)

IBM Installation Manager の使用によるフィックスパックのアンインストール

IBM® Installation Manager を使用して WebSphere® eXtreme Scale クライアント 製品オフリングを前のバージョンにロールバックできます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してアンインストールできます。

[GUI の使用によるフィックスパックのアンインストール](#)

IBM Installation Manager GUI を使用して、WebSphere eXtreme Scale クライアント を前のバージョンにロールバックできます。

[コマンド行の使用によるフィックスパックのアンインストール](#)

コマンド行から IBM Installation Manager を使用して、WebSphere eXtreme Scale クライアント を前のバージョンにロールバックできます。

[応答ファイルを使用したフィックスパックのアンインストール](#)

IBM Installation Manager で応答ファイルを使用して、WebSphere eXtreme Scale クライアント を前のバージョンにロールバックできます。

親トピック: [WebSphere DataPower XC10 アプライアンス の更新](#)

前のトピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

次のトピック: **2.5+** [WebSphere eXtreme Scale Client for .NET のアップグレード](#)

GUI の使用によるフィックスパックのアンインストール

IBM® Installation Manager GUI を使用して、WebSphere® eXtreme Scale クライアント を前のバージョンにロールバックできます。

始める前に

ロールバック・プロセス中は、Installation Manager から前バージョンのパッケージのファイルにアクセスする必要があります。デフォルトでは、これらのファイルはパッケージのインストール時に、ご使用のコンピューターに保管されます。デフォルト設定を変更した場合や保存されたファイルを削除した場合は、前バージョンのインストールに使用したリポジトリに Installation Manager からアクセスする必要があります。

このタスクについて

制約事項: Installation Manager を使用してインストール済み環境をロールバックしたり、フィーチャーを追加または削除したりすることはできません。

手順

- ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
- Installation Manager を開始します。
- 「ロールバック」をクリックします。
- ロールバックするパッケージ・グループを選択します。
- 「次へ」をクリックします。
- ロールバック後のバージョンを選択します。
- 「次へ」をクリックします。
- 要約情報を検討して、「ロールバック」をクリックします。
 - 正常にロールバックされた場合は、プログラムにより、ロールバックが正常に行われたことを示すメッセージが表示されます。
 - ロールバックが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。
- 「終了」をクリックします。
- 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

親トピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

関連タスク:

[GUI の使用によるフィックスパックのインストール](#)

コマンド行の使用によるフィックスパックのアンインストール

コマンド行から IBM® Installation Manager を使用して、WebSphere® eXtreme Scale クライアント を前のバージョンにロールバックできます。

始める前に

制約事項: この手順を実行するには、ご使用のシステムに Installation Manager バージョン 1.5 以降がインストールされている必要があります。

ロールバック・プロセス中は、Installation Manager から前バージョンのパッケージのファイルにアクセスする必要があります。デフォルトでは、これらのファイルはパッケージのインストール時に、ご使用のコンピューターに保管されます。デフォルト設定を変更した場合や保存されたファイルを削除した場合は、前バージョンのインストールに使用したリポジトリに Installation Manager からアクセスする必要があります。

このタスクについて

制約事項: Installation Manager を使用してインストール済み環境をロールバックしたり、完全な WebSphere Application Server プロファイルのフィーチャーを追加または削除したりすることはできません。

手順

1. オプション: リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、**imutilsc** コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

2. システムにログオンします。
3. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
4. Installation Manager をインストールしたディレクトリーの `eclipse/tools` サブディレクトリーに移動します。
5. **imcl** コマンドを使用して、この製品をロールバックします。

UNIX

Linux

```
./imcl rollback offering_ID_offering_version
-repositories source_repository
-installationDirectory installation_directory
-preferences preference_key=value
-properties property_key=value
-keyring keyring_file -password password
-acceptLicense
```

Windows

```
imcl.exe rollback offering_ID_offering_version
-repositories source_repository
-installationDirectory installation_directory
-preferences preference_key=value
-properties property_key=value
-keyring keyring_file -password password
-acceptLicense
```

ヒント:

- `offering_version` は、アンダースコアを使用してオフリング ID にオプションで付加することができますが、これはロールバック対象のオフリングの特定バージョンです (例: `8.5.0.20110503_0200`)。
 - `offering_version` が指定されていない場合、インストール済み環境はそのオフリングの以前インストールされたバージョンにロールバックし、そのバージョンのすべてのインテリム・フィックスがインストールされます。
 - `offering_version` が指定されている場合、インストール済み環境はそのオフリングの指定され

た前のバージョンにロールバックし、そのバージョンのインテリム・フィックスはインストールされません。

オフリングのバージョンは、**historyInfo** コマンドまたは **genHistoryReport** コマンドを `app_server_root/bin` ディレクトリーから実行する際に生成されるレポートの「パッケージ」セクションで、アンダースコアを使用してオフリング ID の最後に付加されている場合があります。

Installation Manager の使用について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

6. オプション: すべてのインストール済みパッケージをリストして、ロールバックを検証します。

UNIX | Linux

```
./imcl listInstalledPackages -long
```

Windows

```
imcl.exe listInstalledPackages -long
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

関連タスク:

[コマンド行の使用によるフィックスパックのインストール](#)

応答ファイルを使用したフィックスパックのアンインストール

IBM® Installation Manager で応答ファイルを使用して、WebSphere® eXtreme Scale クライアント を前のバージョンにロールバックできます。

始める前に

ロールバック・プロセス中は、Installation Manager から前バージョンのパッケージのファイルにアクセスする必要があります。デフォルトでは、これらのファイルはパッケージのインストール時に、ご使用のコンピューターに保管されます。デフォルト設定を変更した場合や保存されたファイルを削除した場合は、前バージョンのインストールに使用したリポジトリに Installation Manager からアクセスする必要があります。

このタスクについて

制約事項: Installation Manager を使用してインストール済み環境をロールバックしたり、完全な WebSphere Application Server プロファイルの機能を追加または削除したりすることはできません。

手順

1. オプション: リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、`imutilsc` コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

2. システムにログオンします。
3. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
4. 応答ファイルを使用して、この製品をロールバックします。

Installation Manager をインストールしたディレクトリーの `eclipse/tools` サブディレクトリーに移動して、この製品をロールバックします。

以下に例を示します。

- **Windows** 管理者または非管理者:

```
imcl.exe
input C:%temp%rollback_response_file.xml
-log C:%temp%rollback_log.xml
-keyring C:%IM%im.keyring
```

- **UNIX** | **Linux** 管理者:

```
./imcl
input /var/temp/rollback_response_file.xml
-log /var/temp/rollback_log.xml
-keyring /var/IM/im.keyring
```

- **UNIX** | **Linux** 非管理者:

```
./imcl
input user_home/var/temp/rollback_response_file.xml
-log user_home/var/temp/rollback_log.xml
-keyring user_home/var/IM/im.keyring
```

注: プログラムが重要なポストインストール指示を標準出力に書き込むことがあります。

Installation Manager の使用については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

5. オプション: すべてのインストール済みパッケージをリストして、ロールバックを検証します。

```
./imcl listInstalledPackages -long
```

Windows

```
imcl.exe listInstalledPackages -long
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

関連タスク:

[応答ファイルの使用によるフィックスパックのインストール](#)

WebSphere eXtreme Scale Client for .NET のアップグレード

インストール・プログラムを実行して、既存の WebSphere® eXtreme Scale Client for .NET インストール済み環境をアップグレードすることができます。プログラムは既存のインストール済み環境を検出し、該当するファイルを置き換えます。

始める前に

- WebSphere eXtreme Scale Client for .NET をアップグレードする前にアプライアンス・ファームウェアを更新してください。詳しくは、[ファームウェアの更新](#)を参照してください。
- WebSphere eXtreme Scale Client for .NET のアップグレードをダウンロードしてください。最新の情報は、IBM ソフトウェア・サポート・センターおよび [Fix Central](#) から入手できます。使用可能なダウンロードが新しいインストール済み環境とアップグレードの両方に適用されます。

このタスクについて

このインストール手順によって、既存のインストール済み環境は即座に置き換えられます。

手順

1. ご使用の環境で実行中のすべてのプロセスを停止します。
2. ウィザードを実行して、WebSphere eXtreme Scale Client for .NET アップグレードをインストールします。インストール・ウィザードを実行して、古いインストール済み環境が検出されたら、その既存のインストール済み環境をアップグレードすることを確認する必要があります。ウィザードの進行状況パネルに、古いバージョンとアップグレードされたバージョンが示されます。

タスクの結果

既存のすべての WebSphere eXtreme Scale Client for .NET コードが置き換えられます。この処理は、ディスク上とグローバル・アセンブリー・キャッシュ (GAC) 内の両方で行われます。公開されたポリシー・ファイルがディスクと GAC にインストールされ、既存のファイルに取って代わります。

次のタスク

- WebSphere eXtreme Scale Client for .NET を構成します。詳しくは、[WebSphere eXtreme Scale Client for .NET の構成](#)を参照してください。
- .NET アプリケーションを開発します。詳しくは、[.NET API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。

[WebSphere eXtreme Scale Client for .NET のフィックスパックのサイドバイサイド・インストールの作成](#)

サイドバイサイド・インストールを作成すると、WebSphere eXtreme Scale Client for .NET のフィックスパックの複数バージョンを同じサーバー上で実行することができます。旧バージョン向けに作成された既存の .NET アプリケーションは、旧クライアント向けに引き続き実行することができます。

親トピック: [WebSphere DataPower XC10 アプライアンス の更新](#)

前のトピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

WebSphere eXtreme Scale Client for .NET のフィックスパックのサイドバイサイド・インストールの作成

サイドバイサイド・インストールを作成すると、WebSphere® eXtreme Scale Client for .NET のフィックスパックの複数バージョンを同じサーバー上で実行することができます。旧バージョン向けに作成された既存の .NET アプリケーションは、旧クライアント向けに引き続き実行することができます。

始める前に

- WebSphere eXtreme Scale Client for .NET の旧バージョンがインストールされているシステムと、最新のフィックスパックをインストールする別のシステムが必要です。
- WebSphere eXtreme Scale Client for .NET の最新バージョンを、実動システムとは別のシステムにインストールします。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストール](#)を参照してください。

このタスクについて

WebSphere eXtreme Scale Client for .NET のフィックスパック・インストールを実行すると、既存のインストール済み環境が新しいバージョンで置き換えられます。環境に応じて、新しいバージョンにアップグレードする前にテストをしたいことがあります。また、アプリケーションによって、旧バージョンを使用させたいものと、最新バージョンを使用させたいものがあることもあります。WebSphere eXtreme Scale Client for .NET の手動インストールによって、複数のバージョンをサイドバイサイドで使用することができます。

WebSphere eXtreme Scale Client for .NET の新しいメジャー・リリースをインストールすると、サイドバイサイド・インストールが自動的に行われます。

手順

1. 最新のインストール済み環境を持つシステムから、[net_client_home](#) インストール・ディレクトリー全体とそのすべてのサブフォルダーをコピーします。これらのファイルは、旧バージョンがインストールされているターゲット・システム上の既存のインストール済み環境にある別のディレクトリーに配置します。このディレクトリーは *sidebyside_home* と呼ばれます。
2. 手動で、最新の WebSphere eXtreme Scale Client for .NET アセンブリーを *sidebyside_home/bin* ディレクトリーから GAC にインストールします。このインストールが完了すると、WebSphere eXtreme Scale Client for .NET アセンブリーの旧バージョンと新バージョンの両方が GAC にインストールされたこととなります。手動でアセンブリーを GAC にインストールする方法は、インストールされている Windows と .NET フレームワークのバージョンによって異なります。例えば、Windows エクスプローラーを使用して、アセンブリー・ファイルを %systemroot%\assembly ディレクトリーにコピーすることがあります。また、Microsoft から *gacutil.exe* ユーティリティーをダウンロードして使用するということがあります。
3. 新しいバージョンを使用するように WebSphere eXtreme Scale Client for .NET アプリケーションを更新します。以下のいずれかのオプションを使用します。
 - 更新したい各アプリケーションで、アセンブリー・リダイレクト・エレメントを構成ファイルに追加します。このエレメントは、古い WebSphere eXtreme Scale Client for .NET アセンブリーの参照を新しいアセンブリーにリダイレクトします。

```
<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <dependentAssembly>
      <assemblyIdentity name="IBM.WebSphere.Caching"
publicKeyToken="b439a24ee43b0816" />
      <bindingRedirect oldVersion="8.6.0.0-8.6.0.1" newVersion="8.6.0.2" />
    </dependentAssembly>
  </assemblyBinding>
</runtime>
```

- アプリケーション・プロジェクトのアセンブリー参照のリストにある新しいアセンブリーを使用して WebSphere eXtreme Scale Client for .NET をビルドし直します。

タスクの結果

- 更新したアプリケーションは WebSphere eXtreme Scale Client for .NET の新しいほうのバージョンを使用し、その他のアプリケーションは旧バージョンを使用します。
- 新しいバージョンを使用しているアプリケーションのログ・ファイルは、既存のアプリケーションが使用するのと同じログ・ディレクトリーにあります。
- デフォルトのクライアント構成は、従来どおり、[net_client_home/config](#) ディレクトリーにある古い WebSphere eXtreme Scale Client for .NET インストール済み環境から取得されます。別のプロパティー・ファイルを使用するには、新しいプロパティー・ファイルのパスを Connect() API に明示的に渡すようにしてくだ

さい。

警告: 行った変更は手動でアンインストールしなければなりません。手動インストール済み環境は、WebSphere eXtreme Scale Client for .NET インストール・プログラムによるアップグレードの対象外です。

親トピック: **2.5+** [WebSphere eXtreme Scale Client for .NET のアップグレード](#)

アプライアンスの構成

シリアル・コンソールを使用してアプライアンスを初期設定した後、アプライアンスの機能を有効にするために、ユーザー・インターフェースを使用してアプライアンスを構成する必要があります。

始める前に

- アプライアンス・ハードウェアのインストールおよび初期化を完了します。詳しくは、[WebSphere DataPower XC10 アプライアンスのインストール](#)を参照してください。
- 以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

シリアル・コンソールを使用する初期設定プロセスによって、アプライアンスはユーザー・インターフェースを使用して管理されるように準備されます。アプライアンスを使用できるように準備するには、追加設定を構成する必要があります。

注: これらのすべての設定が重要ですが、ドメイン・ネーム・システム (DNS) サーバーの追加、Network Time Protocol (NTP) サーバーの追加、およびメール送達の構成を適切に行うことが特に重要です。これらのタスクは、他の何らかの使用が発生する前に実行する必要があります。

[セキュリティ](#)

ユーザー・インターフェース・セキュリティおよびトランスポート・レベル・セキュリティを含めて、アプライアンス上のセキュリティのいくつかの側面を構成できます。

[ユーザーおよびグループの管理](#)

ユーザーおよびユーザー・グループが用意されているため、WebSphere DataPower XC10 アプライアンスへのアクセス・レベルを個人別に管理できます。ユーザー・グループを使用して、ユーザーのグループに許可を適用できます。

[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)

初回のシリアル接続時に、アプライアンスをネットワークに接続するために *mgmt* イーサネット・インターフェースを構成しました。追加のプライベート・イーサネット・ポートをユーザー・インターフェースで定義することができます。

[構成のインポートとエクスポート](#)

新規アプライアンスを構成する際、そのアプライアンスの構成設定をエクスポートして保管することができます。後に、アプライアンスの除去や再インストールを必要とする変更が発生した場合に、構成設定を損失することなく、保管しておいた構成データをインポートすることができます。

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

通信にドメイン・ネーム・システム (DNS) ルックアップ・サービスが使用されるため、IBM® WebSphere DataPower XC10 アプライアンスには DNS サーバーが必須です。アプライアンスの初期化時に、DNS サーバーを指定する必要があります。

[ホスト名に対する IP アドレスのマッピング](#)

アドレス情報を使用して TCP/IP ネットワーク・プロトコルに接続を作成するには、IP アドレスをホスト名に関連付ける必要があります。アプライアンス上で *etc/hosts* ファイルを編集することで、ホスト名に対して IP アドレスを解決できます。*etc/hosts* ファイルはユーザー・インターフェースから編集可能です。

[日時設定の管理](#)

集合内のご使用のアプライアンス全体で同期化された日時を維持するために、Network Time Protocol (NTP) サーバーを使用します。

[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)

アプライアンスのメール送達機能は、ユーザー・パスワードのリセットに使用します。ユーザーが新規パスワードを要求すると、アプライアンスは新規パスワードを記述した E メールをユーザーに送信します。

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

WebSphere DataPower XC10 アプライアンスは、ユーザー・インターフェースから再始動またはシャットダウンできます。

セキュリティ

ユーザー・インターフェース・セキュリティおよびトランスポート・レベル・セキュリティを含めて、アプライアンス上のセキュリティのいくつかの側面を構成できます。

[IBM WebSphere DataPower XC10 アプライアンス セキュリティ概要](#)

IBM WebSphere DataPower XC10 アプライアンスにより、アプライアンス本体と、アプライアンス上に保持されるデータ・グリッド (data grid) データの両方に対するアクセス権限をコントロールできます。

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティの構成](#)

WebSphere DataPower XC10 アプライアンスで提供されるセキュリティ機能の多くは、アプライアンスの構造内に構築されます。追加のセキュリティ・オプションをご使用の環境に提供するために、追加のセキュリティ設定が組み込まれています。

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

IBM WebSphere DataPower XC10 アプライアンスでユーザーを認証するために、任意で Lightweight Directory Access Protocol (LDAP) ディレクトリーを使用できます。

[ユーザーおよびグループの管理](#)

ユーザーおよびユーザー・グループが用意されているため、WebSphere DataPower XC10 アプライアンスへのアクセス・レベルを個人別に管理できます。ユーザー・グループを使用して、ユーザーのグループに許可を適用できます。

[データ・グリッドのセキュリティを使用可能に設定する](#)

データ・グリッドの作成後、データ・グリッド (data grid) のセキュリティはデフォルトで使用不可になります。データ・グリッド (data grid) のセキュリティ設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

Java

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

Java

[クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

アプリケーション用に構成しているデータ・グリッドでセキュリティを使用する場合、client.properties ファイルを構成して、そのデータ・グリッド・アプリケーションに受け渡し設定を含むようにします。

[データ・グリッド・アプリケーション用の TLS の構成](#)

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

[REST ゲートウェイ: セキュリティ構成](#)

REST ゲートウェイを介してデータ・グリッドにアクセスするには、データ・グリッドのセキュリティが使用可能に設定されているかどうかに関わらず、ユーザーは WebSphere DataPower XC10 アプライアンスに認証される必要があります。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを常に提供しなければなりません。REST ゲートウェイを介してデータ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。

親トピック: [アプライアンスの構成](#)

関連タスク:

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要

IBM® WebSphere® DataPower® XC10 アプライアンスにより、アプライアンス本体と、アプライアンス上に保持されるデータ・グリッド (data grid) データの両方に対するアクセス権限をコントロールできます。

アプライアンス・セキュリティー

アプライアンスをセキュア・ファウンデーションとするための主なフィーチャーを以下に示します。

不正開封防止ケースに入れられたアプライアンス

シャーシに侵入検知スイッチがあり、絶えずモニターされています。スイッチが起動した場合、アプライアンスは始動しません。アプライアンスを再び始動するには、アプライアンスを IBM に返却する必要があります。ケースの不正開封防止ねじなど、ケースが開かれないようにするための追加部品も使用されています。アプライアンスは、ケースを開かずにアプライアンスの背面からお客様が交換可能なアイテムにアクセスできるように設計されています。

シェル経由でのオペレーティング・システムへのアクセス不可

アプライアンスのオペレーティング・システムにはコマンド・シェルがありません。設計上、セキュリティーの脆弱性を削減するために、アプライアンスにはコマンド・インタープリターが組み込まれていません。アプライアンスには、オペレーティング・システムのユーザー ID が 1 つだけ存在します。使用可能なシェルがないため、ユーザー ID を使用して外部からアプライアンスにログオンすることはできません。

ユーザー提供のロジックを、アプライアンスで実行することはできません。

アプライアンスでは、ユーザーが実行可能スクリプトまたはコードをアップロードすることはできません。上記に関する唯一の例外は、システム・ファームウェア更新です。システム・ファームウェア更新では、スクリプトを実行して、更新されたファームウェアをアプライアンスにインストールできます。このようなシステム更新は、予防措置としてファームウェア製造元によって署名されています。ユーザー提供の非トラステッド・ソフトウェアをアプライアンスで実行することはできません。

データ・グリッドセキュリティー

ご使用のデータ・グリッドに含まれる情報へのアクセスをコントロールできます。ご使用のデータ・グリッド (data grid) でセキュリティーを有効にしないと、どのようなアプリケーションからでもデータ・グリッド (data grid) 内の情報にアクセス可能になります。データ・グリッド (data grid) において一般的なセキュリティーを有効にすることができますが、この場合はアプライアンス上にユーザー・アカウントとパスワードを持つユーザーなら誰もがデータ・グリッド (data grid) にアクセスできます。データ・グリッド (data grid) で許可を使用可能に設定することにより、特定のユーザーまたはユーザー・グループにアクセスを制限することもできます。

Transport Layer Security (TLS)

鍵ストア、トラストストア、および証明書別名を構成することにより、TLS を使用してデータ・グリッドおよびユーザー・インターフェースを保護できます。TLS の設定は、集合内のすべてのアプライアンスに適用されます。

ユーザーおよびユーザー・グループ

アプライアンスの管理およびデータ・グリッド (data grid) のセキュリティーの両方に対して、ユーザーおよびユーザー・グループの許可を定義できます。

親トピック: [セキュリティー](#)

関連概念:

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[クライアント・セキュリティーの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

[クライアント・プロパティー・ファイル](#)

WebSphere Application Server 用の Transport Layer Security (TLS) の構成

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

始める前に

- TLS は、バージョン 1.0.0.4 以降で構成できます。
 - WebSphere® eXtreme Scale クライアント、バージョン 7.1 フィックス 1 以降を使用している必要があります。
 - アプライアンス管理権限が割り当てられている必要があります。
 - アプライアンス構成に追加する、鍵ストアまたはトラストストアと、関連付けられたパスワードが必要です。
 - 既存のトラストストアを変更したければ、アプライアンスからトラストストアをダウンロードすることができます。
 - クライアントの公開証明書を使用してトラストストアを更新する必要があります。アプライアンスは、接続するクライアントを信頼する必要があります。
 - 指定されたトラストストアには、鍵ストア内のエントリーに対応する公開証明書が含まれている必要があります。アプライアンスの証明書別名の可能な構成オプションとして鍵ストアの証明書別名を指定できるようにするためには、その鍵ストアの証明書別名をトラストストア内で信頼する必要があります。
 - WebSphere Application Server のグローバル・セキュリティー設定によって、サーバーが WebSphere DataPower® XC10 アプライアンス への接続を 試行する方法が決まります。
 - グローバル・セキュリティー設定が使用不可の場合は、TCP/IP を使用して接続が試行されます。
 - グローバル・セキュリティー設定が使用可能な場合は、アプライアンスの公開証明書を WebSphere Application Server トラストストアに追加する必要があります。
- WebSphere DataPower XC10 アプライアンスの構成が「**TLS 必須**」になっている場合は、グローバル・セキュリティーを使用可能にする必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#)を参照してください。

このタスクについて

TLS 設定は、ユーザー・インターフェースおよびデータ・グリッドに適用されます。設定は、集合内のすべてのアプライアンスに適用されます。

手順

1. **WebSphere Application Server では必須:** アプライアンスの公開証明書を WebSphere Application Server のデフォルト・トラストストアに追加します。

- デフォルトのアプライアンス・トラストストアを使用する場合:

デプロイメント・マネージャーの `was_root/bin` ディレクトリーから `addXC10PublicCert.py` スクリプトを実行します。以下のコマンドを使用してください。

```
wsadmin -lang jython -f addXC10PublicCert.py
```

- アプライアンスのカスタム・キーを使用する場合:

`-certPath` コマンド行オプションを使用して、デプロイメント・マネージャーの `was_root/bin` ディレクトリーから `addXC10PublicCert.py` スクリプトを実行します。`-certPath` コマンド行オプションの値は、アプライアンスの鍵ストア用に構成された別名に対応する公開証明書のディスクの場所です。

```
wsadmin -lang jython -f addXC10PublicCert.py -certPath ./trustStore.jks
```

2. **WebSphere Application Server に必要:** アプライアンスのトラストストアと WebSphere Application Server の公開証明書をダウンロードし、`keytool` ユーティリティーを実行して、証明書をトラストストアに追加します。このツールは、WebSphere Application Server の証明書を組み込むようにアプライアンス・トラストストアを更新します。

- a. デフォルトのアプライアンス・トラストストアを使用する場合は、アクティブなトラストストアをダウンロードします。「集合」 > 「設定」 > 「**Transport Layer Security (TLS)**」をクリックします。「アクティブなトラストストアのダウンロード」をクリックし、ディスク上のファイルの保存場所 (例えば `/downloads/trustStore.jks` ディレクトリーなど) を覚えておきます。
- b. WebSphere Application Server の公開証明書を抽出します。
 - i. WebSphere Application Server 管理コンソールで、「セキュリティー」 > 「**SSL 証明書および鍵管理**」 > 「鍵ストアと証明書」をクリックします。
 - ii. 「鍵ストアの使用法」から「ルート証明書鍵ストア」を選択します。
 - iii. 「**DmgrDefaultRootStore**」を選択します。
 - iv. 「個人証明書」を選択します。

- v. ルート鍵ストア内の証明書の横にあるチェック・ボックスをクリックします。抽出する証明書の完全修飾ファイル名 (例: /certificates/public.cer) を指定します。
- vi. コマンド行ウィンドウで、次のコマンドを実行します: `cd /java_home/bin`
- vii. `keytool` ユーティリティを実行します。

```
keytool -import -noprompt -alias "example alias" -keystore
/downloads/trustStore.jks -file /certificates/public.cer -storepass
xc10pass -storetype jks
```

- viii. 追加でインポートする証明書があれば、この手順を繰り返し、証明書を抽出して `keytool` ユーティリティを再度実行します。
3. 鍵ストアとトラストストアの情報をアプライアンスにアップロードします。アプライアンス・ユーザー・インターフェイスで、「集合」 > 「設定」 > 「**Transport Layer Security (TLS)**」をクリックします。WebSphere Application Server の手順を完了したら、更新済みの /downloads/trustStore.jks ファイルをアップロードします。鍵ストアまたはトラストストアをアップロードした後、関連付けられたパスワードを更新する必要があります。デフォルトのトラストストアを使用する場合、パスワードは xc10pass です。
4. 集合の証明書別名を選択します。
5. トランスポート・タイプを指定します。以下のトランスポート・タイプ設定のいずれかを選択してください。
 - **TLS サポート**: データ・グリッドは TCP/IP、SSL、または TLS を使用して通信を行います。ユーザー・インターフェイスは HTTP および HTTPS を使用してアクセスできます。
 - **TLS 必須**: データ・グリッドは SSL または TLS のみを使用して通信を行います。ユーザー・インターフェイスは HTTPS のみを使用してアクセスできます。
 - **データ・グリッド TLS 使用不可**: データ・グリッドは非セキュア接続を使用して通信を行います。ユーザー・インターフェイスは HTTP および HTTPS を使用してアクセスできます。
6. 通信を可能にするための信頼できる証明書の送付をクライアントに要求するために、「**クライアント証明書認証を使用可能にする**」を選択します。
7. 「**TLS 設定の送信**」をクリックして、変更を構成に保存します。

タスクの結果

TLS 構成の変更を完了するために、集合を再始動する必要があります。

集合が再始動しているときには、ユーザー・インターフェイスの限られた一部分がアクセス可能です。ユーザー・インターフェイスの一部分にアクセスできない場合は、適当な時間を置いて、再度要求を実行依頼してください。「タスク」パネルに成功の状況が表示されて、いくつかの TLS 変更が完了したことが自動的に示されます。

アプライアンスが使用する証明書別名を変更した場合、ブラウザを再始動したり、ユーザー・インターフェイスをログアウトして再度ログインしたり、ブラウザ・プロンプトで新しい証明書を信頼したりする必要が生じる場合もあります。

クライアント認証が使用可能なときにユーザー・インターフェイスを使用できないようであれば、信頼できるクライアント証明書をブラウザにインポートしたかどうか確認してください。信頼できるクライアント証明書がブラウザにインポートされていない場合は、ユーザー・インターフェイスにアクセスできません。ユーザー・インターフェイスに正常にログオンした後、タスクによって、TLS 構成の成功が示されます。

次のタスク

ベスト・プラクティス

- 別のアプライアンスからユーザー・インターフェイスにアクセスする際、ブラウザからの警告を回避するために、鍵ストアの証明書の共通名 (CN) に、ワイルドカードを含めることを考慮してください。各アプライアンスは、TLS 構成に、証明書別名で指定されたものと同じ証明書を使用します。例えば、myhost.mycompany.com ではなく、*.mycompany.com を使用して、証明書を mycompany ドメイン内のすべてのホストに有効にすることができます。
- プライベート認証局 (CA) を使用して、TLS 構成で選択した証明書別名に関連付けられた証明書に署名することもできます。そうすると、CA 証明書をブラウザにインポートして、プライベート CA で署名された証明書によって、プロンプトの出力なしにそれぞれの集合を信頼することができます。一般に、プライベート CA の使用は、専用イントラネットでのアクセスにのみ適しています。

親トピック: [セキュリティ](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティ概要](#)
[ユーザー許可](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェイス・セキュリティの構成](#)
[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)
[ユーザーおよびグループの管理](#)
[データ・グリッドのセキュリティを使用可能に設定する](#)

Java [クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)
[データ・グリッド・アプリケーション用の TLS の構成](#)
[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)
[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)
[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)
[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)
[xscmd ユーティリティーによるモニター](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)
[splicer.properties ファイル](#)

関連情報:

[IKEYMAN グラフィカル・インターフェースを使用した鍵の管理](#)

IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成

WebSphere® DataPower® XC10 アプライアンス で提供されるセキュリティー機能の多くは、アプライアンスの構造内に構築されます。追加のセキュリティー・オプションをご使用の環境に提供するために、追加のセキュリティー設定が組み込まれています。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

アプライアンスに組み込まれるセキュリティー機能について理解するには、『[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)』を参照してください。

このタスクについて

アプライアンスのセキュリティー適用範囲を広げるために、ユーザーの振る舞いを制御するための オプションをいくつか構成できます。

手順

- 「設定」パネルにナビゲートします。セキュリティー・オプションを管理するには、以下のいずれかの方法で「設定」パネルまでナビゲートしてください。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ようこそ」ページで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
- 「セキュリティー」を展開します。
- 使用するセキュリティー権限を設定します。
 - 「新規ユーザーに自分のアカウントの作成を許可する」フィールドを設定します。このフィールドのデフォルト値は「使用不可」です。このフィールドには、ユーザーが自分自身のアカウントを作成できるかどうかを指定します。このフィールドが「使用可能」の場合、「登録」ボタンがログイン画面に表示されます。ユーザー自己登録について詳しくは、『[新規ユーザー・アカウントの登録](#)』を参照してください。
 - 「シリアル・コンソールからのパスワードのリセットを許可する」フィールドを設定します。このフィールドのデフォルト値は「使用不可」です。

「使用不可:」SMTP サーバーおよびxcadmin ユーザーの E メール・アドレスを必ず構成してください。これらの構成により、xcadmin パスワードを忘れた場合でも、パスワードをリセットする方法が用意されます。このフィールドが使用不可で、かつこれらの構成が行われていない場合、忘れた xcadmin パスワードをリセットすることができず、再設定のためにアプライアンスを IBM に返却する必要があります。

「使用可能:」その他の資格情報を必要とせず、また SMTP メッセージが表示されることもなく、シリアル接続を使用して xcadmin ユーザーのパスワードをリセットできます。このオプションを選択している場合は、WebSphere DataPower XC10 アプライアンス への物理的アクセスが通常よりもさらに重要です。マシンへの物理的アクセスが可能であれば、すべてのユーザーがアプライアンスに対する管理者権限を取得できます。
- アプライアンスを Lightweight Directory Access Protocol (LDAP) ディレクトリーを使用してユーザーを認証するように構成します。LDAP ディレクトリーを使用した認証のためのアプライアンスの構成について詳しくは、『[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)』を参照してください。

タスクの結果

上記の手順を正常に終わると、アプライアンスでの特定のセキュリティー関連シナリオの扱い方、およびユーザー・インターフェースへのアクセスに外部認証を使用するかどうかの指定が完了します。

次のタスク

ユーザーとグループを構成してユーザー・インターフェースへのアクセスを提供します。また、ユーザーとグループを使用してデータ・グリッドへのアクセスを提供します。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成

IBM® WebSphere® DataPower® XC10 アプライアンス でユーザーを認証するために、任意で Lightweight Directory Access Protocol (LDAP) ディレクトリーを使用できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーの認証に LDAP サーバーを使用するのはオプションです。IBM WebSphere DataPower XC10 アプライアンスのユーザーを LDAP ディレクトリーで認証するために、「ユーザー名」属性が使用されます。主アプライアンス管理者は別にして、LDAP ディレクトリーに登録されていないユーザーを認証することはできません。

2.5+ SSL 接続経由で LDAP の認証を行うように WebSphere DataPower XC10 アプライアンス を構成することができます。これは、LDAP サーバーへの接続のための Lightweight Directory Access Protocol over SSL (LDAPS) URL を指定して行います。LDAPS を使用する場合は、LDAP サーバーの SSL 証明書が信頼されるようにアプライアンスのトラストストアを変更する必要があります。この証明書の発行元が認証局である場合は、その認証局のルート署名者証明書をアプライアンスのトラストストアに追加する必要があります。LDAP サーバーの SSL 証明書が自己署名証明書である場合は、その証明書自体をアプライアンスのトラストストアに追加する必要があります。

2.5+ LDAP 認証を構成している場合、構成した基本ドメイン・ネームとユーザー用の構成済み検索フィルターが一致している LDAP ユーザー ID であれば、アプライアンスへの認証ができます。各ユーザーは、その特定ユーザーに付与される許可とデータ・グリッド権限、さらにそのユーザーが所属する LDAP グループに付与される許可とデータ・グリッド権限を保有します。個々のユーザーを集合に追加する必要はありません。すべての LDAP ユーザーは、Everyone グループに付与されている許可とアクセス権限も保有します。個々のユーザーにデータ・グリッド権限と許可が構成されるようにするために、ユーザーを集合に追加することが必要な場合があります。

LDAP 認証を構成している場合、LDAP グループのみを集合に追加できます。アクセス権限と許可は、指定されたグループに対して付与できます。

LDAP 認証を構成している場合、アプライアンスの管理コンソールやアプライアンス・プログラミング・インターフェースを使用して、グループのメンバーを追加/削除することはできません。グループ・メンバーシップは、LDAP ディレクトリーの管理ツールを使用して管理します。

V2.5 より前のリリースの IBM WebSphere DataPower XC10 アプライアンス では、集合に明確に追加されたユーザーのみが、許可やアクセス権限を付与されます。汎用的な LDAP アクセスのサポートは V2.5 で追加されています。V2.5 より前のリリースでは、グループが集合に追加されるときに、アプライアンスが、LDAP からグループ・メンバーシップをインポートします。その後は、アプライアンス上でグループが保守されます。このため、メンバーシップは、LDAP に保管されていたものとは異なるものとなる可能性があります。V2.5 からは、LDAP 認証が構成されている場合のグループ・メンバーシップは、常に、LDAP サーバーに照会することで解決されます。

マイグレーションの考慮事項: 以前のファームウェア・バージョンを実行するアプライアンスと V2.5 が一緒に含まれる集合では、LDAP 内に存在するがアプライアンス集合に保管されていないユーザーは、以前のファームウェアを使用するデバイス上の制限付きリソースにアクセスできません。したがって、すべてのデバイスがアップグレードされるまで、クライアントは、アプライアンス集合に追加されているユーザー ID しか使用できません。

集合に V2.5 より古いファームウェア・バージョンを実行するメンバーが含まれている場合は、以前のアプライアンス上に保管されているグループ・メンバーシップが、LDAP に保管されているものとは異なる可能性があります。こうした矛盾が原因で問題が発生する可能性があります。例えば、ユーザー ID が以前のアプライアンスに保管されているグループに含まれ、許可やアクセス権限がそのグループに関連付けられているものの、そのグループが LDAP 内に存在しない場合、またはそのアプライアンスに保管されたグループのメンバーシップが、LDAP 内にあるものとは異なる場合、そのユーザー ID は、新しいアプライアンス上の制限付きリソースにアクセスできない可能性があります。こうした動作が起きるのは、V2.5 のアプライアンスが LDAP を直接検査し、グループ・メンバーシップのローカル・バージョンを検査していないためです。マイグレーションを行う場合は、アプライアンス・データのアクセスに使用するユーザー ID が、個々のユーザー ID に関連付けられた必要なアクセス権限や許可を持っていることを必ず確認してください。また、それらのユーザー ID が、必要な許可を持つ LDAP グループのメンバーになっていることも確認してください。

手順

1. 「設定」パネルにナビゲートします。次のいずれかの方式を使用します。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ようこそ」ページで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「セキュリティ」を展開します。
3. アプライアンスを LDAP ディレクトリーを使用してユーザーを認証するように構成します。
 - a. LDAP 認証を使用可能にするには、「LDAP 認証を使用可能に設定」の横にあるチェック・ボックスを選択

します。「**LDAP 認証を使用可能に設定**」チェック・ボックスは、デフォルトでは選択されていません。このチェック・ボックスを選択すると、WebSphere DataPower XC10 アプライアンス でログイン時にユーザーを認証するために指定された LDAP サーバーを使用できるようになります。

- b. JNDI プロバイダー URL を入力します。非 SSL LDAP の例:

```
ldap://mycompany.com:389/
```

または

```
ldap://mycompany.com/
```

ポートが明示的に指定されていない場合は、デフォルトのポート番号が 389 になります。SSL LDAP の例:

```
ldaps://mycompany.com:636/
```

または

```
ldaps://mycompany.com/
```

ポートが明示的に指定されていない場合は、デフォルトのポート番号が 636 になります。

- c. JNDI ベース DN (ユーザー) を入力します。例:

```
CN=users,DC=mycompany,DC=com
```

- d. JNDI ベース DN (グループ) を入力します。例:

```
DC=mycompany,DC=com
```

- e. 検索フィルター (ユーザー) を入力します。例:

```
(&(sAMAccountName={0})(objectcategory=user)) または uid={0}
```

注: ユーザー ID はプレースホルダー {0} に組み込まれます。{0} は、ログイン画面で入力したログイン・ユーザー ID で置き換えられます。

- f. JNDI セキュリティー認証を入力します。このフィールドは、LDAP サーバーで匿名 LDAP 照会の実行が許可されていない場合以外はオプションです。例:

```
CN=Administrator,CN=users,DC=mycompany,DC=com
```

- g. パスワードを入力します。このフィールドは JNDI セキュリティー資格情報であり、LDAP サーバーで匿名 LDAP 照会の実行が許可されていない場合以外はオプションです。

4. 構成した LDAP 認証設定をテストします。LDAP サーバーで認証を構成するために使用した設定をテストできます。このセクションでは、LDAP 照会を実行し、指定されたユーザーおよびグループを探すことができます。

- a. 「**LDAP 認証設定のテスト**」をクリックし、このセクションを展開します。
b. ユーザー名をテストするには、「LDAP ユーザー名」フィールドにユーザー名を入力し、関連する「**LDAP 照会のテスト**」ボタンをクリックします。例:

```
test_user@us.ibm.com
```

照会が正常に行われると、「*Found LDAP User DN: <ユーザー情報>*」というメッセージが表示されます。照会が正常に行われなかった場合は、エラー・メッセージが表示されます。

- c. グループ名をテストするには、「LDAP グループ名」フィールドにグループ名を入力し、関連する「**LDAP 照会のテスト**」ボタンをクリックします。例:

```
Test Group
```

照会が正常に行われると、「*Found LDAP Group DN: <ユーザー情報>*」というメッセージが表示されます。照会が正常に行われなかった場合は、エラー・メッセージが表示されます。

タスクの結果

ユーザー・インターフェースへアクセスする際の外部認証に LDAP ディレクトリーが指定されます。

次のタスク

ご使用の環境のさまざまな領域に対するユーザー・アクセスを制御する方法を理解することは、セキュリティー・ソリューションの重要な部分です。ユーザーおよびグループとその許可の管理方法について詳しくは、『[ユーザーおよびグループの管理](#)』を参照してください。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)
[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

ユーザーおよびグループの管理

ユーザーおよびユーザー・グループが用意されているため、WebSphere® DataPower® XC10 アプライアンス へのアクセス・レベルを個人別に管理できます。ユーザー・グループを使用して、ユーザーのグループに許可を適用できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーおよびユーザー・グループは、WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用して管理できます。

[ユーザーの作成](#)

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ユーザーが WebSphere DataPower XC10 アプライアンス にアクセスし、これを管理できるようにするには、以下の手順で新規アカウントを作成します。

[ユーザーの管理](#)

ユーザーの作成後に追加の権限が必要になった場合、ユーザー設定を手動で変更する必要があります。また、この手順は、情報が変わった場合に、ユーザーを変更するためにも使用できます。

[ユーザーの除去](#)

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ただし、特定のユーザーが不要になった場合は、WebSphere DataPower XC10 アプライアンス からそのユーザーを除去できます。

[新規ユーザー・アカウントの登録](#)

管理者が「[新規ユーザーに自分のアカウントの作成を許可する](#)」オプションを使用可能にしている場合、自分自身のユーザー・アカウントを作成できます。このタスクを使用して、ユーザー・アカウントをログイン画面から作成します。

[ユーザー・グループの作成](#)

特定の WebSphere DataPower XC10 アプライアンス リソースに対するユーザーの許可の管理を容易にするために、ユーザー・グループを作成できます。

[ユーザー・グループの管理](#)

ユーザー・グループを作成した当初は、このユーザー・グループにはメンバーとして指定されたユーザーは存在しません。LDAP 認証が使用可能でない場合、手動でユーザーをグループに追加する必要があります。

[ユーザー・グループの除去](#)

不要になったユーザー・グループは WebSphere DataPower XC10 アプライアンス から除去できます。

[ユーザー許可](#)

ユーザー許可は、各ユーザーに向けて表示できるパネルを決定したり、特定オブジェクトに対するユーザー・アクセスを決定したりするために定義します。

親トピック: [アプライアンスの構成](#)

親トピック: [セキュリティ](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[セキュリティ](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティの構成](#)

[データ・グリッド のセキュリティを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

ユーザーの作成

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ユーザーが WebSphere® DataPower® XC10 アプライアンス にアクセスし、これを管理できるようにするには、以下の手順で新規アカウントを作成します。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。 Lightweight Directory Access Protocol (LDAP) を使用してユーザーを認証する場合、まず、登録しようとしているユーザーが LDAP リポジトリに存在していなければなりません。LDAP 構成の作成について詳しくは、『[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)』を参照してください。

このタスクについて

WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用してユーザーを作成するには、以下の手順に従ってください。

手順

1. 「ユーザー」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「集合」 > 「ユーザー」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「ユーザーの作成」リンクをクリックします。
2. 追加アイコン (+) をクリックして、新規ユーザーの追加を開始します。
3. 「ユーザー名」フィールドに ID を入力します。このフィールドの値は 64 文字まで指定可能です。このフィールドをブランクにすることはできません。すべての英数字と大部分の特殊文字を使用することができます。スペースと特殊文字 < # は使用できません。このフィールドは、ユーザーの作成後には変更できません。LDAP 認証を使用する場合、登録しようとしているユーザーは LDAP リポジトリに存在していなければなりません。
4. オプション: 「フルネーム」フィールドにユーザーの名前を入力します。このフィールドは、ユーザー・インターフェースでの表示用に使用されます。このフィールドに値を入力しなければ、ユーザー名が表示されません。ユーザーを作成した後は、そのユーザーのみがこのフィールドを編集できます。ユーザーが作成された後、管理者はこのフィールドの値を変更できません。
5. 「パスワード」フィールドにユーザーのパスワードを入力します。パスワードは、「ユーザー名」フィールドに使用できる文字と同じ文字を使用できます。Simple Mail Transfer Protocol (SMTP) が使用可能であれば、ユーザーの作成時に「パスワード」フィールドは空のままにすることができ、パスワードは自動的に生成されます。LDAP 認証が使用可能になっている場合は、LDAP レジストリーのパスワードが認証に使用されるため、「パスワード」フィールドは表示されません。「パスワードの確認」フィールドに、上と同じユーザーのパスワードを再入力します。
6. 「E メール・アドレス」フィールドにユーザーの有効な E メール・アドレスを入力します。このフィールドには、ユーザーがパスワードを忘れた場合の新規パスワードの供給や、その他の通知の配信に使用される E メール・アドレスを指定します。Simple Mail Transfer Protocol (SMTP) サーバーを使用するときに E メール・アドレスが必要です。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)を参照してください。
7. 「OK」をクリックします。

タスクの結果

これで、ユーザー・インターフェースへのログオンに使用できる新規ユーザー・アカウントができました。

次のタスク

ユーザーの作成当初は、デフォルトの権限のみが割り当てられています。新規ユーザーに追加の許可を与える場合は、ユーザーへの許可の追加について『[ユーザーの管理](#)』を参照してください。新規ユーザーをユーザー・グループに追加する方法については、『[ユーザー・グループの管理](#)』を参照してください。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザーの管理

ユーザーの作成後に追加の権限が必要になった場合、ユーザー設定を手動で変更する必要があります。また、この手順は、情報が変わった場合に、ユーザーを変更するためにも使用できます。




始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーを作成した時点では、デフォルト権限のみが割り当てられています。ユーザー・アカウントに追加の権限が必要である場合は、最初のユーザー作成後に、これらの権限を手動で追加する必要があります。ユーザー・アカウントが自己登録機能を使用して作成された場合は、ユーザー情報のサブセットのみが使用可能になります。残りの情報は、アプライアンス管理権限が割り当てられているユーザーが追加する必要があります。WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェースを使用してユーザーを変更するには、以下の手順に従ってください。

手順

1. 「ユーザー」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「集合」 > 「ユーザー」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「ユーザーの作成」リンクをクリックします。
2. 変更するユーザーの「ユーザー名」をクリックします。ユーザーの作成後は、表示名およびユーザー名を変更できません。
3. ユーザーのパスワードおよび E メール・アドレスを編集することができます。パスワードを変更するには、フィールドの【編集】をクリックします。新規パスワードを入力し、パスワードを変更します。
4. ユーザー・アクティビティを表示します。以下のユーザーの活動状況をユーザー・アカウント画面に表示できます。
 - **現在の状況:** このフィールドには、ユーザーの状況が示されます。以下のリストには、起こり得るユーザー状況が含まれています。
 - : 最近 5 分間はアクティブ
 - : 5 分を超えて非アクティブ
 - : ユーザーがまだログインしていません
 - **ユーザー・グループ:** このフィールドには、ユーザーがメンバーになっているすべてのユーザー・グループがリストされます。

グループ名を入力し、ユーザーをグループに追加します。ユーザー・グループ名を入力すると、入力内容に一致するユーザー・グループのリストが表示されます。ユーザーをグループに追加するには、ユーザー・グループ名をクリックしてください。ユーザー・グループ名を入力するだけでは、ユーザーはグループに追加されません。ユーザーをユーザー・グループに追加すると、ユーザー・グループの権限がユーザーに割り当てられます。ユーザーに割り当てられていた以前の権限レベルは保持されません。ユーザーをユーザー・グループから除去する場合は、ユーザーを除去するグループの隣にある【除去】リンクをクリックします。ユーザーは、(Everyone グループ以外の) すべてのグループから除去されると、最後に除去されたグループに割り当てられていた権限を保持します。

5. このユーザーの権限を変更します。ユーザーに割り当てられるアクセス・レベルを制御するために、権限を選択またはクリアできます。ユーザーがグループ (Everyone グループは含めません) のメンバーの場合は、ユーザー権限は変更できません。ユーザーがグループのメンバーの場合、そのグループで定義された権限をユーザーは持ちます。ユーザーが複数グループのメンバーの場合、それらのグループで定義された権限の合計をユーザーは持ちます。グループに定義されている権限を変更すると、その変更は、そのグループのすべてのメンバーに伝搬します。ユーザーごとに選択可能な権限は、以下のとおりです。
 - アプライアンス管理
 - アプライアンス・モニター
 - データ・グリッドの作成これらの権限設定について詳しくは、『[ユーザー許可](#)』を参照してください。

タスクの結果

ユーザー・アカウントは正常に変更されています。

次のタスク

ユーザーの変更後、このユーザーをユーザー・グループに追加できます。ユーザー・グループの作成方法、およびユーザーをユーザー・グループに追加する方法については、『[ユーザー・グループの作成](#)』および『[ユーザー・グループの管理](#)』を参照してください。Lightweight Directory Access Protocol (LDAP) サーバーを認証用を使用して、セキュリティーのレイヤーをアプライアンスに追加できます。LDAP サーバーによるアプライアンスの保護について詳しくは、

『[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)』を参照してください。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザーの除去

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ただし、特定のユーザーが不要になった場合は、WebSphere® DataPower® XC10 アプライアンス からそのユーザーを除去できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーを削除すると、そのユーザーが所有していたすべてのリソースが、自動的に実行者に転送されます。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用してユーザー・アカウントをアプライアンスから除去するには、以下の手順に従ってください。

手順

1. 「ユーザー」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「集合」 > 「ユーザー」にナビゲートします。
 - 「ホーム」ページで、「ステップ 1: アプライアンスのセットアップ」セクションにある「ユーザーの作成」リンクをクリックします。
2. ユーザーの **<user_name>** をクリックして、削除するユーザーを選択します。

注: 「管理者」ユーザー・アカウント xadmin は削除できません。
3. 「削除」アイコン (✖) をクリックして、ユーザーの削除を開始します。メッセージ・ボックスが表示され、このユーザーが完全に削除されることに対する確認が要求されます。
4. 「OK」をクリックします。

タスクの結果

これで、アプライアンスからユーザー・アカウントが除去されました。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

新規ユーザー・アカウントの登録

管理者が「新規ユーザーに自分のアカウントの作成を許可する」オプションを使用可能にしている場合、自分自身のユーザー・アカウントを作成できます。このタスクを使用して、ユーザー・アカウントをログイン画面から作成します。

始める前に

新規ユーザー・アカウントを登録するには、「新規ユーザーに自分のアカウントの作成を許可する」フィールドが「使用可能」に設定されている必要があります。ユーザー登録を使用可能にする方法については、『[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)』を参照してください。

手順

1. WebSphere DataPower XC10 アプライアンス のログイン画面にアクセスします。
2. 「登録...」 ボタンをクリックして、ユーザー・アカウントの作成を開始します。
3. 「ユーザー名」フィールドにログイン ID を入力します。このフィールドに入力された値は、ユーザー名およびこのユーザーの表示名の両方に使用されます。このフィールドの値は 64 文字まで指定可能です。すべての英数字を使用できます。特殊文字 !@#%^*-&+= も使用可能です。

重要: Lightweight Directory Access Protocol (LDAP) を使用してユーザーを認証する場合は、登録されるユーザーは LDAP リポジトリに存在していなければなりません。LDAP 構成の作成については、『[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)』を参照してください。

4. 「パスワード」フィールドにユーザーのパスワードを入力します。パスワードは、「ユーザー名」フィールドに使用できる文字と同じ文字を使用できます。Simple Mail Transfer Protocol (SMTP) または LDAP サーバーが定義されていない場合は、「パスワード」フィールドは必須です。SMTP が使用可能であれば、パスワードを入力するか、あるいは、そのフィールドを空白のままにして生成されたパスワードを E メール・アドレスに送信するようにするかを選択できます。LDAP を使用してユーザーを認証する場合、既存の LDAP パスワードが使用され、パスワードを入力する必要はありません。
5. 「パスワードの確認」フィールドに、上と同じユーザーのパスワードを再入力します。このフィールドに入力する値は、「パスワード」フィールドに入力した値と同じでなければなりません。これらのフィールドの値が一致しない場合、「登録」をクリックするとエラーが表示されます。ユーザーを作成するためには、このエラーを解決する必要があります。
6. 「E メール・アドレス」フィールドに E メール・アドレスを入力します。Simple Mail Transfer Protocol (SMTP) サーバーを使用するときに E メール・アドレスが必要です。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)を参照してください。
7. 「登録」ボタンをクリックして、登録プロセスを完了します。

タスクの結果

上記の手順が正常に完了すると、ユーザー・インターフェースへのログインに使用できるユーザー・アカウントが登録されます。デフォルトでは、アプライアンス・モニター権限のみが割り当てられます。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザー・グループの作成

特定の WebSphere® DataPower® XC10 アプライアンス リソースに対するユーザーの許可の管理を容易にするために、ユーザー・グループを作成できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザー・グループを作成して、ユーザーのコレクション (集合) の許可をリソースまたはリソースのグループに迅速に割り当てます。ユーザー・グループは、最初に作成した時点では空です。そのため、ユーザーを新規ユーザー・グループにそれぞれ手動で追加する必要があります。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用してユーザーを作成するには、以下の手順に従ってください。

手順

1. 「集合」 > 「ユーザー・グループ」にナビゲートします。
2. 追加アイコン (+) をクリックして、グループを作成します。
3. 「グループ名」フィールドに名前を入力します。このフィールドの値は 64 文字まで指定可能です。このフィールドを空白にすることはできません。すべての英数字と、特殊文字 !@#%^*-&-+= を使用できます。LDAP 認証が構成されている場合、LDAP 内に存在するグループのみを集合に追加することができます。
4. 「説明」フィールドにその他の追加情報を入力します。このフィールドは、ユーザー・グループに関する詳細を付加するために使用できます。
5. 「OK」をクリックします。

タスクの結果

上記の手順が正常に完了すると、WebSphere DataPower XC10 アプライアンス ユーザーの許可の管理に役立つ 新規ユーザー・グループが作成されます。

次のタスク

作成したグループにユーザーを追加できます。ユーザーをユーザー・グループに追加する手順については、『[ユーザー・グループの管理](#)』を参照してください。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザー・グループの管理

ユーザー・グループを作成した当初は、このユーザー・グループにはメンバーとして指定されたユーザーは存在しません。LDAP 認証が使用可能でない場合、手動でユーザーをグループに追加する必要があります。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェースを使用して、ユーザーをユーザー・グループに追加するか、またはユーザー・グループから除去するには、以下の手順に従ってください。

手順

1. 「集合」 > 「ユーザー・グループ」にナビゲートします。
2. 「<group_name>」をクリックして、変更するグループを選択します。
3. ユーザー・グループの説明を変更したい場合は、既存の説明をクリックして、変更内容を入力します。
4. ユーザーをグループに追加する場合は、追加するユーザーを入力して、その「<user_name>」をクリックします。

ユーザー名を入力すると、入力内容に一致するユーザーのリストが表示されます。ユーザーをグループに追加するには、ユーザー名をクリックしてください。ユーザー名を入力するだけでは、ユーザーはグループに追加されません。ユーザーをユーザー・グループに追加すると、ユーザー・グループの権限がユーザーに割り当てられます。ユーザーに割り当てられていた以前の権限レベルは保持されません。

注: LDAP 認証が使用可能な場合、WebSphere DataPower XC10 アプライアンス 内のグループのメンバーシップを変更することはできません。

5. グループに割り当てられた権限を変更します。

ユーザー・グループに適用可能な権限は以下のとおりです。

- アプライアンス管理
- アプライアンス・モニター
- データ・キャッシュの作成

これらの権限設定について詳しくは、『[ユーザー許可](#)』を参照してください。

6. ユーザーをグループから除去する場合は、除去するユーザーの隣にある【除去】リンクをクリックします。ユーザーを除去するときに確認は要求されません。したがって、ユーザー・グループを管理するときは、十分な注意が必要です。ユーザーは、Everyone グループ以外のすべてのグループから除去されると、最後に除去されたグループに割り当てられていた権限を保持します。

タスクの結果

これで、ユーザー・グループの変更が完了しました。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの除去](#)

ユーザー・グループの除去

不要になったユーザー・グループは WebSphere® DataPower® XC10 アプライアンス から除去できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

この手順を実行すると、WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用して、ユーザー・グループをアプライアンスから除去できます。

手順

1. 「集合」 > 「ユーザー・グループ」をクリックします。
2. **<user_group_name>**をクリックして、除去するユーザー・グループを選択します。
注: 「全員」ユーザー・グループは除去できません。
3. 「除去」アイコン (✖) をクリックして、グループの除去を開始します。
4. 選択されたユーザー・グループを除去することを確認して、「**OK**」をクリックします。

タスクの結果

各ユーザー・グループ・メンバーがグループから除去され、ユーザー・グループが削除されます。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

ユーザー許可

ユーザー許可は、各ユーザーに向けて表示できるパネルを決定したり、特定オブジェクトに対するユーザー・アクセスを決定したりするために定義します。

ユーザーに割り当てた許可によって、そのユーザーが実行可能な WebSphere® DataPower® XC10 アプライアンス管理用タスクが定義されます。表示される管理ページが決定され、さらに「ようこそ」ページの内容が、ユーザーに割り当てたアクセス・レベルに応じて表示されるように、動的に生成されます。ユーザーが最初に登録する時に、アプライアンスのモニター権限が与えられます。すべてのアプライアンス管理者は、データ・グリッド (data grid) の作成権限またはアプライアンス管理権限を割り当てる必要があります。

表 1. 各許可レベルの表示可能パネル

	アプライアンス・モニター権限	データ・グリッド作成権限	アプライアンス管理権限
ウェルカム・ページ	はい	はい	はい
データ・グリッドの作成	いいえ	はい	はい
データ・グリッドの削除	いいえ	はい	はい
モニター・メニューの表示	はい	いいえ	はい
タスクの表示	はい	はい	はい
2.5+ 完了したタスクの削除	いいえ	いいえ	はい
集合およびゾーンの表示と作成	いいえ	いいえ	はい
アプライアンスの設定の変更	いいえ	いいえ	はい
ユーザーおよびグループの変更	いいえ	いいえ	はい
2.5+ データ管理の表示	はい	いいえ	はい
2.5+ メッセージ・センターの表示	はい	いいえ	はい

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)
[xcadminパスワード](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理

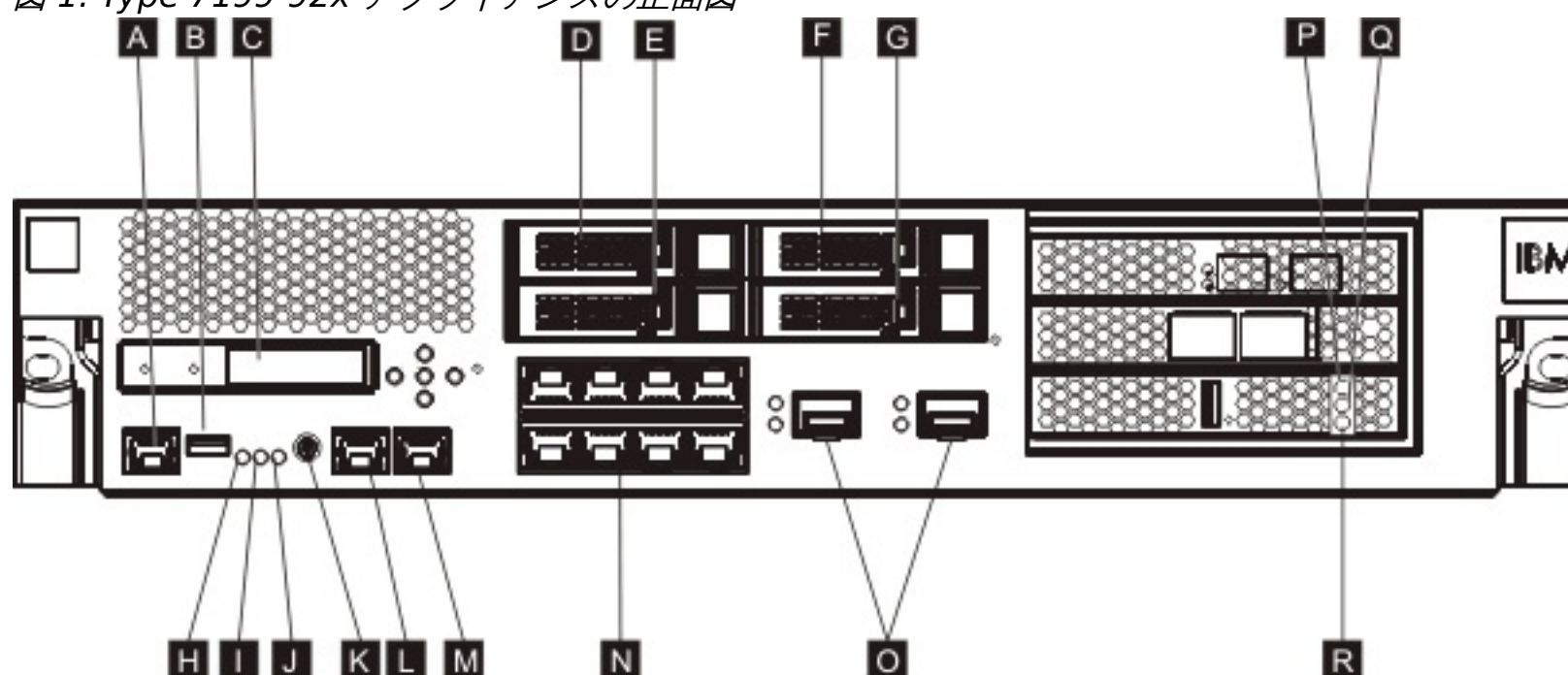
初回のシリアル接続時に、アプライアンスをネットワークに接続するために *mgmt* イーサネット・インターフェースを構成しました。追加のプライベート・イーサネット・ポートをユーザー・インターフェースで定義することができます。

始める前に

アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

図 1. Type 7199-92x アプライアンスの正面図



- mgmt0 (L)
- mgmt1 (M)
- 左方 1 ギガビット・イーサネット・モジュール eth0-eth7 (N)
- 右方 10 ギガビット・イーサネット・モジュール eth8-eth9 (O)

Type 7199-92x アプライアンスでは、データ・グリッドに 1 ギガビット・イーサネット・ポートまたは 10 ギガビット・イーサネット・ポートのいずれかを使用します。使用するポートは、すべて 1 ギガビット・ポートであるか、すべて 10 ギガビット・ポートでなければなりません。初期構成完了後は、ポート・タイプを変更できません。MGMT0 (L) の管理ポートに接続します。

スタンドアロンのアプライアンスのイーサネット・インターフェースを編集する場合は、設定の変更後、構成をクリアし、アプライアンスを再始動する必要があります。アプライアンスが集合内にある場合、イーサネット・インターフェースは更新できません。

手順

1. スタンドアロンのアプライアンスのイーサネット・インターフェースを編集します。「設定」パネルにナビゲートします。
 - WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「イーサネット・インターフェース」を展開します。
3. 「使用可能」チェック・ボックスを選択またはクリアして、イーサネット・インターフェースを使用可能または使用不可にします。 *mgmt* インターフェースは使用不可にできません。
4. 「IP アドレス/マスク」を変更します。 <ip_address>/<subnet_mask> という形式で IP アドレスと サブネット・マスクを入力します。サブネット・マスクは、クラスレス・ドメイン間ルーティング (CIDR) 表記を使用して入力する必要があります。例: 長い表記での 255.255.255.0 は CIDR 表記では 24 です。
5. 「デフォルト・ゲートウェイ」を変更します。アプライアンスは、宛先ベースのルーティングではなく、送信元ベースのルーティングを使用します。パケットは、パケットが受信されたものと同じインターフェース上の宛先へ送信されます。各インターフェースには、他のインターフェースから分離された、専用のルーティング・テーブルがあります。ローカル・サブネット経由で宛先に到達する必要があるインターフェースごとに、そのインターフェースから直接到達可能なデフォルトの経路を指定してください。
6. 指定した IP アドレスがプライベート IP アドレスであるかどうかを指定します。
7. 「最大伝送単位 (MTU)」を変更します。このフィールドには、イーサネット・インターフェースを使用した通信時に、プロトコル・データ単位で可能な最大サイズ (バイト単位) を指定します。デフォルト値は 1500 バイトで、これは、このフィールドの最大許容値でもあります。

8. 「モード」を変更します。イーサネット・インターフェースでは以下のイーサネット・モードを使用できます。
 - 自動
 - 10baseT-HD
 - 10baseT-FD
 - 100baseTx-HD
 - 100baseTx-FD
 - 1000baseTx-FD
9. チェック・ボックス「使用可能」を選択することで、構成をクリアし、アプライアンスを再始動します。このアクションによって構成がクリアされます。WebSphere DataPower XC10 アプライアンス プロセスがイーサネット・インターフェースにバインドするためには、再始動が必要です。

次のタスク

イーサネット・インターフェースの状況をモニターするには、[シリアル接続を使用したイーサネット・インターフェースの状況詳細のモニター](#)を参照してください。

[集約インターフェースの追加](#)

WebSphere DataPower XC10 アプライアンス で集約インターフェースを構成することができます。

[集約インターフェースの編集](#)

集約インターフェースのメンバーの追加と削除、および IP アドレス、デフォルト・ゲートウェイ、最大伝送単位 (MTU) などのプロパティの変更を行えます。集約ポリシーを変更することもできます。

[集約インターフェースの削除](#)

集約インターフェースを削除することができます。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集約インターフェースの追加

WebSphere® DataPower® XC10 アプライアンス で集約インターフェースを構成することができます。

始める前に

以下のステップを実行するには、WebSphere DataPower XC10 アプライアンス 管理権限が割り当てられている必要があります。

このタスクについて

複数のネットワーク・インターフェースを 1 つの集約インターフェースにリンクできます。1 つの集約インターフェースが 1 つ以上のイーサネット・インターフェースで構成され、単一の論理装置として機能します。この機能を使用して、ネットワーク上でトラフィックを分散することができます。集約インターフェースの作成は、コマンド行インターフェースまたは Web コンソールを使用して行うことができます。集約インターフェースの推奨される作成方法は、コマンド行インターフェースを使用する方法です。いずれの場合も、アプライアンスにログインしてから行う必要があります。操作が完了したら、**clear-all** コマンドを実行し、アプライアンスを再始動します。

スイッチ間のリンク集約により、接続性、冗長性が増加し、処理能力が拡大します。

- 処理能力の増大

複数のイーサネット・インターフェースが、集約インターフェースとして、1 つの論理リンクに結合されて、これにより、アプライアンスが使用できる処理能力が増大します。

要確認: アプライアンスでは、1 Gbps ポートまたは 10 Gbps ポートのいずれかを使用可能に設定できます。すなわち、10 Gbps イーサネット・ポートを使用可能に設定する場合、10 Gbps ポートを集約することもできます。

- 自動フェイルオーバー

集合内で障害が起こったイーサネット・ポートのトラフィックは、作動しているイーサネット・ポートに転送されます。単一のスイッチを使用すると、これによりポートの障害を防ぐことができます。また、予備のスイッチを使用することで、ネットワーク・スイッチの障害に対して防御することも可能です。

- ロード・バランシング

ロード・バランシング・ポリシーを選択して、インバウンド・トラフィックとアウトバウンド・トラフィックの両方を分散することができます。

- 冗長性のサポート

2 つのシステムを、さまざまな集約構成の複数のリンクによって接続することができます。

- 管理の改善

管理者は、複数のイーサネット・インターフェースを単一の集約インターフェース・ユニットとして管理できます。

手順

- コマンド行インターフェースを使用して集約インターフェースを追加する場合。次の例を使用して集約インターフェースを作成します。

```
Console> create aggregate-interface myAgg
Console aggregate-interface:myAgg> member eth3 eth4
Console aggregate-interface:myAgg> ip
CWZBR02205I: Entering "ip" mode
Console aggregate-interface:myAgg ip> address 10.5.5.5/24
Console aggregate-interface:myAgg ip> exit
Console aggregate-interface:myAgg> primary-member eth3
Console aggregate-interface:myAgg> aggregation-policy balance-tlb
Console aggregate-interface:myAgg> show
aggregate-interface myAgg:
name "myAgg"
AdminState "Enabled"
use-arp "true"
mtu "1500"
ip
  use-dhcp "false"
  address "10.5.5.5/24"
  use-slaac "false"
```

```
dad-transmits "1"
dad-retransmit-timer "1000"
end ip
aggregation-policy "balance-tlb"
lacp-selection-logic "stable"
transmit-hash-policy "layer2"
member "eth3" "eth4"
primary-member "eth3"

Console aggregate-interface:myAgg> exit
```

- Web コンソールを使用して集約インターフェースを追加する場合。次のいずれかの方法を使用して「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
1. 「集約インターフェース」を展開します。
 2. 「新規集約インターフェースの追加」をクリックします。
 3. 追加する集約インターフェースを記述するためのフォームに入力します。

名前

インターフェースの名前を指定します。

IP アドレス/マスク

集約インターフェースに割り当てる IP アドレスを指定します。 <ip_address>/<subnet_mask> という形式で IP アドレスと サブネット・マスクを入力します。

ヒント: IPv4 アドレスと IPv6 アドレスは、いずれもこのプロパティを使用して割り当てられます。アドレスの重複は許可されません。

集約ポリシー

アクティブ・バックアップ

高可用性を求める場合は、アクティブ・バックアップ・ポリシーを指定します。このポリシーを使用すると、集約インターフェースの 1 メンバーとして同時に使用可能になるのは、1 つのイーサネット・インターフェースのみです。そのイーサネット・インターフェースに障害が発生した場合、別のメンバーが処理を再開します。

LACP

高可用性と処理能力の両方を求める場合は、Link Aggregation Control Protocol (LACP) ポリシーを指定します。LACP ポリシーは、そのモードが「OFF」に設定されていない場合にのみ使用できます。デフォルトで、選択ロジックは「安定」で、送信ハッシュ・モードは「layer2」です。

balance-tlb

ロード・バランシングと高可用性を求める場合は、balance-tlb ポリシーを指定します。このポリシーは、各メンバーの現在の負荷に応じてアウトバウンド・トラフィックを分散します。インバウンド・トラフィックは、基本メンバーとして選択されているイーサネット・インターフェースに送信されます。受信中のイーサネット・インターフェースに障害が発生した場合、別のメンバーが処理を再開します。

メンバー

集約インターフェースに追加するイーサネット・インターフェースを指定します。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

制約事項: イーサネット・インターフェースは、複数の集約インターフェースのメンバーになることはできません。

重要: イーサネット・インターフェースを集約インターフェースに追加するには、そのインターフェースが必ず使用不可の状態になっていなければなりません。イーサネット・インターフェースを使用不可にするには、「イーサネット・インターフェース」パネル上の「使用可能」チェック・ボックスのチェックを外す必要があります。このパネルへのナビゲートの方法については、[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)を参照してください。

4. 「OK」をクリックします。

重要: 集約インターフェースは、デフォルトで使用可能になっています。新規集約インターフェースを使用せずに、既存の集約インターフェースの使用を継続する場合、まず新規のインターフェースを無効にする必要があります。集約インターフェースを無効にするには、集約インターフェースを展開して、追加したばかりの新規集約インターフェースを見つけます。「使用可能」チェック・ボックスのチェックを外します。新規集約インターフェースを使用可能のままにすると、次に WebSphere DataPower XC10 アプライアンスを再始動する際に、その新規集約インターフェースがデータ・グリッドのバインドに使用されるため、予

期しない結果につながるおそれがあります。

注: 集約インターフェースを変更した場合は常に、WebSphere DataPower XC10 アプライアンス が変更内容を使用する前に再始動が必要です。しかし、変更を有効にするために clear-all コマンドを発行し、次に device-restart コマンドを発行する必要があります。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集約インターフェースの編集

集約インターフェースのメンバーの追加と削除、および IP アドレス、デフォルト・ゲートウェイ、最大伝送単位 (MTU) などのプロパティの変更を行えます。集約ポリシーを変更することもできます。

このタスクについて

集約インターフェースの編集は、コマンド行インターフェースまたは Web コンソールを使用して行うことができます。集約インターフェースの推奨される編集方法は、コマンド行インターフェースを使用する方法です。いずれの場合も、アプライアンスにログインしてから行う必要があります。操作が完了したら、**clear-all** コマンドを実行し、アプライアンスを再始動します。

手順

- コマンド行インターフェースを使用して集約インターフェースを編集する場合。次の例を使用して集約インターフェースを編集します。

```
Console aggregate-interface:myAgg> aggregation-policy lacp
Console aggregate-interface:myAgg> transmit-hash-policy layer3+4
Console aggregate-interface:myAgg> exit
Console> show aggregate-interface myAgg
aggregate-interface myAgg: [Up]

name "myAgg"
AdminState "Enabled"
use-arp "true"
mtu "1500"
ip
  use-dhcp "false"
  address "10.5.5.5/24"
  use-slaac "false"
  dad-transmits "1"
  dad-retransmit-timer "1000"
end ip
aggregation-policy "lacp"
lacp-selection-logic "stable"
transmit-hash-policy "layer3+4"
member "eth3" "eth4"
primary-member "eth3"
```

- Web コンソールを使用して集約インターフェースを編集する場合。次のいずれかの方法を使用して「設定」パネルにナビゲートします。
 - WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」>「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。

1. 「集約インターフェース」を展開します。
2. 既存の集約インターフェースのメンバーを追加したり、削除したりできます。この集約のメンバーであるイーサネット・インターフェースのリストが表示されます。

ヒント: メンバーが 1 つのみの場合は削除できません。また、基本メンバーも削除することはできません。

重要: イーサネット・インターフェースを集約インターフェースに追加するには、そのインターフェースが必ず使用不可の状態になっていなければなりません。イーサネット・インターフェースを使用可能もしくは使用不可に設定する方法について詳しくは、[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#) を参照してください。

- メンバーを追加するには、「さらに追加...」をクリックしてください。
 - イーサネット・インターフェースを選択して、「OK」をクリックします。
3. `<ip_address>/<subnet_mask>` という形式で IP アドレスとサブネット・マスクを変更します。サブネット・マスクは、クラスレス・ドメイン間ルーティング (CIDR) 表記を使用して入力する必要があります。例えば、長い表記での `255.255.255.0` は、CIDR 表記では `24` です。
 4. 「デフォルト・ゲートウェイ」を変更します。アプライアンスは、宛先ベースのルーティングではなく、送信元ベースのルーティングを使用します。パケットは、パケットが受信されたものと同じインターフェース上の宛先へ送信されます。各インターフェースには、他のインターフェースから分離された、専用のルーティング・テーブルがあります。ローカル・サブネット経由で宛先に到達する必要があるインターフェースごとに、そのインターフェースから直接到達可能なデフォルトの経路を指定してください。

5. 「**最大伝送単位 (MTU)**」を変更します。このフィールドには、イーサネット・インターフェースを使用した通信時に、プロトコル・データ単位で可能な最大サイズ (バイト単位) を指定します。デフォルト値は 1500 バイトで、これは、このフィールドの最大許容値でもあります。
6. 「**編集**」をクリックして集約ポリシーを変更します。変更する集約ポリシーに応じて、以下のプロパティが表示されます。

集約ポリシー

アクティブ・バックアップ

高可用性を求める場合は、アクティブ・バックアップ・ポリシーを指定します。このポリシーを使用すると、一度に使用可能になるのは、(集約インターフェースの 1 メンバーである) 1 つのイーサネット・インターフェースのみです。そのイーサネット・インターフェースに障害が発生した場合、別のメンバーが引き継ぎます。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

基本メンバー

基本メンバーとして指定するイーサネット・インターフェースを特定します。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

集約ポリシー

LACP

高可用性と処理能力の両方を求める場合は、Link Aggregation Control Protocol (LACP) ポリシーを指定します。LACP ポリシーは、そのモードが「OFF」に設定されていない場合にのみ使用できます。デフォルトでは、選択ロジックは「安定」で、送信ハッシュ・モードは「layer2」です。

選択ロジック

- 安定

最大の処理能力を持つイーサネット・インターフェースを指定します。安定設定を選択すると、有効な集約インターフェースに使用可能なメンバーがないときに、このイーサネット・インターフェースが再度選択されます。安定設定はデフォルトです。

- 処理能力

最大の処理能力を持つイーサネット・インターフェースを指定します。別のメンバーが追加/削除されるか、あるいは、集約インターフェースが使用可能/使用不可に設定されると、このイーサネット・インターフェースが再度選択されます。

- Count

メンバーとしてのイーサネット・インターフェース数が最も多い集約インターフェースを指定します。

ハッシュ・ポリシー

- layer2

ハッシュを生成する場合は、MAC アドレスの排他 OR (XOR) 結果を指定します。

- layer2+3

ハッシュを生成する場合は、MAC アドレスと IP アドレスの XOR 結果を指定します。

- layer3+4

ハッシュを生成する場合は、IP アドレスとポート番号の XOR 結果を指定します。

集約ポリシー

balance-tlb

ロード・バランシングと高可用性を求める場合は、balance-tlb ポリシーを指定します。このポリシーは、各メンバーの現在の負荷に応じてアウトバウンド・トラフィックを分散します。インバウンド・トラフィックは、基本メンバーとして選択されているイーサネット・インターフェースに常に送信されます。受信中のイーサネット・インターフェースに障害が発生した場合、別のメンバーが引き継ぎます。

基本メンバー

基本メンバーとして指定するイーサネット・インターフェースを特定します。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

7. 「**OK**」をクリックします。

重要: 集約インターフェースを編集する際、アプライアンスを再始動するまで変更は有効になりません。WebSphere DataPower XC10 アプライアンス プロセスがイーサネット・インターフェースにバインドするためには、再始動が必要です。変更した集約インターフェースを使用不可に設定してから、WebSphere DataPower XC10 アプライアンス を再始動する必要があります。インターフェースを使用不可に設定す

るには、「集約インターフェース」を展開して、変更したばかりの集約インターフェースを見つけます。
「使用可能」チェック・ボックスのチェックを外します。集約インターフェースを使用不可にした後、コマンド行インターフェースで clear-all コマンドと device-restart コマンドを実行する必要があります。これらのコマンドを実行しないと、変更が有効になりません。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

関連タスク:

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集約インターフェースの削除

集約インターフェースを削除することができます。

このタスクについて

集約インターフェースを削除は、コマンド行インターフェースまたは Web コンソールを使用して行うことができます。集約インターフェースの推奨される削除方法は、コマンド行インターフェースを使用する方法です。いずれの場合も、アプライアンスにログインしてから行う必要があります。操作が完了したら、**clear-all** コマンドを実行し、アプライアンスを再始動します。

手順

- コマンド行インターフェースを使用して集約インターフェースを削除する場合。次の例を使用して集約インターフェースを削除します。

```
Console> delete aggregate-interface myAgg
Console> list aggregate-interface
```

- Web コンソールを使用して集約インターフェースを削除する場合。次のいずれかの方法を使用して「設定」パネルにナビゲートします。
 - WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。

1. 集約インターフェースを除去するには、「除去」アイコン (✖) をクリックします。

重要: 集約インターフェースを削除する場合、アプライアンスを再始動するまで変更は有効になりません。集約インターフェースを削除した後、コマンド行インターフェースで **clear-all** コマンドと **device-restart** コマンドを実行する必要があります。これらのコマンドを実行しないと、WebSphere DataPower XC10 アプライアンス は、このインターフェースの使用を続行します。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

構成のインポートとエクスポート

新規アプライアンスを構成する際、そのアプライアンスの構成設定をエクスポートして保管することができます。後に、アプライアンスの除去や再インストールを必要とする変更が発生した場合に、構成設定を損失することなく、保管しておいた構成データをインポートすることができます。

始める前に

構成をインポートあるいはエクスポートするには、まず、データ・グリッド、ユーザー/グループ、Lightweight Directory Access Protocol (LDAP) 情報を含めたアプライアンス環境を構成する必要があります。詳しくは、[アプライアンスの構成](#)を参照してください。

このタスクについて

管理者としてアプライアンスをセットアップするとき、データ・グリッドのユーザーおよびグループと、LDAP 構成設定の作成が要求されます。このセットアップを完了したら、これらの構成をファイルにエクスポートすることで、この情報を保存することができます。そうすると、そのエクスポート・ファイルを後で使用して、エクスポート済み構成情報をインポートしてアプライアンスに戻すことができます。以下の構成情報をインポートおよびエクスポートすることができます。

- データ・グリッド構成
- ユーザー構成
- グループ構成
- LDAP 構成

集合からのアプライアンス削除か、あるいは、スタンドアロン環境でのアプライアンスの再インストールが必要とされるエラーが発生した場合に、構成データのインポートおよびエクスポートが必要になることがあります。例えば、集合に追加するアプライアンスから構成情報をエクスポートすると、インポート機能を使用して、集合を結合する前のそのアプライアンスの構成にリカバリーすることができます。スタンドアロン環境もしくは集合環境のいずれでも、インポート・アクションおよびエクスポート・アクションを使用すると、削除したアプライアンスの構成情報を手動で再作成する時間を省けます。

スタンドアロン環境の場合: スタンドアロン・アプライアンスを使用している場合、ステップ 1 と 4 を実行してください。これには、構成のインポートおよびエクスポート用の次の **config** コマンド構文が含まれます。

```
config <import|export> -file <filename> [-silent]
```

config export コマンドは、値が 0 のパラメーターも受け入れます。このコマンドは、-file フラグを使用して指定できないアーカイブされたエクスポートを実行するからです。

コマンドの使用法について参照するには、**config usage** と入力してください。

手順

1. 構成情報を、後で使用できるようにファイルにエクスポートします。例えば、ファイル `foo.json` にエクスポートします。コマンド行インターフェースを使用し、指定可能なパラメーターのリストを後ろに指定して **config** コマンドを実行します。

```
config <export> -file foo.json -silent
```

前の例での **-silent** パラメーターはオプションです。**-silent** フラグを指定すると構成状況メッセージが画面に表示されなくなります。これらのメッセージは、デフォルトで表示されることになっています。

2. オプション: 新規アプライアンスを集合に追加します。
3. 構成データを破損するようなエラーが発生したら、アプライアンスを集合から削除します。ログ分析ツールを使用して、ランタイム環境のパフォーマンスを分析したり、お使いのアプライアンス環境内で発生した問題を解決したりできます。
4. 最初のステップで作成した `foo.json` ファイルをインポートします。コマンド行インターフェースを使用し、指定可能なパラメーターのリストを後ろに指定して **config** コマンドを実行します。

```
config <import> -file foo.json -silent
```

前の例での **-silent** パラメーターはオプションです。**-silent** フラグを指定すると構成状況メッセージが画面に表示されなくなります。これらのメッセージは、デフォルトで表示されることになっています。

5. 原型としてそのアプライアンスを含めた新規集合を作成します。
6. その新規集合に追加する形で他のアプライアンスを一度に一つずつ、戻します。

2.5+ [インポートされた構成の管理と構成のエクスポート](#)

いつ WebSphere DataPower XC10 アプライアンス から構成をエクスポートするかを指定することができます

す。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

インポートされた構成の管理と構成のエクスポート

いつ WebSphere® DataPower® XC10 アプライアンス から構成をエクスポートするかを指定することができます。

このタスクについて

構成のエクスポートは、月に 1 回、週に 1 回、または日に 1 回の割合で行えます。また、エクスポートを実行する時刻を指定することもできます。例えば、データ・グリッド構成のエクスポートを月の最初の日に行い、エクスポートした構成をヒストリーに 1 年間保存するように指定することができます。

エクスポートされた構成は最大 10 個保管できます。これには、手動で作成したエクスポート、および Web コンソールを使って作成したエクスポートも含まれます。

手順

1. Web コンソールで、「アプライアンス」 > 「構成のインポートおよびエクスポート」をクリックします。
2. 「エクスポート」セクションを展開します。
 - a. 「エクスポートのスケジュール」を選択します。
 - b. エクスポートを開始する日時を指定します。
 - c. このスケジュールをどのくらいの頻度で続けるかを指定します。
3. 「スケジュールリングの更新」をクリックします。
4. オプション: スケジュールに入れられたエクスポートとエクスポートの間で構成をエクスポートしたい場合は、手動で構成をエクスポートします。手動で構成をエクスポートするには、「今すぐエクスポート」をクリックします。

親トピック: [構成のインポートとエクスポート](#)

ドメイン・ネーム・システム (DNS) サーバーの管理

通信にドメイン・ネーム・システム (DNS) ルックアップ・サービスが使用されるため、IBM® WebSphere® DataPower® XC10 アプライアンスには DNS サーバーが必須です。アプライアンスの初期化時に、DNS サーバーを指定する必要があります。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

DNS サーバーは、アプライアンスが正しく機能するために必要です。ご使用の DNS サーバーには、WebSphere DataPower XC10 アプライアンスで管理されている IP アドレス範囲の、順方向および逆方向の DNS 項目がある必要があります。WebSphere DataPower XC10 アプライアンスは、仮想システムのデプロイ中に、リバース・ルックアップによって取り出されたホスト名を使用します。ホスト名が定義されていないためにリバース・ルックアップが失敗した場合、デプロイメントは、IP アドレスだけでなくホスト名も必要とするため、失敗します。

手順

1. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「ドメイン・ネーム・サーバー」を展開します。
3. オプション: 既存の DNS サーバーを変更するには、「<IP アドレス>」をクリックします。
4. 新しい DNS サーバーを追加するには、「追加するにはここをクリック」を選択します。DNS サーバーは、アプライアンスの初期化中に構成されます。
5. オプション: DNS サーバーを除去するには、「除去」アイコン (✖) をクリックします。

タスクの結果

上記の手順を正常に終わると、通信時のルックアップに使用する DNS サーバーの定義は完了します。

次のタスク

コマンド行インターフェースを使用して、DNS サーバーを設定することもできます。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。

親トピック: [アプライアンスの構成](#)

関連概念:

[xcadminパスワード](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

ホスト名に対する IP アドレスのマッピング

アドレス情報を使用して TCP/IP ネットワーク・プロトコルに接続を作成するには、IP アドレスをホスト名に関連付ける必要があります。アプライアンス上で etc/hosts ファイルを編集することで、ホスト名に対して IP アドレスを解決できます。etc/hosts ファイルはユーザー・インターフェースから編集可能です。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェースから etc/hosts ファイルを編集します。

手順

1. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「ホスト名に対する IP アドレス」を展開します。
3. 「マッピングの作成」をクリックして、IP アドレスとホスト名を指定します。
4. 「OK」をクリックして、マッピングを作成します。このマッピングは、アプライアンス上の etc/hosts ファイルを編集します。
5. オプション: マッピングを除去したい場合は、除去アイコン (✕) をクリックします。

タスクの結果

上記手順を正常に終わると、IP アドレスをホスト名と関連付ける etc/hosts ファイルの編集が完了します。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

日時設定の管理

集合内のご使用の appliance 全体で同期化された日時を維持するために、Network Time Protocol (NTP) サーバーを使用します。

始める前に

以下の手順を実行するには、appliance 管理権限が割り当てられている必要があります。

このタスクについて

集合を使用する場合、NTP サーバーを構成することは重要です。NTP サーバーにより、集合内の異なる appliance 間のログ・ファイルが関連付けられ、ログ・エントリーの日時の一貫性が確実に保たれるようになります。

手順

1. 「設定」パネルにナビゲートします。日時設定を管理するには、以下のいずれかの方法で「設定」パネルまでナビゲートしてください。
 - WebSphere® DataPower® XC10 appliance ユーザー・インターフェース上部のメニュー・バーで、「appliance」>「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: appliance のセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「日時」を展開します。
3. 「現在の時間帯」ドロップダウン・メニューから正しい時間帯を選択します。
4. 新しい NTP サーバーを追加するには、「追加するにはここをクリック」をクリックします。デフォルトでは、NTP サーバーは構成されません。
5. サーバー名をクリックおよびドラッグして、各 NTP サーバーを再配列します。リスト内の最初の使用可能 NTP サーバーが、同期を維持するために使用されます。
6. NTP サーバーを削除するには、「除去」アイコン (✖) をクリックします。
7. appliance を再始動して、変更を有効にします。詳しくは、[ユーザー・インターフェースからの appliance のシャットダウンまたは再始動](#)を参照してください。

タスクの結果

ご使用の集合内の appliance 全体でクロック同期を維持するための NTP サーバーの定義は完了します。

親トピック: [appliance の構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 appliance のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[IBM WebSphere DataPower XC10 appliance の E メール送達の管理](#)

[ユーザー・インターフェースからの appliance のシャットダウンまたは再始動](#)

IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理

アプライアンスのメール送達機能は、ユーザー・パスワードのリセットに使用します。ユーザーが新規パスワードを要求すると、アプライアンスは新規パスワードを記述した E メールをユーザーに送信します。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

Simple Mail Transfer Protocol (SMTP) サーバーを、WebSphere® DataPower® XC10 アプライアンス で使用できるように構成する必要があります。メール送達機能は、ユーザーが自分のパスワードを忘れた場合に、新規パスワードを送信するために使用します。ユーザーがパスワードを忘れた場合に、SMTP サーバーが構成されていないと、そのユーザーは 新規パスワードを受け取ることができません。

手順

1. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部の メニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「メール送達」を展開します。
3. SMTP サーバーを追加します。WebSphere DataPower XC10 アプライアンス で使用する SMTP サーバーの IP アドレスまたはホスト名を指定します。このフィールドでホスト名が使用される場合、ホスト名は、アプライアンスに定義されている ドメイン・ネーム・システム (DNS) サーバーで解決可能でなければなりません。DNS サーバーの追加について詳しくは、『[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)』を参照してください。
4. 返信先アドレスを追加します。このフィールドには、管理者の E メール・アドレスを使用してください。

タスクの結果

パスワード・リセットに使用する SMTP サーバーと返信先アドレスの指定は完了します。「パスワードをお忘れですか?」リンクがアプライアンスのログイン画面に表示されるので、ユーザーはパスワードを再設定することができます。

親トピック: [アプライアンスの構成](#)

関連概念:

[xcadminパスワード](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動

WebSphere® DataPower® XC10 アプライアンス は、ユーザー・インターフェースから再始動またはシャットダウンできます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。アプライアンスのシャットダウンまたはリスタートを行う場合は、稼働しているプロセスがすべて完了していることをまず確認する必要があります。「タスク」をクリックすると、稼働中のプロセスの進行をチェックできます。

このタスクについて

WebSphere DataPower XC10 アプライアンスは、ユーザー・インターフェースからシャットダウンまたは再始動できます。

手順

1. 「設定」パネルにナビゲートします。アプライアンスへの電源を管理するには、以下のいずれかの方法で「設定」パネルまでナビゲートしてください。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「トラブルシューティング」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「電源」を展開します。
3. 「アプライアンスの再始動」をクリックして、アプライアンスをリサイクルします。再始動中に、アプライアンス上のすべてのソフトウェアが停止され、その後アプライアンスは再始動されます。アプライアンスを即座に再始動するか、またはすべてのアクティブ・タスクが完了するまで待ってから再始動するかを選択できます。
4. 「アプライアンスのシャットダウン」をクリックして、アプライアンスの電源を オフにします。シャットダウン中に、アプライアンス上のすべてのソフトウェアが停止され、アプライアンスは停止されます。アプライアンスを即座にシャットダウンするか、またはすべてのアクティブ・タスクが完了するまで待ってからシャットダウンするかを選択できます。アプライアンスへの電源を停止するには、アプライアンスの背面にある物理的な電源スイッチを使用してアプライアンスの電源を切る必要があります。

タスクの結果

2.5+ アプライアンスの始動状況をユーザー・インターフェースで追跡することができます。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。

上記の手順が正常に完了すると、選択内容に基づいてアプライアンスがシャットダウンまたは再始動されます。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集合およびゾーンの構成

集合およびゾーンは、WebSphere® DataPower® XC10 アプライアンス の物理的なトポロジーまたはロケーションの定義をサポートします。

1. [集合の作成](#)

複数の WebSphere DataPower XC10 アプライアンスを可用性、スケーラビリティ、および管理の目的でグループ化するには、集合を作成する必要があります。

2. [ゾーンの作成および変更](#)

ゾーンを使用して、アプライアンスの物理的な場所を指定できます。この物理的な場所は同じデータ・センター内の電源グリッドの 2 つの異なる エリアの場合や、おそらくは国内の別々の場所である、まったく異なるデータ・センターにあるアプライアンスの場合が考えられます。

3. [集合間のマルチマスター複製の構成](#)

マルチマスター複製は、複数のデプロイメント環境全体で連続可用性を確保するための技法です。集合内のカタログ・サーバー間に集合リンクを確立することで、2 つの異なる集合内のデータ・グリッド間で非同期的にデータを複製することができます。

4. **2.5+** [IBM eXtremeIO \(XIO\) の構成](#)

IBM® eXtremeIO (XIO) は、オブジェクト・リクエスト・ブローカー (ORB) を置き換えるトランスポート・メカニズムです。

集合の作成

複数の WebSphere® DataPower® XC10 アプライアンスを可用性、スケーラビリティ、および管理の目的でグループ化するには、集合を作成する必要があります。

始める前に

集合をセットアップする前に、集合に追加するアプライアンスのコンソールに対するアプライアンス管理権限を備える必要があります。

このタスクについて

集合を作成するには、集合にグループ化するすべてのアプライアンスのすべてのホスト名と共通鍵を、アプライアンスの 1 つの構成に追加しなければなりません。アプライアンスは、すべての構成情報と対話できます。この情報にはデータ・グリッド、モニター情報、ユーザーおよびグループ、および集合のメンバーシップが含まれます。

アプライアンスはすべて、同じファームウェア・レベルにしてからでないと集合に追加することはできません。ファームウェア・アップグレードの実行について詳しくは、[ファームウェアの更新](#)を参照してください。

手順






1. 集合の他のアプライアンスのホスト名および共通鍵を追加するアプライアンスを 1 つ選択します。他のアプライアンスのそれぞれにおいて、新しいブラウザー・ウィンドウを開いて以下のステップを完了します。
 - a. ユーザー・インターフェースで、「集合」 > 「メンバー」をクリックします。
 - b. 集合内のアプライアンスのリストが表示されます。ログインしているアプライアンスには、コンソール・アイコン () のマークが付いています。このアプライアンスの「ホスト名」と「共通鍵」をメモします。

表 1. アプライアンスの状態

アイコン	定義
	ログインしているアプライアンスを示し、そのアプライアンスが稼働中であることを示します。
	アプライアンスが開始済みで、構成および使用が可能であることを示します。
	アプライアンスが停止済みで、使用不可であることを示します。

2. 各アプライアンスのホスト名と共通鍵を、グループ内のアプライアンスの 1 つに追加します。
 - a. ユーザー・インターフェースで、「集合」 > 「メンバー」をクリックします。
 - b. 追加アイコンをクリックします ()。
 - c. 集合に追加したいアプライアンスの「ホスト名」および「共通鍵」を入力します。
3. タスク・ビューで、アプライアンスの集合への追加の進行をモニターできます。ユーザー・インターフェースで、「タスク」をクリックします。

タスクの結果

アプライアンスは、スケーラビリティおよび管理の目的で集合にグループ化されます。

親トピック: [集合およびゾーンの構成](#)

次のトピック: [ゾーンの作成および変更](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

ゾーンの作成および変更

ゾーンを使用して、アプライアンスの物理的な場所を指定できます。この物理的な場所は同じデータ・センター内の電源グリッドの2つの異なるエリアの場合や、おそらくは国内の別々の場所である、まったく異なるデータ・センターにあるアプライアンスの場合が考えられます。

始める前に

ゾーンを作成および構成する前に、ご使用のアプライアンスを集合にインストールし、構成、およびアセンブルする必要があります。その後、集合内のアプライアンスを特定のゾーンに存在するように選択します。ゾーンの作成後に追加のアプライアンスを集合に追加する場合は、アプライアンスの追加の際にゾーンを指定できます。

このタスクについて

アプライアンスのロケーションの物理的な境界を定めるためにゾーンを指定することにより、データ・グリッド内に保管されているデータを適切なロケーションに配置できます。DataPower® XC10 アプライアンスは、ゾーンを使用してキャッシュのために基本およびレプリカの情報どこに配置するかを決定します。例えば、基本データ・グリッド (data grid) がゾーン 1 にある場合、そのデータ・グリッド (data grid) のレプリカはゾーン 2 に配置されます。1つのゾーンしかない場合には、レプリカは作成されません。

初めてアプライアンスが初期化されるときに、**DefaultZone** と呼ばれるデフォルト・ゾーンに追加されます。作成したゾーンにアプライアンスを追加するときに、アプライアンスは **DefaultZone** ゾーンから除去されます。アプライアンスが **DefaultZone** ゾーンから除去された後は、そのアプライアンスを再び **DefaultZone** ゾーンに戻すことはできません。

手順

1. ユーザー・インターフェースで、「**集合**」 > 「**ゾーン**」をクリックします。
2. ゾーンを追加します。「**ゾーンの追加**」をクリックします。追加するゾーンの名前を指定します。ゾーン名がコンソール・ページに表示されます。複数のゾーンを作成する場合は、これらの手順を繰り返します。
3. ゾーンにアプライアンスを追加します。ゾーン名を展開して、ゾーンに追加するアプライアンスを選択します。集合内のすべてのアプライアンスがゾーンに追加されるまで、これらの手順を繰り返します。すべてのアプライアンスをゾーンに配置する必要があります。単一のゾーンには、異なる集合のアプライアンスを含めることはできません。
4. ゾーンの変更を適用します。ゾーンの変更内容は、「**ゾーンの変更をアクティブにする**」をクリックするまで有効になりません。ゾーンの情報を変更する場合は、削除アイコン (✖) をクリックしてゾーンを除去するか、「**ゾーンの変更をリセットする**」をクリックしてゾーンをアクティブにした最後の構成にリセットします。

タスクの結果

DataPower XC10 アプライアンスは、定義した物理ロケーション、即ちゾーンに基づいて、基本およびレプリカのデータ・キャッシュを作成できます。ユーザー・インターフェースで、「**集合**」 > 「**ゾーン**」をクリックすれば、いつでもゾーンのトポロジーの表示か変更を行うことができます。

次のタスク

ご使用のアプリケーションおよびデータ・グリッドを作成し、構成します。

親トピック: [集合およびゾーンの構成](#)

前のトピック: [集合の作成](#)

次のトピック: [集合間のマルチマスター複製の構成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

集合間のマルチマスター複製の構成

マルチマスター複製は、複数のデプロイメント環境全体で連続可用性を確保するための技法です。集合内のカタログ・サーバー間に集合リンクを確立することで、2つの異なる集合内のデータ・グリッド間で非同期的にデータを複製することができます。

始める前に

- 少なくとも2つの集合を構成する必要があります。集合間でデータを複製するには、各集合内に同一のデータ・グリッド構成を作成する必要があります。
- 定義する複数の集合トポロジーのタイプを決定します。マルチマスター・トポロジーについて詳しくは、[マルチマスター複製を実装する集合のリンクのトポロジー](#)を参照してください。
- 集合間のリンクは、ユーザー・インターフェースまたは **xscmd** ユーティリティーで構成できます。

このタスクについて

障害の誤検出が起こる可能性があるため、単一の集合の範囲は信頼できないネットワーク上には及びません。しかし、信頼できないネットワークへの接続を持つアプライアンス間で、データ・グリッドのデータを複製する必要が生じる場合もあります。このタイプのトポロジーを使用することになる一般的なシナリオには以下のようなものがあります。

- 1つの集合がアクティブでもう1つの集合がバックアップとして使用されているデータ・センター間での災害復旧。
- 地理的に近いクライアントのために、すべての集合がアクティブになっている地理的に分散されたデータ・センター。

2つの集合を接続すると、同じ名前を持つすべてのデータ・グリッドが集合間で非同期的に複製されます。これらのデータ・グリッドは、各集合内に同じ数のレプリカを持ち、同じ動的マップ構成を持っていない限りなりません。

手順

1. 集合間にリンクを確立します。
 - **ユーザー・インターフェースで集合リンクを確立する:**
 - a. ユーザー・インターフェースで、「集合」 > 「集合リンク」をクリックします。
 - b. 追加アイコン (+) をクリックし、リモート・アプライアンスのホスト名または IP アドレス、ユーザー名、およびパスワードを入力します。
 - **xscmd ユーティリティーで集合リンクを確立する:**
 - a. 各集合内のカタログ・サーバーのカタログ・サーバー IP アドレスおよびポートを取得します。ユーザー・インターフェースで、「集合」 > 「メンバー」 > 「**collective_member_name**」をクリックします。カタログ・サーバーおよびポート番号のリストが表示されます。接続する各集合についてこのステップを繰り返します。
 - b. **xscmd** ユーティリティーを集合の1つに接続します。**xscmd** ユーティリティーの開始について詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

```
xscmd.sh -ts xsatruststore.jks -tst jks -ssl -tsp xc10pass
-user xcadmin -tsp xcadmin -cep myxc10.mycompany.com:2809
[additional_xscmd_parameters]
```

- c. 現在接続している集合から、次のコマンドを実行します。

```
xscmd -c establishLink -cep myxc10.mycompany.com:2809 -fd dname
-fe myxc102.mycompany.com:2809,myxc103.mycompany.com:2809
```

-fd dname は、リンクを確立しようとしているリモート集合の名前を指定します。この例では、**dname** がリモート集合の名前です。メンバー詳細パネルでカタログ・サーバー・リストを調べれば、集合の名前が分かります。集合名はリスト内の最初の IP アドレスです。

リンクが確立されると、集合内のカタログ・サーバーが相互に複製を開始します。双方向にリンクを確立する必要はありません。

2. 集合間のリンクを除去することも可能です。
 - **ユーザー・インターフェースで集合リンクを除去する:** ユーザー・インターフェースで、除去アイコン (X) をクリックし、選択したリンクを除去します。
 - **xscmd ユーティリティーで集合リンクを除去する:** **xscmd** ユーティリティーで、次のコマンドを実行します。

```
xscmd -c dismissLink -cep myxc10.mycompany.com:2809 -fd dname
```

親トピック: [集合およびゾーンの構成](#)

前のトピック: [ゾーンの作成および変更](#)

次のトピック: **2.5+** [IBM eXtremeIO \(XIO\) の構成](#)

関連概念:

[マルチマスター複製を実装する集合のリンクのトポロジー](#)

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

カタログ・サービス・ドメインのトランスポート・タイプの表示

カタログ・サービス・ドメインで現在使用されているトランスポート・タイプを表示できます。

始める前に

スタンドアロン・カタログ・サービス・ドメイン、または WebSphere® Application Server で実行されているカタログ・サービス・ドメインで使用されているトランスポート・タイプを表示できます。

- スタンドアロン・カタログ・サービス・ドメインを使用している場合は、**xscmd** ユーティリティーを使用して、カタログ・サービス・ドメインに関するトランスポート情報を表示します。**xscmd** ユーティリティーのセットアップについて詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。
- WebSphere Application Server で実行されているカタログ・サービス・ドメインを使用している場合は、**wsadmin** ユーティリティーを使用して、トランスポート・タイプを表示できます。**wsadmin** ユーティリティーについて詳しくは、[『wsadmin スクリプトによる wsadmin スクリプト・クライアントの開始』](#)を参照してください。

手順

- スタンドアロン・カタログ・サービス・ドメインのトランスポート・タイプを表示します。**xscmd** ユーティリティーで、以下のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showTransport`
 - **Windows** `xscmd.bat -c showTransport`このコマンドにより、トランスポート・タイプが表示されます。表示される値は、「eXtremeIO」または「Object Request Broker」です。
- WebSphere Application Server で実行されているカタログ・サービス・ドメインのトランスポート・タイプを表示します。
 - 管理コンソールで、「システム管理」 > 「WebSphere eXtreme Scale」 > 「カタログ・サービス・ドメイン」 > 「**catalog_service_domain_name**」をクリックします。「**IBM eXtremeIO (XIO) 通信を有効にする**」設定が選択されているかどうかを確認します。
 - **wsadmin** ユーティリティーで、以下のコマンドを実行します。
 - Jacl を使用:

```
$AdminTask getTransport {-domainName TestDomain }
```

- Jython スtring を使用:

```
AdminTask.getTransport ('[-domainName testDomain]')
```

このコマンドにより、トランスポート・タイプが表示されます。表示される値は、「eXtremeIO」または「Object Request Broker」です。リモート・サーバーを含むカタログ・サービス・ドメインでこのコマンドを実行した場合、または `catalogServerName` がリモート・サーバーである場合は、エラーとなります。リモート・サーバーの場合は、**xscmd -c showTransport** コマンドを使用する必要があります。**wsadmin** ユーティリティーの **getTransport** コマンドについて詳しくは、[カタログ・サービス・ドメイン管理用タスク](#)を参照してください。

親トピック: [2.5+ IBM eXtremeIO \(XIO\) の構成](#)

データ・グリッドの構成

3つのタイプのデータ・グリッドが作成できます。単純データ・グリッド、セッションデータ・グリッド、およびダイナミック・キャッシュデータ・グリッドです。

[単純データ・グリッドの作成](#)

単純データ・グリッド (data grid)を使用して、オペレーションの作成、取得、更新、および削除を行うことができます。特に、単純データ・グリッド (data grid)を使用すると、データベース内に保管されたデータにより速くアクセスできるようになります。

[セッション・データ・グリッドの作成](#)

Java または .NET アプリケーションからの HTTP セッションを保管するデータ・グリッドを作成することができます。

Java [動的キャッシュの作成データ・グリッド](#)

IBM® WebSphere® DataPower® XC10 アプライアンス を使用して、WebSphere Application Server 動的キャッシュ・インスタンスのデータを保管することができます。この機能をセットアップすることにより、WebSphere Application Server 動的キャッシュ・インスタンスを使用するよう作成されたアプリケーションでアプライアンスのフィーチャーおよびパフォーマンス機能を活用できるようにします。

[データ・グリッドの最大容量の構成](#)

集合内の各データ・グリッドの最大容量を定義できます。容量限度を構成すると、特定のデータ・グリッドで使用可能なデータ・ストレージの容量が制限されます。この容量限度により、確実に、集合の使用可能なストレージ容量が予測可能な形で使用されるようになります。

[データ・グリッドのセキュリティーを使用可能に設定する](#)

データ・グリッドの作成後、データ・グリッド (data grid)のセキュリティーはデフォルトで使用不可になります。データ・グリッド (data grid)のセキュリティー設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

[データ・グリッドのクリア](#)

データ・グリッド内のすべての項目を恒久的に削除することができます。失効した情報またはテスト項目を除去するために、データ・グリッドをクリアできます。

[データ・グリッドの除去](#)

データ・グリッド (data grid)のデータを初期化する場合は、データ・グリッド (data grid)を除去してからデータ・グリッド (data grid)を再作成します。

[Spring キャッシュ・プロバイダーの構成](#)

Spring Framework バージョン 3.1 では、新しいキャッシュの抽象化が導入されています。この新しい抽象化を使用すると、既存の Spring アプリケーションに対してキャッシュを容易に追加できます。WebSphere DataPower XC10 アプライアンス をキャッシュ抽象化のキャッシュ・プロバイダーとして使用できます。

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

単純データ・グリッドの作成

単純データ・グリッド (data grid)を使用して、オペレーションの作成、取得、更新、および削除を行うことができます。特に、単純データ・グリッド (data grid)を使用すると、データベース内に保管されたデータにより速くアクセスできるようになります。

始める前に

単純データ・グリッドは、WebSphere® Application Server またはスタンドアロンの Java™ アプリケーションで使用できます。どちらかのシナリオに WebSphere eXtreme Scale クライアント がインストールされている必要があります。

このタスクについて

単純データ・グリッドを使用すると、データベースの負荷を軽減することで、動的 Web アプリケーションを高速化できます。メモリー内の任意のデータのキー値のペアを保管して、高価な データベース照会を削減できます。鍵は、java.lang.String または Integer といった 既存の Java 型にできます。値は、任意の直列化可能オブジェクト・タイプにできます。データが必要なたびに、アプライアンス上の単純 データ・グリッド (data grid)が最初にチェックされます。アプライアンスにそのデータがない場合、データベースからデータが取り出され、単純データ・グリッド (data grid)に挿入されます。

手順

1. 単純データ・グリッド (data grid)を作成します。ユーザー・インターフェースで、「データ・グリッド」 > 「単純データ・グリッド」をクリックします。追加アイコン (+) をクリックし、作成する単純データ・グリッド (data grid)の名前を指定します。データ・グリッドの名前では、文字 ^ . ¥¥ / , # \$ @ : ; ¥ * ? < > | = + & % [] " " は使用できません。
2. 単純データ・グリッド (data grid)用の objectgrid.xml ファイルをダウンロードします。作成した単純データ・グリッド (data grid)の構成で、ダウンロード・アイコン (📄) をクリックして、ファイルをご使用のローカル・ファイル・システムに保存します。
3. データ・グリッド (data grid)にアクセスするアプリケーションを作成します。詳しくは、[単純データ・グリッドにアクセスするアプリケーションの開発](#)を参照してください。

次のタスク

- データ・グリッドへのデータの送信を開始する前に、セキュリティーを構成します。詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。
- レプリカを構成します。レプリカにより、基本コピーで障害が発生した場合にもデータ・グリッドのデータが使用可能になります。レプリカを構成するには、「データ・グリッド」 > 「単純データ・グリッド」 > 「拡張属性の表示」をクリックします。アプライアンスが集合内にある場合にのみ、レプリカが作成されます。集合内のアプライアンスの数が n の場合、レプリカの最大数は $n-1$ になります。したがって、3つのレプリカを構成するが、集合内にあるアプライアンスが2つのみである場合、作成されるレプリカは1つのみです。集合にアプライアンスを追加すると、追加のレプリカが作成されます。アプライアンスが集合に追加されたときに新規レプリカを作成できるようにするため、レプリカの数適切な数に設定してください。レプリカ数を編集すると、データ・グリッドの内容が消去されます。
- データ・グリッドの容量限度を構成します。データ・グリッドの容量限度の構成により、集合のストレージ容量が予測可能な形で使用されることを確実にできます。詳しくは、[データ・グリッドの最大容量の構成](#)を参照してください。
- 単純グリッドの存続時間 Evictor を構成します。詳しくは、[存続時間 \(TTL\) Evictor の構成](#)を参照してください。
- DataPower® XC10 アプライアンス ユーザー・インターフェース でデータ・グリッド (data grid)をモニターできます。詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

[動的マップの構成](#)

マップ・テンプレートの事前定義セットに基づいて、動的にマップを作成することができます。

[ロック・ストラテジーの構成](#)

WebSphere eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。WebSphere eXtreme Scale Client for .NET からアクセスするバックキング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。

[存続時間 \(TTL\) Evictor の構成](#)

単純グリッドを作成する際、1つのデフォルト (静的) マップと一連の動的マップが作成されます。デフォルトでは、デフォルト・マップ用に構成された存続時間 Evictor はありません。動的マップがある場合は、作成時間 (*CT)、最終更新時間 (*LUT)、または最終アクセス時間 (*LAT) の TTL 値を設定できます。TTL Evictor がデフォルト・マップでも使用可能になるようにこのデフォルトの動作を変更することも可能です。

Java

[ニア・キャッシュの構成](#)

ローカルのインライン・キャッシュを持つようにクライアントを構成することができます。オプションのこのキ

キャッシュはニア・キャッシュと呼ばれます。これは、各クライアントにある独立したデータ・グリッドであり、リモート用のキャッシュ(サーバー・サイド・キャッシュ)として機能します。ニア・キャッシュは、ロックが使用不可になっているかオプティミスティックとして構成されている場合、デフォルトで使用可能であり、ロックがペシミスティックに構成されている場合は使用することができません。

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

接続済みのクライアントとサーバーに対してネットワーク・トランスポートが公開するホスト名は、**publishHost** プロパティを使用して構成します。

親トピック: [データ・グリッドの構成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

動的マップの構成

マップ・テンプレートの事前定義セットに基づいて、動的にマップを作成することができます。

始める前に

- 単純データ・グリッドを作成します。詳しくは、[単純データ・グリッドの作成](#)を参照してください。
- 動的マップで、どの構成オプションを使用する必要があるかを決定します。詳しくは、[動的マップの構成オプション](#)を参照してください。

次のタスク

定義済みテンプレートを使用して、以下のように動的マップを作成します。

- **Java** **Java API** を使用: 動的マップを定義するために `Session.getMap(String)` メソッドを呼び出す場合の例は、[Java API を使用した動的マップの作成](#)を参照してください。
- **.NET** **2.5+** **.NET API** を使用: 詳しくは、[.NET API を使用した動的マップの作成](#)を参照してください。
- **REST** ゲートウェイを使用: 詳しくは、[REST ゲートウェイの例: 動的マップの作成](#)を参照してください。

親トピック: [単純データ・グリッドの作成](#)

関連タスク:

[ロック・ストラテジーの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

Java [ニア・キャッシュの構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

Java **2.5+** [ニア・キャッシュ無効化の構成](#)

関連資料:

[動的マップの構成オプション](#)

ロック・ストラテジーの構成

WebSphere® eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。 WebSphere eXtreme Scale Client for .NET からアクセスするバックイング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。

始める前に

使用したいロック・ストラテジーを決めます。詳しくは、[ロック・ストラテジー](#)を参照してください。

このタスクについて

次のいずれかのロック・ストラテジーを使用するように各 BackingMap インスタンスを構成することができます。

- オプティミスティック・ロック・モード (デフォルト)
- ペシミスティック・ロック・モード (.NET アプリケーションの場合は必要な)
- なし

手順

オプティミスティック・ロック・モード、ペシミスティック・ロック・モード、またはロックなしモードについては、[動的マップの構成オプション](#)を参照してください。

親トピック: [単純データ・グリッドの作成](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[動的マップの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

 [ニア・キャッシュの構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

 [Java アプリケーションにおけるロックの構成と実装](#)

[ロック・ストラテジーの構成](#)

 [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

 [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

例: [flush メソッドのロック順序](#)

存続時間 (TTL) Evictor の構成

単純グリッドを作成する際、1つのデフォルト (静的) マップと一連の動的マップが作成されます。デフォルトでは、デフォルト・マップ用に構成された存続時間 Evictor はありません。動的マップがある場合は、作成時間 (*CT)、最終更新時間 (*LUT)、または最終アクセス時間 (*LAT) の TTL 値を設定できます。TTL Evictor がデフォルト・マップでも使用可能になるようにこのデフォルトの動作を変更することも可能です。

始める前に

- ご使用の構成用の単純データ・グリッドを作成します。
- どの時間ベースの Evictor を使用するか決定します。特定の Evictor タイプの使用については、[Evictor](#) を参照してください。

手順

1. ユーザー・インターフェースで、「データ・グリッド」 > 「単純データ・グリッド」 > 「`data_grid_name`」 > 「拡張属性の表示」をクリックします。
2. 「次の名前前のマップ用 Evictor」リスト・ボックスで、Evictor タイプを選択します。このオプションを「なし」のままにする場合は、存続時間 Evictor はデフォルト・マップに対してのみ使用不可になります。または、Evictor タイプに「なし」以外を選択すると、TTL がデフォルト・マップに対して使用可能になります。
3. 「次の時間が経過したら存続時間マップからデータを除去」ボックスに値を指定します。デフォルトの TTL 値は 3600 秒です。Evictor タイプが「なし」以外に設定されている場合、ここで指定する値はデフォルト・マップにも適用されます。値が「0」の場合は、グリッド全体に対して TTL 除去がオフになります。
4. 「変更の適用」をクリックして保存します。グリッドの再始動後のデータ損失について警告が表示されます。

例

デフォルトのシナリオ:

このシナリオでは、「次の名前前のマップ用 Evictor」リスト・ボックスで選択された Evictor タイプは「なし」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「3600」です。この結果、*CT、*LUT、および *LAT で終わる動的マップについてのみ、1 時間後にエントリーが除去されます。

デフォルトの Evictor タイプを「なし」以外にカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 Evictor」リスト・ボックスで選択された Evictor タイプは「作成時刻」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「3600」です。この結果、*CT、*LUT、および *LAT で終わる動的マップについて、1 時間後にエントリーが除去されます。デフォルト・マップは、作成時間から 1 時間後にエントリーを除去します。

デフォルトの Evictor タイプと TTL 値をカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 Evictor」リスト・ボックスで選択された Evictor タイプは「最終更新時刻 (Last updated time)」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「4800」です。この結果、*CT、*LUT、および *LAT で終わる動的マップについて、80 分後にエントリーが除去されます。デフォルト・マップは、エントリーが最後に更新された時間から 80 分後にエントリーを除去します。

TTL 値のみをカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 Evictor」リスト・ボックスで選択された Evictor タイプは「なし」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「0」です。この結果、このグリッドのすべての (デフォルトおよび動的) マップについて、TTL の除去はオフになります。

デフォルトの Evictor タイプと TTL 値をカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 Evictor」リスト・ボックスで選択された Evictor タイプは「最終アクセス時間」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「0」です。この結果、このグリッドのすべての (デフォルトおよび動的) マップについて、TTL の除去はオフになります。

Evictor

Evictor は、データ・グリッドからデータを削除します。単純グリッド上の動的マップおよびデフォルト・マップ用の Evictor を構成できます。

親トピック: [単純データ・グリッドの作成](#)

関連概念:

[Evictor](#)

関連タスク:

[動的マップの構成](#)

[ロック・ストラテジーの構成](#)

[Java](#) [ニア・キャッシュの構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

[Java](#) **2.5+** [ニア・キャッシュ無効化の構成](#)

関連資料:

[動的マップの構成オプション](#)

Evictor

Evictor は、データ・グリッドからデータを削除します。単純グリッド上の動的マップおよびデフォルト・マップ用の Evictor を構成できます。

Evictor タイプ

Evictor は、存続時間の概念に基づいてエントリーを除去します。作成時間、最終アクセス時間、または最終更新時間に基づく Evictor を選択できます。デフォルトでは、Evictor は動的マップで作成されます。単純グリッドのデフォルト・マップで Evictor を使用可能にするには、[存続時間 \(TTL\) Evictor の構成](#) を参照してください。

なし

エントリーの期限切れがないように、それによってマップからエントリーが除去されることがないように指定します。

作成時間

作成された時に応じてエントリーが除去されるように指定します。

作成時間 Evictor を使用している場合、Evictor は、作成からの時間がその TTL 値に達するとエントリーを除去します。TTL 値は、アプリケーション構成でミリ秒単位で設定されます。TTL 値を 10 秒に設定すると、エントリーは挿入の 10 秒後に自動的に除去されます。

この値を作成時間 Evictor タイプに設定する場合は注意が必要です。この Evictor は、一定時間にのみ使用される、キャッシュへの妥当な追加量がある場合に、最も有効に使用されます。このストラテジーによって、作成されたものはすべて、一定時間後に除去されます。

作成時間 Evictor タイプは、20 分以下の間隔で株価情報を最新表示するようなシナリオで役立ちます。例えば、ある Web アプリケーションは株価情報の取得をしますが、最新情報を取得することは重要でないとします。この場合、株価情報は 20 分間、データ・グリッドにキャッシュされます。20 分後、マップの有効期限が切れ、除去されます。ほぼ 20 分ごとに、グリッドはデータをデータベースのデータでリフレッシュします。データベースは 20 分ごとに最新の株価情報によって更新されます。

最終アクセス時間

(読み取りか更新かに関わらず) 最後にアクセスされた時に応じてエントリーが除去されるように指定します。

最終更新時間

最後に更新された時に応じてエントリーが除去されるように指定します。

最終アクセス時間 または最終更新時間の Evictor タイプを使用している場合は、作成時間 Evictor を使用している場合よりも TTL 値を低い数値に設定します。エントリーは、アクセスされるたびにリセットされるからです。言い換えれば、値が 15 で、エントリーが 14 秒間存在し、それからアクセスされた場合、このエントリーはあと 15 秒間有効期限が切れることはありません。TTL 値を比較的高い数値に設定した場合は、多くのエントリーがまったく除去されなくなる可能性があります。ただし、この値を 15 秒程度に設定すると、エントリーは頻繁にアクセスされない場合に除去されることとなります。

最終アクセス時間 または最終更新時間 Evictor タイプは、データ・グリッド・マップを使用してクライアントからのセッション・データを保持するシナリオなどで役立ちます。セッション・データは、クライアントがそのセッション・データを一定時間使用しない場合は破棄する必要があります。例えば、セッション・データは、クライアントによるアクティビティが 30 分間なかった後にタイムアウトになるとします。この場合、最終アクセス時間 または最終更新時間の Evictor タイプを使用し、TTL 値を 30 分に設定するのが、このアプリケーションにおいて適切です。

親トピック: [存続時間 \(TTL\) Evictor の構成](#)

関連タスク:

[存続時間 \(TTL\) Evictor の構成](#)

関連資料:

[動的マップの構成オプション](#)

ニア・キャッシュの構成

ローカルのインライン・キャッシュを持つようにクライアントを構成することができます。オプションのこのキャッシュはニア・キャッシュと呼ばれます。これは、各クライアントにある独立したデータ・グリッドであり、リモート用のキャッシュ(サーバー・サイド・キャッシュ)として機能します。ニア・キャッシュは、ロックが使用不可になっているかオプティミスティックとして構成されている場合、デフォルトで使用可能であり、ロックがペシミスティックに構成されている場合は使用することができません。

始める前に

ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。

このタスクについて

ニア・キャッシュは、リモート側で保管されているキャッシュ・データ・セット全体のサブセットへのローカル・メモリー内アクセスを可能にするため、高速です。デフォルトで、クライアント・サイドのニア・キャッシュには最大サイズがないため、クライアントのメモリー不足エラーが発生する可能性があります。ニア・キャッシュのサイズを制御するには、クライアント上で存続時間(TTL)または最長未使用時間(LRU) Evictor を使用可能に設定するクライアント・サイドのオーバーライドを構成してください。クライアント設定のオーバーライドについては、[XML 構成を使用した Java クライアントの構成](#)および[Java クライアントのプログラマチック構成](#)を参照してください。

手順

動的マップ名を設定することで、クライアント上にニア・キャッシュが作成されるように設定できます。動的マップは、オプティミスティック・ロックまたは非ロック式になるように設定する必要があります。ニア・キャッシュを使用可能に設定しているマップ名の例を以下に示します。

```
my_grid.NONE  
my_grid.NONE.0
```

詳しくは、[動的マップの構成](#)を参照してください。動的マップ構成オプションのリストについては、[動的マップの構成オプション](#)を参照してください。

タスクの結果

ニア・キャッシュが使用可能になっているかどうかを確認するには、クライアントで `BackingMap.isNearCacheEnabled()` メソッドを実行します。また、ログ・ファイルで `CWOBJ1128I` メッセージを検索して、ニア・キャッシュが使用可能になっているかどうかを確認することもできます。

Java 2.5+ [ニア・キャッシュ無効化の構成](#)

ニア・キャッシュ無効化を構成して、失効データをニア・キャッシュから可能な限り迅速に除去できます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化操作がトリガーされます。このメカニズムは、もう 1 つのオプション(ニア・キャッシュでの存続時間(TTL) 除去の使用)よりも迅速に機能します。

親トピック: [単純データ・グリッドの作成](#)

関連タスク:

[動的マップの構成](#)

[ロック・ストラテジーの構成](#)

[存続時間\(TTL\) Evictor の構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

ニア・キャッシュ無効化の構成

2.5+ ニア・キャッシュ無効化を構成して、失効データをニア・キャッシュから可能な限り迅速に除去できます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化操作がトリガーされます。このメカニズムは、もう 1 つのオプション (ニア・キャッシュでの存続時間 (TTL) 除去の使用) よりも迅速に機能します。

始める前に

- IBM eXtremeIO を使用している必要があります。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。
- ニア・キャッシュを使用している必要があります。ニア・キャッシュが使用可能になっているかどうかを確認するには、クライアントで `BackingMap.isNearCacheEnabled()` メソッドを実行します。ニア・キャッシュの構成について詳しくは、[ニア・キャッシュの構成](#)を参照してください。
- ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。

このタスクについて

ニア・キャッシュ無効化を使用可能にすると、リモート・データが変更されるとニア・キャッシュが更新されるため、リモート・データ・グリッドからのデータ・セットがより正確なものになります。

手順

2.5+ ニア・キャッシュ無効化オプションを含む動的マップを作成します。詳しくは、[動的マップの構成](#)を参照してください。ニア・キャッシュ無効化が使用可能になっているマップの以下の例を参照してください。

```
my_grid.NCI
my_grid.CT.NCI
my_grid.LAT.NCI
my_grid.LUT.NCI
my_grid.NONE.NCI
my_grid.CT.0.NCI
my_grid.LAT.0.NCI
my_grid.LUT.0.NCI
my_grid.NONE.0.NCI
```

動的マップ構成オプションのリストについては、[動的マップの構成オプション](#)を参照してください。

親トピック: [Java](#) [ニア・キャッシュの構成](#)

関連タスク:

[動的マップの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

関連資料:

[動的マップの構成オプション](#)

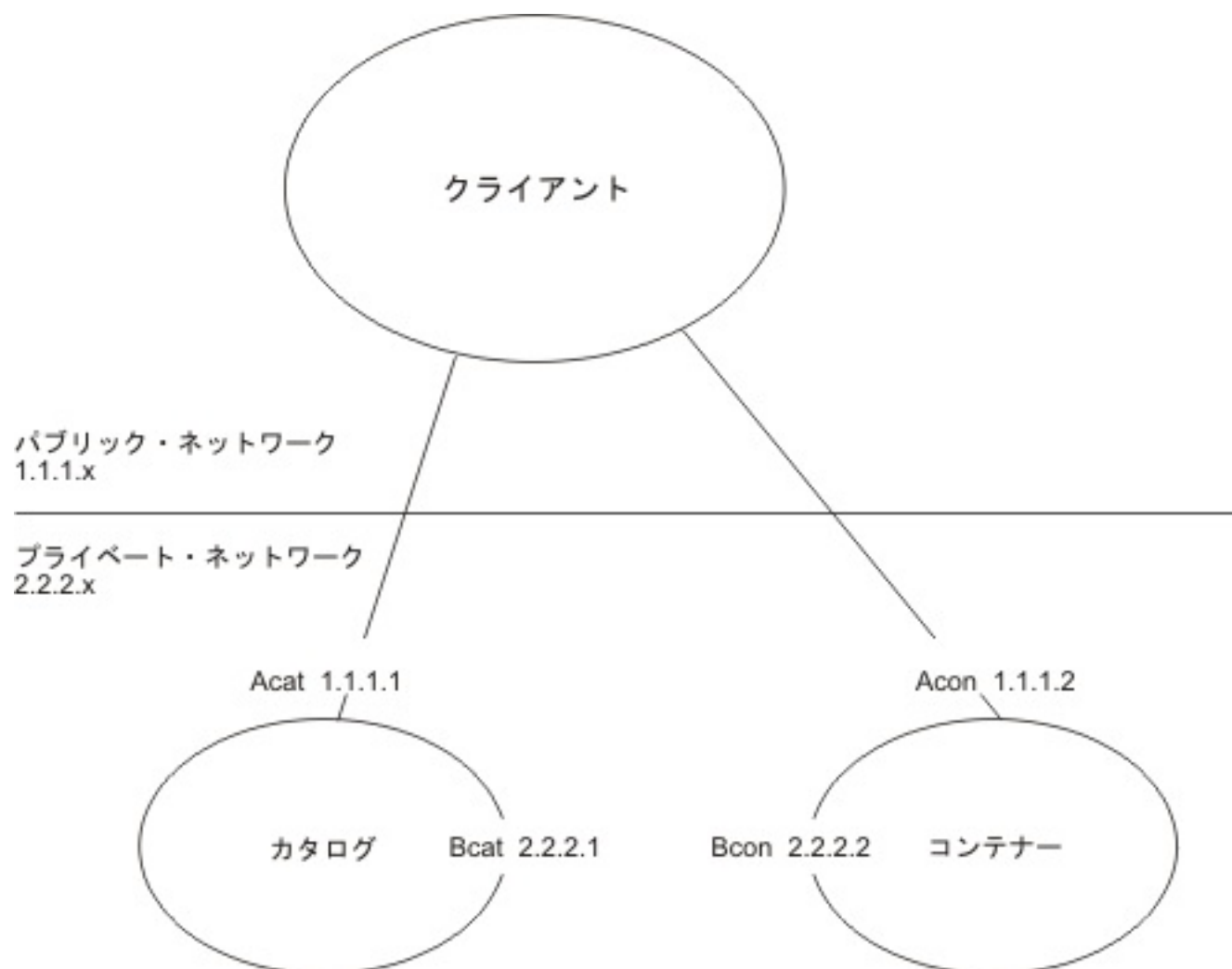
クライアントとサーバー間の通信用ホスト名の管理

接続済みのクライアントとサーバーに対してネットワーク・トランスポートが公開するホスト名は、**publishHost** プロパティを使用して構成します。

このタスクについて

ドメイン・ネーム・システム (DNS) サーバーまたはホスト・ファイルを併用して、**publishHost** プロパティで公開されるホストを構成したり、オーバーライドしたりすることができます。これで、クライアントとサーバーの間に行き来する通信について、一層きめ細かい制御ができるようになり、また、データ・グリッド内のノード間の通信を最適化する機能も提供されます。例えば、次の図は、どのようにクライアントをパブリックのインターネット・プロトコル (IP) (1.1.1.x) に接続し、サーバーをプライベートの IP (2.2.2.x) に接続できるかを示しています。この例では、カタログ・サーバーとコンテナ・サーバーには、それぞれのネットワーク・カードに対する 2 つのホスト名、すなわち、パブリックで使用可能な IP アドレス用の A* とプライベート IP アドレス用の B* があります。

図 1. **publishHost** プロパティのトポロジー



C* ホスト名は、公開する総称ホスト名として使用される、**publishHost** プロパティの新しい値を表します。これは A* または B* の代わりとしても使用されます。A* 名はプライベート・ネットワークで通信したいサーバーには不適切であり、B* 名はパブリック・ネットワークで通信したいクライアントには不適切であるからです。これにより、代替パブリック・アドレスおよび代替プライベート・アドレスの中から選択できるようになります。

手順

1. **publishHost** プロパティをカタログ・サーバーの Ccat およびコンテナ・サーバーの Ccon に設定します。このプロパティは、クライアントとサーバーの通信時に公開され、DNS にも公開されるホスト名を設定します。
2. オプション: **listenerHost** プロパティを設定します。このプロパティは、各サーバーがどの IP アドレスを listen するかを制御します。
3. パブリック・ネットワークを使用するクライアントの DNS (または /etc/hosts ファイル) が Ccat および Ccon ホストを指していることを確認します。これらのホストのパブリック IP アドレスはそれぞれ Ccat 1.1.1.1 および Ccon 1.1.1.2 です。プライベート・ネットワーク上のサーバーの DNS (または /etc/hosts ファイル) が Ccat および Ccon ホストを指していることを確認します。これらのホストのプライベート IP アドレスはそれぞれ Ccat 2.2.2.1 および Ccon 2.2.2.2 です。
4. カタログ・サーバー (Ccat) の **publishHost** 値を、クライアントおよびサーバー上のオブジェクト・グリッド構成で使用されるカタログ・サーバー・エンドポイントのホスト名として使用します。

タスクの結果

これで、サーバー間を流れるトラフィックはパブリック・ネットワーク上にあり、クライアントからコンテナに流れるトラフィックはプライベート・ネットワーク上にあるようになりました。

親トピック: [単純データ・グリッドの作成](#)

関連タスク:

[動的マップの構成](#)

[ロック・ストラテジーの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

[Java ニア・キャッシュの構成](#)

セッション・データ・グリッドの作成

Java または .NET アプリケーションからの HTTP セッションを保管するデータ・グリッドを作成することができます。

[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パーシスタンスの構成](#)

アプライアンスをセッション管理で使用するよう WebSphere® Application Server アプリケーションを構成します。新規アプリケーションのインストール時にアプライアンスを選択することも、既存のアプリケーションまたはサーバーの設定をアプライアンスを使用するよう更新することもできます。

[WebSphere Application Server での HTTP セッション・マネージャーの構成](#)

WebSphere Application Server はセッション管理機能を備えていますが、要求の数が増えるとパフォーマンスが低下します。WebSphere eXtreme Scale には、セッション・レプリカ生成、高可用性、優れたスケーラビリティ、および堅固な構成オプションを備えたセッション管理実装がバンドルされています。

[WebSphere Portal での HTTP セッション・マネージャーの構成](#)

WebSphere Portal の HTTP セッションをデータ・グリッドに保持できます。

[各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成](#)

WebSphere eXtreme Scale クライアント には、Web コンテナの デフォルト・セッション・マネージャーをオーバーライドするセッション管理実装がバンドルされています。この実装は、セッション・レプリカ生成、高可用性、より優れたスケーラビリティと 構成オプションを提供します。Tomcat などさまざまなアプリケーション・サーバーに対して WebSphere eXtreme Scale クライアント セッション複製マネージャーおよび汎用組み込み ObjectGrid コンテナの開始を有効にします。

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

以前に設定したメモリー間複製セッションまたはデータベース・セッションをマイグレーションして、WebSphere eXtreme Scale セッション管理を使用できます。

.NET [2.5+ ASP.NET セッション状態ストア・プロバイダーの構成](#)

データ・グリッド内の ASP.NET アプリケーションからセッション状態を保管できます。

[2.5+ HTTP セッション・サイズを表示](#)

`xscmd` コーティリティーを使用して、Java アプリケーションでのセッションのサイズを表示することができます。セッション・サイズを表示することで、データ・グリッドの問題のトラブルシューティングが容易になります。

[サーブレット・コンテキスト初期化パラメーター](#)

以下に示すサーブレット・コンテキスト初期化パラメーターのリストは、選択した接続メソッドに必要なスプライサー・プロパティ・ファイルに指定できるものです。

[splicer.properties ファイル](#)

`splicer.properties` ファイルには、サーブレット・フィルター・ベースのセッション・マネージャーを構成するための、すべての構成オプションが含まれます。

親トピック: [データ・グリッドの構成](#)

データ・グリッド (data grid)への WebSphere Application Server HTTP セッション・パーシスタンスの構成

アプライアンスをセッション管理で使用するよう WebSphere® Application Server アプリケーションを構成します。新規アプリケーションのインストール時にアプライアンスを選択することも、既存のアプリケーションまたはサーバーの設定をアプライアンスを使用するよう更新することもできます。

始める前に

WebSphere Application Server で構成を変更する前に、以下の情報が必要です。

- 構成する WebSphere Application Server セルに対するアクセス権限。
- アプライアンスの IP アドレスまたは完全修飾ホスト名。
- アプライアンスのユーザー・インターフェースにログインするために使用するユーザー ID とパスワード。データ・グリッドを作成するには、データ・キャッシュ作成許可が必要です。
- WebSphere Application Server 構成に WebSphere eXtreme Scale クライアントがインストールされていること。詳しくは、[WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- WebSphere Application Server 管理コンソールで使用可能に設定されるグローバル・セキュリティー。アプライアンスのトランスポート層セキュリティーが使用可能に設定されている場合、またはクライアントでトランスポート層セキュリティーを使用するようになりたい場合は、グローバル・セキュリティーを使用可能に設定してください。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

手順

- アプリケーションのインストール時にセッション管理を構成するには、以下の手順を実行します。
 1. WebSphere Application Server の管理コンソールで、「アプリケーション」 > 「新規アプリケーション」 > 「新規エンタープライズ・アプリケーション」をクリックします。「詳細」パスを選択してアプリケーションを作成し、初期のウィザード・ステップを完了します。
 2. ウィザードの「**eXtreme Scale** セッション管理設定」ステップで、使用するデータ・グリッドを構成します。「セッション・パーシスタンスの管理」フィールドで、**WebSphere DataPower XC10** アプライアンスを選択します。アプライアンス、および使用するアプライアンス上のデータ・グリッド (data grid)の情報を入力します。データ・グリッド (data grid)を作成することも、既にアプライアンスで構成されている既存のデータ・グリッド (data grid)を使用することもできます。

アプライアンス上の既存のデータ・グリッド (data grid)内でセッションを保存する場合は、使用するデータ・グリッド (data grid)の名前を知っていなければなりません。ただし、アプリケーションの構成時にアプライアンスにデータ・グリッド (data grid)を作成することができます。アプライアンス・ユーザー・インターフェースでアプリケーションを構成する前にセッション・データ・グリッド (data grid)を作成したい場合は、「データ・グリッド」 > 「セッション」をクリックします。「追加」アイコン (+) をクリックして、作成するセッション・データ・グリッドの名前を指定します。データ・グリッドの名前では、文字 ^ . ¥ ¥ / , # \$ @ : ; ¥ * ? < > | = + & % [] " " は使用できません。

3. ウィザードのステップを完了して、アプリケーションのインストールを終了します。

wsadmin スクリプトを使用して、アプリケーションをインストールすることもできます。次の例では、**-SessionManagement** パラメーターにより、管理コンソールで作成可能なものと同じ構成を作成します。

```
AdminApp.install('C:/A.ear', '[ -nopreCompileJSPs -distributeApp
-nouseMetaDataFromBinary -nodeployejb -appname A -edition 8.0
-createMBeansForResources -noreloadEnabled -nodeployws -validateinstall
off -noprocessEmbeddedConfig -filepermission .*¥.dll=755#.*¥.so=755#.*¥.a=755#.*
¥.sl=755
-buildVersion Unknown -noallowDispatchRemoteInclude -noallowServiceRemoteInclude
-asyncRequestDispatchType DISABLED -nouseAutoLink -SessionManagement [[true
XC10SessionManagement myXC10.ibm.com:!:username:!:password:!:AGrid80]]
-MapWebModToVH [[MicroWebApp microwebapp.war,WEB-INF/web.xml default_host]
[MicroSipApp
microsipapp.war,WEB-INF/web.xml default_host] [MicroDG1App microdglapp.war,WEB-
INF/web.xml
default_host] [MicroDG2App microdgd2app.war,WEB-INF/web.xml default_host]
[MicroSip2App
microsip2app.war,WEB-INF/web.xml default_host]]]')
```

- **WebSphere Application Server** の管理コンソールで既存のアプリケーション上に セッション管理を構成する場合は以下を行います。

注: WebSphere DataPower XC10 アプライアンス を使用するようアプリケーションが設定されると、「セッ

セッション管理のオーバーライド」ボックスにチェック・マークが付けられます。つまり、WebSphere Application Server 構成に対して行われたサーバー・レベルのセッション設定はアプリケーション・レベルのセッション設定によって上書きされることとなります。これが望ましくない場合は、WebSphere DataPower XC10 アプライアンスをサーバー・レベルで使用可能にすることができます。

1. WebSphere Application Server の管理コンソールで、「アプリケーション」 > 「アプリケーション・タイプ」 > 「WebSphere エンタープライズ・アプリケーション」 > 「application_name」 > 「Web モジュール・プロパティ」 > 「セッション管理」 > 「eXtreme Scale セッション管理設定」をクリックします。
2. フィールドを更新して、データ・グリッドへのセッション・パーシスタンスを使用可能にします。

wsadmin スクリプトを使用して、アプリケーションを更新することもできます。次の例では、**-SessionManagement** パラメーターにより管理コンソールで作成するものと同じ構成を作成できます。

```
AdminApp.edit('A-edition9.0', '[ -SessionManagement [[true  
XC10SessionManagement myXC10.ibm.com::username::password::AGrid80]]]')
```

渡される :: 文字は、区切り文字として使用されます。渡される値は次のとおりです。

```
applicationIdentifier::username::password::  
gridName
```

変更を保存する場合、アプリケーションはアプライアンス上のセッション・パーシスタンスに構成済みのデータ・グリッド (data grid) を使用します。

- 既存のサーバー上でセッション管理を構成するには、以下を行います。

1. WebSphere Application Server の管理コンソールで、「サーバー」 > 「サーバー・タイプ」 > 「WebSphere アプリケーション・サーバー」 > 「server_name」 > 「セッション管理」 > 「eXtreme Scale セッション管理設定」をクリックします。
2. フィールドを更新して、セッション・パーシスタンスを使用可能にします。

また、以下の wsadmin ツール・コマンドで、既存のサーバーでのセッション管理を構成できます。

```
AdminTask.configureServerSessionManagement('[-nodeName my_node  
-serverName server1 -enableSessionManagement true -sessionManagementType  
XC10SessionManagement -XC10SessionManagement [-applianceIdentifier myserver.ibm.com  
-userName -password ***** -gridName myTestGrid]]')
```

変更を保存すると、今後サーバーはサーバー上で稼働するすべてのアプリケーションでセッション・パーシスタンスの構成済み データ・グリッド (data grid) を使用します。

- HTTP セッション構成の他の局面を編集する場合は、splicer.properties ファイルを編集します。**sessionFilterProps** カスタム・プロパティを見つけ、splicer.properties ファイルのパスの場所を取得できます。サーバー・レベルでセッション・パーシスタンスを構成した場合、カスタム・プロパティの名前は com.ibm.websphere.xs.sessionFilterProps です。アプリケーション・レベルでセッション・パーシスタンスを構成した場合、カスタム・プロパティの名前は `<application_name>,com.ibm.websphere.xs.sessionFilterProps` です。これらのカスタム・プロパティは、おそらく、次のいずれかの場所にあります。

- WebSphere Application Server Network Deployment 環境の場合: splicer.properties ファイルはデプロイメント・マネージャー・プロファイル・パス上にあります。
- スタンドアロン WebSphere Application Server 環境の場合: アプリケーション・サーバー上のカスタム・プロパティ

示されたファイルを開き、変更し、ノードを同期できます。それによって、更新されたプロパティ・ファイルが、構成内の他のノードに伝搬されます。セッションを適切に存続させるために、すべてのアプリケーション・サーバー・ノードは、splicer.properties ファイルが指定されたパスにあることを必要とします。

重要: URL 再書き込みを使用するセッションのパーシスタンスを有効にするには、splicer.properties ファイル内の **useURLEncoding** プロパティを true に設定します。

splicer.properties ファイルのプロパティに関して詳しくは、[splicer.properties ファイル](#)を参照してください。

タスクの結果

データ・グリッド (data grid) へのセッションをパーシストするように HTTP セッション・マネージャーが構成されました。セッションがタイムアウトになると、項目はデータ・グリッドから除去されます。WebSphere Application Server 管理コンソールでのセッション・タイムアウト値の更新について詳しくは、[セッション管理設定](#)を参照してください。

アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアント

トから到達できない場合、クライアントは、代わりに WebSphere Application Server の基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとアプライアンス間にネットワーク上の問題がある。
- アプライアンス上のサーバー・プロセスが停止されている。

最長未使用時間セッションは、Web コンテナのセッション・キャッシュから無効にされます。アプライアンス上のデータ・グリッドが使用可能になった場合、Web コンテナ・キャッシュから無効化されたセッションは、リモート・データ・グリッドからデータを取得して、データを新規セッションにロードすることができます。アプライアンス上のデータ・グリッド全体が使用不可の状態、セッションがセッション・キャッシュから無効にされている場合、ユーザー・セッション・データは失われます。この問題があるため、負荷をかけられた状態でシステムを実行している場合は、実動データ・グリッド全体をシャットダウンしないでください。

注意:

このシナリオを構成すると、**IBM WebSphere DataPower XC10** アプライアンスのセキュリティー資格情報が **WebSphere Application Server** 構成に自動的に保管されます。初期構成後にデータ・グリッドの資格情報を変更した場合には、**WebSphere Application Server** には適切な資格情報が含まれなくなります。 **eXtreme Scale** セッション管理設定を再度適用することで、資格情報を再設定できます。

次のタスク

- データ・グリッドへのデータの送信を開始する前に、セキュリティーを構成します。詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。
- レプリカを構成します。レプリカにより、基本コピーで障害が発生した場合にもデータ・グリッドのデータが使用可能になります。レプリカを構成するには、「データ・グリッド」 > 「セッション」 > 「拡張属性の表示」をクリックします。アプライアンスが集合内にある場合にのみ、レプリカが作成されます。集合内のアプライアンスの数が n の場合、レプリカの最大数は $n-1$ になります。したがって、3つのレプリカを構成するが、集合内にあるアプライアンスが2つのみである場合、作成されるレプリカは1つのみです。集合にアプライアンスを追加すると、追加のレプリカが作成されます。アプライアンスが集合に追加されたときに新規レプリカを作成できるようにするため、レプリカの数適切な数に設定してください。レプリカ数を編集すると、データ・グリッドの内容が消去されます。
- データ・グリッドの容量限度を構成します。データ・グリッドの容量限度の構成により、集合のストレージ容量が予測可能な形で使用されることを確実にできます。詳しくは、[データ・グリッドの最大容量の構成](#)を参照してください。
- DataPower XC10 アプライアンス ユーザー・インターフェース でセッション・データ・グリッド (data grid) をモニターできます。詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

親トピック: [セッション・データ・グリッドの作成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連資料:

[splicer.properties](#) ファイル

関連情報:

☞ [コンソールを使用したエンタープライズ・アプリケーション・ファイルのインストール](#)

☞ [wsadmin スクリプト](#)

WebSphere Application Server での HTTP セッション・マネージャーの構成

WebSphere® Application Server はセッション管理機能を備えています。要求の数が増えるとパフォーマンスが低下します。WebSphere eXtreme Scale には、セッション・レプリカ生成、高可用性、優れたスケーラビリティ、および堅固な構成オプションを備えたセッション管理実装がバンドルされています。

始める前に

- デフォルトでは、XC10 スプライサー・プロパティ・ファイルはシステムが生成するユーザー ID とパスワードを使用します。システム生成ユーザー ID は XC10_USER_GridName です。このユーザー ID のパスワードは変更できません。ただし、ユーザー ID を別のユーザーに変更する場合は、credentialGeneratorProps プロパティを変更することによって、splicer.properties ファイル内のユーザー ID とパスワードの組み合わせを手動で更新する必要があります。詳しくは、[splicer.properties ファイルを編集する方法](#)を参照してください。システム生成のユーザー ID とパスワードは、始動時に splicer.properties ファイルを取得するためにだけ使用されます。これらの資格情報は WebSphere Application Server クライアントでは使用されなくなりました。
- eXtreme Scale セッション・マネージャーを使用するには、WebSphere eXtreme Scale は、WebSphere Application Server または WebSphere Application Server Network Deployment セルにインストールする必要があります。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- HTTP セッション複製用の WebSphere eXtreme Scale が WebSphere Application Server で使用される場合は、該当するすべての Web アプリケーションおよびその Web アプリケーションをホストするアプリケーション・サーバーについて「オーバーフローの許可」セッション管理設定にチェック・マークを付ける必要があります。詳しくは、「[セッション管理設定](#)」を参照してください。
- アプライアンスで Secure Sockets Layer (SSL) を使用可能に設定した場合は、WebSphere Application Server でグローバル・セキュリティーを使用可能に設定する必要があります。SSL がサポートされている集合に対して SSL を使用したい場合も、これを使用可能に設定する必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#)を参照してください。

このタスクについて

WebSphere eXtreme Scale HTTP セッション・マネージャーは、キャッシング用に、組み込みサーバーとリモート・サーバーの両方をサポートします。

• 組み込みシナリオ

組み込みシナリオでは、サーブレットが実行される同じプロセス内でデータ・グリッド・サーバーが相互に連結されています。セッション・マネージャーはローカルの ObjectGrid インスタンスと直接通信できるため、コストのかかるネットワーク遅延を回避することができます。

WebSphere Application Server を使用している場合は、提供された [wxs_home/session/samples/objectGrid.xml](#) および [wxs_home/session/samples/objectGridDeployment.xml](#) ファイルを、ご使用の Web アーカイブ (WAR) ファイルの META-INF ディレクトリーに配置してください。アプリケーションが始動して、セッション・マネージャーと同じプロセス内の eXtreme Scale コンテナを自動的に始動すると、eXtreme Scale がこれらのファイルを自動的に検出します。

objectGridDeployment.xml ファイルは変更することができます。複製を同期にするか非同期にするか、またはレプリカをいくつ構成するかによってこのファイルを変更します。

• リモート・サーバー・シナリオ

リモート・サーバー・シナリオでは、実行されるコンテナ・サーバーはサーブレットとは異なるプロセスにあります。セッション・マネージャーはリモートのコンテナ・サーバーと通信します。リモートのネットワーク接続のコンテナ・サーバーを使用するためには、カタログ・サービス・ドメインのホスト名およびポート番号によってセッション・マネージャーを構成する必要があります。そうすると、セッション・マネージャーは、eXtreme Scale クライアント接続を使用して、カタログ・サーバーおよびコンテナ・サーバーと通信します。

コンテナ・サーバーを独立したスタンドアロン・プロセス内で開始する場合は、セッション・マネージャーのサンプル・ディレクトリーで提供される objectGridStandAlone.xml ファイルと objectGridDeploymentStandAlone.xml ファイルを使用して、データ・グリッド・コンテナを開始します。

手順

- アプリケーションを接合することで、アプリケーションがセッション・マネージャーを使用できるようにします。セッション・マネージャーを使用するためには、適切なフィルター宣言をアプリケーションの Web デプロイメント記述子に追加する必要があります。さらに、セッション・マネージャー構成パラメーターが、デプロイメ

ント記述子内のサブレット・コンテキスト初期化パラメーターという形式でセッション・マネージャーに渡されます。この情報は、以下に示す複数の方法でアプリケーションに導入することができます。

- **WebSphere Application Server との自動接合**

アプリケーションのインストール時にデータ・グリッドに対して HTTP セッション・マネージャーを使用するようにアプリケーションを構成することができます。また、アプリケーションまたはサーバーの構成を編集して、WebSphere eXtreme Scale HTTP セッション・マネージャーを使用することもできます。詳しくは、[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パースタンスの構成](#)を参照してください。

- **カスタム・プロパティを使用したアプリケーションの自動接合**

WebSphere Application Server または WebSphere Application Server Network Deployment でアプリケーションを実行している場合には、アプリケーションを手動で接合する必要はありません。

カスタム・プロパティをセルまたはサーバーに追加して、そのスコープにあるすべての Web アプリケーションに `splicer.properties` ファイルを設定します。次のステップを実行して、カスタム・プロパティを構成します。

- a. WebSphere Application Server 管理コンソールで、カスタム・プロパティを設定する正しいパスにナビゲートし、`splicer.properties` ファイルの場所を指示します。
 - カスタム・プロパティをすべてのアプリケーションまたは特定のアプリケーションに設定するには、「システム管理」 > 「セル」 > 「カスタム・プロパティ」をクリックします。
 - 特定のアプリケーション・サーバー上のすべてのアプリケーションに適用するカスタム・プロパティを設定するには、「アプリケーション・サーバー」 > 「<server_name>」 > 「管理」 > 「カスタム・プロパティ」をクリックします。プロパティ名は `com.ibm.websphere.xs.sessionFilterProps` で、その値はアプリケーションが必要とする `splicer.properties` ファイルの場所です。ファイルの場所のパスは、例えば `/opt/splicer.properties` です。
- b. `com.ibm.websphere.xs.sessionFilterProps` カスタム・プロパティを追加します。このカスタム・プロパティの値には、編集する `splicer.properties` ファイルのロケーションが指定されています。このファイルは、デプロイメント・マネージャーに存在します。セル・レベルのカスタム・プロパティを使用して特定のアプリケーション用の `splicer.properties` ファイルを指示する必要がある場合は、カスタム・プロパティの名前を `<application_name>,com.ibm.websphere.xs.sessionFilterProps` のように入力します。ここで、`application_name` は、カスタム・プロパティを適用するアプリケーションの名前を示します。

重要: 更新済み `splicer.properties` ファイルが、セッション・レプリカ生成のために接合されるアプリケーションをホスティングしているアプリケーション・サーバーを含んでいるすべてのノードで、同じパス上に存在することを確認してください。

使用可能なスコープはセル、サーバー、およびアプリケーションであり、デプロイメント・マネージャーで実行している場合にのみ使用可能です。別のスコープが必要な場合は、Web アプリケーションを手動で接合してください。

要確認: 自動接合オプションは、アプリケーションを実行しているすべてのノードの同じパスに `splicer.properties` ファイルが存在する場合にのみ機能する点にも注意してください。Windows ノードと UNIX ノードがともに存在する混合環境では、このオプションは使用できないため、アプリケーションを手動で接合する必要があります。

- **addObjectGridFilter スクリプトによるアプリケーションの接合**

eXtreme Scale とともに提供されるコマンド行スクリプトを使用して、フィルター宣言と構成によってアプリケーションをサブレット・コンテキスト初期化パラメーターの形式で接合します。WebSphere Application Server デプロイメントの場合、このスクリプトは `<was_home>/optionalLibraries/ObjectGrid/session/bin/addObjectGridFilter.bat/sh` にあります。スタンドアロン・デプロイメントの場合、スクリプトは `WXS_HOME/ObjectGrid/session/bin/addObjectGridFilter.sh/bat` にあります。

addObjectGridFilter スクリプトは 2 つのパラメーターを使用します。

- アプリケーション - 接合するエンタープライズ・アーカイブ・ファイルへの絶対パス
- 各種構成プロパティが入ったスプライサー・プロパティ・ファイルへの絶対パス

このスクリプトの使用形式は次のとおりです。

Windows

```
addObjectGridFilter.bat [ear_file] [splicer_properties_file]
```

UNIX

```
addObjectGridFilter.sh [ear_file] [splicer_properties_file]
```

UNIX UNIX 上の WebSphere Application Server にインストールされている eXtreme Scale の使用例:

- a. cd `wxs_home/optionalLibraries/ObjectGrid/session/bin`
- b. `addObjectGridFilter.sh /tmp/mySessionTest.ear`
`was_root/optionalLibraries/ObjectGrid/session/samples/splicer.properties`

UNIX UNIX 上のスタンドアロン・ディレクトリーにインストールされている eXtreme Scale の使用例:

- a. cd `was_root/session/bin`
- b. `addObjectGridFilter.sh /tmp/mySessionTest.ear`
`was_root/session/samples/splicer.properties`

接合されるサーブレット・フィルターは構成値のデフォルトを保持します。これらのデフォルト値は、2 番目の引数にあるプロパティ・ファイルで指定する構成オプションでオーバーライドできます。使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

eXtreme Scale インストールとともに提供されるサンプルの `splicer.properties` ファイルを変更して使用することができます。また、各サーブレットを拡張することによってセッション・マネージャーを挿入する、`addObjectGridServlets` スクリプトも使用できます。ただし、推奨スクリプトは `addObjectGridFilter` スクリプトです。

◦ **Ant** ビルド・スクリプトによるアプリケーションの手動接合

WebSphere eXtreme Scale には Apache Ant で使用できる `build.xml` ファイルが同梱されています。このファイルは WebSphere Application Server インストールの `was_root/bin` フォルダーに含まれています。`build.xml` ファイルを変更して、セッション・マネージャー構成プロパティを変更できます。構成プロパティは `splicer.properties` ファイル内のプロパティ名と同一です。`build.xml` を変更し、次のコマンドを実行して Ant プロセスを開始します。

- **UNIX** `ant.sh`、`ws_ant.sh`
- **Windows** `ant.bat`、`ws_ant.bat`

(UNIX) または (Windows)

◦ **Web** 記述子の手動更新

Web アプリケーションに同梱されている `web.xml` ファイルを編集して、フィルター宣言、そのサーブレット・マッピング、およびサーブレット・コンテキスト初期化パラメーターが組み込まれるようにします。この方法はエラーを起こしやすいため、使用しないようにしてください。

使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

2. アプリケーションをデプロイします。サーバーやクラスターに対して通常使用する手順に従ってアプリケーションをデプロイしてください。アプリケーションをデプロイした後、アプリケーションを始動することができます。
3. アプリケーションにアクセスします。これで、セッション・マネージャーおよび WebSphere eXtreme Scale と対話するアプリケーションにアクセスすることができます。

次のタスク

アプリケーションの装備時にセッション・マネージャーの構成属性の大部分を変更して、セッション・マネージャーを使用するようにすることができます。これらの属性には、同期または非同期のレプリカ生成、メモリー内セッション・テーブル・サイズなどがあります。アプリケーションの装備時に変更できる属性を別にすれば、アプリケーションのデプロイメント後に変更できるその他の構成属性は、WebSphere eXtreme Scale サーバー・クラスター・トポロジーと、それらのクラスターのクライアント (セッション・マネージャー) がそれらのクラスターに接続する方法に関する属性のみです。

リモート・シナリオの動作: アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアントから到達できない場合、クライアントは、代わりに WebSphere Application Server の基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとリモート・コンテナ・サーバー間のネットワークの問題
- リモート・コンテナ・サーバーのプロセスが停止した場合

sessionTableSize パラメーターによって指定される、メモリー内に保持されるセッション参照の数は、セッションが基本 Web コンテナ内に保管されている場合、そのまま維持されます。セッション数が **sessionTableSize** の値を超えると、最長未使用時間を基にセッションが Web コンテナ・セッション・キャッシュで無効化されます。リモート・データ・グリッドが使用可能になると、Web コンテナ・キャッシュで無効化されたセッションは、リモート・データ・グリッドからデータを取得し、データを新規セッションにロードできます。リモート・データ・グリッド全体が使用不可なまま、セッションがセッション・キャッシュで無効化されると、ユーザー・セッション・データは失われま

す。このような問題があるため、負荷の下でシステムを実行する場合、実動リモート・データ・グリッド全体をシャットダウンすることはしないでください。

親トピック: [セッション・データ・グリッドの作成](#)

WebSphere Portal での HTTP セッション・マネージャーの構成

WebSphere® Portal の HTTP セッションをデータ・グリッドに保持できます。

始める前に

WebSphere eXtreme Scale クライアント と WebSphere Portal 環境が、次の要件を満たしている必要があります。

- WebSphere Application Server および WebSphere Portal ノードに WebSphere eXtreme Scale クライアント をインストールしてください。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- WebSphere Portal バージョン 7 以降。
- カスタム・ポートレットは、WebSphere Portal 内で構成しなければなりません。現在、WebSphere Portal に 付属の管理ポートレットは、データ・グリッドと統合できません。

このタスクについて

WebSphere DataPower XC10 アプライアンス を WebSphere Portal 環境に導入することは、以下のシナリオでメリットがあります。

重要: 以下のシナリオでは、メリットについて紹介しますが、WebSphere DataPower XC10 アプライアンス を環境に導入することにより、WebSphere Portal 層での プロセッサ使用量が増える場合もあります。

- **セッション・パーシスタンスが必要な場合**

例えば、WebSphere Portal Server の 障害時でもカスタム・ポートレットのセッション・データを使用可能な状態で 維持する必要がある場合は、HTTP セッションを WebSphere DataPower XC10 アプライアンス データ・グリッドに 保持できます。複数のサーバーにデータを複製しておく、データの可用性が高まります。

- **複数データ・センター・トポロジー**

ロケーションが物理的に異なる複数のデータ・センターに またがるトポロジーの場合、WebSphere Portal HTTP セッションを WebSphere DataPower XC10 アプライアンス データ・グリッドに 保持できます。セッションは、複数あるデータ・センター内のデータ・グリッドに 複製されます。あるデータ・センターで障害が起こると、セッションは、データ・グリッドのデータのコピーを 保持する別のデータ・センターにロールオーバーされます。

- **WebSphere Portal Server 層のメモリー所要量を低下させる場合**

セッション・データを コンテナ・サーバーのリモート層にオフロードすることで、セッションのサブセットが WebSphere Portal Server 上に 存在することになります。このデータのオフロードにより、WebSphere Portal Server 層のメモリー所要量が低下します。

手順

1. セッションをデータ・グリッドに格納できるように、wps WebSphere Portal アプリケーションと任意のカスタム・ポートレットを構成します。

詳しくは、[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パーシスタンスの構成](#)を参照してください。この操作により、カスタム・ポートレットが接合され、データ・グリッドでセッション・パーシスタンスが有効になります。

2. WebSphere Portal サーバーとアプライアンスに対して Transport Layer Security/Secure Sockets Layer (TLS/SSL) が構成されている場合は、TLS/SSL トラストストアを構成する必要があります。
 - この結果、WebSphere Portal サーバーからアプライアンスへのアウトバウンド通信で TL/SSL 通信が使用される場合は、アプライアンス証明書を WebSphere Application Server 構成に追加する必要があります。以下のように **addXC10PublicCert.py** スクリプトを使用します。このスクリプトは `was_root/bin` ディレクトリーにあります。

```
wsadmin.bat -conntype SOAP -port <PORTAL_SERVER_SOAP_PORT> -lang jython  
-user wpsadmin -password wpsadmin -f addXC10PublicCert.py
```

- この結果、アプライアンスから WebSphere Portal サーバーへのインバウンド通信で TLS/SSL が使用される場合、WebSphere Portal サーバーの公開証明書を組み込むようにアプライアンス・トラストストアを更新してください。トラストストアを更新することで、アプライアンスと WebSphere Portal の間の通信が可能になります。
 - a. Portal Server 個人証明書の公開鍵を抽出します。 **IKEYMAN** ユーティリティーを使用します。このユーティリティーにより `.arm` ファイルが作成されます。詳しくは、『[トラストストア・ファイル用の公開証明書の抽出](#)』を参照してください。
 - b. アプライアンスのパブリック・トラストストアをダウンロードします。詳しくは、[WebSphere](#)

[Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

- c. **iKeyman** ユーティリティを使用して、アプライアンスから抽出した `truststore.jks` ファイルを、`.arm` ファイル内の Portal Server 公開証明書で更新します。詳しくは、『[署名者証明書のインポート](#)』を参照してください。
 - d. 更新したトラストストア・ファイルをアプライアンスにアップロードします。トラストストアをアップロードした後に、「**TLS 設定の送信**」をクリックします。TLS 設定を送信し、新規トラストストアが集合内の他のアプライアンスに追加されると、集合が自動的に再始動します。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。
3. WebSphere Portal サーバーの一部のバージョンでは、Cookie が HTTP 応答に追加されるとランタイム・エラーが起こることがあります。WebSphere DataPower XC10 アプライアンスはフェイルオーバーその他の目的で Cookie を追加するので、これらの Cookie は WebSphere Portal サーバーの Cookie 無視リストに追加する必要があります。詳しくは、IBM WebSphere Portal wiki に掲載されている、複数のユーザーが共有するページのキャッシングに関するページの `cookie.ignore.regex` パラメーターのセクションを参照してください。リストに追加する必要がある 2 つの Cookie は `IBMID.*` と `IBMSessionHandle.*` です。更新されたリストは、例えば次のようになります。`"digest\\.ignore.*|LtpaToken|LtpaToken2|JSESSIONID|IBMID.*|IBMSessionHandle.*"`。詳しくは、IBM WebSphere Portal wiki の[複数のユーザーが共有するページのキャッシング \(Caching pages shared by multiple users\)](#)を参照してください。
 4. WebSphere Portal Server を再始動します。詳しくは、『[WebSphere Portal Version 7: Starting and stopping servers, deployment managers, and node agents \(WebSphere Portal バージョン 7: サーバー、デプロイメント・マネージャー、およびノード・エージェントの開始と停止\)](#)』を参照してください。

タスクの結果

WebSphere Portal Server にアクセスでき、構成されているカスタム・ポートレットの HTTP セッション・データはデータ・グリッドに保持されます。

アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアントから到達できない場合、クライアントは、代わりに WebSphere Application Server の基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとリモート・コンテナ・サーバー間のネットワークの問題
- リモート・コンテナ・サーバーのプロセスが停止した場合

sessionTableSize パラメーターによって指定される、メモリー内に保持されるセッション参照の数は、セッションが基本 Web コンテナ内に保管されている場合、そのまま維持されます。セッション数が **sessionTableSize** の値を超えると、最長未使用時間を基にセッションが Web コンテナ・セッション・キャッシュで無効化されます。リモート・データ・グリッドが使用可能になると、Web コンテナ・キャッシュで無効化されたセッションは、リモート・データ・グリッドからデータを取得し、データを新規セッションにロードできます。リモート・データ・グリッド全体が使用不可なまま、セッションがセッション・キャッシュで無効化されると、ユーザーのセッション・データは失われます。このような問題があるため、負荷の下でシステムを実行する場合、実動リモート・データ・グリッド全体をシャットダウンすることはしないでください。

親トピック: [セッション・データ・グリッドの作成](#)

関連タスク:

[各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成](#)
[HTTP コマンド・インターフェースでの管理](#)

関連情報:

☞ [IKEYMAN グラフィカル・インターフェースを使用した鍵の管理](#)

各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成

WebSphere® eXtreme Scale クライアントには、Web コンテナのデフォルト・セッション・マネージャーをオーバーライドするセッション管理実装がバンドルされています。この実装は、セッション・レプリカ生成、高可用性、より優れたスケーラビリティと構成オプションを提供します。Tomcat などさまざまなアプリケーション・サーバーに対して WebSphere eXtreme Scale クライアント セッション複製マネージャーおよび汎用組み込み ObjectGrid コンテナの開始を有効にします。

始める前に

WebSphere DataPower® XC10 アプライアンスで SSL と一緒に Transport Layer Security (TLS) を使用可能に設定することができます。詳しくは、[データ・グリッド・アプリケーション用の TLS の構成](#)を参照してください。

このタスクについて

WebSphere Application Server Community Edition などの WebSphere Application Server を実行していない他のアプリケーション・サーバーで HTTP セッション・マネージャーを使用できます。データ・グリッドを使用するように他のアプリケーション・サーバーを構成するには、アプリケーションを接合して、WebSphere eXtreme Scale クライアント Java アーカイブ (JAR) ファイルをアプリケーションに取り込む必要があります。

手順

1. アプリケーションを接合することで、アプリケーションがセッション・マネージャーを使用できるようにします。セッション・マネージャーを使用するためには、適切なフィルター宣言をアプリケーションの Web デプロイメント記述子に追加する必要があります。さらに、セッション・マネージャー構成パラメーターが、デプロイメント記述子内のサーブレット・コンテキスト初期化パラメーターという形式でセッション・マネージャーに渡されます。この情報は、以下に示す 3 とおりの方法でアプリケーションに導入することができます。

- **addObjectGridFilter** スクリプト:

WebSphere eXtreme Scale クライアントとともに提供されるコマンド行スクリプトを使用して、フィルター宣言と構成によってアプリケーションをサーブレット・コンテキスト初期化パラメーターの形式で接合します。[wxs_home/session/bin/addObjectGridFilter.sh|bat](#) スクリプトは、2 つのパラメーターを使用します。1 つは接合するエンタープライズ・アーカイブ (EAR) ファイルまたは Web アーカイブ (WAR) ファイルへの絶対パスで、もう 1 つは各種構成プロパティが含まれたスプライサー・プロパティ・ファイルへの絶対パスです。このスクリプトの使用形式は以下の通りです。

Windows

```
addObjectGridFilter.bat <ear_or_war_file> <splicer_properties_file>
```

UNIX

```
addObjectGridFilter.sh <ear_or_war_file> <splicer_properties_file>
```

UNIX

UNIX 上のスタンドアロン・ディレクトリーにインストールされている WebSphere eXtreme Scale クライアントの使用例:

- a. `cd wxs_home/session/bin`
- b. `addObjectGridFilter.sh /tmp/mySessionTest.ear
wxs_home/session/samples/splicer.properties`

接合されるサーブレット・フィルターは構成値のデフォルトを保持します。これらのデフォルト値は、2 番目の引数にあるプロパティ・ファイルで指定する構成オプションでオーバーライドできます。使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

WebSphere eXtreme Scale クライアント インストールとともに提供されるサンプルの `splicer.properties` ファイルを変更して使用することができます。ただし、ファイルを調べ、使用したいプロパティ (`catalogHostPort` プロパティなど) のコメントが外されていることを確認してください。また、各サーブレットを拡張することによってセッション・マネージャーを挿入する、**addObjectGridServlets** スクリプトも使用できます。ただし、推奨スクリプトは **addObjectGridFilter** スクリプトです。

- Ant ビルド・スクリプト:

WebSphere eXtreme Scale クライアントには Apache Ant で使用できる `build.xml` ファイルが同梱されています。このファイルは WebSphere Application Server インストールの [was_root](#)/bin フォルダに含まれています。`build.xml` ファイルを変更して、セッション・マネージャー構成プロパティを変更できます。構成プロパティは `splicer.properties` ファイル内のプロパティ名と同一です。`build.xml` ファイルの変更後に、`ant.sh`、`ws_ant.sh` (UNIX) または `ant.bat`、`ws_ant.bat` (Windows) を実行することで、Ant プロセスを呼び出します。

- 手動による Web 記述子の更新:

Web アプリケーションに同梱されている web.xml ファイルを編集して、フィルター宣言、そのサーブレット・マッピング、およびサーブレット・コンテキスト初期化パラメーターが組み込まれるようにします。この方法はエラーを起こしやすいため、使用しないようにしてください。

使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

2. WebSphere eXtreme Scale クライアント セッション・レプリカ生成マネージャーの JAR ファイルをアプリケーションに取り込みます。ファイルは、アプリケーション・モジュールの WEB-INF/lib ディレクトリーまたはアプリケーション・サーバーのクラスパスに組み込むことができます。必要な JAR ファイルは、以下のように、使用しているコンテナのタイプによって異なります。
 - リモート・コンテナ・サーバー: ogclient.jar と sessionobjectgrid.jar
 - 組み込みコンテナ・サーバー: objectgrid.jar と sessionobjectgrid.jar
3. アプリケーションをデプロイします。サーバーやクラスターに対して通常使用する手順に従ってアプリケーションをデプロイしてください。アプリケーションをデプロイした後、アプリケーションを始動することができます。
4. アプリケーションにアクセスします。これで、セッション・マネージャーおよび WebSphere eXtreme Scale クライアント と対話するアプリケーションにアクセスすることができます。

次のタスク

アプリケーションの装備時にセッション・マネージャーの構成属性の大多数を変更して、セッション・マネージャーを使用するようにすることができます。これらの属性には、レプリカ生成タイプ (同期または非同期) のバリエーション、メモリー内セッション・テーブルのサイズなどがあります。アプリケーションの装備時に変更できる属性を別にすれば、アプリケーションのデプロイメント後に変更できるその他の構成属性は、WebSphere eXtreme Scale サーバー・クラスター・トポロジーと、それらのクラスターのクライアント (セッション・マネージャー) がそれらのクラスターに接続する方法に関する属性のみです。

リモート・シナリオの動作: アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアントから到達できない場合、クライアントは、代わりにアプリケーション・サーバーの基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとリモート・コンテナ・サーバー間のネットワークの問題
- リモート・コンテナ・サーバーのプロセスが停止した場合

sessionTableSize パラメーターによって指定される、メモリー内に保持されるセッション参照の数は、セッションが基本 Web コンテナ内に保管されている場合、そのまま維持されます。セッション数が **sessionTableSize** の値を超えると、最長未使用時間を基にセッションが Web コンテナ・セッション・キャッシュで無効化されます。リモート・データ・グリッドが使用可能になると、Web コンテナ・キャッシュで無効化されたセッションは、リモート・データ・グリッドからデータを取得し、データを新規セッションにロードできます。リモート・データ・グリッド全体が使用不可なまま、セッションがセッション・キャッシュで無効化されると、ユーザー・セッション・データは失われます。このような問題があるため、負荷の下でシステムを実行する場合、実動リモート・データ・グリッド全体をシャットダウンすることはしないでください。

親トピック: [セッション・データ・グリッドの作成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連タスク:

[WebSphere Portal での HTTP セッション・マネージャーの構成](#)

[HTTP コマンド・インターフェースでの管理](#)

関連資料:

[splicer.properties ファイル](#)

関連情報:

☞ [コンソールを使用したエンタープライズ・アプリケーション・ファイルのインストール](#)

☞ [wsadmin スクリプト](#)

☞ [IKEYMAN グラフィカル・インターフェースを使用した鍵の管理](#)

ASP.NET セッション状態ストア・プロバイダーの構成

2.5+ データ・グリッド内の ASP.NET アプリケーションからセッション状態を保管できます。

始める前に

- WebSphere eXtreme Scale Client for .NET をインストールします。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストールの概要](#)を参照してください。
- ご使用の .NET 環境が、システム要件を満たしている必要があります。詳しくは、[Microsoft .NET に関する考慮事項](#)を参照してください。
- ご使用の .NET アプリケーションがセッション状態を維持するように構成されている必要があります。

2.5+ [ASP.NET セッション状態ストア・プロバイダーで使用するデータ・グリッドの作成](#)

ASP.NET アプリケーションのセッション状態を保存するためのデータ・グリッドの作成

2.5+ [ASP.NET セッション状態ストア・プロバイダーを使用する .NET アプリケーションの構成](#)

ASP.NET セッション状態ストア・プロバイダーを構成するには、web.config ファイルを更新して、ASP.NET アプリケーションにセッション状態ストア・プロバイダーとその構成を組み込む必要があります。

親トピック: [セッション・データ・グリッドの作成](#)

ASP.NET セッション状態ストア・プロバイダーで使用するデータ・グリッドの作成

2.5+ ASP.NET アプリケーションのセッション状態を保存するためのデータ・グリッドの作成

手順

ASP.NET セッション状態ストア・プロバイダーにデータ・グリッドを作成します。ユーザー・インターフェースで、「データ・グリッド」 > 「セッション」をクリックします。「追加」アイコン (+) をクリックして、作成するセッション・データ・グリッドの名前を指定します。デフォルトでは、ASP.NET セッション状態ストア・プロバイダーは、session というデータ・グリッドと連動するように構成されます。session 以外の名前を持つデータ・グリッドを作成した場合は、ASP.NET セッション状態ストア・プロバイダーを、作成したデータ・グリッドの名前で構成する必要があります。

次のタスク

データ・グリッドを使用するために、Web アプリケーションの構成を更新してください。詳しくは、[ASP.NET セッション状態ストア・プロバイダーを使用する .NET アプリケーションの構成](#)を参照してください。

親トピック: **.NET** **2.5+** [ASP.NET セッション状態ストア・プロバイダーの構成](#)

ASP.NET セッション状態ストア・プロバイダーを使用する .NET アプリケーションの構成

2.5+ ASP.NET セッション状態ストア・プロバイダーを構成するには、web.config ファイルを更新して、ASP.NET アプリケーションにセッション状態ストア・プロバイダーとその構成を組み込む必要があります。

始める前に

- ASP.NET HTTP セッション状態を保管するデータ・グリッドを構成します。詳しくは、[ASP.NET セッション状態ストア・プロバイダーで使用するデータ・グリッドの作成](#)を参照してください。
- カタログ・サーバーのホストおよびポートが必要です。アプライアンス・ユーザー・インターフェースでカタログ・サーバーのホストとポートを取得するには、「集合」>「メンバー」をクリックします。集合メンバーを選択します。カタログ・サーバーの IP とポート番号が表示されます。

手順

1. ASP.NET アプリケーションの web.config ファイルを、ASP.NET セッション状態ストア・プロバイダーの設定で更新します。web.config ファイルに、次の例の太字体のテキストを更新または追加する必要があります。

```
<system.web>
  ...
  <sessionState
    mode="Custom"
    customProvider="WxsSessionStateStoreProvider">
    <providers>
      <add
        name="WxsSessionStateStoreProvider"
        type="IBM.WebSphere.Caching.SessionStateStore.WxsSessionStateStore,
          IBM.WebSphere.Caching,
          Version=8.6.0.2, Culture=neutral,
          PublicKeyToken=b439a24ee43b0816"
        wxsPropertyFile="¥optional¥path¥to¥NET-client.properties"
        wxsHostAndPort="optionalHostAndPort"
        wxsGridName="session"
        wxsMapName="ASPNET.SessionState"
      />
    </providers>
  </sessionState>
  ...
</system.web>
```

wxsPropertyFile (オプション)

プロバイダーが Connect API によるデータ・グリッドへの接続時に使用する完全修飾プロパティ・ファイルを指定します。この属性が指定されなかったか、空文字列である場合、プロバイダーは Web アプリケーション・プロセスの現行ランタイム・ディレクトリーで Client.Net.properties ファイルを探します。このファイルがランタイム・ディレクトリーで見つからない場合、プロバイダーは [net_client_home](#)\config ディレクトリーで Client.Net.properties ファイルを探します。

wxsHostAndPort

セッション状態ストア・プロバイダーがデータ・グリッドにアクセスするときに接続する、カタログ・サーバーのホストとポートのペアのコンマ区切りリストを指定します。形式は次のとおりです。

```
<host name or host ip address>:<tcp port>[,<host name or host ip address>:<tcp port>]
```

wxsGridName (オプション)

ASP.NET セッション・ストア・プロバイダーの接続先データ・グリッドの名前を指定します。ASP.NET セッション状態のデータ・グリッドを作成した場合は、作成したデータ・グリッド名を指定してください。値を指定しない場合、プロバイダーは session データ・グリッドに接続します。

wxsMapName (オプション)

プロバイダーの接続先マップを指定します。値を指定しない場合、プロバイダーは ASPNET.SessionState マップに接続します。

2. ターゲット Web アプリケーションを再始動します。IIS がプロバイダーをロードできるように、Web アプリケ

セッションは、再始動する必要があります。多くの場合、web.config ファイルが変更されて、現行 HTTP 要求の処理が完了すると、自動的に再始動が行われます。

タスクの結果

ASP.NET アプリケーションからの ASP.NET セッション状態がデータ・グリッドに保管されます。

親トピック: [.NET 2.5+ ASP.NET セッション状態ストア・プロバイダーの構成](#)

HTTP セッション・サイズの表示

2.5+ **xscmd** ユーティリティを使用して、Java アプリケーションでのセッションのサイズを表示することができます。セッション・サイズを表示することで、データ・グリッドの問題のトラブルシューティングが容易になります。

始める前に

- カタログ・サーバーに接続するためには、**xscmd** ユーティリティがセットアップされていなければなりません。詳しくは、[xscmd ユーティリティによる管理](#)を参照してください。
- セッション・サイズを表示するためには、そのセッションのセッション ID とセッション・ハンドルが必要です。

手順

xscmd -c showSessionSize コマンドを実行します。コマンドの構文は以下のとおりです。

```
xscmd.bat|sh -c showSessionSize -cep hostname:port(,hostname:port)
-sid session_ID -sh session_handle -g grid_name
[-wacr web_application_context_root]
```

例えば、以下のようなコマンドを実行できます。

```
xscmd.bat -c showSessionSize -g session -cep 9.42.139.213:2809 -sh c -sid
LFMzQnDX5k87xztMF3ri6jU -wacr /A -user xcadmin -pwd xcadmin
```

コマンドが完了すると、以下のセッション情報が表示されます。

```
*** Session Id: LFMzQnDX5k87xztMF3ri6jU
Number of session attributes: 2
Total size of session metadata: 488 bytes
Total size of session attributes: 243 bytes
Total size of session: 731 bytes

Metadata Key and Value:
Key: LFMzQnDX5k87xztMF3ri6jU/A
Size of key: 88 bytes
Size of value: 400 bytes

Individual Attribute Key and Value:
Key: LFMzQnDX5k87xztMF3ri6jU/A_session.reqCount
Size of key: 128 bytes
Size of value: 2 bytes
Key: LFMzQnDX5k87xztMF3ri6jU/A_user
Size of key: 104 bytes
Size of value: 9 bytes
```

親トピック: [セッション・データ・グリッドの作成](#)

サブレット・コンテキスト初期化パラメーター

以下に示すサブレット・コンテキスト初期化パラメーターのリストは、選択した接続メソッドに必要なスプライサー・プロパティ・ファイルに指定できるものです。

パラメーター

applicationQualifiedCookies

true または false のいずれかのストリング値。ご使用の環境に、固有の Cookie 名を使用するアプリケーションが複数ある場合は、true に設定します。デフォルトは false で、すべてのアプリケーションが同じ Cookie 名を使用していることが前提となっています。

authenticationRetryCount

資格情報の有効期限が切れている場合の認証の再試行カウントを指定します。値が 0 に設定されていると、認証の再試行は行われません。

catalogHostPort

カタログ・サーバーに接続して、クライアント・サイドの ObjectGrid インスタンスを取得できます。値の形式は、host:port<,host:port> でなければなりません。host は、カタログ・サーバーが実行されているリスナー・ホストです。port は、そのカタログ・サーバー・プロセスのリスナー・ポートです。このリストは任意の長さにすることができ、ブートストラッピングにのみ使用されます。最初の実行可能なアドレスが使用されます。このパラメーターは、**catalog.services.cluster** プロパティが構成されている場合は、WebSphere® Application Server 内でオプションです。

credentialAuthentication

クライアントの資格情報認証のサポートを指定します。指定できる値は以下のとおりです。

- Never- クライアントは資格情報認証をサポートしません。
- Supported - サーバーが資格情報認証をサポートする場合のみ、クライアントも資格情報認証をサポートします。
- Required - クライアントは資格情報認証を必要とします。デフォルト値は Supported です。

cookieDomain

ホストをまたいだセッションへのアクセスが必要かどうかを指定します。値を、ホスト間に共通のドメインの名前に設定してください。

cookiePath

com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator インターフェースを実装するクラスの名前を指定します。このクラスを使用して、クライアントの資格情報が取得されます。

credentialGeneratorClass

com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator インターフェースを実装するクラスの名前。このクラスを使用して、クライアントの資格情報が取得されます。

credentialGeneratorProps

CredentialGenerator 実装クラスのプロパティ。このプロパティが、setProperty(String) メソッドを使用してオブジェクトに設定されます。credentialGeneratorProps 値は、**credentialGeneratorClass** プロパティの値が非ヌルの場合にのみ使用されます。

enableSessionStats

true または false のいずれかのストリング値。eXtreme Scale クライアント HTTP セッションの統計追跡を使用可能にします。

fragmentedSession

true または false のいずれかのストリング値。デフォルト値は true です。この設定を使用して、製品がセッション・データをエントリ全体として保管するか、それぞれの属性を個別に保管するかを制御します。

Web アプリケーションのセッションが持っている属性の数が多い場合や属性のサイズが大きい場合は、fragmentedSession パラメーターを true に設定してください。すべての属性はデータ・グリッド内の同じキーに保管されるため、セッションが持っている属性の数が少ない場合は fragmentedSession を false に設定してください。

以前のフィルター・ベースの実装環境では、このプロパティは「persistenceMechanism」と呼ばれており、設定可能な値は ObjectGridStore (フラグメント化されている場合) および ObjectGridAtomicSessionStore (フラグメント化されていない場合) でした。

objectGridType

REMOTE または EMBEDDED のいずれかのストリング値。デフォルトは REMOTE です。

このパラメーターが REMOTE に設定されている場合、セッション・データは Web アプリケーションが実行されているサーバーの外に保管されます。

このパラメーターが EMBEDDED に設定されている場合は、Web アプリケーションが実行されているアプリケーション・サーバー・プロセス内で組み込みの eXtreme Scale コンテナが開始されます。

objectGridName

特定の Web アプリケーションで使用する ObjectGrid インスタンスの名前を定義するストリング値。デフォルトの名前は「session」です。

eXtreme Scale コンテナ・サーバーの始動に使用する ObjectGrid XML ファイルとデプロイメント XML ファイルの両方で、このプロパティは objectGridName を反映する必要があります。

objectGridXML

objectgrid.xml ファイルが置かれているロケーション。objectGridType=EMBEDDED および **objectGridXML** プロパティが指定されていない場合、eXtreme Scale ライブラリーにパッケージされた組み込み XML ファイルが自動的にロードされます。

objectGridDeploymentXML

objectGrid デプロイメント・ポリシーの XML ファイルが置かれているロケーションを指定します。objectGridType=EMBEDDED および **objectGridDeploymentXML** プロパティが指定されていない場合、eXtreme Scale ライブラリーにパッケージされた組み込み XML ファイルが自動的にロードされます。

replicationInterval

更新されたセッションを ObjectGrid に書き込む時間間隔を秒数で定義する整数値。デフォルトは 10 秒です。0 から 60 までの値を設定できます。0 の場合は、更新されたセッションを ObjectGrid に書き込むのは、各要求のサーブレット・サービス・メソッド呼び出しの最後ということになります。**replicationInterval** 値が高いほどパフォーマンスは向上します。これは、データ・グリッドに書き込まれるアップデートの数が少ないためです。ただし、値が高いとそれだけ構成のフォールト・トレラント性が低くなります。

この設定は、objectGridType が REMOTE に設定されている場合のみ適用されます。

reuseSessionID

true または false のいずれかのストリング値。デフォルトは false です。基礎になっている Web コンテナが異なるホストへの要求でセッション ID を再利用する場合は、true に設定します。このプロパティの値は、Web コンテナの中での値と同じでなければなりません。WebSphere Application Server を使用していて、管理コンソールまたは **wsadmin** ツール・スクリプトを使用して eXtreme Scale HTTP セッション・パーシスタンスを構成している場合、デフォルトで、Web コンテナ・カスタム・プロパティ `HttpSessionIdReuse=true` が追加されます。**reuseSessionID** も true に設定されます。セッション ID の再利用を望まない場合は、eXtreme Scale セッション・パーシスタンスを構成する前に、Web コンテナ・カスタム・プロパティに `HttpSessionIdReuse=false` カスタム・プロパティを設定します。

sessionIdOverrideClass

`com.ibm.websphere.xs.sessionmanager.SessionIDOverride` インターフェースを実装するクラスの名前。このクラスは、`HttpSession.getId()` メソッドで取得した固有のセッション ID をオーバーライドして、すべてのアプリケーションが同じ ID を持つようにするために使用されます。デフォルトでは、`HttpSession.getId()` から得られたユーザー ID が使用されます。

sessionStatsSpec = session.all = enabled

eXtreme Scale クライアント HTTP 統計仕様のストリング。

shareSessionsAcrossWebApps

true または false のいずれかのストリング値。デフォルトは false です。セッションが Web アプリケーション間で共有されるかどうかを指定します。true または false のいずれかのストリング値として指定されます。サーブレット仕様では、HTTP セッションを Web アプリケーション間で共有できないとされています。この共有を可能にするため、サーブレット仕様の拡張が提供されます。

sessionTableSize

メモリー内に保持するセッション参照数を定義する整数値。デフォルトは 1000 です。

EMBEDDED トポロジーは既に Web コンテナと同じ層にセッション・データを持っているため、この設定は REMOTE トポロジーのみに関するものです。

セッションは、最長未使用時間 (LRU) ロジックに基づいて、メモリー内のテーブルから除去されます。メモリー内のテーブルから除去されたセッションは Web コンテナから無効化されます。しかし、データ自体はグリッドから削除されないため、そのセッションに対する後続の要求は引き続きデータを検索することができます。この値は、Web コンテナの最大スレッド・プール値よりも高く設定されなければなりません。そうすると、セッション・キャッシュでの競合を削減できます。

securityEnabled

true または false のいずれかのストリング値。デフォルト値は false です。この設定により、eXtreme Scale クライアント・セキュリティーを使用可能にすることができます。この設定は、eXtreme Scale サーバー・プロパティ・ファイルの **securityEnabled** 設定と一致していなければなりません。これらの設定が一致しないと、例外が発生します。

sessionIdOverrideClass

取得されたアプリケーションのセッション ID をオーバーライドします。デフォルトでは、HttpSession.getId() メソッドから得られた ID が使用されます。eXtreme Scale クライアント HTTP セッションがアプリケーションの固有セッション ID をオーバーライドできるようにします。これにより、すべてのアプリケーションが同じ ID で検索されるようになります。com.ibm.websphere.xs.sessionmanager.SessionIDOverride インターフェースを実装するために設定します。このインターフェースは、HttpServletRequest オブジェクトに基づいて HttpSession ID を決定します。

traceSpec

ストリング値として設定される、IBM® WebSphere のトレース仕様を指定します。この設定は、WebSphere Application Server以外のアプリケーション・サーバーに使用してください。

traceFile

トレース・ファイルの場所をストリング値として指定します。この設定は、WebSphere Application Server以外のアプリケーション・サーバーに使用してください。

useURLEncoding

true または false のいずれかのストリング値。デフォルトは false です。URL 再書き込みを使用可能にする場合は、true に設定します。デフォルト値は false です。これは、セッション・データの保管に Cookie が使用されることを示します。このパラメーターの値は、セッション管理の Web コンテナ設定と同じでなければなりません。

useCookies

true または false のいずれかのストリング値。基礎になっている Web コンテナがさまざまなホストへの要求でセッション ID を再利用する場合は、true に設定します。デフォルトは false です。このプロパティの値は、Web コンテナで設定される値と同じでなければなりません。

親トピック: [セッション・データ・グリッドの作成](#)

splicer.properties ファイル

splicer.properties ファイルには、サブレット・フィルター・ベースのセッション・マネージャーを構成するための、すべての構成オプションが含まれます。

サンプル・スプライサー・プロパティ

このファイルに説明されている追加プロパティを使用する場合は、有効にするプロパティの行のコメントを外してください。

```
# サブレット・フィルター・ベース ObjectGrid
# セッション・マネージャーが使用するように構成できる、
# すべての構成オプションが含まれたプロパティ・ファイル。
#
# このプロパティ・ファイルで、これらの構成設定に割り当てるすべての
# デフォルト値を保持できます。また、このプロパティ・ファイルを
# filtersplicer ANT タスクと組み合わせて使用するには、ANT タスク・
# プロパティを使用して、個々の設定をオーバーライドできます。

# スtring値「REMOTE」または「EMBEDDED」。デフォルトは REMOTE です。
# If it is set to "REMOTE", the session data will be stored outside of
# the server on which the web application is running. If it is set to
# "EMBEDDED", an embedded WebSphere eXtreme Scale container will start
# in the application server process on which the web application is running.

objectGridType = REMOTE

# A string value that defines the name of the ObjectGrid
# instance used for a particular web application. The default name
# is session. This property must reflect the objectGridName in both
# the objectgrid xml and deployment xml files used to start the eXtreme
# Scale containers.

objectGridName = session

# カタログ・サーバーに接続して、クライアント・サイドの ObjectGrid
# インスタンスを取得できます。The value needs to be of the
# form "host:port<,host:port>", where the host is the listener host
# on which the catalog server is running, and the port is the listener
# port for that catalog server process.
# このリストは任意の長さにするのができ、ブートストラッピングにのみ使用
# されます。最初の実行可能なアドレスが使用されます。It is optional inside WebSphere
# if the catalog.services.cluster property is configured.

# catalogHostPort = host:port<,host:port>

# An integer value (in seconds) that defines the time in seconds between
# writing of updated sessions to ObjectGrid. The default is 10. This property
# is only used when objectGridType is set to REMOTE. Possible values are
# from 0 to 60. 0 means updated sessions are written to the ObjectGrid
# at the end of servlet service method call for each request.

replicationInterval = 10

# メモリー内に保持するセッション参照数を定義する整数値。
# デフォルトは 1000 です。This property is only used when
# objectGridType is set to REMOTE. When the number of sessions stored
# in memory in the web container exceeds this value, the least recently
# accessed session is invalidated from the web container. If a request
# comes in for that session after it's been invalidated, a new session
# will be created (with a new session ID if reuseSessionId=false),
# populated with the invalidated session's attributes. This value should
# always be set to be higher than the maximum size of the web container
# thread pool to avoid contention on this session cache.

sessionTableSize = 1000

# A string value of either "true" or "false", default is "true".
# It is to control whether we store session data as a whole entry
```



```
# or store each attribute separately.
# This property was referred to as persistenceMechanism in the
# previous filter-based implementation, with the possible values
# of ObjectGridStore (fragmented) and ObjectGridAtomicSessionStore
# (not fragmented).

fragmentedSession = true

# A string value of either "true" or "false", default is "false".
# Enables eXtreme Scale client security. This setting needs to match
# the securityEnabled setting in the eXtreme Scale server properties
# file. これらの設定が一致しないと、例外が発生します。

securityEnabled = false

# Specifies the client credential authentication support.
# The possible values are:
# Never - The client does not support credential authentication.
# Supported* - The client supports the credential authentication if and only if the
server
# supports too.
# Required - The client requires the credential authentication.
# The default value is Supported.

# credentialAuthentication =

# Specifies the retry count for authentication if the credential
# is expired. If the value is set to 0, there will not be
# any authentication retry.

# authenticationRetryCount =

# Specifies the name of the class that implements the
# com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator
# interface. This class is used to get credentials for clients.

# credentialGeneratorClass =

# Specifies the properties for the CredentialGenerator implementation
# class. The properties are set to the object with the setProperties(String)
# method. The credentialGeneratorProps value is used only if the value of the
# credentialGeneratorClass property is not null.

# credentialGeneratorProps =

# The file location of the objectgrid xml file.
# The built-in xml file packaged in the eXtreme Scale library
# will automatically be loaded if this property
# is not specified and if objectGridType=EMBEDDED

# objectGridXML =

# The file location of the objectGrid deployment policy xml file.
# The built-in xml file packaged in the eXtreme Scale library
# will automatically be loaded if this property
# is not specified and if objectGridType=EMBEDDED

# objectGridDeploymentXML =

# IBM WebSphere トレース仕様を示すストリング。
# useful for all other application servers besides WebSphere.

# traceSpec =

# トレース・ファイルの場所を示すストリング。
# useful for all other application servers besides WebSphere.

# traceFile=
```

```
# This property should be set if you require sessions to be
# accessible across hosts. The value will be the name of the
# common domain between the hosts.

# cookieDomain=

# This property should be set to the same path you have configured
# for your application server cookie settings. The default path
# is /.

# cookiePath

# Set to true if the underlying web container will reuse
# session ID's across requests to different hosts. デフォルト
# is false. The value of this should be the same as what is
# set in the web container.

# reuseSessionId=

# A string value of either "true" or "false", the default is
# "false". Per the servlet specification, HTTP Sessions cannot
# be shared across web applications. An extension to the servlet
# specification is provided to allow this sharing.

# shareSessionsAcrossWebApps = false

# A string value of either "true" or "false", default is "false".
# Set to true if you want to enable urlRewriting. Default is
# false. The value of this should reflect what is set in the
# web container settings for session management.

# useURLEncoding = false

# Set to false if you want to disable cookies as the session tracking
# mechanism. デフォルトは true です。The value of this should reflect what
# is set in the web container settings for session management.

# useCookies = true

# A string value of either "true" or "false", the default is "false".
# Enables eXtreme Scale client HTTP Sessions statistics tracking.

# enableSessionStats = false

# Overrides the retrieved session ID of an application.
# The default is to use the ID derived from the HttpSession.getId() method.
# Enables eXtreme Scale client HTTP Sessions to override
# the unique session ID of an application so that all applications are retrieved
# with the same ID.
# Set to the implementation of the
# com.ibm.websphere.xs.sessionmanager.SessionIDOverride interface.
# This interface determines the HttpSession ID based on
# the HttpServletRequest object.

# sessionIdOverrideClass = # A string of eXtreme Scale client HTTP statistics
specification.

# sessionStatsSpec = session.all = enabled

# Set to true if your environment contains multiple applications that
# use unique cookie names. Default is false, which assumes all applications
# are using the same cookie name.

# applicationQualifiedCookies=false
```

親トピック: [セッション・データ・グリッドの作成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パーシスタンスの構成](#)

[HTTP コマンド・インターフェースでの管理](#)

[各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成](#)

関連情報:

☞ [コンソールを使用したエンタープライズ・アプリケーション・ファイルのインストール](#)

☞ [wsadmin スクリプト](#)

動的キャッシュの作成データ・グリッド

IBM® WebSphere® DataPower® XC10 アプライアンス を使用して、WebSphere Application Server 動的キャッシュ・インスタンスのデータを保管することができます。この機能をセットアップすることにより、WebSphere Application Server 動的キャッシュ・インスタンスを使用するよう作成されたアプリケーションでアプライアンスのフィーチャーおよびパフォーマンス機能を活用できるようにします。

始める前に

- WebSphere eXtreme Scale クライアントを WebSphere Application Server 構成にインストールする必要があります。詳しくは、[WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- アプライアンスの Transport Layer Security (TLS) が有効であるか、クライアントが必ず Transport Layer Security (TLS) を使用するようにしたい場合、WebSphere Application Server 管理コンソールでグローバル・セキュリティーを有効にする必要もあります。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

このタスクについて

WebSphere Application Server 内の動的キャッシュ・プロバイダーを構成して DataPower XC10 アプライアンス を使用する場合、キャッシュ・データは WebSphere Application Server のトポロジーの外に保管されます。キャッシュ・データはすべてアプライアンス内に保管されます。ご使用のアプリケーション・サーバー内でキャッシングに使用される メモリーを他の目的のためにも使用できます。

IBM WebSphere Commerce での WebSphere eXtreme Scale クライアント および DataPower XC10 アプライアンス 動的キャッシュ・データ・グリッドの使用については、IBM WebSphere Commerce 資料の以下のトピックを参照してください。

- [Enabling the dynamic cache service and servlet caching](#)
- [Enabling WebSphere Commerce data cache](#)

手順

1. DataPower XC10 アプライアンスで構成するキャッシュ・インスタンスを作成します。詳しくは、[動的キャッシュ・インスタンスの構成](#)を参照してください。
2. カタログ・サービスを構成します。カタログ・サービスにより、WebSphere Application Server 動的キャッシュ構成で DataPower XC10 アプライアンスと通信できます。カタログ・サービス・ドメインを作成することで、WebSphere Application Server 管理コンソールでカタログ・サービスを構成できます。詳しくは、[WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)を参照してください。
3. DataPower XC10 アプライアンス上にデータ・グリッドを作成して、必要なセキュリティー設定を構成します。dynaCfgToAppliance スクリプトを実行することも、構成を手動で作成することもできます。

dynaCfgToAppliance スクリプトを使用したデータ・グリッド構成の作成:

このスクリプトは、WebSphere eXtreme Scale クライアントのインストール時にデプロイメント・マネージャー・プロファイルの bin ディレクトリーにインストールされます。このスクリプトを実行する前に、アプライアンスとデプロイメント・マネージャーの両方が稼働中であることを確認してください。

```
dynaCfgToAppliance <IP_address> <cache_jndi_name> <appliance_admin> <admin_pw>  
<SOAP_port> <soap.client.props>
```

IP_address

動的キャッシュ・データを保管する DataPower XC10 アプライアンスの IP アドレスを指定します。

cache_jndi_name

動的キャッシュの JNDI 名を指定します。動的キャッシュの JNDI 名の中にスラッシュ (/) があると、DataPower XC10 アプライアンスのデータ・グリッドの名前では、そのスラッシュはダッシュに変換されます。例えば、動的キャッシュ名が services/cache1 であれば、アプライアンスで作成されたデータ・グリッドは、services-cache1 という名前になります。DataPower XC10 アプライアンスでは、データ・グリッドの名前の中に ^ . ¥¥ / , # \$ @ : ; ¥ * ? < > | = + & % [] " " 文字は使用できません。

appliance_admin

DataPower XC10 アプライアンス のユーザー・インターフェース にログインするための管理者 ID を指定します。

admin_pw

DataPower XC10 アプライアンス のユーザー・インターフェースにログインするための管理者パスワードを指定します。

SOAP_port

(オプション) デフォルトの 8879 以外のポート番号を使用する場合は、デプロイメント・マネージャーの SOAP ポートを指定します。

soap.client.props

(オプション) soap.client.props ファイルのパスを指定します。 WebSphere Application Server でセキュリティを有効にした場合は、このファイルを指定する必要があります。このファイルは SOAP セキュリティを有効にし、WebSphere Application Server デプロイメント・マネージャーを管理するためのユーザー名とパスワードを指定します。

```
com.ibm.SOAP.securityEnabled=true
com.ibm.SOAP.loginUserId=
com.ibm.SOAP.loginPassword=
```

soap.client.props ファイルについて詳しくは、[SOAP コネクタおよびプロセス間通信コネクタのプロパティ・ファイル \(SOAP connector and Inter-Process Communications connector properties files\)](#) を参照してください。

このスクリプトは、アプライアンス上にデータ・グリッドを作成します。また、スクリプトは、**appliance_admin** パラメーターおよび **admin_pw** パラメーターで指定した DataPower XC10 アプライアンス固有の ID およびパスワードの資格情報を以下のカスタム・プロパティを使用して設定します。

- xc10.<data_grid_name>.userid
- xc10.<data_grid_name>.password

これらのプロパティ名はケース・センシティブではありません。パスワードの値はエンコードされます。初期構成後にスクリプトを再実行すると、カスタム・プロパティが更新されます。

データ・グリッド構成の手動作成:

- DataPower XC10 アプライアンスのユーザー・インターフェースで動的キャッシュ・データ・グリッドを作成します。「データ・グリッド」 > 「動的キャッシュ」をクリックします。キャッシュの名前は、WebSphere Application Server 構成の動的キャッシュの JNDI 名と一致している必要があります。JNDI 名を入力する際、DataPower XC10 アプライアンスのデータ・グリッドの名前で、スラッシュはすべて (/) ダッシュに置換します。例えば、動的キャッシュ名が services/cache1 であれば、名前を services-cache1 として入力します。
 - WebSphere Application Server セルで、xc10.<data_grid_name>.userid および xc10.<data_grid_name>.password の各カスタム・プロパティを作成します。各カスタム・プロパティの <data_grid_name> の値は、スラッシュ (/) がダッシュと置き換えられた、データ・グリッドの JNDI 名です。例えば、前の例で、カスタム・プロパティ名は xc10.services-cache1.userid と xc10.services-cache1.password です。値は、DataPower XC10 アプライアンス構成内のデータ・グリッドにアクセスできるユーザー ID とパスワードに設定する必要があります。encodePassword スクリプト (デプロイメント・マネージャーの bin ディレクトリーにあります) を使用して、パスワードをエンコードできます。
4. **2.5+** オプション: DataPower XC10 アプライアンス 動的キャッシュ・プロバイダーでマルチマスター複製を使用可能に設定する選択ができます。詳しくは、[集合間のマルチマスター複製の構成](#)を参照してください。

注: WebSphere Portal Server または WebSphere Commerce Server の動的キャッシュ・グリッド・ユーザーは、複数のキャッシュ・インスタンスをそれぞれの WebSphere Application Server 構成内に定義している可能性があります。DataPower XC10 アプライアンス でマルチマスター複製を使用可能に設定すると、この構成は、動的キャッシュ・プロバイダーとして DataPower XC10 アプライアンス を使用するように定義されているキャッシュ・インスタンスのみに影響し、デフォルトの WebSphere Application Server 動的キャッシュ・プロバイダーを使用するように定義されているキャッシュ・インスタンスに影響しません。

タスクの結果

動的キャッシュ・サービスでアプライアンスをキャッシュ・プロバイダーとして使用するように構成することで、アプリケーション・サーバーが使用するメモリーが減少します。すべてのキャッシュ・データがアプライアンスにオフロードされ、アプリケーション・サーバーのメモリーにはもう保管されません。

次のタスク

- データ・グリッドへのデータの送信を開始する前に、セキュリティを構成します。詳しくは、[データ・グリッドのセキュリティを使用可能に設定する](#)を参照してください。
- レプリカを構成します。レプリカにより、基本コピーで障害が発生した場合にもデータ・グリッドのデータが使用可能になります。レプリカを構成するには、「データ・グリッド」 > 「動的キャッシュ」 > 「拡張属性の表示」をクリックします。アプライアンスが集合内にある場合のみ、レプリカが作成されます。集合内のアプライアンスの数が n の場合、レプリカの最大数は $n-1$ になります。したがって、3 つのレプリカを構成するが、集合内にあるアプライアンスが 2 つのみである場合、作成されるレプリカは 1 つのみです。集合にアプライアンスを追加すると、追加のレプリカが作成されます。アプライアンスが集合に追加されたときに新規レプリカを作成できるようにするため、レプリカの数を適切な数に設定してください。レプリカの数を編集すると、データ・グリッドの内容が消去されます。
- 動的キャッシュ・インスタンスのグリッド容量限度は、WebSphere Application Server 動的キャッシュ・インスタンスの作成時に構成されます。このサイズを超えた場合は、項目数を構成された限度内に保つために最長未使用時間 (LRU) 除去ストラテジーが使用されます。
- DataPower XC10 アプライアンス ユーザー・インターフェース 内の動的キャッシュデータ・グリッド (data

grid)はモニターできます。詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

Java 2.5+ [動的キャッシュ・インスタンスの構成](#)

WebSphere Application Server の動的キャッシュ・サービスは、デフォルト・キャッシュ・インスタンス (baseCache) と追加のサブレット・キャッシュ・インスタンスおよびオブジェクト・キャッシュ・インスタンスの両方の作成をサポートします。

Java 2.5+ [動的キャッシュ・プロバイダーの概要](#)

WebSphere Application Server は、デプロイされた Java EE アプリケーションが使用できる動的キャッシュ・サービスを提供します。このサービスは、サブレット、JSP、またはコマンドからの出力などのデータ、および DistributedMap API を使用してエンタープライズ・アプリケーション内にプログラマチックに指定されているオブジェクト・データをキャッシュするために使用されます。この機能をセットアップすることにより、動的キャッシュ・サービスを使用するアプリケーションが WebSphere DataPower XC10 アプライアンス のフィーチャーおよびパフォーマンス機能を使用できるようになります。

Java [WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)

WebSphere DataPower XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

Java 2.5+ [動的キャッシュのニア・キャッシュの構成](#)

アプライアンスまたは集合で、動的キャッシュ・データ・グリッドと一緒に動作するローカルのニア・キャッシュを構成することができます。ニア・キャッシュは、ローカル Java 仮想マシン (JVM) リソースを使用します。一般に、ニア・キャッシュには、アプライアンスの動的キャッシュ・データ・グリッドにあるデータのサブセットが保持されます。

親トピック: [データ・グリッドの構成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

動的キャッシュ・インスタンスの構成

WebSphere® Application Server の動的キャッシュ・サービスは、デフォルト・キャッシュ・インスタンス (baseCache) と追加のサーブレット・キャッシュ・インスタンスおよびオブジェクト・キャッシュ・インスタンスの両方の作成をサポートします。

このタスクについて

デフォルト・キャッシュ・インスタンス (baseCache) は、当初は WebSphere Application Server によってサポートされる唯一の動的キャッシュ・インスタンスでしたが、現在は、WebSphere Commerce Suite によって使用される、すぐに使用可能な動的キャッシュ・インスタンスです。追加のサーブレット・キャッシュ・インスタンスおよびオブジェクト・キャッシュ・インスタンスは、WebSphere Application Server の後継リリースで追加されたもので、WebSphere Application Server 管理コンソールの独立した「キャッシュ・インスタンス」セクションで構成されます。

Java 2.5+ [デフォルト動的キャッシュ・インスタンス \(baseCache\) の構成](#)

デフォルト動的キャッシュ・インスタンス (baseCache と呼ばれる) は、WebSphere Application Server 動的キャッシュ・サービスによって作成された動的キャッシュ・インスタンスのデフォルトです。このサーブレット動的キャッシュ・インスタンスは、IBM WebSphere Commerce などの製品によって使用されます。baseCache は、WebSphere Application Server で定義される他のキャッシュ・インスタンスとは異なり、ただ 1 つのサーバー・インスタンスまたはクラスター・インスタンスに固有のものです。下記の手順を使用して、WebSphere eXtreme Scale と一緒に動的キャッシュ・プロバイダーとして使用するよう WebSphere Application Server で baseCache インスタンスを構成します。

Java 2.5+ [オブジェクトまたはサーブレットの動的キャッシュ・インスタンスの構成](#)

WebSphere Application Server では、デフォルト・インスタンスのほかに、複数のオブジェクト動的キャッシュ・インスタンスやサーブレット動的キャッシュ・インスタンスを構成することができます。下記の手順を使用して、追加のオブジェクト・キャッシュ・インスタンスやサーブレット・キャッシュ・インスタンスを構成します。

Java 2.5+ [カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)

WebSphere Application Server により、動的キャッシュ・インスタンスに対してカスタム・プロパティを設定することができます。

親トピック: **Java** [動的キャッシュの作成データ・グリッド](#)

デフォルト動的キャッシュ・インスタンス (baseCache) の構成

デフォルト動的キャッシュ・インスタンス (baseCache と呼ばれる) は、WebSphere® Application Server 動的キャッシュ・サービスによって作成された動的キャッシュ・インスタンスのデフォルトです。このサーブレット動的キャッシュ・インスタンスは、IBM WebSphere Commerce などの製品によって使用されます。baseCache は、WebSphere Application Server で定義される他のキャッシュ・インスタンスとは異なり、ただ 1 つのサーバー・インスタンスまたはクラスター・インスタンスに固有のものです。下記の手順を使用して、WebSphere eXtreme Scale と一緒に動的キャッシュ・プロバイダーとして使用するよう WebSphere Application Server で baseCache インスタンスを構成します。

始める前に

- 動的キャッシュ・プロバイダーを使用するには、WebSphere Application Server ノードの各デプロイメント (デプロイメント・マネージャー・ノードを含む) 上に WebSphere eXtreme Scale クライアント をインストールしておく必要があります。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- WebSphere eXtreme Scale カタログ・サービス・ドメインが構成されていなければなりません。詳しくは、[WebSphere Application Server でのカタログ・サービス・ドメインの作成](#) を参照してください。
- カタログ・サービス・ドメイン内のカタログ・サーバーで Secure Sockets Layer (SSL) が使用可能になっている場合、または SSL がサポートされたカタログ・サービス・ドメインで SSL を使用する場合は、WebSphere Application Server 管理コンソールでグローバル・セキュリティーを有効にする必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#) を参照してください。

このタスクについて

この手順に含まれるステップは、WebSphere Application Server 管理コンソールのバージョン 8.0 に対応します。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注:

- WebSphere eXtreme Scale バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。
- 下記の手順は、リモート WebSphere eXtreme Scale 動的キャッシュ・トポロジーに固有のものです。現在、WebSphere eXtreme Scale バージョン 8.6 では、その他のトポロジー (組み込み、組み込み区画化、ローカルなど) はすべて非推奨です。
- 次の手順では、WebSphere Application Server V7.0 Fix Pack 27、V8.0 Fix Pack 6、V8.5 Fix Pack 2 以上を使用していることが前提になっています。これらのバージョンには WebSphere Application Server APAR PM71992 が含まれています。

手順

- WebSphere Application Server 管理コンソールを開始します。
- トップ・メニューで、「サーバー」>「サーバー・タイプ」>「**WebSphere** アプリケーション・サーバー」をクリックします。
- 「アプリケーション・サーバー」領域で、「**your server name**」を選択します。
- 「構成」パネルで、「コンテナ・サービス」をクリックし、「動的キャッシュ・サービス」を選択します。
- 「**Cache provider**」ドロップダウン・リストで「WebSphere eXtreme Scale」を選択します。
- キャッシュ・サイズを変更したい場合は、そのキャッシュ・サイズを「キャッシュ・サイズ」ボックスで指定します。キャッシュ・サイズ値は、この動的キャッシュ・インスタンス用の WebSphere eXtreme Scale グリッド内の各区画で許容されるエントリー数の最大数を指定します。デフォルトは 1 区画内 2000 エントリーです。
- 「キャッシュ複製を使用可能にする (**Enable cache replication**)」を選択します。このチェック・ボックスを使用可能にすると、キャッシュ・データはローカルではなくリモート側でデータ・グリッドに保管されます。WebSphere eXtreme Scale をキャッシュ・プロバイダーとして使用するときは、このオプションを選択する必要があります。
- 「適用」または「**OK**」をクリックし、構成を保存します。
- トップ・メニューで、「サーバー」>「サーバー・タイプ」>「**WebSphere** アプリケーション・サーバー」をクリックします。
- 「アプリケーション・サーバー」領域で、「**your server name**」を選択します。
- 「構成」パネルで、「**Web** コンテナ設定」をクリックし、「**Web** コンテナ」を選択します。
- 「サーブレット・キャッシングを有効にする」チェック・ボックスを選択します。
- 「適用」または「**OK**」をクリックし、構成を保存します。

次のタスク

デフォルトでは、WebSphere Application Server で構成されている各動的キャッシュ・インスタンスは、キャッシュ・インスタンスの JNDI 名と同じ名前を持つ動的キャッシュ・データ・グリッドに対応します。またデフォルトでは、そのキャッシュ・インスタンスのデータはその動的キャッシュ・データ・グリッド内の動的マップに保管され、その動的マップ名の接尾部もキャッシュ・インスタンスの JNDI 名に対応します。例えば、cache1 という JNDI 名を持つキャッシュ・インスタンスを WebSphere Application Server で構成すると、cache1 という名前を持つ動的キャッシュ・データ・グリッドがアプライアンスに作成されます。データ・グリッド cache1 の内部では、データを保管するために、IBM_DC_PARTITIONED_cache1 という動的マップが作成されます。

ほとんどの場合、この構成を変更する必要はありません。ただし、状況によっては、異なる JNDI 名を持つ複数のキャッシュ・インスタンスが同じデータ・グリッド・インスタンス内の異なる動的マップにマップされるようにしたいことがあります。また状況が違えば、同じ JNDI 名を持つ複数のキャッシュ・インスタンスが同じ動的キャッシュ・データ・グリッド内の異なる動的キャッシュ・データ・グリッド・インスタンスまたは異なる動的マップ・インスタンスにマップされるようにしたいことがあります。例えば、デフォルトの動的キャッシュ・インスタンス (baseCache) を使用するアプリケーションがある場合、キャッシュ・データを別々のデータ・グリッドに保存するか、同じデータ・グリッド内の別々の動的マップに保存して、テスト・レベル環境と実動レベル環境の両方に対して同じアプライアンスを使用する必要がある場合があります。

この構成を制御するために、キャッシュ・インスタンスで以下のカスタム・プロパティを設定することができます。

ヒント: これらのプロパティは、デフォルトの動的キャッシュ・インスタンス (baseCache) を使用するとき特に役立ちます。これは、キャッシュの JNDI 名は自動的に baseCache に設定され、変更できないからです。

com.ibm.websphere.xs.dynacache.grid_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッド・インスタンス (これに動的キャッシュ・インスタンスが対応します) の名前を指定します。

com.ibm.websphere.xs.dynacache.cache_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッドとそのデータ・グリッド内の動的マップの両方に対して (JNDI 名の代わりに) 使用する名前を指定します。

com.ibm.websphere.xs.dynacache.grid_name カスタム・プロパティも設定された場合は、このプロパティの値は動的マップ名にのみ適用されます。

これらのプロパティの設定方法については、[カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)を参照してください。

親トピック:  **2.5+** [動的キャッシュ・インスタンスの構成](#)

オブジェクトまたはサーブレットの動的キャッシュ・インスタンスの構成

WebSphere® Application Server では、デフォルト・インスタンスのほかに、複数のオブジェクト動的キャッシュ・インスタンスやサーブレット動的キャッシュ・インスタンスを構成することができます。下記の手順を使用して、追加のオブジェクト・キャッシュ・インスタンスやサーブレット・キャッシュ・インスタンスを構成します。

始める前に

- 動的キャッシュ・プロバイダーを使用するには、WebSphere Application Server ノードの各デプロイメント (デプロイメント・マネージャー・ノードを含む) 上に WebSphere eXtreme Scale をインストールしておく必要があります。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- WebSphere eXtreme Scale カタログ・サービス・ドメインが構成されていなければなりません。[WebSphere Application Server でのカタログ・サービス・ドメインの作成](#) を参照してください。
- カタログ・サービス・ドメイン内のカタログ・サーバーで Secure Sockets Layer (SSL) が使用可能になっている場合、または SSL がサポートされたカタログ・サービス・ドメインで SSL を使用する場合は、WebSphere Application Server 管理コンソールでグローバル・セキュリティーを有効にする必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#) を参照してください。

このタスクについて

この手順で作成できるキャッシュ・インスタンスには、オブジェクト・キャッシュ・インスタンスとサーブレット・キャッシュ・インスタンスという 2 つのタイプがあります。オブジェクト・キャッシュ・インスタンスは、デフォルトの共有動的キャッシュであることに加えて、Java 2 Platform, Enterprise Edition (J2EE) アプリケーションがオブジェクトを保管、配布、および共有できるロケーションでもあります。オブジェクト・キャッシュ・インスタンスを構成した後、com.ibm.websphere.cache パッケージ内の DistributedMap または DistributedObjectCache インターフェースを使用して、プログラマチックにキャッシュ・インスタンスにアクセスすることができます。DistributedMap または DistributedObjectCache インターフェースについて詳しくは、[追加のアプリケーション・プログラミング・インターフェース \(API\)](#) を参照してください。サーブレット・キャッシュ・インスタンスは、デフォルト動的キャッシュのほかに、動的キャッシュが呼び出されたサーブレットの出力および副次作用を保管、配布、および共有できるロケーションでもあります。サーブレット・キャッシュ・インスタンスを構成することにより、アプリケーションは柔軟性が増し、キャッシュ・リソースを調整しやすくなります。管理コンソールでキャッシュ・インスタンスに対して指定される Java Naming and Directory Interface (JNDI) 名が、cachespec.xml 構成ファイル内のキャッシュ・インスタンス・エレメントにマップされます。<cache-instance> エレメント内で指定される <cache-entry> エレメントは、そのキャッシュ・インスタンスに作成されます。<cache-instance> エレメント外で指定される <cache-entry> エレメントは、デフォルト動的キャッシュ・インスタンスに格納されます。オブジェクト型およびサーブレット型のキャッシュ・インスタンスについて詳しくは、[キャッシュ・インスタンス](#) を参照してください。

この手順に含まれるステップは、WebSphere Application Server 管理コンソールのバージョン 8.0 に対応します。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注:

- WebSphere eXtreme Scale バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

手順

- WebSphere Application Server 管理コンソールからオブジェクト・キャッシュまたはサーブレット・キャッシュを構成するには、以下のステップを使用します。
 - WebSphere Application Server 管理コンソールを開始します。
 - トップ・メニューで、「リソース」>「キャッシュ・インスタンス」>「オブジェクト・キャッシュ・インスタンス」をクリックします。
 - 「オブジェクト・キャッシュ・インスタンス」領域で、作成したいキャッシュ・インスタンスのタイプを選択します。オブジェクト・キャッシュ・インスタンスまたはサーブレット・キャッシュ・インスタンスのいずれかを選択することができます。
 - キャッシュ・インスタンスの有効範囲を指定します。セルの有効範囲を指定して、セル内のすべてのサーバーがこのキャッシュ・インスタンスを使用できるようにします。ノードの有効範囲を指定すると、キャッシュ・インスタンスはノードのすべてのサーバーで使用可能になります。サーバーの有効範囲を指定すると、キャッシュ・インスタンスは選択したサーバーでのみ使用可能になります。必要に応じて、有効範囲を混用することができます。
 - 「適用」をクリックして範囲を保管します。
 - 「新規」をクリックし、オブジェクト・キャッシュ・インスタンスを定義します。
 - 「Cache provider」ドロップダウン・リストで「WebSphere eXtreme Scale」を選択します。

注: WebSphere eXtreme Scale が動的キャッシュ・プロバイダーとして表示されなかった場合は、WebSphere Application Server プロファイルが WebSphere eXtreme Scale 用に拡張されていま

せん。

8. 動的キャッシュ・インスタンスの JNDI 名を指定します。オブジェクト・キャッシュの場合、この名前はキャッシュの検索時に使用されます。サーブレット・キャッシュの場合、これは `cachespec.xml` ファイル内の `<cache-instance>` エレメントで指定される `name` 属性です。
 9. 動的キャッシュ・オブジェクトの JNDI 名を指定します。
 10. キャッシュ・サイズを変更したい場合は、そのキャッシュ・サイズを「キャッシュ・サイズ」ボックスで指定します。キャッシュ・サイズ値は、この動的キャッシュ・インスタンス用の WebSphere eXtreme Scale グリッド内の各区画で許容されるエントリーの最大数を指定します。デフォルトは 1 区画内 2000 エントリーです。
 11. 「**キャッシュ複製を使用可能にする (Enable cache replication)**」ボックスを選択します。このチェック・ボックスを使用可能にすると、キャッシュ・データはローカルではなくリモート側でグリッドに保管されます。WebSphere eXtreme Scale をキャッシュ・プロバイダーとして使用しているときは、このボックスを選択する必要があります。
 12. 「**適用**」または「**OK**」をクリックし、構成を保存します。
- `cacheinstances.properties` ファイルを使用してオブジェクト・キャッシュまたはサーブレット・キャッシュを構成するには、以下のステップを使用します。
 1. `cacheinstances.properties` ファイルを作成します。必要とされるコンテンツについては、[キャッシュ・インスタンス・プロパティ・ファイル](#) を参照してください。
 2. `cacheinstances.properties` ファイルを、アプリケーション・サーバーかアプリケーション・クラスパスのいずれかに配置します。例えば、アプリケーション WAR (Web アプリケーション・アーカイブ) ファイル (`WEB-INF\classes` ディレクトリー) を使用することもできれば、`server_root\classes` ディレクトリーを作成して、そこにファイルを置くこともできます。

次のタスク

デフォルトでは、WebSphere Application Server で構成されている各動的キャッシュ・インスタンスは、キャッシュ・インスタンスの JNDI 名と同じ名前を持つ動的キャッシュ・データ・グリッドに対応します。またデフォルトでは、そのキャッシュ・インスタンスのデータはその動的キャッシュ・データ・グリッド内の動的マップに保管され、その動的マップ名の接尾部もキャッシュ・インスタンスの JNDI 名に対応します。例えば、`cache1` という JNDI 名を持つキャッシュ・インスタンスを WebSphere Application Server で構成すると、`cache1` という名前を持つ動的キャッシュ・データ・グリッドがアプライアンスに作成されます。データ・グリッド `cache1` の内部では、データを保管するために、`IBM_DC_PARTITIONED_cache1` という動的マップが作成されます。

ほとんどの場合、この構成を変更する必要はありません。ただし、状況によっては、異なる JNDI 名を持つ複数のキャッシュ・インスタンスが同じデータ・グリッド・インスタンス内の異なる動的マップにマップされるようにしたいことがあります。また状況が違えば、同じ JNDI 名を持つ複数のキャッシュ・インスタンスが同じ動的キャッシュ・データ・グリッド内の異なる動的キャッシュ・データ・グリッド・インスタンスまたは異なる動的マップ・インスタンスにマップされるようにしたいことがあります。例えば、デフォルトの動的キャッシュ・インスタンス (`baseCache`) を使用するアプリケーションがある場合、キャッシュ・データを別々のデータ・グリッドに保存するか、同じデータ・グリッド内の別々の動的マップに保存して、テスト・レベル環境と実動レベル環境の両方に対して同じアプライアンスを使用する必要があります。

この構成を制御するために、キャッシュ・インスタンスで以下のカスタム・プロパティを設定することができます。

ヒント: これらのプロパティは、デフォルトの動的キャッシュ・インスタンス (`baseCache`) を使用するとき特に役立ちます。これは、キャッシュの JNDI 名は自動的に `baseCache` に設定され、変更できないからです。

com.ibm.websphere.xs.dynacache.grid_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッド・インスタンス (これに動的キャッシュ・インスタンスが対応します) の名前を指定します。

com.ibm.websphere.xs.dynacache.cache_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッドとそのデータ・グリッド内の動的マップの両方に対して (JNDI 名の代わりに) 使用する名前を指定します。

com.ibm.websphere.xs.dynacache.grid_name カスタム・プロパティも設定された場合は、このプロパティの値は動的マップ名にのみ適用されます。

これらのプロパティの設定方法については、[カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#) を参照してください。

親トピック:  [動的キャッシュ・インスタンスの構成](#)

カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ

WebSphere® Application Server により、動的キャッシュ・インスタンスに対してカスタム・プロパティを設定することができます。

始める前に

デフォルト・キャッシュ・インスタンスまたは追加のオブジェクト型あるいはサーブレット型のキャッシュ・インスタンスのいずれかが既に構成されていなければなりません。 [デフォルト動的キャッシュ・インスタンス \(baseCache\) の構成](#) または [オブジェクトまたはサーブレットの動的キャッシュ・インスタンスの構成](#) を参照してください。

このタスクについて

次のいずれかの方法で、特定の動的キャッシュ・インスタンスに固有のカスタム・プロパティを設定することができます。

- 特定の動的キャッシュ・インスタンスに固有のカスタム・プロパティを設定したい場合は、APAR PM71992 が適用された WebSphere Application Server 管理コンソールを使用します。このフィックスをお持ちでない場合は、WebSphere Application Server サポート・ページ (<http://www.ibm.com/software/webservers/appserv/was/support>) にお問い合わせください。
- `cacheinstances.properties` ファイルが作成されている場合は、そのファイル内のカスタム・プロパティを設定することができます。この方法は、デフォルト動的キャッシュ (baseCache) インスタンスに対するカスタム・プロパティの設定には使用できません。
- WebSphere Application Server 管理コンソールを使用して、Java 仮想マシン (JVM) のカスタム・プロパティの値を変更することができます。

注: 設定した JVM プロパティは特定の JVM 内で使用されたすべてのキャッシュ・インスタンスに影響を及ぼすことがあります。

注: JVM プロパティの有効範囲は WebSphere Application Server JVM 内のすべてのキャッシュ・インスタンスまで及ぶことがあります。したがって、ほとんどの状況では、WebSphere Application Server 管理コンソール (デフォルト・キャッシュ・インスタンスの場合は APAR PM71992 を適用) でのキャッシュ固有カスタム・プロパティの使用または `cacheinstances.properties` ファイルの使用をお勧めします。

手順

- WebSphere Application Server 管理コンソールでキャッシュ・インスタンスに対してカスタム・プロパティを設定するには、以下のステップを使用します。

1. WebSphere Application Server 管理コンソールを開始します。

注: WebSphere Application Server APAR PM71992 を適用するまでは、デフォルト (baseCache) インスタンスに対してこれらのステップを使用することはできません。このフィックスは WebSphere Application Server バージョン 7.0.0.27、8.0.0.6、および 8.5.0.2 以上で使用可能です。このフィックスをお持ちでない場合は、WebSphere Application Server サポート・ページ (<http://www.ibm.com/software/webservers/appserv/was/support>) にお問い合わせください。

2. 既に構成済みである目的のキャッシュ・インスタンスに移動します。
3. そのキャッシュ・インスタンス・パネルから、「追加プロパティ」>「カスタム・プロパティ」をクリックします。
4. 「新規」を選択し、カスタム・プロパティの名前と値を指定します。
5. 「適用」または「OK」をクリックし、構成を保存します。
6. Deployment Manager とすべてのアプリケーション・サーバー・プロセスを再始動します。

- `cacheinstances.properties` ファイルを使用してキャッシュ・インスタンスのカスタム・プロパティを設定するには、以下のステップを使用します。

注: デフォルト (baseCache) インスタンスに対してこれらのステップを使用することはできません。

1. カスタム・プロパティを `cacheinstances.properties` ファイルに追加します。このファイルを作成する必要がある場合は、[キャッシュ・インスタンス・プロパティ・ファイル](#) を参照して必要な内容を確認してください。
2. `cacheinstances.properties` ファイルを、アプリケーション・サーバーかアプリケーション・クラスパスのいずれかに配置します。例えば、アプリケーション WAR (Web アプリケーション・アーカイブ) ファイル (`WEB-INF\classes` ディレクトリー) を使用することもできれば、`server_root\classes` ディレクトリーを作成して、そこにファイルを置くこともできます。

- WebSphere Application Server 管理コンソールを使用して、Java 仮想マシン (JVM) のカスタム・プロパティの値を変更することができます。詳しくは、[Java 仮想マシンのカスタム・プロパティ](#) を参照してください。

Java **2.5+** [キャッシュ・インスタンス・プロパティ・ファイル](#)

cacheinstances.properties ファイルを使用して、オブジェクトまたはサーブレット・キャッシュを構成できます。

Java **2.5+** [動的キャッシュ・カスタム・プロパティ](#)

デフォルト動的キャッシュ・インスタンスやサーブレット・キャッシュ・インスタンスまたはオブジェクト・キャッシュ・インスタンスのカスタム・プロパティを設定する際、この表が参照として役立ちます。

親トピック: **Java** **2.5+** [動的キャッシュ・インスタンスの構成](#)

キャッシュ・インスタンス・プロパティ・ファイル

cacheinstances.properties ファイルを使用して、オブジェクトまたはサブレット・キャッシュを構成できます。

表 1. キャッシュ・インスタンス・プロパティ

プロパティ名 - x は インスタンス番号	必須	有効範囲	可能値	説明
cache.instance.x	はい	キャッシュ・ インスタンス 当たり	任意のストリング (デフォ ルト設定なし)	キャッシュ・インス タンス名または JNDI 名を指定します。
cache.instance.x.cac heSize	いいえ	キャッシュ・ インスタンス 当たり	> 0 (デフォルトは 2000)	WebSphere® eXtreme Scale グ リッド内の単一区画 で許容されるエント リーの最大数を指定 します。これに区画 数を掛ける と、WebSphere eXtreme Scale グ リッド内のキャッシ ュの容量が分かりま す。
cache.instance.x.cre ateCacheAtServerSt artup	いいえ	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	構成されたキャッシ ュ・インスタンスが サーバーの始動時に 作成されるかどうか を指定します。
cache.instance.x.ena bleServletSupport	いいえ	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	キャッシュ・インス タンスがサブレッ ト・キャッシュかオ ブジェクト・キャッシ ュかを指定しま す。
cache.instance.x.ena bleCacheReplication	はい (APAR までの み)	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	キャッシュがアプリ ケーション・サーバ ーからリモートであ ることを示していま す。WebSphere eXtreme Scale リ モート・トポロジー では、このプロパテ ィーを True に設定 する必要があります。
cache.instance.x.cac heProviderName	はい	キャッシュ・ インスタンス 当たり	com.ibm.ws.objectgrid. dynacache.CacheProvid erImpl	動的キャッシュ・プ ロバイダーを示しま す。WebSphere eXtreme Scale が 指定されなかった場 合は、WebSphere Application Server プロバイダーがデフ ォルトとして使用さ れます。
cache.instance.x.ign oreValueInInvalidatio nEvent	いいえ	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	無効化イベントのキ ャッシュ値を無視す るかどうかを指定し ます。true の場合 は、コードが呼び出 し元に返されるとき に、無効化イベント のキャッシュの値が NULL に設定されま

cache.instance.x. <custom property>	追加したいプロパティによって変わります。	キャッシュ・インスタンスあたり	追加したいプロパティによって変わります。	す。 このファイルには、すべてのカスタム・プロパティを追加できます。
--	----------------------	-----------------	----------------------	---------------------------------------

親トピック: [Java 2.5+](#) [カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)

動的キャッシュ・カスタム・プロパティ

デフォルト動的キャッシュ・インスタンスやサブレット・キャッシュ・インスタンスまたはオブジェクト・キャッシュ・インスタンスのカスタム・プロパティを設定する際、この表が参照として役立ちます。

表 1. 動的キャッシュ・カスタム・プロパティ

プロパティ名	必須	有効範囲	指定可能な値	説明
com.ibm.websphere.xs.dynacache.topology	はい (WebSphere® Application Server APAR PM71992 が適用されると、以後このプロパティは不要になります。)	キャッシュ・インスタンスあたり	remote	キャッシュ・インスタンスのトポロジーを示します。remote トポロジーは、WebSphere DataPower® XC10 アプリケーションで使用できる唯一の有効なトポロジーです。。
com.ibm.ws.cache.CacheConfig.ignoreValueInInvalidationEvent	いいえ	キャッシュ・インスタンスあたり	true または false デフォルト: true	無効化イベントのキャッシュ値を無視するかどうかを指定します。true の場合は、コードが呼び出し元に返されるときに、無効化イベントのキャッシュの値が NULL に設定されます。
com.ibm.websphere.xs.dynacache.ignore_value_in_change_event	いいえ	キャッシュ・インスタンスあたり	true または false デフォルト: true	変更イベントのキャッシュ値を無視するかどうかを指定します。true の場合は、コードが呼び出し元に返されるときに、変更イベントのキャッシュの値が NULL に設定されます。
com.ibm.websphere.xs.dynacache.cs_override	いいえ	キャッシュ・インスタンスあたり	カタログ・サービス・エンドポイント 例: 9.5.12.345:2819	このキャッシュ・インスタンスと関連付けるデータ・グリッドのカタログ・サービス・エンドポイントを指定します。このフィールドは、WebSphere Application Server 管理コンソールで指定されなかった場合は必須です。
com.ibm.websphere.xs.dynacache.grid_name	いいえ	キャッシュ・インスタンスあたり	任意のストリング デフォルト: JNDI name of the cache instance	作成したデータ・グリッドの名前を指定してください。

com.ibm.websphere.xs.dynacache.map_template_name	いいえ	キャッシュ・インスタンスあたり	以下のマップ・テンプレートのいずれかを使用できます。 <ul style="list-style-type: none"> • IBM_DC_PARTITIONED_.* (デフォルト) • IBM_DC_NCI_PARTITIONED_.* (このキャッシュがニア・キャッシュを使用することを示します。) 	テンプレート・マップ接頭部の名前を指定します。
com.ibm.websphere.xs.dynacache.cache_name	いいえ	キャッシュ・インスタンスあたり	任意のストリング デフォルト: cache.instance.x の値	テンプレート・マップの名前として使用される固有の接尾部の名前を指定します。例えば、IBM_DC_PARTITIONED.<cache_name>
com.ibm.websphere.xs.dynacache.near_cache_size	いいえ	キャッシュ・インスタンスあたり	ゼロより大きい値。 デフォルト: cache.instance.x.cacheSize に指定された値	ニア・キャッシュ・インスタンスで許容されるエントリーの最大数を指定します。デフォルトでは、この値は、このキャッシュ・インスタンスのリモート ObjectGrid 内の単一区画で許容されるエントリーの最大数と同じです。
com.ibm.websphere.xs.dynacache.request_retry_timeout_override	いいえ	キャッシュ・インスタンスあたり	整数値 (ミリ秒): <ul style="list-style-type: none"> • -1: 無期限。要求はいつまでもタイムアウトになりません。 • 0: 要求はただちにタイムアウトになります。 • >0: 要求がタイムアウトにならずに実行される時間 (ミリ秒)。 • 2000: このカスタム・プロパティまたはクライアント・プロパティ 	要求がタイムアウトにならないで実行できる時間 (ミリ秒) を指定します。このプロパティは、クライアント・プロパティ・ファイルで requestRetryTimeout プロパティが設定されている場合、その値をオーバーライドします。詳しくは、 クライアント・プロパティ・ファイル を参照してください。 動的キャッシュ・インスタンスの要求再試行タイムアウト値が、このカスタム・プロパティでも、クライアント・プロパティ・ファイルでも設定されていない場合、デフォルトで 2000 ミリ秒になります。

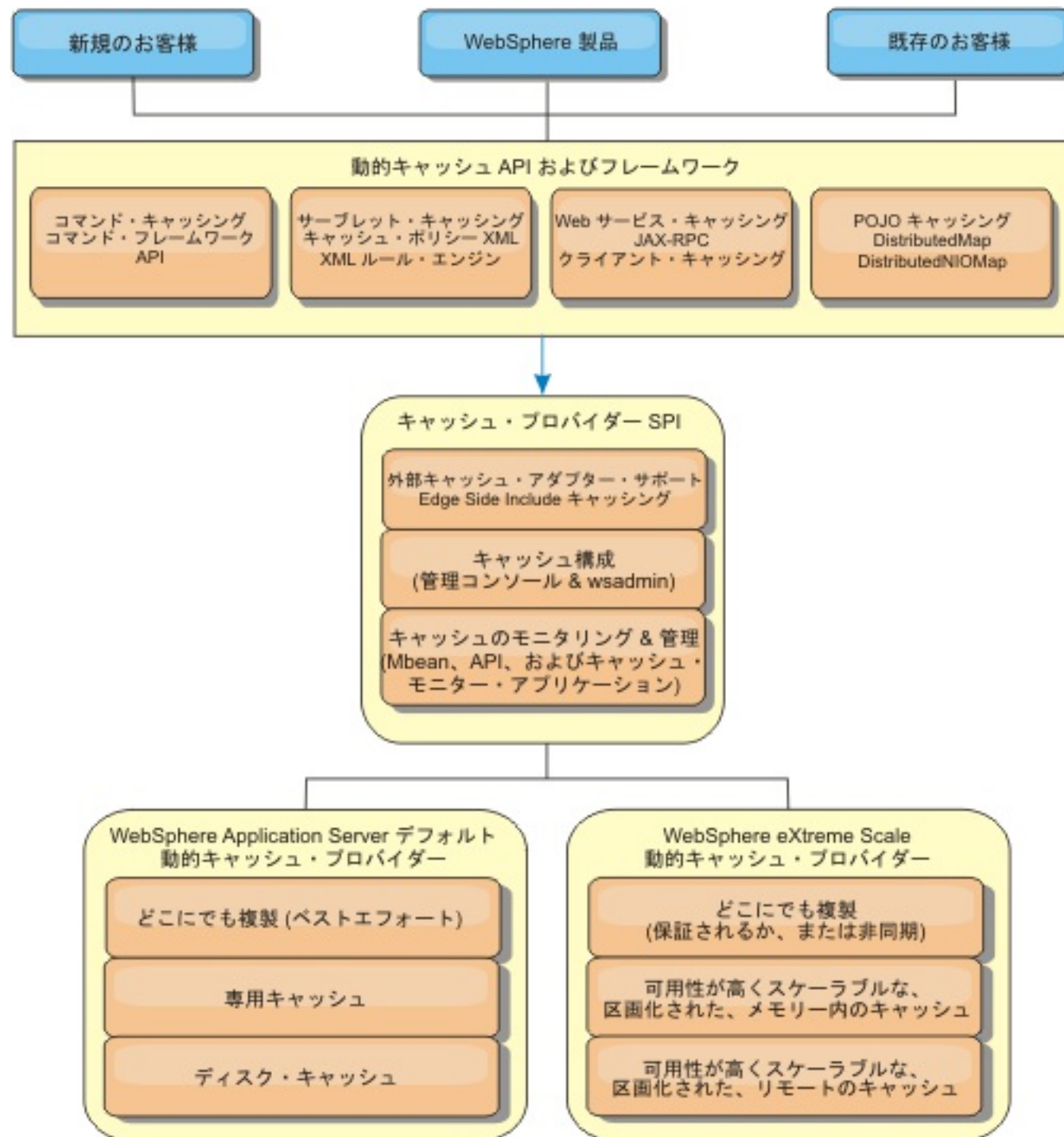
			ー・ファイルを使用してこの値を指定しなかった場合のデフォルト値。	
--	--	--	----------------------------------	--

親トピック: [Java 2.5+](#) [カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)

動的キャッシュ・プロバイダーの概要

Java WebSphere® Application Server は、デプロイされた Java™ EE アプリケーションが使用できる動的キャッシュ・サービスを提供します。このサービスは、サーブレット、JSP、またはコマンドからの出力などのデータ、および DistributedMap API を使用してエンタープライズ・アプリケーション内にプログラマチックに指定されているオブジェクト・データをキャッシュするために使用されます。この機能をセットアップすることにより、動的キャッシュ・サービスを使用するアプリケーションが WebSphere DataPower® XC10 アプライアンス のフィーチャーおよびパフォーマンス機能を使用できるようになります。

当初は、動的キャッシュ・サービスのサービス・プロバイダーのみが、WebSphere Application Server に組み込まれているデフォルト動的キャッシュ・エンジンでした。また、WebSphere DataPower XC10 アプライアンス を任意のキャッシュ・インスタンス用のキャッシュ・プロバイダーに指定することもできます。この機能をセットアップすることにより、動的キャッシュ・サービスを使用するアプリケーションが のフィーチャーおよびパフォーマンス機能を使用できるようになります。



動的キャッシュ・プロバイダーのインストールと構成の手順については、[デフォルト動的キャッシュ・インスタンス \(baseCache\) の構成](#)を参照してください。

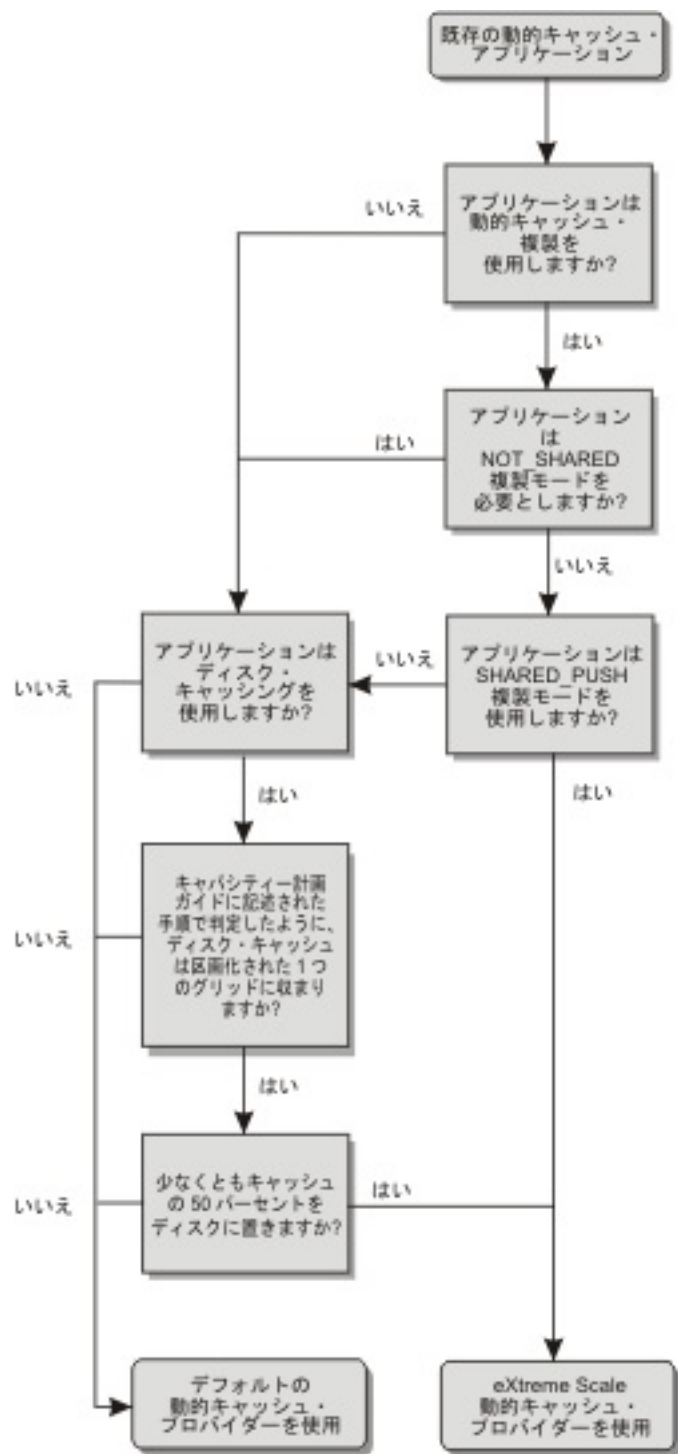
WebSphere DataPower XC10 アプライアンス

の使用方法の決定

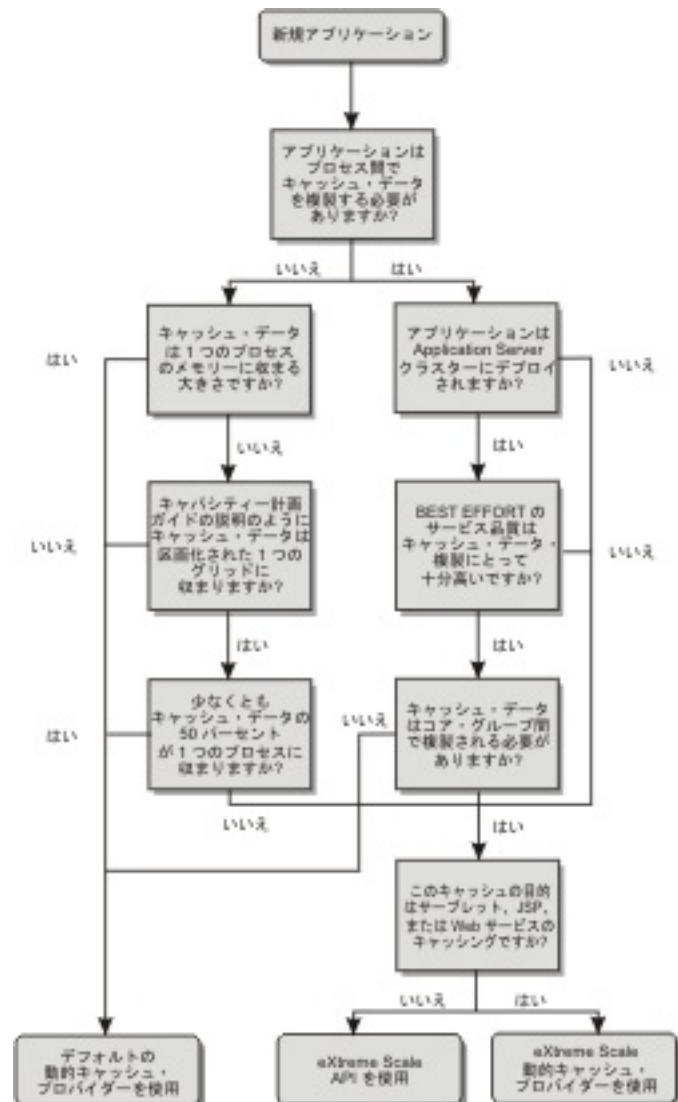
WebSphere DataPower XC10 アプライアンス の使用可能なフィーチャーにより、動的キャッシュ・サービスの分散機能は、デフォルト動的キャッシュ・プロバイダーおよびデータ複製サービスで提供される機能を超えて大幅に向上します。eXtreme Scale を使用することにより、複数のサーバー間で単に複製し、同期化するだけでなく、サーバー間で本当に分散したキャッシュを作成できます。さらに、eXtreme Scale キャッシュは、トランザクション・ベースであり、可用性がとて高く、動的キャッシュ・サービスに関して各サーバーが同じ内容を参照することを保証します。WebSphere DataPower XC10 アプライアンス は、キャッシュ複製に関して、DRS よりも高いサービス品質を提供します。

ただし、これらの利点は、どのようなアプリケーションでも eXtreme Scale 動的キャッシュ・プロバイダー が正しい選択であることを意味するわけではありません。以下のデシジョン・ツリーおよび機能比較マトリックスを使用して、ご使用のアプリケーションに最適のテクノロジーを決定してください。

既存の動的キャッシュ・アプリケーションをマイグレーションする際のデシジョン・ツリー



新規アプリケーションのキャッシュ・プロバイダーを選択する際のデシジョン・ツリー



機能比較

表 1. 機能比較

キャッシュ機能	デフォルト・プロバイダー	eXtreme Scale プロバイダー	eXtreme Scale API
ローカルのメモリー内のキャッシング	はい	ニア・キャッシュ機能を介して	ニア・キャッシュ機能を介して

分散キャッシング	DRS を介して	はい	はい
直線的にスケーラブル	いいえ	はい	はい
信頼できる複製 (同期)	いいえ	はい	はい
ディスク・オーバーフロー	はい	N/A	N/A
除去	LRU/TTL/ヒープ・ベース	LRU/TTL (区画ごと)	LRU/TTL (区画ごと)
Invalidation	はい	はい	はい
関係	依存関係 / テンプレート ID の関係	はい	いいえ (他の関係は可能)
非キー検索	いいえ	いいえ	照会と索引を介して
バックエンド統合	いいえ	いいえ	ローダーを介して
トランザクションの	いいえ	はい	はい
キー・ベースの保管	はい	はい	はい
イベントおよびリスナー	はい	いいえ	はい
WebSphere Application Server 統合	単一セルのみ	複数セル	セルに無関係
Java Standard Edition サポート	いいえ	はい	はい
モニタリングおよび統計	はい	はい	はい
セキュリティー	はい	はい	はい

注: eXtreme Scale 分散 キャッシュは、キーと値が両方とも java.io.Serializable インターフェースを実装するエントリーのみを保管できます。

動的キャッシュ・エンジンおよび eXtreme Scale の機能の相違点

WebSphere DataPower XC10 アプライアンス がバックアップしたキャッシュは、メモリー内キャッシュのサイズに関する統計および操作、またはディスク・オフロードをサポートしない点は除いて、ユーザーはこれら 2 つのキャッシュの機能的な違いに気付かないはずで

デフォルトの動的キャッシュ・プロバイダーを使用しているか、eXtreme Scale キャッシュ・プロバイダーを使用しているかに関わらず、ほとんどの場合、動的キャッシュ API 呼び出しが返す結果にそれほどの相違はありません。一部の操作については、eXtreme Scale を使用して動的キャッシュ・エンジンの動作をエミュレートできません。

動的キャッシュ統計

MBean 呼び出し

WebSphere eXtreme Scale 動的キャッシュ・プロバイダー はディスク・キャッシングをサポートしません。ディスク・キャッシングに関係する MBean 呼び出しはすべて 機能しません。

動的キャッシュ複製ポリシーのマッピング

eXtreme Scale 動的キャッシュ・プロバイダーのリモート・トポロジーは、(デフォルトの WebSphere Application Server 動的キャッシュ・プロバイダーによって使用される用語を使用して) SHARED_PULL および SHARED_PUSH_PULL ポリシーに最も近い複製ポリシーをサポートします。eXtreme Scale 動的キャッシュでは、キャッシュの分散状態はすべてのサーバー間で一貫しています。

グローバル索引の無効化

グローバル索引を使用して、大規模な区画化環境 (例えば、区画数が 40 を超えるような環境) での無効化効率を向上させることができます。グローバル索引フィーチャーを使用しない場合、動的キャッシュ・テンプレートおよび依存関係無効化処理ですべての区画にリモート・エージェント要求を送らなければならないため、結果としてパフォーマンスが低下します。グローバル索引を構成すると、無効化エージェントは、テンプレート ID または依存関係 ID に関連しているキャッシュ・エントリーを含む適用区画にのみ送られます。多数の区画が構成されている環境ではパフォーマンス向上の可能性がより大きくなります。サンプルの動的キャッシュ objectGrid 記述子 XML ファイルに入っている依存関係 ID 索引とテンプレート ID 索引を使用してグローバル索引を構成することができます。

ニア・キャッシュ

アプリケーション・サーバー JVM 内にローカルに存在するニア・キャッシュを作成して維持するように動的キャッシュ・インスタンスを構成することができます。このニア・キャッシュには、リモートの動的キャッシュ・インスタンス内に含まれるエントリーのサブセットが含まれます。詳しくは、[動的キャッシュのニア・キャッシュの構成](#)を参照してください。ニア・キャッシュを調整するためのカスタム・プロパティーもあります。詳しくは、[動的キャッシュ・カスタム・プロパティー](#)を参照してください。

2.5+

マルチマスター複製

WebSphere DataPower XC10 アプライアンス 動的キャッシュ・プロバイダーでマルチマスター複製を使用可能に設定する選択ができます。詳しくは、[動的キャッシュの作成データ・グリッド](#)を参照してください。

追加情報

- [動的キャッシュに関するレッドブック](#)
- 動的キャッシュ文書
 - [WebSphere Application Server 7.0](#)
- DRS 資料
 - [WebSphere Application Server 7.0](#)

親トピック: Java [動的キャッシュの作成データ・グリッド](#)

WebSphere Application Server でのカタログ・サービス・ドメインの作成

WebSphere® DataPower® XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

始める前に

- カタログ・サービス・ドメインの作成は、動的キャッシュ・データ・グリッドの場合にのみ必要です。単純なデータ・グリッドまたはセッション・データ・グリッドを使用する場合には、カタログ・サービス・ドメインを構成する必要はありません。WebSphere Application Server に WebSphere eXtreme Scale クライアントがインストールされている必要があります。

このタスクについて

カタログ・サービス・ドメインを作成することで、カタログ・サーバーの高可用性コレクションが定義されます。カタログ・サービス・ドメインを構成することによって、WebSphere DataPower XC10 アプライアンス で稼働しているカタログ・サーバーへの接続を確立します。このカタログ・サービス・ドメインは、集合内のアプライアンスで実行されているカタログ・サーバーのグループを表します。

手順

1. カタログ・サービス・ドメインを作成します。
 - a. WebSphere Application Server 管理コンソールで、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」 > 「新規」をクリックします。
 - b. カタログ・サービス・ドメインの名前、デフォルト値、JMX 認証資格情報を定義します。
 - c. カタログ・サーバー・エンドポイントを追加します。

リモート・サーバーのグループを指定します。これらは、集合内のアプライアンスで実行されているカタログ・サーバーです。集合内で実行されているカタログ・サーバーを表示するには、「集合」 > 「メンバー」 > 「*member_name*」をクリックします。「カタログ・サーバー」フィールドに、集合内で実行されているカタログ・サーバーがリストされます。IP アドレスまたは完全修飾ホスト名を使用して、エンドポイントを指定する必要があります。アプライアンスには次のポート値を使用します。

- クライアント・ポート: アプライアンス上で稼働するカタログ・サーバーへの接続には必須ではありません。
- リスナー・ポート: 2809

2. カタログ・サービス・ドメイン内のカタログ・サーバーへの接続をテストします。
 - a. WebSphere Application Server 管理コンソールで、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」をクリックします。
 - b. テストするカタログ・サービス・ドメインを選択して、「テスト接続」をクリックします。このボタンをクリックすると、すべての定義されたカタログ・サービス・ドメイン・エンドポイントが1つずつ照会されます。いずれかのエンドポイントが使用可能であれば、カタログ・サービス・ドメインへの接続が成功したことを示すメッセージが返されます。

[カタログ・サービス・ドメイン管理用タスク](#)

Jacl または Jython スクリプト言語を使用して、WebSphere Application Server 構成内のカタログ・サービス・ドメインを管理できます。WebSphere DataPower XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

親トピック: [Java 動的キャッシュの作成データ・グリッド](#)

カタログ・サービス・ドメイン管理用タスク

Jacl または Jython スクリプト言語を使用して、WebSphere® Application Server 構成内のカタログ・サービス・ドメインを管理できます。WebSphere DataPower® XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

要件

WebSphere Application Server 環境に WebSphere eXtreme Scale クライアントをインストールしている必要があります。

すべての管理用タスクのリスト

カタログ・サービス・ドメインに関連したすべての管理用タスクのリストを取得するには、**wsadmin** で以下のコマンドを実行します。

```
wsadmin>$AdminTask help XSDomainManagement
```

コマンド

カタログ・サービス・ドメインの管理用タスクには、以下のコマンドが含まれます。

- [createXSDomain](#)
- [deleteXSDomain](#)
- [getDefaultXSDomain](#)
- [listXSDomains](#)
- [modifyXSDomain](#)
- [getTransport](#)
- [testXSDomainConnection](#)
- [testXSServerConnection](#)

createXSDomain

createXSDomain コマンドは、新規カタログ・サービス・ドメインを登録します。

表 1. createXSDomain コマンド引数

引数	説明
-name (必須)	作成するカタログ・サービス・ドメインの名前を指定します。
-default	カタログ・サービス・ドメインがセルでデフォルトかどうかを指定します。デフォルト値は true です。(ブール値: true または false に設定)
-properties	カタログ・サービス・ドメインのカスタム・プロパティを指定します。
-enableXIO	<p>このカタログ・サービス・ドメイン内でのトランスポート通信に IBM eXtreme IO (XIO) が使用されるかオブジェクト・リクエスト・ブローカー (ORB) が使用されるかを指定します。</p> <p>true XIO が使用されることを指定します。</p> <p>false ORB が使用されることを指定します。</p> <p>値を指定しなかった場合、デフォルトは true (XIO 使用可能) です。カタログ・サービス・ドメイン内にリモート・サーバーがある場合、-enableXIO パラメーターはそれらのリモート・サーバーで XIO も ORB も構成しません。リモート・サーバー上にトランスポートを構成するには、リモート・サーバーの始動時にトランスポート・タイプを指定します。</p>

表 2. defineDomainServers ステップ引数

引数	説明
name_endpoint	<p>カタログ・サービス・エンドポイントの名前を指定します。</p> <ul style="list-style-type: none"> • 既存のアプリケーション・サーバーの場合: エンドポイントの名前は、<code>cell_name ¥node_name¥server_name</code> の形式でなければなりません。 • リモート・サーバーの場合: リモート・サーバーのホスト名を指定します。複数のエンドポイントを同じ名前にすることができますが、クライアント・ポートの値はそれぞれのエンドポイントで固有でなければなりません。
custom	カタログ・サービス・ドメイン・エンドポイントのカスタム・プロパティを指定します。カスタム・プロパティがない場合、この引数には一組の二重引用符 ("") を使用します。

<i>_properties</i>	
<i>endpoint_ports</i>	<p>カタログ・サービス・ドメイン・エンドポイントのポート番号を指定します。ポートは、<i><client_port></i>,<i><listener_port></i> の順序で指定する必要があります。</p> <p>クライアント・ポート カタログ・サービス・ドメイン内のカタログ・サーバー間の通信に使用するポートを指定します。この値は、WebSphere Application Server プロセスのみで稼働するカタログ・サーバーに必要で、他で使用されていないどのポートにも設定できます。</p> <p>リスナー・ポート クライアントとの通信に使用するポートを指定します。この値はリモート・エンドポイントに必要で、カタログ・サービスが開始されたときに使用された値と一致している必要があります。リスナー・ポートは、カタログ・サービスとの通信のために、クライアントおよびコンテナによって使用されます。</p> <p>WebSphere DataPower XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのリモート・エンドポイントに、値 2809 を使用します。</p>

戻り値:

バッチ・モードの使用例

バッチ・モードの場合、コマンド項目が正しくフォーマットされていることが必要です。入力する値が適切に処理されるように、対話モードの使用を検討してください。バッチ・モードを使用する場合、特定のプロパティ配列を使用して **-defineDomainServers** のステップ引数を定義する必要があります。このプロパティ配列のフォーマットは、*name_of_endpoint_custom_properties_endpoint_ports* です。 *endpoint_ports* 値は、*<client_port>*,*<listener_port>* の順序で指定する必要があるポートのリストです。

- Jacl を使用した、リモート・エンドポイントのカタログ・サービス・ドメインの作成:

```
$AdminTask createXSDomain {-name TestDomain -default true -defineDomainServers
{{xhost1.ibm.com "" ,2809}} }
```

- Jython スtring を使用した、リモート・エンドポイントのカタログ・サービス・ドメインの作成:

```
AdminTask.createXSDomain('[-name TestDomain -default true
-defineDomainServers [[xhost1.ibm.com "" ,2809]
[xhost2.ibm.com "" ,2809]] ]')
```

- Jacl を使用した、既存のアプリケーション・サーバー・エンドポイントのカタログ・サービス・ドメインの作成:

```
$AdminTask createXSDomain {-name TestDomain -default true -defineDomainServers
{{cellName/nodeName/serverName "" 1109}} }
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask createXSDomain {-interactive}
```

- Jython スtring を使用:

```
AdminTask.createXSDomain ( '[-interactive]')
```

deleteXSDomain

deleteXSDomain コマンドは、カタログ・サービス・ドメインを削除します。

必須パラメーター:

-name

削除するカタログ・サービス・ドメインの名前を指定します。

戻り値:

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask deleteXSDomain {-name TestDomain }
```

- Jython ストリングを使用:

```
AdminTask.deleteXSDomain('[-name TestDomain ]')
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask deleteXSDomain {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.deleteXSDomain ('[-interactive]')
```

getDefaultXSDomain

getDefaultXSDomain コマンドは、セルのデフォルト・カタログ・サービス・ドメインを返します。

必須パラメーター: なし

戻り値: デフォルト・カタログ・サービス・ドメインの名前。

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask getDefaultXSDomain
```

- Jython ストリングを使用:

```
AdminTask.getDefaultXSDomain
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask getDefaultXSDomain {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.getDefaultXSDomain ('[-interactive]')
```

listXSDomains

listXSDomains コマンドは、既存のカタログ・サービス・ドメインのリストを返します。

必須パラメーター: なし

戻り値: セル内のすべてのカタログ・サービス・ドメインのリスト。

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask listXSDomains
```

- Jython ストリングを使用:

```
AdminTask.listXSDomains
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask listXSDomains {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.listXSDomains ('[-interactive]')
```

modifyXSDomain

modifyXSDomain コマンドは、既存のカタログ・サービス・ドメインを変更します。

バッチ・モードの場合、コマンド項目が正しくフォーマットされていることが必要です。入力する値が適切に処理されるように、対話モードの使用を検討してください。バッチ・モードを使用する場合、特定のプロパティ配列を使用して **-modifyEndpoints**、**-addEndpoints**、および **-removeEndpoints** ステップ引数を定義する必要があります。このプロパティ配列のフォーマットは、*name_of_endpoint host_name custom_properties endpoint_ports* です。 *endpoint_ports* 値は、*<client_port>*、*<listener_port>* の順序で指定する必要があるポートのリストです。

表 3. *modifyXSDomain* コマンド引数

引数	説明
-name (必須)	編集するカタログ・サービス・ドメインの名前を指定します。
-default	true に設定した場合、選択したカタログ・サービス・ドメインがセルのデフォルトであることを指定します。(ブール値)
-properties	カタログ・サービス・ドメインのカスタム・プロパティを指定します。
-enableXIO	<p>このカタログ・サービス・ドメイン内でのトランスポート通信に IBM eXtreme IO (XIO) が使用されるかオブジェクト・リクエスト・ブローカー (ORB) が使用されるかを指定します。</p> <p>true XIO が使用されることを指定します。</p> <p>false ORB が使用されることを指定します。</p> <p>値を指定しなかった場合、デフォルトは true (XIO 使用可能) です。カタログ・サービス・ドメイン内にリモート・サーバーがある場合は、それらのリモート・サーバーで XIO を構成することはできません。</p>

表 4. *modifyEndpoints* ステップ引数

引数	説明
<i>name_of_endpoint</i>	<p>カタログ・サービス・エンドポイントの名前を指定します。</p> <ul style="list-style-type: none"> 既存のアプリケーション・サーバーの場合: エンドポイントの名前は、<i>cell_name ¥node_name¥server_name</i> の形式でなければなりません。 リモート・サーバーの場合: リモート・サーバーのホスト名を指定します。複数のエンドポイントを同じ名前にすることができますが、リスナー・ポートの値はそれぞれのエンドポイントで固有でなければなりません。アプライアンスを構成する場合、この値は完全修飾のドメイン名である必要があります。
<i>endpoint_ports</i>	<p>カタログ・サービス・ドメイン・エンドポイントのポート番号を指定します。エンドポイントは、<i><client_port></i>、<i><listener_port></i> の順序で指定する必要があります。</p> <p>クライアント・ポート カタログ・サービス・ドメイン内のカタログ・サーバー間の通信に使用するポートを指定します。この値は、WebSphere Application Server プロセスのみで稼働するカタログ・サーバーに必要で、他で使用されていないどのポートにも設定できます。</p> <p>リスナー・ポート クライアントとの通信に使用するポートを指定します。この値はリモート・エンドポイントに必要で、カタログ・サービスが開始されたときに使用された値と一致している必要があります。リスナー・ポートは、カタログ・サービスとの通信のために、クライアントおよびコンテナによって使用されます。</p> <p>WebSphere DataPower XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのリモート・エンドポイントに、値 2809 を使用します。</p>

表 5. *addEndpoints* ステップ引数

引数	説明
<i>name_of_endpoint</i>	<p>カタログ・サービス・エンドポイントの名前を指定します。</p> <ul style="list-style-type: none"> 既存のアプリケーション・サーバーの場合: エンドポイントの名前は、<i>cell_name ¥node_name¥server_name</i> の形式でなければなりません。

<code>ndp oin t</code>	<ul style="list-style-type: none"> • リモート・サーバーの場合: リモート・サーバーのホスト名を指定します。複数のエンドポイントを同じ名前にすることができますが、リスナー・ポートの値はそれぞれのエンドポイントで固有でなければなりません。アプライアンスを構成する場合、この値は完全修飾のドメイン名である必要があります。
<code>cus tom _pr ope rti es</code>	カタログ・サービス・ドメイン・エンドポイントのカスタム・プロパティを指定します。カスタム・プロパティがない場合、この引数には一組の二重引用符 ("") を使用します。
<code>end poi nt_ por ts</code>	<p>カタログ・サービス・ドメイン・エンドポイントのポート番号を指定します。エンドポイントは、<code><client_port></code>,<code><listener_port></code> の順序で指定する必要があります。</p> <p>クライアント・ポート</p> <p>カタログ・サービス・ドメイン内のカタログ・サーバー間の通信に使用するポートを指定します。この値は、WebSphere Application Server プロセスのみで稼働するカタログ・サーバーに必要で、他で使用されていないどのポートにも設定できます。</p> <p>リスナー・ポート</p> <p>クライアントとの通信に使用するポートを指定します。この値はリモート・エンドポイントに必要で、カタログ・サービスが開始されたときに使用された値と一致している必要があります。リスナー・ポートは、カタログ・サービスとの通信のために、クライアントおよびコンテナによって使用されます。</p> <p>WebSphere DataPower XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのリモート・エンドポイントに、値 2809 を使用します。</p>

表 6. `removeEndpoints` ステップ引数

引数	説明
<code>name_of_endpoint</code>	削除するカタログ・サービス・エンドポイントの名前を指定します。

戻り値:

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask modifyXSDomain [-name TestDomain -default true -modifyEndpoints
{{xhost1.ibm.com "" ,2809}} -addEndpoints {{xhost2.ibm.com "" ,2809}}
-removeEndpoints {{xhost3.ibm.com}}]
```

- Jython スtring を使用:

```
AdminTask.modifyXSDomain(['-name TestDomain
-default false -modifyEndpoints [[xhost1.ibm.com "" ,2809]]
-addEndpoints [[xhost3.ibm.com "" ,2809]]
-removeEndpoints [[xhost2.ibm.com]]'])
```

- IBM eXtremeIO がオンになるように既存のカタログ・サービス・ドメインを変更します。

```
AdminTask.modifyXSDomain(['-name testDomain -enableXIO true'])
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask modifyXSDomain {-interactive}
```

- Jython スtring を使用:

```
AdminTask.modifyXSDomain (['-interactive'])
```

getTransport

`getTransport` コマンドは、カタログ・サービス・ドメインのトランスポート・タイプ、すなわち IBM eXtremeIO (XIO) またはオブジェクト・リクエスト・ブローカー (ORB) を表示します。リモート・サーバーを含むカタログ・サービス・ドメインでこのコマンドを実行した場合、または `catalogServerName` がリモート・サーバーである場合は、エラーとなります。リモート・サーバーに対しては `xscmd -c showTransport` コマンドを使用する必要があります。

必須パラメーター:

-domainName

トランスポート・タイプを表示するカタログ・サービス・ドメインの名前を指定します。

-catalogServerName

トランスポート・タイプを表示するカタログ・サーバーの名前を指定します。

戻り値: ORB または XIO

カタログ・サービス・ドメインのトランスポートの表示

- Jacl を使用:

```
$AdminTask getTransport {-domainName TestDomain }
```

- Jython ストリングを使用:

```
AdminTask.getTransport('[-domainName testDomain]')
```

カタログ・サーバーのトランスポートの表示

- Jacl を使用:

```
$AdminTask getTransport {-catalogServerName myCell01¥myNode01¥container1 }
```

- Jython ストリングを使用:

```
AdminTask.getTransport('[-catalogServerName myCell01¥myNode01¥container1]')
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask getTransport {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.getTransport ('[-interactive]')
```

testXSDomainConnection

testXSDomainConnection コマンドは、カタログ・サービス・ドメインへの接続をテストします。

必須パラメーター:

-name

接続をテストするカタログ・サービス・ドメインの名前を指定します。

オプション・パラメーター

-timeout

接続されるまで待機する最大時間を秒数で指定します。

戻り値: 接続できた場合、true が返されます。接続できなかった場合は、接続エラー情報が返されます。

バッチ・モードの使用例

- Jacl を使用:

```
$Admintask testXSDomainConnection
```

- Jython ストリングを使用:

```
AdminTask.testXSDomainConnection
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask testXSDomainConnection {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.testXSDomainConnection ('[-interactive]')
```

testXSServerConnection

testXSServerConnection コマンドは、カタログ・サーバーへの接続をテストします。このコマンドは、スタンドアロン・サーバーと、カタログ・サービス・ドメインに属するサーバーの両方で機能します。

必須パラメーター:

ホスト (host)

カタログ・サーバーが存在するホストを指定します。

listenerPort

カタログ・サーバーのリスナー・ポートを指定します。

オプション・パラメーター

timeout

カタログ・サーバーに接続されるまで待機する最大時間を秒数で指定します。

戻り値:

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask testXSServerConnection {-host xhost1.ibm.com -listenerPort 2809}
```

- Jython ストリングを使用:

```
AdminTask.testXSServerConnection('[-host xshost3.ibm.com -listenerPort 2809]')
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask testXSServerConnection {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.testXSServerConnection ('[-interactive]')
```

親トピック: [Java](#) [WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)

動的キャッシュのニア・キャッシュの構成

2.5+ アプライアンスまたは集合で、動的キャッシュ・データ・グリッドと一緒に動作するローカルのニア・キャッシュを構成することができます。ニア・キャッシュは、ローカル Java 仮想マシン (JVM) リソースを使用します。一般に、ニア・キャッシュには、アプライアンスの動的キャッシュ・データ・グリッドにあるデータのサブセットが保持されません。

始める前に

動的キャッシュ・データ・グリッドを作成します。詳しくは、[動的キャッシュの作成データ・グリッド](#)を参照してください。

このタスクについて

動的キャッシュ・データ・グリッドのニア・キャッシュを使用可能にするには、WebSphere Application Server 動的キャッシュ・インスタンスにマップ・テンプレート名を指定するカスタム・プロパティを設定する必要があります。

手順

ニア・キャッシュを使用可能にするには、以下のカスタム・プロパティを設定します。

- `com.ibm.websphere.xs.dynacache.map_template_name`: テンプレート名を変更するには、`IBM_DC_NCI_PARTITIONED_.*` を指定します。キャッシュ・インスタンスでこのカスタム・プロパティを設定すると、そのキャッシュ・インスタンスに対するニア・キャッシュが使用可能になります。

動的キャッシュでのカスタム・プロパティの設定について詳しくは、[カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)を参照してください。

次のタスク

デフォルトでは、動的ニア・キャッシュの最大サイズは、リモートの動的キャッシュ・データ・グリッド内の単一区画の最大エントリー数に相当します。このサイズを維持するために最長未使用時間 (LRU) 除去アルゴリズムが使用されます。ニア・キャッシュのサイズを詳細に制御するには、次のアクションを実行することができます。

- ニア・キャッシュで許容される最大キャッシュ・エントリー数を指定するように `com.ibm.websphere.xs.dynacache.near_cache_size` カスタム・プロパティを構成します。詳しくは、[動的キャッシュ・カスタム・プロパティ](#)を参照してください。

親トピック: [Java](#) [動的キャッシュの作成データ・グリッド](#)

データ・グリッドの最大容量の構成

集合内の各データ・グリッドの最大容量を定義できます。容量限度を構成すると、特定のデータ・グリッドで使用可能なデータ・ストレージの容量が制限されます。この容量限度により、確実に、集合の使用可能なストレージ容量が予測可能な形で使用されるようになります。

始める前に

- ご使用の構成用のデータ・グリッドを作成します。デフォルトでは、データ・グリッドの最大容量限度は構成されません。すべてのタイプのデータ・グリッド (単純データ・グリッド、セッション・データ・グリッド、動的キャッシュ・データ・グリッド) で最大容量を構成できます。

このタスクについて

集合内の各データ・グリッドの最大容量限度の構成後、構成したデータ・グリッド容量限度をデータ・グリッド内の基本データの合計サイズと比較して、容量限度を順守します。データ・グリッドと構成された容量限度とを比較する際には、データのレプリカ・コピーによって使用される容量は比較対象に含まれません。

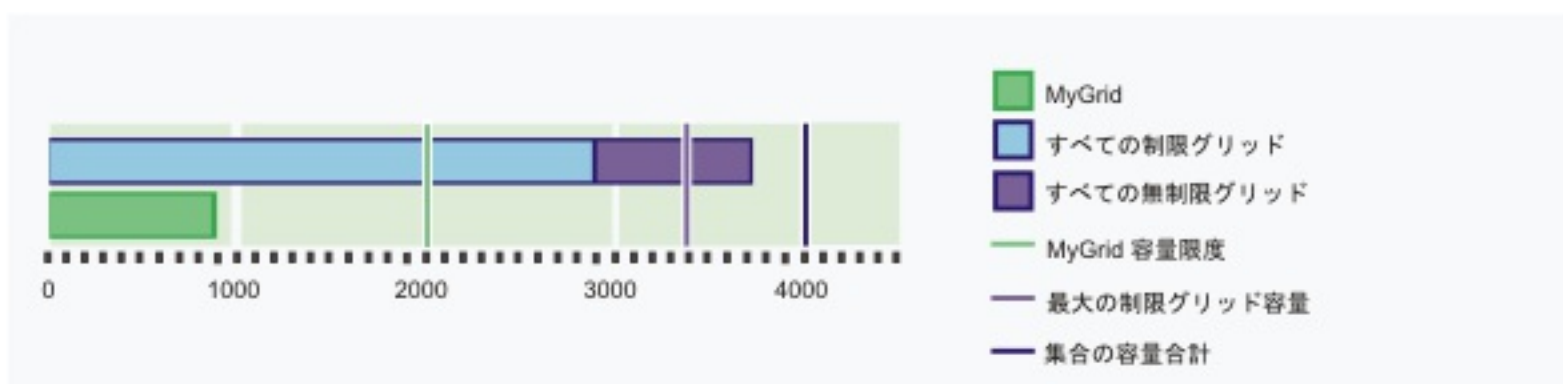
最大容量限度とは、データ・グリッドに挿入可能なデータの最大容量です。この限度を構成しても、データ・グリッドに割り振られるスペース容量は保証されません。したがって、集合にデータを格納できる十分な容量がない場合には、データ・グリッドで構成されている容量限度に達しないことがあります。集合での容量不足の原因としては、データ・グリッドの高い容量限度、または集合内の他のデータ・グリッドに使用された容量の可能性がります。

特定のデータ・グリッドで容量限度を超えると、グリッドは次のいずれかの方法で挿入操作を処理します。

- デフォルトでは、グリッドの限界容量を超えることが予想される挿入操作は拒否されます。クライアント・プロセスでは、挿入要求に対して例外を受け取ります。ただし読み取り操作、更新操作、および削除操作は、データ・グリッドの容量限度を超える場合でも正常に実行されます。これらの限られた操作では、データ・グリッドは最小機能レベルで稼働できますが、データ・グリッドのサイズが大きくなるのが回避されます。
- 単純データ・グリッドでのみ使用可能ですが、「このデータ・グリッドに最長未使用時間 (LRU) の除去を使用」オプションを選択することで、このデフォルトの動作をオーバーライドできます。このオプションを選択すると、最長未使用時間のデータ入力項目を削除することで、単純データ・グリッドへの新しい挿入が受け入れられ、容量限度が保持されます。最大容量をデータ・グリッドに設定しても再始動は必要ありませんが、単純グリッドで LRU オプションを選択した場合は、変更を有効にするためにグリッドが自動的に再始動されます。これは、単純グリッドで LRU オプションをオフにする場合でも同様です。

手順

- ユーザー・インターフェースで「データ・グリッド」 > 「`data_grid_type`」 > 「`data_grid_name`」 > 「拡張属性の表示」とクリックします。
- 「このデータ・グリッドの容量を制限する」を選択します。
- 単純データ・グリッドの最大容量を設定する際、最長未使用時間のデータ入力項目を犠牲にして、グリッドが新規挿入操作を拒否せずに受け入れるようにするには、「このデータ・グリッドに最長未使用時間 (LRU) の除去を使用」を選択します。「変更の適用」をクリックして保存します。再始動を完了するためにグリッド内のデータが失われる旨の警告が表示されます。
- 現在の使用容量を確認し、選択したデータ・グリッドに対して定義する最大容量を決定します。また、集合の合計容量を超えていないことも確認できます。



このグラフでは、構成する現行データ・グリッド (MyGrid) の現在の使用容量が 900 メガバイトです。このデータ・グリッドに対して現在構成されている容量限度は 2000 メガバイトです。集合のレベルでは、その集合の合計容量は 4000 メガバイトです。さらに、容量が制限されているデータ・グリッドの構成の容量限度の合計は 3400 メガバイトです。これらのグリッドの現在の使用容量は 2900 メガバイトです。最後に、この集合内で、容量限度が定義されていないデータ・グリッドが少なくとも 1 つ存在します。容量限度が未定義のこれらのデータ・グリッドによる使用容量は約 900 メガバイトです。

- 基本データの使用量限度の値を MB 単位で入力します。Enter キーを押すと、基本データとレプリカ・データの最大使用可能容量が表示されます。この数値は、定義したレプリカの数に応じて変化します。ただし、レプリカ数は集合内のアプライアンスの数によって制限される点に注意してください。4 つのレプリカが定義されており、集合内に 3 つのアプライアンスがある場合、集合には 1 つの基本データと 2 つのレプリカが含まれます。
- 「変更の適用」をクリックして構成を保存します。新しい限度を有効にするためにデータ・グリッドを再始動する必要はありません。

例

容量限度の例: 複数データ・グリッド

集合内でデータ・グリッド A、B、および C が定義されており、合計ストレージ容量が 600 ギガバイトであるとし、ます。どのデータ・グリッドでもレプリカは定義されていません。データ・グリッド A の容量限度は 100 ギガバイトです。データ・グリッド B の容量限度は 50 ギガバイトです。データ・グリッド C の容量限度は 200 ギガバイトです。このシナリオでは、集合内では常に 250 ギガバイト以上の未使用容量があることになります。3 つのデータ・グリッドの合計サイズが 350 ギガバイトを超えることはできません。

容量限度の例: レプリカ

2 つのアプライアンスが含まれている集合で、データ・グリッド A が定義されています。データ・グリッド A には、3 つのレプリカ (1 つの同期レプリカと 2 つの非同期レプリカ) があります。このグリッドの容量限度は 100 メガバイトとして定義されています。最初はこのグリッドの最大使用容量は 200 メガバイトです。この集合に含まれているアプライアンスが 2 つのみであるため、1 つの基本データと 1 つのレプリカ・コピーが存在します。基本データ・グリッドは 100 メガバイトまで使用できます。レプリカは基本データ・グリッドと同じ割合で増大するため、最大使用容量合計が 200 メガバイトになります。集合に 3 つ目のアプライアンスが追加されると、2 つ目のデータ・レプリカ・コピーが配置されます。グリッドの最大使用量は、1 つの基本データと 2 つのレプリカによって 300 メガバイトになります。

容量限度の例: 容量限度が未定義のデータ・グリッド


集合内でデータ・グリッド A、B、および C が定義されており、合計ストレージ容量が 600 ギガバイトであるとし、ます。データ・グリッド A の容量限度は 100 ギガバイトです。データ・グリッド B の容量限度は 50 ギガバイトです。データ・グリッド C には容量限度が定義されていません。3 つのデータ・グリッドのいずれでもレプリカが定義されていません。データ・グリッド C には限度が定義されていないため、このデータ・グリッドによって、使用可能な容量全体 (600 ギガバイト) が使用される可能性があります。この場合、データ・グリッド A とデータ・グリッド B ではデータを挿入できなくなります。データ・グリッド A またはデータ・グリッド B で挿入されたデータはすべて保持されますが、これらのデータ・グリッドの容量限度に達することは保証されません。データ・グリッド C の最小使用可能容量として 450 ギガバイトが保証されます。これは、システムのその他のデータ・グリッドによる使用容量が、600 ギガバイトの容量のうち、これらのデータ・グリッドの容量限度合計である 150 ギガバイトを超えることがないためです。この 450 ギガバイトを算出する計算では、レプリカ・データによる使用容量が無視されています。2 つ以上の無制限データ・グリッドが集合内にある場合、特定のデータ・グリッドの使用可能容量は保証されません。

親トピック: [データ・グリッドの構成](#)

データ・グリッドのクリア

データ・グリッド内のすべての項目を恒久的に削除することができます。失効した情報またはテスト項目を除去するために、データ・グリッドをクリアできます。

手順

1. ユーザー・インターフェースで、「データ・グリッド」 > 「**data_grid_type**」 > 「**data_grid_name**」をクリックします。
2. 「グリッドのクリア」アイコン()をクリックして、データ・グリッドのすべての項目を削除します。データ・グリッド内のすべての項目を除去することを確認する必要があります。
3. ユーザー・インターフェースで、データ・グリッド項目が削除されたことを確認できます。「モニター」 > 「個別データ・グリッドの概要」 > 「**data_grid_name**」をクリックして、「使用容量対キャッシュ・エントリ数」チャートを表示します。単純データ・グリッドおよびセッション・データ・グリッドの場合、データ・グリッド内の項目数はゼロに近づくはずですが、動的キャッシュ・データ・グリッドでは、データ・グリッドの一部の項目が残ります。これらのデータ・グリッド項目には、動的キャッシュ・データ・グリッドの構成情報が含まれます。

親トピック: [データ・グリッドの構成](#)

データ・グリッドの除去

データ・グリッド (data grid)のデータを初期化する場合は、データ・グリッド (data grid)を除去してから データ・グリッド (data grid)を再作成します。

このタスクについて

手順

1. ユーザー・インターフェースで、「データ・グリッド」 > 「**data_grid_type**」をクリックします。削除する **data_grid_name** を選択します。
2. 除去プロセスを開始するには、「除去」アイコン (✖) をクリックします。メッセージ・ボックスが表示され、このデータ・グリッド (data grid)が完全に除去されることに対する確認が要求されます。「OK」をクリックして、除去を確認します。
3. データ・グリッド (data grid)の除去は、「タスク」ビューでモニターできます。

タスクの結果

親トピック: [データ・グリッドの構成](#)

Spring キャッシュ・プロバイダーの構成

Spring Framework バージョン 3.1 では、新しいキャッシュの抽象化が導入されています。この新しい抽象化を使用すると、既存の Spring アプリケーションに対してキャッシュを容易に追加できます。WebSphere® DataPower® XC10 アプライアンス をキャッシュ抽象化のキャッシュ・プロバイダーとして使用できます。

始める前に

- Spring Framework バージョン 3.1 以降を使用するアプリケーションが必要です。
- ご使用のアプリケーションで注釈を使用してキャッシュ方式を宣言する必要があります。キャッシュ抽象化のためにご使用のアプリケーションを更新する方法について、詳しくは [Spring Framework Reference Documentation : Cache abstraction](#) を参照してください。
- ogclient.jar ファイルが Spring アプリケーションのクラスパスにあることを確認してください。
- アプリケーションを実行する JVM が、WebSphere eXtreme Scale クライアントによってインストールされる JVM でない場合、次の JVM 引数を追加して IBM Object Request Broker (ORB) が使用されるようにする必要があります。

```
-Djava.endorsed.dirs=wx_root/lib/endorsed
```

- ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。
- Spring アプリケーションをセキュア・データ・グリッドに接続する場合、適切な client.properties ファイルを **client-security-config** パラメーターの値として指定する必要があります。このパラメーターは、Spring Inversion of Control (IoC) コンテナ構成にある ObjectGridCatalogServiceDomainBean に指定します。クライアント認証を使用し、セキュアなネットワーク・トランスポート用の TLS を使用するように Spring キャッシュ・プロバイダーを構成することができます。詳しくは、[クライアント・プロパティ・ファイル](#)、[クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)、および[データ・グリッド・アプリケーション用の TLS の構成](#)を参照してください。
- カタログ・サーバーのホスト名とアプライアンスのポートが分かっている必要があります。この情報を得るには、ユーザー・インターフェースで「集合」 > 「メンバー」をクリックします。

このタスクについて

Spring Framework でキャッシュ抽象化を使用することで、Spring アプリケーションのメソッドの実行回数を減らすことができます。構成後は、特定のメソッドの結果はキャッシュに保存されます。メソッドが再実行されると、抽象化でキャッシュがチェックされ、メソッドの結果が既にキャッシュに存在しないか確認されます。結果がキャッシュに存在すれば、結果がキャッシュから返され、メソッドは再実行されません。そのため、負荷のかかるメソッドの実行回数を減らすと共に、ご使用のアプリケーションの平均応答時間を短くすることもできます。

手順

WebSphere DataPower XC10 アプライアンス をキャッシュ・プロバイダーとして使用するよう、Spring Inversion of Control (IoC) コンテナを構成します。WebSphere DataPower XC10 アプライアンス のキャッシュ実装環境は、com.ibm.websphere.objectgrid.spring パッケージ内にあります。Spring IoC コンテナ構成で以下の Bean を定義します。

```
<bean id="wxscsDomain"
class="com.ibm.websphere.objectgrid.spring.ObjectGridCatalogServiceDomainBean"
  p:catalog-service-endpoints="CATALOG_SERVICE_ENDPOINTS"
 />

<bean id="wxsgClient" class="com.ibm.websphere.objectgrid.spring.ObjectGridClientBean"
  p:object-grid-name="OBJECT_GRID_NAME"
  p:catalog-service-domain-ref="wxscsDomain" />

<bean id="cacheManager" class="org.springframework.cache.support.SimpleCacheManager">
  <property name="caches">
    <set>
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="CACHE_NAME"
        p:map-name="MAP_NAME "
        p:object-grid-client-ref="wxsgClient" />
    </set>
  </property>
</bean>
```

CATALOG_SERVICE_ENDPOINTS

カタログ・サーバーのホスト名とポートを指定します。

クライアント・サイドで設定を変更する ObjectGrid XML ファイルの絶対パスまたは相対パスを、Spring リソースとして指定します。Spring 内のリソースの指定については、[Spring Framework Reference Documentation: Resources](#) を参照してください。

例:p:client-override-xml="file:/path/to/objectgrid.xml"

例: p:client-override-xml="classpath:com/example/app/override-objectgrid.xml"

例: p:client-override-xml="http://myserver/override-objectgrid.xml"

例: p:client-override-xml="ftp://myserver/override-objectgrid.xml"

CLIENT_SECURITY_CONFIG (オプション)

client.properties ファイルの絶対パスまたは相対パスを、Spring リソースとして指定します。Spring 内のリソースの指定については、[Spring Framework Reference Documentation: Resources](#) を参照してください。WebSphere DataPower XC10 アプライアンスの client.properties の作成については、[クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)を参照してください。

例: p:client-security-config="file:/path/to/client.properties"

OBJECT_GRID_NAME

ObjectGrid 名を指定します。提供された XML 構成ファイルを使用してコンテナ・サーバーを開始する場合、このパラメーターは不要です。この値は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。

CACHE_NAME

Spring キャッシング・アプリケーションで指定されるキャッシュの名前を指定します。

MAP_NAME

キャッシュのバッキング・マップの名前を指定します。この値は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。デフォルト値以外のマップ名を使用する場合は、動的マップを定義することができます。動的マップの作成については、[動的マップの構成オプション](#)を参照してください。

例

以下のスニペットは、myXC10.myhost.com:2809 でアプライアンスがホスティングする default という名前のキャッシュを作成します。この例は、データ・グリッドにちなんで名前が付けられるデフォルト・マップ・インスタンスを使用しています。

```
<bean id="wxsCSDomain"
class="com.ibm.websphere.objectgrid.spring.ObjectGridCatalogServiceDomainBean"
  p:catalog-service-endpoints ="myXC10.myhost.com:2809" />
<bean id="wxsGridClient" class="com.ibm.websphere.objectgrid.spring.ObjectGridClientBean"
  p:object-grid-name="my_simple_data_grid"
  p:catalog-service-domain-ref="wxsCSDomain" />
<bean id="cacheManager" class="org.springframework.cache.support.SimpleCacheManager">
  <property name="caches">
    <set>
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="default"
        p:map-name="my_simple_data_grid"
        p:object-grid-client-ref="wxsGridClient" />
    </set>
  </property>
</bean>
```

親トピック: [データ・グリッドの構成](#)

クライアントの構成

クライアント上の一部のプロパティは、サーバーで設定されるプロパティをオーバーライドすることで構成できます。これらのプロパティは、クライアントのプロパティ・ファイルでオーバーライドするか、またはプログラムでオーバーライドすることができます。

Java クライアントの構成

スタンドアロン環境または WebSphere® Application Server を使用した環境で実行するように WebSphere eXtreme Scale を構成できます。グリッドのサーバー・サイドの構成変更を WebSphere eXtreme Scale デプロイメントに反映するには、動的に適用するのではなく、プロセスを再始動して変更を有効にする必要があります。一方、クライアント・サイドでは、既存のクライアント・インスタンスの構成設定は変更できませんが、XML ファイルを使用するか、プログラムで、新しいクライアント・インスタンスを必要な設定で作成できます。クライアントの作成時には、現行のサーバー構成からのデフォルト設定をオーバーライド可能です。

.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成

WebSphere eXtreme Scale Client for .NET を構成するには、クライアント・プロパティ・ファイルを使用するか、サーバー・サイド XML 構成を使用するか、または一部のサーバー・プロパティをプログラマチックにオーバーライドします。

Java クライアントの構成

スタンドアロン環境または WebSphere® Application Server を使用した環境で実行するように WebSphere eXtreme Scale を構成できます。グリッドのサーバー・サイドの構成変更を WebSphere eXtreme Scale デプロイメントに反映するには、動的に適用するのではなく、プロセスを再始動して変更を有効にする必要があります。一方、クライアント・サイドでは、既存のクライアント・インスタンスの構成設定は変更できませんが、XML ファイルを使用するか、プログラムで、新しいクライアント・インスタンスを必要な設定で作成できます。クライアントの作成時には、現行のサーバー構成からのデフォルト設定をオーバーライド可能です。

次の方法で、eXtreme Scale クライアント (Java クライアントのみ) を構成することができます。いずれの方法も、クライアント・オーバーライド XML ファイルを使用して、もしくはプログラムで実行することができます。

- XML 構成
- プログラマチック構成
- Spring Framework 構成
- ニア・キャッシュの使用不可化

[Java クライアントのオーバーライド](#)

サーバーの設定をオーバーライドすることにより、要件に基づいて WebSphere eXtreme Scale クライアントを構成することができます。いくつかのプラグインおよび属性をオーバーライドすることができます。

[XML 構成を使用した Java クライアントの構成](#)

ObjectGrid 構成 XML ファイルを使用して、クライアント・サイドの設定をオーバーライドできます。

[Java クライアントのプログラマチック構成](#)

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

[要求タイムアウト値および再試行タイムアウト値の構成](#)

eXtreme Scale クライアント・コードが要求の完了をどれくらい待ち、このクライアント・コードがデータ・グリッドへのアクセスと関連付けられる要求をいつまで再試行するかを制御する調整オプションを指定することができます。

[データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

セッション管理を使用し、WebSphere Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、アプライアンス内のデータ・グリッドを使用してそれらのアプリケーションを管理するように Liberty プロファイルを構成することができます。

[REST ゲートウェイのデプロイ](#)

WebSphere Application Server または Liberty プロファイル・サーバーでデータ・グリッド用の REST ゲートウェイをデプロイして構成できます。

親トピック: [クライアントの構成](#)


Java クライアントのオーバーライド

サーバーの設定をオーバーライドすることにより、要件に基づいて WebSphere® eXtreme Scale クライアントを構成することができます。いくつかのプラグインおよび属性をオーバーライドすることができます。

クライアントの設定をオーバーライドするには、XML 構成またはプログラマチック構成のいずれかを使用することができます。クライアント設定のオーバーライドについて詳しくは、[XML 構成を使用した Java クライアントの構成](#)および[Java クライアントのプログラマチック構成](#)を参照してください。

クライアント上で以下のプラグインをオーバーライドできます。

- **BackingMap** プラグイン
 - Evictor プラグイン
 - MapEventListener プラグイン
 - BackingMapLifecycleListener プラグイン
 - MapSerializerPlugin プラグイン
- **BackingMap** 属性
 - numberOfBuckets 属性

 **非推奨:** このプロパティは非推奨です。ニア・キャッシュを使用可能にするには nearCacheEnabled 属性を使用します。

 - timeToLive 属性
 - ttlEvictorType 属性
 - evictionTriggers 属性
 - nearCacheEnabled 属性
 - nearCacheInvalidationEnabled 属性
 - nearCacheLastAccessTTLSyncEnabled 属性
- **ObjectGrid** プラグイン
 - TransactionCallback プラグイン
 - ObjectGridEventListener プラグイン
 - ObjectGridLifecycleListener プラグイン
- **ObjectGrid** 属性
 - entityMetadataXMLFile 属性
 - txTimeout 属性
 - txIsolation 属性

親トピック: [Java クライアントの構成](#)

XML 構成を使用した Java クライアントの構成

ObjectGrid 構成 XML ファイルを使用して、クライアント・サイドの設定をオーバーライドできます。

このタスクについて

WebSphere® eXtreme Scale クライアントの設定を変更するには、コンテナ・サーバーに使用したファイルと似た構造の ObjectGrid XML ファイルを作成します。

クライアントでオーバーライドできるプラグインおよび属性のリストについては、[Java クライアントのオーバーライド](#)を参照してください。

手順

1. コンテナ・サーバー用のファイルと似た構造の ObjectGrid 構成 XML ファイルを クライアント用に作成します。

以下の XML ファイルがデプロイメント・ポリシー XML ファイルと対になっており、これらのファイルを使用してコンテナ・サーバーが開始されたものと想定します。

companyGridServerSide.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<objectGridConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://ibm.com/ws/objectgrid/config ../objectGrid.xsd"
  xmlns="http://ibm.com/ws/objectgrid/config">

  <objectGrids>
    <objectGrid name="CompanyGrid">
      <bean id="TransactionCallback"
        className="com.company.MyTxCallback" />
      <bean id="ObjectGridEventListener"
        className="com.company.MyOgEventListener" />
      <backingMap name="Customer"
        pluginCollectionRef="customerPlugins" />
      <backingMap name="Item" />
      <backingMap name="OrderLine" nearCacheEnabled="true"
        timeToLive="1600" ttlEvictorType="LAST_ACCESS_TIME" />
      <backingMap name="Order" lockStrategy="PESSIMISTIC"
        pluginCollectionRef="orderPlugins" />
    </objectGrid>
  </objectGrids>

  <backingMapPluginCollections>
    <backingMapPluginCollection id="customerPlugins">
      <bean id="Evictor"
        className="com.ibm.websphere.objectgrid.plugins.builtins.LRUEvictor"
      />
      <bean id="MapEventListener"
        className="com.company.MyMapEventListener" />
    </backingMapPluginCollection>
    <backingMapPluginCollection id="orderPlugins">
      <bean id="MapIndexPlugin"
        className="com.company.MyMapIndexPlugin" />
    </backingMapPluginCollection>
  </backingMapPluginCollections>
</objectGridConfig>
```

コンテナ・サーバーでは、CompanyGrid という名前の ObjectGrid インスタンスが companyGridServerSide.xml ファイルによる 定義に従って動作します。デフォルトでは CompanyGrid クライアントの設定は、サーバーで実行している CompanyGrid インスタンスの設定と同じです。

以下の ObjectGrid XML ファイルを使用すると、CompanyGrid クライアントの属性およびプラグインのいくつかを指定できます。

companyGridClientSide.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<objectGridConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://ibm.com/ws/objectgrid/config ../objectGrid.xsd"
  xmlns="http://ibm.com/ws/objectgrid/config">

  <objectGrids>
    <objectGrid name="CompanyGrid">
      <bean id="TransactionCallback"
        className="com.company.MyClientTxCallback" />
      <bean id="ObjectGridEventListener" className="" />
      <backingMap name="Customer" nearCacheEnabled="true"
        pluginCollectionRef="customerPlugins" />
      <backingMap name="Item" />
      <backingMap name="OrderLine" nearCacheEnabled="true"
        timeToLive="800" ttlEvictorType="LAST_ACCESS_TIME" />
      <backingMap name="Order" lockStrategy="PESSIMISTIC"
        pluginCollectionRef="orderPlugins" />
    </objectGrid>
  </objectGrids>

  <backingMapPluginCollections>
    <backingMapPluginCollection id="customerPlugins">
      <bean id="Evictor"
        className="com.ibm.websphere.objectgrid.plugins.builtins.LRUEvictor"
      />
      <bean id="MapEventListener" className="" />
    </backingMapPluginCollection>
    <backingMapPluginCollection id="orderPlugins">
      <bean id="MapIndexPlugin"
        className="com.company.MyMapIndexPlugin" />
    </backingMapPluginCollection>
  </backingMapPluginCollections>
</objectGridConfig>

```

XML ファイルでは、以下のオーバーライドを定義します。

- クライアントの TransactionCallback Bean は、サーバー・サイドで設定されている com.company.MyTxCallback ではなく、com.company.MyClientTxCallback になります。
- className 値が空ストリングであるため、クライアントには ObjectGridEventListener プラグインがありません。
- クライアントは、Customer backingMap のニア・キャッシュを使用可能にし、その Evictor プラグインを保持し、MapEventListener プラグインを除去します。
- OrderLine backingMap の timeToLive 属性は変更されます。
- 異なる lockStrategy 属性が指定されていても、lockStrategy 属性はクライアント・オーバーライドでサポートされていないため、影響はありません。

2. XML ファイルを使用してクライアントを作成します。

companyGridClientSide.xml ファイルを使用して CompanyGrid クライアントを作成するには、ObjectGrid XML ファイルを URL として、ObjectGridManager インターフェースのいずれかの connect メソッドに渡します。

```

ObjectGridManager ogManager =
    ObjectGridManagerFactory.ObjectGridManager();
ClientClusterContext clientClusterContext =
    ogManager.connect("MyServer1.company.com:2809", null, new URL(
        "file:xml/companyGridClientSide.xml"));

```

親トピック: [Java クライアントの構成](#)

Java クライアントのプログラマチック構成

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

このタスクについて

以下のコード例では、[XML 構成を使用した Java クライアントの構成](#)で説明されているのと同じオーバーライドを作成します。

クライアントでオーバーライドできるプラグインおよび属性のリストについては、[Java クライアントのオーバーライド](#)を参照してください。

手順

以下のコードでは、クライアント・サイドの ObjectGrid インスタンスを作成します。

```
ObjectGridConfiguration companyGridConfig = ObjectGridConfigFactory
    .createObjectGridConfiguration("CompanyGrid");
Plugin txCallbackPlugin = ObjectGridConfigFactory.createPlugin(
    PluginType.TRANSACTION_CALLBACK, "com.company.MyClientTxCallback");
companyGridConfig.addPlugin(txCallbackPlugin);

Plugin ogEventListenerPlugin = ObjectGridConfigFactory.createPlugin(
    PluginType.OBJECTGRID_EVENT_LISTENER, "");
companyGridConfig.addPlugin(ogEventListenerPlugin);

BackingMapConfiguration customerMapConfig = ObjectGridConfigFactory
    .createBackingMapConfiguration("Customer");
customerMapConfig.setNumberOfBuckets(1429);
Plugin evictorPlugin = ObjectGridConfigFactory.createPlugin(PluginType.EVICTOR,
    "com.ibm.websphere.objectgrid.plugins.builtins.LRUEvictor");
customerMapConfig.addPlugin(evictorPlugin);

companyGridConfig.addBackingMapConfiguration(customerMapConfig);

BackingMapConfiguration orderLineMapConfig = ObjectGridConfigFactory
    .createBackingMapConfiguration("OrderLine");
orderLineMapConfig.setNumberOfBuckets(701);
orderLineMapConfig.setTimeToLive(800);
orderLineMapConfig.setTtlEvictorType(TTLType.LAST_ACCESS_TIME);

companyGridConfig.addBackingMapConfiguration(orderLineMapConfig);

List ogConfigs = new ArrayList();
ogConfigs.add(companyGridConfig);

Map overrideMap = new HashMap();
overrideMap.put(CatalogServerProperties.DEFAULT_DOMAIN, ogConfigs);

ogManager.setOverrideObjectGridConfigurations(overrideMap);
ClientClusterContext client = ogManager.connect(catalogServerEndpoints, null, null);
ObjectGrid companyGrid = ogManager.getObjectGrid(client, objectGridName);
```

ObjectGridManager の ogManager インスタンスは、overrideMap マップに組み込まれている ObjectGridConfiguration オブジェクトおよび BackingMapConfiguration オブジェクトのオーバーライドのみをチェックします。例えば、上記のコードは、OrderLine マップ上のバケットの数をオーバーライドします。ただし、そのマップに対する構成が組み込まれていないため、クライアント・サイドの Order マップは変更されないままです。

親トピック: [Java クライアントの構成](#)

要求タイムアウト値および再試行タイムアウト値の構成

eXtreme Scale クライアント・コードが要求の完了をどれくらい待ち、このクライアント・コードがデータ・グリッドへのアクセスと関連付けられる要求をいつまで再試行するかを制御する調整オプションを指定することができます。

このタスクについて

eXtreme Scale クライアントに対して、クライアントがネットワーク接続の作成を試みる時間の長さ、クライアントが区画へのデータ・グリッド要求の処理を試みる時間の長さ、およびクライアントが (アプリケーションに例外を返す前に) 区画へのデータ・グリッド要求の再試行を試みる時間の長さを制御する設定を構成することができます。

XIO および ORB における要求タイムアウト値および再試行タイムアウト値を調整するための要因

一部の調整オプションでは、使用しているトランスポートが eXtremeIO (XIO) であるかオブジェクト・リクエスト・ブローカー (ORB) であるかによって、設定する値が異なります。これらのトランスポート・レベル調整オプションはクライアントとの対話に初期影響を及ぼします。これは、これらのオプションによって、トランスポートがネットワーク・ソケット接続を試みる時間、およびデータ・グリッド操作に類似した個々のリモート・プロシージャ・コール (RPC) が完了するのに認められる時間が調整されるからです。

これらの値を調整する際は、ご使用の環境でどのくらいの最大負荷状態および定常状態が許容できるかを考慮してください。これらの時間間隔をデフォルト値より極端に下に調整すると (例えば、要求再試行タイムアウトのデフォルト値は 30 秒です)、操作は早く失敗することがあります。以下の要因を考慮してください。

- ネットワーク待ち時間
- データベースのような外部リソースとのグリッド対話の結合
- ヒープ・サイズ、ヒープ使用量、およびガーベッジ・コレクション調整ポリシーの組み合わせから生じるガーベッジ・コレクション一時停止

要求タイムアウト値および再試行タイムアウト値を調整するための ORB 設定

ORB 用に以下のタイムアウト設定があります。

com.ibm.CORBA.ConnectionTimeout

ORB がリモート・ロケーションとのソケット接続の作成を試みる時間の長さを指定します。ORB は作成した接続をキャッシュに入れます。したがって、この操作は要求のたびに毎回行われるわけではありません。

com.ibm.CORBA.RequestTimeout

ORB が RPC の完了を待機する時間の長さを指定します。

com.ibm.CORBA.FragmentTimeout

詳しくは、IBM ORB 資料を参照してください。製品はこの値のデフォルト設定を提供します。

com.ibm.CORBA.LocateRequestTimeout

詳しくは、IBM ORB 資料を参照してください。製品はこの値のデフォルト設定を提供します。

RequestTimeout および ConnectionTimeout 設定を調整する際は、デフォルト推奨値に基づく調整が適切である場合があります。要求タイムアウトの長さをどれくらいにしたいかに基づいてこれらの設定を定義する場合は、これらの設定を同じ値にすることもできます。

次の調整レベルは requestRetryTimeout です。各トランスポート・タイプで、RPC が時間内に完了しなかったためにシステム例外をスローした後、データ・グリッドは requestRetryTimeout 設定によって定義される追加時間 (例えば、要求タイムアウトが 10 秒、再試行要求タイムアウトが 20 秒) を使用して、以下のアクションを完了する時間を指定することができます。

- フェイルオーバーのために区画が他のところにある場合に非同期でカタログ・サーバーに最新のルーティング・テーブルを要求する。
- 新しい経路を取り、要求を再試行するか、または試行を停止し、例外をアプリケーションにスローする。

requestRetryTimeout プロパティはミリ秒単位で指定されます。ゼロより大きい値に設定すると、再試行可能な例外が起こったときに要求は再試行されます。値を 0 に設定すると、例外が起こったときに再試行なしで失敗します。デフォルトの動作を使用するには、このプロパティを除去するか、値を -1 に設定します。

要求再試行タイムアウトを調整するための XIO 設定

XIO には以下の強化された設定があります。

- xioTimeout 設定は、XIO トランスポートがネットワーク・ソケット接続の確立を試みる時間の長さを決定します。
- ORB の LocateRequest 設定および FragmentTimeout 設定に相当するものではありません。
- requestRetryTimeout 設定は、RPC が成功しようと試みる最大時間を制御するもので、カタログ・サーバーから新しいルーティング情報を取得したときに RPC の再試行が行われる時間の長さを制御するのに役立ちます。XIO との相違は、XIO をトランスポートとして使用すると、多くの場合、XIO は、RPC が失敗しそうになったら ORB よりもずっと早く報告することです。この失敗は、requestRetryTimeout 値が満了するよりも早く起こります。

要求再試行タイムアウトを設定する方法

要求再試行タイムアウト値はクライアント・プロパティ・ファイルまたはセッション内で構成できます。セッションの値がクライアント・プロパティ設定に優先します。ゼロより大きい値に設定された場合、タイムアウト条件が満たされるか、永続的な障害が起こるまで、要求は試行されます。永続的な障害とは、DuplicateKeyException 例外などです。値ゼロはフェイル・ファースト・モード設定を表し、データ・グリッドは、どのようなタイプのトランザクションの後であっても、トランザクションを再試行しません。

トランザクション・タイムアウトと要求再試行タイムアウト

実行時は、トランザクション・タイムアウト値が要求再試行タイムアウト値と一緒に使用され、要求再試行タイムアウトがトランザクション・タイムアウトを超えないようにします。

トランザクションには 2 つのタイプがあります。自動コミット・トランザクションと明示的に begin メソッドと commit メソッドを使用するトランザクションです。再試行の有効な例外は、これら 2 つのタイプのトランザクション間で異なります。

- セッション内で呼び出されるトランザクションの場合は、ORB CORBA SystemException (XIO の場合は TransportException) および eXtreme Scale クライアント TargetNotAvailable 例外であれば、トランザクションが再試行されます。
- 自動コミット・トランザクションは、CORBA SystemException および eXtreme Scale クライアント可用性例外の場合に再試行されます。これらの例外には、ReplicationVotedToRollbackTransactionException、TargetNotAvailable、および AvailabilityException 例外が含まれます。

アプリケーション障害やその他の永続障害はすぐに再発するので、クライアントはトランザクションを再試行しません。このような永続障害には DuplicateKeyException や KeyNotFoundException 例外があります。例外の後にはトランザクションを再試行せず、すべての例外を返すようにするには、フェイル・ファースト設定を使用します。

クライアントがトランザクションを再試行する例外

- ReplicationVotedToRollbackTransactionException (自動コミットの場合のみ)
- TargetNotAvailable
- org.omg.CORBA.SystemException (この ORB システム例外の XIO 相当物は TransportException です。)
- AvailabilityException (自動コミットの場合のみ)
- LockTimeoutException (自動コミットの場合のみ)
- UnavailableServiceException (自動コミットの場合のみ)

トランザクションが再試行されない永続的な例外

- DuplicateKeyException
- KeyNotFoundException
- LoaderException
- TransactionAffinityException
- LockDeadlockException
- OptimisticCollisionException

手順

- 要求再試行タイムアウト値をクライアント・プロパティ・ファイル内に設定します。

クライアントの requestRetryTimeout 値を設定するには、[クライアント・プロパティ・ファイル](#)内の requestRetryTimeout プロパティを追加または変更します。クライアント・プロパティは、デフォルトでは objectGridClient.properties ファイルです。requestRetryTimeout プロパティはミリ秒単位で指定されます。ゼロより大きい値に設定すると、再試行可能な例外が起こったときに要求は再試行されます。値を 0 に設定すると、例外が起こったときに再試行なしで失敗します。デフォルトの動作を使用するには、このプロパティを除去するか、値を -1 に設定します。objectGridClient.properties ファイル内の値の例を次に示します。

```
requestRetryTimeout = 30000
```

requestRetryTimeout 値はミリ秒で指定されます。この例の場合、値が ObjectGrid インスタンスで使用されると、requestRetryTimeout 値は 30 秒です。

- 要求再試行タイムアウト値をプログラマチックに設定します。

クライアント・プロパティをプログラマチックに設定するには、まず最初にアプリケーションの適切な <ロケーション>にクライアント・プロパティ・ファイルを作成します。以下の例では、クライアント・プロパティ・ファイルは、前のセクションの objectGridClient.properties スニペットを参照します。ObjectGridManager インスタンスに接続した後、前述のようにクライアント・プロパティを設定します。その後、ObjectGrid インスタンスを取得すると、ファイルで定義したクライアント・プロパティがインスタンスに設定されています。クライアント・プロパティ・ファイルを変更する場合は、そのたびに新しい ObjectGrid インスタンスを明示的に取得してください。

```

ObjectGridManager manager = ObjectGridManagerFactory.getObjectGridManager();
String objectGridName = "testObjectGrid";
URL clientXML = null;
ClientClusterContext ccc = manager.connect("localhost:2809", null, clientXML);
File file = new File("<location>/objectGridClient.properties");
URL url = file.toURI().toURL();
ccc.setClientProperties(objectGridName, url);
ObjectGrid objectGrid = ogManager.getObjectGrid(ccc, objectGridName);

```

- セッション・コミット時に指定変更ファイルを設定します。

要求再試行タイムアウト をセッションに設定するか、requestRetryTimeout クライアント・プロパティをオーバーライドするには、Session インターフェースの setRequestRetryTimeout(long) メソッドを呼び出します。

```

Session sessionA = objectGrid.getSession();
sessionA.setRequestRetryTimeout(30000);
ObjectMap mapA = sessionA.getMap("payroll");
String key = "key:" + j;
mapA.insert(key, "valueA");

```

クライアント・プロパティ・ファイルに設定されている値に関係なく、このセッションでは現在 30000 ms または 30 秒という requestRetryTimeout 値が使用されています。セッション・インターフェースの詳細については、[セッションを使用したグリッド内データへのアクセス](#)を参照してください。

例

次の例を考えてみてください。この例では、短いタイムアウト値を設定した結果として、クライアントがサーバー上のネットワーク待ち時間、ガーベッジ・コレクション、一般競合を扱えるようになっています。requestRetryTimeout プロパティは 10 秒で、xioTimeout プロパティは ORB ConnectionTimeout 値 (5 秒) と一致します。

表 1. ORB トランスポート・タイプのデータ・グリッド構成と eXtremeIO トランスポート・タイプのデータ・グリッド構成

グリッド・タイプ	ORB	XIO
eXtreme Scale API に直接アクセスする Java™ または .NET クライアント・アプリケーション	<ul style="list-style-type: none"> • クライアント・アプリケーションの orb.properties ファイルを変更します。以下の値を設定します。 <ul style="list-style-type: none"> ◦ com.ibm.CORBA.RequestTimeout=5 ◦ com.ibm.CORBA.ConnectTimeout=5 ◦ com.ibm.CORBA.FragmentTimeout=5 ◦ com.ibm.CORBA.LocateRequestTimeout=5 <p>注: WebSphere Application Server では、ORB 設定の制御は、orb.properties ファイルではなくデプロイメント・マネージャーを使用していきます。</p> <ul style="list-style-type: none"> • クライアント・アプリケーションの objectGridClient.properties ファイルを requestRetryTimeout=7000 で変更します。 	<p>クライアント・アプリケーションの objectGridClient.properties ファイルを以下の値で変更します。</p> <ul style="list-style-type: none"> • xioRequestTimeout=50000。この値はミリ秒単位で、com.ibm.CORBA.RequestTimeout と等価です。 • xioTimeout=5。この値は秒単位で、com.ibm.CORBA.ConnectTimeout と等価です。 • requestRetryTimeout=70000。この値はミリ秒単位で、ORB トランスポートの場合にも使用されます。 • ORB の FragmentTimeout および LocateRequestTimeout には

		XIO 等値がありません。
HTTP セッション	<p>クライアント・プロパティ・ファイルを requestRetryTimeout=10000 でクラスパスに配置します。</p>	<p>クライアント・アプリケーションのクライアント・プロパティ・ファイルを以下の値で変更します。</p> <ul style="list-style-type: none"> • xioRequestTimeout=10000。この値はミリ秒単位で、com.ibm.CORBA.RequestTimeout と等価です。 • xioTimeout=5。この値は秒単位で、com.ibm.CORBA.ConnectTimeout と等価です。 • requestRetryTimeout=10000。この値はミリ秒単位で、ORB トラnsポートの場合にも使用されます。 • ORB の FragmentTimeout および LocateRequestTimeout には XIO 等値がありません。
動的キャッシュ	<p>キャッシュ・インスタンスに次のプロパティを設定します。</p> <ul style="list-style-type: none"> • com.ibm.websphere.xs.dynacache.request_retry_timeout_override=2000 <p>requestRetryTimeout 設定を、このキャッシュ・インスタンス・プロパティではなくクラスパス内のクライアント・プロパティと一緒に使用することができます (この設定がすべての動的キャッシュ・インスタンスで同じになるようにしたい場合)。</p>	

親トピック: [Java クライアントの構成](#)

データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成

セッション管理を使用し、WebSphere® Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、アプライアンス内のデータ・グリッドを使用してそれらのアプリケーションを管理するように Liberty プロファイルを構成することができます。

始める前に

このタスクには、セッション管理を必要とするアプリケーションをデプロイするための指示が含まれています。セッション管理を使用する Java アプリケーションを作成するには、[Java API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。

このタスクについて

WebSphere Application Server アプリケーションで、セッション管理にアプライアンスを使用する構成ができるように、Liberty プロファイルに対して同じ構成を実行することができます。

動的機能を持つ軽量のサーバーが必要な場合は、アプライアンスで Liberty プロファイルを使用する選択ができます。例えば、機能の単位であるフィーチャーを追加または削除することができ、それにより、特定のサーバーにロードされるランタイム環境の一部を制御することができます。Liberty プロファイルで、例えばセッションを管理するアプリケーションを実行すると、Liberty プロファイル フィーチャーの指定に使用するサーバー定義を作成することができ、これにより、サーバーとアプライアンス上のデータ・グリッドとのやり取りが制御されます。

手順

1. 次のコマンドを実行して、サーバー定義を作成します。

```
wlp_install_root/bin/server create server_name
```

2. サーバー定義の下で server.xml ファイルを見つけて、それを XML エディターで開きます。
3. セッション・アプリケーション、例えば *your_application.jar* を `/wlp/usr/servers/defaultServer/apps` ディレクトリーに配置します。
4. アプライアンスのモニター・コンソールを開始して、「データ・グリッド」 > 「セッション・グリッド」をクリックします。
5. アプライアンスに、`session` という名前のセッション・グリッドを作成します。
6. コマンド行で **JAVA_HOME** プロパティーをエクスポートします。必ず、Liberty プロファイルがインストールされているディレクトリーからコマンドを出してください。
7. 次のコマンドを使用して、Liberty プロファイルを開始します。

```
./server start servername
```

PID が表示されます。

8. 次の URL を使用してセッション・アプリケーションを開きます。

```
http://server:port/A/
```

9. データがアプライアンス内のセッション・グリッドに書き込まれていることを検証するには、セッション・テストを実行します。

SSL を有効に設定していないサーバー定義ファイル

以下に示す、SSL を有効に設定していない基本的な server.xml ファイルの例を参照してください。この例は、印刷の都合上、複数行で表示されています。

注: Web フィーチャーは推奨されません。代わりに webApp フィーチャーを使用してください。webApp フィーチャーをサーバー定義に追加し、セッション・マネージャーを構成した場合、Liberty プロファイルで実行される WebSphere® eXtreme Scale アプリケーションでセッション複製を使用できます。

次の例を参照してください。この例では、webApp フィーチャーを使用しています。

2.5+

```
<server description="new server">
  <!-- Enable features -->
  <featureManager>
```



```

    <feature>jsp-2.2</feature>
    <feature>eXtremeScale.server-1.1</feature>
    <feature>eXtremeScale.webApp-1.1</feature>
</featureManager>

<httpEndpoint id="defaultHttpEndpoint"
    host="*"
    httpPort="9080"
    httpsPort="9443" />

<xsWebApp objectGridName="session" catalogHostPort="remoteHost:2809"
securityEnabled="false" />
<applicationMonitor updateTrigger="mbean"/>
<application id="A" location="A.ear" name="A" type="ear"/>
<httpSession cloneId="A_test"/>

<!-- <com.ibm.ws.xs.server.config catalogServer="true" /> -->
</server>

```

[SSL を有効にしてアクセスするデータ・グリッド用の Liberty プロファイルの構成](#)

WebSphere Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、HTTPS 用の Liberty プロファイルを構成することができ、これにより、保護された Web サーバーでは SSL とデータ暗号化が自動的に使用されます。

[クライアントと一緒に実行する Liberty プロファイルの構成](#)

WebSphere eXtreme Scale クライアント・フィーチャーを使用して、eXtreme Scale クライアントで Liberty プロファイルを実行します。

[Liberty プロファイルでの webApp フィーチャーの使用可能化](#)

Liberty プロファイル・サーバーは、フォールト・トレランス用に HTTP セッション・データを複製するためにアプリケーションのデータをキャッシュに入れるデータ・グリッドをホストできます。

[Liberty プロファイル xsDynacacheApp フィーチャー・プロパティ](#)

データ・グリッドをホストするための Liberty プロファイルを指定します。データ・グリッドは、このフィーチャーを使用して動的キャッシュ・プロバイダーとして構成することができます。

親トピック: [Java クライアントの構成](#)

SSL を有効にしてアクセスするデータ・グリッド用の Liberty プロファイルの構成

WebSphere® Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、HTTPS 用の Liberty プロファイルを構成することができ、これにより、保護された Web サーバーでは SSL とデータ暗号化が自動的に使用されます。

手順

1. 以下のコマンドを実行すると、SSL 証明書が作成されて HTTPS が使用可能に設定されます。

```
cd to lib_dir¥bin
securityUtility createSSLCertificate --server=defaultServer --password=xc10test
```

2. SSL データ暗号化を実行するように Liberty プロファイルを構成するには、以下の SSL 機能を server.xml ファイルに追加します。

2.5+

```
<featureManager>
  <feature>ssl-1.1</feature>
</featureManager>
<keyStore id="defaultKeyStore" password="{xor}MjowbTI+Kyw=" />
```

3. 次のコマンドを使用して Liberty プロファイルを開始します。

```
./server start servername
```

PID が表示されます。

4. 次の URL を使用してセッション・アプリケーションを開きます。

```
http://server:securedport/A/
```

5. データがアプライアンス内のセッション・グリッドに書き込まれていることを検証するには、セッション・テストを実行します。

SSL 対応のサーバー定義ファイルの例

印刷の都合上、一部のコードが複数行に表示されています。以下は、SSL 機能を使用する高度な server.xml ファイルの構成の例です。この例は、印刷の都合上、複数行で表示されています。

注: バージョン 2.5 以降、フィーチャー・バージョン番号が webApp-1.0 から webApp-1.1 に変わりました。

2.5+

```
<server description="new server">

  <!-- Enable features -->
  <featureManager>
    <feature>jsp-2.2</feature>
    <feature>eXtremeScale.server-1.1</feature>
    <feature>eXtremeScale.webApp-1.1</feature>
    <feature>ssl-1.1</feature>
  </featureManager>

  <httpEndpoint id="defaultHttpEndpoint"
    host="*"
    httpPort="9080"
    httpsPort="9443">
    <!--tcpOptions soReuseAddr="true" / -->
  </httpEndpoint>

  <keyStore id="defaultKeyStore" password="{xor}Jzxubys6LCs=" />

  <xsWebApp objectGridName="session" catalogHostPort="remoteHost:2809"
    securityEnabled="true"
    credentialGeneratorClass="com.ibm.websphere.objectgrid.security.plugins.builtins.UserPasswordCredentialGenerator"
    credentialGeneratorProps="xcadmin xcadmin"/>
```

```
<applicationMonitor updateTrigger="mbean"/>
<application id="A" location="A.ear" name="A" type="ear"/>
<httpSession cloneId="A_test"/>
```

```
</server>
```

次のタスク

Liberty プロファイルとデータ・グリッド・コンテナーの間の SSL 構成を設定するために、クライアント・プロパティ・ファイルでクライアント・トランスポート・タイプを指定します。可能な値は以下のとおりです。

- **TCP/IP:** クライアントが、TCP/IP 接続のみをサポートすることを示します。
- **SSL-Supported:** クライアントが TCP/IP 接続と Secure Sockets Layer (SSL) 接続の両方をサポートすることを示します。(デフォルト)
- **SSL-Required:** クライアントが SSL 接続を必要とすることを示します。

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

関連タスク:

[クライアント・セキュリティーの構成](#)

クライアントと一緒に実行する Liberty プロファイルの構成

WebSphere® eXtreme Scale クライアント・フィーチャーを使用して、eXtreme Scale クライアントで Liberty プロファイルを実行します。

始める前に

Liberty プロファイルを構成する前に以下のタスクを実行してください。

- [Liberty プロファイルおよび WebSphere eXtreme Scale をインストールします。](#)

このタスクについて

この構成では、クライアント機能のみが提供されます。サーバー機能は別のプロセスで実行されます。クライアント・フィーチャーを追加すると、アプリケーションは eXtreme Scale API にアクセスでき、またリモート・グリッドに接続できます。

このクライアント構成により、eXtreme Scale データ・グリッドを使用して Web アプリケーションの実行を行うために必要な機能が組み込まれた単一のプロセスが提供されます。クライアント・フィーチャーの追加後は、アプリケーションは eXtreme Scale API に書き込むことができます。

手順

クライアント・フィーチャーを Liberty プロファイル サーバーに追加します。以下のコードを Liberty プロファイルサーバーに追加します。

```
eXtremeScale.client-1.1
```

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

Liberty プロファイルでの webApp フィーチャーの使用可能化

Liberty プロファイル・サーバーは、フォールト・トレランス用に HTTP セッション・データを複製するためにアプリケーションのデータをキャッシュに入れるデータ・グリッドをホストできます。

このタスクについて

インストールした WebSphere® Application Server Liberty プロファイルには、セッション複製は含まれていません。ただし、Liberty プロファイルとともにデータ・グリッドを使用した場合は、セッションを複製できます。これにより、サーバーがダウンした場合に、アプリケーション・ユーザーのセッション・データが失われなくなります。

webApp フィーチャーをサーバー定義に追加し、セッション・マネージャーを構成した場合、Liberty プロファイルで実行されるデータ・グリッド・アプリケーションでセッション複製を使用できます。

手順

以下の webApp フィーチャーを Liberty プロファイル server.xml ファイルに追加します。webApp フィーチャーには、クライアント・フィーチャーが含まれていますが、サーバー・フィーチャーは含まれていません。恐らく、Web アプリケーションはデータ・グリッドから分離する必要があります。例えば、Web アプリケーション用に 1 つの Liberty プロファイルサーバーを使用し、データ・グリッドをホストするために別の Liberty プロファイルサーバーを使用します。

```
<featureManager>
<feature>eXtremeScale_webapp-1.1</feature>
</featureManager>
```

タスクの結果

Web アプリケーションは、データ・グリッドにセッション・データを永続化できるようになりました。

例

server.xml ファイルの以下の例を参照してください。これには、データ・グリッドにリモート側で接続する場合に使用する webApp フィーチャーが含まれています。

```
<server description="Airport Entry eXtremeScale Getting Started Client Web Server">
<!--
This sample program is provided AS IS and may be used, executed, copied and modified
without royalty payment by customer
(a) for its own instruction and study,
(b) in order to develop applications designed to run with an IBM WebSphere product,
either for customer's own internal use or for redistribution by customer, as part of such
an
application, in customer's own products.
Licensed Materials - Property of IBM
5724-X67, 5655-V66 (C) COPYRIGHT International Business Machines Corp. 2012
-->
<!-- Enable features -->
<featureManager>
<feature>eXtremeScale.webapp-1.1</feature>
</featureManager>

<httpEndpoint id="defaultHttpEndpoint"
host="*"
httpPort="${default.http.port}"
httpsPort="${default.https.port}" />

<xsWebApp objectGridName="session" catalogHostPort="remoteHost:2809"
securityEnabled="false" />

</server>
```

次のタスク

webApp フィーチャーには、server.xml ファイルの xsWebApp エレメントで設定できるメタ・タイプ・プロパティが含まれています。詳しくは、[Liberty プロファイル xsWebApp フィーチャー・プロパティ](#)を参照してください。

[Liberty プロファイル xsWebApp フィーチャー・プロパティ](#)


webApp フィーチャーを指定して、Liberty プロファイル Web アプリケーションを拡張します。フォールト・

トランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを追加してください。

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

Liberty プロファイル xsWebApp フィーチャー・プロパティ

webApp フィーチャーを指定して、Liberty プロファイル Web アプリケーションを拡張します。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを追加してください。

 Web フィーチャーは推奨されません。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを使用してください。

server.xml ファイルの xsWebApp エレメントで以下の属性を設定することができます。

パラメーター

objectGridName

特定の Web アプリケーションで使用する ObjectGrid インスタンスの名前を定義する文字列値。デフォルトの名前は「session」です。

eXtreme Scale コンテナ・サーバーの始動に使用する ObjectGrid XML ファイルとデプロイメント XML ファイルの両方で、このプロパティは objectGridName を反映する必要があります。

catalogHostPort

カタログ・サーバーに接続して、クライアント・サイドの ObjectGrid インスタンスを取得できます。値の形式は、host:port<,host:port> でなければなりません。host は、カタログ・サーバーが実行されているリスナー・ホストです。port は、そのカタログ・サーバー・プロセスのリスナー・ポートです。このリストは任意の長さにすることができ、ブートストラッピングにのみ使用されます。最初の実行可能なアドレスが使用されます。このパラメーターは、**catalog.services.cluster** プロパティが構成されている場合は、WebSphere® Application Server 内でオプションです。

replicationInterval

更新されたセッションを ObjectGrid に書き込む時間間隔を秒数で定義する整数値。デフォルトは 10 秒です。0 から 60 までの値を設定できます。0 の場合は、更新されたセッションを ObjectGrid に書き込むのは、各要求のサブレット・サービス・メソッド呼び出しの最後ということになります。replicationInterval 値が高いほどパフォーマンスは向上します。これは、データ・グリッドに書き込まれるアップデートの数が少ないためです。ただし、値が高いとそれだけ構成のフォールト・トレラント性が低くなります。

この設定は、objectGridType が REMOTE に設定されている場合のみ適用されます。

sessionTableSize

メモリー内に保持するセッション参照数を定義する整数値。デフォルトは 1000 です。

EMBEDDED トポロジーは既に Web コンテナと同じ層にセッション・データを持っているため、この設定は REMOTE トポロジーのみに関するものです。

セッションは、最長未使用時間 (LRU) ロジックに基づいて、メモリー内のテーブルから除去されます。メモリー内のテーブルから除去されたセッションは Web コンテナから無効化されます。しかし、データ自体はグリッドから削除されないため、そのセッションに対する後続の要求は引き続きデータを検索することができます。この値は、Web コンテナの最大スレッド・プール値よりも高く設定されなければなりません。そうすると、セッション・キャッシュでの競合を削減できます。

fragmentedSession

true または false のいずれかの文字列値。デフォルト値は true です。この設定を使用して、製品がセッション・データをエントリ全体として保管するか、それぞれの属性を個別に保管するかを制御します。

Web アプリケーションのセッションが持っている属性の数が多き場合や属性のサイズが大きき場合は、fragmentedSession パラメーターを true に設定してください。すべての属性はデータ・グリッド内の同じキーに保管されるため、セッションが持っている属性の数が少ない場合は fragmentedSession を false に設定してください。

以前のフィルター・ベースの実装環境では、このプロパティは「persistenceMechanism」と呼ばれており、設定可能な値は ObjectGridStore (フラグメント化されている場合) および ObjectGridAtomicSessionStore (フラグメント化されていない場合) でした。

securityEnabled

true または false のいずれかの文字列値。デフォルト値は false です。この設定により、eXtreme Scale クライアント・セキュリティーを使用可能にすることができます。この設定は、eXtreme Scale サーバー・プロパティ・ファイルの **securityEnabled** 設定と一致していなければなりません。これらの設定が一致しないと、例外が発生します。

credentialAuthentication

資格情報の認証を実施するか、またはサポートするかを示します。

常になし

クライアント証明書認証は実施されません。

必須

資格情報の認証は必ず実施されます。サーバーが資格情報の認証をサポートしない場合、クライアントはサーバーに接続できません。

サポートされる

(デフォルト) 資格情報の認証は、クライアントとサーバーの両方が資格情報の認証をサポートする場合のみ実施されます。

authenticationRetryCount

`com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator` インターフェースを実装するクラスの名前を指定します。このクラスを使用して、クライアントの資格情報が取得されます。デフォルト値は 0 です。

credentialGeneratorClass

`com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator` インターフェースを実装するクラスの名前。このクラスを使用して、クライアントの資格情報が取得されます。

credentialGeneratorProps

`CredentialGenerator` 実装クラスのプロパティ。このプロパティが、`setProperties(String)` メソッドを使用してオブジェクトに設定されます。`credentialGeneratorProps` 値は、**credentialGeneratorClass** プロパティの値が非ヌルの場合にのみ使用されます。

shareSessionsAcrossWebApps

セッションが Web アプリケーション間で共有されるかどうかを指定します。`true` または `false` のいずれかのストリング値として指定されます。デフォルトは `false` です。サーブレット仕様では、HTTP セッションを Web アプリケーション間で共有できないとされています。この共有を可能にするため、サーブレット仕様の拡張が提供されます。

親トピック: [Liberty プロファイルでの webApp フィーチャーの使用可能化](#)

Liberty プロファイル xsDynacacheApp フィーチャー・プロパティ

データ・グリッドをホストするための Liberty プロファイルを指定します。データ・グリッドは、このフィーチャーを使用して動的キャッシュ・プロバイダーとして構成することができます。

server.xml ファイルの xsDynacacheApp エレメントで以下の属性を設定することができます。

パラメーター

gridName

特定の動的キャッシュ・インスタンスで使用するデータ・グリッド・インスタンスの名前を定義する文字列値。

cacheName

テンプレート・マップの名前として使用される固有の接尾部の名前を指定します。例えば、次のとおりです。

```
IBM_DC_PARTITIONED.cache_name
```

mapName

動的キャッシュ・プロバイダーであるデータ・グリッドのマップの名前を指定します。

remoteDomain

Liberty プロファイル内のデータ・グリッド動的キャッシュ・プロバイダーのリモートのクライアント・ドメイン名を指定します。

clientObjectgridXML

eXtreme Scale のクライアント objectgrid.xml ファイルのファイル・ロケーションを指定します。

requestRetryTimeout

要求がタイムアウトにならないで実行できる時間 (ミリ秒) を指定します。このプロパティは、クライアント・プロパティ・ファイルで **requestRetryTimeout** プロパティが設定されている場合、その値をオーバーライドします。動的キャッシュ・インスタンスのデフォルト値は 2000 ミリ秒です。

templateName

テンプレート・マップ接頭部の名前を指定します。以下のマップ・テンプレートのいずれかを使用できます。

- IBM_DC_PARTITIONED_.* (デフォルト)
- IBM_DC_NCI_PARTITIONED_.* (このキャッシュがニア・キャッシュを使用することを示します)

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

REST ゲートウェイのデプロイ

WebSphere® Application Server または Liberty プロファイル・サーバーでデータ・グリッド用の REST ゲートウェイをデプロイして構成できます。

始める前に

Liberty プロファイル・サーバーが作成されていることを確認します。詳しくは、[Liberty プロファイルのインストール](#)を参照してください。

このタスクについて

REST ゲートウェイは、wxsRESTGateway.war Web アーカイブ (WAR) ファイルに定義されたサーブレットです。この REST ゲートウェイでは、Uniform Resource Identifier (URI) を使用して、データ・グリッド内のデータにアクセスします。

手順

1. server.xml ファイルを手動で編集するか、Liberty Profile Developer Tools を使用して、REST ゲートウェイ機能を使用可能にします。
 - Liberty プロファイル server.xml ファイルで REST ゲートウェイを使用可能にします。

```
<featureManager>
  <feature>eXtremeScale.rest-1.1</feature>
</featureManager>
```

- Liberty Profile Developer Tools を使用して、Liberty プロファイル server.xml ファイルで REST ゲートウェイを使用可能にします。
 - IBM® WebSphere Application Server バージョン 8.6 Liberty Profile Developer Tools を開始します。
 - 「設計」タブで、「フィーチャー・マネージャー」を選択します。「フィーチャー・マネージャーの詳細」セクションで「追加」をクリックします。「**eXtremeScale.rest-1.1**」フィーチャーを選択して追加します。
 - 選択したフィーチャー・マネージャーの「フィーチャー・マネージャーの詳細」セクションで「追加」をクリックします。「**servlet-3.0**」フィーチャーを選択して追加します。
 - server.xml ファイルを保存します。
 - WebSphere Application Server で REST ゲートウェイを使用可能にします。
 - WebSphere Application Server とともに WebSphere eXtreme Scale をインストールします。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
 - `was_install_root¥optionalLibraries¥ObjectGrid¥restgateway¥wxsRESTGateway.war` ファイルを WebSphere Application Server にデプロイします。
2. REST ゲートウェイを構成します。
 - a. server.xml ファイルで REST ゲートウェイを構成します。以下のコード行を入力します。

```
<xsREST contextRoot="myContextRoot" remoteDomain="myDomain"/>
```

重要: 属性 contextRoot および remoteDomain はオプションです。デフォルトのコンテンツ・ルートは resources です。

- b. eXtreme Scale サーバーを構成します。
- c. コンテナ・サービスを構成します。

コンテナ・サービスを構成するために以下のオプションが使用可能です。

- 有効な objectgrid.xml ファイルを (objectGridDeployment.xml ファイルとともに、またはこのファイルなしで) `wlp_home/usr/servers/server_name/grids` ディレクトリーにコピーします。この grids ディレクトリーは、実行時に製品によってモニターされます。このディレクトリー内のファイルが変更されると、Liberty プロファイル ランタイム環境でイベントが開始されます。例えば、新規 objectgrid.xml または objectGridDeployment.xml、あるいはその両方のファイルが見つかった場合は、新規コンテナ・サーバーが作成されます。これらのファイルの 1 つが削除されると、eXtreme Scale はそのコンテナ・サーバーを停止します。ファイルが変更されると、eXtreme Scale はコンテナを停止して再始動します。複数の断片コンテナが同じ eXtreme Scale 内に存在できます。この場合、grids ディレクトリー内にサブディレクトリーが存在する必要があります。

3. Liberty プロファイル・サーバーを始動して、REST クライアント・ゲートウェイを実行します。

次のタスク

REST ゲートウェイが使用可能になっている場合は、サーブレットに対するアクセス権を備えた任意のユーザーが、データ・グリッドのデータにアクセスできます。そのため、WebSphere Application Server で Web アプリケーション・セキュリティーを使用して、許可を制御する必要があります。この REST ゲートウェイを使用する Web アプリケーションの保護について詳しくは、WebSphere Application Server インフォメーション・センターの『[Web アプリケーションの保護](#)』を参照してください。

wxsRESTGateway.war ファイルは、ご使用のインストール済み環境に応じて、以下の場所にあります。このファイルには、セキュリティー構成用の web.xml ファイルが含まれています。

- `wlp_install_root/wxs/web/rest`
- `was_install_root/optionalLibraries/ObjectGrid/restgateway`
- `wxs_standalone_install_root/ObjectGrid/restgateway`

これで、Liberty プロファイルで REST データ・サービスの使用を開始して、URI を使用してデータ・グリッドと通信できるようになりました。詳しくは、[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。

親トピック: [Java クライアントの構成](#)

WebSphere eXtreme Scale Client for .NET の構成

WebSphere® eXtreme Scale Client for .NET を構成するには、クライアント・プロパティ・ファイルを使用するか、サーバー・サイド XML 構成を使用するか、または一部のサーバー・プロパティをプログラマチックにオーバーライドします。

このタスクについて

クライアントがデータ・グリッドへの Connect メソッドを呼び出すと、指定されたクライアント・プロパティ・ファイルを通じて構成が行われます。プロパティ・ファイルを指定しないと、Client.Net.properties ファイルが使用されます。

クライアントが Connect メソッドを呼び出した後でクライアント・プロパティ・ファイルに加えられた変更は、データ・グリッドへのクライアント接続を再始動して初めて有効になります。

Client.Net.properties ファイル内の一部のプロパティについては動的構成を構成することができます。

クライアント・プロパティ・ファイルに正しくないプロパティ値、正しくないクライアント・プロパティ・ロケーション、またはその他のエラーが含まれていると、Connect メソッドから例外が発生することがあります。また、WebSphere eXtreme Scale Client for .NET は、Client for .NET API を使用してプログラマチックにオーバーライドできるサーバー・サイド XML 構成プロパティ値をいくつか受け取ります。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET オーバーライド](#)

サーバー・プロパティをオーバーライドして WebSphere eXtreme Scale Client for .NET を構成することができます。これらのサーバー・プロパティは objectgrid.xml 構成ファイルを使用して設定されます。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET のプログラマチックな構成](#)

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET の動的構成を使用可能に設定](#)

クライアント・プロパティ・ファイル内のプロパティ値に対して行われた変更を動的に検出するように WebSphere eXtreme Scale Client for .NET を構成することができます。データ・グリッドへの既存の接続を再始動しなくても変更が有効になります。

親トピック: [クライアントの構成](#)

.NET

WebSphere eXtreme Scale Client for .NET オーバーライド

サーバー・プロパティをオーバーライドして WebSphere® eXtreme Scale Client for .NET を構成することができます。これらのサーバー・プロパティは objectgrid.xml 構成ファイルを使用して設定されます。

クライアントの設定をオーバーライドするには、WebSphere eXtreme Scale Client for .NET でプログラマチック構成を使用することができます。プログラマチック・オーバーライドについて詳しくは、[WebSphere eXtreme Scale Client for .NET のプログラマチックな構成](#)を参照してください。

クライアントで以下の属性をオーバーライドすることができます。

BackingMap 属性

- timeToLive 属性
- lockTimeout 属性

ObjectGrid 属性

- txTimeout 属性
- txIsolation 属性

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成](#)

WebSphere eXtreme Scale Client for .NET のプログラマチックな構成

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

このタスクについて

以下のコード例では、[WebSphere eXtreme Scale Client for .NET オーバーライド](#)で説明されているのと同じオーバーライドを作成します。

手順

以下のコードでは、クライアント・サイドの ObjectGrid インスタンスを作成します。

```
IGridManager gm = GridManagerFactory.GetGridManager();
ICatalogDomainInfo cdi =
gm.CatalogDomainManager.CreateCatalogDomainInfo("localhost:2809");
ctx = gm.Connect(cdi, "Sample.Client.Net.properties");
IGrid grid = gm.GetGrid( ctx, "Grid");

//Overriding grid's txTimeout value
grid.TransactionTimeout = new TimeSpan(0, 0, 30);

IGridMapPessimisticTxObject, Object gridMap;
gridMap = grid.GetGridMapPessimisticTx<Object, Object>("Map1");

//Overriding timeToLive value
gridMap.TimeToLive = new TimeSpan(0, 0, 50);

//Overriding lockTimeout value
gridMap.LockTimeout = new TimeSpan(0, 0, 20);

//Overriding txTimeout value
gridMap.Transaction.TransactionTimeout = new TimeSpan(0, 0, 40);

//Overriding txIsolation value
gridMap.Transaction.TransactionIsolationLevel = TxnIsolationLevel.ReadUncommitted;

//Calling ResetToDefaults(), resets the value of timeToLive, lockTimeout,
//txTimeout, txIsolation back to values when .NET client initialized the grid.

gridMap.ResetToDefaults();
```

プログラムで **timeToLive** 値をオーバーライドする前に、objectgrid.xml ファイル内の **ttlEvictorType** 値を LAST_ACCESS_TIME または LAST_UPDATE_TIME に設定します。WebSphere eXtreme Scale Client for .NET が **ttlEvictorType** 値を設定することなく **timeToLive** のプログラマチックなオーバーライドを試みると、例外がスローされます。TimeSpan.Zero に設定されている **txTimeout** 値は、タイムアウト制限が無いことを示しています。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成](#)

関連情報:

[IGridTransaction.TransactionTimeout プロパティ](#)

[IGridMapPessimisticAutoTx\(Of TKey, TValue\).LockTimeout プロパティ](#)

[IGridMapPessimisticAutoTx\(Of TKey, TValue\).TimeToLive プロパティ](#)

[IGridMapPessimisticTx\(Of TKey, TValue\).LockTimeout プロパティ](#)

[IGridMapPessimisticTx\(Of TKey, TValue\).TimeToLive プロパティ](#)

[IGrid.TransactionTimeout プロパティ](#)

WebSphere eXtreme Scale Client for .NET の動的構成を使用可能に設定

クライアント・プロパティ・ファイル内のプロパティ値に対して行われた変更を動的に検出するように WebSphere® eXtreme Scale Client for .NET を構成することができます。データ・グリッドへの既存の接続を再始動しなくても変更が有効になります。

このタスクについて

例外が発生した後いつまで(ミリ秒) 要求の処理を続行するかを指定するように、requestRetryTimeout プロパティを動的に構成することができます。他のプロパティはすべて読み取り専用です。読み取り専用プロパティの値を変更しても変更は有効になりません。システム・ログ・ファイルにエラーが書き込まれます。

手順

1. WebSphere eXtreme Scale Client for .NET のクライアント・プロパティ・ファイルにある enableDynamicConfiguration プロパティの値を true に設定します。

.NET 2.5+ enableDynamicConfiguration

これが true に設定されると、クライアント・プロパティ・ファイル内の requestRetryTimeout プロパティに対して行われた変更が動的に検出されます。新しい要求再試行タイムアウト値を計算するために新しいプロパティ値がただちに使用されます。

デフォルト: false

2. クライアント・プロパティ・ファイルにある requestRetryTimeout プロパティの値を更新します。

Java .NET requestRetryTimeout

例外が発生してから要求の処理をいつまで続行するかをミリ秒で指定します。以下の有効値のいずれかを使用します。

- 値が 0 の場合、要求は、フェイル・ファーストになり、内部の再試行ロジックは飛ばされます。
- 値が -1 の場合、要求再試行のタイムアウトは設定されません。すなわち、要求の所要時間は、トランザクション・タイムアウトによって制御されます。(デフォルト)
- 0 より大きい値は、要求再試行のタイムアウト値をミリ秒で示しています。正常に作成されない例外が返されます。DuplicateException などの例外が再試行されるときでも、成功しないときはその例外も返されます。トランザクション・タイムアウトは、最大待ち時間として引き続き使用されます。

タスクの結果

requestRetryTimeout クライアント・プロパティの値は、データ・グリッドへの接続を再始動しなくても動的に更新することができます。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成](#)

データ・グリッドの管理

コンソール、コマンド行、および HTTP コマンド・インターフェースを使用して、データ・グリッドを管理できます。

データの照会、表示、および無効化

モニター・コンソールおよび **xscmd** ユーティリティの QUERY インターフェースを使用して、マップから小規模なキーと値のセットを取得し、一連のデータを無効化することができます。

xscmd ユーティリティによる管理

xscmd ユーティリティを使用して、などの管理タスクを環境内で実行することができます。

HTTP コマンド・インターフェースでの管理

HTTP コマンド・インターフェースを使用して、アプライアンスでの操作の実行、アプライアンス設定の構成、およびデータ・グリッド、集合、ゾーンの管理を行えます。

データの照会、表示、および無効化

モニター・コンソールおよび **xscmd** ユーティリティの QUERY インターフェースを使用して、マップから小規模なキーと値のセットを取得し、一連のデータを無効化することができます。

始める前に

- **xscmd** を使用してデータの照会、表示、および無効化を行う場合は、**xscmd** ユーティリティをセットアップします。詳しくは、[xscmd ユーティリティによる管理](#)を参照してください。

このタスクについて

データ・グリッドの内容を照会するには、コンソールまたは **xscmd** ユーティリティを使用できます。データ・キーで正規表現を実行することで、データを照会できます。その後、同じ照会を使用してデータを無効にすることができます。正規表現の例については、[正規表現の構文](#)を参照してください。

手順

- コンソールでデータの照会、表示、または無効化を行います。
 1. コンソールで、照会ページに進みます。ユーザー・インターフェースで、「データ管理」 > 「データ・グリッド内容の照会」とクリックします。
 2. マップ内のデータを検索またはフィルター処理します。データの検索またはフィルター処理を行うには、以下のいずれかのオプションを使用します。
 - フィールドに正規表現を入力し、「検索」ボタン (🔍) をクリックします。その正規表現に一致するキーのリストが表示されます。このデータのリストは、一致するすべてのデータのサブセットである場合があります。
 - パーティション・セットで結果のフィルター処理をするには、「フィルター」ボタン (🔍) をクリックします。次に正規表現を入力して、結果のフィルター処理をするパーティションの範囲を選択します。
 3. 表示されているキーの値を表示します。「値の表示 (Show values)」を選択します。値が表に表示されます。値が長すぎて表示できない場合は、省略符号 (...) で値が切り捨てられます。完全フィールドを表示する値をクリックします。値は、テキスト・ストリングとして返されます。一部の値は人間が理解できるストリングには変換されないことがあり、16 進数が表示されます。

重要: アプリケーションは、Java™ クラスがサーバーにとって不明なオブジェクト値を保管することがあります。アプリケーションが eXtreme Data Format (XDF) を使用している場合は、これらの値が表示されます。XDF が使用されておらず、Java クラスがサーバーにとって不明な場合は、オブジェクトのクラスがサーバーで使用できなかったというメッセージが返されます。
 4. データを無効化します。データを無効化すると、データは、データ・グリッドから永続的に削除されます。

選択したキー

無効化するキーをテーブルから選択することができます。エントリーを個々にクリックすることも、あるいは、「すべて選択」チェック・ボックスをクリックする (これによりテーブル内の最大 500 個のエントリーが選択される) こともできます。削除したいエントリーを選択したら、「無効化」 > 「選択したキー」とクリックします。

照会に一致するすべてのキー

指定した正規表現に一致するすべてのデータを無効化することもできます。このオプションを使用すると、コンソールに表示される最大 500 のエントリーだけでなく、正規表現に一致する、データ・グリッド内のすべてのデータが削除されます。正規表現を使用して選択したエントリーを無効化するには、「無効化」 > 「照会に一致するすべてのキー」とクリックします。

5. マップのコンテンツ全体を削除します。「マップをクリア」をクリックします。選択したマップ内のすべての項目を削除することを確認する必要があります。
- **xscmd** ユーティリティでデータの照会、表示、または無効化を行います。

データの照会:

```
xscmd.sh -c findbykey -g <data_grid> -m <map>
-fs <find_string> [-fp <partitionid>]
```

検索ストリング値のデータ・グリッド、マップ、および正規表現を含める必要があります。区画 ID でフィルターリングすることもできます。結果は照会全体のサブセットを返します。

データの無効化:

照会によって選択されたデータを無効化するには、コマンドに **-inv** 引数を含めます。

```
xscmd -c findbykey -g <data_grid> -m <map>
-fs <find_string> [-fp <partitionid>] -inv
```

検索STRING値のデータ・グリッド、マップ、および正規表現を含める必要があります。区画 ID でフィルタリングすることもできます。無効化を実行する際、照会によって返される小規模なセットだけでなく、一致するすべての値が無効化されます。

以下のように、照会されたデータの値を表示します。

コマンドに **-rv** 引数を含めて、照会で選択されたデータの値を表示します。

```
xscmd.sh -c findbykey -g <data_grid> -m <map>
-fs <find_string> -rv
```

検索STRING値のデータ・グリッド、マップ、および正規表現を含める必要があります。区画 ID でフィルタリングすることもできます。結果では、照会全体のサブセットが返され、各キーの値が含まれています。

UNIX | **Linux** **重要:** 使用する正規表現が文字 `.*` で始まる場合、コマンドを実行する際にその文字は正常に処理されない可能性があります。この問題を解決するには、ご使用の正規表現を次のいずれかの方法でフォーマットしてください。

- 正規表現をアポストロフィ文字で囲みます: `-fs '.*'`
- 円記号 (`¥`) を使用してアスタリスク文字をエスケープします: `-fs .¥*`

例:

次の例は、Grid データ・グリッドと Map1 マップにあるすべてのエントリーを検索します。

```
xscmd -c findbykey -g Grid -m Map1 -fs ".*"
```

このコマンドは、次の結果を戻します。

```
3 matching keys were found.

Partition Key
-----
2          keyghi
4          keydef
6          keyabc
```

親トピック: [データ・グリッドの管理](#)

関連タスク:

[xscmd ユーティリティーによる管理](#)

関連資料:

[xscmd ユーティリティー・リファレンス](#)

xscmd ユーティリティーによる管理

xscmd ユーティリティーを使用して、などの管理タスクを環境内で実行することができます。

始める前に

- アプライアンスを開始する必要があります。
- クライアント・インストールから **xscmd** ユーティリティーを実行している場合: アクティブなカタログ・サーバーの IP アドレスとポート番号が必要です。ユーザー・インターフェースで、「集合」 > 「メンバー」をクリックします。集合メンバーを選択します。カタログ・サーバーの IP とポート番号が表示されます。
- クライアント・インストールから **xscmd** ユーティリティーを実行している場合: `JAVA_HOME`環境変数が、製品と一緒にインストールされたランタイム環境を使用するように設定されているか、確認してください。製品の試用版を使用している場合は、`JAVA_HOME` 環境変数を設定する必要があります。

このタスクについて

2.5+ **xscmd** ユーティリティーは、クライアント・インストールまたはアプライアンス・コマンド行インターフェースから実行できます。アプライアンス・コマンド行インターフェースから **xscmd** ユーティリティーを実行すると、ユーティリティーは集合内のカタログ・サーバーに自動的に接続します。**xscmd** ユーティリティーをアプライアンス・コマンド行インターフェースから実行するときは、環境変数の設定とトラストストアのインポートは不要です。アプライアンス・コマンド行インターフェースを使用するときは、ローカル集合にしか接続できません。リモート集合に接続するには、クライアント・インストールを使用するか、またはリモート集合内のアプライアンスの 1 つでアプライアンス・コマンド行インターフェースを使用することができます。

手順

1. クライアント・インストールから **xscmd** ユーティリティーを実行している場合: アプライアンス用のアクティブ・トラストストアをクライアントにダウンロードしてください。アプライアンス・ユーザー・インターフェースで、「集合」 > 「設定」 > 「**Transport Layer Security**」 > 「アクティブなトラストストアのダウンロード」をクリックします。デフォルトのトラストストア・ファイルは、`xsatruststore.jks` ファイルです。このファイルのデフォルトのパスワードは、`xc10pass` です。
2. オプション: クライアント認証が使用可能な場合: コマンド行ウィンドウをクライアントのインストール済み環境で開きます。コマンド行で、適切な環境変数を設定します。
3. **xscmd** ユーティリティーをアプライアンスに接続します。
 - クライアント・インストールから **xscmd** ユーティリティーを実行している場合:

クライアントのインストール済み環境の `bin` ディレクトリーから、以下のコマンドを実行します。

```
xscmd.bat|sh -ts xsatruststore.jks -tst jks -tsp xc10pass -user xadmin -pwd xadmin -cep myxc10.mycompany.com -prot TLS -cxpv IBMJSSE2 -tt TCP/IP [additional parameters]
```

- **2.5+** アプライアンス・コマンド行インターフェースから **xscmd** ユーティリティーを実行する場合:
 - a. コマンド行インターフェースに接続します。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
 - b. **xscmd** ユーティリティーを実行します。

コマンドの一般構文は以下のとおりです。

```
Console> xscmd -c <command_name> -opt1 [arg1] -opt2 [arg2] -opt3
```

次のコマンドはアプライアンスのヘルプを表示します。

```
Console> xscmd -h
```

4. さまざまな **xscmd** オプションのヘルプを表示します。アプライアンス・コマンド行インターフェースから **xscmd** ユーティリティーを実行する場合、`.bat|.sh` 拡張子は不要です。
 - 一般ヘルプを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -h`
 - **Windows** `xscmd.bat -h`
 - すべてのコマンドのリストを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -lc`
 - **Windows** `xscmd.bat -lc`
 - 特定のコマンドのヘルプを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -h command_name`
 - **Windows** `xscmd.bat -h command_name`

- コマンド・グループのリストを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -lcg`
 - **Windows** `xscmd.bat -lcg`
 - コマンド・グループ内のコマンドのリストを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -lc command_group_name`
 - **Windows** `xscmd.bat -lc command_group_name`
5. 特定のカタログ・サーバーに接続するコマンドを実行します。 アプライアンス上で実行されるデータ・グリッドに関する情報を取得するには、カタログ・サーバーの IP アドレスとポートの組み合わせを 1 つ以上指定する必要があります。 アプライアンス・コマンド行インターフェースを使用するときは、ローカル集合にしか接続できません。 リモート集合に接続するには、クライアント・インストールを使用するか、またはリモート集合内のアプライアンスの 1 つでアプライアンス・コマンド行インターフェースを使用することができます。
- 接続するのカタログ・サーバーのリストを指定します。
 - **UNIX** `./xscmd.sh -c <command_name> -cep hostname:port(,hostname:port)`
 - **Windows** `xscmd.bat -c <command_name> -cep hostname:port(,hostname:port)`
 上記のコマンドで、*command_name* は実行しようとしているコマンドの名前です。*hostname:port* 値は、カタログ・サーバーのホスト名とリスナー・ポートです。

注意:

WebSphere® DataPower® XC10 アプライアンス 環境では、次のコマンドは使用しないでください。

- **-c releaseShard**
 - **-c reserveShard**
 - **-c swapShardWithPrimary**
 - **-c suspendBalancing**
 - **-c resumeBalancing**
 - **-c teardown**
 - **-c triggerPlacement**
 - **-c enableForPlacement**
6. オプション: コマンドの実行時にタイムアウト値を設定します。 任意のコマンドでグローバル・パラメーターとして **-to** または **--timeout** オプションを使用できます。この値は、コマンドでカタログ・サーバーに接続しているときにタイムアウトになるまでの秒数を指定します。オペレーティング・システムまたは他のネットワークのタイムアウトが原因で使用可能でないことがあるカタログ・サーバーに接続している場合は、このオプションを使用すると、制御した時間に待機時間を減らすために役に立つ場合があります。

デフォルトのタイムアウト値は 30 秒に設定されています。

[xscmd ユーティリティーのためのセキュリティー・プロファイルの構成](#)

セキュリティー・プロファイルを作成すると、保存されたセキュリティー・パラメーターを使用して、セキュアな環境で **xscmd** ユーティリティーを使用できます。

親トピック: [データ・グリッドの管理](#)

関連タスク:

[データの照会、表示、および無効化](#)

関連資料:

[xscmd ユーティリティー・リファレンス](#)

xscmd ユーティリティーのためのセキュリティー・プロファイルの構成

セキュリティー・プロファイルを作成すると、保存されたセキュリティー・パラメーターを使用して、セキュアな環境で **xscmd** ユーティリティーを使用できます。

始める前に

xscmd ユーティリティーのセットアップについて詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

このタスクについて

xscmd コマンドの残り部分で **-ssp profile_name** または **--saveSecProfile profile_name** パラメーターを使用して、セキュリティー・プロファイルを保存できます。プロファイルには、ユーザー名とパスワード、資格情報生成プログラム、鍵ストア、トラストストア、およびトランスポート・タイプについての設定を含めることができます。

xscmd ユーティリティー内の **ProfileManagement** コマンド・グループには、セキュリティー・プロファイルの管理のためのコマンドが含まれます。

手順

- セキュリティー・プロファイルを保存します。

セキュリティー・プロファイルを保存するには、コマンドの残り部分で **-ssp profile_name** または **--saveSecProfile profile_name** パラメーターを使用します。このパラメーターをコマンドに追加すると、次のパラメーターが保存されます。

```
-al,--alias <alias>
-arc,--authRetryCount <integer>
-ca,--credAuth <support>
-cgc,--credGenClass <className>
-cgp,--credGenProps <property>
-cxpv,--contextProvider <provider>
-ks,--keyStore <filePath>
-ksp,--keyStorePassword <password>
-kst,--keyStoreType <type>
-prot,--protocol <protocol>
-pwd,--password <password>
-ts,--trustStore <filePath>
-tsp,--trustStorePassword <password>
-tst,--trustStoreType <type>
-tt,--transportType <type>
-user,--username <username>
```

セキュリティー・プロファイルは、`user_home¥.xscmd¥profiles¥security ¥<profile_name>.properties` ディレクトリーに保存されます。

重要: `profile_name` パラメーターでは、ファイル名拡張子 `.properties` を含めないようにしてください。この拡張子は自動的にファイル名に付加されます。

- 保存したセキュリティー・プロファイルを使用します。

保存したセキュリティー・プロファイルを使用するには、実行するコマンドに、**-sp profile_name** または **--securityProfile profile_name** パラメーターを追加します。

コマンド例: `xscmd -c listHosts -cep myhost.mycompany.com -sp myprofile`

- **ProfileManagement** コマンド・グループの中のコマンドをリストします。

次のコマンドを実行します。 **xscmd -lc ProfileManagement**

- 既存のセキュリティー・プロファイルをリストします。

次のコマンドを実行します。 **xscmd -c listProfiles -v**

- セキュリティー・プロファイルの中に保存されている設定を表示します。

次のコマンドを実行します。 **xscmd -c showProfile -pn profile_name**

- 既存のセキュリティー・プロファイルを削除します。

次のコマンドを実行します。 **xscmd -c RemoveProfile -pn profile_name**

親トピック: [xscmd ユーティリティーによる管理](#)

関連資料:

[xscmd ユーティリティー・リファレンス](#)

HTTP コマンド・インターフェースでの管理


HTTP コマンド・インターフェースを使用して、アプライアンスでの操作の実行、アプライアンス設定の構成、およびデータ・グリッド、集合、ゾーンの管理を行えます。

このタスクについて

HTTP コマンド・インターフェースを使用して、HTTP POST JSON ステートメントで操作を実行することができます。これらのステートメントを結合してスクリプトにし、構成タスクや管理タスクを自動化することも可能です。

手順

1. ユーザー・インターフェースの HTTP コマンド・インターフェースで使用可能なコマンドを表示します。

ユーザー・インターフェースで使用可能な全コマンドの表を表示するには、 (ヘルプ) > 「**HTTP コマンド・インターフェースのヘルプ**」をクリックします。コマンド名をクリックすると、「**使用法の詳細**」と「**JSON の実行依頼例**」を表示できます。

インフォメーション・センター内の全コマンドのリストを表示するには、[HTTP コマンド・インターフェースのリファレンス](#)を参照してください。

2. 実行する操作の JSON ステートメントを作成します。このステートメントには次の内容を含める必要があります。

- 実行するコマンド。
- コマンドに適切なパラメーター。

cURL ツールで次のコマンドを使用します。

```
curl -v -k -u XC_ADMIN_USER:XC_ADMIN_PWD -H "Content-Type: application/json" --data-binary 'HTTP_INTERFACE_JSON_STATEMENT' https://XC10_HOST_NAME/resources/appTaskInterface
```

以下の変数を定義します。

XC_ADMIN_USER:XC_ADMIN_PWD

WebSphere® DataPower® XC10 アプライアンス 管理者のユーザー名およびパスワードを指定します。

HTTP_INTERFACE_JSON_STATEMENT

選択可能な JSON ステートメントのいずれかを指定します。これらは、HTTP コマンド・インターフェースに対して実行依頼されます。実行したい特定のコマンドの「**HTTP コマンド・インターフェースのヘルプ**」から「**JSON の実行依頼**」ステートメントの例をコピーすることができます。コマンドは単一引用符 (') で囲む必要があります。

XC10_HOST_NAME

WebSphere DataPower XC10 アプライアンス の完全修飾ホスト名または IP アドレスを指定します。

例えば、「HTTP コマンド・インターフェースのヘルプ」から ViewAllUsers コマンドの JSON の実行依頼例をコピーすることができます。その後、次のコマンドを実行することができます。

```
curl -v -k -u xcadmin:xcadmin -H "Content-Type: application/json" --data-binary '{"task":{"stopOnTaskFailure":"true","command":"ViewAllUsers"}}' https://myXC10.mycompany.com/resources/appTaskInterface
```

3. コマンドを実行して出力を表示します。

親トピック: [データ・グリッドの管理](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連情報:

[cURL ツールおよび libcurl ツール](#)

[HTTP コマンド・インターフェースのリファレンス](#)

単純データ・グリッドにアクセスするアプリケーションの開発

マップの作成や単純データ・グリッドの読み取り、書き込み、および更新には、2つの異なるオプションを使用できます。ObjectMap API を使用する Java クライアント・アプリケーションを作成できます。また、REST ゲートウェイを使用して、単純データ・グリッドにアクセスする非 Java アプリケーションを開発することもできます。

始める前に

既存の単純データ・グリッドが必要です。単純データ・グリッドの作成については、[単純データ・グリッドの作成](#)を参照してください。

動的キャッシュまたはセッション・データ・グリッドを使用する場合、既存のアプリケーションを変更せずに使用できます。

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

カタログ・サーバーに接続したり、ObjectGrid インスタンスを取得したり、ObjectMap API を使用したりすることができます。

[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

Representational State Transfer (REST) ゲートウェイを使用して、集合がホスティングする単純データ・グリッドにアクセスできます。この REST ゲートウェイは、非 Java 環境からデータ・グリッドにアクセスする必要があるときに便利です。

.NET 2.5+ [.NET API を使用したデータ・グリッド・アプリケーションの開発](#)

Java™ アプリケーションと同じデータ・グリッドを使用する Microsoft .NET アプリケーションを開発できます。

Java API を使用したデータ・グリッド・アプリケーションの開発

カタログ・サーバーに接続したり、ObjectGrid インスタンスを取得したり、ObjectMap API を使用したりすることができます。

開始する前に

ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。

サンプル・データ・グリッド・アプリケーション・ウォークスルー

1. **ClientClusterContext** インスタンスを取得することによって、カタログ・サービスに 接続します。

指定すべきカタログ・サーバーが、作成した単純データ・グリッドのユーザー・インターフェースのページに表示されます。「データ・グリッド」 > 「単純データ・グリッド」 > 「**my_simple_data_grid**」をクリックし、「カタログ・サービス」フィールドにある値を使用します。

カタログ・サーバーに 接続するには、ObjectGridManager API の connect メソッドを使用します。使用する connect メソッドが必要とするのは、*hostname:port* という形式のカタログ・サーバー・エンドポイントのみです。*hostname:port* 値のリストをコンマで区切って、複数のカタログ・サーバー・エンドポイントを示すことができます。以下のコード・スニペットは、カタログ・サーバーへの接続方法 と ClientClusterContext インスタンスの取得方法を示します。

```
ClientClusterContext ccc =
ObjectGridManagerFactory.getObjectGridManager().connect("myXC10.myhost.com:2809",
null, null);
```

connect メソッドは、接続が成功するまで リスト内の各アプライアンスに接続しようとします。他のアプライアンスの 1 つが応答しなければ、自動フェイルオーバーが提供されます。

カタログ・サーバーへの 接続が成功すれば、connect メソッド は ClientClusterContext インスタンスを戻します。この ClientClusterContext インスタンス は、ObjectGridManager API から ObjectGrid を取得するのに必要です。

2. **ObjectGrid** インスタンスを取得します。

ObjectGrid インスタンスを 取得するには、ObjectGridManager API の getObjectGrid メソッドを使用します。getObjectGrid メソッドは、ClientClusterContext インスタンスと、データ・グリッド・インスタンスの名前との両方を必要とします。ClientClusterContext インスタンスは、カタログ・サーバーへの接続中に取得されます。データ・グリッド・インスタンスの名前は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。以下のコード・スニペットは、ObjectGridManager API の getObjectGrid メソッドを呼び出すことによってデータ・グリッドを取得する方法を示します。

```
ObjectGrid grid = ObjectGridManagerFactory.getObjectGridManager().getObjectGrid(ccc,
"my_simple_data_grid");
```

3. 必要なセキュリティー資格情報を設定します。

クライアント・セキュリティー構成を作成するほか、アプリケーションに提供するユーザー名とパスワードを使用して資格情報生成プログラムを作成します。使用するユーザー名とパスワードには、アプライアンス上のデータ・グリッドにアクセスするための 権限が必要です。権限があるユーザーの作成について詳しくは、[ユーザーおよびグループの管理](#)を参照してください。

```
file // Creates a ClientSecurityConfiguration object using the specified
ClientSecurityConfiguration clientSC =
ClientSecurityConfigurationFactory.getClientSecurityConfiguration();
clientSC.setSecurityEnabled(true);
// Creates a CredentialGenerator using the passed-in user and
password.
CredentialGenerator credGen = new
UserPasswordCredentialGenerator(username,password);
clientSC.setCredentialGenerator(credGen);
return clientSC;
```

4. **Session** インスタンスを取得します。

取得した ObjectGrid インスタンス から、Session を取得することができます。Session インスタンスは、ObjectMap インスタンスの取得とトランザクション区分の実行のために必要です。以下のコード・スニペットは、ObjectGrid API の getSession メソッドを 呼び出すことによって Session インスタンスを取得する方法を示します。

```
Session sess = grid.getSession();
```

5. ObjectMap インスタンスを取得します。

Session を取得した後、Session API の getMap メソッドを呼び出すことによって、Session インスタンスから ObjectMap インスタンスを取得することができます。getMap メソッドに渡す マップ・インスタンス名は、ユーザー・インターフェースで作成したデータ・グリッドと同じ名前です。以下のコード・スニペットは、Session API の getMap メソッドを呼び出すことによって ObjectMap を取得する方法を示します。

```
ObjectMap map1 = sess.getMap("my_simple_data_grid");
```

前の例は、データ・グリッドにちなんで名前が付けられるデフォルト・マップ・インスタンスを使用しています。次の例のように、新規マップ名を指定することもできます。

```
ObjectMap map2 = sess.getMap("my_simple_data_grid.CT.P");  
ObjectMap map3 = sess.getMap("my_new_map.NONE");
```

my_simple_data_grid.CT.P マップは作成時除去と ペシミスティック・ロックを使用するマップです。my_new_map.NONE マップには、除去もロック設定もありません。詳しくは、[動的マップの構成オプション](#)を参照してください。

6. ObjectMap メソッドを使用します。

ObjectMap インスタンスを取得した後、ObjectMap API を使用できます。ObjectMap インターフェースはトランザクション・マップであり、Session API の begin メソッドおよび commit メソッドを使用することによるトランザクション区分を必要とすることに注意してください。アプリケーションに明示的なトランザクション区分がない場合、ObjectMap 操作は自動コミット・トランザクションで実行します。

以下のコード・スニペットは、自動コミット・トランザクションでの ObjectMap API の使用方法を示しています。

```
map1.insert(key1, value1);
```

使用する鍵は、java.lang.String または Integer といった既存の Java 型にできます。値は、任意の直列化可能オブジェクト・タイプから構成できます。

以下のコード・スニペットは、明示的なトランザクション区分がある場合の ObjectMap API の使用方法を示しています。

```
sess.begin();  
map1.insert(key1, value1);  
sess.commit();
```

[Java API 資料へのアクセス](#)

WebSphere DataPower XC10 アプライアンス用の Java™ API 資料は、zip ファイルのアーカイブをダウンロードしてご使用の開発環境に取り込んで利用するか、インフォメーション・センターで閲覧できます。

Java [例: 単純データ・グリッド \(data grid\) ・アプリケーション](#)

この例では、ObjectMap API を使用して、データ・グリッド (data grid) に対する単純な作成、取得、更新、および削除操作を実行します。

[Java API を使用した動的マップの作成](#)

データ・グリッドがインスタンス化されたら、Java API を使用して動的マップを作成できます。マップ・テンプレートの事前定義セットに基づいて、動的にマップをインスタンス化することができます。

2.5+ [Java アプリケーションにおけるトランザクションのためのプログラミング](#)

トランザクションを必要とする Java アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

[データの索引付けのためのプラグイン](#)

WebSphere eXtreme Scale クライアントは、作成する索引のタイプに応じて、索引の作成のために BackingMap に追加できる組み込みプラグインを提供します。

2.5+ [クライアントへの連続照会を使用したマップ更新の通知](#)

データ・グリッドでオブジェクトまたはエントリーが挿入または更新されたときに、クライアント Java 仮想マシン (JVM) で通知を受けることができます。

親トピック: [単純データ・グリッドにアクセスするアプリケーションの開発](#)

関連タスク:

[単純データ・グリッドの作成](#)

関連資料:

[例: 単純データ・グリッド \(data grid\) ・アプリケーション
クライアント・プロパティ・ファイル](#)

関連情報:

[クライアント API 仕様](#)

Java API 資料へのアクセス

WebSphere® DataPower® XC10 アプライアンス 用の Java™ API 資料は、zip ファイルのアーカイブをダウンロードしてご使用の開発環境に取り込んで利用するか、インフォメーション・センターで閲覧できます。

このタスクについて

Java API 資料は、次のいずれかの場所からアクセスできます。

インフォメーション・センター

情報の検索には、WebSphere DataPower XC10 アプライアンス の製品情報の他に、インフォメーション・センターの API 資料を利用すると便利です。

zip ファイル・アーカイブ

このファイルは、リリースごとにダウンロードできます。その後、比較ツールを使ってリリースごとの API の変更内容を確認できます。また、objectgrid.jar ファイルをコンパイルしている場合は、ご使用の Eclipse プロジェクトに圧縮ファイルを直接リンクできます。このリンクを使って、統合開発環境に API 資料を組み込みます。

オンライン形式

オンライン形式は、IBM® の Web サイトにある API 資料の公開版です。サイトの URL は、Eclipse に直接リンクできます。現行バージョンに対するリンクは、常に最新バージョンにアップグレードされるので、修正や変更が加えられた資料を自動的に閲覧できます。

手順

- インフォメーション・センターの API 資料を閲覧します。詳しくは、

[API 資料 \(API documentation\)](#) を参照してください。

- API 資料の zip アーカイブをダウンロードします。

API 資料をダウンロードしてオフラインでご覧になる場合は、次のページから該当するリリースの zip ファイルをダウンロードしてください。[WebSphere DataPower XC10 アプライアンス wiki: API 資料 \(API documentation\)](#)

- オンライン形式の API 資料を閲覧します。常に最新バージョンにアップグレードされるリンクをブックマークするか、特定のバージョンに対するリンクを選んでご利用ください。リンク一覧については、[WebSphere DataPower XC10 アプライアンス wiki: API 資料 \(API documentation\)](#) を参照してください。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

例: 単純データ・グリッド (data grid)・アプリケーション

この例では、ObjectMap API を使用して、データ・グリッド (data grid) に対する 単純な作成、取得、更新、および削除操作を実行します。

アプリケーション例 SimpleGrid.java

```
import java.net.MalformedURLException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Date;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Random;
import java.util.HashSet;
import java.util.BitSet;
import java.util.concurrent.ConcurrentLinkedQueue;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.io.File;
import java.io.PrintWriter;
import java.io.FileWriter;
import java.io.PrintWriter;
import java.io.Serializable;
import java.io.IOException;
import java.text.DecimalFormat;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import com.ibm.websphere.objectgrid.ClientClusterContext;
import com.ibm.websphere.objectgrid.ConnectException;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectMap;
import com.ibm.websphere.objectgrid.ObjectGridException;
import com.ibm.websphere.objectgrid.ObjectGridManager;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
import com.ibm.websphere.objectgrid.ObjectGridRuntimeException;
import com.ibm.websphere.objectgrid.Session;
import com.ibm.websphere.objectgrid.config.BackingMapConfiguration;
import com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory;
import com.ibm.websphere.objectgrid.config.ObjectGridConfiguration;
import com.ibm.websphere.objectgrid.plugins.TransactionCallbackException;
import com.ibm.websphere.objectgrid.security.config.ClientSecurityConfiguration;
import com.ibm.websphere.objectgrid.security.config.ClientSecurityConfigurationFactory;
import com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator;
import
com.ibm.websphere.objectgrid.security.plugins.builtins.UserPasswordCredentialGenerator;
import com.ibm.websphere.objectgrid.server.CatalogServerProperties;

public class SimpleGrid {

    static String gridName = "test";
    static String mapName = gridName;
    static String username="xadmin";
    static String password="xadmin";
    static String hostName="localhost";
    static ObjectGrid clientGrid=null;
    static ConcurrentLinkedQueue<Session> sessions = new
ConcurrentLinkedQueue<Session>();

    static synchronized public ObjectGrid getObjectGrid() {
        if (clientGrid == null) {
            ClientClusterContext ccc = null;
            try {
                new
```

```

java.io.File(System.getProperty("java.io.tmpdir")).mkdirs();
    } catch (Throwable t) {
        t.printStackTrace();
    }
    ObjectGridManager ogm =
ObjectGridManagerFactory.getObjectGridManager();
    ClientSecurityConfiguration clientSC = getAdminClientConfig();
    List<ObjectGridConfiguration> ogConfigs=new
ArrayList<ObjectGridConfiguration>();
    ObjectGridConfiguration lclGridConfig =
ObjectGridConfigFactory.createObjectGridConfiguration(gridName);
    BackingMapConfiguration bmc =
ObjectGridConfigFactory.createBackingMapConfiguration(mapName);
    bmc.setNumberOfBuckets(0);
    lclGridConfig.addBackingMapConfiguration(bmc);
    ogConfigs.add(lclGridConfig);
    try {
        ccc = ogm.connect(hostName+":2809", clientSC, null);
    } catch (Throwable e) {
        e.printStackTrace();
    }
    if (ccc != null) {
        HashMap<String,List<ObjectGridConfiguration>> overrideMap = new
HashMap<String,List<ObjectGridConfiguration>>();
        overrideMap.put(ccc.getClusterName(),ogConfigs);
        ogm.setOverrideObjectGridConfigurations(overrideMap);
        try {
            clientGrid = ogm.getObjectGrid(ccc, gridName);
        } catch (ObjectGridRuntimeException ogre) {
            ogre.printStackTrace();
        }
    }
    }
    return clientGrid;
}

static public Session getSession() throws TransactionCallbackException,
ObjectGridException {
    Session session = sessions.poll();
    if (session == null && getObjectGrid()!=null) {
        session = getObjectGrid().getSession();
    }
    if (session == null)
        throw new IllegalStateException("unable to initialize connection
to objectgrid");
    return session;
}

static public void putSession(Session session) {
    if (session.isTransactionActive()) {
        try {
            session.rollback();
        } catch (Exception e) {
        }
    }
    sessions.add(session);
}

public static ClientSecurityConfiguration getAdminClientConfig() {
    // Creates a ClientSecurityConfiguration object using the specified file
    ClientSecurityConfiguration clientSC =
ClientSecurityConfigurationFactory.getClientSecurityConfiguration();
    clientSC.setSecurityEnabled(true);
    // Creates a CredentialGenerator using the passed-in user and password.
    CredentialGenerator credGen = new
UserPasswordCredentialGenerator(username,password);
    clientSC.setCredentialGenerator(credGen);
    return clientSC;
}

```

```

    }

    public static void main(String args[]) throws Exception {
    for (int i=0;i<args.length;i++) {
        if(args[i].startsWith("-username:")) {
            username = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-password:")) {
            password = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-gridname:")) {
            gridName = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-mapname:")) {
            mapName = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-hostname:")) {
            hostName = args[i].substring(args[i].indexOf(":")+1);
        } else {
            System.out.println("usage: SimpleGrid [optional args]");
            System.out.println("    -username:<username>");
            System.out.println("    -password:<password>");
            System.out.println("    -gridname:<gridname>");
            System.out.println("    -mapname:<mapname>");
            System.out.println("    -hostname:<hostname>");
            System.exit(1);
        }
    }
    System.out.println("-----");
    System.out.println("Simple Grid Test");
    System.out.println("-----");
    System.out.println("username      : "+username);
    System.out.println("password      : "+password);
    System.out.println("gridname      : "+gridName);
    System.out.println("mapname       : "+mapName);
    System.out.println("hostname      : "+hostName);
    System.out.println("-----");

    if (getObjectGrid() == null) {
        System.out.println("ERROR: unable to connect to objectgrid at "+hostName);
        System.exit(1);
    }

    Session session = getSession();
    ObjectMap map=session.getMap(mapName);
    session.begin();
    Object data = map.get("TestKey");
    if (data!=null)
        map.remove("TestKey");
    map.insert("TestKey", "TestValue");
    session.commit();
    putSession(session);
}
}
}

```

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連タスク:

[単純データ・グリッドの作成](#)

関連情報:

[クライアント API 仕様](#)

Java API を使用した動的マップの作成

データ・グリッドがインスタンス化されたら、Java API を使用して動的マップを作成できます。マップ・テンプレートの事前定義セットに基づいて、動的にマップをインスタンス化することができます。

始める前に

動的マップで、どの構成オプションを使用する必要があるかを決定します。詳しくは、[動的マップの構成オプション](#)を参照してください。

手順

Session.getMap(String) メソッドを呼び出します。

事前定義テンプレートの 1 つの正規表現と一致するストリングを受け渡すと、適切なマップが作成されます。

```
ObjectMap map2 = sess.getMap("my_simple_data_grid.CT.P");  
ObjectMap map3 = sess.getMap("my_new_map.NONE");
```

my_simple_data_grid.CT.P マップは、作成時除去、ペシミスティック・ロック、および、ニア・キャッシュ無効化不可を使用するマップです。my_new_map.NONE マップには、除去設定も、ロック設定も、ニア・キャッシュ無効化設定もありません。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連資料:

[動的マップの構成オプション](#)

[REST ゲートウェイの例: 動的マップの作成](#)

Java アプリケーションにおけるトランザクションのためのプログラミング

トランザクションを必要とする Java™ アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

Java

2.5+ [Java アプリケーションのためのトランザクション内のデータとの対話](#)

セッションを使用してデータと対話します (挿入操作や更新操作など)。

2.5+ [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

データがデータ・グリッド内の複数の区画に分散されている場合は、単一のトランザクションで複数の区画を読み取って更新することができます。このタイプのトランザクションは複数区画トランザクションと呼ばれ、障害の際は 2 フェーズ・コミット・プロトコルを使用してトランザクションの調整とリカバリーを行います。

2.5+ [ロックの使用](#)

ロックにはライフサイクルがあり、さまざまな種類のロックはさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

Java アプリケーションのためのトランザクション内のデータとの対話

セッションを使用してデータと対話します (挿入操作や更新操作など)。

このタスクについて

ObjectMap インターフェースには、put、get、および remove などの一般的なマップ操作が含まれています。しかし、get、getForUpdate、insert、update、および remove といった、より具体的な操作名を使用してください。これらのメソッドは、従来のマップ API より意図を正確に伝えます。

注: upsert および upsertAll メソッドが ObjectMap の put および putAll メソッドに取って代わります。データ・グリッド内のエントリがキーと値をグリッドに挿入する必要があることを BackingMap に知らせるには、upsert メソッドを使用します。BackingMap は、insert または update のいずれかを行って値をグリッドに挿入します。

また、フレキシブルな 索引付けサポートを使用することもできます。

手順

- データを挿入します。

セッションを取得した後、次のコード・フラグメントを使用して、データの挿入に Map API を使用することができます。

```
Session session = ...;
ObjectMap personMap = session.getMap("PERSON");
session.begin();
Person p = new Person();
p.name = "John Doe";
personMap.insert(p.name, p);
session.commit();
```

- データを更新します。

データの更新に Map API を使用するには、次のコード・フラグメントを使用します。

```
session.begin();
Person p = (Person)personMap.getForUpdate("John Doe");
p.name = "John Doe";
p.age = 30;
personMap.update(p.name, p);
session.commit();
```

アプリケーションでは、通常は、単純な get ではなく、getForUpdate メソッドを使用してレコードをロックします。update メソッドは、更新済みの値を実際にマップに提供するために呼び出す必要があります。update を呼び出さないと、そのマップは変更されません。

- **2.5+** 次のメソッドを呼び出すことによって、2 フェーズ・コミットメント・プロトコルでデータを挿入します。session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE); session.begin(); 次のコード・スニペットは、2 フェーズ・コミット・プロトコルを使用してグリッドで操作を作成、取得、更新、および削除する方法を示しています。

```
Session session = og.getSession();
Objectmap map1 = session.getMap("Map1");
Objectmap map2 = session.getMap("Map2");
Objectmap map3 = session.getMap("Map3");
session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE);
session.begin();
map1.insert("randKey345", "HelloMap1");
map2.insert("randKey58901", "HelloMap2");
map3.insert("randKey58", "HelloMap3");
session.commit();
```

親トピック: **2.5+** [Java アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[データ・アクセスおよびトランザクション](#)

2.5+ [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

単一のトランザクションで複数の区画を更新するアプリケーションの開発

データがデータ・グリッド内の複数の区画に分散されている場合は、単一のトランザクションで複数の区画を読み取って更新することができます。このタイプのトランザクションは複数区画トランザクションと呼ばれ、障害の際は 2 フェーズ・コミット・プロトコルを使用してトランザクションの調整とリカバリーを行います。

2.5+ [2 フェーズ・コミットとエラー・リカバリー](#)

2 フェーズ・コミット・プロトコルは、分散トランザクションに関与するすべての区画を、そのトランザクションをコミットするかロールバックするかに基づいて調整します。

2.5+ [スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクションに書き込むアプリケーションの開発](#)

スタンドアロン WebSphere® eXtreme Scale 環境で複数の区画を備えた分散データ・グリッド用のアプリケーションを作成できます。

2.5+ [トランザクションを使用する eXtreme Scale クライアント・コンポーネントの開発](#)

WebSphere eXtreme Scale リソース・アダプターは、クライアント接続管理およびローカル・トランザクション・サポートを提供します。このサポートによって、Java Platform, Enterprise Edition (Java EE) アプリケーションは、Java EE ローカル・トランザクションや eXtreme Scale API を使用して、eXtreme Scale のクライアント接続を調べたり、ローカル・トランザクションを区別したりすることができます。

親トピック: **2.5+** [Java アプリケーションにおけるトランザクションのためのプログラミング](#)

関連タスク:

Java [Java アプリケーションのためのトランザクション内のデータとの対話](#)

スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクションに書き込むアプリケーションの開発

2.5+ スタンドアロン WebSphere® eXtreme Scale 環境で複数の区画を備えた分散データ・グリッド用のアプリケーションを作成できます。

始める前に

eXtremeIO プロトコルを使用可能にします。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

制約事項: マルチ区画トランザクションへの書き込みを行うアプリケーションの開発を開始する前に、以下の制約事項に注意してください。

- 複数の区画に書き込むトランザクションでマルチマスター複製を使用することはできません。
- .NET 環境において WebSphere eXtreme Scale クライアントでマルチ区画を使用することはできません。
- ロダー・プラグインで構成された BackingMap は、マルチ区画トランザクションのマップを読み取ることはできますが、書き込むことはできません。
- ロック・ストラテジー NONE を使用している BackingMap は、マルチ区画トランザクションに参加できません。

このタスクについて

TxCommitProtocol 設定セッション API を使用して、スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクション・サポートを使用可能にします。新規 API では、以下の 2 つのオプションが用意されています。

- TxCommitProtocol.ONEPHASE: デフォルトの 1 フェーズ・コミットでトランザクションをコミットする必要があることを指示するトランザクション・コミット・プロトコル定数。このオプションを使用すると、トランザクションは複数の区画から読み取ることができ、書き込みは単一の区画にしか行えません。トランザクションが複数の区画に書き込むと、TransactionException 例外が発生します。
- TxCommitProtocol.TWOPHASE: 1 フェーズ・コミットまたは 2 フェーズ・コミットのいずれかでトランザクションをコミットする必要があることを指示するトランザクション・コミット・プロトコル定数。トランザクションが単一の区画に書き込む場合は、1 フェーズ・コミット・プロトコルが使用されます。そうではない場合は、トランザクションのコミットに 2 フェーズ・プロトコルが使用され、複数の区画への書き込み操作が行われます。

WebSphere Application Server 内で WebSphere eXtreme Scale のマルチトランザクション・サポートを構成することもできます。詳しくは、[トランザクションを使用する eXtreme Scale クライアント・コンポーネントの開発](#)を参照してください。

手順

1. データ・グリッドに接続します。詳しくは、[Java API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。
2. ObjectGrid.getSession メソッドを使用して、データ・グリッド・セッション・インスタンスを取得します。詳しくは、[Java API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。
3. 以下のコード・スニペットを設定することで、2 フェーズ・コミット・プロトコルを使用可能にします。session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE); session.begin(); 以下のコード・スニペットで、2 フェーズ・コミット・プロトコルを使用してグリッドで操作を作成、取得、更新、および削除する方法を示します。

```
Session session = og.getSession();
Objectmap map1 = session.getMap("Map1");
Objectmap map2 = session.getMap("Map2");
Objectmap map3 = session.getMap("Map3");
session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE);
session.begin();
map1.insert("randKey345", "HelloMap1");
map2.insert("randKey58901", "HelloMap2");
map3.insert("randKey58", "HelloMap3");
session.commit();
```

次のタスク

マルチ区画トランザクションに対してトレースを使用可能にすることができます。詳しくは、[ログおよびトレース・データの分析](#)を参照してください。

親トピック: **2.5+** [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

関連概念:

トランザクションを使用する eXtreme Scale クライアント・コンポーネントの開発

WebSphere® eXtreme Scale リソース・アダプターは、クライアント接続管理およびローカル・トランザクション・サポートを提供します。このサポートによって、Java™ Platform, Enterprise Edition (Java EE) アプリケーションは、Java EE ローカル・トランザクションや eXtreme Scale API を使用して、eXtreme Scale のクライアント接続を調べたり、ローカル・トランザクションを区分したりすることができます。

始める前に

eXtreme Scale 接続ファクトリーのリソース参照を作成してください。

このタスクについて

eXtreme Scale データ・アクセス API を使用した作業に関するオプションがいくつかあります。いずれの場合も、eXtreme Scale 接続ファクトリーをアプリケーション・コンポーネントに注入するか、Java Naming Directory Interface (JNDI) で検索する必要があります。接続ファクトリーが検索された後、トランザクションを区分し、eXtreme Scale API にアクセスするための接続を作成することができます。

オプションで、接続ハンドルを取得するための追加オプションを提供する `com.ibm.websphere.xs.ra.XSConnectionFactory` に `javax.resource.cci.ConnectionFactory` インスタンスをキャストすることができます。結果の接続ハンドルは、`getSession` メソッドを提供する `com.ibm.websphere.xs.ra.XSConnection` インターフェイスにキャストする必要があります。`getSession` メソッドは `com.ibm.websphere.objectgrid.Session` オブジェクト・ハンドルを返します。このハンドルにより、アプリケーションが eXtreme Scale データ・アクセス API (ObjectMap API や EntityManager API など) をどれでも使用できるようになります。

Session ハンドルと派生オブジェクトは XSConnection ハンドルが存続している限り有効です。

以下の手順を使用して eXtreme Scale トランザクションを区分することができます。それぞれの手順を混合することはできません。例えば、同じアプリケーション・コンポーネントという状況下でグローバル・トランザクション区分とローカル・トランザクション区分を混用することはできません。

手順

- 自動コミット・ローカル・トランザクションを使用します。データ・アクセス操作、またはアクティブ・トランザクションをサポートしない操作を自動的にコミットするには、以下のステップに従います。
 1. グローバル・トランザクションのコンテキスト外で `com.ibm.websphere.xs.ra.XSConnection` 接続を取得します。
 2. `com.ibm.websphere.objectgrid.Session` セッションを取得し、このセッションを使用してデータ・グリッドと対話します。
 3. 自動コミット・トランザクションをサポートするデータ・アクセス操作を呼び出します。
 4. 接続をクローズします。
- ObjectGrid セッションを使用してローカル・トランザクションを区分します。Session オブジェクトを使用して ObjectGrid トランザクションを区分するには、以下のステップに従います。
 1. `com.ibm.websphere.xs.ra.XSConnection` 接続を取得します。
 2. `com.ibm.websphere.objectgrid.Session` セッションを取得します。
 3. `Session.begin()` メソッドを使用してトランザクションを開始します。
 4. セッションを使用してデータ・グリッドと対話します。
 5. `Session.commit()` メソッドまたは `rollback()` メソッドを使用してトランザクションを終了します。
 6. 接続をクローズします。
- `javax.resource.cci.LocalTransaction` トランザクションを使用してローカル・トランザクションを区分します。`javax.resource.cci.LocalTransaction` インターフェイスを使用して ObjectGrid トランザクションを区分するには、以下のステップに従います。
 1. `com.ibm.websphere.xs.ra.XSConnection` 接続を取得します。
 2. `XSConnection.getLocalTransaction()` メソッドを使用して `javax.resource.cci.LocalTransaction` トランザクションを取得します。
 3. `LocalTransaction.begin()` メソッドを使用してトランザクションを開始します。
 4. `com.ibm.websphere.objectgrid.Session` セッションを取得し、このセッションを使用してデータ・グリッドと対話します。
 5. `LocalTransaction.commit()` メソッドまたは `rollback()` メソッドを使用してトランザクションを終了します。

6. 接続をクローズします。
- 接続をグローバル・トランザクションに登録します。この手順はコンテナ管理トランザクションにも適用されます。
 1. javax.transaction.UserTransaction インターフェースを介するか、コンテナ管理トランザクションを使用するかして、グローバル・トランザクションを開始します。
 2. com.ibm.websphere.xs.ra.XSConnection 接続を取得します。
 3. com.ibm.websphere.objectgrid.Session セッションを取得して使用します。
 4. 接続をクローズします。
 5. グローバル・トランザクションをコミットまたはロールバックします。
 - **2.5+** トランザクションで複数の区画を作成するように接続を構成します。Session オブジェクトを使用して ObjectGrid トランザクションを区分するには、以下のステップに従います。
 1. 新規 com.ibm.websphere.xs.ra.XSConnectionSpec オブジェクトを作成します。
 2. XSConnectionSpec メソッドおよび setMultiPartitionSupportEnabled メソッドを、引数 true を指定して呼び出します。
 3. com.ibm.websphere.xs.ra.XSConnection 接続を取得して、XSConnectionSpec を ConnectionFactory.getConnection メソッドに渡します。
 4. com.ibm.websphere.objectgrid.Session セッションを取得して使用します。

例

次のコード例を参照してください。このコード例は、eXtreme Scale トランザクションを区分する先行ステップを示しています。

```
// (C) Copyright IBM Corp. 2001, 2012.
// All Rights Reserved. Licensed Materials - Property of IBM.
package com.ibm.ws.xs.ra.test.ee;

import javax.naming.InitialContext;
import javax.resource.cci.Connection;
import javax.resource.cci.ConnectionFactory;
import javax.resource.cci.LocalTransaction;
import javax.transaction.Status;
import javax.transaction.UserTransaction;

import junit.framework.TestCase;

import com.ibm.websphere.objectgrid.ObjectMap;
import com.ibm.websphere.objectgrid.Session;
import com.ibm.websphere.xs.ra.XSConnection;

/**
 * This sample requires that it runs in a J2EE context in your
 * application server. For example, using the JUnitEE framework servlet.
 *
 * The code in these test methods would typically reside in your own servlet,
 * EJB, or other web component.
 *
 * The sample depends on a configured WebSphere eXtreme Scale connection
 * factory registered at of JNDI Name of "eis/embedded/wxscf" that defines
 * a connection to a grid containing a Map with the name "Map1".
 *
 * The sample does a direct lookup of the JNDI name and does not require
 * resource injection.
 */
public class DocSampleTests extends TestCase {
    public final static String CF_JNDI_NAME = "eis/embedded/wxscf";
    public final static String MAP_NAME = "Map1";

    Long          key = null;
    Long          value = null;
    InitialContext ctx = null;
    ConnectionFactory cf = null;

    public DocSampleTests() {
```

```

}
public DocSampleTests(String name) {
    super(name);
}
protected void setUp() throws Exception {
    ctx = new InitialContext();
    cf = (ConnectionFactory)ctx.lookup(CF_JNDI_NAME);
    key = System.nanoTime();
    value = System.nanoTime();
}
/**
 * This example runs when not in the context of a global transaction
 * and uses autocommit.
 */
public void testLocalAutocommit() throws Exception {
    Connection conn = cf.getConnection();
    try {
        Session session = ((XSConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
    }
    finally {
        conn.close();
    }
}

/**
 * This example runs when not in the context of a global transaction
 * and demarcates the transaction using session.begin()/session.commit()
 */
public void testLocalSessionTransaction() throws Exception {
    Session session = null;
    Connection conn = cf.getConnection();
    try {
        session = ((XSConnection)conn).getSession();
        session.begin();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
        session.commit();
    }
    finally {
        if (session != null && session.isTransactionActive()) {
            try { session.rollback(); }
            catch (Exception e) { e.printStackTrace(); }
        }
        conn.close();
    }
}

/**
 * This example uses the LocalTransaction interface to demarcate
 * transactions.
 */
public void testLocalTranTransaction() throws Exception {
    LocalTransaction tx = null;
    Connection conn = cf.getConnection();
    try {
        tx = conn.getLocalTransaction();
        tx.begin();
        Session session = ((XSConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
        tx.commit(); tx = null;
    }
    finally {
        if (tx != null) {
            try { tx.rollback(); }
            catch (Exception e) { e.printStackTrace(); }
        }
    }
}

```



```

        }
        conn.close();
    }
}

/**
 * This example depends on an externally managed transaction,
 * the externally managed transaction might typically be present in
 * an EJB with its transaction attributes set to REQUIRED or REQUIRES_NEW.
 * NOTE: If there is NO global transaction active, this example runs in auto-
commit
 * mode because it doesn't verify a transaction exists.
 */
public void testGlobalTransactionContainerManaged() throws Exception {
    Connection conn = cf.getConnection();
    try {
        Session session = ((XSCConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
    }
    catch (Throwable t) {
        t.printStackTrace();
        UserTransaction tx =
(UserTransaction)ctx.lookup("java:comp/UserTransaction");
        if (tx.getStatus() != Status.STATUS_NO_TRANSACTION) {
            tx.setRollbackOnly();
        }
    }
    finally {
        conn.close();
    }
}

/**
 * This example demonstrates starting a new global transaction using the
 * UserTransaction interface. Typically the container starts the global
 * transaction (for example in an EJB with a transaction attribute of
 * REQUIRES_NEW), but this sample will also start the global transaction
 * using the UserTransaction API if it is not currently active.
 */
public void testGlobalTransactionTestManaged() throws Exception {
    boolean started = false;
    UserTransaction tx = (UserTransaction)ctx.lookup("java:comp/UserTransaction");
    if (tx.getStatus() == Status.STATUS_NO_TRANSACTION) {
        tx.begin();
        started = true;
    }
    // else { called with an externally/container managed transaction }
    Connection conn = null;
    try {
        conn = cf.getConnection(); // Get connection after the global tran starts
        Session session = ((XSCConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
        if (started) {
            tx.commit(); started = false; tx = null;
        }
    }
    finally {
        if (started) {
            try { tx.rollback(); }
            catch (Exception e) { e.printStackTrace(); }
        }
        if (conn != null) { conn.close(); }
    }
}
/**
/**

```

```

    * This example demonstrates a multi-partition transaction.
    */
    public void
testGlobalTransactionTestManagedMultiPartition() throws Exception {
        boolean started = false;
        XSConnectionSpec connSpec = new
XSConnectionSpec();

connSpec.setWriteToMultiplePartitions(true);

        UserTransaction tx =
(UserTransaction)ctx.lookup("java:comp/UserTransaction");
        if (tx.getStatus() ==
Status.STATUS_NO_TRANSACTION) {
            tx.begin();
            started = true;
        }
        // else { called with an externally/container managed
transaction }

        Connection conn = null;
        try {
conn = cf.getConnection(connSpec); // Get
connection after the global tran starts

        Session session =

ObjectMap map = session.getMap(MAP_NAME);
map.insert(key, value); // Or various data

        if (started) {
            tx.commit(); started =
            }
        }
        finally {
            if (started) {
                try {
                catch (Exception
                }
            }
            if (conn != null) { conn.close();
            }
        }
    }
}

```

親トピック: **2.5+** [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

ロックの使用

ロックにはライフサイクルがあり、さまざまな種類のロックはさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

2.5+ [Java アプリケーションにおけるロックの構成と実装](#)

WebSphere® eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。

2.5+ [ロック・シナリオでの例外処理の実装 \(Java アプリケーション\)](#)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

Java **2.5+** [例: flush メソッドのロック順序](#)

ObjectMap インターフェイスで、コミット前に flush メソッドを呼び出すと、ロック順序に関する追加の考慮事項が発生する場合があります。通常、flush メソッドは、マップに対して行われた変更を、Loader プラグインを通じてバックエンドに強制するために使用されます。

Java **2.5+** [Java のトランザクション分離の例](#)

トランザクション分離は、1つの操作で行われた変更がどのように他の並行操作に可視になるのかを定義します。以下の例を使用して、Java アプリケーションのトランザクション分離レベルを定義することができます。

親トピック: **2.5+** [Java アプリケーションにおけるトランザクションのためのプログラミング](#)

Java アプリケーションにおけるロックの構成と実装

WebSphere® eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。

始める前に

- 使用したいロック・ストラテジーを決めます。詳しくは、[ロック・ストラテジー](#)を参照してください。
- 動的マップの構成によってロック・ストラテジーを構成することもできます。詳しくは、[ロック・ストラテジーの構成](#)を参照してください。

このタスクについて

java.lang.IllegalStateException 例外を避けるために、ObjectGrid インスタンスで initialize メソッドまたは getSession メソッドを呼び出す前に、setLockStrategy メソッドを呼び出す必要があります。

手順

1. Java™ アプリケーションでロック・ストラテジーを構成します。
 - オプティミスティック・ロック・ストラテジーの構成。setLockStrategy メソッドを使用します。

```
import com.ibm.websphere.objectgrid.BackingMap;
import com.ibm.websphere.objectgrid.LockStrategy;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
...
ObjectGrid og =
    ObjectGridManagerFactory.getObjectGridManager().createObjectGrid("test"
);
BackingMap bm = og.defineMap("optimisticMap");
bm.setLockStrategy( LockStrategy.OPTIMISTIC );
```

- ペシミスティック・ロック・ストラテジーの構成。setLockStrategy メソッドを使用します。

```
import com.ibm.websphere.objectgrid.BackingMap;
import com.ibm.websphere.objectgrid.LockStrategy;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
...
ObjectGrid og =
    ObjectGridManagerFactory.getObjectGridManager().createObjectGrid("test"
);
BackingMap bm = og.defineMap("pessimisticMap");
bm.setLockStrategy( LockStrategy.PESSIMISTIC );
```

- ロックなしストラテジーの構成。setLockStrategy メソッドを使用します。

注: ロックなしストラテジーを使用するように構成された BackingMap は、複数区画トランザクションに
関与できません。

```
import com.ibm.websphere.objectgrid.BackingMap;
import com.ibm.websphere.objectgrid.LockStrategy;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
...
ObjectGrid og =
    ObjectGridManagerFactory.getObjectGridManager().createObjectGrid("test"
);
BackingMap bm = og.defineMap("noLockingMap");
bm.setLockStrategy( LockStrategy.NONE );
```

2. ロック・タイムアウト値を構成します。BackingMap インスタンスの setLockTimeout メソッドを使用しま
す。

```
bm.setLockTimeout( 60 );
```

setLockTimeout メソッドのパラメーターは、Java プリミティブの整数で、eXtreme Scale がロック・モードを
認可されるのを待たなければならない秒数を指定します。BackingMap に構成されているロック待ちタイムアウト

ト値よりも長くトランザクションが待つ場合は、com.ibm.websphere.objectgrid.LockTimeoutException 例外が発生します。

3. ペシミスティック・ロック・ストラテジーを使用している場合は、lock メソッドを使用して、データ・グリッド内のキーをロックするか、またはキーをロックして値がデータ・グリッド内に存在するかどうかを確認することができます。これまでのリリースでは、get API および getForUpdate API を使用してデータ・グリッド内のキーをロックしていました。しかし、クライアントからデータを取得する必要がなかった場合は、大きくなる可能性がある値オブジェクトを取得してクライアントに渡すことになるため、パフォーマンスが低下します。containsKey メソッドはロックを保持しないため、ペシミスティック・ロックを使用しているときには、get および getForUpdate メソッドを使用して適切なロックを取得しなければなりません。現在は、ロックを保持しているとき、lock API により containsKey メソッドが提供されるようになりました。次の例を参照してください。
 - 次のメソッドはマップ内のキーをロックします。キーが存在する場合は true を返し、キーが存在しない場合は false を返します。

```
boolean ObjectMap.lock(Object key, LockMode lockMode);
```

- 次のメソッドはマップ内のキーのリストをロックし、true または false 値のリストを返します。キーが存在する場合は true を返し、キーが存在しない場合は false を返します。

```
List<Boolean> ObjectMap.lockAll(List keys, LockMode lockMode);
```

LockMode は列挙型で以下に示す値を取ることができ、ここでロックしたいキーを指定できます。

- SHARED、UPGRADABLE、および EXCLUSIVE

LockMode パラメーターの設定例を次に示します。

```
session.begin();
map.lock(key, LockMode.UPGRADABLE);
map.upsert();
session.commit();
```

親トピック: **2.5+** [ロックの使用](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[ロック・ストラテジーの構成](#)

関連資料:

例: [flush メソッドのロック順序](#)

ロック・シナリオでの例外処理の実装 (Java™ アプリケーション)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

手順

1. 例外をキャッチし、結果のメッセージを表示します。

```
try {
    ...
} catch (ObjectGridException oe) {
    System.out.println(oe);
}
```

結果、次の例外が表示されます。

```
com.ibm.websphere.objectgrid.plugins.LockDeadlockException: Message
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるメッセージを表示します。

2. 例外の後、トランザクションをロールバックします。

```
Session sess = ...;
ObjectMap person = sess.getMap("PERSON");
boolean activeTran = false;
try
{
    sess.begin();
    activeTran = true;
    Person p = (IPerson)person.get("Lynn");
    // Lynn had a birthday, so we make her 1 year older.
    p.setAge( p.getAge() + 1 );
    person.put( "Lynn", p );
    sess.commit();
    activeTran = false;
}
finally
{
    if ( activeTran ) sess.rollback();
}
```

コード・スニペットの finally ブロックは、予期しない例外が発生したときにトランザクションがロールバックされるようにしています。LockDeadlockException 例外のみでなく、発生する可能性のあるその他の予期しない例外もすべて処理します。finally ブロックは、commit メソッドの呼び出し時に例外が発生するケースも処理します。この例は、予期しない例外を処理する唯一の方法ではありません。アプリケーションが、発生する予期しない例外のいくつかをキャッチし、そのアプリケーション例外の 1 つを表示するケースも存在するかもしれません。適宜 catch ブロックを追加できますが、アプリケーションは、コード・スニペットがトランザクションを完了せずに終了しないようにする必要があります。

親トピック: [2.5+ ロックの使用](#)

例: flush メソッドのロック順序

ObjectMap インターフェイスで、コミット前に flush メソッドを呼び出すと、ロック順序に関する追加の考慮事項が発生する場合があります。通常、flush メソッドは、マップに対して行われた変更を、Loader プラグインを通じてバックエンドに強制するために使用されます。

この状況では、バックエンドは独自のロック・マネージャーを使用して並行性を制御するので、ロック待ち状態とデッドロックが WebSphere® eXtreme Scale クライアント ロック・マネージャーではなくバックエンドで起こることがあります。次のトランザクションについて検討します。

```
Session sess = ...;
ObjectMap person = sess.getMap("PERSON");
boolean activeTran = false;
try
{
    sess.begin();
    activeTran = true;
    Person p = (IPerson)person.get("Lynn");
    p.setAge( p.getAge() + 1 );
    person.put( "Lynn", p );
    person.flush();
    ...
    p = (IPerson)person.get("Tom");
    p.setAge( p.getAge() + 1 );
    sess.commit();
    activeTran = false;
}
finally
{
    if ( activeTran ) sess.rollback();
}
```

何かほかのトランザクションが Tom も更新し、flush メソッドを呼び出し、次に Lynn を更新したとします。この状態が発生した場合、2 つのトランザクションの以下のインターリーピングの結果、データベースはデッドロック状態になります。

```
flush の実行時に "Lynn" のトランザクション 1 に対して X ロックが認可されます。
flush の実行時に "Tom" のトランザクション 2 に対して X ロックが認可されます。
コミット処理中に "Tom" のトランザクション 1 によって、X ロックが要求されます。
(トランザクション 1 は、
トランザクション 2 によって所有されたロックを待機するのをブロックします。)
コミット処理中に "Lynn" のトランザクション 2 によって、X ロックが要求されます。
(トランザクション 2 は、
トランザクション 1 によって所有されたロックを待機するのをブロックします。)
```

この例は、flush メソッドの使用により、WebSphere eXtreme Scale クライアント 内ではなくデータベース内でデッドロックが発生することを示しています。このデッドロック例は、どのロック・ストラテジーを使用しても発生する可能性があります。アプリケーションで flush メソッドを使用している場合や、Loader が BackingMap にプラグインされている場合は、この種のデッドロックが起こらないように注意する必要があります。上記の例は、WebSphere eXtreme Scale クライアント がロック待ちタイムアウト機構を備えている もう 1 つの理由を示しています。データベース・ロックを待機するトランザクションは、WebSphere eXtreme Scale クライアント マップ・エントリーのロックを所有している間、待機し続ける可能性があります。データベース・レベルの問題により、WebSphere eXtreme Scale クライアント ロック・モードの待機時間が過大になり、LockTimeoutException 例外が発生する可能性があります。

親トピック: [2.5+ ロックの使用](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[ロック・ストラテジーの構成](#)

Java のトランザクション分離の例

トランザクション分離は、1つの操作で行われた変更がどのように他の並行操作に可視になるのかを定義します。以下の例を使用して、Java™ アプリケーションのトランザクション分離レベルを定義することができます。

ペシミスティック・ロックでの反復可能読み取り

```
map = session.getMap("Order");
session.setTransactionIsolation(Session.TRANSACTION_REPEATABLE_READ);
session.begin();

// An S lock is requested and held and the value is copied into
// the transactional cache.
Order order = (Order) map.get("100");
// The entry is evicted from the transactional cache.
map.invalidate("100", false);

// The same value is requested again. It already holds the
// lock, so the same value is retrieved and copied into the
// transactional cache.
Order order2 = (Order) map.get("100");

// All locks are released after the transaction is synchronized
// with cache map.
session.commit();
```

ペシミスティック・ロックでの読み取りコミット済み

```
map1 = session1.getMap("Order");
session1.setTransactionIsolation(Session.TRANSACTION_READ_COMMITTED);
session1.begin();

// An S lock is requested but immediately released and
// the value is copied into the transactional cache.

Order order = (Order) map1.get("100");

// The entry is evicted from the transactional cache.
map1.invalidate("100", false);

// A second transaction updates the same order.
// It acquires a U lock, updates the value, and commits.
// The ObjectGrid successfully acquires the X lock during
// commit since the first transaction is using read
// committed isolation.

Map orderMap2 = session2.getMap("Order");
session2.begin();
order2 = (Order) orderMap2.getForUpdate("100");
order2.quantity=2;
orderMap2.update("100", order2);
session2.commit();

// The same value is requested again. This time, they
// want to update the value, but it now reflects
// the new value
Order order1Copy = (Order) map1.getForUpdate("100");
```

親トピック: [2.5+ ロックの使用](#)

関連概念:

[トランザクション分離](#)

データの索引付けのためのプラグイン

WebSphere® eXtreme Scale クライアント は、作成する索引のタイプに応じて、索引の作成のために BackingMap に追加できる組み込みプラグインを提供します。

HashIndex

組み込み HashIndex である `com.ibm.websphere.objectgrid.plugins.index.HashIndex` クラスは、動的索引を作成するために BackingMap に追加可能な `MapIndexPlugin` プラグインです。このクラスは、`MapIndex` と `MapRangeIndex` の両方のインターフェースをサポートします。索引を定義し、実装すると、照会のパフォーマンスを大幅に改善できます。

[索引によるデータへのアクセス \(索引 API\)](#)

より効率的なデータ・アクセスのために索引付けを使用します。

[セッションを使用したグリッド内データへのアクセス](#)

アプリケーションは、`Session` インターフェースを介してトランザクションを開始および終了できます。`Session` インターフェースは、アプリケーションを基にした `ObjectMap` および `JavaMap` インターフェースへのアクセスも提供します。

[HashIndex プラグインの構成](#)

組み込み HashIndex である `com.ibm.websphere.objectgrid.plugins.index.HashIndex` クラスは、動的索引を使用してプログラマチックに、構成することができます。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

索引によるデータへのアクセス (索引 API)

より効率的なデータ・アクセスのために索引付けを使用します。

このタスクについて

HashIndex クラスは、組み込みアプリケーション索引インターフェースである MapIndex と MapRangeIndex の両方をサポートすることのできる組み込み索引プラグイン実装です。ユーザー独自の索引を作成することもできます。HashIndex 動的索引としてバックアップ・マップに追加し、MapIndex または MapRangeIndex の索引プロキシ・オブジェクトを取得し、その索引プロキシ・オブジェクトを使用してキャッシュ・オブジェクトを検索することができます。

ローカル・マップ内のキーを反復処理する場合は、デフォルトの索引を使用できます。この索引はまったく構成を必要としませんが、エージェントを使用するか ShardEvents.shardActivated(ObjectGrid shard) メソッドから取得した ObjectGrid インスタンスを使用して、断片に対して使用しなければなりません。

注: 分散環境では、索引オブジェクトがクライアント ObjectGrid から取得された場合、その索引のタイプはクライアント索引オブジェクトになり、すべての索引操作はリモート・サーバー ObjectGrid で実行されます。マップが区画化されている場合、索引操作は各区画でリモートに実行されます。各区画の結果はマージされてからアプリケーションに戻されます。パフォーマンスは、区画数と、各区画が戻す結果のサイズによって決まります。これらの要因が両方とも大きいと、パフォーマンスが低下することがあります。

手順

索引を使用してマップのキーと値にアクセスします。

- ローカル索引:

ローカル・マップ内のキーと値を反復処理する場合は、デフォルトの索引を使用できます。デフォルトの索引は、エージェントを使用するか、ShardEvents.shardActivated(ObjectGrid shard) メソッドから取得した ObjectGrid インスタンスを使用するかして、断片に対してのみ機能します。次の例を参照してください。

```
MapIndex keyIndex = (MapIndex)
objMap.getIndex(MapIndexPlugin.SYSTEM_KEY_INDEX_NAME);
Iterator keyIterator = keyIndex.findAll();
```

- 動的索引:

BackingMap インスタンスから動的索引を、いつでもプログラマチックに作成および除去することができます。動的索引と静的索引の違いは、動的索引は、索引を含む ObjectGrid インスタンスが初期化されたあとでも作成できる、という点です。動的索引付けは、静的索引付けとは違って非同期プロセスであり、使用される前に作動可能状態になっている必要があります。このメソッドは、動的索引の取得および使用に、静的索引と同じアプローチを使用します。動的索引は、不要になると除去できます。BackingMap インターフェースには、動的索引を作成および除去するためのメソッドがあります。

createDynamicIndex メソッドおよび removeDynamicIndex メソッドの詳細については、[BackingMap API](#) を参照してください。

```
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
import com.ibm.websphere.objectgrid.ObjectGridManager;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.BackingMap;

ObjectGridManager ogManager =
ObjectGridManagerFactory.getObjectGridManager();
BackingMap bm = og.getMap("person");

// create index after ObjectGrid initialization without
DynamicIndexCallback.
bm.createDynamicIndex("CODE", true, "employeeCode", null);

try {
// If not using DynamicIndexCallback, need to wait for the Index to be
ready.
// The waiting time depends on the current size of the map
Thread.sleep(3000);
} catch (Throwable t) {
// ...
}
```

```

// When the index is ready, applications can try to get application index
// interface instance.
// Applications have to find a way to ensure that the index is ready to use,
// if not using DynamicIndexCallback interface.
// The following example demonstrates the way to wait for the index to be
ready
// Consider the size of the map in the total waiting time.

Session session = og.getSession();
ObjectMap m = session.getMap("person");
MapRangeIndex codeIndex = null;

int counter = 0;
int maxCounter = 10;
boolean ready = false;
while (!ready && counter < maxCounter) {
    try {
        counter++;
        codeIndex = (MapRangeIndex) m.getIndex("CODE");
        ready = true;
    } catch (IndexNotReadyException e) {
        // implies index is not ready, ...
        System.out.println("Index is not ready. continue to wait.");
        try {
            Thread.sleep(3000);
        } catch (Throwable tt) {
            // ...
        }
    } catch (Throwable t) {
        // unexpected exception
        t.printStackTrace();
    }
}

if (!ready) {
    System.out.println("Index is not ready. Need to handle this
situation.");
}

// Use the index to perform queries
// Refer to the MapIndex or MapRangeIndex interface for supported
operations.
// The object attribute on which the index is created is the EmployeeCode.
// Assume that the EmployeeCode attribute is Integer type: the
// parameter that is passed into index operations has this data type.

Iterator iter = codeIndex.findLessEqual(new Integer(15));

// remove the dynamic index when no longer needed

bm.removeDynamicIndex("CODE");
// Close the session (optional in Version 7.1.1 and later) for improved
performance
session.close();

```

次のタスク

DynamicIndexCallback インターフェースを使用して、索引付けイベントの発生時に通知を受けることができます。詳しくは、[DynamicIndexCallback インターフェース](#)を参照してください。

[DynamicIndexCallback インターフェース](#)

DynamicIndexCallback インターフェースは、作動可能、エラー、または破棄という索引付けイベントの発生時に、そのことを通知してもらう必要のあるアプリケーションのために設計されています。

DynamicIndexCallback は、BackingMap の createDynamicIndex メソッドのオプション・パラメーターです。アプリケーションは、索引付けイベントの通知を受け取ると、登録済みの DynamicIndexCallback インスタンスを使用して、ビジネス・ロジックを実行することができます。

親トピック: [データの索引付けのためのプラグイン](#)

DynamicIndexCallback インターフェース

DynamicIndexCallback インターフェースは、作動可能、エラー、または破棄という索引付けイベントの発生時に、そのことを通知してもらう必要のあるアプリケーションのために設計されています。DynamicIndexCallback は、BackingMap の createDynamicIndex メソッドのオプション・パラメータです。アプリケーションは、索引付けイベントの通知を受け取ると、登録済みの DynamicIndexCallback インスタンスを使用して、ビジネス・ロジックを実行することができます。

索引付けイベント

例えば、作動可能イベントは、索引を使用する準備が整ったことを意味します。アプリケーションは、このイベントの通知を受け取ると、アプリケーション索引インターフェースのインスタンスの取得および使用を試行することができます。

例: DynamicIndexCallback インターフェースの使用

```
BackingMap personBackingMap = ivObjectGrid.getMap("person");
DynamicIndexCallback callback = new DynamicIndexCallbackImpl();
personBackingMap.createDynamicIndex("CODE", true, "employeeCode", callback);

class DynamicIndexCallbackImpl implements DynamicIndexCallback {
    public DynamicIndexCallbackImpl() {
    }

    public void ready(String indexName) {
        System.out.println("DynamicIndexCallbackImpl.ready() -> indexName = " +
indexName);

        // Simulate what an application would do when notified that the index is
ready.
        // Normally, the application would wait until the ready state is reached and
then proceed
        // with any index usage logic.
        if("CODE".equals(indexName)) {
            ObjectGridManager ogManager =
ObjectGridManagerFactory.getObjectGridManager();
            ObjectGrid og = ogManager.createObjectGrid( "grid" );
            Session session = og.getSession();
            ObjectMap map = session.getMap("person");
            MapIndex codeIndex = (MapIndex) map.getIndex("CODE");
            Iterator iter = codeIndex.findAll(codeValue);
            // Close the session (optional in Version 7.1.1 and later) for improved
performance
            session.close();
        }
    }

    public void error(String indexName, Throwable t) {
        System.out.println("DynamicIndexCallbackImpl.error() -> indexName = " +
indexName);
        t.printStackTrace();
    }

    public void destroy(String indexName) {
        System.out.println("DynamicIndexCallbackImpl.destroy() -> indexName = " +
indexName);
    }
}
```

親トピック: [索引によるデータへのアクセス \(索引 API\)](#)

セッションを使用したグリッド内データへのアクセス

アプリケーションは、Session インターフェースを介してトランザクションを開始および終了できます。Session インターフェースは、アプリケーションを基にした ObjectMap および JavaMap インターフェースへのアクセスも提供します。

各 ObjectMap または JavaMap インスタンスは、特定のセッション・オブジェクトに直接結合しています。eXtreme Scale にアクセスしたい各スレッドは、まず最初に ObjectGrid オブジェクトからセッションを取得しなければなりません。セッション・インスタンスは、スレッド間で同時に共有することはできません。WebSphere® eXtreme Scale は、スレッドのローカル・ストレージをまったく使用しませんが、プラットフォームの制約事項により、あるスレッドから別のスレッドへのセッションの受け渡しの機会が制限されることがあります。

方式

get メソッド

アプリケーションは ObjectGrid.getSession メソッドを使用して、セッション・インスタンス を ObjectGrid オブジェクトから取得します。次の例は、Session インターフェースを取得する方法を示しています。

```
ObjectGrid objectGrid = ...; Session sess = objectGrid.getSession();
```

セッションを取得した後、スレッドはそのセッションへの参照を専用に保持します。getSession メソッドを複数回呼び出すと、その度に新規セッション・オブジェクトが戻されます。

トランザクション・メソッドとセッション・メソッド

セッションは、トランザクションの開始、コミット、またはロールバックに使用できます。ObjectMap と JavaMap を使用した BackingMap に対する操作は、セッション・トランザクション内では非常に効率よく実行されます。トランザクションが開始された後は、そのトランザクションの有効範囲にある 1 つ以上の BackingMap に対するすべての変更は、そのトランザクションがコミットされるまで、特別のトランザクション・キャッシュに保管されます。トランザクションがコミットされると、保留になっている変更内容は BackingMap とローダーに適用され、その ObjectGrid のその他のクライアントから見えるようになります。

WebSphere eXtreme Scale は、トランザクションを自動的にコミットする機能 (自動コミットともいう) もサポートします。すべての ObjectMap オペレーションがアクティブ・トランザクションのコンテキストの外部で実行される場合は、暗黙のトランザクションはそのオペレーションの前に開始され、そのトランザクションはアプリケーションに制御が戻される前に自動的にコミットされます。

```
Session session = objectGrid.getSession();
ObjectMap objectMap = session.getMap("someMap");
session.begin();
objectMap.insert("key1", "value1");
objectMap.insert("key2", "value2");
session.commit();
objectMap.insert("key3", "value3"); // auto-commit
```

Session.flush メソッド

Session.flush メソッドは、ローダーが BackingMap に関連付けられているときにのみ意味があります。flush メソッドは、トランザクション・キャッシュ内の変更内容の現行セットを使用してローダーを呼び出します。ローダーは、変更内容をバックエンドに適用します。これらの変更内容は、flush が呼び出される時はコミットされません。flush 呼び出しの後、セッション・トランザクションがコミットされると、flush 呼び出しの後で発生する更新のみがローダーに適用されます。flush 呼び出しの後、セッション・トランザクションがロールバックされると、フラッシュされた変更内容はトランザクション内のその他すべての保留している変更内容と一緒に廃棄されます。flush メソッドは、ローダーに対するバッチ操作の機会を制限するので、慎重に使用してください。以下は、Session.flush メソッドの使用例です。

```
Session session = objectGrid.getSession();
session.begin();
// make some changes
...
session.flush(); // push these changes to the Loader, but don't commit yet
// make some more changes
...
session.commit();
```

NoWriteThrough メソッド

いくつかのマップはローダーによってバックアップされます。ローダーはマップ内のデータ用に永続ストレージを提供

します。eXtreme Scale マップのみにデータをコミットし、ローダーにデータをプッシュアウトしないことが有益な場合があります。Session インターフェースは、この目的のために beginNoWriteThrough メソッドを提供します。beginNoWriteThrough メソッドは、begin メソッドのようなトランザクションを開始します。beginNoWriteThrough メソッドでは、トランザクションがコミットされると、データはメモリー内のマップにのみコミットされ、ローダーが提供する永続ストレージにはコミットされません。このメソッドが非常に役立つのは、データがマップにプリロードされることです。

分散 ObjectGrid インスタンスを使用する場合、サーバーで遠くのキャッシュは変更せずにニア・キャッシュのみを変更するには、beginNoWriteThrough メソッドが役立ちます。ニア・キャッシュでデータが不整合であると認識されている場合は、beginNoWriteThrough メソッドを使用すると、エントリーをサーバーでは無効にせずに、ニア・キャッシュで無効にすることができます。

Session インターフェースは、現在活動中のトランザクション・タイプを判別する isWriteThroughEnabled メソッドも提供します。

```
Session session = objectGrid.getSession();
session.beginNoWriteThrough();
// make some changes ...
session.commit(); // these changes will not get pushed to the Loader
```

TxID オブジェクト・メソッドの取得

TxID オブジェクトは、内部が見えないオブジェクトで、活動中のトランザクションを識別します。以下の目的には、TxID オブジェクトを使用します。

- ある特定のトランザクションを検索している場合の比較用
- TransactionCallback とローダーのオブジェクト間で共有データを保管するため
- 1 フェーズ・コミット・プロトコルまたは 2 フェーズ・コミット・プロトコルを使用しているセッション・トランザクションからこのトランザクションが開始されたかどうかを判別します。TxID.toString() 出力を調べることによって、このトランザクションが単一区画トランザクションに対するものであったか複数区画トランザクションに対するものであったかを決定することができます。ストリングがキーワード「Local」で始まっている場合、これは単一区画トランザクションを示します。例えば、Local-40000139-72B2-C037-E000-1C271366B073 などです。ストリングがキーワード「WXS」で始まっている場合、これは複数区画トランザクションを示します。例えば WXS-40000139-72B2-BD3A-E000-1C271366B073 などです。

オブジェクト・スロット・フィーチャーについての追加情報は、『TransactionCallback プラグイン』と『ローダー』を参照してください。

パフォーマンス・モニター・メソッド

eXtreme Scale を WebSphere Application Server 内で使用する場合、パフォーマンス・モニタリング用にトランザクション・タイプをリセットすることが必要になることがあります。トランザクション・タイプの設定には、setTransactionType メソッドを使用できます。setTransactionType メソッドについて詳しくは、『WebSphere Application Server Performance Monitoring Infrastructure (PMI) を使用した ObjectGrid パフォーマンスのモニター』を参照してください。

完全な LogSequence メソッドの処理

WebSphere eXtreme Scale は、ある Java™ 仮想マシンから別のマシンへマップを配布する手段として、マップ変更セットを ObjectGrid リスナーに伝搬できます。リスナーが受信済み LogSequences を処理するのを容易にするために、Session インターフェースは processLogSequence メソッドを提供します。このメソッドは LogSequence 内で各 LogElement を検査し、LogSequence MapName によって識別される BackingMap に対して適切なオペレーション（例えば、挿入、更新、無効化など）を実行します。ObjectGrid セッションは、processLogSequence メソッドが呼び出される前に使用可能になっていなければなりません。アプリケーションは、セッションを完了するために適切な commit または rollback 呼び出しを実行する役割があります。自動コミット処理は、このメソッド呼び出しには使用できません。リモート JVM での受信側 ObjectGridEventListener による通常の処理では、この processLogSequence メソッドの呼び出しが続く beginNoWriteThrough メソッド（変更内容のアドレスな伝搬を防止するもの）を使用し、次にトランザクションをコミットまたはロールバックすることで、セッションを開始することになります。

```
// Use the Session object that was passed in during
//ObjectGridEventListener.initialization...
session.beginNoWriteThrough();
// process the received LogSequence
try {
    session.processLogSequence(receivedLogSequence);
} catch (Exception e) {
    session.rollback(); throw e;
}
// commit the changes
session.commit();
```


markRollbackOnly メソッド

このメソッドを使用して、現行トランザクションに「rollback only」とマークを付けます。トランザクションに「rollback only」とマークを付けると、アプリケーションで commit メソッドが呼び出された場合でも、トランザクションはロールバックされます。このメソッドは、通常、トランザクションのコミットが許可されている場合にデータ破壊が発生する可能性があるとして認識されているとき、ObjectGrid 自体またはアプリケーションで使用されます。このメソッドが呼び出されると、このメソッドに渡される Throwable オブジェクトが com.ibm.websphere.objectgrid.TransactionException 例外にチェーニングされます。この例外は、以前に「rollback only」とマーク付けされたセッションで commit メソッドが呼び出された場合の結果です。既に「rollback only」とマーク付けされているトランザクションのこのメソッドに対する以降の呼び出しは、無視されます。つまり、ヌル以外の Throwable 参照を渡す最初の呼び出しのみが使用されます。マークされたトランザクションが完了すると、「rollback only」マークは除去されるため、セッションで開始される次のトランザクションはコミットされます。

isMarkedRollbackOnly メソッド

セッションが現在「rollback only」とマークされている場合に返されます。 markRollbackOnly メソッドが以前このセッションで呼び出されており、セッションで開始されたトランザクションがアクティブな場合、かつこの場合に限り、このメソッドによってブール値 true が返されます。

setTransactionTimeout メソッド

このセッションで開始される次のトランザクションのトランザクション・タイムアウトを特定の秒数に設定します。このメソッドは、このセッションで以前に開始されたトランザクションのトランザクション・タイムアウトには影響を与えません。このメソッドが呼び出された後に開始されたトランザクションにのみ影響を与えます。このメソッドが呼び出されない場合は、com.ibm.websphere.objectgrid.ObjectGrid メソッドの setTxTimeout メソッドに渡されたタイムアウト値が使用されます。

getTransactionTimeout メソッド

このメソッドは、トランザクション・タイムアウト値 (秒単位) を返します。タイムアウト値として setTransactionTimeout メソッドに渡された最後の値は、このメソッドによって返されます。 setTransactionTimeout メソッドが呼び出されない場合は、com.ibm.websphere.objectgrid.ObjectGrid メソッドの setTxTimeout メソッドに渡されたタイムアウト値が使用されます。

transactionTimedOut

このメソッドは、このセッションで開始された現行トランザクションがタイムアウトになると、ブール値 true を返します。

isFlushing メソッド

このメソッドは、呼び出されたセッション・インターフェースの flush メソッドの結果として、すべてのトランザクション変更が Loader プラグインにフラッシュされる場合、かつこの場合に限り、ブール値 true を返します。 Loader プラグインでは、batchUpdate メソッドが呼び出された理由を確認する必要がある場合にこのメソッドが役立ちます。

isCommitting メソッド

このメソッドは、呼び出されたセッション・インターフェースの commit メソッドの結果として、すべてのトランザクション変更がコミットされる場合、かつこの場合に限り、ブール値 true を返します。 Loader プラグインでは、batchUpdate メソッドが呼び出された理由を確認する必要がある場合にこのメソッドが役立ちます。

setRequestRetryTimeout メソッド

このメソッドは、セッションの要求再試行タイムアウト値 (ミリ秒) を設定します。クライアントが要求再試行タイムアウトを設定してある場合、セッション設定値がクライアント値をオーバーライドします。

getRequestRetryTimeout メソッド

このメソッドは、セッションの現行の要求再試行タイムアウト設定を取得します。値 -1 は、タイムアウトが設定されていないことを表します。値 0 は、フェイル・ファースト・モードであることを表します。0 より大きい値は、ミリ秒単位のタイムアウト設定値です。

親トピック: [データの索引付けのためのプラグイン](#)

HashIndex プラグインの構成

組み込み HashIndex である `com.ibm.websphere.objectgrid.plugins.index.HashIndex` クラスは、動的索引を使用してプログラマチックに、構成することができます。

このタスクについて

複合索引の構成は、XML を使用した 通常の索引の構成と同じですが、**attributeName** プロパティ値は例外です。複合索引の場合、**attributeName** プロパティの値は、コンマ区切りの属性のリストです。例えば、値クラス `Address` は、`city`、`state`、および `zipcode` の 3 つの 属性を持つとします。この場合、"`city,state,zipcode`" という **attributeName** プロパティ値を 使用して複合索引を定義し、複合索引に `city`、`state`、および `zipcode` が含まれていることを 示すことができます。

また、複合 HashIndexes は、範囲検索をサポートしないため、`RangeIndex` プロパティを `true` に設定しないよう注意してください。

手順

プログラムで複合索引を構成します。動的索引にのみ適用されます。

次のサンプル・コードも同じ複合索引を作成します。

```
        HashIndex mapIndex = new HashIndex();
        mapIndex.setName("Address.CityStateZip");
        mapIndex.setAttributeName(("city,state,zipcode"));
        mapIndex.setRangeIndex(true);

BackingMap bm = objectGrid.getMap("mymap");
    bm.createDynamicIndex(mapIndex, null);

    try {
        // If not using DynamicIndexCallback, need to wait for the Index to be ready.
        // The waiting time depends on the current size of the map
        Thread.sleep(3000);
    } catch (Throwable t) {
        // ...
    }

    // When the index is ready, applications can try to get application index
    // interface instance.
    // Applications have to find a way to ensure that the index is ready to use,
    // if not using DynamicIndexCallback interface.
    // The following example demonstrates the way to wait for the index to be ready
    // Consider the size of the map in the total waiting time.

    Session session = objectGrid.getSession();
    ObjectMap m = session.getMap("mymap");
    MapRangeIndex codeIndex = null;

    int counter = 0;
    int maxCounter = 10;
    boolean ready = false;
    while (!ready && counter < maxCounter) {
        try {
            counter++;
            codeIndex = (MapRangeIndex) m.getIndex("Address.CityStateZip");
            ready = true;
        } catch (IndexNotReadyException e) {
            // implies index is not ready, ...
            System.out.println("Index is not ready. continue to wait.");
            try {
                Thread.sleep(3000);
            } catch (Throwable tt) {
                // ...
            }
        } catch (Throwable t) {
            // unexpected exception
        }
    }
}
```

```
        t.printStackTrace();
    }
}

if (!ready) {
    System.out.println("Index is not ready. Need to handle this situation.");
}
```

HashIndex プラグイン属性

次の属性を使用して、HashIndex プラグインを構成できます。

複合索引の使用

複合 HashIndex により、照会のパフォーマンスが向上し、高いコストがかかる マップのスキャンを避けることができます。また、この機能は、検索条件に多くの属性が関係する際、キャッシュ・オブジェクトを検索するための便利な方法を HashIndex API に提供します。

2.5+ グローバル索引の使用

グローバル索引を実装することで、大規模な区画化環境 (例えば、100 個の区画を含む環境) におけるデータ検索パフォーマンスを向上させることができます。

親トピック: [データの索引付けのためのプラグイン](#)

HashIndex プラグイン属性

次の属性を使用して、HashIndex プラグインを構成できます。

属性

Name

索引の名前を指定します。名前は各マップで固有でなければなりません。この名前は、バッキング・マップのオブジェクト・マップ・インスタンスから索引オブジェクトを取り出すのに使用されます。

AttributeName

索引に対する属性の名前をコンマで区切ったリストを指定します。フィールド・アクセス索引の場合、属性名はフィールド名と同じです。プロパティ・アクセス索引の場合、属性名は JavaBean 互換のプロパティ名です。属性名が 1 つだけであれば、HashIndex は単一属性索引です。この属性がリレーションシップの場合は、リレーションシップ索引でもあります。複数の属性名が含まれている場合は、HashIndex は複合索引です。

FieldAccessAttribute

非エンティティ・マップに使用されます。true の場合、フィールドを使用してオブジェクトに直接アクセスします。指定されていないか、false の場合、データのアクセスには、属性の getter メソッドが使用されます。

GlobalIndexEnabled

これが true に設定されると、グローバル索引が使用可能になり、アプリケーションは取得した索引オブジェクトを MapGlobalIndex インターフェースにキャストすることができます。

HashIndex の GlobalIndexEnabled プロパティが true に設定されると、HashIndex 構成のほかに MapGlobalIndex インターフェースもサポートするように HashIndex のグローバル索引機能が使用可能になります。大規模な区画化環境でデータを効率的に検索する方法を提供します。

POJOKeYIndex

非エンティティ・マップに使用されます。true の場合、索引はマップのキー部分でオブジェクトをイントロスペクトします。この設定は、キーが複合キーで、値にキーが組み込まれていない場合に役立ちます。指定されていないか、false の場合、索引はマップの値部分でオブジェクトをイントロスペクトします。

RangIndex

true の場合、範囲索引付けが使用可能にされ、アプリケーションは取り出された索引オブジェクトを MapRangIndex インターフェースにキャストできます。RangeIndex プロパティが false と構成されている場合は、アプリケーションは取り出された索引オブジェクトを MapIndex インターフェースにしかキャストできません。

単一属性 HashIndex と複合 HashIndex

HashIndex の AttributeName プロパティに複数の属性名が含まれている場合、HashIndex は複合索引です。そうではなく、含まれている属性名が 1 つのみの場合は、単一属性索引です。例えば、AttributeName プロパティ値が city, state, zipcode であるような、複合 HashIndex が考えられます。この例では、3 つの属性がコンマで区切られています。もし AttributeName プロパティ値が単に zipcode であれば、属性は 1 つだけなので、単一属性 HashIndex であるということになります。

複合 HashIndex は、検索条件に多くの属性が関係するような場合に、キャッシュ・オブジェクトを検索する効果的な方法を提供します。ただし、範囲索引はサポートしないため、RangIndex プロパティは false に設定されている必要があります。

詳しくは、[複合索引の使用](#)を参照してください。

リレーションシップ HashIndex

単一属性 HashIndex の索引属性が、単一値または複数値のリレーションシップの場合、その HashIndex はリレーションシップ HashIndex です。リレーションシップ HashIndex の場合、HashIndex の RangIndex プロパティは「false」に設定されている必要があります。

キー HashIndex

非エンティティ・マップの場合、HashIndex の POJOKeYIndex プロパティが true に設定されていると、HashIndex はキー HashIndex であり、エントリーのキー部分が索引付けに使用されます。HashIndex の AttributeName プロパティが指定されていないと、キー全体に索引が付けられます。指定されていると、キー HashIndex は単一属性 HashIndex にしかありません。

範囲 HashIndex

HashIndex の RangIndex プロパティが true に設定されている場合、その HashIndex は範囲索引であり、MapRangIndex インターフェースをサポートできます。MapRangIndex は、範囲関数 greater than や less than、あるいは両方を使用するデータ検出をサポートしますが、MapIndex は equals 関数のみをサポートします。単

一属性索引の場合、**RangeIndex** プロパティを true に設定できるのは、索引付けられる属性のタイプが Comparable の場合に限りです。単一属性索引が照会によって使用される場合、RangeIndex プロパティは true に設定されていなければならない、索引付けられる属性のタイプは Comparable でなければなりません。リレーションシップ HashIndex および複合 HashIndex の場合、RangeIndex プロパティは false に設定されていなければならない。

RangeIndex プロパティの値が true なので、前記の例は範囲 HashIndex です。

以下の表に、範囲索引の使用についての要約を示します。

表 1. 範囲索引のサポート。HashIndex のタイプが範囲索引をサポートするかどうかを記述します。

HashIndex タイプ	範囲索引のサポート
単一属性 HashIndex: 索引付けられるキーまたは属性のタイプは Comparable である	はい
単一属性 HashIndex: 索引付けられるキーまたは属性のタイプは Comparable でない	いいえ
複合 HashIndex	いいえ
リレーションシップ HashIndex	いいえ

親トピック: [HashIndex プラグインの構成](#)

複合索引の使用

複合 HashIndex により、照会のパフォーマンスが向上し、高いコストがかかる マップのスキャンを避けることができます。また、この機能は、検索条件に多くの属性が関係する際、キャッシュ・オブジェクトを検索するための便利な方法を HashIndex API に提供します。

パフォーマンスの改善

複合 HashIndex を使用すると、一致検索条件に入れた複数の属性によって、キャッシュ・オブジェクトを高速かつ簡単に見つけることができます。複合索引は、完全属性一致検索をサポートしますが、範囲検索はサポートしません。

注: 複合索引は ObjectGrid 照会言語での BETWEEN 演算子の使用をサポートしません。BETWEEN は範囲サポートを必要とすることがあるためです。より大 (>)、より小 (<) 条件も、範囲索引を必要とするため機能しません。

複合索引の構成

複合索引は動的索引としてプログラマチックに構成することができます。

プログラマチック構成

次の例は複合索引を作成するものです。

```
HashIndex mapIndex = new HashIndex();
mapIndex.setName("Address.CityStateZip");
mapIndex.setAttributeName(("city,state,zipcode"));
mapIndex.setRangeIndex(false);

BackingMap bm = objectGrid.getMap("mymap");
bm.createDynamicIndex(mapIndex, null);
```

複合索引の構成は、XML を使用した 通常の索引の構成と同じですが、attributeName プロパティ値は例外なので注意してください。複合索引の場合、attributeName の値は、コンマ区切りの属性のリストです。例えば、値クラス Address は、city、state、および zipcode の 3 つの属性を持つとします。この場合、"city,state,zipcode" という attributeName プロパティ値を使用して複合索引を定義し、複合索引に city、state、および zipcode が含まれていることを示すことができます。

複合 HashIndexes は、範囲検索をサポートしないため、true に設定された RangeIndex プロパティを持つことができません。

複合索引の検索の実行

複合索引が構成されたら、アプリケーションは、MapIndex インターフェースの findAll(Object) メソッドを使用して、検索を実行できます。

2.5+ 制約事項: MapIndex.EMPTY_VALUE は複合グローバル索引に対してはサポートされません。

```
Session sess = objectgrid.getSession();
ObjectMap map = sess.getMap("MAP_NAME");
MapIndex codeIndex = (MapIndex) map.getIndex("INDEX_NAME");
Object[] compositeValue = new Object[]{ MapIndex.EMPTY_VALUE,
    "MN", "55901"};
Iterator iter = mapIndex.findAll(compositeValue);
// Close the session (optional in Version 7.1.1 and later) for improved performance
sess.close();
```

MapIndex.EMPTY_VALUE は compositeValue[0] に割り当てられ、評価から city 属性が除外されることを示します。結果には、state 属性が"MN" に等しく、zipcode 属性が"55901" に等しいオブジェクトのみが含まれます。

マイグレーションおよびインターオペラビリティ

複合索引の使用に関する唯一の制約は、異種のコンテナがある分散環境では、アプリケーションが複合索引を構成できないことです。古いコンテナ・サーバーと新しいコンテナ・サーバーを混用することはできません。というのは、古いほうのコンテナ・サーバーは複合索引構成を認識しないからです。複合索引は、既存の通常の属性索引とよく似ていますが、複合索引では、複数の属性にまたがる索引付けが許可される点が異なります。通常の属性索引のみを使用する場合、コンテナ混在環境はそのまま存続できます。

親トピック: [HashIndex プラグインの構成](#)

グローバル索引の使用

2.5+ グローバル索引を実装することで、大規模な区画化環境 (例えば、100 個の区画を含む環境) におけるデータ検索パフォーマンスを向上させることができます。

このフィーチャーはまた、索引付き属性の所在を検索する手段を提供し、索引付き属性に関連するエージェント操作や照会操作を向上させることができます。グローバル索引の機能については、MapGlobalIndex API 資料を参照してください。

パフォーマンスの改善

大規模な区画化環境では、キャッシュ・オブジェクトがすべての区画に分散されます。索引、照会、またはエージェントを使用したデータの検索で完全な結果を得るためには、すべてのサーバーに対して検索を実行する必要があります。この種の検索は、各区画をロードして検索するために必要なリモート・コールが発生するので低速になります。その上、検索条件を満たすデータがすべての区画にあるとは限りません。グローバル索引は、一致するデータを実際を含む区画に対してのみ検索を実行するため、検索パフォーマンスを向上させます。グローバル索引フィーチャーは、索引付き属性の所在を追跡したり、すべての区画から属性に合った適用区画を決定したりすることができます。通常、適用区画は全区画のサブセットです。したがって、適用区画での索引、照会、およびエージェントの実行は、グローバル索引によって相殺されるときでさえ、すべての区画でこれらの項目を実行するときよりもはるかに速くなります。

データの検索

アプリケーションはキーを使用してデータを検索することができます。アプリケーションは索引を使用したデータを検索することもできますが、これは、当該データが 1 つ以上の属性を持ち、それらの属性に対して索引が定義されていることが前提となります。従来から、アプリケーションは、クライアント索引プロキシを使用してすべての区画からエントリー・キーを取得することもできれば、エージェントを使用してすべての区画で索引検索を実行し、キャッシュ・キー、値、またはその両方を返すこともできます。グローバル索引フィーチャーを使用すれば、アプリケーションは、MapGlobalIndex API を通じて、操作を適用区画のみで実行する効率的な方法によりエントリー・キー、値、またはその両方を検索することができます。

エージェント操作

例えば、エージェント操作が索引付き属性に関連している場合は、索引付き属性を使用してエントリーを無効化することにより、アプリケーションはグローバル索引を使用してまず属性で適用区画を検索することができます。その後、アプリケーションはエージェントをこれらの適用区画に送ることができます。MapGlobalIndex.findPartitions() メソッドを使用して、属性によって適用区画を検索することができます。

グローバル索引の使用可能化

グローバル索引は HashIndex プラグインの拡張機能であり、既存の任意の HashIndex 構成で使用可能にすることができます。

グローバル索引検索の実行

グローバル索引機能は MapGlobalIndex API で定義されます。HashIndex プラグインでグローバル索引が使用可能になれば、アプリケーションは、取得した索引プロキシを MapGlobalIndex タイプにキャストし、グローバル索引を開始することができます。

```
// in client ObjectGrid process
MapGlobalIndex mapGlobalIndexCODE = (MapGlobalIndex)m.getIndex("CODE", false);
Object[] attributes = new Object[] {new Integer(1)};
Collection partitions = mapGlobalIndexCODE.findPartitions(attributes);
Set keys = mapGlobalIndexDependency.findKeys(attributes);
Set values = mapGlobalIndexDependency.findValues(attributes);
Map entries = mapGlobalIndexDependency.findEntries(attributes);
```

マイグレーションおよびインターオペラビリティ

グローバル索引を使用する際の唯一の制約は、異種コンテナを含む分散環境ではアプリケーションがグローバル索引を構成できないということです。古いコンテナ・サーバーと新しいコンテナ・サーバーは混在できません。これは、古いコンテナ・サーバーはグローバル索引構成および複合グローバル索引構成を認識しないからです。

グローバル索引または複合グローバル索引を使用するためには、まず、アプリケーションで使用しているコンテナ・サーバーとコンテナ・クライアントをすべて停止する必要があります。その後、HashIndex 構成でグローバル索引を使用可能にし、コンテナ・サーバーとコンテナ・クライアントを再始動します。

親トピック: [HashIndex プラグインの構成](#)

クライアントへの連続照会を使用したマップ更新の通知

2.5+ データ・グリッドでオブジェクトまたはエントリーが挿入または更新されたときに、クライアント Java™ 仮想マシン (JVM) で通知を受けることができます。

始める前に

連続照会を使用する場合は、コンテナ・サーバーとクライアント間の通信に使用されるトランスポート・メカニズムである IBM® eXtremeIO を使用可能にする必要があります。eXtremeIO の使用可能化について詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

このタスクについて

データ・グリッドと対話するクライアント・アプリケーションを開発する場合は、フィルター条件に一致するエントリーが挿入、更新、または削除されたときに自動的にリアルタイムの結果を取得する照会が必要になることがあります。例えば、頻繁な更新が必要な株価アプリケーションを開発するとします。この更新は、株式市場で行われた変更を反映します。そのため、正確でタイムリーな結果を提供できるように、アプリケーションが変更について即時に通知を受けるのが重要になります。連続照会は低メモリー・フットプリントであり、データ・グリッド内で変更が行われたときに能動的にクライアントに通知することができます。

以下の手順を使用して、連続照会を使用するようにクライアント・アプリケーションをプログラミングします。

制約事項: 値属性パス null を指定する照会は、値オブジェクトがストリングや整数などのプリミティブ Java 型でなければサポートされません。null が指定されると、値オブジェクト全体を照会するために照会フィルターが使用されます。

手順

1. クライアント・アプリケーションで連続照会マネージャーを呼び出します。例えば、以下のコード行を挿入します。

```
ContinuousQueryManager cqMan = ContinuousQueryManagerFactory.getManager(og);
```

2. フィルターまたはフィルター・チェーンを定義します。独自のフィルターを実装するか、提供されている基本フィルター AND、OR、LT、GT、EQ などを使用できます。インスタンス化されたフィルターまたはフィルター・チェーンには、固有 ID が付与されます。サポートされるすべてのフィルターについては、[Java API 資料へのアクセス](#)を参照して、連続照会 API を確認してください。

以下のコード例で、等価 (EQ) 基本フィルターを使用する一つの方法を示します。データ・グリッドに、フィールド firstName がある Customer オブジェクトが含まれているものと仮定します。フィルターは、firstName が Larry に等しいと、true を返します。

```
EQFilter<String, String> equalsFilter = new EQFilter<String, String>("firstName", "Larry");
```

3. 前のステップで作成したフィルターを使用して照会を定義します。例えば、以下のようになります。

```
ContinuousQueryTopic<String, Customer> topic =  
    cqMan.<String, Customer> defineContinuousQuery("myMapName", equalsFilter, true,  
    true, true);
```

4. オプション: 連続照会キャッシュを取得して、連続照会のクライアント・サイドの結果にアクセスします。照会が鍵のみの照会として定義されている場合は、照会を満たす鍵のみが、連続照会キャッシュに入ります。例:

```
ContinuousQueryCache cache = topic.getCache();
```

5. オプション: さらに、ContinuousQueryTopic インスタンスを使用して ContinuousQueryListener インターフェースを実装するクラスを登録して、連続照会の結果が変更されたときに通知を受け取ることができます。addListener メソッドを呼び出してリスナーを登録します。例えば、以下のようになります。

```
ContinuousQueryListener<String, Customer> listener = new MyCQListener<String,  
Customer>();  
topic.addListener(listener);
```

次のタスク

連続照会 API について詳しくは、[API 資料: パッケージ com.ibm.websphere.objectgrid.continuousquery](#) を参照してください。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発

Representational State Transfer (REST) ゲートウェイを使用して、集合がホスティングする単純データ・グリッドにアクセスできます。この REST ゲートウェイは、非 Java 環境からデータ・グリッドにアクセスする必要があるときに便利です。

始める前に

- REST ゲートウェイは、WebSphere® DataPower® XC10 アプライアンス、バージョン 8.6 以降で使用できます。
- アプライアンス上に単純データ・グリッドを作成する必要があります。単純データ・グリッドの作成について詳しくは、[単純データ・グリッドの作成](#)を参照してください。

このタスクについて

REST ゲートウェイを使用して、DataPower XI50 アプライアンスや .NET アプリケーションなどの非 Java 環境から単純データ・グリッドのデータにアクセスします。REST ゲートウェイを使用すると、Java ベースの ObjectMap API が使用する IBM® オブジェクト・リクエスト・ブローカー (ORB) をホスティングできない Java™ 仮想マシンからマップ・データにアクセスすることもできます。

トランザクション

WebSphere DataPower XC10 アプライアンスに対する各 REST 操作により、データ・グリッドへの独立したトランザクションが開始して終わります。複数の操作をチェーニングして、単一トランザクションにまとめることはできません。

ロード・バランシング

REST ゲートウェイを使用するときは、クライアントの責任において、WebSphere DataPower XC10 アプライアンス集合への要求のロード・バランスを取る必要があります。クライアント・プログラムでは、外部ロード・バランサーを使用するか、使用する HTTP クライアントにロジックを追加できます。

セキュリティ

REST ゲートウェイを介した通信では、データ・グリッドでセキュリティを有効にしていなくても、常にセキュアな構成が実現します。データ・グリッドへのアクセスに使用するユーザー・グループを構成し、このグループにデータ・グリッドへの all 許可限を割り当ててください。

WebSphere eXtreme Scale REST データ・サービスへの関係

REST ゲートウェイは、Microsoft ADO.NET データ・サービス・インターフェースを実装する WebSphere eXtreme Scale REST データ・サービスとは別個のエンティティです。

2.5+ グリッド別名

複数のデータ・グリッドに同時にデータを取り込む必要がある場合、REST ゲートウェイを使用して、グリッド別名を作成し、管理することができます。グリッド別名を使用すると、グリッドを切り替え、それらのグリッドに同時にデータを取り込むことができます。例えば、グリッド A に対してデータ・グリッド操作を実行したいが、もう 1 つのグリッドであるグリッド B に切り替えてデータを取り込むための間接層も必要であるという場合があります。この場合、グリッド A を指すグリッド C という別名を作成することができます。この別名を使用して、グリッド A に対してデータ・グリッド操作を実行し、同時にグリッド B にデータを取り込むことができます。グリッド A に対して (グリッド C という別名を使用して) データ・グリッド操作を実行すると同時に、別名をグリッド A からグリッド B を指すように切り替えることができます。グリッド別名は、作成、照会、あるいは REST リソース `resources/gridalias` から削除することができます。REST ゲートウェイを使用したグリッド別名の管理について詳しくは、[REST ゲートウェイ: REST 操作](#)を参照してください。

[REST ゲートウェイ: URI フォーマット](#)

特定のフォーマットで URI を指定すると、単純データ・グリッドにアクセスし、操作を実行できます。

[REST ゲートウェイ: データ・フォーマット](#)

REST ゲートウェイは、HTTP 要求内の Content-Type ヘッダーを使用して、データ・グリッドに保管するデータのデータ・フォーマットを判定します。

[REST ゲートウェイ: REST 操作](#)

HTTP POST、GET、および DELETE 操作を使用して、データ・グリッドのデータの挿入、更新、取得、および削除を行います。REST ゲートウェイは、ご使用のデータ・グリッドを指すグリッド別名を管理するための HTTP 要求もサポートします。グリッド別名は、複数のデータ・グリッドに一度にデータを取り込んだり、データ・グリッドを切り替えたりする必要があるときに便利です。グリッド別名は作成、照会、および削除することができます。REST リソース `/resource/gridalias` を使用します。

[REST ゲートウェイの例: データ・グリッドのマップ項目の挿入と取得](#)

HTTP メソッドの POST および GET を使用して、データ・グリッドのマップ項目を挿入および取得できます。

REST ゲートウェイの例: Java クライアントから REST マップへのデータの挿入およびデータへのアクセス (ObjectMap API を使用)

REST ゲートウェイを使用してデータがマップに挿入される場合、提供されたコンテンツ・タイプおよび要求本体をラップするためにタイプ `com.ibm.websphere.xsa.RestValue` のラッパー・クラスが使用されます。同じ `RestValue` クラスを使用して、Java クライアントからデータを挿入したり、マップからデータを取得したりすることができます。これには ObjectMap API を使用します。

REST ゲートウェイの例: データ・グリッドのマップ項目のクリア

REST ゲートウェイの HTTP DELETE メソッドを使用して、データ・グリッド内の マップをクリアできます。

REST ゲートウェイの例: 動的マップの作成

単純データ・グリッドを作成すると、同じ名前のデフォルト・マップがデフォルトで作成されます。アプリケーションの必要に応じて、マップ・テンプレートを使用して追加のマップを作成することもできます。

REST ゲートウェイの例: 存続時間 (TTL) の有効期限

TTL 値は、最終更新時間 (*.LUT) マップと最終アクセス時間 (*.LAT) マップの 両方に設定できます。デフォルトの TTL は、いずれのマップ・タイプも 1 時間です。

REST ゲートウェイ: セキュリティー構成

REST ゲートウェイを介してデータ・グリッドにアクセスするには、データ・グリッドのセキュリティーが使用可能に設定されているかどうかに関わらず、ユーザーは WebSphere DataPower XC10 アプライアンス に認証される必要があります。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを 常に提供しなければなりません。REST ゲートウェイを介してデータ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。

REST ゲートウェイ: HTTP セッションと Cookie

Set-Cookie: ヘッダーを指定し、HTTP セッションと Cookie を REST ゲートウェイで使用します。

親トピック: [単純データ・グリッドにアクセスするアプリケーションの開発](#)

関連タスク:

[単純データ・グリッドの作成](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

[動的マップの構成オプション](#)

REST ゲートウェイ: URI フォーマット

特定のフォーマットで URI を指定すると、単純データ・グリッドにアクセスし、操作を実行できます。

URI フォーマット

WebSphere® DataPower® XC10 アプライアンス上の単純データ・グリッドにアクセスするための REST URI フォーマットは次のとおりです。

```
/resources/datacaches/[grid_name]/[map_name]/[key]
```

デフォルトのコンテキスト・ルートは resources です。

MyDataGrid という名前の単純データ・グリッドを、ホスト名 myxc10.ibm.com のアプライアンス上に作成した場合、キー名 my.data.item にアクセスするための URL は、結果として次のようになります。

```
http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyMap/my.data.item
```

前の例では、MyDataGrid グリッド内の MyMap デフォルト・マップが使用されます。このデフォルト・マップに 存続時間 (TTL) の期限はありません。データ・グリッド内に配置された項目は、明示的に削除されるまでデータ・グリッド内にとどまります。TTL の期限を構成するには、[REST ゲートウェイの例: 存続時間 \(TTL\) の有効期限](#)を参照してください。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイ: データ・フォーマット

REST ゲートウェイは、HTTP 要求内の Content-Type ヘッダーを使用して、データ・グリッドに 保管するデータのデータ・フォーマットを判定します。

データ・フォーマット

REST ゲートウェイは、HTTP 要求内の Content-Type ヘッダーを使用して、データ・グリッド内に 保管するデータのデータ・フォーマットを判定します。タイプが application/xml のコンテンツを挿入した場合に、アプリケーションが同じキャッシュ・キーに対して GET 操作を 実行すると、応答の本体と Content-type は同じフォーマット・タイプになります。この例の場合、応答の本体は application/xml フォーマットになります。複数のコンテンツ・タイプのデータを同一データ・グリッドに保管できます。以下に、有効なコンテンツ・タイプの例をいくつか示します。

表 1. HTTP 要求内の content-type ヘッダーのコンテンツ・タイプ

コンテンツ・タイプ	用途
application/xml	XML
application/json	JavaScript データ
application/octet-stream	シリアライズド・オブジェクト、汎用データ

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイ: REST 操作

HTTP POST、GET、および DELETE 操作を使用して、データ・グリッドのデータの挿入、更新、取得、および削除を行います。REST ゲートウェイは、ご使用のデータ・グリッドを指すグリッド別名を管理するための HTTP 要求もサポートします。グリッド別名は、複数のデータ・グリッドに一度にデータを取り込んだり、データ・グリッドを切り替えたりする必要があるときに便利です。グリッド別名は作成、照会、および削除することができ、REST リソース `/resource/gridalias` を使用します。

データ・グリッドにデータを取り込むための REST 操作

表 1. 操作、それに相当する HTTP メソッド、および応答コードの定義

操作	HTTP メソッド	応答コード
挿入または更新	POST	<ul style="list-style-type: none">200 CREATED: データは正常にデータ・グリッドに挿入されたか、更新されました。400 BAD REQUEST: データの挿入操作または更新操作は正常に完了しませんでした。
取得	GET	<ul style="list-style-type: none">200 OK: 直前の挿入操作または更新操作の応答の本体と content-type が取得されました。404 NOT FOUND: 指定されたキーがデータ・グリッド内に存在しません。400 BAD REQUEST: アプライアンスは要求を処理できませんでした。
削除	DELETE	<ul style="list-style-type: none">200 NO CONTENT: 項目がデータ・グリッドから削除されました。400 BAD REQUEST: アプライアンスは要求を処理できませんでした。

2.5+

データ・グリッドの別名を管理するための REST 操作

表 2. 操作、それに相当する HTTP メソッド、および応答コードの定義

操作	HTTP メソッド	応答コード
別名の追加または更新	POST <code>/resources/gridalias/ <aliasname>?src= <sourceGridName></code>	<ul style="list-style-type: none">204 SUCCESS: グリッド別名が正常に作成されました。401 SOURCE GRID DOES NOT EXIST: 別名が指すデータ・グリッドが存在しないため、別名を作成できませんでした。409 DATA GRID EXISTS: 同じ名前を持つデータ・グリッドが既に存在しているため、別名を作成できませんでした。
指定した別名を持つ現行データ・グリッドを取得	GET <code>/resources/gridalias/ <aliasname></code>	<ul style="list-style-type: none">200 OK
すべての別名のリストを取得	GET <code>/resources/gridalias/</code>	<ul style="list-style-type: none">200 OK
別名の削除	DELETE <code>/resources/gridalias/ <aliasname></code>	<ul style="list-style-type: none">204 SUCCESS: 別名が削除されます。404 BAD REQUEST: 別名が存在していません。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: データ・グリッドのマップ項目の挿入と取得

HTTP メソッドの POST および GET を使用して、データ・グリッドのマップ項目を挿入および取得できます。

例: 挿入操作

定義済み URI とデータ・フォーマットを使用して、データ・グリッドに情報を挿入できます。次の例では、キー「bob」を MyGrid グリッドと MyGrid マップに挿入します。

```
POST /resources/datacaches/MyGrid/MyGrid/bob
Content-type: application/xml
<mydata>this is some data</mydata>
```

例: 取得操作

前の例で挿入したキーを取得するには、次の URI を使用できます。

```
GET /resources/datacaches/MyGrid/MyGrid/bob
```

GET 操作は、個々のキーに対して実行しなければなりません。すべてのマップ項目を取得することはできません。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: Java クライアントから REST マップへのデータの挿入およびデータへのアクセス (ObjectMap API を使用)

REST ゲートウェイを使用してデータがマップに挿入される場合、提供されたコンテンツ・タイプおよび要求本体をラップするためにタイプ `com.ibm.websphere.xsa.RestValue` のラッパー・クラスが使用されます。同じ `RestValue` クラスを使用して、Java クライアントからデータを挿入したり、マップからデータを取得したりすることができます。これには `ObjectMap` API を使用します。

REST マップにアクセスするための Java クライアント・コード

```
RestValue rv = new RestValue();
rv.setContentType("application/xml");
String myXml("<customer>brian</customer>");
rv.setValue(myXml.getBytes("UTF8"));
ogSession.begin();
ObjectMap map = ogSession.getMap("myMap.LUT");
map.insert("brian", rv);
ogSession.commit();
```

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: データ・グリッドのマップ項目のクリア

REST ゲートウェイの HTTP DELETE メソッドを使用して、データ・グリッド内の マップをクリアできます。

個々の項目のクリア

個々の項目を削除するには、DELETE メソッドとオブジェクトのキー名を使用します。

```
DELETE http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyDataGrid/my.data.item
```

データ・グリッド上のマップ全体のクリア

データ・グリッド内のマップ全体をクリアするには、URI のキー部分を省略して HTTP DELETE メソッドを使用します。例えば、MyDataGrid データ・グリッド上の MyDataMap.LUT マップをクリアするには、次の操作を使用します。

```
DELETE http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyDataMap.LUT
```

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: 動的マップの作成

単純データ・グリッドを作成すると、同じ名前のデフォルト・マップがデフォルトで作成されます。アプリケーションの必要に応じて、マップ・テンプレートを使用して追加のマップを作成することもできます。

動的マップの作成

あるマップに対して最初の操作を実行したとき、そのマップがマップ・テンプレートと一致しても、まだ作成されていないマップであると、新規動的マップが作成されます。例えば、*.LUT テンプレートを使用する新規動的マップを MyMap.LUT という名前で作成するには、GET、DELETE、または POST 操作に次の URI を使用します。

```
http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyMap.LUT/a.key
```

動的マップを命名する方法については、[動的マップの構成オプション](#) を参照してください。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: 存続時間 (TTL) の有効期限

TTL 値は、最終更新時間 (*.LUT) マップと最終アクセス時間 (*.LAT) マップの両方に設定できます。デフォルトの TTL は、いずれのマップ・タイプも 1 時間です。

例

最終更新時間 (*.LUT) マップと最終アクセス時間 (*.LAT) マップのいずれに TTL 値を設定する場合も、値を秒数で指定した TTL 要求パラメーターを提供します。例えば、a.key キーに対して TTL 値 600 秒を設定するには、HTTP POST メソッドを使用して値をデータ・グリッドに挿入または更新するときに ttl 要求パラメーターを指定します。

```
http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyMap.LUT/a.key?ttl=600
```

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイ: セキュリティー構成

REST ゲートウェイを介してデータ・グリッドにアクセスするには、データ・グリッドのセキュリティーが使用可能に設定されているかどうかに関わらず、ユーザーは WebSphere® DataPower® XC10 アプライアンス に認証される必要があります。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを常に提供しなければなりません。REST ゲートウェイを介してデータ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。

認証と許可

REST ゲートウェイを介してデータ・グリッド・マップにアクセスするには、ユーザーまたはユーザー・グループが認証される必要があります。また URI 内で指定したデータ・グリッドにアクセスする権限が必要です。データ・グリッドでセキュリティーを構成していない場合でも、REST ゲートウェイを介した通信に使用するユーザー・グループを構成し、このグループにデータ・グリッドへの all アクセス権限を割り当てる必要があります。データ・グリッドへのアクセス権限の構成について詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを提供しなければなりません。

```
Authorization: Basic <base64 encoded string of "userid:password">
```

Basic 許可ヘッダーの形式について詳しくは、[Wikipedia: Basic access authentication](#) を参照してください。

セキュア・データ・グリッド

セキュア・データ・グリッドの構成で REST ゲートウェイを使用できます。セキュア・データ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。ユーザーが認証される必要があります。また URI 内で指定したデータ・グリッドにアクセスする権限が必要です。

表 1. セキュア・データ・グリッド

許可	取得	ポスト	削除
READ	X		
WRITE	X		
CREATE	X	X	
ALL	X	X	X

トランスポート・セキュリティー

REST ゲートウェイを使用するクライアントは、トランスポート・セキュリティーが必要な場合、HTTPS プロトコルを使用できます。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

REST ゲートウェイ: HTTP セッションと Cookie

Set-Cookie: ヘッダーを指定し、HTTP セッションと Cookie を REST ゲートウェイで使⽤します。

REST ゲートウェイ・コードは、現在セッション中でないクライアントから要求を受け取ると、HTTP セッションを作成します。不必要なセッションの作成を避け、最高のパフォーマンスを得るため、REST クライアントは、REST ゲートウェイから Set-Cookie: ヘッダーを使用して返される Cookie を保存し、後続の要求ではそれらと同じ Cookie を、Cookie: ヘッダーを使用して REST ゲートウェイに提供しなければなりません。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

.NET API を使用したデータ・グリッド・アプリケーションの開発

2.5+ Java™ アプリケーションと同じデータ・グリッドを使用する Microsoft .NET アプリケーションを開発できます。

.NET [.NET 開発環境の設定](#)

Microsoft Visual Studio で WebSphere® eXtreme Scale クライアント for .NET を使用するには、開発環境をインストールし、WebSphere eXtreme Scale クライアント for .NET アセンブリーを使用するようにプロジェクトを構成する必要があります。

.NET [.NET API を使用した動的マップの作成](#)

データ・グリッドがインスタンス化されたら、.NET API を使用して動的マップを作成できます。マップ・テンプレートの事前定義セットに基づいて、動的にマップをインスタンス化することができます。

.NET [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

ClassAlias および FieldAlias 注釈を使用して、Java と .NET クラス間でのデータ・グリッド・データの共有を使用可能にします。

.NET [PartitionKey 注釈を使用したキーから区画へのマップ](#)

PartitionKey 別名を使用して、データが保存される区画を判別するためにハッシュ・コード計算を実行するフィールドまたは属性を識別します。PartitionKey 注釈は、キー属性でのみ有効です。

.NET [.NET アプリケーションにおけるトランザクションのためのプログラミング](#)

トランザクションを必要とする .NET アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

.NET [WebSphere eXtreme Scale Client for .NET 用のデータ・グリッド・セキュリティーの構成](#)

Secure Sockets Layer (SSL) を介して通信し、ユーザー/パスワード認証ロジックを使用するように .NET および Java を構成できます。

.NET [WebSphere eXtreme Scale Client for .NET 用の TLS の構成](#)

WebSphere eXtreme Scale Client for .NET 用に Transport Layer Security (TLS) を構成することができます。

.NET [WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

WebSphere eXtreme Scale Client for .NET からサーバー・サイドに資格情報を送信するには、ICredentialGenerator および ICredential インターフェースを実装する必要があります。これらのインターフェースでは、データ・グリッドに渡されてサーバー・サイドで解釈される資格情報オブジェクトが生成されます。サーバー・サイドで、対応するプラグインが資格情報オブジェクトを解釈します。

.NET [WebSphere eXtreme Scale Client for .NET のためのカスタム資格情報のプログラミング](#)

マップのユーザー資格情報を指定することができます。マップのユーザー資格情報があれば、2 人のユーザーが Web アプリケーションを介して同じデータ・グリッドと対話するようにすることができます。

親トピック: [単純データ・グリッドにアクセスするアプリケーションの開発](#)

.NET 開発環境の設定

Microsoft Visual Studio で WebSphere® eXtreme Scale クライアント for .NET を使用するには、開発環境をインストールし、WebSphere eXtreme Scale クライアント for .NET アセンブリーを使用するようにプロジェクトを構成する必要があります。

始める前に

- サポートされる Microsoft Visual Studio リリースのリストについては、[Microsoft .NET に関する考慮事項](#)を参照してください。
- WebSphere eXtreme Scale クライアント for .NET をインストールします。インストール・ウィザードで、「カスタム」パスを選択し、開発環境を選択します。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストール](#)を参照してください。

手順

1. Microsoft Visual Studio 環境で、プロジェクトを開きます。
2. WebSphere eXtreme Scale クライアント for .NET アセンブリーへの参照を追加します。アセンブリーは、[net_client_home](#)¥bin ディレクトリーにあります。IBM.WebSphere.Caching.dll ファイルを選択します。
3. WebSphere eXtreme Scale クライアント for .NET API を使用するために、以下の行をアプリケーションに追加します。

```
using IBM.WebSphere.Caching;  
using IBM.WebSphere.Caching.Map;
```

タスクの結果

アセンブリーを開発環境に統合すると、WebSphere eXtreme Scale クライアント for .NET API 用に IntelliSense が使用可能になります。

次のタスク

クライアント・アプリケーションで WebSphere eXtreme Scale クライアント for .NET API を使用します。API 資料へのアクセスについて詳しくは、[WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)を参照してください。

[WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)

WebSphere eXtreme Scale クライアント for .NET API 資料へのアクセスは、.chm ファイル内でも、インフォメーション・センターの API 資料を表示しても行うことができます。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

WebSphere eXtreme Scale Client for .NET API 資料へのアクセス

WebSphere® eXtreme Scale クライアント for .NET API 資料へのアクセスは、.chm ファイル内でも、インフォメーション・センターの API 資料を表示しても行うことができます。

手順

次のいずれかのオプションを使用して WebSphere eXtreme Scale クライアント for .NET API 資料を開きます。

- 製品と一緒にインストールされる .NET Client API 資料を使用します。 .NET クライアント API 資料をローカルで開くには、 [net_client_home¥doc¥IBM.WebSphere.Caching.chm](#) ファイルを開きます。
- インフォメーション・センターの API 資料を閲覧します。 詳しくは、 [Client for .NET API 資料](#) を参照してください。

親トピック: [.NET .NET 開発環境の設定](#)

.NET API を使用した動的マップの作成

2.5+ データ・グリッドがインスタンス化されたら、.NET API を使用して動的マップを作成できます。マップ・テンプレート
の事前定義セットに基づいて、動的にマップをインスタンス化することができます。

始める前に

動的マップで、どの構成オプションを使用する必要があるかを決定します。詳しくは、[動的マップの構成オプション](#)を参照してください。

手順

GetGridMapPessimisticTx メソッドを呼び出します。

```
IGridManager gm = GridManagerFactory.GetGridManager( );
ICatalogDomainInfo cdi =
    gm.CatalogDomainManager.CreateCatalogDomainInfo( catalogServerHostsList );
IClientConnectionContext ccc = gm.Connect( cdi, "SimpleClient.properties" );
grid = gm.GetGrid( ccc, "Grid" );
IGridMapPessimisticTx<Object, Object> map =
    grid.GetGridMapPessimisticTx<Object, Object>( "SessionState.LAT.P" );
```

SessionState.LAT.P マップは、最終アクセス時刻除去、ペシミスティック・ロック、および、ニア・キャッシュ無効化不可を使用するマップです。

親トピック: **.NET 2.5+** [.NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連資料:

[動的マップの構成オプション](#)

.NET

.NET アプリケーションにおけるトランザクションのためのプログラミング

トランザクションを必要とする .NET アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

.NET

[.NET アプリケーションのためのトランザクション内のデータとの対話](#)

WebSphere® eXtreme Scale クライアントの API では、各スレッドが別々の IGridMapPessimisticTx または IGridMapPessimisticAutoTx オブジェクトを持つ必要があります。IGridMapPessimisticTx オブジェクトの場合は、トランザクションの開始、コミット、またはロールバックを明示的に行うために Transaction プロパティが使用されます。IGridMapPessimisticAutoTx オブジェクトでは、トランザクションの開始、コミット、およびロールバック操作が自動的に行われます。セッションを使用してデータと対話します (Add、Put、および Replace 操作など)。

.NET

[.NET アプリケーションにおけるロックの構成と実装](#)

WebSphere eXtreme Scale Client for .NET からアクセスするバッキング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。マップ・インスタンスのロック・タイムアウト値をオーバーライドすることもできます。ロックを構成したら、マップ内の個々のキーまたはキーのリストをロックすることができます。

.NET

[ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[トランザクション](#)

.NET アプリケーションのためのトランザクション内のデータとの対話

WebSphere® eXtreme Scale クライアントの API では、各スレッドが別々の IGridMapPessimisticTx または IGridMapPessimisticAutoTx オブジェクトを持つ必要があります。IGridMapPessimisticTx オブジェクトの場合は、トランザクションの開始、コミット、またはロールバックを明示的に行うために Transaction プロパティが使用されます。IGridMapPessimisticAutoTx オブジェクトでは、トランザクションの開始、コミット、およびロールバック操作が自動的に行われます。セッションを使用してデータと対話します (Add、Put、および Replace 操作など)。

このタスクについて

IGridMapPessimisticTx および IGridMapPessimisticAutoTx インターフェースは、データを処理する Add、Get、Put、Replace、および Remove などの操作を提供します。IGridMapPessimisticTx インターフェースは、データへの同時アクセスを制御する Lock および GetAndLock などの追加操作を提供します。

手順

- データを追加します。

次のコード・フラグメントは、IGridMapPessimisticTx インターフェースを使用して、新しいトランザクションを開始し、データ・グリッドの項目を作成し、そしてトランザクション全体をコミットする方法を示しています。

```
IGridMapPessimisticTx<String,Person> ptmap;  
ptmap = grid.GetGridMapPessimisticTx<String,Person>("PERSON");  
ptmap.Transaction.Begin();  
Person p = new Person();  
p.name = "John Doe";  
ptmap.Add(p.name, p);  
ptmap.Transaction.Commit();
```

- データを置き換えます。

次のコード・フラグメントは、IGridMapPessimisticTx インターフェースを使用して、新しいトランザクションを開始し、データ・グリッド内の項目をロックしてその値を取得し、項目値を置き換え、そしてトランザクションをコミットする方法を示しています。

```
IGridMapPessimisticTx<String,Person> ptmap;  
ptmap = grid.GetGridMapPessimisticTx<String,Person>("PERSON");  
ptmap.Transaction.Begin();  
Person p = ptmap.GetAndLock("John Doe", LockMode.Upgradable);  
p.age = 30;  
ptmap.Replace(p.name, p);  
ptmap.Transaction.Commit();
```

アプリケーションでは、通常は、単純な get ではなく、GetAndLock メソッドを使用してレコードをロックします。メソッドは、更新済みの値をマップに提供するために呼び出す必要があります。Replace メソッドが呼び出されないと、マップは変更されません。

親トピック: [.NET アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[データ・アクセスおよびトランザクション](#)

.NET アプリケーションにおけるロックの構成と実装

WebSphere® eXtreme Scale Client for .NET からアクセスするバッキング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。マップ・インスタンスのロック・タイムアウト値をオーバーライドすることもできます。ロックを構成したら、マップ内の個々のキーまたはキーのリストをロックすることができます。

始める前に

- 使用したいロック・ストラテジーを決めます。詳しくは、[ロック・ストラテジー](#)を参照してください。
- 動的マップでペシミスティック・ロック・ストラテジーを構成します。詳しくは、[ロック・ストラテジーの構成](#)を参照してください。

手順

1. バッキング・マップでペシミスティック・ロック・ストラテジーを構成します。WebSphere eXtreme Scale Client for .NET はペシミスティック・ロック・ストラテジーのみをサポートします。詳しくは、[ロック・ストラテジーの構成](#)を参照してください。
2. 単一 IGridMapPessimisticTx インスタンスのロック待ちタイムアウトをオーバーライドします。**IGridMapPessimisticTx.LockTimeout** プロパティを使用して、特定の IGridMapPessimisticTx インスタンスのロック・タイムアウト値をオーバーライドします。ロック・タイムアウト値は、新規のタイムアウト値の設定後に開始されたすべてのトランザクションに影響します。このメソッドは、ロック競合が選択トランザクションで起こりうる、あるいは予想される場合に便利です。
3. マップ内の個々のキーまたはキーのリストをロックします。Lock メソッドを使用して、データ・グリッド内のキーをロックするか、またはキーをロックして値がデータ・グリッド内に存在するかを確認します。
 - 次のメソッドはマップ内のキーをロックします。キーが存在する場合は true を返し、キーが存在しない場合は false を返します。

```
bool IGridMapPessimisticTx.Lock(Tkey key, LockMode lockMode);
```

- 次のメソッドはマップ内のキーのリストをロックし、true または false 値のリストを返します。キーが存在する場合は true を返し、キーが存在しない場合は false を返します。

```
IList<bool> IGridMapPessimisticTx.LockAll(IList<TKey> keyList, LockMode lockMode);
```

LockMode は列挙型で以下に示す値を取ることができ、ここでロックしたいキーを指定できます。

- Shared、Upgradable、Exclusive

LockMode パラメーターの設定例を次に示します。

```
ptmap.Transaction.Begin();  
ptmap.Lock(key, LockMode.Upgradable);  
ptmap.Put(key, value);  
ptmap.Transaction.Commit();
```

親トピック: [.NET .NET アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[ロック・タイプ](#)
[ロック・ストラテジー](#)
[デッドロック](#)

関連タスク:

[.NET ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)
[ロック・ストラテジーの構成](#)

ロック・シナリオでの例外処理の実装 (.NET アプリケーション)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

手順

1. 例外をキャッチし、結果のメッセージを表示します。

```
try {  
    ...  
} catch (GridException ge) {  
    System.Console.WriteLine(ge.ToString());  
}
```

LockDeadlockException 例外がスローされると、この例外は内部例外として別の例外の中に含まれることがあります。前のコード・スニペットは、内部例外チェーン全体 (存在する場合) を含む最上位例外を表示します。LockDeadlockException 例外に固有の例外メッセージに、ロック競合の詳細が含まれています。このメッセージの解釈の仕方については、[デッドロックのトラブルシューティング](#)を参照してください。

```
IBM.WebSphere.Caching.Map.LockDeadlockException: Message
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるメッセージを表示します。

2. 例外の後、トランザクションをロールバックします。

```
IGridMapPessimisticTx<String,Person> ptmap;  
ptmap = grid.GetGridMapPessimisticTx<String,Person>("PERSON");  
try {  
    ptmap.Transaction.Begin();  
    Person p = ptmap.Get("Lynn");  
    // Lynn had a birthday, so we make her 1 year older.  
    p.Age++;  
    ptmap.Put(p.name, p);  
    ptmap.Transaction.Commit();  
}  
catch (GridException ge) {  
    System.Console.WriteLine(ge.ToString());  
}  
finally {  
    if ( ptmap.Transaction.Active )  
        ptmap.Transaction.Rollback();  
}
```

コード・スニペットの finally ブロックは、予期しない例外が発生したときにトランザクションがロールバックされるようにしています。LockDeadlockException 例外のみでなく、発生する可能性のあるその他の予期しない例外もすべて処理します。finally ブロックは、commit メソッドの呼び出し時に例外が発生するケースも処理します。この例は、予期しない例外を処理する唯一の方法ではありません。アプリケーションが、発生する予期しない例外のいくつかをキャッチし、そのアプリケーション例外の 1 つを表示するケースも存在するかもしれません。適宜 catch ブロックを追加できますが、アプリケーションは、コード・スニペットがトランザクションを完了せずに終了しないようにする必要があります。

親トピック: [.NET .NET アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[.NET .NET アプリケーションにおけるロックの構成と実装](#)

[ロック・ストラテジーの構成](#)

WebSphere eXtreme Scale Client for .NET 用のデータ・グリッド・セキュリティの構成

Secure Sockets Layer (SSL) を介して通信し、ユーザー/パスワード認証ロジックを使用するように .NET および Java™ を構成できます。

始める前に

ご使用の環境用の key.jks ファイルおよび trust.jks ファイルを用意しておく必要があります。

手順

サーバーでセキュリティを使用可能にして構成します。セキュリティがサーバーでまだ構成されていない場合は、以下の手順を使用して、外部オーセンティケーター・サンプルでセキュリティを構成します。

- サンプル・セキュリティ・ファイルを取得します。 security_extauth.zip ファイルのサンプル・ファイルを、[WebSphere eXtreme Scale wiki](#) からダウンロードします。
 - xsjaas3.config: Java Authentication and Authorization Service (JAAS) 構成を定義します。
 - sampleKS3.jks: JAAS ユーザーおよびパスワードの値の鍵ストアが入っています。
 - security3.xml: セキュリティに使用するオーセンティケーターを定義します。
- xsjaas3.config ファイルを編集し、sampleKS3.jks ファイルのパスを修正します。
- サンプル sampleKS3.jks ファイルではなく、独自の秘密鍵ストアを生成する場合は、**keytool** ユーティリティを使用して秘密鍵を生成します。

```
keytool -genkey -alias myalias -keysize 2048 -keystore key.jks -keyalg rsa -dname "CN=www.mydomain.com" -storepass password -keypass password -validity 3650
```

- sampleServer.properties を編集して、セキュリティを使用可能にします。 sampleServer.properties ファイルは、[wxs_install_root](#)¥properties ディレクトリーにあります。以下のプロパティ値のコメントを外して編集します。

```
securityEnabled=true
secureTokenManagerType=none
alias=ogsample
contextProvider=IBMJSSE2
protocol=SSL
keyStoreType=JKS
keyStore=../../../../../xio.test/etc/test/security/key.jks
keyStorePassword=ogpass
trustStoreType=JKS
trustStore=../../../../../xio.test/etc/test/security/trust.jks
trustStorePassword=ogpass
```

次のタスク

WebSphere® eXtreme Scale Client for .NET 用の Transport Layer Security (TLS) を構成します。詳しくは、[WebSphere eXtreme Scale Client for .NET 用の TLS の構成](#)を参照してください。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

WebSphere eXtreme Scale Client for .NET 用の TLS の構成

WebSphere® eXtreme Scale Client for .NET 用に Transport Layer Security (TLS) を構成することができます。

始める前に

- WebSphere eXtreme Scale Client for .NET 構成に追加する、鍵ストアと、関連付けられたパスワードが必要です。

手順

1. オプション: keytool ユーティリティを使用して、key.jks ファイルから公開証明書を抽出します。

```
keytool -export -alias myalias -keystore key.jks -file public.cer -storepass password
```

証明書管理ツール certmgr.msc を使用して、この公開鍵を Windows の証明書ストアにインポートして、鍵を「信頼されたルート証明機関」または「信頼されたユーザー」証明書フォルダーにインポートします。(client.properties ファイルの **keyStore** プロパティは、このファイルを指すことができます)

2. Client.Net.properties ファイルを編集して、以下のプロパティの値を含めます。

```
securityEnabled=true
credentialAuthentication=supported
authenticationRetryCount=3
credentialGeneratorAssembly=IBM.WebSphere.Caching.CredentialGenerator,Version=8.6.0.0,
Culture=neutral,PublicKeyToken=b439a24ee43b0816
credentialGeneratorProps=manager manager1
transportType=ssl-required
publicKeyFile=<name>.cer
```

credentialGeneratorProps プロパティの値 manager manager1 は、Credential オブジェクトでサーバーに提供されるユーザー名とパスワードの値として使用されます。

publicKeyFile プロパティは、.NET ランタイムの相対パスとして設定されます。**publicKeyFile** プロパティが設定されていない場合は、Windows 証明書ストアで public.cer ファイルが検索されます。

publicKeyFile プロパティが設定されている場合は、指定されたファイルが SSL 公開証明書ファイルとして使用されます。指定されたファイルが見つからない場合は、.NET クライアントは、一致する public.cer ファイルを証明書ストアで見つけようとしています。

3. オプション: credentialGeneratorProps プロパティの値をエンコードします。プロパティ値をエンコードするには、Client.Net.properties ファイルを、Java ベースの WebSphere eXtreme Scale クライアントインストール済み環境を持つコンピューターに転送します。**FilePasswordEncoder** ユーティリティを実行して credentialGeneratorProps プロパティをエンコードします。

```
FilePasswordEncoder.bat Client.Net.Properties credentialGeneratorProps
```

このユーティリティをプロパティ・ファイルに対して実行すると、ファイル内のすべてのコメントが削除されます。

4. `net_client_home`¥IBM.WebSphere.Caching.CredentialGenerator.dll を `net_client_home`¥sample¥SimpleClient¥bin¥<ConfigurationName> ディレクトリーにコピーします。
5. `ConfigurationName` プロジェクト・コンテキストを使用してサンプルをビルドします。サンプルをサーバーに対して実行します。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング

WebSphere® eXtreme Scale Client for .NET からサーバー・サイドに資格情報を送信するには、ICredentialGenerator および ICredential インターフェースを実装する必要があります。これらのインターフェースでは、データ・グリッドに渡されてサーバー・サイドで解釈される資格情報オブジェクトが生成されます。サーバー・サイドで、対応するプラグインが資格情報オブジェクトを解釈します。

このタスクについて

認証を実行するには、.NET アプリケーションが以下のインターフェースを実装している必要があります。

- ICredential: Credential は、クライアント資格情報 (ユーザー ID とパスワードのペアなど) を表します。
- ICredentialGenerator: CredentialGenerator は、資格情報を生成するための資格情報ファクトリーを表します。

.NET クライアント・アプリケーションが認証を必要とするサーバーに接続すると、クライアントはクライアント資格情報を提供するように要求されます。クライアント資格情報は、ICredential インターフェースによって表されます。クライアント資格情報には、ユーザー名とパスワードのペア、Kerberos チケット、クライアント証明書、またはクライアントとサーバーが同意する任意の形式でのデータがあります。このインターフェースでは、equals(Object) メソッドおよび hashCode メソッドを定義します。Credential オブジェクトをサーバー・サイドの鍵として使用することによって認証済み Subject オブジェクトがキャッシュされるため、この 2 つのメソッドは重要です。ICredentialGenerator インターフェースを使用して資格情報を生成することもできます。このインターフェースは、資格情報に有効期限がある場合に役立ちます。Credential プロパティを取得することにより、新規資格情報が生成されます。

また、提供されている CredentialGenerator プラグインを使用して、Client.Net.Properties ファイルの **credentialGeneratorProps=** 設定に基づいた資格情報を作成することもできます。資格情報プラグインを定義する追加設定は、**credentialGeneratorAssembly** および **credentialGeneratorClass** です。

手順

.NET アプリケーションで、ICredentialGenerator および ICredential インターフェースを実装します。

要確認: このクライアントの資格情報プラグインを実装する場合は、WebSphere eXtreme Scale Client for .NET からの認証資格情報を解釈して受け取ることができる、対応サーバーの資格情報プラグインも実装する必要があります。

以下の例を使用して、アプリケーションを開発できます。

- [例: .NET アプリケーション用のユーザー・パスワード資格情報の実装](#)
- [例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装](#)

.NET

[例: .NET アプリケーション用のユーザー・パスワード資格情報の実装](#)

この例を使用して、ICredential インターフェースの実装を独自に作成することができます。ユーザー・パスワードの資格情報では、ユーザー ID とパスワードを保管します。

.NET

[例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装](#)

この例を使用して、ICredentialGenerator インターフェースの実装を独自に作成することができます。このインターフェースはユーザー ID およびパスワードを受け取ります。UserPasswordCredential オブジェクトには、読み取り専用の資格情報プロパティから取得される、ユーザー ID およびパスワードが含まれています。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連資料:

.NET

[例: .NET アプリケーション用のユーザー・パスワード資格情報の実装](#)

.NET

[例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装](#)[クライアント・プロパティ・ファイル](#)

関連情報:

[ICredential インターフェース](#)[ICredentialGenerator インターフェース](#)

例: .NET アプリケーション用のユーザー・パスワード資格情報の実装

この例を使用して、ICredential インターフェースの実装を独自に作成することができます。ユーザー・パスワードの資格情報では、ユーザー ID とパスワードを保管します。

UserPasswordCredential.cs

```
// Module : UserPasswordCredential.cs

using System;
using IBM.WebSphere.Caching.Security;

namespace com.ibm.websphere.objectgrid.security.plugins.builtins
{
    public class UserPasswordCredential : ICredential
    {
        private String ivUserName;

        private String ivPassword;

        /// <summary>
        ///Creates a UserPasswordCredential with the specified user name and
        /// password.
        ///
        /// ArgumentException if userName or password is null
        /// </summary>
        /// <param name="userName">the user name for this credential</param>
        /// <param name="password">the password for this credential</param>
        public UserPasswordCredential(String userName, String password)
        {
            if (userName == null || password == null) {
                throw new ArgumentException("User name and password cannot be null.");
            }
            this.ivUserName = userName;
            this.ivPassword = password;
        }

        /// <summary>Gets the user name for this credential.</summary>
        /// <returns>the user name argument that was passed to the constructor
        ///or the setUsername(String) method of this class </returns>
        public String GetUserName() {
            return ivUserName;
        }

        /// <summary>Sets the user name for this credential.
        ///ArgumentException if userName is null
        /// </summary>
        /// <param name="userName">userName the user name to set.</param>
        public void SetUserName(String userName) {
            if (userName == null) {
                throw new ArgumentException("User name cannot be null.");
            }
            this.ivUserName = userName;
        }

        /// <summary>Gets the password for this credential.
        /// </summary>
        /// <returns>the password argument that was passed to the constructor or the
        setPassword(String) method of this class</returns>
        public String GetPassword() {
            return ivPassword;
        }

        /// <summary>Sets the password for this credential.
        ///ArgumentException if password is null
        /// </summary>
        /// <param name="password">the password to set.</param>
```



```

public void SetPassword(String password) {
    if (password == null)
    {
        throw new ArgumentException("Password cannot be null.");
    }
    this.ivPassword = password;
}

/// <summary>Checks two UserPasswordCredential objects for equality.
///<p>
/// Two UserPasswordCredential objects are equal if and only if their user names
/// and passwords are equal.
/// </summary>
/// <param name="o">the object we are testing for equality with this object.
</param>
/// <returns>>true if both UserPasswordCredential objects are equivalent.</returns>
public bool Equals(ICredential credential)
{
    if (this == credential) {
        return true;
    }
    if (credential is UserPasswordCredential) {
        UserPasswordCredential other = (UserPasswordCredential)credential;
        return other.ivPassword.Equals(ivPassword) &&
other.ivUserName.Equals(ivUserName);
    }
    return false;
}

/// <summary>Returns the hashcode of the UserPasswordCredential object.
/// </summary>
/// <returns>return the hash code of this object</returns>
public override int GetHashCode() {
    int ret = ivUserName.GetHashCode() + ivPassword.GetHashCode();
    return ret;
}

/// <summary>this.Object as a string
/// </summary>
/// <returns>return the string presentation of the UserPasswordCredential object.
</returns>
public override String ToString() {
    return typeof(UserPasswordCredential).FullName + "[" + ivUserName +
",xxxxxx]";
}

}
}

```

親トピック: [.NET WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連タスク:

[.NET WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連情報:

[ICredential インターフェース](#)

[ICredentialGenerator インターフェース](#)

例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装

この例を使用して、ICredentialGenerator インターフェースの実装を独自に作成することができます。このインターフェースはユーザー ID およびパスワードを受け取ります。UserPasswordCredential オブジェクトには、読み取り専用の資格情報プロパティから取得される、ユーザー ID およびパスワードが含まれています。

UserPasswordCredentialGenerator.cs

```
// Module : UserPasswordCredentialGenerator.cs
//
// Source File Description: Reference Documentation
//
using System;
using System.Security.Authentication;
using IBM.WebSphere.Caching.Security;
using com.ibm.websphere.objectgrid.security.plugins.builtins;

namespace IBM.WebSphere.Caching.Security
{
    public class UserPasswordCredentialGenerator : ICredentialGenerator
    {
        private String ivUser;

        private String ivPwd;

        public ICredential Credential { get { return _getCredential(); } }

        public string Properties { set { _setProperties(value);} }

        public UserPasswordCredentialGenerator()
        {
            ivUser = null;
            ivPwd = null;
        }

        public UserPasswordCredentialGenerator(String user=null, String pwd=null)
        {
            ivUser = user;
            ivPwd = pwd;
        }

        /// <summary>Creates a new UserPasswordCredential object using this object's user
        name and password.
        /// </summary>
        /// <returns>new UserPasswordCredential instance</returns>
        private ICredential _getCredential()
        {
            try
            {
                ICredential MyCredential = new UserPasswordCredential(ivUser, ivPwd) as
                ICredential;
                return (ICredential) MyCredential;
            }
            catch (Exception e)
            {
                AuthenticationException CannotGenerateCredentialException = new
                AuthenticationException(e.ToString());
                throw CannotGenerateCredentialException;
            }
        }

        /// <summary>Gets the password for this credential generator.
        /// </summary>
        /// <returns>the password argument that was passed to the constructor</returns>
        public String getPassword()
    }
}
```

```

    {
        return ivPwd;
    }

    /// <summary>Gets the user name for this credential.
    /// </summary>
    /// <returns>the user argument that was passed to the constructor of this
class</returns>
    public String getUsername()
    {
        return ivUser;
    }

    /// <summary>Sets additional properties namely a user name and password.
    ///Throws ArgumentException if the format is not valid
    /// </summary>
    /// <param name="properties">properties a properties string with a user name and
a password separated by a blank.</param>
    private void _setProperty(string properties)
    {
        String token = properties;
        char[] Separator = { ' ' };
        String[] StringProperty = properties.Split(Separator);
        if (StringProperty.Length != 2)
        {
            throw new ArgumentException(
                "The properties should have a user name and password and separated by
a space.");
        }

        ivUser = StringProperty[0];
        ivPwd = StringProperty[1];
    }

    /// <summary>Checks two UserPasswordCredentialGenerator objects for equality.
    ///<p>
    ///Two UserPasswordCredentialGenerator objects are equal if and only if
    ///their user names and passwords are equal.
    /// </summary>
    /// <param name="obj">the object we are testing for equality with this object.
</param>
    /// <returns><code>>true</code> if both UserPasswordCredentialGenerator objects are
equivalent</returns>
    public override bool Equals(Object obj)
    {
        if (obj == this)
        {
            return true;
        }

        if (obj != null && obj is UserPasswordCredentialGenerator)
        {
            UserPasswordCredentialGenerator other =
(UserPasswordCredentialGenerator)obj;

            Boolean bothUserNull = false;
            Boolean bothPwdNull = false;

            if (ivUser == null)
            {
                if (other.ivUser == null)
                {
                    bothUserNull = true;
                }
                else
                {
                    return false;
                }
            }
        }
    }

```

```

        }

        if (ivPwd == null)
        {
            if (other.ivPwd == null)
            {
                bothPwdNull = true;
            }
            else
            {
                return false;
            }
        }
        return (bothUserNull || ivUser.Equals(other.ivUser)) && (bothPwdNull ||
ivPwd.Equals(other.ivPwd));
    }
    return false;
}

/// <summary>Returns the hashcode of the UserPasswordCredentialGenerator object.
/// </summary>
/// <returns>the hash code of this object</returns>
public override int GetHashCode()
{
    return ivUser.GetHashCode() + ivPwd.GetHashCode();
}

}

}

```

親トピック: [.NET WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連タスク:

[.NET WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連情報:

[ICredential インターフェース](#)

[ICredentialGenerator インターフェース](#)

WebSphere® eXtreme Scale Client for .NET のためのカスタム資格情報のプログラミング

マップのユーザー資格情報を指定することができます。マップのユーザー資格情報があれば、2人のユーザーが Web アプリケーションを介して同じデータ・グリッドと対話することができます。

手順

1. client.properties ファイルでユーザー資格情報を設定します。

```
credentialAuthentication=required
authenticationRetryCount=3
credentialGeneratorAssembly=IBM.WebSphere.Caching.CredentialGenerator,
Version=8.6.0.0, Culture=neutral, PublicKeyToken=b439a24ee43b0816
credentialGeneratorClass=IBM.WebSphere.Caching.Security.UserPasswordCredentialGenerator
credentialGeneratorProps=manager manager1
```

2. アプリケーション・プロジェクト内の IBM.WebSphere.Caching.CredentialGenerator.dll ファイルへの参照を追加します。このプラグイン DLL には ICredentialGenerator 実装が含まれています。
3. .NET アプリケーションで ICredentialGenerator API を使用します。

```
//GridManagerFactory.GetGridManager
IGridManager gm = GridManagerFactory.GetGridManager();
//IGridManager.Connect
ICatalogDomainInfo cdi =
gm.CatalogDomainManager.CreateCatalogDomainInfo(hostAndPort);
ctx = gm.Connect(cdi, "client.properties");
//IGridManager.GetGrid
IGrid grid = gm.GetGrid( ctx, "Grid");
ICredentialGenerator credGenManager = new UserPasswordCredentialGenerator("manager",
"manager1");
ICredentialGenerator credGenOperator = new
UserPasswordCredentialGenerator("operator", "operator1");
//IGrid.GetGridMap
IGridMapPessimisticAutoTx<Object, Object> gridMap1 =
grid.GetGridMapPessimisticAutoTx<Object, Object>("Map1", credGenManager);

IGridMapPessimisticAutoTx<Object, Object> gridMap2 =
grid.GetGridMapPessimisticAutoTx<Object, Object>("Map1", credGenOperator);
```

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

モニター

ご使用の WebSphere® DataPower® XC10 アプライアンス に関するさまざまな局面をモニターでき、「タスク」ビューによる構成変更の状況や、ユーザー・インターフェース内の「モニター」ビューで見る データ・グリッド のパフォーマンスが含まれます。

始める前に

ユーザー・インターフェース内の「タスク」ビューまたは「モニター」ビューでデータを見るには、ご使用のアプライアンスが構成されており、アプリケーションからの新規エントリーを受け取るように データ・グリッド がアプリケーションからの新規エントリーを受け入れるように作成しなければなりません。

2.5+ [アプライアンスの始動状況のモニター](#)

アプライアンスの始動状況をモニターすることができます。始動状況をモニターすることは、問題の判別とトラブルシューティングに役立ちます。

[ユーザー・インターフェースでのデータ・グリッドのモニター](#)

WebSphere DataPower XC10 アプライアンス のグラフ機能を使用して、ご使用の環境における データ・グリッド の全体的なパフォーマンスを表示することができます。

[タスクによるアクティビティのモニター](#)

例えば集合にアプライアンスを追加するなど、管理上の変更の進行をモニターするためにタスクが使用できます。

[xscmd ユーティリティーによるモニター](#)

xscmd ユーティリティーを使用すれば、ご使用のアプライアンス上で 実行されるデータ・グリッドに関するテキスト情報を表示できます。

[CSV ファイルによるモニター](#)

モニター・データは自動的に CSV ファイルに書き込まれます。これらの CSV ファイルには、サーバー、マップ、またはデータ・グリッドに関する情報を含めることができます。

[xsadmin ユーティリティーによるモニター](#)

xsadmin ユーティリティーを使用して、WebSphere DataPower XC10 アプライアンス上で 実行しているデータ・グリッドに関するテキスト情報をフォーマット設定し、表示できます。このサンプル・ユーティリティーは現在のデプロイメント・データの解析とディスクバリーの方法を提供するもので、カスタム・ユーティリティーの作成基盤として使用することができます。

2.5+ [環境のヘルスのモニター](#)

メッセージ・センターでは、ログおよび初期障害データ・キャプチャー機能 (FFDC) メッセージのイベント通知の集約ビューが示されます。これらのイベント通知は、Web コンソールのメッセージ・センター、**xscmd** ユーティリティー、ヘルス・ログ・ファイル、HTTP コマンド・インターフェース、または MBeans を使用したプログラマチック方式で表示することができます。

[Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

Simple Network Monitoring Protocol (SNMP) サポートにより、IBM® WebSphere DataPower XC10 アプライアンス の状況を、データ・センターのシステムのより大きなグループの一部として、モニターできます。SNMP モニターは、システム問題を素早く通知し、それらに迅速に応答する機能を強化します。

[リモート・ロギングの構成](#)

リモート・ロギングを使用可能にすれば、ログ・エントリーをアプライアンス外のリモート・サーバーに保存することができます。リモート・ロギングは、問題の分離や長期にわたる動作のモニターを容易にするために詳細デバッグ・ログ・レベルを設定する必要があるときに役立つことがあります。

アプライアンスの始動状況のモニター

2.5+ アプライアンスの始動状況をモニターすることができます。始動状況をモニターすることは、問題の判別とトラブルシューティングに役立ちます。

始める前に

- アプライアンスを初期化、始動、または再始動します。
- これにはxcadmin ユーザー名およびパスワードが必要です。

このタスクについて

始動ユーザー・インターフェース・パネルまたは **start-progress** コマンドを使用して、アプライアンスの始動状況をモニターすることができます。

アプライアンス始動ユーザー・インターフェース・パネル

このユーザー・インターフェース・パネルを使用して、始動状況 (完了パーセントや個々のサーバーおよびプロセスの状況など) を表示することができます。状況は 15 秒ごとに自動的にリフレッシュします。

start-progress コマンド

start-progress コマンドはコマンド行インターフェースで実行します。このコマンドは始動状況の現在のスナップショットを返します。最新の始動状況を取得するには、このコマンドをもう一度実行してください。

手順

- ユーザー・インターフェースでアプライアンス始動状況パネルを表示します。
 1. 次の URL を使用してアプライアンス始動状況パネルにアクセスします。

```
https://<appliance_hostname>:9443/
```

始動の進行状況がパーセントで表示されます。

2. 100 パーセントになる前に進行が止まった場合は、ログ・ファイルをダウンロードして問題のトラブルシューティングを行うことができます。「[ログ・ファイルのダウンロード](#)」をクリックします。
- コマンド行インターフェースでアプライアンス始動状況を表示します。
 1. コマンド行インターフェースにログインします。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
 2. **start-progress** コマンドを実行します。アプライアンス始動シーケンスの状況が表示されます。
 3. 最新の状況を取得するには、**start-progress** コマンドをもう一度実行します。

例

次の例は成功した始動シーケンスです。

```
Volume 1 mounted
Volume 2 mounted
Grid administrative service started.
Catalog server started.
Grid configuration service started.
Container server 01 started.
Container server 02 started.
Container server 03 started.
Container server 04 started.
Container server 05 started.
Container server 06 started.
Container server 07 started.
Container server 08 started.
Administrative console started.
```

- 集合内の最初の 3 つのアプライアンスでカタログ・サーバーが実行されます。アプライアンスが 3 つを超えている場合は、カタログ・サーバー始動メッセージが表示されないことがあります。
- 始動されるコンテナ・サーバーの数は、使用しているトランスポート・タイプによって決まります。IBM eXtremeIO (XIO) の場合は、デフォルトで 8 つのコンテナ・サーバーが始動されます。オブジェクト・リクエスト・ブローカー (ORB) の場合は、デフォルトで 16 のコンテナ・サーバーが始動されます。トランスポート・タイプについて詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。
- Administrative console started. 状況が表示されたら、ユーザー・インターフェースにログインすることができます。

次のタスク

- **request suspend** および **request force-suspend** コマンドを実行すると、アプライアンス始動パネルも延期状態を示します。

親トピック: [モニター](#)

ユーザー・インターフェースでのデータ・グリッドのモニター

WebSphere® DataPower® XC10 アプライアンス のグラフ機能を使用して、ご使用の環境における データ・グリッドの全体的なパフォーマンスを表示することができます。

始める前に

データ・グリッドを作成し、アプリケーションで データ・グリッドを使用するように構成したら、統計が使用可能になるまで少し時間を置きます。例えば、動的キャッシュ データ・グリッド (data grid)を使用する場合、動的キャッシュを実行する WebSphere Application Server がアプライアンス上の動的キャッシュ データ・グリッド (data grid) に接続するまで、統計は使用できません。集合を使用している場合は、統計が使用可能になる前に、集合の初期設定が完了していなければなりません。一般的には、統計における変化を見るために、主要な構成変更の後で 1 分ほど待ちます。

このタスクについて

グラフの性質

日、週、および月の一連のデータは、長期間保持されます。しかし、「最終時刻」の一連のデータは、メモリー内のみ保管されます。最終時刻データは、次の場所に保管されます。

- データ・グリッドのメモリー内
- コンソール・キャッシュのメモリー内

これらのメモリー内のどちらかの場所でのみ統計データの部分が逸失した場合、そのデータはまだ使用可能です。しかし、両方の場所でデータが逸失した場合は、グラフの中にデータは表示されません。

メモリー内のデータ損失の条件:

- データ・グリッドが移動された場合、統計が集計されていないときに期間が発生します。
- アプライアンスが再始動したとき、データ・グリッドとコンソール・キャッシュの両方のメモリー・データが消去されます。
- コンソール・キャッシュのメモリーがいっぱいになると、最長未使用時間データはコンソール・キャッシュから除去されます。コンソール・キャッシュは、2048 の統計項目を保持します。
- コンソールが 30 分間非アクティブの場合、コンソール・プロセスが再始動し、メモリーは消去されます。

統計の収集では、最終時刻の一連のデータの再集計を続行します。

ヒント: グラフ内の任意のデータ・ポイントに関するより具体的な情報を表示するには、そのデータ・ポイントの上にマウス・ポインターを移動させてください。

手順

- データ・グリッド全体のパフォーマンスを表示するには、「モニター」 > 「データ・グリッドの概要」をクリックします。このページには、以下の情報が含まれています。

「使用容量」タブ

現行データ・グリッドの使用容量の分布

このグラフには、アプライアンスまたは集合の使用可能容量の合計、基本データおよびレプリカ・データ・コピーに使用されている容量の合計、容量限度が設定されている場合の残りの制限容量、使用容量が最も大きいデータ・グリッドのビューが含まれています。次のオプションを使用してデータをソートできます。上位 25 のデータ・グリッドのみが表示されます。

- 最大使用容量コンシューマー (Largest used capacity consumers)
- 最大使用済み制限容量の割合 (Largest percent of limited capacity used)

一定時間の使用容量

このグラフには、選択した期間の使用容量が表示されます。

「平均スループット」タブ

トランザクション/秒で測定した平均スループットによるトップ 5 の最もアクティブな データ・グリッド

このグラフには、平均スループット (トランザクション/秒で測定) で編成された、トップ 5 のデータ・グリッドのリストが含まれています。

一定時間の平均スループット

このグラフには、選択した期間の平均スループット (トランザクション/秒で測定) が表示されます。

「トランザクション平均時間」タブ

平均トランザクション時間 (ミリ秒) が下位 5 位までの低速なデータ・グリッド

このグラフには、平均トランザクション時間で編成された、下位 5 の最も低速なデータ・グリッドのリストが含まれています。

一定時間の平均トランザクション

このグラフには、選択した期間の平均トランザクションが表示されます。

- 個々のデータ・グリッドを見るには、「モニター」 > 「個別データ・グリッドの概要」 > 「`data_grid_name`」をクリックします。このページには、最後の 30 秒間のキャッシュ・エントリー数、平均トランザクション時間、平均スループット、キャッシュのヒット率、および制限容量の割合を含むサマリーが表示されます。以下のグラフも表示できます。

使用容量

このグラフには、キャッシュの使用容量と実際のエントリー数の比較、および構成されているキャッシュの容量限度が示されます。使用容量には基本データとレプリカ・データの両方が含まれます。表示される時刻範囲を、次のように編集できます。最終時刻、最終日付、先週、先月。グラフの表示内容の詳細レベルは、選択した時刻範囲によって変化します。

キャッシュ・エントリー数の解釈:

- **すべてのデータ・グリッドの場合:** キャッシュ・エントリー数はプライマリー断片のみを表します。プライマリー断片とレプリカ断片の両方を組み合わせた断片のキャッシュ・エントリー数を表示するには、`xscmd -c showMapSizes` コマンドを使用します。
- **動的キャッシュ・データ・グリッドの場合:** 動的キャッシュ・データ・グリッドごとに、デフォルトで 166 個のキャッシュ・エントリーが作成されます。各動的キャッシュ・グリッドには 83 のパーティションがあり、各パーティションまたは断片は、動的キャッシュ・グリッドの 2 個のエントリーで初期化されます。したがって、初期化時のキャッシュ・エントリーの数は、166 になります。これらのキャッシュ・エントリーには、WebSphere Application Server の動的キャッシュ・プロバイダー統計および動的キャッシュ構成が含まれます。そのため、データを動的キャッシュ・データ・グリッドに追加する前に、モニター・パネルに 166 個のキャッシュ・エントリーが表示されます。
- **セッション・データ・グリッドの場合:** データ・グリッド内のエントリー数には、セッション数、すべてのセッションの属性数、および追い出し表エントリーのカウントが含まれます。Web コンテナによって無効化されていないタイムアウト・セッションが存在する場合に、追い出し表にデータが追加されます。追い出し表には、アプリケーション・サーバーのフェイルオーバー時にのみデータが追加されます。

キャッシュの使用

このグラフは、キャッシュに対する成功した照会の数の視覚化に便利です。キャッシュの試行数、キャッシュのヒット数、およびキャッシュのヒット率をグラフに表示できます。

平均スループット

このグラフは、指定された時刻範囲で処理された、1 秒当たりの平均トランザクション数と、各トランザクションの平均時間長を示します。

- 特定のデータ・グリッドに関するより詳細な情報を表示するには、「モニター」 > 「データ・グリッドの詳細レポート」をクリックします。ツリーにはご使用の構成におけるすべてのデータ・グリッドが表示されています。特定のデータ・グリッドをドリルダウンして、そのデータ・グリッドの一部であるマップを表示します。詳細情報は、データ・グリッド名またはマップをクリックします。

データ・グリッドの詳細

使用容量と、データ・グリッドが属するゾーンのリストを表示できます。使用容量には基本データとレプリカ・データの両方が含まれます。データ・グリッド表示内の上位 25 のマップによって消費された容量を示すグラフです。合計プールも表示でき、これにはゾーンによる容量が含まれます。グラフは、上位 25 ゾーン全体で消費されるデータ・グリッド容量を示します。

マップの詳細

各グリッド内のマップに関する詳細情報を表示でき、マップ内のプライマリー・キャッシュ・エントリーの合計数、平均スループット、平均トランザクション時間、および上位 25 区画ごとに分割されたマップの合計容量などの詳細が含まれます。

親トピック: [モニター](#)

関連情報:

[入門チュートリアル・レッスン 4: 環境のモニター](#)

タスクによるアクティビティのモニター

例えば集合にアプライアンスを追加するなど、管理上の変更の進行をモニターするためにタスクが使用できます。

このタスクについて

初めから終わりまでタスクの状態を表示できます。タスクには、待機、成功、または失敗という全般的な状況があります。

表 1. 全般的なタスク状況のアイコン

アイコン	説明
待機 (🕒)	タスクはタスク・キューに入りましたが、まだ実行されていません。
実行中 (🕒)	タスクは実行中です。
成功 (✅)	タスクはエラーなしで完了しました。
失敗 (❌)	タスクは完了しませんでした。発生したエラーについては、タスク内のメッセージを参照してください。

各タスクは、タスク状況の詳細を示す 1 つ以上のメッセージから成ります。

表 2. タスク・メッセージのアイコン

アイコン	説明
情報 (ℹ️)	メッセージには、実行中のタスクのステップに関する情報が含まれます。
エラー (❌)	タスクは完了しませんでした。発生したエラーについては、タスク内のメッセージを参照してください。
2.5+ 警告 (⚠️)	メッセージには、タスクが失敗する可能性のあることを知らせる情報が含まれています。ただし、タスクは成功する可能性もあります。

手順

1. ユーザー・インターフェースで、「タスク」をクリックします。
2. 詳しくは、タスク名をクリックすると特定のタスクを表示できます。各タスクには、タスクのタイプ、タスクの開始および完了日時、さらにタスクの進行に関するより詳しい情報を表示するための特定の状況メッセージのリストが含まれます。タスクが合格または失敗の状況で完了するまで、ページが 10 秒ごとにリフレッシュして、タスクの状況を更新します。
3. **2.5+** オプション: 完了したタスクを削除します。完了したタスクの状況は成功または失敗です。「完了したタスクをすべて削除」(🗑️) をクリックします。このアイコンをクリックする場合は、すべてのタスクが削除されてもかまわないことを確認してください。「OK」をクリックすると、完了したタスクがすべて削除されます。状況が待機または実行中のタスクは削除されません。

親トピック: [モニター](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

xscmd ユーティリティーによるモニター

xscmd ユーティリティーを使用すれば、ご使用のアプライアンス上で実行されるデータ・グリッドに関するテキスト情報を表示できます。

始める前に

- xscmd ユーティリティーの開始について詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

このタスクについて

xscmd ユーティリティーを使用して、マップの内容など、データ・グリッドの現在のレイアウトおよび特定の状態を表示できます。この例では、このタスクのデータ・グリッドのレイアウトは、MapSetA マップ・セットに属する 1 つの MapA マップを持つ 1 つの ObjectGridA データ・グリッドから成ります。この例では、データ・グリッドにすべてのアクティブ・コンテナを表示する方法と、MapA マップのマップ・サイズに関するフィルタリングされたメトリックを印刷する方法を説明します。使用できるコマンド・オプションをすべて知りたい場合は、引数なしか、または **-help** オプションを付けて xscmd ユーティリティーを実行してください。

手順

1. xscmd ユーティリティーを使用して、環境をモニターします。

- すべてのサーバーの統計を使用可能にするには、以下のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c setStatsSpec -spec ALL=enabled -g ObjectGridA`
 - **Windows** `xscmd.bat -c setStatsSpec -spec ALL=enabled -g ObjectGridA`
- データ・グリッドのすべてのオンライン・コンテナ・サーバーを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showPlacement -g ObjectGridA -ms MapSetA`
 - **Windows** `xscmd.bat -c showPlacement -g ObjectGridA -ms MapSetA`

すべてのコンテナ情報が表示されます。

重要: Transport Layer Security/Secure Sockets Layer (TLS/SSL) が使用可能であるときにこの情報を取得するには、JMX サービス・ポートを設定してカタログ・サーバーおよびコンテナ・サーバーを始動する必要があります。JMX サービス・ポートを設定するには、**startOgServer** または **startXsServer** スクリプトで **-JMXServicePort** オプションを使用するか、ServerProperties インターフェイスで `setJMXServicePort` メソッドを呼び出すことができます。

- ObjectGridA データ・グリッドのマップについての情報を表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showMapSizes -g ObjectGridA -ms MapSetA`
 - **Windows** `xscmd.bat -c showMapSizes -g ObjectGridA -ms MapSetA`
- カタログ・サービスに接続して、カタログ・サービス・ドメイン全体の MapA マップに関する情報を表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showMapSizes -g ObjectGridA -ms MapSetA -m MapA -cep CatalogMachine:6645`
 - **Windows** `xscmd.bat -c showMapSizes -g ObjectGridA -ms MapSetA -m MapA -cep CatalogMachine:6645`
- 構成内の構成されている配置とランタイムの配置を表示するには、次のいずれかのコマンドを実行します。
 - `xscmd -c placementServiceStatus`
 - `xscmd -c placementServiceStatus -g ObjectGridA -ms MapSetA`
 - `xscmd -c placementServiceStatus -ms MapSetA`
 - `xscmd -c placementServiceStatus -g ObjectGridA`

配置情報については、全体の構成、1 つのデータ・グリッド、1 つのマップ・セット、またはデータ・グリッドとマップ・セットの組み合わせを表示するそれぞれのコマンドを使用できます。

2. 環境内の複製状態の要約を表示します。

- 各コンテナ・サーバーの未処理レビジョンの要約を表示します。**-ct** 引数を指定して特定のコンテナ・サーバーに対してコマンドを実行できます。または、引数を含めずに、すべてのコンテナ・サーバーに対して実行できます。
 - **UNIX** `./xscmd.sh -c showReplicationState -ct container1`
 - **Windows** `xscmd.bat -c showReplicationState -ct container1`

このコマンドの出力の情報には、アウトバウンド複製およびインバウンド複製が含まれています。アウトバウンド複製には、当該コンテナ・サーバー上のプライマリ断片から他のコンテナ・サーバー上のレプリカ断片にプッシュアウトする必要がある変更が含まれています。インバウンド複製には、他のコンテナ・サーバー上のプライマリ断片から当該コンテナ・サーバー上のレプリカにプッシュする必要がある変更が含まれています。これらの統計により、複製のヘルスを把握できます。コンテナ・サーバー上の未処理レビジョンの数が突然多くなった場合は、コンテナの問題が存在する可能性があります。

- カタログ・サービス・ドメイン間の断片の未処理レビジョンの要約を表示します。特定のコンテナ・サーバーおよびカタログ・サービス・ドメインに対してコマンドを実行できます。または、引数を含めずに、構成全体に対して実行できます。

- **UNIX** `./xscmd.sh -c showDomainReplicationState -dom domainA -ct container1`

- **Windows** `xscmd.bat -c showDomainReplicationState -dom domainA -ct container1`
このコマンドの出力の情報には、リンクされた各カタログ・サービス・ドメインの各コンテナ・サーバーの未処理リビジョンの要約が含まれています。このコマンドは、各プライマリ断片と、別のカタログ・サービス・ドメイン内にある対応するリモート・プライマリ断片との間で複製する必要がある変更を返します。

親トピック: [モニター](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

CSV ファイルによるモニター

モニター・データは自動的に CSV ファイルに書き込まれます。これらの CSV ファイルには、サーバー、マップ、またはデータ・グリッドに関する情報を含めることができます。

このタスクについて

デフォルトで CSV ファイルに書き込まれるモニター・データ。アプライアンス上で実行中のサーバーの履歴データをダウンロードし、分析することができます。サーバーが始動すると、データの収集が開始されます。その後は、この CSV ファイルをいつでもダウンロードでき、自由にファイルを使用できます。

手順

1. CSV ファイルをダウンロードします。アプライアンスからログ・ファイルをダウンロードする際、CSV ファイルは trace.zip ファイルに含まれています。このファイルをダウンロードするには、「アプライアンス」 > 「トラブルシューティング」 > 「ロギング」 > 「ログ・ファイルのダウンロード」をクリックします。trace.zip ファイル内で、CSV ファイルは server_name/logs ディレクトリーにあります。ファイル名は jvmstats.log、mapstats.log、および ogstats.log です。
2. データの処理に使用するプログラム (スプレッドシートなど) に CSV ファイルをインポートします。

CSV ファイルの統計定義

サーバーにダウンロード可能な CSV ファイルには、ヒストリカル・チャートを作成するために使用可能な統計やその他の情報が含まれています。

親トピック: [モニター](#)

関連タスク:

[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)

関連資料:

[CSV ファイルの統計定義](#)

CSV ファイルの統計定義

サーバーにダウンロード可能な CSV ファイルには、ヒストリカル・チャートを作成するために使用可能な統計やその他の情報が含まれています。

Java 仮想マシン (JVM) の統計ログ

TimeStamp (列 1)

Java 仮想マシン (JVM) に対して取られた統計スナップショットの日時を示します。

ServerName (列 2)

JVM のサーバー名を指定します。

Hostname (列 3)

JVM のホスト名を指定します。

FreeMemory (列 4)

JVM で使用可能なバイト数を指定します。

MaxMemory (列 5)

JVM に割り振り可能な最大バイト数を指定します。

TotalMemory (列 6)

サーバー実行時における実メモリ使用量を表示します。

AvailProcs (列 7)

このカタログ・サービスおよびそのマップで使用可能なプロセッサの数を表示します。最高の安定度を実現するために、60% のプロセッサ負荷でサーバー、さらに 60% のヒープ負荷で JVM ヒープを稼働してください。スパイクでプロセッサ使用量を 80% から 90% の間に引き上げることができますが、通常はこのレベルより高いレベルでサーバーを稼働しないようにしてください。

マップの統計ログ

TimeStamp (列 1)

マップに対して取られた統計スナップショットの日時を指定します。

MapName (列 2)

マップの名前を指定します。

OgName (列 3)

このマップが属しているデータ・グリッドの名前を指定します。

PartitionId (列 4)

区画 ID を指定します。

MapSetName (列 5)

このマップが属しているマップ・セットの名前を指定します。

HitRate (列 6)

選択したマップのヒット・レート (ヒット率) を表示します。高ヒット・レートが望ましい状態です。ヒット・レートは、パーシスタント・ストアへのアクセスの回避にデータ・グリッドがどのくらい役立っているかを示します。

Count (列 7)

サーバーの始動以降におけるデータ・グリッド内の項目数を示します。例えば、100 の値は、その項目がサーバーの始動以降収集された 100 番目のサンプル項目であることを示します。

TotalGetCount (列 8)

データを取得するためにマップがパーシスタント・ストアにアクセスする必要があった回数の合計を表示します。

TotalHitCount (列 9)

要求データがマップ内で検出され、パーシスタント・ストアにアクセスする必要がなかった回数の合計を表示します。

StartTime (列 10)

最後のリセット呼び出しからカウンターが開始する時刻を示します。リセットは、サーバーが始動または再始動したときに行われます。

LastCount (列 11)

最後のデータ・サンプルが取られてから経過した時間を示します。

LastTotalGetCount (列 12)

現在のキャッシュからの取得操作の総数から、前の期間の取得操作の数を引いた数を表示します。

LastTotalHitCount (列 13)

現在のキャッシュからのヒット総数から、前の期間のヒット数を引いた数を示します。

UsedBytes (列 14)

このマップによるメモリー消費量を表示します。使用バイト統計は、単純なオブジェクトまたは COPY_TO_BYTES コピー・モードを使用している場合に限り正確です。

MinUsedBytes (列 15)

このカタログ・サービスおよびそのマップによるメモリー消費量の最低点を表示します。使用バイト統計は、単純なオブジェクトまたは COPY_TO_BYTES コピー・モードを使用している場合に限り正確です。

MaxUsedBytes (列 16)

このカタログ・サービスおよびそのマップによるメモリー消費量の最高点を表示します。使用バイト統計は、単純なオブジェクトまたは COPY_TO_BYTES コピー・モードを使用している場合に限り正確です。

LastUsedBytes (列 17)

現在の UsedBytes 値から以前の統計収集期間の UsedBytes 値を引いた数を示します。

SampleLen (列 18)

データのサンプルが取られた期間の長さをミリ秒単位で示します。

ObjectGrid 統計ログ

TimeStamp (列 1)

データ・グリッドに対して取られた統計スナップショットの日時を指定します。

OgName (列 2)

データ・グリッドの名前を指定します。

PartitionId (列 3)

区画 ID を指定します。

Count (列 4)

サーバーの始動以降に収集されたデータ・グリッド項目の数を示します。例えば、100 の値は、その項目がサーバーの始動以降収集された 100 番目のサンプル項目であることを示します。

Hostname (列 5)

ホスト名を指定します。

DomainName (列 6)

このデータ・グリッドが属しているカタログ・サービス・ドメインを指定します。

MaxTime (列 7)

このサーバーで最も時間がかかった トランザクションで費やした時間を表示します。

MinTime (列 8)

このサーバーで最も時間がかからなかった トランザクションで費やした時間を表示します。

MeanTime (列 9)

トランザクションにかかった平均時間を示します。

TotalTime (列 10)

このサーバーでトランザクションに費やした、このサーバーの初期設定時からの合計時間を表示します。

AvgTransTime (列 11)

このサーバーでトランザクションを完了するために必要な平均時間を表示します。

AvgThroughPut (列 12)

このサーバーの 1 秒当たりのトランザクションの平均数を表示します。

SumOfSquares (列 13)

トランザクション時間の二乗値の合計を示します。この値は、ある時点での平均からの偏差を測定します。

SampleLen (列 14)

データのサンプルが取られた期間の長さをミリ秒単位で示します。

LastDataSample (列 15)

最後のサンプルが取られてから経過した時間を示します。

LastTotalTime (列 16)

現在のデータ・サンプルの合計時間から前の合計時間を引いた値を示します。

StartTime (列 17)

データの最後のリセット以降に統計の収集が開始された時刻を示します。サーバーが再始動すると、データはリセットされます。

親トピック: [CSV ファイルによるモニター](#)

関連タスク:


[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)

[CSV ファイルによるモニター](#)

xsadmin ユーティリティーによるモニター

xsadmin ユーティリティーを使用して、WebSphere® DataPower® XC10 アプライアンス上で実行しているデータ・グリッドに関するテキスト情報をフォーマット設定し、表示できます。このサンプル・ユーティリティーは現在のデプロイメント・データの解析とディスクバリーの方法を提供するもので、カスタム・ユーティリティーの作成基盤として使用することができます。



始める前に

-  **2.5+** **xsadmin** ユーティリティーは現在非推奨になっています。代わりに、**xscmd** ユーティリティーを使用します。**xscmd** ユーティリティーは、環境のモニターおよび管理のための、サポートされるユーティリティーとして提供されています。詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

このタスクについて

xsadmin ユーティリティーは、Managed Bean (MBeans) の実装を使用します。[com.ibm.websphere.objectgrid.management](#) パッケージ内のインターフェースを使用して、このユーティリティーの機能を拡張できます。**xsadmin** アプリケーションのソース・コードは、スタンドアロン・インストールの場合は `wxs_client_home/samples/xsadmin.jar` ファイルで確認でき、WebSphere Application Server インストールの場合は `wxs_client_home/optionalLibraries/ObjectGrid/xsadmin.jar` ファイルで確認できます。

手順

1. アプライアンスのアクティブなトラストストアをクライアントにダウンロードします。WebSphere DataPower XC10 アプライアンスのユーザー・インターフェースで、「アプライアンス」 > 「設定」 > 「**Transport Layer Security**」 > 「アクティブなトラストストアのダウンロード」をクリックします。デフォルトのトラストストア・ファイルは、`xsatruststore.jks` ファイルです。このファイルのデフォルトのパスワードは、`xc10pass` です。
2. コマンド行で、`JAVA_HOME` 環境変数を設定します。
 -  `export JAVA_HOME=javaHome`
 -  `set JAVA_HOME=javaHome`
3. `bin` ディレクトリーに移動します。

```
cd wxs_client_home/bin
```

4. **xsadmin** ユーティリティーを実行します。アプライアンスに接続するには、ダウンロードしたトラストストアのセキュリティー引数、アプライアンスへのログインに使用するユーザー名とパスワード、およびアプライアンスのホスト名をコマンドを実行するたびに組み込む必要があります。

```
xsadmin.sh -trustPath xsatruststore.jks -trustType jks -ssl -trustPass xc10pass  
-username xcadmin -password xcadmin -ch myxc10.mycompany.com  
[additional_xsadmin_parameters]
```

構成ファイルを作成して、これらのパラメーターを保存することもできます。必要なパラメーターが組み込まれたプロパティー・ファイルの例を次に示します。

```
XSADMIN_TRUST_PATH=xsatruststore.jks  
XSADMIN_TRUST_TYPE=JKS  
XSADMIN_TRUST_PASS=xc10pass  
XSADMIN_USERNAME=xcadmin  
XSADMIN_PASSWORD=xcadmin
```

プロパティー・ファイルを指定して **xsadmin** ユーティリティーを実行するには、**-profile** 引数を使用してプロパティー・ファイルの場所を示します。

```
xsadmin.sh -profile myxc10.properties -ssl -ch myxc10.mycompany.com  
[additional_xsadmin_parameters]
```

[xsadmin ユーティリティー・リファレンス](#)

xsadmin ユーティリティーには、コマンド行引数を使用する方式、またはプロパティー・ファイルを使用する方式で、引数を渡すことができます。

[xsadmin ツールから xscmd ツールへのマイグレーション](#)


これまでのリリースでは、**xsadmin** ツールは環境の状態をモニターするサンプルのコマンド行ユーティリティーでした。**xscmd** ツールは、管理およびモニター用の、正式にサポートされるコマンド行ツールとして導入されました。これまで **xsadmin** ツールを使用していた場合は、新しい **xscmd** ツールにコマンドをマイグレーションすることを検討してください。

親トピック: [モニター](#)

xsadmin ユーティリティー・リファレンス

xsadmin ユーティリティーには、コマンド行引数を使用する方式、またはプロパティー・ファイルを使用する方式で、引数を渡すことができます。

xsadmin 引数

 **2.5+** 注: **xsadmin** ユーティリティーは現在非推奨になっています。代わりに、**xscmd** ユーティリティーを使用します。**xscmd** ユーティリティーは、環境のモニターおよび管理のための、サポートされるユーティリティーとして提供されています。詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

xsadmin ユーティリティー用の プロパティー・ファイルは、WebSphere® eXtreme Scale クライアント、バージョン 7.1 フィックス 1 以上で定義できます。プロパティー・ファイルを作成すると、ユーザー名などの使用頻度の高い引数の一部を保存できます。プロパティー・ファイルに追加できるプロパティーは、次の表のとおりです。プロパティー・ファイル内のプロパティーとそれに相当するコマンド行引数を 両方指定した場合、コマンド行引数値がプロパティー・ファイルの値に優先します。

表 1. **xsadmin** ユーティリティーの引数

コマンド行引数	プロパティー・ファイル内で相当するプロパティー名	説明と有効な値
-bp	n/a	リスナー・ポートを指示します。 デフォルト:2809
-ch	n/a	カタログ・サーバーの JMX ホスト名を指示します。 デフォルト:localhost
-clear	n/a	指定されたマップをクリアします。 使用できるフィルター: -fm
-containers	n/a	データ・グリッドおよびマップ・セットのそれぞれについて、コンテナー・サーバーのリストを表示します。 使用できるフィルター: -fnp
-continuous	n/a	データ・グリッドをモニターするためにマップ・サイズの結果を継続的に表示させたい場合は、このフラグを指定してください。 -mapsizes 引数を指定してこのコマンドを実行すると、20 秒ごとに マップ・サイズが表示されます。
-coregroups	n/a	カタログ・サーバーのすべてのコア・グループを表示します。この引数は、拡張診断に使用されます。
-dismissLink <catalog_service_domain>	n/a	このコマンド行引数は、WebSphere DataPower® XC10 アプライアンス構成には適用されません。
-dmgr	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
-empties	n/a	出力に空のコンテナーを表示する場合、このフラグを指定します。
-establishLink <foreign_domain_name> <host1:port1, host2:port2...>	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
-fc	n/a	このコンテナーのみにフィルタリングします。 WebSphere Application Server Network Deployment 環境でコンテナー・サーバーをフィルタリングする場合は、次の形式を使用します。

		<p><cell_name>/<node_name>/<serverName_containerSuffix></p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec</p>
-fh	n/a	<p>このホストのみにフィルタリングします。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec、-routetable</p>
-fm	n/a	<p>このマップのみにフィルタリングします。</p> <p>この形式を指定できる引数: -clear、-mapsizes</p>
-fnp	n/a	<p>プライマリ断片を持たないサーバーをフィルタリングします。</p> <p>この形式を指定できる引数: -containers</p>
-fp	n/a	<p>この区画のみにフィルタリングします。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec、-routetable</p>
-fs	n/a	<p>このサーバーのみにフィルタリングします。</p> <p>WebSphere Application Server Network Deployment 環境でアプリケーション・サーバーをフィルタリングする場合は、次の形式を使用します。</p> <p><cell_name>/<node_name>/<server_name></p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec</p>
-fst	n/a	<p>この断片タイプのみでフィルタリングします。プライマリ断片のみ場合は P を、非同期レプリカ断片のみ場合は A を、同期レプリカ断片のみ場合は S を指定します。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec</p>
-fz	n/a	<p>このゾーンのみにフィルタリングします。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec、-routetable</p>
-force	n/a	<p>いずれのプリエンティブなプロンプトも無効にして、コマンド内のアクションを強制します。この引数はバッチ・コマンドを実行するときに便利です。</p>
-g	n/a	<p>ObjectGrid 名を指定します。</p>
-getstatsspec	n/a	<p>現在の統計仕様を表示します。統計仕様は、-setstatsspec 引数で設定できます。</p> <p>使用できるフィルター: -fst -fc -fz -fs -fh -fp</p>
-getTraceSpec	n/a	<p>現在のトレース仕様を表示します。トレース仕様</p>

<code>gettracespec</code>	n/a	<p>構成内のトレースは、<code>-settracespec</code> 引数で設定できます。</p> <p>使用できるフィルター: <code>-fst -fc -fz -fs -fh -fp</code></p>
<code>-h</code>	n/a	xsadmin ユーティリティーのヘルプを表示します。ヘルプには引数のリストも含まれます。
<code>-hosts</code>	n/a	構成内のすべてのホストを表示します。
<code>-jmxUrl</code>	XSADMIN_JMX_URL	<p>JMX API コネクタ・サーバーのアドレスを次の形式で指定します: <code>service:jmx:protocol:sap</code>。 <code>protocol</code> および <code>sap</code> 変数の定義は、次のとおりです。</p> <p>protocol</p> <p>コネクタ・サーバーに接続するために使用するトランスポート・プロトコルを指定します。</p> <p>sap</p> <p>コネクタ・サーバーが検出されるアドレスを指定します。</p> <p>JMX サービス URL の形式の詳細については、Class JMXServiceURL (Java™ 2 Platform SE 5.0) を参照してください。</p>
<code>-l</code>	n/a	すべての既知のデータ・グリッドとマップ・セットを表示します。
<code>-m</code>	n/a	マップ・セットの名前を指定します。
<code>-mapsizes</code>	n/a	<p>断片間でキーの分布が一樣か確認するため、カタログ・サーバー上の各マップのサイズを表示します。</p> <p>使用できるフィルター: <code>-fm -fst -fc -fz -fs -fh -fp</code></p>
<code>-mbeanservers</code>	n/a	すべての MBean サーバー・エンドポイントのリストを表示します。
<code>-overridequorum</code>	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
<code>-password</code>	XSADMIN_PASSWORD	xsadmin ユーティリティーにログインするためのパスワードを指定します。パスワードを保護する必要がある場合、プロパティ・ファイル内にパスワードは指定しないでください。
<code>-p</code>	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
	n/a	<p>構成内の構成されている配置とランタイムの配置を表示します。出力の有効範囲はデータ・グリッドとマップ・セットの組み合わせにしたり、構成全体にしたりできます。</p> <ul style="list-style-type: none"> 構成全体: <pre>-placementStatus</pre> 特定のデータ・グリッドの場合: <pre>-placementStatus -g my_grid</pre> 特定のマップ・セットの場合: <pre>-placementStatus -m my_mapset</pre> 特定のデータ・グリッドとマップ・セットの場合: <pre>-placementStatus -g my_grid -m my_mapset</pre>

-primaries	n/a	プライマリー断片のリストを表示します。
-profile	n/a	xsadmin ユーティリティー用のプロパティ・ファイルへの完全修飾パスを指定します。
-quorumstatus	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
-releaseShard <container_server_name> <objectgrid_name> <map_set_name> > <partition_name>	n/a	-reserveShard 引数と一緒に使用されます。 -releaseShard 引数は、断片が予約されて配置された後に呼び出されます。 -releaseShard 引数は、ContainerMBean.release() メソッドを呼び出します。
-reserved	n/a	-reserveShard 引数によって予約された断片のみを表示するために -containers 引数と一緒に使用されます。
-reserveShard <container_server_name> <objectgrid_name> <map_set_name> > <partition_name>	n/a	プライマリー断片を指定されたコンテナ・サーバーに移動します。この引数によって、ContainerMBean.reserve() メソッドが呼び出されます。
-resumeBalancing <objectgrid_name> <map_set_name> >	n/a	要求のバランスを取るを試みます。指定された ObjectGrid とマップ・セットの将来の再バランシング試行を可能にします。
-revisions	n/a	カタログ・サービス・ドメインの改訂 ID を表示します。これには、特定の断片ごとの各データ・グリッド、区画番号、区画タイプ (基本またはレプリカ)、カタログ・サービス・ドメイン、存続期間 ID、およびデータ改訂の回数が含まれます。この引数を使用して、非同期レプリカまたはリンクされたドメインがキャッチアップされたかどうかを判別できます。この引数は、ObjectGridMBean.getKnownRevisions() メソッドを呼び出します。 使用できるフィルター: -fst -fc -fz -fs -fh -fp
-routetable	n/a	クライアント・サーバーの観点から、データ・グリッドの現在の状態を表示します。経路テーブルは、ObjectGrid クライアント・サーバーがデータ・グリッドとの通信に使用する情報です。接続問題または TargetNotAvailable 例外を特定する場合に、診断援助として経路テーブルを使用します。 必須引数: スタンドアロン環境では、カタログ・サーバー・ホストのブートストラップ・リスナー・ポートおよび JMX ポートにデフォルト値を使用していない場合、この引数と一緒に -bp パラメーターおよび -p パラメーターを指定しなければなりません。 使用できるフィルター: -fz -fh -fp
-settracespec <trace_string> >	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。アプライアンスのトレースの構成について詳しくは、 WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作

		<p>を参照してください。</p> <p>使用できるフィルター: -fst -fc -fz -fs -fh -fp</p>
-swapShardWithPrimary <container_server_name> <objectgrid_name> <map_set_name> > <partition_name>	n/a	指定されたコンテナ・サーバーからの指定されたレプリカ断片を、プライマリ断片と交換します。このコマンドを実行することで、必要なときに手動でプライマリ断片をバランシングできます。
-setstatsspec <stats_spec>	n/a	統計の収集を有効にします。この引数は、DynamicServerMBean.setStatsSpec メソッドと DynamicServerMBean.getStatsSpec メソッドを呼び出します。 使用できるフィルター: -fm -fst -fc -fz -fs -fh -fp
-suspendBalancing <objectgrid_name> <map_set_name> >	n/a	指定された ObjectGrid およびマップ・セットの将来のバランシング試行をさせません。
-ssl	n/a	Secure Sockets Layer (SSL) が有効なことを示します。
-teardown	n/a	<p>使用できるフィルター: -fst -fc -fz -fs -fh -fp</p> <p>サーバーのリストを指定する場合の形式:</p> <pre>server_name_1,server_name_2 ...</pre> <p>特定ゾーン内のすべてのサーバーを停止する場合、-fz 引数を組み込みます。</p> <pre>-fz <zone_name></pre> <p>特定ホスト上のすべてのサーバーを停止する場合、-fh 引数を組み込みます。</p> <pre>-fh <host_name></pre> <p>このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。</p>
	n/a	デプロイメント XML ファイルに構成されている numInitialContainers の値を 無視して、断片配置の実行を強制します。サーバーのメンテナンスを実行中、この引数を使用すると、 numInitialContainers の値が構成値より小さい場合でも断片配置の実行を続行できるようになります。
-trustPass	XSADMIN_TRUST_PASS	指定したトラストストアのパスワードを指定します。
-trustPath	XSADMIN_TRUST_PATH	トラストストア・ファイルへのパスを指定します。 例: etc/test/security/server.public
-trustType	XSADMIN_TRUST_TYPE	トラストストアのタイプを指定します。

	-type	有効値: JKS、JCEK、PKCS12 など。
-unassigned	n/a	データ・グリッド上に配置できない断片のリストを表示します。配置サービスに配置を妨げる制約があると、断片は配置できません。
-username	XSADMIN_USERNAME	xsadmin ユーティリティーにログインするためのユーザー名を指定します。
-v	n/a	詳細コマンド行アクションを使用可能にします。環境変数、プロパティ・ファイル、またはその両方を使用して特定のコマンド行引数を指定し、それらの値を表示する必要がある場合、このフラグを使用します。
-xml	n/a	PlacementServiceMBean.listObjectGridPlacement() メソッドからのフィルタリングされていない出力を表示します。他の xsadmin 引数は、このメソッドの出力をフィルタリングし、データをさらに消費可能な形式に編成します。

親トピック: [xsadmin ユーティリティーによるモニター](#)

xsadmin ツールから xscmd ツールへのマイグレーション

これまでのリリースでは、**xsadmin** ツールは環境の状態をモニターするサンプルのコマンド行ユーティリティでした。 **xscmd** ツールは、管理およびモニター用の、正式にサポートされるコマンド行ツールとして導入されました。これまで **xsadmin** ツールを使用していた場合は、新しい **xscmd** ツールにコマンドをマイグレーションすることを検討してください。

xsadmin および xscmd コマンド同等


 **2.5+** **重要:** **xsadmin** ユーティリティは現在非推奨になっています。代わりに、**xscmd** ユーティリティを使用します。**xscmd** ユーティリティは、環境のモニターおよび管理のための、サポートされるユーティリティとして提供されています。詳しくは、[xscmd ユーティリティによる管理](#)を参照してください。

表 1. **xsadmin** ユーティリティの引数と、**xscmd** 同等コマンド。一部の **xscmd** コマンドは、短い形式と長い形式を持ちます。短い形式のダッシュ (-) を持ち、長い形式のコマンドは、2 つのダッシュ (--) を持ちます。どちらの形式も区別なく使用できます。

xsadmin コマンド行引数	xscmd 同等コマンド	xscmd コマンド・パラメーター
-bp	<ul style="list-style-type: none"> • -cep <i>hostname:listener_port</i> • --catalogEndpoint <i>hostname:listener_port</i> 	n/a
-ch	<ul style="list-style-type: none"> • -cep <i>hostname:listener_port</i> • --catalogEndpoint <i>hostname:listener_port</i> 	n/a
-clear	-c clearGrid	-g, -ms, -v, -m, (-cep)
-containers	<ul style="list-style-type: none"> • -c showPlacement -container <i>containerName</i> • -c showPlacement -server <i>serverName</i> 	-e, -i, , -st, -snp, -ct, -s, -p, -hf,
-continuous	n/a	n/a
-coregroups	<ul style="list-style-type: none"> • -c listCoreGroupMembers -cg <i>core_group</i> 	n/a
-dismissLink <i><catalog_service_domain></i>	-c dismissLink	<ul style="list-style-type: none"> • -fd <i><foreignCatalogServiceDomain></i> • --foreignCatalogServiceDomain <i><foreignCatalogServiceDomain></i>
-dmgr	n/a - この引数は、 xscmd で自動的に判別される	n/a
-empties	新規コマンドに固有の引数	n/a
-establishLink <i><foreign_domain_name></i> <i><host1:port1,host2:port2...></i>	-c establishLink	<ul style="list-style-type: none"> • -fd <i><foreignCatalogServiceDomain></i> <i><host1:port1,host2:port2...></i> • --foreignCatalogServiceDomain <i><foreignCatalogServiceDomain></i> -fd <i><host1:port1,host2:port2...></i>
-fc	<ul style="list-style-type: none"> • -ct • --container 	n/a
-fh	<ul style="list-style-type: none"> • -hf • --hostFilter 	n/a
-fm	<ul style="list-style-type: none"> • -m • --map 	n/a
-fnp	<ul style="list-style-type: none"> • -snp • --serversWithNoPrimaries 	n/a
-fp	<ul style="list-style-type: none"> • -p • --partitionId 	n/a
-fs	<ul style="list-style-type: none"> • -s • --server 	n/a

-fst	<ul style="list-style-type: none"> • -st <shard_type> • --shardType <shard_type> 断片値: P=primary A=asyncReplica S=syncReplica	n/a
-fz	<ul style="list-style-type: none"> • -z • --zone 	n/a
-force	新規コマンドに固有の引数	
-g	<ul style="list-style-type: none"> • -g • --objectGrid 	n/a
-getstatsspec	-c getStatsSpec	n/a
-getTraceSpec	-c getTraceSpec	n/a
-h	特定のコマンド名を付けても付けなくても help を実行できる <ul style="list-style-type: none"> • -h • --help • -h <command_name> • --help <command_name> 	n/a
-hosts	-c listHosts	-g, -ms, -st, -c, -s, -hf, -z
-jmxUrl	<ul style="list-style-type: none"> • -cep hostname:listener_port • --catalogEndpoint hostname:listener_port 	n/a
-l	-c listObjectGridNames	n/a
-m	<ul style="list-style-type: none"> • -ms • --mapSet 	n/a
-mapsizes	-c showMapSizes	-g, -ms, -i, [-ct, -z, -s, -hf, sht [P
-mbeanservers	-c listAllJMXAddresses	n/a
-overridequorum	-c overrideQuorum	n/a
-password	<ul style="list-style-type: none"> • -pwd • --password 	n/a
-p	<ul style="list-style-type: none"> • -cep hostname:listener_port • --catalogEndpoint hostname:listener_port 	n/a
-placementStatus	-c placementServiceStatus	-g, -ms
-primaries	-c showPlacement -sf P	-e, -i, , -st, -snp, -ct, -s, -p, -hf,
-profile	現行セキュリティー設定をセキュリティー・ プロファイルとして保存する場合: <ul style="list-style-type: none"> • -ssp profile_name • --saveSecProfile profile_name 指定されたセキュリティー・プロファイル を使用する場合: <ul style="list-style-type: none"> • -sp profile_name • --securityProfile profile_name 	
-quorumstatus	-c showQuorumStatus	n/a
-releaseShard	-c releaseShard	-c, -g, -ms, -p
重要: このコマンド行引数は、 WebSphere® DataPower® XC10 アプ ライアンス構成には適用され ません。	重要: このコマンド行引数は、 WebSphere DataPower XC10 アプ ライアンス構成には適用され ません。	

-reserved	<ul style="list-style-type: none"> • -sf R • --shardFilter R 	n/a
-reserveShard 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c reserveShard 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c, -g, -ms, -p
-resumeBalancing <objectgrid_name> <map_set_name> 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c resumeBalancing 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-g, -ms
-revisions	-c revisions	-s, -p, -g, -m
-routetable	-c routetable	-z, -hf, -p, -g, -ms
-settracespec <trace_string>	-c setTraceSpec	-spec <trace_string>
-swapShardWithPrimary 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c swapShardWithPrimary 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c -g, -ms, -p
-setstatsspec <stats_spec>	-c setStatsSpec	-spec <stats_spec>
-suspendBalancing <objectgrid_name> <map_set_name> 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c suspendBalancing 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-g, -ms
-ssl	<ul style="list-style-type: none"> • -ssl • --enableSSL 	n/a
-teardown 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c teardown 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-f, , -st, -snp, -c, -s, -p, -hf, -z,
-triggerPlacement 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c triggerPlacement 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-g, -ms
-trustPass	<ul style="list-style-type: none"> • -tsp • --trustStorePassword 	n/a
-trustPath	<ul style="list-style-type: none"> • -ts 	n/a

	<ul style="list-style-type: none"> • --trustStore 	
-trustType	<ul style="list-style-type: none"> • -tst • --trustStoreType 	n/a
-unassigned	-c showPlacement -sf U	-e, -i, , -st, -snp, -ct, -s, -p, -hf,
-username	<ul style="list-style-type: none"> • -user • --username 	n/a
-v	<ul style="list-style-type: none"> • -v • --verbose 	n/a
-xml	-c showPlacement	n/a

親トピック: [xsadmin ユーティリティによるモニター](#)

環境のヘルスのモニター

2.5+ メッセージ・センターでは、ログおよび初期障害データ・キャプチャー機能 (FFDC) メッセージのイベント通知の集約ビューが示されます。これらのイベント通知は、Web コンソールのメッセージ・センター、**xscmd** ユーティリティ、ヘルス・ログ・ファイル、HTTP コマンド・インターフェース、または MBeans を使用したプログラマチック方式で表示することができます。

2.5+ [ヘルス・モニターの概要](#)

メッセージ・センターは、集合内のすべてのコンテナ・サーバーおよびカタログ・サーバーからヘルス状況イベントをリアルタイムで収集します。メッセージ・センターが構成されていると、さまざまなサーバーで発生している現行クリティカル・イベントの概要を、各サーバーのログを収集せずに表示することができます。

2.5+ [メッセージ・センターでのヘルス・イベント通知の表示](#)

Web コンソールでメッセージ・センターを使用して、データ・グリッドと 集合全体のリアルタイム・ヘルスを評価できます。メッセージ・センターに表示されるイベントは、最も重大な問題を表示するようにフィルターが掛けられた、イベントのサブセットです。

2.5+ [xscmd ユーティリティを使用したヘルス通知の表示](#)

xscmd ユーティリティを使用して、現在のイベント通知を表示したり、イベント通知履歴を表示したり、メッセージ・センターから通知フィルターを設定したりすることができます。

次のタスク

アプライアンスのハードウェアおよびソフトウェアのヘルス状況全体の要約を表示することができます。以下のいずれかのオプションを使用します。

- ヘルス・ログ・ファイルを表示してダウンロードする。詳しくは、[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)を参照してください。
- HTTP コマンド・インターフェースの **GetHealthStatus** コマンドを使用して、アプライアンスのハードウェアおよびソフトウェアのヘルス状況全体を表示する。詳しくは、[HTTP コマンド・インターフェースでの管理](#)および [HTTP コマンド・インターフェースのリファレンス](#)を参照してください。

親トピック: [モニター](#)

ヘルス・モニターの概要

2.5+ メッセージ・センターは、集合内のすべてのコンテナ・サーバーおよびカタログ・サーバーからヘルス状況イベントをリアルタイムで収集します。メッセージ・センターが構成されていると、さまざまなサーバーで発生している現行クリティカル・イベントの概要を、各サーバーのログを収集せずに表示することができます。

メッセージ・センターの実装

メッセージ・センターはデフォルトで使用可能です。ユーザー・インターフェースでメッセージ・センターを使用不可にすることができます。

データ・グリッド・デプロイメントは多数の分散サーバー・プロセスを伴うことがあります。問題が発生した場合は、影響を受けたコンテナ・サーバーの現行ログ・ファイルを開いて、その問題を詳しく分析することができます。

メッセージ・センターは以下のコンポーネントから成ります。

イベント集約

カタログ・サーバーに対するヘルス・モニターを構成すると、カタログ・サービス・ドメイン全体のヘルスに影響を及ぼしている集約イベントを受け取ります。この枠組みには、次に示すタイプのイベントの原因と重大度の指示が含まれます。

- すべての FFDC イベント
- すべての WARNING または SEVERE ログ・エントリ
- フィルタリングされたすべてのログ・エントリのリスト (INFO、WARNING、および SEVERE ログ・エントリを含む)
- サーバー始動操作およびサーバー停止操作
- データ・グリッドがほぼ満杯である
- クォーラムの損失または回復
- 使用可能に設定された SNMP トラップがトリガーされる
- 複製が 15 分間遅れる

Web コンソール内のメッセージ・センター

Web コンソール内のメッセージ・センターは集約されたイベント・レコードを表示します。これらのイベントには、最近発生したイベントと、コンソールが開かれた後から発生したイベントのリアルタイム更新通知の両方が含まれます。

xscmd ユーティリティーでのイベント

xscmd ユーティリティーを使用して最近発生したイベントのリストを表示することもできます。イベントの発生について、イベント・レコードをリダイレクトして自動スクリプティング・ユーティリティーを作成することができます。

他のモニター・ソフトウェアとの統合のための MBean

有効な管理 MBean を使用してメッセージ・センターを他の Java Management Extensions (JMX) モニター・ソフトウェアにプラグインすることもできます。この MBean の資料は API 資料に含まれています。

2.5+

ヘルスの要約

ヘルス・ログおよび HTTP コマンド・インターフェースの **GetHealthStatus** コマンドを使用して、アプライアンスのソフトウェアおよびハードウェアのヘルス状況の要約を取得することができます。この情報には、データ・グリッドの配置と複製の状況、ハードウェア警告、システムとネットワークのヘルス状況、およびその他の状況メッセージなどがあります。

ログ・アナライザーと対比したメッセージ・センター

ログ・アナライザーは、一連のログ・メッセージを分析するもう 1 つのツールです。このツールを使用するには、環境内のさまざまなサーバーからログを手動で収集する必要があります。その後、このツールを実行して問題条件のレポートを作成することができます。メッセージ・センターで表示できるメッセージ数 (1000 メッセージ) を超える一連のメッセージを分析する必要があるときには、ログ・アナライザーを使用してログの事後分析を実施してください。発生している問題を素早く特定するには、メッセージ・センターを使用してデータ・グリッドのヘルスをリアルタイムでモニターしてください。そうすれば、関連するコンテナ・サーバーのログ・ファイルを検討することもできれば、ログ・アナライザーを使用して問題を詳しく調査することもできます。

ヘルス・モニターの構成とアーキテクチャー

メッセージ・センターはデフォルトで使用可能です。ユーザー・インターフェースでメッセージ・センターを使用不可にすることができます。メッセージ・センターが使用可能に設定されると、集合内のカタログ・サーバーがヘルス・モニターのためのハブとしてアクティブ化されます。一般に、メッセージ・センター内のメッセージの送信元は、最初に集合を作成したときのアプライアンス上にあるカタログ・サーバーです。各ハブには、独自のサブスクリプションと個別のイベント・ヒストリーがあります。ヒストリー内の各イベントにはシーケンス番号が付いています。個々のカタログ・サーバーにあるイベント・ヒストリーは同期されず、それぞれ異なります。カタログ・サーバーは、他のカタログ

グ・サーバーからのログ・イベントおよび FFDC イベントにサブスクライブすることができます。

親トピック: **2.5+** [環境のヘルスのモニター](#)

メッセージ・センターでのヘルス・イベント通知の表示

Web コンソールでメッセージ・センターを使用して、データ・グリッドと 集合全体のリアルタイム・ヘルスを評価できます。メッセージ・センターに表示されるイベントは、最も重大な問題を表示するようにフィルターが掛けられた、イベントのサブセットです。

手順

- Web コンソールで、重大エラー、FFDC メッセージ、およびサーバーの始動と停止のイベントの通知を表示します。これらの通知は、Web コンソールで任意のページにログインすると、自動的に表示されます。
- メッセージ・センターでメッセージを表示します。Web コンソールで、「モニター」 > 「メッセージ・センター」をクリックします。メッセージ・センターには、カタログ・サーバー・メッセージ・ハブを介して送信された、最新の 1000 件の重大メッセージが表示されます。

カタログ・サーバー・メッセージ・ハブは、1000 件のメッセージが表示されるように、表示されるメッセージをフィルターに掛けます。そのため、メッセージ・センターには、カタログ・サーバーで発生している全重大イベントのサブセットのみが含まれています。ページを開いた状態のときに新しいメッセージが使用可能になると、ページを最新表示するオプションを備えた情報メッセージが、ページの上部に表示されます。

- メッセージ・センターに表示されるメッセージをフィルターに掛けます。最大で 3 つのフィルター・ルールを追加できます。ルールは、列、条件、および値から成ります。
 1. Web コンソールで、「モニター」 > 「メッセージ・センター」をクリックします。
 2. フィルター・ボタン (🔍) をクリックします。
 3. ルールを追加します。
 - a. 追加ボタン (+) をクリックします。
 - b. メッセージ・センターで、フィルターに掛ける列を以下のものから選択します。

ID

メッセージ・センターで生成されたイベント ID。

タイプ

メッセージの重大度を示す、メッセージのタイプ。有効な値は、「重大」、「警告」、「エラー」、および「通知」です。

日付

メッセージが生成された日時。

ソース

メッセージが発信されたサーバー。

メッセージ

メッセージ・イベントのメッセージ・テキスト。

- c. フィルターを適用する条件を選択します。日付およびタイプ以外の、ほとんどの列で有効な条件を以下のリストに示します。
 - 次を含む (Contains)
 - 次と等しい (Is)
 - 次で始まる (Starts with)
 - 次で終わる (Ends with)
 - d. 列をフィルターに掛ける値を入力します。

例: server1 のメッセージのみを表示するには、「ソース」列を選択します。次に、「次と等しい (Is)」条件を選択します。値として、「server1」を入力します。
4. 定義したルールの一部またはすべてと突き合わせるように選択できます。
 5. 「フィルター」をクリックして、構成されているフィルターをメッセージ・センターの出力に適用します。
- **2.5+** メッセージ・センターを使用不可に設定します。メッセージ・センターを使用不可に設定すると、新しいメッセージのポーリングが停止されます。「メッセージ・センターおよびトースター・アラートを使用不可に設定」をクリックします。メッセージ・センターを使用不可に設定した後、設定を適用する必要があります。

次のタスク

ログ・ファイルについて詳細な分析を実行できます。詳しくは、[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)を参照してください。

親トピック: **2.5+** [環境のヘルスのモニター](#)

xscmd ユーティリティーを使用したヘルス通知の表示

2.5+ xscmd ユーティリティーを使用して、現在のイベント通知を表示したり、イベント通知履歴を表示したり、メッセージ・センターから通知フィルターを設定したりすることができます。

始める前に

- xscmd ユーティリティーを開始し、カタログ・サービス・ドメインに接続します。詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

手順

- xscmd ユーティリティーを使用して、イベント通知履歴を表示します。出力が表形式で表示されます。

```
xscmd -c showNotificationHistory -cep hostname:port(,hostname:port)
```

- 新規通知を listen します。

```
xscmd -c listenForNotifications -cep hostname:port(,hostname:port)
```

出力はロー形式であり、コマンドを停止するまで出され続けます。追加スクリプトを作成して、出力を構文解析できます。

- 今後作成されるすべてのログ・エントリ (INFO、WARNING、SEVERE ログ・エントリなど) に対してフィルタリングを使用可能に設定します。。デフォルトでは、メッセージ・センターとコマンドでは、WARNING エラー、SEVERE エラー、およびイベントのみが表示されます。環境内のすべてのサーバー、または単一のサーバーに対してフィルターを設定できます。この設定は今後の結果にのみ影響します。

```
xscmd -c setNotificationFilter -fs <regular expression> [-server <servername>]
```

- 環境内のすべてのサーバーまたは単一のサーバーの現在の通知フィルターを表示します。

```
xscmd -c getNotificationFilter [-s servername]
```

親トピック: **2.5+** [環境のヘルスのモニター](#)

Simple Network Monitoring Protocol (SNMP) によるモニター

Simple Network Monitoring Protocol (SNMP) サポートにより、IBM® WebSphere® DataPower® XC10 アプライアンス の状況を、データ・センターのシステムのより大きなグループの一部として、モニターできます。SNMP モニターは、システム問題を素早く通知し、それらに迅速に応答する機能を強化します。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成しなくても以下の手順を実行できますが、SNMP モニターを使用できるのは、クライアントを構成した場合のみです。WebSphere DataPower XC10 アプライアンス は、SNMP バージョン v2c をサポートします。

このタスクについて

以下の手順に従って、IBM WebSphere DataPower XC10 アプライアンスで SNMP モニターを使用可能にします。SNMP 設定はアプライアンスに固有であり、集合内の他の アプライアンスには伝搬されません。

要確認: 1 つのアプライアンスに関する SNMP 統計が示されるため、報告される値は、ユーザー・インターフェースのモニター・パネルで報告される値と異なります。ユーザー・インターフェースでは、集合に関するデータが報告されます。例えば、SNMP gridStats 表、mapStats 表、または jvmStats 表の値は、ユーザー・インターフェースで報告される合計値の一部を報告します。

1. [アプライアンスでの SNMP モニターの使用可能化](#)

IBM WebSphere DataPower XC10 アプライアンス では、Simple Network Monitoring Protocol (SNMP) モニターを使用可能にすることができます。SNMP クライアントが使用できる SNMP データを指定するには、アプライアンスで提供される管理情報ベース (MIB) をダウンロードします。

2. [SNMP コミュニティーの構成](#)

1 つ以上の Simple Network Monitoring Protocol (SNMP) コミュニティーを作成することで、アプライアンス上の SNMP データへのアクセスを定義できます。モニターが使用可能な場合は、SNMP コミュニティーが必要です。

3. [SNMP トラップ・サブスクリプションの構成](#)

トラップ・サブスクリプションは、アプライアンスに組み込まれた Simple Network Monitoring Protocol (SNMP) エージェントとの通信に使用される SNMP クライアントです。

4. [SNMP トラップ・サブスクリプションの管理](#)

Simple Network Monitoring Protocol (SNMP) トラップは、アプライアンスに組み込まれた SNMP エージェントによって開始されるイベント通知または状態通知です。SNMP トラップ・サブスクリプションを使用して、エージェントが SNMP クライアントに伝達する SNMP トラップを定義します。WebSphere DataPower XC10 アプライアンス は、SNMP バージョン v2c をサポートします。

親トピック: [モニター](#)

アプライアンスでの **SNMP** モニターの使用可能化

IBM® WebSphere® DataPower® XC10 アプライアンスでは、Simple Network Monitoring Protocol (SNMP) モニターを使用可能にすることができます。SNMP クライアントが使用できる SNMP データを指定するには、アプライアンスで提供される管理情報ベース (MIB) をダウンロードします。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成しなくても以下の手順を実行できますが、SNMP モニターを使用できるのは、クライアントを構成した場合のみです。WebSphere DataPower XC10 アプライアンスは、SNMP バージョン v2c をサポートします。

このタスクについて

デフォルトでは、IBM WebSphere DataPower XC10 アプライアンスでは、SNMP モニターが使用可能になっていません。SNMP 設定はアプライアンスに固有であり、集合内の他のアプライアンスには伝搬されません。

手順

1. 「モニター」パネルにナビゲートします。IBM WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースで、「アプライアンス」 > 「**SNMP 設定**」をクリックします。
2. SNMP モニターを有効にするには、チェック・ボックスにチェック・マークを付けます。SNMP モニターを無効にする場合は、チェック・ボックスのチェックを外します。
3. アプライアンスで使用可能な管理情報ベース (MIB) ファイルをダウンロードします。

エンタープライズ MIB ファイルは、組み込み SNMP エージェントで 使用可能な機能とデータを記述して、クライアントが適切にこれらの機能とデータにアクセスできるようにします。クライアントは、SNMP GET、GET-NEXT、および GET-BULK コマンドを発行できます。MIB ファイルをダウンロードし、クライアントにインポートすると、基本 MIB-II データ定義以外のデータにアクセスできます。アプライアンスから MIB をダウンロードするには、エンタープライズ **MIB** を展開し、各 MIB の名前をクリックします。

統計 MIB

統計 MIB には、ユーザー・インターフェースのモニター機能を使用して確認できる統計と類似した情報が含まれます。この MIB には Java™ 仮想マシン (JVM) の統計も含まれます。

HW 状況 MIB

HW 状況 MIB には、温度や日時などのハードウェアの状態に関する情報が含まれます。

HW 通知 MIB

HW 通知 MIB は、SNMP TRAP コマンドで使用可能な情報を定義するために使用されます。

タスクの結果

上記手順を終えると、アプライアンスで SNMP モニターが使用可能になり、SNMP クライアント用の MIB データがダウンロードされます。

次のタスク

SNMP クライアントを使用して MIB データを表示します。

要確認: 1 つのアプライアンスに関する SNMP 統計が示されるため、報告される値は、ユーザー・インターフェースのモニター・パネルで報告される値と異なります。ユーザー・インターフェースでは、集合に関するデータが報告されます。例えば、SNMP 統計の gridStats 表、mapStats 表、または jvmStats 表の値は、ユーザー・インターフェースで報告される合計値の一部です。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

次のトピック: [SNMP コミュニティーの構成](#)

SNMP コミュニティーの構成

1 つ以上の Simple Network Monitoring Protocol (SNMP) コミュニティーを作成することで、アプライアンス上の SNMP データへのアクセスを定義できます。モニターが使用可能な場合は、SNMP コミュニティーが必要です。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成しなくても以下の手順は実行できますが、SNMP モニターを使用できるのは、クライアントを構成した場合のみです。WebSphere® DataPower® XC10 アプライアンスは SNMP バージョン v2c をサポートします。

このタスクについて

アプライアンスに組み込まれた SNMP エージェントを認証し、SNMP データにアクセスするには、コミュニティが必要です。アプライアンスに組み込まれた SNMP エージェントが要求されたデータを返すには、エージェントを認証するための定義済みのコミュニティ名をクライアントが指定する必要があります。コミュニティが指定されていない場合、SNMP エージェントはクライアントからの要求を無視します。SNMP コミュニティーはアプライアンスに固有であり、集合内の他のアプライアンスには伝搬されません。

手順

1. 「モニター」パネルにナビゲートします。IBM® WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」>「**SNMP モニター**」にナビゲートします。
2. 「**SNMP コミュニティー**」を展開します。
3. 「**コミュニティの作成**」をクリックします。
4. 作成する SNMP コミュニティーを記述するためのフォームに入力します。

名前

このフィールドには、SNMP コミュニティーの説明に使用する名前を指定します。

ホスト制限

このフィールドには、通信で使用する IP アドレスを IPv4 形式および IPv6 形式で指定します。また、このフィールドはホストに解決されるホスト名を受け入れます。クラスレス・ドメイン間ルーティング (CIDR) IP アドレスまたはホスト名を使用して、指定したサブネットにさらにアクセスを制限することができます。ホスト制限が指定されている場合、他の IP アドレスまたはサブネットとの通信はすべて拒否されます。このフィールドを空にすると、すべての IP アドレスとの通信が許可されます。

5. SNMP コミュニティーを除去するには、除去アイコン (✖) をクリックします。既存のコミュニティは変更できません。コミュニティを変更する必要がある場合は、削除してから再作成する必要があります。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

前のトピック: [アプライアンスでの SNMP モニターの使用可能化](#)

次のトピック: [SNMP トラップ・サブスクリバターの構成](#)

SNMP トラップ・サブスクリバターの構成

トラップ・サブスクリバターは、アプライアンスに組み込まれた Simple Network Monitoring Protocol (SNMP) エージェントとの通信に使用される SNMP クライアントです。

始める前に

SNMP トラップ・サブスクリバターは、既存の SNMP クライアントである必要があります。IBM® WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース でトラップ・サブスクリバターを作成する前に、トラップを受信するように SNMP クライアントを構成します。

このタスクについて

サブスクリブした SNMP クライアントに、アプライアンスに関する通知を送信するには SNMP トラップ・サブスクリバターが必要です。トラップ・サブスクリバターの作成時に、SNMP トラップを受信する SNMP クライアントに関する情報を指定する必要があります。トラップは、IBM WebSphere DataPower XC10 アプライアンス 管理者が注意を払う必要のある状況を記述します。SNMP エージェントは、サブスクリバターとして定義されていないクライアントに SNMP トラップを送信しません。

SNMP トラップ・サブスクリバターはすべて、データ・グリッドのストレージ制限とアプライアンスのストレージ制限に関するトラップを自動受信します。設定可能な具体的なトラップについては、[SNMP トラップ・リファレンス](#) を参照してください。

手順

1. 「SNMP 設定」パネルにナビゲートします。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースで、「アプライアンス」 > 「**SNMP 設定**」をクリックします。
2. 「トラップ・サブスクリバター」を展開します。
3. 「トラップ・サブスクリバターの作成」をクリックします。
4. 作成する SNMP トラップ・サブスクリバターを記述するためのフォームに入力します。

ホスト

SNMP クライアントがトラップ情報を listen する対象の IP アドレスを指定します。

クライアント・ポート番号

トラップ情報を listen している SNMP クライアントの所属元のポートを指定します。

コミュニティ

クライアントがメンバーとして所属している SNMP コミュニティを指定します。

5. 除去アイコン (✖) をクリックして、SNMP トラップ・サブスクリバターを削除します。既存のサブスクリバターは変更できません。サブスクリバターを変更する必要がある場合は、削除してから再作成する必要があります。

次のタスク

アプライアンスにアクセスできる少なくとも 1 つの SNMP トラップ・サブスクリバターを指定すると、SNMP エージェントに伝達させるトラップを構成できます。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

前のトピック: [SNMP コミュニティの構成](#)

次のトピック: [SNMP トラップ・サブスクリプションの管理](#)

SNMP トラップ・サブスクリプションの管理

Simple Network Monitoring Protocol (SNMP) トラップは、アプライアンスに組み込まれた SNMP エージェントによって開始されるイベント通知または状態通知です。SNMP トラップ・サブスクリプションを使用して、エージェントが SNMP クライアントに伝達する SNMP トラップを定義します。WebSphere® DataPower® XC10 アプライアンスは、SNMP バージョン v2c をサポートします。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成せずに以下の手順を実行できますが、有効な SNMP クライアントが構成されていない場合は、SNMP トラップを受信できません。

このタスクについて

以下の手順を使用して、通知されるトラップを具体的に設定します。

手順

1. 「SNMP 設定」パネルにナビゲートします。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェイス上部のメニュー・バーで、「アプライアンス」 > 「**SNMP 設定**」 > 「トラップ・サブスクリプション」にナビゲートします。
2. 個別のトラップを選択またはクリアします。個々のトラップを選択およびクリアすることで、SNMP トラップを選択できます。SNMP トラップ・サブスクリイバーはすべて、データ・グリッドのストレージ制限とアプライアンスのストレージ制限に関するトラップを自動受信します。設定可能な具体的なトラップについては、[SNMP トラップ・リファレンス](#)を参照してください。

タスクの結果

上記手順を終えると、サブスクライブした SNMP クライアントに報告される SNMP トラップの集合が定義されます。

[SNMP トラップ・リファレンス](#)

以下のイベントは、IBM WebSphere DataPower XC10 アプライアンスの Simple Network Management Protocol (SNMP) エージェントによってトラップされ、サブスクライブした SNMP クライアントに伝達されます。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

前のトピック: [SNMP トラップ・サブスクリイバーの構成](#)

SNMP トラップ・リファレンス

以下のイベントは、IBM® WebSphere® DataPower® XC10 アプライアンスの Simple Network Management Protocol (SNMP) エージェントによってトラップされ、サブスクライブした SNMP クライアントに伝達されます。

IBM WebSphere DataPower XC10 アプライアンスのモニターに使用可能な SNMP トラップ

表 1. IBM WebSphere DataPower XC10 アプライアンス の SNMP トラップ。

次のトラップはデフォルトで構成されています。これらのトラップは使用不可にできません。

説明
この通知は、特定の XC10 データ・グリッドがその容量の 60 パーセントに到達したことを示しています。
この通知は、指定された XC10 データ・グリッドがその容量の 60 パーセントを下回る状態に戻ったことを示しています。
この通知は、特定の XC10 データ・グリッドがその容量の 80 パーセントに到達したことを示しています。
この通知は、指定された XC10 データ・グリッドがその容量の 80 パーセントを下回る状態に戻ったことを示しています。
この通知は、特定の XC10 データ・グリッドがその容量に到達したことを示しています。
この通知は、指定された XC10 データ・グリッドがその容量を下回る状態に戻ったことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 80 パーセントに到達したことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 80 パーセントを下回る状態に戻ったことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 90 パーセントに到達したことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 90 パーセントを下回る状態に戻ったことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量に到達したことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量を下回る状態に戻ったことを示しています。

表 2. アプライアンスの SNMP トラップ。

以下のトラップは、デフォルトで使用可能になっています。ユーザー・インターフェースで、これらのトラップを使用不可にすることができます。

デフォルト	説明
Y	イーサネット・インターフェースは、非アクティブ状態に遷移しています。
Y	イーサネット・インターフェースは、アクティブ状態に遷移しました。
Y	SNMP エージェントが、潜在的な構成変更で再初期設定されました。
Y	SNMP エージェントがシャットダウンしています。
Y	ファン・センサーが元の正常な状態に遷移しました。
Y	ファン・センサーが望ましくない状態に遷移しました。
Y	電圧センサーが元の正常な状態に遷移しました。
Y	電圧センサーが望ましくない状態に遷移しました。
Y	温度センサーが元の正常な範囲に遷移しました。
Y	温度センサーが望ましくない範囲に遷移しました。
Y	電源機構センサーが正常な状態に遷移しました。
Y	電源機構センサーが望ましくない状態に遷移しました。

親トピック: [SNMP トラップ・サブスクリプションの管理](#)

リモート・ロギングの構成

リモート・ロギングを使用可能にすれば、ログ・エントリをアプライアンス外のリモート・サーバーに保存することができます。リモート・ロギングは、問題の分離や長期にわたる動作のモニターを容易にするために詳細デバッグ・ログ・レベルを設定する必要があるときに役立つことがあります。

始める前に

- イベントを listen してキャプチャーするためには、syslog サーバーが使用可能でなければなりません。
- カタログ・サーバー、コンテナ・サーバー、およびアプリケーション・サーバー (WebSphere® Application Server を使用している場合) の名前は、英数字のみからなる名前ではなければなりません。Syslog RFC 1364 では、TAG フィールドに非英数字を使用することは許可されていません。TAG フィールドには syslog メッセージ内のサーバー名が含まれています。

このタスクについて

履歴データの分析のためにリモート・ロギングを使用します。アプライアンスは、システム内で、限られた数のログ・ファイルしか保持しません。さらなる分析に備えてより多くのログ・ファイルを保存する必要がある場合は、リモート・ロギングを構成してください。リモート・ロギング・サーバーは複数のアプライアンスからのデータを集約するので、ファイルが同じリモート・ロギング・サーバーに送られるように集合全体を構成することができます。

手順

アプライアンスのユーザー・インターフェースでリモート・ロギング・サーバーを構成します。

- a. ユーザー・インターフェースで、「アプライアンス」 > 「トラブルシューティング」 > 「ロギング」 > 「リモート・ロギングの構成」をクリックします。
- b. リモート・ロギング・サーバーのリモート・ホストおよびポートを入力します。
- c. リモート・ロギング・サーバーに送るメッセージの重大度のしきい値を入力します。警告メッセージと重大メッセージの両方を送る場合は、**WARNING** を選択します。重大メッセージのみを送る場合は、**SEVERE** を選択します。

タスクの結果

保存と分析のために構成したリモート・ロギング・サーバーにメッセージが送られます。

親トピック: [モニター](#)

クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成

アプリケーション用に構成しているデータ・グリッドでセキュリティーを使用する場合、client.properties ファイルを構成して、そのデータ・グリッド・アプリケーションに受け渡す設定を含むようにします。

始める前に

WebSphere® DataPower® XC10 アプライアンス を使用するデータ・グリッド・アプリケーションが必要です。

手順

1. アプリケーションが使用しているデータ・グリッドを保護します。詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。
2. ユーザーの資格情報を提供するようにデータ・グリッド・アプリケーションを構成します。ユーザーの資格情報を構成するには、client.properties ファイルを使用する必要があります。WebSphere eXtreme Scale クライアント インストール済み環境の [wxs_home/properties](#) ディレクトリーにある sampleClient.properties ファイルを使用して、プロパティー・ファイルを作成することができます。client.properties ファイルに次の変更を行います。
 - **securityEnabled: securityEnabled** プロパティーをデフォルト値の true に設定して、認証を含むクライアント・セキュリティーを使用可能にします。
 - **credentialAuthentication: credentialAuthentication** プロパティーをデフォルト値の Supported に設定します。これは、クライアントが資格情報の認証をサポートすることを示します。
 - **transportType: transportType** プロパティーを TCP/IP に設定します。これは、Secure Sockets Layer (SSL) を使用しないことを意味します。
 - **singleSignOnEnabled: singleSignOnEnabled** プロパティーをデフォルト値の false に設定します。シングル・サインオンは使用不可になります。
 - データ・グリッドへのアクセス権限を付与されているユーザーの資格情報:

```
credentialGeneratorClass=com.ibm.websphere.objectgrid.security.plugins.builtins
.UserPasswordCredentialGenerator
credentialGeneratorProps=<username> <password>
```

例:

```
credentialGeneratorClass=com.ibm.websphere.objectgrid.security.plugins.builtins
.UserPasswordCredentialGenerator
credentialGeneratorProps=xcadmin xcadmin
```

- エンコードされたパスワードを使用する場合は、**FilePasswordEncoder** ユーティリティーを使用できます。このユーティリティーは、[wxs_home/ObjectGrid/bin](#) ディレクトリーにあります。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

データ・グリッド・アプリケーション用の TLS の構成

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

始める前に

- WebSphere® DataPower® XC10 アプライアンス を使用するデータ・グリッド・アプリケーションが必要で
す。
- アプライアンス管理権限が割り当てられている必要があります。
- アプライアンス構成に追加する、鍵ストアまたはトラストストアと、関連付けられたパスワードが必要です。既存
のトラストストアを変更したければ、アプライアンスからトラストストアをダウンロードすることができます。
クライアントの公開証明書を使用してトラストストアを更新する必要があります。詳しくは、[WebSphere
Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。
- **keytool** ツールに対するアクセス権限が必要です。このツールは、[java_home/bin](#) ディレクトリーにありま
す。

このタスクについて

アプライアンスは、データ・グリッドに接続するクライアントを信頼する必要があります。TLS 設定は、ユーザー・イ
ンターフェースおよびデータ・グリッドに適用されます。設定は、集合内のすべてのアプライアンスに適用されます。

手順

1. アクティブなトラストストアをダウンロードします。ユーザー・インターフェースで、「集合」 > 「設定」 >
「**Transport Layer Security (TLS)**」をクリックします。「アクティブなトラストストアのダウンロード」を
クリックし、ディスク上のファイルを保存した場所 (例えば/downloads/trustStore.jks ディレクトリーな
ど) を覚えておきます。
2. 必要に応じて、証明書を作成し、公開証明書をエクスポートします。
 - a. 鍵ストアに秘密鍵を作成します。次のコマンドは、key.jks 鍵ストアと、そこに保管される鍵
「ogsample」を作成します。この key.jks 鍵ストアが、SSL 鍵ストアとして使用されます。以下のコマ
ンドを実行します。

```
keytool -genkey -alias ogsample -keystore key.jks -storetype JKS -keyalg rsa
-dname "CN=ogsample, U=Your Organizational Unit, O=Your Organization, L=Your
City, S=Your State, C=Your Country" storepass ogpass -keypass ogpass -validity
3650
```

- b. 公開証明書をエクスポートします。次のコマンドは、「ogsample」という鍵の公開証明書を抽出して、そ
の鍵を temp.key ファイルに格納します。

```
keytool -export -alias ogsample -keystore key.jks -file temp.key -storepass
ogpass
```

3. クライアントの証明書をトラストストアに追加します。 **keytool** ツールを実行して、クライアントの公開証明書
をトラストストアにインポートします。

```
keytool -import -noprompt -alias "ogsample" -keystore /downloads/trustStore.jks
-file temp.key -storepass xc10pass -storetype jks
```

4. トラストストアの情報をアプライアンスにアップロードします。ユーザー・インターフェースで、「アプライア
ンス」 > 「設定」 > 「**Transport Layer Security (TLS)**」をクリックします。更新した
/downloads/trustStore.jks ファイルをアップロードします。「**TLS 設定の送信**」をクリックして、構成を
保存します。
5. client.properties ファイルを更新します。このファイルの場所およびこのファイル内のプロパティーについ
て詳しくは、[クライアント・プロパティー・ファイル](#)を参照してください。client.properties ファイル内の
以下のプロパティーを設定します。

```
securityEnabled=true
transportType=SSL-Required
alias=ogsample
contextProvider=IBMJSSE2
protocol=TLS
keyStoreType=JKS
keyStore=key.jks
keyStorePassword=ogpass
```

```
trustStoreType=JKS
trustStore=/downloads/trustStore.jks
trustStorePassword=xc10pass
```

次のタスク

- 新しい構成を使用してアプリケーションを再始動し、Transport Layer Security に接続します。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)
[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

トラブルシューティング

ご使用の環境で遭遇した問題のトラブルシューティングには、ログならびにトレース、監査データ、ハードウェア・データ、およびリリース情報が役立ちます。

[WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

ご使用の IBM 製品の問題を切り分け、解決するために、トラブルシューティング情報とサポート情報を利用できます。この情報には、ご使用の IBM 製品 (WebSphere® DataPower® XC10 アプライアンス を含む) と共に提供される問題判別リソースを使用するための説明が示されています。

[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)

WebSphere DataPower XC10 アプライアンス と関連付けられているログ・ファイルはアプライアンスに保管されます。表示可能なログは、ユーザー・インターフェースを使用してアプライアンスから直接表示するか、または検討するために ローカル・ファイル・システムへダウンロードすることができます。

[監査データのダウンロード](#)

監査アクティビティは WebSphere DataPower XC10 アプライアンス によって取り込まれます。適切な範囲の監査が有効になるように、一連の Auditable オブジェクトに対する ユーザー・アクティビティが保存されます。

[ログおよびトレース・データの分析](#)

ログ分析ツールを使用して、ランタイム環境のパフォーマンスを分析したり、環境内で発生した問題を解決したりできます。

[ハードウェア温度のモニター](#)

温度センサーは、アプライアンスの各種内部構成装置の温度を常に測定しています。これらの温度は、ユーザー・インターフェースを使用すると、単一のパネルで容易にモニターできます。

[シリアル接続を使用したイーサネット・インターフェースの状況詳細のモニター](#)

IBM® WebSphere DataPower XC10 アプライアンス でネットワーク問題が発生した場合、イーサネット・インターフェースのアクティビティおよび状況の追加詳細情報を表示できます。このトピックでは、アプライアンスとのシリアル接続を使用したこれらの詳細の表示方法を説明しますが、ユーザー・インターフェースでこれらの詳細を表示することもできます。

[アプライアンスからのアウトバウンド接続の検査](#)

アウトバウンド接続機能を使用すると、アプライアンスからネットワーク・アドレスに 到達可能かどうかを確認できます。

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

コマンドを実行して、アプライアンス・ハードウェアを再始動したり、アプライアンスを元の工場出荷時設定に再設定したり、アプライアンスをシャットダウンしたりすることができます。

[デッドロックのトラブルシューティング](#)

以下のセクションでは、いくつかの最も一般的なデッドロック・シナリオを説明し、その回避方法を提示します。

2.5+ [マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

説明するシナリオは、ロック・タイムアウト例外を引き起こしているマルチ区画トランザクションの例です。トランザクションの状態に応じて、解決策で、この問題を手動で解決する方法を説明します。

[キャッシュ統合のトラブルシューティング](#)

HTTP セッションや動的キャッシュ構成など、キャッシュ統合構成の問題をトラブルシューティングする場合、この情報を使用してください。

[インストールのトラブルシューティング](#)

インストールおよび更新の問題をトラブルシューティングする場合、この情報を使用してください。

[管理のトラブルシューティング](#)

サーバーの開始や停止、**xscmd** ユーティリティの使用など、管理についてのトラブルシューティングを行う場合、この情報を使用してください。

[リリース情報](#)

製品のサポート Web サイト、製品資料、および製品の最新の更新、制限、および既知の問題へのリンクが提供されています。

WebSphere DataPower XC10 アプライアンス のトラブルシューティング およびサポート

ご使用の IBM 製品の問題を切り分け、解決するために、トラブルシューティング情報とサポート情報を利用できます。この情報には、ご使用の IBM 製品 (WebSphere® DataPower® XC10 アプライアンス を含む) と共に提供される問題判別リソースを使用するための説明が示されています。

[問題のトラブルシューティングのための手法](#)

トラブルシューティングとは、体系的な方法で問題を解決することです。トラブルシューティングの目的は、想定どおりに機能しない理由を明らかにして、問題の解決方法を説明することです。ある種の一般的な手法が、トラブルシューティングのタスクに役立つこともあります。

[知識ベースの検索](#)

以下は英語のみの対応となります。多くの場合、IBM 知識ベースを検索することによって、問題の解決策を見つけることができます。使用可能なリソース、サポート・ツール、および検索方式を使用して、最適の結果を得ることができます。

[フィックスの入手](#)

以下は英語のみの対応となります。お客様の問題の解決に、プロダクトのフィックスが有効な場合があります。

[IBM サポートへの連絡](#)

IBM サポートでは、FAQ での回答や、製品に関する問題解決でユーザーのお手伝いをするにより、製品の障害に関する支援を提供します。

[IBM サポート用の診断情報の収集](#)

コマンド行インターフェースを使用して、IBM サポートに送信するログを収集できます。

[IBM との情報の交換](#)

以下は英語のみの対応となります。問題の診断や特定を行うには、ご使用のシステムからのデータと情報を IBM サポートに提供していただく必要がある場合があります。また、IBM サポートから、問題判別に使用するツールまたはユーティリティーをご提供する場合があります。

[サポート更新情報のサブスクリプション](#)

ご使用の IBM 製品に関する重要な情報を常に入手するために、更新情報をサブスクリプションすることができます。

親トピック: [トラブルシューティング](#)

問題のトラブルシューティングのための手法

トラブルシューティングとは、体系的な方法で問題を解決することです。トラブルシューティングの目的は、想定どおりに機能しない理由を明らかにして、問題の解決方法を説明することです。ある種の一般的な手法が、トラブルシューティングのタスクに役立つこともあります。

トラブルシューティング・プロセスの最初のステップは、問題の内容を完全に記述することです。問題の記述によって、問題の原因を突き止めるためにどこから着手すべきかを、お客様自身と IBM 技術サポート担当者がわかるようになります。このステップでは、以下の基本的な項目について明確にする必要があります。

- 問題の症状はどのようなものか。
- 問題の発生場所はどこか。
- 問題はいつ発生したか。
- 問題が発生する条件は何か。
- 問題は再現可能か。

通常は、これらの質問に回答することで問題が適切に記述され、問題解決につながります。

問題の症状はどのようなものか。

問題の記述を始めるときの最も明確な質問は「問題は何か」ということです。この質問は単純なように思われますが、問題をより具体的に説明する、さまざまな観点からの質問に細分化することができます。細分化した質問には以下のようなものがあります。

- だれが、または何が問題を報告するか。
- エラー・コードおよびメッセージの内容は何か。
- システムにどのような障害が発生したか。ループ、ハング、異常終了、パフォーマンス低下、不適切な結果などがあります。

問題の発生場所はどこか。

問題の発生源を特定することは、必ずしも簡単ではありませんが、これは問題を解決する上で最も重要な段階の 1 つです。報告元のコンポーネントと障害が発生しているコンポーネントの間に、いくつかのテクノロジー層が存在している場合があります。ネットワーク、データ・グリッド、およびサーバーは、問題を調査するときに考慮すべきコンポーネントの一部にすぎません。

次のような質問は、問題の層の切り分けのため、問題の発生箇所に焦点が集まるようにします。

- 問題は 1 つのプラットフォームまたはオペレーティング・システムに固有か、それとも複数のプラットフォームまたはオペレーティング・システムに共通か。
- 現在の環境と構成はサポートされているか。
- その問題はどのユーザーに対しても起こるのか。
- (マルチサイトのインストール済み環境の場合。) その問題はすべてのサイトで起こるのか。

ある層で問題が報告されても、必ずしもその層が問題の発生源とは限りません。問題の発生源の特定作業には、問題が存在する環境を理解することが含まれます。しばらく時間をかけて、問題のある環境のすべての内容 (オペレーティング・システムとバージョン、対応するすべてのソフトウェアとバージョン、ハードウェア情報など) を記述してください。サポートされる構成の環境で実行していることを確認してください。多くの場合、問題をトレースすると、ソフトウェア・レベルに互換性がないことがわかります (一緒に実行することを意図していないソフトウェア・レベルであったり、完全には一緒にテストされていないソフトウェア・レベルであったりします)。

問題はいつ発生したか。

障害発生に至るイベントについて、特に発生が 1 回限りのケースについて、詳しい時系列対照表を作成してください。最も簡単に時系列対照表を作成する方法は、逆方向に作業することです。エラーが報告された時点 (できればミリ秒単位に至るまで精密に) から開始して、使用可能なログと情報を通じて逆方向に作業します。一般に、診断ログ内で最初に見つかる疑わしいイベントまで調べる必要があります。

イベントの詳細な時系列対照表を作成するには、以下の質問に答えます。

- 問題が昼間または夜間の特定の時刻のみに発生するか。
- どれくらいの頻度で問題が発生するか。
- どのような一連のイベントを経て、問題が報告された時点に至ったのか。
- 環境の変更 (ソフトウェアまたはハードウェアのアップグレードやインストールなど) の後に問題が発生したか。

このようなタイプの質問に回答することで、問題調査のための基準の枠組みが得られます。

問題が発生する条件は何か。

問題の発生時に稼働していたシステムとアプリケーションを知ることは、トラブルシューティングの重要な部分です。ご使用の環境に関する次のような質問は、問題の根本原因の特定に役立ちます。

- 問題は同じタスクの実行時に常に発生するか。
- この問題が発生するときに、必ず発生する一連のイベントがあるか。
- 他のいずれかのアプリケーションでも同時に障害が発生したか。

このようなタイプの質問に回答することによって、問題が発生した環境が明らかになり、依存関係の相関付けができません。同じ時間の頃に複数の問題が発生したからといって、それらの問題が必ずしも関連しているとは限らないことを覚えておいてください。

問題は再現可能か。

トラブルシューティングの観点からすると、理想的な問題とは、再現できる問題であるということです。通常、問題を再現できる場合は、調査に役立つために自由に使用できるツールまたは手順の数が多くなります。このため、再現できる問題は、多くの場合デバッグや解決が簡単です。

しかし、再現できる問題には欠点があります。すなわち、その問題がビジネスに大きな影響を与える場合、問題が再発生することは好ましくありません。可能であれば、テスト環境または開発環境で問題を再現してください。一般に、このようにすると、調査時の柔軟性と管理能力が向上します。

- 問題をテスト・システムで再現することができるか。
- 複数のユーザーまたはアプリケーションで同じタイプの問題が発生しているか。
- 単一のコマンド、一連のコマンド、特定のアプリケーションのいずれを実行することによって問題を再現できるか。

親トピック: [WebSphere DataPower XC10 アプライアンスのトラブルシューティングおよびサポート](#)

知識ベースの検索

以下は英語のみの対応となります。多くの場合、IBM 知識ベースを検索することによって、問題の解決策を見つけることができます。使用可能なリソース、サポート・ツール、および検索方式を使用して、最適の結果を得ることができます。

このタスクについて

WebSphere® DataPower® XC10 アプライアンス のインフォメーション・センターを検索すれば、役立つ情報を見つけることができます。ただし、疑問の回答を得たり問題を解決するために、インフォメーション・センター以外で情報を検索する必要があることもあります。

手順

必要な情報を知識ベースで検索するには、以下のいずれか 1 つまたは複数の方法を使用します。

- IBM® Support Assistant (ISA) を使用してコンテンツを検索します。

ISA は、IBM ソフトウェア製品に関する質問への回答や問題の解決に役立つ、無料のソフトウェア保守ワークベンチです。ISA のダウンロードとインストールの手順については、[ISA Web サイト](#)を参照してください。

- [IBM Support Portal](#) を使用して必要なコンテンツを検索します。

IBM Support Portal は、IBM のすべてのシステム、ソフトウェア、およびサービスの、すべての技術サポート・ツールと情報が 1 つにまとめられた、中心となる場所です。この IBM Support Portal という 1 つの場所から、IBM エレクトロニック支援ポートフォリオにアクセスできます。各ページを調整して、問題の予防と迅速な問題解決に必要な情報やリソースに焦点を当てることができます。IBM Support Portal をよく理解するためには、このツールに関する[デモ・ビデオ](https://www.ibm.com/blogs/SPNA/entry/the_ibm_support_portal_videos) (https://www.ibm.com/blogs/SPNA/entry/the_ibm_support_portal_videos) をご覧ください。このビデオは、IBM Support Portal について紹介し、トラブルシューティングやその他のリソースについて説明し、ポートレットの移動、追加、削除によってページを調整する方法のデモを示します。

- WebSphere DataPower XC10 アプライアンス に関するコンテンツを検索するには、以下の追加の技術資料のいずれかを使用します。
 - [WebSphere DataPower XC10 アプライアンス リリース情報](#)
 - [WebSphere DataPower XC10 アプライアンス サポート Web サイト](#)
 - [WebSphere DataPower XC10 アプライアンスフォーラム \(forum\)](#)
- IBM マストヘッド検索を使用してコンテンツを検索します。IBM マストヘッド検索を使用するには、ibm.com® の任意のページの最上部にある「検索」フィールドに検索文字列を入力します。
- 外部の検索エンジン (Google、Yahoo、Bing など) を使用して、コンテンツを検索します。外部検索エンジンを使用すると、ibm.com ドメイン以外の情報が結果に含まれる可能性が高くなります。ただし、ibm.com 以外の場所にあるニュースグループ、フォーラム、およびブログで、IBM 製品に関する問題解決のための有用な情報が見つかることもあります。

ヒント: IBM 製品に関する情報を検索する場合は、「IBM」と製品の名称を検索に含めてください。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

フィックスの入手

以下は英語のみの対応となります。 お客様の問題の解決に、プロダクトのフィックスが有効な場合があります。

手順

フィックスを見つけてインストールするには、以下のようにします。

1. フィックスを入手するために必要なツールを取得します。 [IBM Update Installer](#) を使用して、WebSphere eXtreme Scale または WebSphere eXtreme Scale クライアントのさまざまなタイプの保守パッケージをインストールして、適用します。Update Installer は定期的に保守されるため、そのツールの最新バージョンを使用する必要があります。
2. 必要なフィックスを特定します。 [WebSphere DataPower® XC10 アプライアンスの推奨フィックス](#) を参照して、最新のフィックスを選択してください。フィックスを選択すると、そのフィックスのダウンロード資料が開きます。
3. フィックスをダウンロードします。ダウンロード資料で、「Download package」セクションの最新フィックスのリンクをクリックします。
4. フィックスを適用します。ダウンロード資料の「Installation Instructions」セクションに記載されている説明に従ってください。
5. フィックスやその他の IBM サポート情報について週次の E メール通知を受信できるよう、登録してください。

[Fix Central](#) からのフィックスの入手

以下は英語のみの対応となります。Fix Central を使用して、IBM サポートが推奨する、WebSphere DataPower XC10 アプライアンス を含むさまざまな製品のフィックスを見つけることができます。Fix Central では、ご使用のシステム用のフィックスを検索、選択、注文、およびダウンロードすることができ、その際に配信オプションを選択できます。以下は英語のみの対応となります。 お客様の問題の解決に、WebSphere DataPower XC10 アプライアンス プロダクトのフィックスが有効な場合があります。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

Fix Central からのフィックスの入手

以下は英語のみの対応となります。Fix Central を使用して、IBM サポートが推奨する、WebSphere® DataPower® XC10 アプライアンス を含むさまざまな製品のフィックスを見つけることができます。Fix Central では、ご使用のシステム用のフィックスを検索、選択、注文、およびダウンロードすることができ、その際に配信オプションを選択できます。以下は英語のみの対応となります。 お客様の問題の解決に、WebSphere DataPower XC10 アプライアンス プロダクトのフィックスが有効な場合があります。

手順

フィックスを見つけてインストールするには、以下のようにします。

1. フィックスを入手するために必要なツールを取得します。製品アップデート・インストーラーがインストールされていない場合は、それを取得します。インストーラーは [Fix Central](#) からダウンロードできます。このサイトでは、アップデート・インストーラーのダウンロード、インストール、および構成の手順を提供しています。
2. 製品として選択し、解決したい問題に関連のある 1 つ以上のチェック・ボックスを選択します。
3. 必要なフィックスを特定して選択します。
4. フィックスをダウンロードします。
 - a. ダウンロード資料を開き、「Download Package」セクションのリンクをたどります。
 - b. ファイルのダウンロード時に、保守ファイルの名前が変更されていないことを確認してください。このような変更は、意図的に行われる場合もあれば、特定の Web ブラウザーやダウンロード・ユーティリティーによって偶発的に行われる場合もあります。
5. フィックスを適用します。
 - a. ダウンロード資料の「Installation Instructions」セクションに記載されている説明に従ってください。
 - b. 詳しくは、製品資料の『Update Installer を使用したフィックスのインストール』のトピックを参照してください。
6. オプション: フィックスおよびその他の IBM サポートの更新については、登録して各週の E メール通知を受け取ってください。

親トピック: [フィックスの入手](#)

IBM サポートへの連絡

IBM サポートでは、FAQ での回答や、製品に関する問題解決でユーザーのお手伝いをするにより、製品の障害に関する支援を提供します。

始める前に

問題点に関する回答や解決方法について、リリース情報などのセルフ・ヘルプ・オプションを使用しても判明しない場合は、IBM サポートにご連絡ください。IBM サポートに連絡するには、お客様の会社または組織が有効な IBM 保守契約を締結しており、お客様が IBM に問題を送信する許可を受けている必要があります。使用可能なサポートのタイプについては、「*Software Support Handbook*」の『[Support portfolio](#)』のトピックを参照してください。

手順

問題について IBM サポートに連絡を取るには、次のようにしてください。

1. 問題を定義し、バックグラウンド情報を収集して、問題の重大度を判別します。詳しくは、「*Software Support Handbook*」の『[Getting IBM support](#)』のトピックを参照してください。
2. 診断情報を収集します。
3. IBM サポートに以下のいずれかの方法で問題を送信します。
 - IBM® Support Assistant (ISA) を使用する。を参照してください。
 - [IBM サポート・ポータル](#)によるオンライン・サポート: 「Service Request」ページの「Service Request」ポートレットから、お客様のすべてのサービス要求をオープン、更新、および表示できます。
 - 電話: 自国での連絡先の電話番号を調べるには、[Directory of worldwide contacts](#) の Web ページを参照してください。

タスクの結果

送信した問題がソフトウェア障害に関するものである場合、または文書の不備や不正確さに関するものである場合、IBM サポートはプログラム診断依頼書 (APAR) を作成します。APAR には問題が詳細に記載されます。IBM サポートでは、APAR が解決されてフィックスが配信されるまで、お客様が実施可能な回避策を可能な限り提供します。IBM では、解決された APAR を IBM サポート Web サイトに毎日公開しているため、同じ問題が発生した他のユーザーも同じ解決策を利用できます。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

IBM サポート用の診断情報の収集

コマンド行インターフェースを使用して、IBM サポートに送信するログを収集できます。

始める前に

コマンド行インターフェースの接続が必要です。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。

手順

- アプライアンス構成と状況情報のみを収集します。コマンド行インターフェースで、次のコマンドを実行します。

```
platform collect-pd <PDFFileName>
```

ファイル名を指定しない場合は、デフォルトで collect-pd.txt ファイルが作成されます。このファイルには、アプライアンスの状況コマンドからの出力とネットワーク構成の詳細が含まれます。

- アプライアンスのログ・ファイルとトレース・ファイルを収集します。コマンド行インターフェースで、次のコマンドを実行します。

```
platform must-gather <tarfilename> [<PDFFileName>]
```

このコマンドにより、アプライアンスのログ・ファイルとトレース・ファイルが含まれる tar ファイルが作成されます。コマンドを実行するには、tar ファイル名を指定する必要があります。このコマンドは、tar ファイルを作成する前に **platform collect-pd** コマンドを実行します。

- ログ・ファイルをクリアします。アプライアンスが長期間、実行を継続している場合は、ログ・ファイルが大きくなっている可能性があります。 **platform must-gather** コマンドからの tar ファイルが 1 ギガバイトを超える場合は、次のコマンドを実行して、ログ・ファイルをすべて、長さゼロにリセットすることができます。

```
clear-logs
```

- IBM サポートに送信するため、アプライアンスから生成済みファイルをコピーします。
 1. 以下のコマンドを使用して生成したファイルをリストします。

```
Console> file list
```

```
logsTest.tgz 1810748334 bytes created 2011-06-08 08:22:17-0500
```

```
collect-pd.txt 17477 bytes created 2011-06-08 08:18:34-0500
```

2. 以下のコマンドを使用して、アプライアンスのログ・ファイルをコピーします。

```
Console> file put logsTest.tgz
```

```
scp://root@linux010.myco.com:/opt/cp/logsTest.tgz
```

```
Password:*****
```

次のタスク

ファイルを IBM サポートに送信します。詳しくは、[IBM との情報の交換](#)を参照してください。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

IBM との情報の交換

以下は英語のみの対応となります。問題の診断や特定を行うには、ご使用のシステムからのデータと情報を IBM サポートに提供していただく必要がある場合があります。また、IBM サポートから、問題判別に使用するツールまたはユーティリティをご提供する場合があります。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

IBM サポートへの情報の送信

問題解決に必要な時間を減らすため、トレースおよび診断情報を IBM サポートに送信していただくことができます。

手順

IBM サポートに診断情報を提出するには、以下のようになります。

1. 問題管理レコード (PMR) を開きます。
2. 必要な診断データを収集します。診断データは、PMR の解決にかかる時間の節約に役立ちます。診断データは、手動で収集することも自動的に収集することもできます。
 - データを手動で収集する。
 - データを自動的に収集する。
3. .zip または .tar ファイル形式を使用してファイルを圧縮します。
4. ファイルを IBM に転送します。以下のいずれかの方法を使用して、IBM にファイルを転送することができます。
 - [IBM® Support Assistant](#)
 - [Service Request ツール](#)
 - 標準的なデータのアップロード方法: FTP、HTTP
 - 機密保護機能のあるデータのアップロード方法: FTPS、SFTP、HTTPS
 - E メール

z/OS 製品を使用していて、PMR の送信に ServiceLink / IBMLink を使用している場合は、E メールまたは FTP を使用して IBM サポートに診断データを送信できます。

これらのデータ交換方法は、すべて [IBM サポートの Web サイトで説明されています。](#)

IBM サポートからの情報の受信

IBM 技術サポートの担当者から、診断ツールやその他のファイルのダウンロードをお願いする場合があります。FTP を使用して、これらのファイルをダウンロードすることができます。

始める前に

IBM 技術サポートの担当者から、ファイルのダウンロードに使用する推奨サーバーと、アクセス先の正確なディレクトリおよびファイル名が指定されていることを確認してください。

手順

IBM サポートからファイルをダウンロードするには、以下のようになります。

1. FTP を使用して、IBM 技術サポートの担当者が指定したサイトに接続し、anonymous としてログインします。パスワードとして E メール・アドレスを使用してください。
2. 以下のように、適切なディレクトリに移動します。
 - a. /fromibm ディレクトリに移動します。

```
cd fromibm
```

- b. IBM 技術サポートの担当者が指定したディレクトリに移動します。

```
cd nameofdirectory
```

3. セッションのバイナリー・モードを有効にします。

```
バイナリー
```

4. **get** コマンドを使用して、IBM 技術サポートの担当者が指定したファイルをダウンロードします。

```
get filename.extension
```

5. FTP セッションを終了します。

quit

サポート更新情報のサブスクライブ

ご使用の IBM 製品に関する重要な情報を常に入手するために、更新情報をサブスクライブすることができます。

このタスクについて

製品の更新情報を受け取るようにサブスクライブすることによって、特定の IBM サポート・ツールおよびリソースに関する重要な技術情報と更新情報を受け取ることができます。次の 2 つの方法のうちいずれかを使用して、更新情報をサブスクライブできます。

ソーシャル・メディアのサブスクリプション

製品に関し、次の RSS フィードが使用可能です。

- [WebSphere® DataPower® XC10 アプライアンスフォーラム](#)の RSS フィード

RSS の一般情報 (入門情報、RSS を使用できる IBM Web ページのリストなど) については、[IBM ソフトウェア・サポート RSS フィード](#)のサイトをご覧ください。

マイ通知 (My Notifications)

「マイ通知 (My Notifications)」では、任意の IBM 製品のサポート更新情報をサブスクライブできます。「マイ通知 (My Notifications)」は、以前に使用されていた同様のツール「マイ・サポート」に置き換わるものです。「マイ通知 (My Notifications)」を使用すると、E メールによる告知を、毎日、または週 1 回受信するように指定できます。発表、ヒント、製品フラッシュ (アラートとも言う)、ダウンロード、ドライバーなど、受信したい情報のタイプを指定できます。「マイ通知 (My Notifications)」では、通知を受信する製品、およびユーザーのニーズに最も適した配信方法をカスタマイズして分類することができます。

手順

サポート更新情報をサブスクライブするには、次のようにします。

1. [WebSphere DataPower XC10 アプライアンスフォーラム](#)の RSS フィードをサブスクライブします。
 - a. 「サブスクリプション」ページで、RSS フィード・アイコンをクリックします。
 - b. フィードのサブスクライブに使用するオプションを選択してください。
 - c. 「サブスクライブ」をクリックします。
2. 「マイ通知 (My Notifications)」をサブスクライブするには、[IBM® サポート・ポータル](#)にアクセスし、「通知」ポートレットの「**マイ通知 (My Notifications)**」をクリックします。
3. IBM ID とパスワードを使用してサインインし、「送信」をクリックします。
4. 更新を受け取る内容と方法を指定します。
 - a. 「サブスクライブ」タブをクリックします。
 - b. 該当するソフトウェア・ブランドまたはハードウェアのタイプを選択します。
 - c. 1 つ以上の製品名を選択して、「**続行**」をクリックします。
 - d. 更新を E メールで受け取るか、オンラインで指定のフォルダーに受け取るか、RSS または Atom フィードで受け取るか、更新の受信方法に関する設定を選択します。
 - e. 受信する資料更新のタイプ (製品ダウンロードに関する新規情報、ディスカッション・グループのコメントなど) を選択します。
 - f. 「**実行依頼**」をクリックします。

タスクの結果

RSS フィードと「マイ通知 (My Notifications)」の設定を変更するまで、要求した更新情報に関する通知を受け取ることになります。必要に応じて、この設定内容は変更することができます (ある製品の使用を中止し、別の製品の使用を開始した場合など)。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

関連情報

🔗 [IBM ソフトウェア・サポート RSS フィード](#)

🔗 [「マイ通知 \(My Notifications\)」サポート・コンテンツの更新情報のサブスクライブ](#)

🔗 [IBM 技術サポートの「マイ通知 \(My Notifications\)」](#)

🔗 [IBM 技術サポート概要の「マイ通知 \(My Notifications\)」](#)

WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作

WebSphere® DataPower® XC10 アプライアンス と関連付けられているログ・ファイルはアプライアンスに保管されます。表示可能なログは、ユーザー・インターフェースを使用してアプライアンスから直接表示するか、または検討するために ローカル・ファイル・システムへダウンロードすることができます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ログ・データは、アプライアンス上に直接保管されます。 trace.zip ファイルを IBM® のサポート・チームに提供できます。

手順

1. ユーザー・インターフェースで、「アプライアンス」「トラブルシューティング」をクリックし、「ロギング」を展開します。 ログ・データはアプライアンスに保管されます。「ロギング」セクションを展開すると、ログ・ビューアーを使用して使用可能なログにアクセスできるようになります。また、使用可能なログをファイル・システムにダウンロードして、さらに検討することもできます。
2. 現在のログ・ファイルを表示します。 次のいずれかのリンクをクリックしてログを表示します。
 - 現行エラー・ファイルの表示
 - 現行トレース・ファイルの表示
 - **2.5+** 現行の正常性ファイルの表示

ログ・ビューアー用に新しい Web ブラウザー・ウィンドウが開きます。ログ・ビューアーは、選択したログの末尾 10 行を表示するために使用されます。新しいログ項目が発生すると、ログ・ビューアーに付加されます。ログ・ビューアーには、ログ・ビューアーの動作を制御するいくつかのアクションがあります。

- a. 新しいログ項目の付加を停止するには、「一時停止」をクリックします。このアクションは、ログ・ビューアーが新しい項目を受け入れている場合にのみ使用できます。
 - b. 新しい項目を付加できるようにするには、「再始動」をクリックします。このアクションは、ログ・ビューアーが新しい項目を受け入れていない場合にのみ使用できます。
 - c. ログ・ビューアーからすべてのデータをクリアするには、「クリア」をクリックします。このアクションは、ログ・ビューアーが新しい項目を受け入れているかどうかにかかわらず使用できます。
3. 使用可能なすべてのログをファイル・システムに保存するには、「ログ・ファイルのダウンロード」をクリックします。発生したイベントに関する情報を表示したい場合は、このリンクを使用する必要があります。ウィンドウが表示され、圧縮ファイルを開いたり、ファイル・システムに保存したりすることができます。 trace.zip ファイルには、サポート・チームに提供するログ・ファイルがすべて含まれています。

trace.zip ファイルには一連の CSV ファイルが含まれていますが、これはアプライアンス上のサーバーの履歴データをトラッキングする際に使用できます。 trace.zip ファイル内で、CSV ファイルは `server_name/logs` ディレクトリにあります。ファイル名は `jvmstats.log`、`mapstats.log`、および `ogstats.log` です。マップまたはサーバーによるメモリー使用量、マップのヒット率、トランザクション時間、スループットなどの情報を CSV ファイルから収集することができます。

4. 「トレース・レベルの構成」セクションで、トレース・レベルの表示または変更を行うことができます。「管理コンソール」または「データ・グリッド」のトレース・レベルが編集できます。管理コンソールの場合、デフォルト・ロガーの出力を以下のトレース・レベルのいずれか 1 つに変更できます。
 - OFF
 - SEVERE
 - WARNING
 - INFO
 - FINE
 - a. トレース・ストリングを追加します。「トレース設定の追加」をクリックし、有効なトレース・ストリングを入力します。新規トレース・ストリングのトレース・レベルは、デフォルトで *INFO* に設定されます。
 - b. トレース・ストリングを削除します。トレース・ストリングを削除するには、そのトレース・ストリングの横にある「除去」アイコン (✖) をクリックします。
 - c. トレース・レベルを変更します。トレース・レベル値をクリックして、新しいトレース・レベルを選択します。指定したトレース・ストリングの新しいトレース・レベルをコミットするには、「保存」をクリックします。

次のタスク

Web コンソールが使用不可でも、trace.zip ファイルを取り出すことができます。コマンド行から以下のステップを実行して、IBM サポートに提供するログ・ファイルを取得してください。

1. ログ・ファイルを収集します。

```
Console> platform must-gather
```

2. 生成されたログ・ファイルをリストします。

```
Console> file list
```

3. アプライアンスからログをコピーします。

```
Console> file put <whateverNameIWant>.tgz  
scp://root@remoteMachine.myco.com:/myRemoteMachinePath/<whateverNameIWant>.tgz
```

[syslog を使用して WebSphere DataPower XC10 アプライアンス ログ・レコードをリモートの UNIX システムに送信する](#)

リモート・ロギングを使用可能に設定して、ログ・エントリをリモート・サーバーに送信することができます。リモート・ロギングは監査目的で情報を保存する場合に便利です。

親トピック: [トラブルシューティング](#)

関連タスク:

[CSV ファイルによるモニター](#)

関連資料:

[CSV ファイルの統計定義](#)

syslog を使用して WebSphere DataPower XC10 アプライアンス ログ・レコードをリモートの UNIX システムに送信する

リモート・ロギングを使用可能に設定して、ログ・エントリをリモート・サーバーに送信することができます。リモート・ロギングは監査目的で情報を保存する場合に便利です。

始める前に

- アプライアンス管理権限が必要です。
- syslog デーモンのセットアップ方法と syslog レコードの受信方法についての具体的な説明は、ご使用のオペレーティング・システムの資料を参照してください。
- WebSphere® DataPower® XC10 アプライアンス は syslog プロトコル RFC 3164 をサポートします。
- ログ・レコードはユーザー・データグラム・プロトコル (UDP) で送信され、したがって応答不要送信です。UDP プロトコルのとおり、メッセージ送達は保証されません。
- どのコンテナ・サーバー・プロセスがログに記録されて、WebSphere DataPower XC10 アプライアンス 上のリモート・システムに送信される対象になるか注意してください。

- **cs**
- **xsServer00-xsServerXY**: ここで **XY** は、アプライアンス上のコンテナ・プロセスの数を表します。
- **xsa.admin**

リモート・ロギングを使用可能に設定すると、すべてのプロセスが、syslogd への送信対象となります。リモート・ロギングが使用不可になっていると、どのプロセスも syslogd に送信されません。いずれかのプロセスが停止していたり、あるいは一時的に使用不可になっている場合、ロギングは、まだアクティブなプロセス上で続行されます。WebSphere DataPower XC10 アプライアンス 上の syslog に問題の可能性が疑われ、トラブルシューティングを行う必要がある場合は、アプライアンス・ユーザー・インターフェースでトレース・レベル **SYSLOG=all** を追加することができます。

このタスクについて

アーカイブ・ログ・イベントのアーカイブ用にリモート・ロギングを使用します。WebSphere DataPower XC10 アプライアンス では、スペースを節約するために、保存するログ・ファイルの数および保存期間を制限してログ・レコードを保存しています。ログ・レコードを長期間アーカイブしたり、ログ有効期限を過ぎたアーカイブ・ログ・レコードを分析したりするには、WebSphere DataPower XC10 アプライアンス でリモート・ロギングを構成し、使用可能に設定してください。

手順

1. アプライアンス・ユーザー・インターフェースで、「アプライアンス」 > 「トラブルシューティング」 > 「ロギング」をクリックします。
2. 「リモート・ロギングの構成」セクションで、「リモート・ロギングの有効化」を選択します。この設定により、履歴データ分析のためのリモート・ロギングが使用可能になります。イベントを listen してキャプチャーするためには、syslogd サーバーが使用可能でなければなりません。以下の設定を指定します。
 - a. リモート・ホスト - ログ・レコードの送信先である、リモート syslogd サーバーのホスト名または IP アドレスを指定します。値を「localhost」、「localhost-v6」、「127.0.0.1」、「::1」にすることはできません。アプライアンスから、そのホスト、または IP アドレスに ping できることを確認してください。
 - b. リモート・ポート - ログ・レコードの送信先の syslogd サーバーのポート番号を指定します。有効な値は、0-65535 で、デフォルト値は 512 です。
 - c. しきい値: リモート・ロギング・サーバーに送るメッセージの重大度のしきい値を指定します。警告メッセージと重大メッセージの両方を送る場合は、**WARNING** という値を入力してください。重大メッセージのみを送る場合は、**SEVERE** を選択します。
 - d. **syslog 機能**: ログ・メッセージの送信に使用される syslog ロギング機能を指定します。この設定により、リモート・システム上で実行されている syslog デーモンがログ・メッセージを入力するファイルが決定されます。例えば、この機能を「ユーザー」に設定すると、ほとんどの syslog デーモンはメッセージを /var/log/user.log ファイルに保管します。「メール」を選択すると、syslog デーモンはメッセージを /var/log/mail.log ファイルに保管します。ログ・レコードの宛先は、syslog デーモンをどのように構成したかによって決まります。syslogd 構成については、ご使用のオペレーティング・システムのための指示を参照してください。

設定を編集するには、「変更の適用」をクリックします。変更は syslog メッセージの影響を受けるすべてのプロセスに適用されます。

タスクの結果

保存と分析のために、アプライアンスのログ・レコードが、構成済みのリモート・ロギング・サーバーに送信されます。

親トピック: [WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)

監査データのダウンロード

監査アクティビティは WebSphere® DataPower® XC10 アプライアンス によって取り込まれます。適切な範囲の監査が有効になるように、一連の Auditable オブジェクトに対する ユーザー・アクティビティが保存されます。

このタスクについて

特定の Auditable オブジェクトについてのユーザー・アクティビティを収集した監査データが、アプライアンス上に保管されます。監査データに含まれる Auditable オブジェクトについては、下の表を参照してください。

表 1. Auditable オブジェクト

オブジェクト	追加/作成	削除	更新	その他
同化	Y	Y	N/A	
グループ	Y	Y	Y	
Session	N/A	N/A	N/A	タイムアウト、ログイン
ユーザー	Y	Y	Y	
ゾーン	Y	Y	Y	
ゾーン・アクティベーション	Y	Y	Y	

適切な監査手順のためには、データ安全性が非常に重要です。データ安全性を保証するため、監査データが必要になったときに、アプライアンスから監査データを取得する必要があります。監査データのダウンロード後は、ユーザーがそれぞれの方法でデータ安全性を維持する責任があります。以下の手順で、アプライアンスから監査データを取得してください。

手順

1. ユーザー・インターフェースの「監査」セクションを開きます。「アプライアンス」>「トラブルシューティング」をクリックします。「監査」セクションを展開します。「監査」セクションを展開すると、アプライアンスで使用可能な監査データにアクセスできます。
2. 監査データをダウンロードします。すべてのデータまたはフィルターに掛けられたデータをダウンロードできます。

- 「データをすべてダウンロード」

「データをすべてダウンロード」をクリックして、audit.zip ファイルをダウンロードします。audit.zip ファイルには、アプライアンスで現在使用可能なすべてのデータが含まれています。コマ区切り値 (CSV) フォーマットの監査データが必要な場合、または使用可能な監査データのサブセットのみが必要な場合は、日付範囲を調整して、フィルター処理を行ったデータをダウンロードできます。

- 「フィルターに掛けられたデータのダウンロード」

日付範囲を調整し、時間帯を設定します。日付範囲および時間帯の設定は、フィルター処理されたデータの生成に使用されます。これらの設定は、データをフィルターに掛ける場合にのみ適用され、すべてのデータをダウンロードする際には適用されません。時間帯設定のデフォルト値は、アプライアンスに関連付けられた時間帯です。デフォルトの日付範囲には、最新の 30 日分のデータが含まれます。時間帯と日付範囲を調整することで、データに組み込む監査イベントを指定します。

「フィルターに掛けられたデータのダウンロード」をクリックして、指定した日付範囲の監査データを CSV フォーマットでダウンロードします。audit.csv ファイルはコマ区切り値フォーマットのファイルであり、ファイル・システムにダウンロードできます。CSV フォーマットのデータをベンダー・ソフトウェアにインポートすれば、フォーマットや表示の設定をさらに操作できます。

親トピック: [トラブルシューティング](#)

ログおよびトレース・データの分析

ログ分析ツールを使用して、ランタイム環境のパフォーマンスを分析したり、環境内で発生した問題を解決したりできます。

このタスクについて

環境内の既存のログ・ファイルやトレース・ファイルからレポートを生成できます。これらのビジュアル・レポートには次のような用途があります。

- **ランタイム環境の状況およびパフォーマンスの分析**
 - デプロイメント環境の整合性
 - ロギングの頻度
 - 実行中トポロジと構成されているトポロジの比較
 - 予定外のトポロジ変更
 - クォーラム状況
 - 区画の複製の状況
 - メモリー、スループット、プロセッサ使用量などの統計
- **環境内の問題のトラブルシューティング**
 - 特定時点でのトポロジ・ビュー
 - クライアント障害時のメモリー、スループット、プロセッサ使用量などの統計
 - 現在のフィックスパック・レベル、チューニング設定
 - クォーラム状況

[ログ分析の概要](#)

環境内の問題のトラブルシューティングに役立つ **xsLogAnalyzer** ツールを使用できます。

[ログ・ファイルの保管ポリシー](#)

[ログ分析の実行](#)

任意のコンピューターのログ・ファイルやトレース・ファイルのセットに対して **xsLogAnalyzer** ツールを実行できます。

[ログ分析用カスタム・スキャナーの作成](#)

ログ分析用のカスタム・スキャナーを作成できます。スキャナーを構成してから、**xsLogAnalyzer** ツールを実行すると、結果がレポート内に生成されます。カスタム・スキャナーは、指定された正規表現に基づいてイベント・レコードのログをスキャンします。

[ログ分析のトラブルシューティング](#)

xsLogAnalyzer ツールおよびこのツールで生成されるレポートに関する問題を診断し、修正する場合、次のトラブルシューティング情報を使用してください。

親トピック: [トラブルシューティング](#)

ログ分析の概要

環境内の問題のトラブルシューティングに役立つ **xsLogAnalyzer** ツールを使用できます。

すべてのフェイルオーバー・メッセージ

フェイルオーバー・メッセージの総数を一定時間のチャートで表示します。また、影響を受けたサーバーを含む、フェイルオーバー・メッセージのリストも表示します。

すべての **eXtreme Scale** 重大メッセージ

メッセージ ID を関連する説明およびユーザー処置と一緒に表示します。これにより、メッセージを検索する時間を節約できます。

すべての例外

メッセージ、発生回数、例外の影響を受けたサーバーも含めて、上位 5 つの例外を表示します。

トポロジーの要約

ログ・ファイルに基づいて、どのようにトポロジーが構成されているかをダイアグラムで表示します。この要約を使用して実際の構成と比較することができ、構成エラーを特定できる場合があります。

トポロジーの整合性: オブジェクト・リクエスト・ブローカー (**ORB**) 比較表

環境での ORB 設定を表示します。環境全体で設定が整合しているか判断するのに助けるために、この表を使用できます。

イベント・タイムライン・ビュー

データ・グリッドで発生したライフサイクル・イベント、例外、重大なメッセージ、初期障害データ・キャプチャー機能 (FFDC) イベントなどのさまざまなアクションのタイムライン図を表示します。

親トピック: [ログおよびトレース・データの分析](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析用カスタム・スキャナーの作成](#)

[ログ分析のトラブルシューティング](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ログ・ファイルの保管ポリシー

下記の表は、WebSphere® DataPower® XC10 アプライアンス 上でのログ・ファイルの保管に関するポリシーについて説明したものです。

表 1. xsa.app トレース・ファイル

xsa.app トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	5	true
audit.log	30	2097152	5	true
*stats.log	N/A	N/A	N/A	N/A

表 2. xsa.admin トレース・ファイル

xsa.admin トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	2	true
audit.log	30	2097152	5	true
health.log	30	2097152	2	true
*stats.log	N/A	N/A	N/A	N/A

表 3. cs トレース・ファイル

cs トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	2	true
audit.log	30	2097152	5	true
health.log	30	2097152	2	true
*stats.log	N/A	N/A	N/A	N/A

表 4. xsServer* トレース・ファイル

xsServer* トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	2	true
audit.log	30	2097152	5	true
health.log	30	2097152	2	true
*stats.log	N/A	104857600	10	true

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析用カスタム・スキャナーの作成](#)

[ログ分析のトラブルシューティング](#)

ログ分析の実行

任意のコンピューターのログ・ファイルやトレース・ファイルのセットに対して **xsLogAnalyzer** ツールを実行できます。

始める前に

- WebSphere® eXtreme Scale クライアント インストール済み環境が必要です。詳しくは、[WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- アプライアンスからログ・ファイルとトレース・ファイルを集めます。詳しくは、[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)を参照してください。
- 生成されるレポートのカスタム・スキャナーを作成する場合は、ツールを実行する前に スキャナー仕様プロパティ・ファイルと構成ファイルを作成してください。詳しくは、[ログ分析用カスタム・スキャナーの作成](#)を参照してください。

手順

1. **xsLogAnalyzer** ツールを実行します。

スクリプトは次の場所にあります。

- スタンドアロン・インストールの場合: [wxs_install_root](#)/ObjectGrid/bin
- WebSphere Application Server と統合されたインストールの場合: [was_root](#)/bin

ヒント: ログ・ファイルが大きい場合、レポートを実行するときに **-startTime**、**-endTime**、および **-maxRecords** パラメーターを使用して、スキャンするログ・エントリーの数 を制限することを検討してください。レポートを実行するときにこれらのパラメーターを使用すると、レポートが見やすくなるうえ、レポートをより効率的に実行できます。同一セットのログ・ファイルを対象に複数のレポートを実行できます。

```
xsLogAnalyzer.sh|bat -logsRoot c:¥myxslogs -outDir c:¥myxslogs¥out  
-startTime 11.09.27_15.10.56.089 -endTime 11.09.27_16.10.56.089 -maxRecords 100
```

-logsRoot

評価するログ・ディレクトリーへの絶対パスを指定します (必須)。

-outDir

レポートの出力を書き込む既存のディレクトリーを指定します。値を指定しないと、レポートは **xsLogAnalyzer** ツールの ルート・ロケーションに書き込まれます。

-startTime

ログ内の評価する開始時刻を指定します。日付のフォーマットは、*year.month.day_hour.minute.second.millisecond*です。

-endTime

ログ内の評価する終了時刻を指定します。日付のフォーマットは、*year.month.day_hour.minute.second.millisecond*です。

-trace

トレース・ストリング (ObjectGrid*=all=enabled など) を指定します。

-maxRecords

レポート内に生成するレコードの最大数を指定します。デフォルトは 100 です。値を 50 と指定した場合、指定された期間の最初の 50 レコードが生成されます。

2. 生成されたファイルを開きます。出力ディレクトリーを定義しなかった場合、レポートは *report_date_time* というフォルダー内に生成されます。レポートのメインページを開くには、*index.html* ファイルを開きます。
3. レポートを使用して、ログ・データを分析します。次のヒントを使用して、レポート表示のパフォーマンスを最大にしてください。
 - ログ・データの照会のパフォーマンスを最大にするには、できるだけ具体的な情報を使用します。例えば、*server_host_name* の照会より *server* の照会のほうが実行時間が長くなり、返される結果も多くなります。
 - 一部のビューでは、一度に表示されるデータ・ポイントの数が制限されます。ビュー内の現在のデータを変更して (開始時刻や終了時刻を変更するなどして)、表示される時間セグメントを調整できます。

次のタスク

xsLogAnalyzer ツールや生成されるレポートのトラブルシューティングの詳細については、[ログ分析のトラブルシューティング](#)を参照してください。

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析用カスタム・スキャナーの作成](#)

[ログ分析のトラブルシューティング](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ログ分析用カスタム・スキャナーの作成

ログ分析用のカスタム・スキャナーを作成できます。スキャナーを構成してから、**xsLogAnalyzer** ツールを実行すると、結果がレポート内に生成されます。カスタム・スキャナーは、指定された正規表現に基づいてイベント・レコードのログをスキャンします。

手順

1. カスタム・スキャナーで実行する正規表現を指定したスキャナー仕様プロパティ・ファイルを作成します。
 - a. プロパティ・ファイルを作成し、保存します。ファイルは `loganalyzer_root/config/custom` ディレクトリ内に存在しなければなりません。ファイルには好きな名前を付けることができます。ファイルは新規スキャナーで使用されるので、スキャナーの名前をプロパティ・ファイルの名前の一部にすると (例えば、`my_new_server_scanner_spec.properties`) 便利です。
 - b. 次のプロパティを `my_new_server_scanner_spec.properties` ファイルに組み込みます。

```
include.regular_expression = REGULAR_EXPRESSION_TO_SCAN
```

`REGULAR_EXPRESSION_TO_SCAN` 変数は、ログ・ファイルのフィルタリングに使用する正規表現です。

例: `xception` と `rror` 両方のストリングを含んでいる行 (ストリングの出現順序は問いません) のインスタンスをスキャンするには、**include.regular_expression** プロパティを次の値に設定します。

```
include.regular_expression = (xception.+rror)|(rror.+xception)
```

この正規表現によって、`rror` ストリング または `xception` ストリングのいずれかが存在する場合、イベントが記録されます。

例:

ログ内の各行をスキャンして、句 `xception` または句 `rror` のいずれかのストリングを含んでいる行のインスタンスを探すには、**include.regular_expression** プロパティを次の値に設定します。

```
include.regular_expression = (xception)|(rror)
```

`rror` ストリングまたは `xception` ストリングのいずれかが存在する場合、この正規表現によってイベントが記録されます。

2. **xsLogAnalyzer** ツールがスキャナーを作成するために使用する構成ファイルを作成します。
 - a. 構成ファイルを作成し、保存します。ファイルは `loganalyzer_root/config/custom` ディレクトリ内に存在しなければなりません。ファイルの名前は、`scanner_nameScanner.config` のようにします。ここで、`scanner_name` は、新規スキャナーの固有の名前です。例えば、このファイルは `serverScanner.config` という名前にできます。
 - b. 次のプロパティを `scanner_nameScanner.config` ファイルに組み込みます。

```
scannerSpecificationFiles = LOCATION_OF_SCANNER_SPECIFICATION_FILE
```

`LOCATION_OF_SCANNER_SPECIFICATION_FILE` 変数は、前のステップで作成した仕様ファイルの場所 (パス) です。例: `loganalyzer_root/config/custom/my_new_scanner_spec.properties`。セミコロンで区切ったリストを使用して、複数のスキャナー仕様ファイルを指定することもできます。

```
scannerSpecificationFiles =  
LOCATION_OF_SCANNER_SPECIFICATION_FILE1;LOCATION_OF_SCANNER_SPECIFICATION_FILE2
```

3. **xsLogAnalyzer** ツールを実行します。詳しくは、[ログ分析の実行](#)を参照してください。

タスクの結果

xsLogAnalyzer ツールを実行すると、構成した カスタム・スキャナー用の新しいタブがレポートに含まれています。各タブには、次のビューがあります。

チャート

記録されたイベントを示すプロット・グラフ。イベントは検出された順序で表示されます。

テーブル

記録されたイベントのテーブル表示。

要約レポート

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析のトラブルシューティング](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ログ分析のトラブルシューティング

xsLogAnalyzer ツールおよびこのツールで生成されるレポートに関する問題を診断し、修正する場合、次のトラブルシューティング情報を使用してください。

手順

- **問題:** **xsLogAnalyzer** ツールを使用してレポートを生成中、メモリー不足状態が発生する。発生する可能性があるエラーの例は、次のとおりです: `java.lang.OutOfMemoryError: GC overhead limit exceeded`。

解決策: **xsLogAnalyzer** ツールは Java 仮想マシン (JVM) 内で実行されます。**xsLogAnalyzer** ツールの実行時に、ある設定を指定することで、ヒープ・サイズを大きくするよう JVM を構成してから、ツールを実行することができます。ヒープ・サイズを大きくすることで、より多くの イベント・レコードを JVM メモリー内に保管できるようになります。オペレーティング・システムに十分なメイン・メモリーがあれば、最初は 2048M を設定してはじめてください。**xsLogAnalyzer** ツールを実行するのと同じコマンド行インスタンスで、次のように最大 JVM ヒープ・サイズを設定します。

```
java -XmxHEAP_SIZEm
```

`HEAP_SIZE` 値は、任意の整数にでき、JVM ヒープに割り振られるメガバイト数を表します。

例えば、`java -Xmx2048m` を実行できます。メモリー不足メッセージが続く場合、または 2048m 以上のメモリーを割り振るためのリソースがない場合は、ヒープ内に保持するイベントの数を制限してください。

`-maxRecords` パラメーターを **xsLogAnalyzer** コマンドに渡すと、ヒープ内のイベントの数を制限できます。

- **問題:** **xsLogAnalyzer** ツールで生成されたレポートを開くと、ブラウザーがハングするか、ページが読み込まれない。

原因: 生成された HTML ファイルが大きすぎるため、ブラウザーが読み込むことができません。これらのファイルが大きすぎる理由は、分析対象のログ・ファイルの範囲が広すぎるためです。

解決策: **xsLogAnalyzer** ツールを実行するときに `-startTime`、`-endTime`、および `-maxRecords` パラメーターを使用して、スキャンするログ・エントリーの数を制限することを検討してください。レポートを実行するときにこれらのパラメーターを使用すると、レポートが見やすくなるうえ、レポートをより効率的に実行できます。同一セットのログ・ファイルを対象に複数のレポートを実行できます。

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析用カスタム・スキャナーの作成](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ハードウェア温度のモニター

温度センサーは、アプライアンスの各種内部構成装置の温度を常に測定しています。これらの温度は、ユーザー・インターフェースを使用すると、単一のパネルで容易にモニターできます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

IBM® WebSphere® DataPower® XC10 アプライアンス は、各種内部構成装置をモニターして、アプライアンスが安全な温度で作動していることを確認します。

手順

1. 「アプライアンス」 > 「トラブルシューティング」にナビゲートします。
2. 「ハードウェア温度」を展開します。
3. ハードウェア温度を確認します。

各温度センサーの最新の読み取り値が、3桁の有効数字を使用して表示されます。

- 構成装置の温度が標準的な作動温度である場合は、緑色のアイコン (✅) が表示されます。
- 構成装置の温度が安全な作動レベルを超えている場合は、黄色の警告アイコン (⚠️) が表示されます。

タスクの結果

上記の手順が正常に完了すると、有効な温度読み取り値が表示され、アプライアンスが安全な温度で作動していることを確認できます。

次のタスク

引き続き、発生している問題の[トラブルシューティング](#)を行います。

親トピック: [トラブルシューティング](#)

シリアル接続を使用したイーサネット・インターフェースの状況詳細のモニター

IBM® WebSphere® DataPower® XC10 アプライアンス でネットワーク問題が発生した場合、イーサネット・インターフェースのアクティビティおよび状況の追加詳細情報を表示できます。このトピックでは、アプライアンスとのシリアル接続を使用したこれらの詳細の表示方法を説明しますが、ユーザー・インターフェースでこれらの詳細を表示することもできます。

始める前に

アプライアンスに対する物理的なアクセス権限を保持しており、xcadmin としてログインできる必要があります。

手順

1. xcadmin ユーザーとして、アプライアンスとのシリアル接続を確立します。

ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC を、アプライアンスのシリアル・ポートにシリアル接続する必要があります。シリアル・コンソールとして PC を使用している場合は、Windows 用または Linux 用の PC ベースのシリアル通信プログラムを使用します。提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して、アプライアンスに接続します。

重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。

シリアル・コンソール設定は `9600.8.n.1` です。

2. イーサネット・インターフェースの状況を表示します。以下のコマンドを使用して、イーサネット・インターフェースの状況を表示します。

```
netif status <interface>
```

ここで、<interface> は、照会するイーサネット・インターフェースの名前です。出力例を以下に示します。

```
Console> netif status mgt0
mgt0    generic MTU:1500 flags:UP BROADCAST RUNNING MULTICAST
        inet addr:127.0.0.1 mask:255.255.255.128
        inet6 addr: 2001:DB8:0:0:0:0:0:0/32 mask: ffff:ffff:ffff:ffff::
        ethernet MAC: 00:1a:64:88:a0:6c autoneg:on duplex:Full port:TP
        speed:100Mbps
        statistics collisions:0 multicast:13 rx_bytes:101069093
        rx_compressed:0 rx_crc_errors:0 rx_dropped:0 rx_errors:0
        rx_fifo_errors:0 rx_frame_errors:0 rx_length_errors:0
        rx_missed_errors:0 rx_over_errors:0 rx_packets:1515625
        tx_aborted_errors:0 tx_bytes:10045272 tx_carrier_errors:0
        tx_compressed:0 tx_dropped:0 tx_errors:0 tx_fifo_errors:0
        tx_heartbeat_errors:0 tx_packets:20104 tx_window_errors:0
```

タスクの結果

これらのステップを完了すると、アプライアンス上のイーサネット・インターフェースに関する可用性状況の詳細の検討は完了です。

親トピック: [トラブルシューティング](#)

アプライアンスからのアウトバウンド接続の検査

アウトバウンド接続機能を使用すると、アプライアンスからネットワーク・アドレスに到達可能かどうかを確認できます。



始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

アウトバウンド接続機能は、アプライアンスからターゲットのネットワーク・アドレスに到達可能であるかを確認することによって、問題を切り分けるためのトラブルシューティング作業を支援します。このツールは、あらゆる問題をデバッグする際に適したものではありませんが、ターゲット・ネットワーク・アドレスが使用可能であること、およびそのネットワーク・アドレスへの接続がファイアウォールまたはネットワーク問題によって妨害されていないことを確認するための便利な方法として機能します。

手順

1. 「アプライアンス」 > 「トラブルシューティング」にナビゲートします。
2. 「アウトバウンド接続」を展開します。
3. IP アドレスまたは完全修飾ホスト名を入力します。このフィールドに入力されたネットワーク・アドレスは、ping コマンドの発行時にターゲット・アドレスとして使用されます。
4. 指定したネットワーク・アドレスに対して ping コマンドを実行するには、「**ping**」をクリックします。
 - 入力したネットワーク・アドレスへの接続試行が成功すると、が表示されます。
 - 入力したネットワーク・アドレスへの接続試行が成功しなかった場合は、が表示されます。

タスクの結果

上記の手順が正常に完了すると、ターゲット・アドレスが現在、アプライアンスからネットワーク経由で到達可能かどうか判別されます。

次のタスク

引き続き、発生している問題の[トラブルシューティング](#)を行います。

親トピック: [トラブルシューティング](#)

コマンド行インターフェースを使用したアプライアンスでの操作の実行

コマンドを実行して、アプライアンス・ハードウェアを再始動したり、アプライアンスを元の工場出荷時設定に再設定したり、アプライアンスをシャットダウンしたりすることができます。

このタスクについて

コマンド行インターフェースを使用して、アプライアンス構成データを再設定したり、アプライアンスを再始動したり、アプライアンスをシャットダウンしたりすることができます。

手順

1. xcadmin ユーザーを使用して、アプライアンスとの接続を確立します。 シリアル接続を使用するか、セキュア・シェル (SSH) でコマンド行インターフェースに接続することができます。
 - **シリアル接続:** ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC を、アプライアンスのシリアル・ポートにシリアル接続する必要があります。 シリアル・コンソールとして PC を使用している場合は、Windows 用または Linux 用の PC ベースのシリアル通信プログラムを使用する必要があります。 提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して、アプライアンスに接続します。

重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。
 - **リモート・ログイン:** SSH でアプライアンスに接続するには、アプライアンスの URL を SSH クライアントに指定します。

2. アプライアンスとの接続を確立したら、コマンドを実行することができます。

2.5+ 実行できるコマンドについては、[CLI コマンド・リファレンス \(CLI command reference\)](#) を参照してください。

次のタスク

アプライアンスを再設定した場合には、アプライアンスを再初期化できます。 アプライアンスの構成について詳しくは、『[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)』を参照してください。

親トピック: [トラブルシューティング](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[タスクによるアクティビティのモニター](#)

デッドロックのトラブルシューティング

以下のセクションでは、いくつかの最も一般的なデッドロック・シナリオを説明し、その回避方法を提示します。

始める前に

アプリケーションに例外処理を実装します。詳しくは、[ロック・シナリオでの例外処理の実装 \(Java アプリケーション\)](#) および [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#) を参照してください。

結果、次の例外が表示されます。

```
com.ibm.websphere.objectgrid.plugins.LockDeadlockException: Message
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるストリングを表します。

手順

- **問題:** LockTimeoutException 例外が発生する。

説明: トランザクションまたはクライアントが特定のマップ・エントリーに対するロックの付与を求めると、その要求は、現在のクライアントがロックを解放するまで待機させられ、その後、要求が実行依頼されることがしばしばあります。ロック要求が長い期間アイドル状態のままになり、いつまでもロックが付与されない場合、デッドロックを回避するために LockTimeoutException 例外が作成されます。デッドロックについては、次のセクションで詳しく説明します。ペシミスティック・ロック・ストラテジーを構成すると、ロックはトランザクションがコミットするまで解放されないため、この例外が発生する可能性がより高くなります。

詳細の取得

- **Java** LockTimeoutException 例外は、ストリングを返す getLockRequestQueueDetails メソッドを含んでいます。このメソッドを使用して、例外のトリガーとなった状態についての詳細説明を確認できます。以下に、例外をキャッチして、エラー・メッセージを表示するサンプル・コードを示します。

```
try {
    ...
}
catch (LockTimeoutException lte) {
    System.out.println(lte.getLockRequestQueueDetails());
}
```

ObjectGridException 例外 catch ブロック内で例外を受け取る場合、次のコードのように例外を判定し、キューの詳細を表示できます。コードでは、findRootCause ユーティリティー・メソッドも使用します。

```
try {
    ...
}
catch (ObjectGridException oe) {
    Throwable Root = findRootCause( oe );
    if (Root instanceof LockTimeoutException) {
        LockTimeoutException lte = (LockTimeoutException)Root;
        System.out.println(lte.getLockRequestQueueDetails());
    }
}
```

- **.NET** LockTimeoutException 例外は、ストリングを返す getMessage メソッドを含んでいます。このメソッドを使用して、例外のトリガーとなった状態についての詳細説明を確認できます。

解決策: LockTimeoutException 例外を使用して、アプリケーションでデッドロックが発生する可能性を回避できます。このタイプの例外は、例外が一定時間待機すると発生します。ロック・タイムアウト値を設定することで、例外が待機する時間を設定できます。アプリケーションで実際にはデッドロックが発生していない場合は、ロック・タイムアウトを調整して、LockTimeoutException を回避してください。

ロック・タイムアウト値をプログラマチックに構成してください。

- **Java** [Java アプリケーションにおけるロックの構成と実装](#)
- **.NET** [.NET アプリケーションにおけるロックの構成と実装](#)

- **問題:** 単一キーでデッドロックが発生する。

説明: 以下のシナリオでは、S ロックを使用して単一キーにアクセスし、その後、そのキーを更新するときにデッドロックがどのように発生するかを示しています。このアクションが 2 つのトランザクションから同時に実行されたとき、デッドロックが発生します。

表 1. 単一キーのデッドロックのシナリオ

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	get key1	get key1	key1 に対して S ロックが両方のトランザクションに認可されます。
3	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> Put key1 		U ロックはありません。更新はトランザクション・キャッシュで実行されます。
4		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> put key1 	U ロックはありません。更新はトランザクション・キャッシュで実行されます。
5	トランザクションのコミット		ブロックされます。スレッド 2 が S ロックを保有しているため、key1 に対する S ロックは X ロックにアップグレードできません。
6		トランザクションのコミット	デッドロック: T1 が S ロックを保有しているため、key1 に対する S ロックは X ロックにアップグレードできません。

表 2. 単一キーのデッドロック (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	get key1		key1 に対して S ロックが認可されます。
3	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	get key1	key1 に対して S ロックが U ロックにアップグレードされます。
4		get key1	key1 に対して S ロックが認可されます。
5		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	ブロックされます。T1 が既に U ロックを保有しています。
6	トランザクションのコミット		デッドロック: key1 に対する U ロックはアップグレードできません。
7		トランザクションのコミット	デッドロック: key1 に対する S ロックはアップグレードできません。

表 3. 単一キーのデッドロック (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。

2	get key1		key1 に対して S ロックが認可されます。
3	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 		key1 に対して S ロックが U ロックにアップグレードされます。
4		get key1	key1 に対して S ロックが認可されます。
5		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	ブロックされます。スレッド 1 が既に U ロックを保有しています。
6	トランザクションのコミット		デッドロック: スレッド 2 が S ロックを保有しているため、key1 に対する U ロックは X ロックにアップグレードできません。

ObjectMap.getForUpdate を使用して S ロックを回避すれば、デッドロックは回避されます。

表 4. 単一キーのデッドロック (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 		key1 のスレッド 1 に対して U ロックが認可されます。
3		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	U ロック要求がブロックされます。
4	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> Put key1 	<blocked>	
5	トランザクションのコミット	<blocked>	key1 に対する U ロックは正常に X ロックにアップグレードできます。
6		<released>	スレッド 2 に対して U ロックが最終的に key1 に認可されます。
7		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key2 ◦ <code>.NET</code> Put key2 	key2 に対して U ロックがスレッド 2 に認可されます。
8		トランザクションのコミット	key1 に対する U ロックは正常に X ロックにアップグレードできます。

解決策:

- get メソッドではなく getForUpdate または GetAndLock メソッドを使用し、S ロックではなく U ロックを獲得します。
 - 読み取りコミット済みのトランザクション分離レベルを使用し、S ロックの保有を回避します。トランザクション分離レベルを下げると、非反復可能読み取りの可能性が増します。しかし、いずれかのクライアントからの非反復可能読み取りが可能になるのは、同じクライアントによってトランザクション・キャッシュが明示的に無効化された場合に限られます。
 - **Java** オプティミスティック・ロック・ストラテジーを使用します。オプティミスティック・ロック・ストラテジーを使用するには、オプティミスティック競合例外を処理する必要があります。(Java アプリケーションのみ)
- **問題:** 順序付けられた複数のキーでデッドロックが発生する。

説明: このシナリオでは、2 つのトランザクションが同一エントリを直接更新しようとしたときに、他のエントリに対して S ロックを保有しているとうなるかを説明します。

表 5. 順序付けされた複数のキーのデッドロックのシナリオ

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	get key1	get key1	key1 に対して S ロックが両方のトランザクションに認可されます。
3	get key2	get key2	key2 に対して S ロックが両方のトランザクションに認可されます。
4	次のいずれか 1 つ: ◦ Java update key1 ◦ .NET Put key1		U ロックはありません。更新はトランザクション・キャッシュで実行されます。
5		次のいずれか 1 つ: ◦ Java update key2 ◦ .NET Put key2	U ロックはありません。更新はトランザクション・キャッシュで実行されます。
6	トランザクションのコミット		ブロックされます。スレッド 2 が S ロックを保有しているため、key1 に対する S ロックは X ロックにアップグレードできません。
7		トランザクションのコミット	デッドロック: スレッド 1 が S ロックを保有しているため、key2 に対する S ロックはアップグレードできません。

ObjectMap.getForUpdate メソッドを使用して、S ロックを回避すれば、デッドロックを回避できます。

表 6. 順序付けされた複数のキーのデッドロックのシナリオ (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	次のいずれか 1 つ: ◦ Java getForUpdate key1 ◦ .NET GetAndLock key1		key1 に対して U ロックがトランザクション T1 に認可されます。
3		次のいずれか 1 つ: ◦ Java getForUpdate	U ロック要求がブロックされます。

		<ul style="list-style-type: none"> ◦ <code>key1</code> ◦ <code>.NET</code> GetAndLock key1 	
4	get key2	<blocked>	key2 に対して S ロックが T1 に認可されます。
5	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> Put key1 	<blocked>	
6	トランザクションのコミット	<blocked>	key1 に対する U ロックは正常に X ロックにアップグレードできます。
7		<released>	T2 に対して U ロックが最終的に key1 に認可されます。
8		get key2	key2 に対して S ロックが T2 に認可されます。
9		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key2 ◦ <code>.NET</code> Put key2 	key2 に対して U ロックが T2 に認可されます。
10		トランザクションのコミット	key1 に対する U ロックは正常に X ロックにアップグレードできます。

解決策:

- get メソッドではなく getForUpdate または GetAndLock メソッドを使用して、最初のキーに対する U ロックを直接取得します。この戦略が機能するのは、メソッド順序が決定論的な場合に限られます。
 - 読み取りコミット済みのトランザクション分離レベルを使用し、S ロックの保有を回避します。この解決策は、メソッド順序が決定論的でない場合に、最も簡単に実装できます。トランザクション分離レベルを下げると、非反復可能読み取りの可能性が増します。しかし、非反復可能読み取りが起こりうるのは、トランザクション・キャッシュが明示的に無効化された場合に限られます。
 - `Java` オプティミスティック・ロック・ストラテジーを使用します。オプティミスティック・ロック・ストラテジーを使用するには、オプティミスティック競合例外を処理する必要があります。(Java アプリケーションのみ)
- **問題:** 順序付けられていない U ロックからデッドロックが発生する。

説明: キーが要求される順序が保証できない場合でも、デッドロックは起きる可能性があります。

表 7. U ロックで順序付けがないシナリオ

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key2 ◦ <code>.NET</code> GetAndLock key2 	key1 と key2 に対して U ロックが正常に認可されます。
3	get key2	get key1	key1 と key2 に対して S ロックが認可されます。
4	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> Put key1 	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key2 ◦ <code>.NET</code> Put key2 	
5	トランザクションのコミット		T2 が S ロックを保有しているため、U ロックは X ロックにアップグレードできません。
6		トランザクション	T1 が S ロックを保有しているため、U ロック

解決策:

- すべての作業を単一のグローバル U ロックでラップします (mutex)。この方法は、並行性を低下させますが、アクセスおよび順序が決定論的でない場合に、すべてのシナリオを処理できます。
- 読み取りコミット済みのトランザクション分離レベルを使用し、S ロックの保有を回避します。この解決策は、メソッド順序が決定論的でない場合に、最も簡単に実装でき、最大の並行性を提供します。トランザクション分離レベルを下げると、非反復可能読み取りの可能性が増します。しかし、非反復可能読み取りが起こりうるのは、トランザクション・キャッシュが明示的に無効化された場合に限られます。
- **Java** オプティミスティック・ロック・ストラテジーを使用します。オプティミスティック・ロック・ストラテジーを使用するには、オプティミスティック競合例外を処理する必要があります。(Java アプリケーションのみ)

親トピック: [トラブルシューティング](#)

マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング

2.5+ 説明するシナリオは、ロック・タイムアウト例外を引き起こしているマルチ区画トランザクションの例です。トランザクションの状態に応じて、解決策で、この問題を手動で解決する方法を説明します。

始める前に

アプリケーションに例外処理を実装します。詳しくは、[ロック・シナリオでの例外処理の実装 \(Java アプリケーション\)](#) を参照してください。

結果、次の例外が表示されます。

```
Caused by: com.ibm.websphere.objectgrid.LockTimeoutException:
Local-40000139-DEF8-05EA-E000-64A856931719 timed out waiting
for lock mode S to be granted for map name: TS2_MapP, key: key12
granted = X
lock request queue
->[WXS-40000139-DEF6-FA84-E000-1CB456931719, state = Granted, requested
73423 milli-seconds ago, marked to keep current mode false,
snapshot mode 0, mode = X, thread name = xIOReplicationWorkerThreadPool : 29]
->[Local-40000139-DEF8-05EA-E000-64A856931719, state
= Waiting for 5000 milli-seconds, marked to keep current mode false,
snapshot mode 0, mode = S, thread name = xIOWorkerThreadPool : 28]
dump of all locks for WXS-40000139-DEF6-FA84-E000-1CB456931719
Key: key12, map: TS2_MapP
strongest currently granted mode for key is X
->[WXS-40000139-DEF6-FA84-E000-1CB456931719, state = Granted,
requested 73423 milli-seconds ago, marked to keep current mode false,
snapshot mode 0, mode = X, thread name = xIOReplicationWorkerThreadPool : 29]
dump of all locks for Local-40000139-DEF8-05EA-E000-64A856931719
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるストリングを表します。

手順

問題: ロック・タイムアウト例外が発生し、ロックの所有者がマルチ区画トランザクションである。または、ログ・フォルダーがログ・メッセージで増大している。

診断:

以下のようなログ・メッセージが繰り返し現れ、ログ・フォルダーが満たされています。

```
00000099 TransactionLog I CW0BJ8705I:

Automatic resolution of transaction
WXS-40000139-DF01-216D-E002-1CB456931719
at RM:TestGrid:TestSet2:20 is still waiting for a decision.
Another attempt to resolve the transaction will occur in 30 seconds.
```

ロックを引き起こしているトランザクションのタイプを判別します。トランザクション ID のプレフィックスが WXS- の場合は、マルチ区画トランザクションです。トランザクション ID のプレフィックスが Local- の場合は、トランザクションは単一区画トランザクションです。

原因: 恐らく、コミットまたはロールバックが行われなかったため、アプリケーションがロックを保持しています。

解決策: トランザクションの状態およびその状態が続いている期間を判別します。オプション `-d` (詳細出力) を指定してコマンド・ユーティリティ `xscmd -c listindoubts` を使用するか、トランザクション MBean を使用します。

2.5+ [ロック・タイムアウト例外の解決](#)

`xscmd -c listindoubt` コマンドを使用して、トランザクションの状態を表示して、アクションの方針を判別できます。

親トピック: [トラブルシューティング](#)

関連概念:

関連タスク:

[Java](#) [ロック・タイムアウト例外の解決](#)

ロック・タイムアウト例外の解決

2.5+ `xscmd -c listindoubt` コマンドを使用して、トランザクションの状態を表示して、アクションの方針を判別できます。

親トピック: **2.5+** [マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

関連概念:

[ロック・ストラテジー](#)

[2 フェーズ・コミットとエラー・リカバリー](#)

関連タスク:

[マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

`xscmd -c listindoubts` コマンドを使用したロック・タイムアウト例外の解決

手順

- 以下のコマンドで、環境内のトランザクションの詳細リストを表示します。 `xscmd -c listindoubt -d`
- 適切なアクションを実行して、トランザクションを解決します。 **問題:** トランザクションは、TM でコミット済みとマークを付けられますが、RM は未確定です。

```
[1] WXS-40000139-DEF8-EF60-E002-1CB456931719
Timestamp          Partition  Role  State      Container      Resync
Attempts
-----
--
2012-09-19 10:40:19.824 TestSet1:11  TM    COMMIT    MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.824 TestSet1:7   RM    PREPARED  MPTBasic0_C-1 Primary  0
2012-09-19 10:40:19.839 TestSet2:20  RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.824 TestSet2:6   RM    PREPARED  MPTBasic0_C-1 Primary  0
```

解決策: リソース・マネージャー (**RM**) 区画をコミットしてから、トランザクションを無視します。

- 以下のコマンドを発行してトランザクション WXS-40000139-DEF8-EF60-E002-1CB456931719 の RM 区画をコミットします。 `xscmd -c listIndoubts -xid WXS-40000139-DEF8-EF60-E002-1CB456931719 -cm -rm`
- 以下のコマンドを発行して、このトランザクションを無視します。 `xscmd -c listIndoubts -xid WXS-40000139-DEF8-EF60-E002-1CB456931719 -f`

問題: トランザクションは、すべての区画で未確定です。

```
[1] WXS-40000139-DEF6-FA84-E000-1CB456931719
Timestamp          Partition  Role  State      Container      Resync
Attempts
-----
--
2012-09-19 10:38:11.603 TestSet1:10  RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:38:11.588 TestSet1:5   TM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:38:11.603 TestSet2:11  RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:38:11.619 TestSet2:13  RM    PREPARED  MPTBasic2_C-0 Primary  0
```

解決策: **TM** 区画をまずロールバックしてから、後続の **RM** 区画をロールバックします。次に、トランザクションを無視します。

- 以下のコマンドを発行してトランザクション WXS-40000139-DEF6-FA84-E000-1CB456931719 の TM 区画をロールバックします。 `xscmd -c listIndoubts -xid WXS-40000139-DEF6-FA84-E000-1CB456931719 -r -tm`
- 以下のコマンドを発行してこのトランザクションの RM 区画をロールバックします。 `xscmd -c listIndoubts -xid WXS-40000139-DEF6-FA84-E000-1CB456931719 -r -rm`
- 以下のコマンドを発行してこのトランザクションを無視します。 `xscmd -c listIndoubts -xid WXS-40000139-DEF6-FA84-E000-1CB456931719 -f`

問題: トランザクションは、すべての RM 区画で未確定ですが、トランザクションの決定は、TM では不明です。

```
[1] WXS-40000139-DEF8-EF31-E000-1CB456931719
Timestamp          Partition  Role  State      Container      Resync  Attempts
-----
2012-09-19 10:40:19.777 TestSet1:11  RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.792 TestSet2:5   RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.777 TestSet2:6   RM    PREPARED  MPTBasic2_C-1 Primary  0
```

解決策: **RM** 区画をロールバックします。

- 以下のコマンドを発行してトランザクション WXS-40000139-DEF8-EF31-E000-1CB456931719 の RM 区画をロールバックします。xscmd -c listIndoubts -xid WXS-40000139-DEF8-EF31-E000-1CB456931719 -r

キャッシュ統合のトラブルシューティング

HTTP セッションや動的キャッシュ構成など、キャッシュ統合構成の問題をトラブルシューティングする場合、この情報を使用してください。

手順

- **問題:** HTTP セッション ID が再使用されない。

原因: セッション ID は再使用できます。セッション・パーシスタンスのデータ・グリッドをバージョン 7.1.1 以上で作成すると、セッション ID の再使用は自動的に有効になります。しかし、それより前の構成で作成した場合、この設定が既に誤った値で設定されている可能性があります。

解決策: 次の設定をチェックして、HTTP セッション ID の再使用が有効であるか確認します。

- splicer.properties ファイルの reuseSessionId プロパティは、true に設定する必要があります。
- HttpSessionIdReuse カスタム・プロパティ値は、true に設定する必要があります。このカスタム・プロパティは、WebSphere® Application Server 管理コンソールの次のいずれかのパスで設定されている可能性があります。
 - 「サーバー」 > 「*server_name*」 > 「セッション管理」 > 「カスタム・プロパティ」
 - 「動的クラスター」 > 「*dynamic_cluster_name*」 > 「サーバー・テンプレート」 > 「セッション管理」 > 「カスタム・プロパティ」
 - 「サーバー」 > 「サーバー・タイプ」 > 「WebSphere Application Server」 > 「*server_name*」を選択してから、「サーバー・インフラストラクチャー」セクションの下で次の順にクリックします。「Java およびプロセス管理」 > 「プロセス定義」 > 「Java 仮想マシン」 > 「カスタム・プロパティ」
 - 「サーバー」 > 「サーバー・タイプ」 > 「WebSphere Application Server」 > 「*server_name*」 > 「Web コンテナ設定」 > 「Web コンテナ」

カスタム・プロパティ値を更新した場合、eXtreme Scale セッション管理を再構成し、splicer.properties ファイルが変更を認識できるようにしてください。

- **問題:** データ・グリッドを使用して HTTP セッションを保管する際、トランザクションの負荷が高いと SystemOut.log ファイルに CWOBJ0006W メッセージが出力される。

```
CWOBJ0006W: 例外が発生しました:  
com.ibm.websphere.objectgrid.ObjectGridRuntimeException:  
java.util.ConcurrentModificationException
```

このメッセージは、Web アプリケーションが HttpSession の属性として設定された List オブジェクトを変更した場合にのみ発生します。

解決策: 変更された List オブジェクトを含んでいる属性を複製し、複製した属性をセッション・オブジェクトに設定します。

- **問題:** Web アプリケーションを Servlet 3.0 仕様で実行しているとき、Web アプリケーションのフィルターとリスナーが WebSphere eXtreme Scale セッション管理によって起動されません。例えば、WebSphere eXtreme Scale でリモート・コンテナ除去を使用してセッションが無効化されると、リスナーがコールバックされません。

原因: WebSphere eXtreme Scale は、注釈を使用して定義されたりプログラマチックに定義されたりしたフィルターやリスナーを特定しません。

解決策: フィルターやリスナーは、Web アプリケーションの web.xml ファイルで明示的に宣言する必要があります。

親トピック: [トラブルシューティング](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

インストールのトラブルシューティング

インストールおよび更新の問題をトラブルシューティングする場合、この情報を使用してください。

手順

- **問題:** インストール・コマンドをリモート・コンピューター (¥¥mymachine¥downloads¥ など) から実行すると、次のメッセージが表示される: 現在のディレクトリとして上記のパスで CMD.EXE を開始しました。UNC パスはサポートされません。 Windows ディレクトリを既定で使用します。結果として、インストールが正常に完了しない。

解決策: リモート・コンピューターをネットワーク・ドライブにマップしてください。例えば、Windows の場合、「マイ コンピュータ」を右クリックし、「ネットワーク ドライブの割り当て」を選択して、リモート・コンピューターへの 汎用名前付け規則 (UNC) パスを組み込むことができます。その後は、ネットワーク・ドライブから正常にインストール・スクリプトを実行できます。例: y:¥mymachine¥downloads¥WXS¥install.bat。

- **問題:** インストールが正常に完了しない。

解決策: ログ・ファイルをチェックして、インストールがどこで失敗したか 確認します。インストールが正常に完了しない場合は、ログは [wxs_install_root/logs/wxs](#) ディレクトリにあります。

- **問題:** インストール時に重大な障害が発生する。

解決策: ログ・ファイルをチェックして、インストールがどこで失敗したか 確認します。インストールが部分的に完了した時点でインストールが失敗した場合、ログは 一般的に `user_root/wxs_install_logs/` ディレクトリにあります。

- **Windows** **問題:** Windows 上で WebSphere® eXtreme Scale クライアント をインストールする場合、インストールの結果に 次のテキストが表示されることがある。

```
Success: The installation of the following product was successful:
WebSphere eXtreme Scale Client. Some configuration steps have errors.
For more information, refer to the following log file:
<WebSphere Application Server install root>¥logs¥wxs_client¥install¥log.txt"
Review the installation log (log.txt) and review the deployment manager
augmentation log.
```

解決策: `iscdeploy.sh` ファイルで発生した障害の場合、エラーは 無視してかまいません。このエラーでは、問題は生じません。

- **Linux** **問題:**

フルインストールの環境で、WebSphere eXtreme Scale クライアント に対してアップデート・インストーラーを使用して保守のみを適用しようとする、以下のメッセージが表示されます。

```
Prerequisite checking has failed. Click Back to select a different package,
or click Cancel to exit.
```

```
Failure messages are:
```

```
Required feature wxs.client.primary is not found.
```

WebSphere eXtreme Scale クライアント がインストールされていて、アップデート・インストーラーを使用してすべての保守パッケージを適用しようとする、次のメッセージが表示されます。

```
Prerequisite checking has failed. Click Back to select a different package,
or click Cancel to exit.
```

```
Failure messages are:
```

```
Required feature wxs.primary is not found.
```

解決策: インストールする保守パッケージはインストール済み環境のタイプと一致している必要があります。ご使用のインストール済み環境のタイプに適合する保守パッケージをダウンロードし、適用してください。

- **Linux** **問題:** インストールがハングします。

解決策: 非 root ユーザーで Linux に WebSphere eXtreme Scale をインストールしている時に、インストーラーがハングする場合があります。これは、オープン・ファイルの最大数が Linux オペレーティング・システム上であまりに低く設定されていることが原因と考えられます。 `/etc/limits.conf` ファイルまたは `/etc/security/limits.conf` ファイル (ファイルの場所は、個別の Linux 配布版によって異なります) 内で許

可される限度を少なくとも 8192 まで上げる必要があります。

親トピック: [トラブルシューティング](#)

関連タスク:

[Update Installer を使用して保守パッケージをインストールする](#)

管理のトラブルシューティング

サーバーの開始や停止、**xscmd** ユーティリティの使用など、管理についてのトラブルシューティングを行う場合、この情報を使用してください。

手順

- **問題:** **xscmd** コマンドの実行時に次のメッセージが画面に表示される。

```
java.lang.IllegalStateException: Placement service MBean not available.
[]
    at
com.ibm.websphere.samples.objectgrid.admin.OGAdmin.main(OGAdmin.java:1449)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:60)
    at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:37)
)
    at java.lang.reflect.Method.invoke(Method.java:611)
    at com.ibm.ws.bootstrap.WSLauncher.main(WSLauncher.java:267)
Ending at: 2011-11-10 18:13:00.000000484
```

原因: カタログ・サーバーで接続の問題が発生しました。

解決策: カタログ・サーバーが実行中であり、ネットワーク経由で使用可能なことを確認します。このメッセージは、カタログ・サービス・ドメインが定義されている場合に、実行中のカタログ・サーバーが2つより少ないとき発生することもあります。そのような環境は、2つのカタログ・サーバーが開始されるまで使用できません。

- **問題:** **xscmd** コマンドの実行時に次のメッセージが画面に表示される。

```
CWXSI0066E: 一致しない引数 argument_name が検出されました。
```

原因: 入力されたコマンド・フォーマットは **xscmd** ユーティリティによって認識されませんでした。

解決策: コマンドのフォーマットを確認してください。 **-c findbyKey** コマンドを使用して正規表現を実行するときに、この問題が発生する可能性があります。詳しくは、[データの照会、表示、および無効化](#)を参照してください。

親トピック: [トラブルシューティング](#)

参照

参照情報は、特定の項目を迅速に見つけることができるように編成されています。

[xscmd ユーティリティー・リファレンス](#)

xscmd ユーティリティーを使用する場合、次のコマンドのリストをリファレンスとして使用できます。

2.5+ [HTTP コマンド・インターフェースのリファレンス](#)

[クライアント・プロパティ・ファイル](#)

WebSphere® eXtreme Scale クライアント・プロセスの要件に基づいて、プロパティ・ファイルを作成します。

[正規表現の構文](#)

正規表現は、他のストリングとのマッチングに使用される構造化ストリングです。正規表現は、データを無効化する場合およびメッセージに表示されるメッセージをフィルタリングする場合に使用することができます。

[動的マップの構成オプション](#)

ご使用のクライアント・アプリケーションを特定の名前の付いたマップに接続させることで、データ・グリッド内に追加のマップを作成できます。この接続が発生すると、マップが自動的に作成されます。

[ユーザー・インターフェース設定](#)

この参照情報では、WebSphere Application Server 管理コンソールなどのページで表示および構成できる設定について説明します。

xscmd ユーティリティー・リファレンス

xscmd ユーティリティーを使用する場合、次のコマンドのリストをリファレンスとして使用できます。

汎用コマンド・パラメーター

xscmd ユーティリティー・コマンドの一般的な形式は次のとおりです。オプション・パラメーターは、大括弧 [] で囲まれています。

注意:

WebSphere® DataPower® XC10 アプライアンス 環境では、次のコマンドは使用しないでください。

- **-c releaseShard**
- **-c reserveShard**
- **-c swapShardWithPrimary**
- **-c suspendBalancing**
- **-c resumeBalancing**
- **-c teardown**
- **-c triggerPlacement**
- **-c enableForPlacement**

```
Usage: xscmd -c <cmdName> | -h <cmdName> | -lc [<cmdGroupName>] | -lcg
      [-tt <type>] [-prot <protocol>] [-trf <filePath>] [-ks
      <filePath>] [-ksp <password>] [-user <username>] [-al <alias>]
      [-cgp <property>] [-kst <type>] [-cep <endpoints>] [-ssp
      <profileName>] [-tsp <password>] [-arc <integer>] [-trs
      <traceSpec>] [-tst <type>] [-to <serverTimeout>] [-cxpv
      <provider>] [-sp <profileName>] [-pwd <password>] [-ca <support>]
      [-cgc <className>] [-ts <filePath>]
```

Options:

- | | |
|--------------------------------------|--|
| -al, --alias <alias> | Alias name in the keystore. |
| -arc, --authRetryCount <integer> | The retry count for authentication if the credential is expired. If the value is set to 0, then authentication retries do not occur. |
| -c, --command <cmdName> | Specifies the name of the command to execute |
| -ca, --credAuth <support> | Set the client credential authentication support [Never, Supported, Required]. |
| -cep, --catalogEndPoints <endpoints> | Specifies one or more catalog service endpoints in the format <host>[:<listenerPort>][,<host>[:<listenerPort>]]. Default endpoint: localhost:2809 |
| -cgc, --credGenClass <className> | Specifies the name of the class that implements the CredentialGenerator interface. This class is used to get credentials for clients. |
| -cgp, --credGenProps <property> | Specifies the properties for the CredentialGenerator implementation class. The properties are set to the object with the setProperties(String) method. |
| -cxpv, --contextProvider <provider> | Context provider. Examples: IBMJSSE2, IBMJSSE, IBMJSSEFIPS. |
| -h, --help <cmdName> | Invokes general command-line help |
| -ks, --keyStore <filePath> | Absolute path to keystore. Example: /etc/test/security/server.public |

-ksp,--keyStorePassword <password>	Specifies the password to the keystore.
-kst,--keyStoreType <type>	Keystore type. Example: JKS, JCEK, PKCS12.
-lc,--listCommands <cmdGroupName>	List all commands for a command group
-lcg,--listCommandGroups	List all command groups
-lpc,--listPrivateCommands	List all private commands.
-prot,--protocol <protocol>	Protocol. Examples: SSL, SSLv2, SSLv3, TLS, TLSv1.
-pwd,--password <password>	eXtreme Scale password security credential
-sp,--secProfile <profileName>	Specifies a profile name.
-ssp,--saveSecProfile <profileName>	Save security parameter values in security profile.
-to,--timeout <serverTimeout>	Server connection timeout in seconds
-trf,--traceFile <filePath>	Specifies the absolute path to the generated trace file for xscmd command output
-trs,--traceSpec <traceSpec>	Specifies the trace specification for xscmd command output
-ts,--trustStore <filePath>	Absolute path to truststore. Example: /etc/test/security/server.public
-tsp,--trustStorePassword <password>	Truststore password
-tst,--trustStoreType <type>	Truststore type. Examples: JKS, JCEK, PKCS12.
-tt,--transportType <type>	Transport layer security configuration type. Examples: TCP/IP, SSL-Supported, SSL-Required.
-user,--username <username>	eXtreme Scale user name security credential

すべてのコマンド

次に、**xscmd** ユーティリティ・コマンドをすべてリストします。

注: 列名は変わることがあります (列のタイプと位置は変わりません)。また、コマンドが拡張された場合、必要に応じて新しい列が表の最後に追加されることがあります。

特定のコマンドの詳細およびパラメーターを表示するには、**xscmd -h command_name** を実行してください。これらのコマンドを使用してカスタマイズ・スクリプトを作成する計画の場合は、**./xscmd.sh -c commandName -Options 2> /dev/null** を実行して、標準エラー・パスをヌル・デバイスに指定変更してください。

例

以下の例は、**xscmd** ユーティリティからコマンドを実行する方法を示しています。

```
./xscmd.sh -lc (このコマンドにより、使用可能なコマンドがすべてリストされます。)
```

```
./xscmd.sh -h showMapSizes (このコマンドにより、showMapSizes コマンドのヘルプがリストされます。)
```

重要: 以下のコマンドの出力と使用法については、将来変更される可能性があります。コマンドにアスタリスク (*) が付いている場合は、そのコマンドの出力のみ、変更の可能性があることを意味しています。

- listHosts
- showPlacement
- placementServiceStatus

- showDomainReplicationState
- showReplicationState
- *osgiAll
- *osgiCheck
- *osgiCurrent
- *osgiUpdate
- *showLinkedPrimaries
- *triggerPlacement

Command Name	Description
-----	-----
balanceShardTypes	Attempt shard redistribution so that the ratio of primaries and replicas within each container server are within one shard.
balanceStatus	Check the balance status of the data grid for the ObjectGrid and map set specified.
clearGrid	Clears data from the data grid.
dismissLink	Disconnects from the specified catalog service domain.
enableForPlacement	Re-enable shard placement to a container that is disabled for shard placement because of the failure of a previous shard placement operation.
establishLink	Connects to the specified catalog service domain with the specified catalog service endpoints.
findByKey	Finds matching keys in a map.
getCatTraceSpec	Retrieves the trace specification for all the catalog servers that are known by this process.
getNotificationFilter	Displays the notification filters for the servers in the environment.
getStatsSpec	Retrieves the statistics specification.
getTraceSpec	Retrieves the trace specification.
listAllJMXAddresses	Displays all JMX MBean server addresses.
listCoreGroups	List all core groups.
listDisabledForPlacement	List the containers that are disabled for shard placement because shard placement operations failed for them.
listHosts	Lists all hosts. The command usage and output is subject to change in the future.
listIndoubts	List and resolve indoubt transactions.
listObjectGridNames	Lists all known ObjectGrid instances and map sets.
listProfiles	Lists profiles.
listenForNotifications	Subscribes to notifications received by the messaging hub. Prints the errors, warnings and other messages as they are received.
osgiAll	Show all available OSGi service rankings. Use the -sn option to display a single service. The command output only is subject to change in the future.
osgiCheck	Check whether specified OSGi service rankings are

	available. The command output only is subject to change in the future.
osgiCurrent	Show all OSGi service rankings currently in use. Use the -sn option to display a single service. The command output only is subject to change in the future.
osgiUpdate	Update the OSGi services to the specified rankings. The command output only is subject to change in the future.
overrideQuorum	Notify the catalog server to override quorum.
placementServiceStatus	Displays the ObjectGrid placement operation status. The command usage and output is subject to change in the future.
releaseShard	Releases the specified primary shard from the specified container server.
removeProfile	Remove profile from file system.
reserveShard	Reserves the specified primary shard on the specified container server.
resumeBalancing	Attempts to balance and allow future balancing attempts for the specified ObjectGrid instance and map set.
revisions	Displays all known revision history.
routetable	Displays the current routing table.
setCatTraceSpec	Sets the trace specification for all the catalog servers known by this process.
setNotificationFilter	Sets the notification filter. The messaging hub processes INFO, WARNING, and SEVERE messages that match the regular expression.
setStatsSpec	Sets the statistics specification.
setTraceSpec	Trace specification in the form: traceType1=traceLevel1=traceState1[: traceTypeN=traceLevelN=traceStateN]*
showCoreGroupMembers	Shows all core group members.
showDomainReplicationState	Displays the outbound and inbound revisions yet to be replicated between primary shards on linked domains, for each container across all domains. The command usage and output is subject to change in the future.
showInfo	Retrieves the environment specifications including installed versions and JVM information.
showLinkedDomains	Displays the linked foreign domains
showLinkedPrimaries	Displays the primary shards and all of their foreign or domestic linked primary shards. The command output only is subject to change in the future.
showMapSizes	Displays all map sizes.
showNotificationHistory	Displays the most recent environment errors,

	warnings and messages that are stored in the messaging hub.
showPlacement	Lists all container servers and their shards. The command usage and output is subject to change in the future.
showProfile	Show the details of the specified profile.
showQuorumStatus	Displays catalog server quorum status.
showReplicationState	Displays the outbound and inbound revisions yet to be replicated between the primary and replica shards for each container. The command usage and output is subject to change in the future.
showTransport	Displays the transport used by the catalog service domain. Types include ORB and eXtremeIO.
suspendBalancing	Prevents future attempts to balance the specified ObjectGrid instance and map set.
swapShardWithPrimary	Swaps the specified replica shard for the specified container server with its primary shard.
teardown	Stops a list of catalog and container servers.
triggerPlacement	Triggers a placement operation for the specified ObjectGrid instance and map set. The numInitialContainers value is ignored. The command output only is subject to change in the future.

親トピック: [参照](#)

関連タスク:

[xscmd ユーティリティーによる管理](#)

[xscmd ユーティリティーのためのセキュリティー・プロファイルの構成](#)

[データの照会、表示、および無効化](#)

HTTP command interface reference

With the HTTP command interface, you can run operations on your appliance, configure appliance settings, and administer data grids, collectives, and zones.

Table of contents

- [List of APPLIANCE commands](#)
- [List of COLLECTIVE commands](#)
- [List of GRID commands](#)
- [List of TASK commands](#)

List of APPLIANCE commands

CreateAdminTrace

D
e
s
c
r
i
p
t
i
o
n:

Adds or modifies an existing administrative trace string.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**tr
a**

**c
e
N
a
m
e** Specifies the name of the trace.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage** A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult** A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e** Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"AutoCustomLogger", "command": "CreateAdminTrace" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

CreateAggregateInterface

D
e

description:
Required Parameters:

Creates a new Aggregate Interface.

**m
e
m
b
e
r**

Lists all ethernet interface members of this aggregation.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**p
r
i
m
a
r
y
_
m
e
m
b
e
r**

Specifies which of the aggregated ethernet interfaces is the primary one.

r

agg
r
e
g
a
t
i
o
n
-
p
o
l
i
c
y

Specifies the aggregation policy of the interface.

a
d
d
r
e
s
s

Specifies IP address of the interface.

n
a
m
e

Specifies the interface name.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "member": "eth0,eth1", "stopOnTaskFailure": "true",  
"primary_member": "eth0", "aggregation_policy": "active-backup",  
"address": "1:2:3:4/24", "name": "agg1", "command":  
"CreateAggregateInterface" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

CreateDataCacheTrace

D
e
s
c
r
i
p
t
i
o
n:

Adds or modifies an existing data cache trace string.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**F
a
i
l
u
r
e**

**tr
a
c
e
N
a
m
e**

Specifies the name of the trace.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"autoDataCacheLogger", "command": "CreateDataCacheTrace" } }
```

C
o
m
m
a
n
d
T
y
p
e

appliance

p
e:

CreateSNMPCommunity

D
e
s
c
r
i
p
t
i
o
n:

Creates a Simple Network Management Protocol (SNMP) community.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**h
o
s
t
R
e
s
t
r
i
c
t
i
o
n**

(Optional) Specifies an IP address on which to restrict SNMP communication. Communication with any other IP address or subnet is denied.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
c
o
m
m
u
n
i
t
y
N
a
m
e**

Specifies the name of the SNMP community to create.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "hostRestriction": "autorestriction", "stopOnTaskFailure":  
"true", "command": "CreateSNMPCommunity", "communityName":  
"autocommunity" } }
```

**C
o
m
m
a
n
d
T
y
p
e**

appliance

p
e:

DeleteSNMPCommunity

D
e
s
c
r
i
p
t
i
o
n:

Deletes a Simple Network Management Protocol (SNMP) community.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

c
o
m
m
u
n
i
t
y
N
a
m
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies the name of SNMP community to delete.

e

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"DeleteSNMPCommunity", "communityName": "autocommunity" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

EnableCreateAccount

D
e
s
c
r
i
p

Enables a setting that allows users to initiate the creation of their

tion:

own accounts.

Required Parameters:

stopOnTaskFailure
enable

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Enables account creation. Set the value to false to disable account creation. Set the value to true to enable account creation.

Result Parameter

rs
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "enable": "true", "command": "EnableCreateAccount" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

EnablePasswordReset

D
e
s
c
r
i
p
t
i
o
n:

Enables the ability to reset the xadmin password with a serial connection. No other credentials or SMTP messages are required.

R
e
q
u
i
r
e
d
P
a
r
a
m
e

ters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

enable

Enables the password to be reset. Set the value to true to enable password resets.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

E

X
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "enable": "true", "command":  
"EnablePasswordReset" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

EnableSNMP

D
e
s
c
r
i
p
t
i
o
n:

Enables or disables Simple Network Management Protocol (SNMP).

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**il
u
r
e

e
n
a
b
l
e**

Enables SNMP. Set the value to false to disable SNMP. Set the value to true to enable SNMP.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

- errorMessage** A message that explains the reason for the failure, if the command failed.
- status** Specifies the status of the command that was run. The result can be either success or failed.
- commandResult** A JSON-formatted statement that contains the result of the command that was run.
- commandName** Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "stopOnTaskFailure": "true", "enable": "true", "command": "EnableSNMP" } }
```

**C
o
m
m
a
n
d
T
y
p
e:**

appliance

ModifyAdminDefaultTrace

D
e
s
c
r
i
p
t
i
o
n:

Changes the trace level for the administrative default logger.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**tr
a
c
e
L
e
v
e
l**

Specifies the trace level. Valid values: OFF, SEVERE, WARNING, or INFO.

R
e

S
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "traceLevel": "SEVERE",  
"command": "ModifyAdminDefaultTrace" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyAdminTrace

D
e
s
c
r
i
p
t
i
o
n:

Changes an administrative logger trace level.

R
e

q
u
i
r
e
d
p
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**tr
a
c
e
L
e
v
e
l**

Specifies the trace level. Valid values: OFF, SEVERE, WARNING, INFO, FINE, FINER, FINEST, ALL.

**tr
a
c
e
N
a
m
e**

Specifies the name of the trace to modify.

R
e
s
u
l
t
P
a
r
a

parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceLevel": "SEVERE",  
"traceName": "autoCustomLogger", "command": "ModifyAdminTrace"  
} }
```

Command Type:

appliance

ModifyAdministratorEmail

Description:

Changes the default administrator email address.

Required P

a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**a
d
m
i
n
i
s
t
r
a
t
o
r
E
m
a
i
l**

Specifies a new email address for the default administrator account.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "administratorEmail": "somenewname@us.ibm.com", "command": "ModifyAdministratorEmail" } }
```

Command Type:

appliance

ModifyAdministratorName

Description:

Changes the default administrator account name.

Required Parameters:

st

**o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**a
d
m
i
n
i
s
t
r
a
t
o
r
N
a
m
e**

Specifies the new name for the default administrator account.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E

X
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "administratorName":  
"somenewname", "command": "ModifyAdministratorName" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyAdministratorPassword

D
e
s
c
r
i
p
t
i
o
n:

Changes the password for the default administrator account.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**st
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**n
e
w
P
a
s
s
w
o
r
d**

Specifies a new default administrator account password.

**p
a
s
s
w
o
r
d
V
e
r
i
f
i
c
a
t
i
o
n**

Specifies a new default administrator account password, entered a second time for verification.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "newPassword": "somepass",  
"passwordVerification": "somepass", "command":  
"ModifyAdministratorPassword" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyAggregateInterface

D
e
s
c
r
i
p
t
i
o
n:

Modifies Aggregate Interface parameters.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

n

a
m
e Specifies the interface name.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage** A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult** A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e** Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "mtu": "1500", "member": "eth0,eth1",  
"stopOnTaskFailure": "true", "transmit_hash_policy": "layer2",  
"dad_retransmit_timer": "1000", "use_slaac": "true", "mode": "Auto",  
"use_dhcp": "false", "use_arp": "true", "primary_member": "eth0",  
"aggregation_policy": "active-backup", "address": "1:2:3:4/24",  
"name": "agg1", "command": "ModifyAggregateInterface",  
"dad_transmits": "1", "lACP_selection_logic": "stable",  
"ipv4_default_gateway": "1:2:3:5" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyDataCacheTrace

D
e
s

cr
ip
ti
o
n:

Adds or modifies an existing data cache trace string.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**tr
a
c
e
L
e
v
e
l**

Specifies the trace level. Valid values: OFF, SEVERE, WARNING, INFO, FINE, FINER, FINEST, ALL.

**tr
a
c
e
N
a
m
e**

Specifies the name of the trace.

R

Result Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceLevel": "SEVERE",  
"traceName": "autoDataCacheLogger", "command":  
"ModifyDataCacheTrace" } }
```

Command Type:

appliance

ModifyEthernetInterface

Description:

Modifies Ethernet Interface parameters.

R

Required Parameters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

name

Specifies the interface name.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

comma

ndResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "mtu": "1500", "use_dhcp": "false", "use_arp": "true", "stopOnTaskFailure": "true", "dad_retransmit_timer": "1000", "use_slaac": "true", "address": "1:2:3:4/24", "name": "eth0", "dad_transmits": "1", "command": "ModifyEthernetInterface", "ipv4_default_gateway": "1:2:3:5", "mode": "Auto" } }
```

Command Type:

appliance

ModifyLDAP

Description:

Configures the appliance to use Lightweight Directory Access Protocol (LDAP) for user login authentication.

Required Parameters:

jndiSecurityPrincipal Specifies the JNDI security principal.

**nci
pal**

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

ldapUserSearchFilterPattern LDAP user search filter pattern.

newPassword Specifies the new LDAP password.

groupBaseDn Specifies the LDAP JNDI base distinguished name (DN) for groups.

passwordVerification Specifies the new LDAP password a second time for password verification. The value must be the same as the newPassword parameter.

userBaseDn Specifies the LDAP JNDI base distinguished name (DN) for users.

enable Specifies if LDAP user authentication is enabled. Set the value to true to enable LDAP user authentication.

jndiProviderURL Specifies the URL for the LDAP provider.

rs
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "jndiSecurityPrincipal":  
"CN=Administrator,CN=users,DC=mycompany,DC=com",  
"ldapUserIdSearchFilterPattern": "thefilterpattern", "newPassword":  
"somepassword", "passwordVerification": "somepassword",  
"groupBaseDn": "cn=group", "userBaseDn": "cn=user", "enable":  
"true", "command": "ModifyLDAP", "jndiProviderURL": "someurl" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyNtpServers

D
e
s
c
r
i
p
t
i
o
n:

Changes the list of Network Time Protocol (NTP) servers.

R
e
q
u
i
r
e
d
P
a
r
a
m

eters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

listOfServers

Specifies a list of comma-separated Network Time Protocol (NTP) servers.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "listOfServers":  
"time.nist.gov,10.10.2.2", "command": "ModifyNtpServers" } }
```

Command Type:

appliance

ModifySMTPEmail

Description:

Changes the reply-to email address that is used for sending passwords that are reset by users.

Required Parameters:

stopOn

Specifies whether to stop running the batch routine

**T
a
s
k
F
a
i
l
u
r
e** when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
m
a
i
l
A
d
d
r
e
s
s** Specifies the reply-to email address. Use the email address for the administrator.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage** A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult** A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e** Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ModifySMTPEmail", "emailAddress": "admin@mycompany.com" } }
```

C
o

m
m
a
n
d
T
y
p
e:

appliance

ModifySMTPServer

D
e
s
c
r
i
p
t
i
o
n:

Changes the Simple Mail Transfer Protocol (SMTP) server.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

h
o
s
t

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Name Specifies the SMTP host address.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command": "ModifySMTPServer", "hostName": "sicdsjc" } }
```

Command Type:

appliance

ModifySearchFilterUsers

De

S
c
r
i
p
t
i
o
n:

Changes the Lightweight Directory Access Protocol (LDAP) search filter for users.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

ldapUserSearchFilterPattern LDAP user search filter pattern.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true",
"ldapUserIdSearchFilterPattern": "(&(uid={0})
(objectclass=inetOrgPerson))", "command":
"ModifySearchFilterUsers" } }
```

Command Type:

appliance

ModifyTimeZone

Description:

Changes the time zone for the appliance.

Required Parameters:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**l
o
c
a
l
T
i
m
e
Z
o
n
e**

Specifies the new time zone as an abbreviated string. Valid values: EST : Eastern Standard Time
CST : Central Standard Time MST : Mountain Standard Time PST : Pacific Standard Time HAST : Hawaii AKST : Alaska AST : Atlantic UTC : Coordinated Universal Time GMT : Greenwich Mean Time CET : Central Europe EET : Eastern Europe MSK : Moscow AST_ARABIA : Arabia KRT : Pakistan IST : India NOVT : Novosibirsk CST_CHINA : China JST : Japan AWST : Australia West ACST : Australia Central AEST : Australia East

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e**

```
{ "task": { "stopOnTaskFailure": "true", "command":
```

pl "ModifyTimeZone", "localTimeZone": "EST" } }

e:

C
o
m
m
a
n
d
T
y
p
e:

appliance

PingRemoteHost

D
e
s
c
r
i
p
t
i
o
n:

Tests the visibility of a host from this appliance.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

e
h
o
s
t
N
a
m
e

Specifies the IP address or host name of the remote host.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"PingRemoteHost", "hostName": "io03.rtp.raleigh.ibm.com" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

RemoveAdminTrace

Description:

Removes an existing administrative console trace string.

Required Parameters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

traceName

Specifies the name of the trace.

Result:

Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"AutoCustomLogger", "command": "RemoveAdminTrace" } }
```

Command Type:

appliance

RemoveAggregateInterface

Description:

Deletes Aggregate Interface of a given name.

Require

readParameters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

name

Specifies the interface name.

ResultParameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "name": "agg1", "command": "RemoveAggregateInterface" } }
```

Command Type:

appliance

RemoveDataCacheTrace

Description:

Removes an existing data cache trace string.

Required Parameters:

stopOnT

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the

stopOnTaskFailure batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

traceName Specifies the name of the trace.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"autoDataCacheLogger", "command": "RemoveDataCacheTrace" } }
```

Command:

appliance

d
T
y
p
e:

RestartAppliance

D
e
s
c
r
i
p
t
i
o
n:

Restarts or shuts down the appliance.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**i
m
m
e
d
i
a
t
e**

Specifies if the appliance waits to complete active tasks before restarting. If the value is set to true, the appliance restarts immediately. If the value is set to false, the appliance waits for all active tasks to complete before restarting.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
s
h
u
t
d
o
w
n**

Specifies if the appliance restarts or shuts down. If the value is set to true, the appliance is shut down. If the value is set to false, the appliance restarts.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "immediate": "false", "stopOnTaskFailure": "true",  
"command": "RestartAppliance", "shutdown": "true" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ViewAllAggregateInterfaces

D
e
s
c
r
i
p
t
i
o
n:

Displays the information for every aggregate interface.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**S
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t

ers
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ViewAllAggregateInterfaces" } }
```

Command Type:

appliance

ViewAllEthernetInterfaces

Description:

Displays the information for every Ethernet Interface.

Required Parameters

me
te
rs
:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Re
su
lt
Pa
ra
me
te
rs
:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ViewAllEthernetInterfaces" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

List of COLLECTIVE commands

AddApplianceToCollective

D
e
s
c
r
i
p
t
i
o
n:

Adds an existing appliance to the current collective.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**a
p
p
l
i
a
n
c
e
l
P**

Specifies the IP address of the appliance to add to the collective.

**s
t
o
p
O**

n Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

T
a
s
k
F
a
i
l
u
r
e

s
e
c
r
e
t
K
e
y

Specifies the secret key of the appliance.

w
a
i
t
O
n
T
a
s
k

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

z
o
n
e
N
a
m
e

Specifies the name of the zone to which you want to assign the appliance.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorM
essage

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "applianceIP": "192.168.222.128", "stopOnTaskFailure": "true", "secretKey": "P5Xa4F8MQeSxuuhKxnaStw==", "waitOnTask": "true", "command": "AddApplianceToCollective", "zoneName": "DefaultZone" } }
```

Command Type:

collective

AddRoleToGroup

Description:

Assigns an access role to a group. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

Required Parameters:

**r
o
u
p
N
a
m
e**

Specifies the name of the group that is being assigned to a role.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**r
o
l
e**

Specifies the role to assign to the group. Valid values: 2 = appliance administration, 3 = appliance monitoring, 5 = data cache creation.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
X

sample: { "task": { "stopOnTaskFailure": "true", "groupName": "autoGroup1003", "command": "AddRoleToGroup", "role": "5" } }

Command Type: collective

AddRoleToUser

Description: Assigns an access role to a user. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

Required Parameters:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**r
e**

**r
o
l
e**

Specifies the role to assign to the user. Valid values:
2 = appliance administration, 3 = appliance
monitoring, 5 = data cache creation.

**u
s
e
r
N
a
m
e**

Specifies the name of the user that is being
assigned a role.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the
failure, if the command failed.

status

Specifies the status of the command that was
run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the
result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command
interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command": "AddRoleToUser",  
"role": "2", "userName": "autoUser1003" } }
```

C
o
m
m
a
n
d
T

collective

y
p
e:

AddUserToGroup

D
e
s
c
r
i
p
t
i
o
n:

Adds a user to a group. After the user is added to the group, the user has the roles that are assigned to the selected group only. Any individual user roles are lost.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of a defined group.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**u
s
e
r
N
a
m
e**

Specifies the name of a defined user.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName":  
"autoGroup1003", "command": "AddUserToGroup", "userName":  
"autoUser1003" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

CreateCollectiveLink

Description:

Creates a link to another collective.

Required Parameters:

applianceName

Specifies the host name or IP address of an appliance that is a member of another collective.

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

r

**e
m
o
t
e
P
a
s
s
w
o
r
d**

Specifies the password that is used to log in to the remote appliance.

**r
e
m
o
t
e
U
s
e
r
N
a
m
e**

Specifies the user name that is used to log in to the remote appliance.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s**
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
X**

```
{ "task": { "applianceName": "myappliance.mydomain.com",
```

example: "stopOnTaskFailure": "true", "command": "CreateCollectiveLink",
"remotePassword": "somePassword", "remoteUserName": "someUser"
} }

Command
Type:

collective

CreateGroup

Description:

Creates a new group. Groups are useful for assigning multiple users the same set of roles.

Required Parameters:

**group
Name**

Specifies a name for the new group. Group names must be unique.

st

**o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
o
u
p
D
e
s
c
r
i
p
t
i
o
n**

Specifies a description of the group and its users.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
X**

example: { "task": { "stopOnTaskFailure": "true", "groupName": "autoGroup1003", "groupDescription": "this group is huge", "command": "CreateGroup" } }

Command Type: collective

CreateUser

Description: Creates a new user.

Required Parameters:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**as
s
w
o
r
d
V
e
r
i
f
i
c
a
t
i
o
n**

Specifies the password, entered a second time for verification.

**us
er
P
a
s
s
w
o
r
d**

Specifies a new password for the selected user.

**e
m
a
i
l
A
d
d
r
e
s
s**

Specifies an email address for the new user.

**fu
ll
U
s
e
r
N
a
m
e
us
er
N
a
m
e**

Specifies a full name for the user. Example: John E. Doe.

Specifies a short name for the new user. Example: xadmin.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e**

rs
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "passwordVerification":  
"someuserpass", "userPassword": "someuserpass", "command":  
"CreateUser", "emailAddress": "someuser@us.ibm.com",  
"fullUserName": "someuser isme", "userName": "someuser" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

DeleteCollectiveLink

D
e
s
c
r
i
p
t
i
o
n:

Removes the link between the local collective and the specified collective.

R
e
q
u
i
r
e
d
P
a
r
a
m
e

ters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

name

Specifies the name of the collective to delete.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "name": "myCollectiveName",  
"command": "DeleteCollectiveLink" } }
```


e:
C
o
m
m
a
n
d
T
y
p
e:

collective

DeleteGroup

D
e
s
c
r
i
p
t
i
o
n:

Deletes a group. Users that belonged to the group no longer belong to the group and might lose any assigned roles.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

**s
t
o
p
O
n**

Specifies the name of a defined group.

**T
a
s
k
F
a
i
l
u
r
e** Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage** A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult** A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e** Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName":  
"autoGroup1003", "command": "DeleteGroup" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

DeleteUser

Description:

Deletes a user. If the user belongs to a group, the user is removed from the group.

Required Parameters:

stopOnTaskFailure
userName

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies the name of a defined user.

Result

Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command": "DeleteUser", "userName": "autoUser1003" } }
```

Command Type:

collective

GetCollectiveName

Description:

Retrieves the collective name.

Requires:

e
d
p
a
r
a
m
e
t
e
r
s
:

s
t
o
p
o
n
t
a
s
k
f
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x

Example: { "task": { "stopOnTaskFailure": "true", "command": "GetCollectiveName" } }

Command Type: collective

GetHealthStatus

Description: Displays the hardware and software Health Status information for the appliance.

Required Parameters:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

il u r e

R e s u l t P a r a m e t e r s :

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E x a m p l e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"GetHealthStatus" } }
```

C o m m a n d T y p e:

collective

ModifyGroupDescription

D e

s
c
r
i
p
t
i
o
n
:

Changes the description of a selected group.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of a defined group.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
o
u
p
D
e
s
c**

Specifies a description of the group and its users.

ri p t i o n

R e s u l t P a r a m e t e r s :

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E x a m p l e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "somegroup",  
"groupDescription": "somegroupdesc", "command":  
"ModifyGroupDescription" } }
```

C o m m a n d T y p e:

collective

ModifyScheduleExportSettings

D e

S
c
r
i
p
t
i
o
n:

Enable and disable the scheduled exports feature. Also, specify an interval to generate a configuration export.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**d
a
y
l
n
t
e
r
v
a
l**

Specifies how many days to wait in between scheduled configuration exports.

**t
i
m
e**

Specifies the scheduled date that the first export will occur. Must be in the following format: YYYY-MM-DD. For example, "2014-09-08".

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
n
a
b
l
e**

Specifies if the scheduled configuration export feature is enabled or disabled.

d
a
t CLI.parameter.date.desc
e

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "dayInterval": "3", "stopOnTaskFailure": "true", "time":  
"16:25", "enable": "true", "command":  
"ModifyScheduleExportSettings", "date": "2014-02-14" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

ModifyTransportAndStorageMode

D
e

scription:

Changes the transport and storage mode setting for the collective. When you update the transport and storage mode, a task is created to update the collective settings and all appliances in the collective are restarted.

Required Parameters:

transport Specifies the transport and storage mode. Possible values are ORB, XIO, XIO_XM. Transport and storage modes enable the exchange of objects and data between different server processes. When you enable IBM eXtremeIO (XIO), relative response time is faster than the Object Request Broker (ORB). ORB is deprecated. With XIO enabled, you can also create an enterprise data grid. With an enterprise data grid, client applications that are written in different programming languages, such as Java and .NET, can access the same data grid. When you use IBM eXtremeMemory, cache entries are stored in native memory. When you use heap memory, cache entries are stored in the Java heap.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "command": "ModifyTransportAndStorageMode", "mode": "XIO" } }
```

Command Type:

collective

ModifyUserEmail

Description:

Changes the email address for a user.

Required Parameters:

s
t

**o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

e
m
a
i
l
A
d
d
r
e
s
s

u
s
e
r
N
a
m
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies a new email address for the user.

Specifies the user name of the user to update.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes**

A JSON-formatted statement that contains the

ult result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ModifyUserEmail", "emailAddress": "user1@mycompany.com",  
"userName": "user1" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

ModifyUserPassword

D
e
s
c
r
i
p
t
i
o
n:

Changes the password for a specified user.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

stopOn Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the

Task Failure batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

new Password Specifies a new password for the selected user.

password Verification Specifies the new password for the selected user, entered a second time for verification.

username Specifies the name of the user to update.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

comma

ndResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "newPassword": "newpass",
"passwordVerification": "newpass", "command":
"ModifyUserPassword", "userName": "autoUser1003" } }
```

Command Type:

collective

RemoveApplianceFromCollective

Description:

Removes an existing appliance from the current collective.

Required Parameters:

appliance

a n c e i p	Specifies the IP address of the appliance to be removed.
s t o p o n T a s k F a i l u r e	Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.
w a i t O n T a s k	Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

Result Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
comma	

ndName Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "applianceIP": "192.168.222.128", "stopOnTaskFailure":  
"true", "waitOnTask": "true", "command":  
"RemoveApplianceFromCollective" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

RemoveRoleFromGroup

D
e
s
c
r
i
p
t
i
o
n:

Removes an access role from a group. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a**

Specifies the name of the group for which you are removing an access role.

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

role

Specifies the role to remove from the group. Valid values: 2 = appliance administration, 3 = appliance monitoring, 5 = data cache creation.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "groupName":  
"autoGroup1003", "command": "RemoveRoleFromGroup", "role": "5"  
} }
```

C

o
m
m
a
n
d
T
y
p
e:

collective

RemoveRoleFromUser

D
e
s
c
r
i
p
t
i
o
n:

Removes an access role from a user. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**r
o
l
e**

Specifies the role to remove from the user. Valid values: 2 = appliance administration, 3 = appliance monitoring, 5 = data cache creation.

**u
s
e
r
N
a
m
e**

Specifies the name of for which you are removing an access role.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"RemoveRoleFromUser", "role": "2", "userName": "autoUser1003" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

RemoveUserFromGroup

Description:

Removes a user from a group. After a user is removed from a group, the user continues to have the same roles as the group. However, the user roles for the user no longer change when the group roles change. You must modify the user roles for the selected user individually.

Required Parameters:

groupname

Specifies the name of a defined group.

stopontaskfailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

username

Specifies the name of a defined user.

**a
m
e**

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

- errorMessage** A message that explains the reason for the failure, if the command failed.
- status** Specifies the status of the command that was run. The result can be either success or failed.
- commandResult** A JSON-formatted statement that contains the result of the command that was run.
- commandName** Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName":  
"autoGroup1003", "command": "RemoveUserFromGroup",  
"userName": "autoUser1003" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

ViewAllGroups

D
e
s
c
r

ip
ti
o
n:

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

Displays information about all groups.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command": "ViewAllGroups" } }
```

Command Type:

collective

ViewAllUsers

Description:

Displays the information for every user.

Required Parameters:

s

**t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

- errorMessage** A message that explains the reason for the failure, if the command failed.
- status** Specifies the status of the command that was run. The result can be either success or failed.
- commandResult** A JSON-formatted statement that contains the result of the command that was run.
- commandName** Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "stopOnTaskFailure": "true", "command": "ViewAllUsers" } }
```

**C
o
m
m
a
n
d
T
y
p
e**

collective

p
e:

ViewCollectiveLinks

D
e
s
c
r
i
p
t
i
o
n:

Displays the information for every collective link.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a

m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ViewCollectiveLinks" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

ViewGroup

D
e
s
c
r
i
p
t
i
o
n:

Displays information about a selected group.

R
e
q
u
i
r
e
d
P
a

parameters:

**group
name**
**stop
on
Task
Failure**

Specifies the name of a defined group.

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "autoGroup1003", "command": "ViewGroup" } }
```

Command Type:

collective

ViewMemberDetails

Description:

Displays information about a member in a collective.

Required Parameters:

ipAddress

Specifies the IP Address of the member.

**c
e
l
p**

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

- errorMessage** A message that explains the reason for the failure, if the command failed.
- status** Specifies the status of the command that was run. The result can be either success or failed.
- commandResult** A JSON-formatted statement that contains the result of the command that was run.
- commandName** Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "applianceIP": "192.168.222.128", "stopOnTaskFailure": "true", "command": "ViewMemberDetails" } }
```

**C
o**

m
m
a
n
d
T
y
p
e:

collective

ViewTransportAndStorageMode

D
e
s
c
r
i
p
t
i
o
n:

Displays the transport and storage mode.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "command": "ViewTransportAndStorageMode" } }
```

Command Type:

collective

ViewUser

Description:

Displays information about a selected user.

Required Parameters:

s

**t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

u
s
e
r
N
a
m
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies the name of a defined user.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "stopOnTaskFailure": "true", "command": "ViewUser",  
"userName": "autoUser1003" } }
```


C
o
m
m
a
n
d
T
y
p
e:

collective

List of GRID commands

ClearGrid

D
e
s
c
r
i
p
t
i
o
n:

Clears data from a data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to be cleared.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"command": "ClearGrid" } }
```

C
o
m
m
a
n
d
T
y
p
e:

grid

CreateGrid

D
e
s
c
r
i
p
t
i
o
n:

Creates a new simple, dynamic cache, or session data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

g
r
i
d
N
a
m
e

Specifies the name of the data grid to create.

w
a
i
t
O
n
T
a
s
k

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

k

gridType

Specifies the type of data grid to create. Valid values: simple, dynamic, or session.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"waitOnTask": "true", "command": "CreateGrid", "gridType": "simple"  
} }
```

Command Type:

grid

DeleteGrid

D
e
s
c
r
i
p
t
i
o
n:

Deletes a data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

g
r
i
d
N
a
m
e

Specifies the name of the data grid to delete.

w
a
i
t
O
n
T
a
s
k

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

k

Result Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid", "command": "DeleteGrid" } }
```

Command Type:

grid

ExportGrid

Description:

(Simple data grids only) Exports data grid information to XML.

o
n:
R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to export.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",
"command": "ExportGrid" } }
```

Command Type:

grid

GrantGroupAccess

Description:

Gives a group access rights to a data grid.

Required Parameter:

rs
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of the group to grant the given access rights.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to grant access rights.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

comma

A JSON-formatted statement that contains the

ndResult result of the command that was run.
commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "somegroup",  
"gridName": "myGrid", "command": "GrantGroupAccess" } }
```

Command Type:

grid

GrantUserAccess

Description:

Gives a user access rights to a data grid.

Required Parameters:

stop

**O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to which the user is assigned access.

**u
s
e
r
N
a
m
e**

Specifies the name of the user to give access rights.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E

X
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"command": "GrantUserAccess", "userName": "someuser" } }
```

C
o
m
m
a
n
d
T
y
p
e:

grid

ModifyGridCapacity

D
e
s
c
r
i
p
t
i
o
n:

Changes the capacity for a simple data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**ai
lu
r
e**

**u
s
e
L
R
U**

(Simple data grids only) Enables least recently used (LRU) eviction when set to true.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to update.

**w
a
i
t
O
n
T
a
s
k**

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

**g
r
i
d
C
a
p
a
c
i
t
y
L
i
m
i
t**

Specifies the maximum capacity for the selected data grid in megabytes.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "useLRU": "false",
"gridName": "myGrid", "waitOnTask": "true", "command":
"ModifyGridCapacity", "gridCapLimit": "10" } }
```

Command Type:

grid

ModifyGridReplication

Description:

Changes the replication settings for a data grid.

Required Parameters:

st

opOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

gridName Specifies the name of the data grid to update.

maximumSyncReplicas Specifies the maximum number of synchronous replicas for this data grid.

waitOnTask Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

maximumAsyncreplicas Specifies the maximum number of asynchronous replicas for this data grid.

Result Parameters:

errorM

message	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",
"waitOnTask": "true", "maximumSyncReplicas": "4", "command":
"ModifyGridReplication", "maximumAsyncReplicas": "2" } }
```

Command Type:

grid

ModifyGridSecurity

Description:

Changes the security settings for a data grid.

Required Parameters:

rs
:

stopOnTaskFailure
authorizationEnabled

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Enables authorization for the data grid when set to true.

gridName
waitOnTask

Specifies the name of the data grid to update.

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

securityEnabled

Enables security for the data grid when set to true.

R
e
s
u
l
t

Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "authorizationEnabled": "false", "gridName": "myGrid", "waitOnTask": "true", "command": "ModifyGridSecurity", "securityEnabled": "false" } }
```

Command Type:

grid

ModifyGridTTL

Description:

(Simple data grids only) Changes time to live (TTL) settings for the data grid.

Requirements:

r
e
d
P
a
r
a
m
e
t
e
r
s
:

**de
fa
ul
tM
ap
Ev
ict
or
Ty
pe**

(Optional) Specifies the time to live eviction type used for the default map. Valid values: NONE, CREATION_TIME, LAST_ACCESS_TIME, LAST_UPDATE_TIME.

**st
op
O
nT
as
kF
ail
ur
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**gr
id
N
a
m
e**

Specifies the name of the data grid to update.

**w
ait
O
nT
as
k**

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

**gr
id
TT
L**

Specifies the amount of time, in seconds, to keep data before evicting the data from data grid.

R
e
s
u
l
t
P
a
r
a

m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "defaultMapEvictorType": "CREATION_TIME",  
"stopOnTaskFailure": "true", "gridName": "myGrid", "waitOnTask":  
"true", "command": "ModifyGridTTL", "gridTTL": "60000" } }
```

C
o
m
m
a
n
d
T
y
p
e:

grid

ModifyGroupAccess

D
e
s
c
r
i
p
t
i
o
n:

Changes the group access rights for a data grid.

R
e
q
u
i
r
e
d
P

a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of the group for which you want to modify access rights.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to update.

**a
c
c
e
s
s
T
y
p
e**

Specifies the type of access to give to the group. Valid values: 1 = read, 2 = write, 3 = create, 4 = all.

R
e
s
u
l

t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "somegroup",  
"gridName": "myGrid", "command": "ModifyGroupAccess",  
"accessType": "1" } }
```

C
o
m
m
a
n
d
T
y
p
e:

grid

ModifyUserAccess

D
e
s
c
r
i
p
t
i
o
n:

Changes user access rights for a data grid.

R
e
q

u
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

g
r
i
d
N
a
m
e

Specifies the name of the data grid to update.

u
s
e
r
N
a
m
e

Specifies the name of the user for which access rights are being updated.

a
c
c
e
s
s
T
y
p
e

Specifies the type of access to give to the user. Valid values: 1 = read, 2 = write, 3 = create, 4 = all.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"command": "ModifyUserAccess", "userName": "someuser",  
"accessType": "1" } }
```

C
o
m
m
a
n

クライアント・プロパティ・ファイル

WebSphere® eXtreme Scale クライアント・プロセスの要件に基づいて、プロパティ・ファイルを作成します。

サンプル・クライアント・プロパティ・ファイル

Java `wxs_home/properties` ディレクトリーにある `sampleClient.properties` ファイルを使用して、プロパティ・ファイルを作成できます。

.NET `net_client_home¥config¥` ディレクトリーにある `Client.Net.properties` ファイルを使用して、プロパティ・ファイルを作成することができます。

Java

Java クライアントのクライアント・プロパティ・ファイルの指定

クライアント・プロパティ・ファイルは、以下のいずれかの方法で指定できます。このリスト内の項目のいずれかを使用して後で設定を指定すると、前の設定はオーバーライドされます。例えば、クライアント・プロパティ・ファイルのシステム・プロパティ値を指定すると、そのファイルのプロパティにより、クラスパスにある `objectGridClient.properties` ファイルの値が指定変更されます。

1. クラスパス内の任意の場所にあるわかりやすい名前のファイルで指定。このファイルをシステム現行ディレクトリーに置くことはサポートされていません。

```
objectGridClient.properties
```

2. スタンドアロンまたは WebSphere Application Server 構成のいずれかでシステム・プロパティとして指定。この値に、システム現行ディレクトリー内のファイルを指定することはできますが、クラスパス内のファイルは指定できません。

```
-Dobjectgrid.client.props=file_name
```

注: WebSphere Application Server 構成で、特定のクライアント・プロパティ・ファイルをシステム・プロパティで使いたい場合は、クライアント・プロパティ・ファイル (プロパティ・ファイルを特定のプロファイルに適用されたいのか、またはインストール済み環境全体に適用したいのか) に応じて、`was_root/properties` または `profile_root/properties` など) をクラスパスに置く必要があります。

3. `ClientClusterContext.getClientProperties` メソッドを使用したプログラマチックな指定変更。オブジェクトのデータに、プロパティ・ファイルからのデータが取り込まれます。セキュリティ・プロパティは、このメソッドで構成できません。

.NET

WebSphere eXtreme Scale Client for .NET のクライアント・プロパティ・ファイルの指定

デフォルトのクライアント・プロパティ・ファイルは `net_client_home¥config¥Client.Net.properties` ディレクトリーにあります。独自のプロパティ・ファイルを指定したい場合は、各 `Connect` メソッド呼び出しへの絶対プロパティ・ファイル・パスを指定します。

```
IGridManager gm = GridManagerFactory.GetGridManager( );
ICatalogDomainInfo cdi = gm.CatalogDomainManager.CreateCatalogDomainInfo( "localhost:2809" );
ccc = gm.Connect( cdi, @"C:¥MyLocation¥MyClient.properties" );
grid = gm.GetGrid( ccc, gridName );
```

クライアント・プロパティ

クライアント・プロパティ

.NET **2.5+** `enableDynamicConfiguration`

これが `true` に設定されると、クライアント・プロパティ・ファイル内の `requestRetryTimeout` プロパティに対して行われた変更が動的に検出されます。新しい要求再試行タイムアウト値を計算するために新しいプロパティ値がただちに使用されます。

デフォルト: `false`

Java

listenerHost

オブジェクト・リクエスト・ブローカー (ORB) または eXtremeIO (XIO) トランスポート・プロトコルが通信用にバインドするバインド先のホスト名を指定します。この値は、完全修飾ドメイン名または IP アドレスである必要があります。ご使用の構成に複数のネットワーク・カードが含まれている場合は、JVM 内のトランスポート・メカニズムがソバ

インド先の IP アドレスを知ることができるようにリスナー・ホストとリスナー・ポートを設定してください。使用する IP アドレスを指定しなければ、接続タイムアウトや異常な API 障害、クライアントがハングしたように見える状態などの症状が現れることがあります。

Java listenerPort

オブジェクト・リクエスト・ブローカーまたは eXtremeIO (XIO) トランスポート・プロトコルが通信用にバインドするバインド先のポート番号を指定します。listenerPort に対して定義されたポート番号は、クライアントとカタログ・サーバーの間の通信、および同じドメイン内にあるコンテナ・サーバーとカタログ・サーバーの間の通信に使用されます。また、ドメイン間およびドメイン内のカタログ・サーバー間通信にも使用されます。

デフォルト: 2809

注: WebSphere Application Server 内でデータ・グリッド・サーバーが実行され、ORB トランスポート・プロトコルが使用されている場合は、別のポート ORB_LISTENER_ADDRESS も開く必要があります。BOOTSTRAP_ADDRESS ポートは要求をこのポートに転送します。XIO トランスポート・プロトコルを使用している場合は、XIO_ADDRESS ポートを開く必要があります。

Java preferLocalProcess

このプロパティは、現在使用されていません。将来の使用のために予約済みです。

Java preferLocalHost

このプロパティは、現在使用されていません。将来の使用のために予約済みです。

Java .MET preferZones

優先ルーティング・ゾーンのリストを指定します。各ゾーンは、preferZones=ZoneA,ZoneB,ZoneC の形式でコマンドで区切って指定します。

デフォルト: 値なし

Java .MET requestRetryTimeout

例外が発生してから要求の処理をいつまで続行するかをミリ秒で指定します。以下の有効値のいずれかを使用します。

- 値が 0 の場合、要求は、フェイル・ファーストになり、内部の再試行ロジックは飛ばされます。
- 値が -1 の場合、要求再試行のタイムアウトは設定されません。すなわち、要求の所要時間は、トランザクション・タイムアウトによって制御されます。(デフォルト)
- 0 より大きい値は、要求再試行のタイムアウト値をミリ秒で示しています。正常に作成されない例外が返されます。DuplicateException などの例外が再試行されるときでも、成功しないときはその例外も返されます。トランザクション・タイムアウトは、最大待ち時間として引き続き使用されます。

Java .MET shuffleBootstrapAddresses

データ・グリッドへのブートストラップ時にクライアントによってカタログ・サービス・アドレスが使用されるとき、カタログ・サービス・アドレスをランダム化するかどうかを指定します。値は true または false のいずれかでなければなりません。

デフォルト: true

Java 2.5+ xioTimeout

eXtremeIO によるアウトバウンド・ソケット接続試行のタイムアウト値を秒で指定します。これは、内部 eXtremeIO ロジックのデフォルト・タイムアウトとしても使用されます。

Java 2.5+ xioRequestTimeout

要求が応答を待つ秒数を指定します (この秒数を過ぎると待つのを止めます)。このプロパティは、ネットワーク停止の障害が発生した場合にクライアントがフェイルオーバーするまでに要する時間に影響します。このプロパティの値を極端に低く設定すると、要求が誤ってタイムアウトになる可能性があります。不用意なタイムアウトを回避するためにこのプロパティの値は慎重に考慮してください。

セキュリティー・クライアント・プロパティ

一般セキュリティー・プロパティ

Java .MET securityEnabled

WebSphere eXtreme Scale クライアント・セキュリティーを使用可能にします。この設定は、WebSphere eXtreme Scale サーバー・プロパティ・ファイルの securityEnabled 設定と一致している必要があります。設定が一致しないと、データ・グリッドへのクライアント接続は失敗します。

デフォルト: false

資格情報認証の構成プロパティ

Java .MET credentialAuthentication

クライアントの資格情報認証のサポートを指定します。以下の有効値のいずれかを使用します。

- Never: クライアントは、資格情報認証をサポートしません。
- Supported: サーバーも資格情報認証をサポートする場合に、クライアントは資格情報認証をサポートします。(デフォルト)
- Required: クライアントは資格情報認証を必要とします。

Java .NET authenticationRetryCount

資格情報の有効期限が切れている場合に認証を試行できる回数を指定します。この値が 0 に設定されていると、認証の再試行はできません。

デフォルト: 3

.NET credentialGeneratorAssembly

WebSphere eXtreme Scale Client for .NET の資格情報を生成するために使用されるアセンブリーの名前を指定します。このプロパティを指定するためには、credentialAuthentication プロパティを Supported または Required に設定する必要があります。値は、バージョン、国別情報その他のプロパティが含まれている有効な C# .dll アセンブリー名でなければなりません。

例: IBM.WebSphere.Caching.CredentialGenerator, Version=8.6.0.0, Culture=neutral, PublicKeyToken=b439a24ee43b0816

Java .NET credentialGeneratorClass

com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator インターフェースを実装するクラスの名前を指定します。このプロパティを指定するためには、credentialAuthentication プロパティを Supported または Required に設定する必要があります。このクラスを使用して、クライアントの資格情報が取得されます。

デフォルト: 値なし

Java .NET credentialGeneratorProps

CredentialGenerator 実装クラスの プロパティを指定します。このプロパティが、setProperty(String) メソッドを使用してオブジェクトに設定されます。このプロパティを指定するためには、credentialAuthentication プロパティを Supported または Required に設定する必要があります。credentialGeneratorProps 値は、credentialGeneratorClass プロパティの値が非ヌルの場合にのみ使用されます。

Transport Layer Security の構成プロパティ

Java .NET transportType

クライアントのトランスポート・タイプを指定します。可能な値は以下のとおりです。

- TCP/IP: クライアントが、TCP/IP 接続のみをサポートすることを示します。
- SSL-Supported: クライアントが TCP/IP 接続と Secure Sockets Layer (SSL) 接続の両方をサポートすることを示します。(デフォルト)
- SSL-Required: クライアントが SSL 接続を必要とすることを示します。

SSL 構成プロパティ

Java 別名 (alias)

鍵ストア内の別名を指定します。鍵ストアに複数の鍵ペア証明書があり、いずれか 1 つの証明書を選択したい場合は、このプロパティを使用します。

デフォルト: 値なし

Java contextProvider

トラスト・サービスのコンテキスト・プロバイダーの名前を指定します。有効でない値を指定すると、コンテキスト・プロバイダー・タイプが正しくないことを示すセキュリティー例外が発生します。

有効値: IBMJSSE2、IBMJSSE、IBMJSSEFIPS など。

Java keyStore

鍵ストア・ファイルへの完全修飾パスを指定します。

例:

etc/test/security/client.private

Java keyStoreType

鍵ストアのタイプを示します。有効でない値を指定すると、ランタイム・セキュリティー例外が発生します。

有効値: JKS、JCEK、PKCS12 など。

Java .NET プロトコル

クライアントに使用するセキュリティー・プロトコルのタイプを指定します。このプロトコルの値は、使用している

セキュリティー・プロバイダーに基づいて設定します。有効でない値を指定すると、プロトコル値が正しくないことを示すセキュリティー例外が発生します。

Java 有効値: SSL、SSLv3、TLS、TLSv1 など。

.NET 有効な値: SSLv2、SSLv3、TLS、または Default (SSLv3 または TLS1.0)

.NET publicKeyFile

サーバーからエクスポートされた公開鍵が入っているファイルへの完全修飾パス名を指定します。

例: c:\tmp\wxs\serverA.cer

Java trustStoreType

トラストストアのタイプを示します。有効でない値を指定すると、ランタイム・セキュリティー例外が発生します。

有効値: JKS、JCEK、PKCS12 など。

Java trustStore

トラストストア・ファイルへの完全修飾パスを指定します。

例:

etc/test/security/server.public

Java keyStorePassword

鍵ストアのストリング・パスワードを指定します。この値はエンコードすることも、実際の値を使用することもできます。

Java trustStorePassword

トラストストアのストリング・パスワードを指定します。この値はエンコードすることも、実際の値を使用することもできます。

親トピック: [参照](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[クライアント・セキュリティーの構成](#)

.NET [WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連情報:

[ICredential インターフェース](#)

[ICredentialGenerator インターフェース](#)

正規表現の構文

正規表現は、他の文字列とのマッチングに使用される構造化文字列です。正規表現は、データを無効化する場合およびメッセージに表示されるメッセージをフィルタリングする場合に使用することができます。

正規表現の例

正規表現	説明
.	任意の 1 文字を検索します。
.*	任意の 1 文字を 0 回以上検索します。
A.*	「A」で始まるすべての文字列を検索します。
.*A	「A」で終わるすべての文字列を検索します。
A?	「A」を 0 回または 1 回検索します。
A*	「A」を 0 回または複数回検索します。
A+	「A」を 1 回または複数回検索します。
A?B*C+	後に 0 個または 1 個の A が続き、その後に 0 個または複数個の B が続き、その後に 1 個または複数個の C が続く任意の 1 文字を検索します
[ABC].*	「A」、「B」、「C」のいずれかの文字で始まる文字列を検索します。
.*ABC.*	文字列のどこかに「ABC」という文字列を含む文字列を検索します。
(ABC DEF).*	文字列「ABC」または「DEF」で始まる文字列を検索します。

詳しい正規表現の構文については `java.util.regex.Pattern` API の資料を参照してください。

親トピック: [参照](#)

動的マップの構成オプション

ご使用のクライアント・アプリケーションを特定の名前の付いたマップに接続させることで、データ・グリッド内に追加のマップを作成できます。この接続が発生すると、マップが自動的に作成されます。

動的マップの命名

データ・グリッドを作成すると、デフォルトでは2つのマップが作成されます。最初のマップには、ご使用のデータ・グリッドと同じ名前が付けられます。例えば、myGrid というデータ・グリッドを作成すると、自動的に myGrid というマップが作成されます。2番目のマップはニア・キャッシュ用で、myGrid.NCI という名前が付けられます。しかし、データ・グリッドにマップを追加することも可能です。クライアント・アプリケーションが次の命名規則を使用したマップに接続する際、1つのマップが自動作成されます。

```
<map_name>.<template>.<locking_option>.<near_cache_invalidation>
```

各部の意味は、次のとおりです。

map_name (必須)

マップの名前を指定します。

template (必須)

存続時間 (TTL) の動作を定義することで、エントリーの有効期限がマップから切れる時期を定義するテンプレートを指定します。使用できるオプションのリストについては、[マップ・テンプレート](#) を参照してください。

locking_option

マップに使用されるロック・メカニズムを指定します。使用できるオプションのリストについては、[ロック・オプション](#) を参照してください。

2.5+ near_cache_invalidation

ニア・キャッシュの無効化をマップに使用するかを指定します。ニア・キャッシュの無効化は、マップで、ロック・タイプを NONE または OPTIMISTIC に設定している場合にのみ使用されます。できるだけ速やかにニア・キャッシュから失効データが削除されるようにニア・キャッシュの無効化を構成することができます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化がトリガーされます。使用可能なオプションのリストは、[ニア・キャッシュ無効化オプション](#) を参照してください。

マップのテンプレート名を含める必要があります。ロック・オプションを指定しなければ、マップ上でロックは発生しません。

マップ・テンプレート

表 1. 動的マップ・テンプレート

マップ・テンプレート	説明
*.NONE	存続時間 (TTL) に有効期限がないマップを指定します。
*.LUT	項目の最終更新時間に基づいて項目の有効期限が切れる マップを指定します。デフォルトの TTL は 1 時間です。
*.LAT	項目の最終アクセス時間に基づいて項目の有効期限が切れるマップです。デフォルトの TTL は 1 時間です。
*.CT	エントリーの作成時間と TTL 値に基づいてエントリーの有効期限が切れるマップです。デフォルトの TTL は 1 時間です。

ロック・オプション

表 2. 動的マップのロック・オプション

ロック・オプション	説明
(ブランク)	ロック・オプションを指示しなければ、ロック・メカニズムは使用されません。
.P	マップにペシミスティック・ロックがあるよう指定します
.O	マップにオプティミスティック・ロックがあることを指定します

2.5+

ニア・キャッシュ無効化オプション

表 3. ニア・キャッシュ無効化オプション

ロック・オ	
-------	--

プシヨン	説明
(プランク)	値を指定しないと、ニア・キャッシュ無効化は使用不可に設定されます。
.NCI	マップでニア・キャッシュ無効化を使用することを指定します。お使いのマップで、オプティミスティック・ロックまたは非ロック式が使用可能に設定されている必要があります。

親トピック: [参照](#)

関連概念:

[Evictor](#)

関連タスク:

[存続時間 \(TTL\) Evictor の構成](#)

[動的マップの構成](#)

[Java 2.5+ ニア・キャッシュ無効化の構成](#)

[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

[単純データ・グリッドの作成](#)

[.NET .NET API を使用した動的マップの作成](#)

CLI コマンド解説書

このページは、WebSphere® DataPower® XC10 アプライアンスにより提供されるコマンド行インターフェース (CLI) を記載しています。以下の 3 セットのコマンドがサポートされています。

- 様々な構成オブジェクトを処理する共通コマンド。 [共通コマンド](#) を参照してください。
- 特殊目的コマンド。 [特殊目的コマンド](#) を参照してください。
- フェイルセーフ・モードで使用可能なコマンド。

共通コマンド

このセクションに記載されているコマンドは、様々な構成オブジェクトを処理するために使用されます。共通コマンドは、指定された構成オブジェクトを処理するか、または構成オブジェクトの編集セッション内で使用されます。編集セッションとは、構成オブジェクトの値を変更できる CLI のモードです。編集セッションは「edit」または「create」コマンドで始まり、「exit」または「cancel」コマンドで終わります。これらのコマンドの一部、例えば「append」や「set」などは、編集セッション内でのみ使用可能です。

現在サポートされている構成オブジェクトは以下のとおりです。

- aggregate-interface
- ethernet-interface
- ldap-auth
- logging-settings
- netsec-settings
- network-settings
- vlan-interface

編集セッション内で機能するコマンドには、以下のコマンドが含まれます。

[**cancel**](#)

[**create**](#)

[**delete**](#)

[**disable**](#)

[**domain list**](#)

[**domain show**](#)

[**domain switch**](#)

[**edit**](#)

[**enable**](#)

[**exit**](#)

[**list**](#)

[**reset**](#)

[**show**](#)

[**show properties**](#)

[**show types**](#)

[**status**](#)

親トピック: [CLI コマンド解説書](#)

cancel

目的

編集セッション内で、編集セッション中またはネストされた編集セッション中に行われた構成オブジェクトへの変更すべてを廃棄します。

構文

cancel

パラメーター

なし。

使用上の注意

ネストされた編集セッションでは、**cancel** は編集サブモード内で行われた変更のみを廃棄します。

関連コマンド

[exit](#) を参照してください。

例

編集セッション中に network-settings 構成オブジェクトに行った変更をキャンセルします。

```
Console> edit network-settings
Console network-settings> set tcp-retries 10
Console network-settings> cancel
Cancelled
Console>
```

親トピック: [共通コマンド](#)

create

目的

編集セッションに入り、構成オブジェクトのインスタンスを作成します。

構文

create *object-type object-instance*

パラメーター

object-type

作成されるオブジェクトのタイプ

object-instance

作成されるオブジェクト・インスタンスの名前

使用上の注意

- 一部のオブジェクトは、コマンド行インターフェースで作成できません。これらのオブジェクトには以下のものが含まれます。
 - シングルトン・オブジェクトと呼ばれる、インスタンスが1つだけのオブジェクト
 - ethernet-interface オブジェクトなど、事前定義された有限のセットのあるオブジェクト
- デフォルトでは、新規作成されたオブジェクトの AdminState は使用可能になっています。

関連コマンド

[delete](#)、[edit](#)、および [list](#) を参照してください。

例

v1000 という名前の vlan-interface を作成します。

```
Console> create vlan-interface v1000
Console vlan-interface:v1000>
...
```

親トピック: [共通コマンド](#)

delete

目的

構成オブジェクトのインスタンスを削除します。

構文

delete *object-type object-instance*

パラメーター

object-type

削除されるオブジェクトのタイプ

object-instance

削除されるオブジェクト・インスタンスの名前

使用上の注意

一部のオブジェクトは削除できません。これらのオブジェクトには以下のものが含まれます。

- シングルトン・オブジェクトと呼ばれる、インスタンスが 1 つだけのオブジェクト
- ethernet-interface オブジェクトなど、事前定義された有限のセットのあるオブジェクト

関連コマンド

[create](#)、[edit](#)、および [list](#) を参照してください。

例

v1000 という名前の vlan-interface を削除します。

```
Console> delete vlan-interface v1000
Console>
```

親トピック: [共通コマンド](#)

disable

目的

- 構成オブジェクト、または構成オブジェクトのインスタンスを使用不可に設定します。
- このコマンドは、管理状態 (AdminState) を「disabled」に変更します。エラーが発生しなければ、操作状態 (OpState) が「down」に変わります。
- このコマンドの結果、他の構成値は変更されません。

構文

disable *object-type* [*object-instance*]

パラメーター

object-type

使用不可に設定されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、使用不可に設定されるオブジェクト・インスタンスの名前

使用上の注意

構成変更は管理上使用不可のオブジェクトにも行えますが、オブジェクトが使用可能になるまでは変更は適用されません。

関連コマンド

[enable](#) を参照してください。

例

v1000 という名前の vlan-interface を使用不可に設定します。

```
Console> disable vlan-interface v1000
Console>
```

親トピック: [共通コマンド](#)

domain list

目的

サポートされるドメインを表示します。

構文

domain list

パラメーター

なし。

使用上の注意

現在は 1 つのドメイン「default」のみがサポートされています。

関連コマンド

[domain show](#) および [domain switch](#) を参照してください。

例

現在サポートされているドメインを表示します。

```
Console> domain list
default
Console>
```

親トピック: [共通コマンド](#)

domain show

目的

現在アクティブなドメインを表示します。

構文

domain show

パラメーター

なし。

使用上の注意

現在は 1 つのドメイン「default」のみがサポートされています。

関連コマンド

[domain list](#) および [domain switch](#) を参照してください。

例

現在アクティブなドメインを表示します。

```
Console> domain show
In domain "default"
Console>
```

親トピック: [共通コマンド](#)

domain switch

目的

現在アクティブなドメインを変更します。

構文

```
domain switch domain
```

パラメーター

domain

要求されるアクティブ・ドメイン

使用上の注意

現在は 1 つのドメイン「default」のみがサポートされています。

関連コマンド

[domain list](#) および [domain show](#) を参照してください。

例

現在アクティブなドメインを変更します。

```
Console> domain switch default
Switched to "default" domain
Console>
```

親トピック: [共通コマンド](#)

edit

目的

構成モードに入り、構成オブジェクト、または構成オブジェクトのインスタンスを変更します。

構文

edit *object-type* [*object-instance*]

パラメーター

object-type

編集されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、編集されるオブジェクト・インスタンスの名前

使用上の注意

edit コマンドによって、構成されたオブジェクトの編集セッションに入ります。この編集セッション内では、**edit** サブコマンドにより、構成されたオブジェクトのプロパティの値を変更することができます。

関連コマンド

[create](#)、[delete](#)、および [list](#) を参照してください。

例

ethernet-interface eth3 を編集して MTU を 4096 に設定します。

```
Console> edit ethernet-interface eth3
Console ethernet-interface:eth3> set mtu 4096
Console ethernet-interface:eth3> exit
Console>
```

親トピック: [共通コマンド](#)

enable

目的

- 構成オブジェクト、または構成オブジェクトのインスタンスを使用可能に設定します。
- このコマンドは、管理状態 (AdminState) を「enabled」に変更します。エラーが発生しなければ、操作状態 (OpState) が「up」に変わります。
- このコマンドの結果、他の構成値は変更されません。

構文

enable *object-type* [*object-instance*]

パラメーター

object-type

使用可能に設定されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、使用可能に設定されるオブジェクト・インスタンスの名前

関連コマンド

[disable](#) を参照してください。

例

v1000 という名前の vlan-interface を使用可能に設定します。

```
Console> enable vlan-interface v1000
Console>
```

親トピック: [共通コマンド](#)

exit

目的

編集モードまたは編集サブモードを終了します。

構文

exit

パラメーター

なし。

使用上の注意

構成変更は、編集サブモードの終了時ではなく、編集セッションの終了時に保存されます。このため、検証エラーが予想より遅く表示されることがあります。

関連コマンド

[cancel](#) を参照してください。

例

ethernet-interface オブジェクトの編集セッションに入り、ip サブモードに入ります。IP アドレスを追加し、その後 ip サブモードを終了し、編集セッションを終了します。

```
Console> edit ethernet-interface eth2
Console ethernet-interface:eth2> ip
Entering "ip" mode
Console ethernet-interface:eth2 ip> append address 192.168.0.200/24
Console ethernet-interface:eth2 ip> exit
Console ethernet-interface:eth2> exit
Console>
```

親トピック: [共通コマンド](#)

list

目的

指定されたオブジェクト・タイプの構成されたオブジェクトの名前を表示します。

構文

list *object-type*

パラメーター

object-type

リストされるオブジェクトのタイプ

使用上の注意

オブジェクト・タイプのリストを取得するには、[show types](#) を使用してください。

関連コマンド

[create](#)、[delete](#)、[edit](#)、および [show types](#) を参照してください。

例

ethernet-interfaces のすべてのインスタンスをリストします。

```
Console> list ethernet-interface
    eth0
    eth1
    eth2
    eth3
Console>
```

親トピック: [共通コマンド](#)

reset

目的

構成オブジェクト、または構成オブジェクトの一部をデフォルト設定にリストアします。

構文

- シングルトン構成オブジェクトをリセットする: **reset object-type**
- 構成オブジェクトのインスタンスをリセットする: **reset object-type object-instance**
- 編集セッション内の構成オブジェクトのフィールドをリセットする: **reset property**

パラメーター

object-type

リセットされるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、リセットされるオブジェクトのインスタンス

property

リセットされる構成オブジェクト内のフィールド

例

- ethernet-interface eth3 の IP アドレスをリセットします。

```
Console> edit ethernet-interface eth3
Console ethernet-interface:eth3> ip
Entering "ip" mode
Console ethernet-interface:eth3 ip> reset address
Console ethernet-interface:eth3 ip> exit
Console ethernet-interface:eth3> exit
Console>
```

- ethernet-interface eth3 の構成オブジェクト全体をデフォルト値にリセットします。

```
Console> reset ethernet-interface eth3
Console>
```

親トピック: [共通コマンド](#)

show

注: **show** コマンドは、構成オブジェクトとその他の特定データの両方を表示するのに使用されます。このセクションは、構成オブジェクトに対する **show** コマンドの使用について記載しています。[特殊目的コマンド](#) セクション内の **show** コマンドも参照してください。

目的

指定された構成オブジェクト、または構成オブジェクトのインスタンスに構成されている値を表示します。

構文

```
show [ object type [ object instance ] ]
```

パラメーター

object type

表示されるオブジェクトのタイプ

object instance

構成オブジェクトがシングルトン・オブジェクトでない場合、表示されるオブジェクト・インスタンスの名前

使用上の注意

- 編集セッション内では、**show** コマンドは編集集中のオブジェクト・インスタンスで現在構成されている値を表示します。
- オブジェクトの操作状態も表示されます。必須のプロパティと、割り当てられたデフォルト値と異なる値が構成されたプロパティのみが表示されます。明示的に構成されていないオプションのプロパティは表示されません。
- ここに表示される値は、オブジェクトに要求される構成です。対照的に、[status](#) コマンドはオブジェクトの実際のランタイム値を表示します。

例

network-settings (シングルトン) オブジェクト用に構成された値を表示します。

```
Console> show network-settings
network-settings : [Up]

name "network-settings"
AdminState "Enabled"
icmp-options " "
explicit-congestion-notification "false"
destination-based-routing "false"
interface-isolation "relaxed"
tcp-retries "5"
arp-retries "8"
arp-retry-interval "500"
tso-offload "true"
reverse-path-filtering "false"
tcp-window-scaling "true"
```

親トピック: [共通コマンド](#)

show properties

目的

編集セッション内で、オブジェクトの構成可能プロパティを表示します。

構文

show properties

パラメーター

なし。

使用上の注意

ネストされたレベルのプロパティを表示するには、**edit** サブモードに入り、コマンドを繰り返します。

例

ethernet-interface の構成可能プロパティを表示します。

```
Console> edit ethernet-interface eth3
Console ethernet-interface:eth3> show properties
Basic properties:
  AdminState
  mac-address
  mode
  mtu
  use-arp
  userdata

Sub-mode (complex) properties:
  ip

Console ethernet-interface:eth3> ip
Entering "ip" mode
Console ethernet-interface:eth3 ip> show properties
Basic properties:
  dad-retransmit-timer
  dad-transmits
  use-dhcp
  use-slaac

Array (list) properties:
  address
  static-route

Sub-mode (complex) properties:
  ipv4-default-gateway
  ipv6-default-gateway

Console ethernet-interface:eth3 ip>
```

親トピック: [共通コマンド](#)

show types

目的

アプライアンスでサポートされている構成オブジェクトのタイプを表示します。これは構成されたインスタンスを反映せず、サポートされるタイプを示すだけです。

構文

show types

パラメーター

なし。

使用上の注意

なし。

例

アプライアンスの構成オブジェクトを表示します。

```
Console> show types
The following types are defined:
  aggregate-interface
  ethernet-interface
  logging-settings
  network-settings
  vlan-interface
```

親トピック: [共通コマンド](#)

status

注: **status** コマンドは、構成されたオブジェクトの状況と特定の機能状況情報の両方を表示するのに使用されます。このセクションは、構成オブジェクトに対する **status** コマンドの使用について記載しています。[特殊目的コマンド](#) セクション内の status コマンドも参照してください。

目的

指定された構成オブジェクト、または構成オブジェクトのインスタンスの状況を表示します。

構文

```
status object-type [ object-instance ]
```

パラメーター

object-type

表示されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、表示されるオブジェクト・インスタンスの名前

使用上の注意

- シングルトン・オブジェクトでない構成オブジェクトで *object-instance* が省略されている場合、指定された *object-type* のすべてのインスタンスの状況が表示されます。
- [status](#) コマンドは、選択された構成オブジェクトのみでサポートされます。
- ここに表示される値は、オブジェクトのランタイム値です。対照的に、[show](#) コマンドはオブジェクトの構成されたプロパティを表示します。

例

ethernet-interface eth0 の状況を表示します。

```
Console> status ethernet-interface eth0
eth0      OpState:[Up]
          generic MTU:1500 carrier:true flags:UP BROADCAST RUNNING MULTICAST
          index:5
          inet addr:9.42.77.48 flags:PERMANENT mask:255.255.255.0
          scope:GLOBAL
          inet6 addr: 2002:92a:8f7a:901:9:42:77:48 flags:PERMANENT
          mask: ffff:ffff:ffff:ffff:: scope:GLOBAL
          inet6 addr: fe80::20c:29ff:fedd:7baf flags:PERMANENT
          mask: ffff:ffff:ffff:ffff:: scope:LINK
          ethernet Link:on MAC: 00:0c:29:dd:7b:af autoneg:on duplex:Full
          port:TP speed:1000Mbps
          statistics collisions:0 multicast:0 rx_bytes:69088545
          rx_compressed:0 rx_crc_errors:0 rx_dropped:0 rx_errors:0
          rx_fifo_errors:0 rx_frame_errors:0 rx_length_errors:0
          rx_missed_errors:0 rx_over_errors:0 rx_packets:842919
          tx_aborted_errors:0 tx_bytes:3512656 tx_carrier_errors:0
          tx_compressed:0 tx_dropped:0 tx_errors:0 tx_fifo_errors:0
          tx_heartbeat_errors:0 tx_packets:18144 tx_window_errors:0
```

親トピック: [共通コマンド](#)

特殊目的コマンド

このセクションに記載されているコマンドは、アプライアンスで特定のアクションを行うか、特定の情報を表示します。

[add-jvm-args](#)

[alias](#)

[clear-all](#)

[clear-jvm-args](#)

[clear-logs](#)

[clear-tls-config](#)

[collect-logs](#)

[component firmware update](#)

[config](#)

[datetime get](#)

[datetime set](#)

[deleteExport](#)

[device RESET](#)

[device battery-replaced](#)

[device intrusion allow](#)

[device intrusion clear](#)

[device intrusion disallow](#)

[device nvdim clear](#)

[device clear-intrusion](#)

[device restart](#)

[device shutdown](#)

[echo](#)

[export](#)

[file delete](#)

[file get](#)

[file list](#)

[file put](#)

[firmware pristine-install](#)

[firmware rollback](#)

[firmware upgrade](#)

[force recycle](#)

[get-dns-search](#)

[get-dns-servers](#)

[get-ntp-servers](#)

[help](#)

[import](#)

[license accept](#)

[license get](#)

[license reset](#)

[listExports](#)

[locale get](#)

[locale set](#)

[locate-led](#)

[log message](#)

[netif](#)

[net-test available](#)

[net-test dns](#)

[net-test ping](#)

[net-test tcp](#)

[net-test traceroute](#)

[nodename get](#)

[nodename set](#)

[packet-capture clear](#)

[packet-capture list](#)

[packet-capture start](#)

[packet-capture stop](#)

[platform collect-pd](#)

[platform log-level get](#)

[platform log-level set](#)

[platform service disable ssh](#)

[platform service disable telnet](#)

[platform service enable ssh](#)

[platform service enable telnet](#)

[platform must-gather](#)

[raid delete](#)

[raid re-use](#)

[raid foreign-config import](#)

[raid foreign-config clear](#)

2.5+ [request force-suspend](#)

2.5+ [request resume](#)

2.5+ [request suspend](#)

[set-dns-search](#)

[set-dns-servers](#)

[set-ntp-servers](#)

[show commandref](#)

[show components](#)

[show locales](#)

[show log](#)

[show timezones](#)

[show version](#)

[sshkey fingerprint](#)

[sshkey reset](#)

[sshkey verify](#)

[start-progress](#)

[status battery](#)

[status cpu-usage](#)

[status cpu-utilization](#)

[status dynamic-tunnels](#)

[status fan](#)

[status flash](#)

[status intrusion](#)

[status memory](#)

[status nvdimmm](#)

[status power-supply](#)

[status raid all](#)

[status raid battery](#)

[status raid physical](#)

[status voltage](#)

[status volume](#)

[status temperature](#)

[status uptime](#)

[timezone get](#)

[timezone set](#)

[unalias](#)

[usage](#)

[user add](#)

[user delete](#)

[user known-hosts list](#)

[user known-hosts delete](#)

[user list](#)

[user password](#)

[user sshkey add](#)

[user sshkey delete](#)

add-jvm-args

目的

事前設定された JVM 構成パラメーターのセットを、始動中にアプライアンスのプロセスのいずれかに追加します。

構文

add-jvm-args process name listOfOneOrMoreJVMArgs

パラメーター

このコマンドには、以下の 2 つの必須パラメーターがあります。

process name

構成される必要があるプロセスの名前。有効な processName 引数は以下のとおりです。

- all - xsa.app、カタログ・サーバー、およびすべてのコンテナ・サーバーを含む、すべてのプロセス。
- console - xsa.app プロセスのみを含む。
- grids - すべてのコンテナ・サーバー・プロセスのみ。
- gridNN - ここで NN は単一のコンテナ・サーバーを指定する数値 (つまり 01)。

listOfOneOrMoreJVMArgs

プロセスで有効な JVM 引数をリストします。指定可能である有効な JVM 引数は以下のとおりです。

- -Dcom.ibm.CORBA.Debug
- -Dcom.ibm.CORBA.CommTrace
- -Djavax.net.debug
- -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true
- -verbose:gc
- -Xverbosegclog
- -Xhealthcenter
- -Xcheck:jni

注: リストに含めたいパラメーターが複数ある場合は、`!!` で区切ります。(例: `add-jvm-args grid01 -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true!!`)

```
-Dcom.ibm.CORBA.CommTrace add-jvm-args all -Dcom.ibm.CORBA.CommTrace
```

関連コマンド

[device restart](#) を参照してください。

例

```
Console> add-jvm-args grid01 -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true
Updating JVM Args to include new args...
Adding the following JVM Arg: -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true
[Wed Apr 10 2013 17:00:22] Run device restart to enable the new JVM args settings.
```

親トピック: [特殊目的コマンド](#)

alias

目的

- コマンド行インターフェース (CLI) コマンドの代替名またはショートカットを定義します。
- 構成された別名を表示します。

構文

- 定義された別名のリストを表示する: **alias**
- 既存の別名の定義を表示する: **alias name**
- 指定された CLI コマンドの別名を定義する: **alias name value**

パラメーター

name

コマンド行インターフェース (CLI) コマンドの代替名。

value

指定された名前への代替値。代替値にスペースが含まれる場合は、値を二重引用符で囲みます。

関連コマンド

[unalias](#) を参照してください。

例

特定のアドレスへの接続のテストにショートカットを定義します。それから構成された別名を表示します。

```
Console> alias test-connectivity "net-test ping 9.42.106.2"
Console> alias
alias test-connectivity "net-test ping 9.42.106.2"
Console>
```

親トピック: [特殊目的コマンド](#)

clear-all

目的

アプライアンスの構成データをリセットします (すべてのデータ・グリッドおよび新規ユーザーの削除を含む)。アプライアンスで実行中のすべての内部プロセスは、再始動されます。このコマンドを実行しても、アプライアンス・ハードウェアは再始動されません。

構文

clear-all

パラメーター

なし。

関連コマンド

なし。

例

```
Console> clear-all
Force Stopped all XC-10 processes
Deleting configuration data and logs
Deleting grid data
```

親トピック: [特殊目的コマンド](#)

clear-jvm-args

目的

始動中にアプライアンスに適用されたプロセスのいずれかに対する JVM 構成パラメーターをクリアします。

構文

clear-jvm-argsprocessName

パラメーター

process name

構成される必要があるプロセスの名前。有効な processName 引数は以下のとおりです。

- all - xsa.app、カタログ・サーバー、およびすべてのコンテナ・サーバーを含む、すべてのプロセス。
- console - xsa.app プロセスのみを含む。
- grids - すべてのコンテナ・サーバー・プロセスのみ。
- gridNN - ここで NN は単一のコンテナ・サーバーを指定する数値 (つまり 01)。

関連コマンド

[add-jvm-args](#) を参照してください。

例

```
Console> clear-jvm args all
```

親トピック: [特殊目的コマンド](#)

clear-logs

目的

WebSphere® DataPower® XC10 アプライアンスのログ・ファイルをすべて削除します。

構文

clear-logs

パラメーター

なし。

関連コマンド

なし。

例

```
Console> clear-logs  
Cleared All Logs
```

親トピック: [特殊目的コマンド](#)

clear-tls-config

目的

Transport Layer Security (TLS) 構成をリセットします。すべての構成データを失いたくないので clear-all は選択肢にならない、しかし TLS 構成が破損したかデフォルトの TLS 値をリストアしたいという場合に、このコマンドを実行します。ほとんどの場合には、TLS 変更用のユーザー・インターフェースを使用するか、**clear-all** コマンドを使用して、アプライアンスの元の構成に戻します。集合内の各アプライアンスで、**clear-tls-config** コマンドを実行します。各アプライアンスでコマンドを実行した後、集合内の各アプライアンスの中のプロセスを再始動します。集合が正常に通信している場合、**device restart** コマンドを使用します。集合内のすべてのアプライアンスがユーザー・インターフェースを通してアクセス可能で、「集合」パネルで開始済みであることが確認できる時、集合は正しく通信しています。しかし、TLS 構成によって集合の通信が妨げられていて、**device restart** コマンドでアプライアンスがバックアップしない場合、**force-recycle** コマンドを使用して、データを保存せずにアプライアンスの中のすべてのプロセスを強制的に停止して始動することができます。

構文

clear-tls-config

パラメーター

なし。

関連コマンド

[clear-all](#)、[device restart](#)、および [force recycle](#)。

例

```
Console> clear-tls-config
Cleared Transport Layer Security Configuration
Run clear-tls-config on the other appliances in the collective and restart each appliance
for changes to take effect
```

親トピック: [特殊目的コマンド](#)

collect-logs

このコマンドは推奨されません。代わりに、[platform must-gather](#) を使用してください。

親トピック: [特殊目的コマンド](#)

component firmware update

目的

コンポーネントの現行のファームウェア・バージョンが構成されたバージョンと一致しない場合に、ファームウェア・バージョンを更新します。

構文

component firmware update component type

パラメーター

component type

- コンポーネントのタイプ名。サポートされるコンポーネントのタイプの完全なリストを取得するには、このコマンドのヘルプを参照してください。
- [show components](#) を使用していて、コンポーネントのファームウェア・バージョンが構成されたバージョンと一致しない場合、このコマンドを使用してファームウェア・バージョンを更新できます。

関連コマンド

[show components](#) を参照してください。

例

BMC のコンポーネント・ファームウェアを更新します。

```
Console> component firmware update bmc
CWZBR03003I: Are you sure you want to update component firmware for bmc
After update, device will reboot
CWZBR03012I: yes/no:yes
CWZBR03013I: Component firmware update is starting.....
CWZBR03006I: Component firmware update succeeded and reboot device
```

親トピック: [特殊目的コマンド](#)

config

目的

アプライアンスの構成をインポートまたはエクスポートします。

構文

```
config <export|import|usage|listExports> [-file] <filename> [-keyStore] <keystore>
[-trustStore] <truststore> [-silent] [-noRestart]
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

-truststore

-trustStore フラグの後に、アプライアンスの構成の一部であるトラストストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-keystore

-keyStore フラグの後に、アプライアンスの構成の一部である鍵ストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-noRestart

特定のアプライアンス設定では、アプライアンスを再始動することが要求されます。このフラグを設定すると、アプライアンスはどれも再始動されません。しかし、**-noRestart** パラメーターを設定した場合には、再始動が要求される設定を保存するために、アプライアンスを手動で再始動する必要があります。

-silent

これが指定されると、**config** コマンドの状況メッセージが表示されません。これはデフォルトの動作です。

使用上の注意

なし

関連コマンド

このコマンドには、以下のサブコマンドがあります。[usage](#)、[export](#)、[import](#)、[listExports](#)、および [deleteExport](#) を参照してください。

例

アプライアンスの構成をインポートまたはエクスポートします。

```
config export -file foo.json -silent
config import -file foo.json -keyStore keyStore.jks -trustStore trustStore.jks
config import -file foo.json -noRestart
config usage
config listExports
config deleteExport export_1942-09-06_13-50-14.036.json
```

親トピック: [特殊目的コマンド](#)

datetime get

目的

アプライアンスに構成された時刻を、グリニッジ標準時と地方時の両方で表示します。

構文

datetime get

パラメーター

なし。

関連コマンド

[datetime set](#)、[timezone get](#)、および [timezone set](#) を参照してください。

例

アプライアンスに構成された時刻を表示します。

```
Console> datetime get
GMT: 2012-03-08T20:20:21Z
Local: Mar 8, 2012 3:20:21 PM
Console>
```

親トピック: [特殊目的コマンド](#)

datetime set

目的

アプライアンスに現在時刻を構成します。

構文

datetime set *time*

パラメーター

time

- 地方時 (時間帯設定により決まる) での現在時刻
- 形式は以下の仕様による YYYY-MM-DD hh:mm:ss または YYYY-MM-DD hh:mm:ssZ です。

YYYY

年を示します

MM

月を示します

DD

日付を示します

hh

時を示します

mm

分を示します

ss

秒を示します

Z

時刻に付加してグリニッジ標準時 (GMT) を指定します。

関連コマンド

[datetime get](#)、[timezone get](#)、および [timezone set](#) を参照してください。

例

構成された時間帯を表示し、アプライアンスの現在時刻を設定し、それから構成された時刻を表示します。

```
Console> timezone get
Timezone is EST
Console> datetime set 2012-03-22 10:01:00
Console> datetime get
GMT: 2012-03-22T15:01:03Z
Local: Mar 22, 2012 10:01:03 PM
Console>
```

親トピック: [特殊目的コマンド](#)

deleteExport

目的

指定されたエクスポート・ファイルを、ローカルに保管されたエクスポートのリストから削除します。

構文

```
config deleteExport
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

使用上の注意

deleteExport コマンドを受け渡し、**-file** フラグの付いたファイルを 1 つのみ指定する必要があります。

例

エクスポートされたアプライアンスの構成を削除します。

```
config deleteExport export_1942-09-06_13-50-14.036.json
```

親トピック: [特殊目的コマンド](#)

device RESET

目的

アプライアンスのすべてのデータを消去して、工場出荷状態に戻します。SSH サーバー鍵、ライセンス同意、およびログを含めて、すべての構成が削除されます。すべてのユーザー ID とパスワードも削除され、工場で設定されたデフォルトのユーザー ID とパスワードに置換されます。アプライアンスはこのコマンドを実行した後で自動的にリブートします。

コマンドの「RESET」部分は、間違って入力してしまう可能性を減らすために、すべて大文字で入力する必要があります。このコマンドは、このデバイスで実行されているファームウェアを変更しないため、**firmware pristine-install** とは異なります。したがって、このコマンドを実行するときにファームウェア・イメージは必要ありません。このコマンドは構成のみを変更します。

構文

device RESET [noprompt]

オプション

noprompt

確認のプロンプトを表示しません。このコマンドはすぐに開始されます。このオプションが存在しない場合、次に表示される確認プロンプトに対してユーザーが肯定的に回答しない限り、コマンドは開始されません。

パラメーター

なし。

使用上の注意

ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、構成がクリアされるたびにネットワーク接続が終了します。このコマンドのすべてのメッセージを表示したい場合には、シリアル・コンソールに接続する必要があります。

関連コマンド

[firmware pristine-install](#) を参照してください。

例

```
Console> device RESET

This command will result in the network configuration being reset; this may end your
current connection to the appliance.

Undo all existing configuration and reset the appliance to factory-new condition? y/n: y
You have chosen to reset the device, proceeding with reset...
Reset complete; ready to reboot
Console>
```

親トピック: [特殊目的コマンド](#)

device battery-replaced

目的

内部バッテリーが交換された日付を記録します。バッテリーは、アプライアンスが電源オフで AC 電源に接続していないときに、ハードウェア構成と RAID ディスク書き込み可能キャッシュにバックアップ電源を提供します。

バッテリーは特定の期間有効であり、このコマンドはその期間の開始を記録します。このコマンドは工場で行われており、ユーザーが実行する必要があるのは現場でバッテリーが交換された場合のみです。既存のバッテリーの現行の取り付け日と有効期限は、**status battery** コマンドで表示できます。

構文

device battery-replaced

オプション

なし。

パラメーター

なし。

関連コマンド

[status battery](#) を参照してください。

例

```
Console> device battery-replaced
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device intrusion allow

目的

- ケースを開くことが許可されるデバイスにおいて、それを許可します。
- [device intrusion clear](#) を参照してください。

構文

device intrusion allow

パラメーター

なし。

例

```
Console> device intrusion allow
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device intrusion clear

目的

- アプライアンスのケースが物理的に開いている場合に、記録されたデバイスの侵入状態をクリアし、ケースが侵入されたと思われなくなるようにします。
- アプライアンスのお客様交換可能部品や現場交換可能部品の一部には、ケースを開くことが必要なものがあります。部品を交換した後でこのコマンドを実行して、ケースを閉じます。

構文

device intrusion clear

パラメーター

なし。

使用上の注意

- このコマンドは、7198 または 7199 マシン・タイプのみで有効です。
- 9235 マシン・タイプでケースを開くと、アプライアンスが永続的に操作不能になり、アプライアンスを工場に戻さなければならなくなるので、9235 マシン・タイプのケースを開いてはいけません。これらの物理アプライアンスには、ケースが開かれたときに検出するセンサーがあります。物理アプライアンスの他のマシン・タイプでは、このコマンドは作用しません。このコマンドは仮想アプライアンスには適用されません。
- [status intrusion](#) も参照してください。

例

```
Console> device intrusion clear
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device intrusion disallow

目的

以前に [device intrusion allow](#) を呼び出した後で、侵入検出を再び使用可能にします。

構文

device intrusion disallow

パラメーター

なし。

使用上の注意

- このコマンドは、マシン・タイプ 7198 および 7199 にのみ作用します。詳細については、[device intrusion clear](#)を参照してください。
- このコマンドは、侵入センサーのリセットを行いません。一般に、アプライアンスをリブートする前に [device intrusion clear](#) を呼び出すことをお勧めします。

例

```
Console> device intrusion disallow
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device nvdimm clear

目的

nvDIMM ボリューム・ラベルをクリアします。

構文

device nvdimm clear

パラメーター

なし。

使用上の注意

このコマンドは 9006 マシン・タイプでのみ許可されています。

関連コマンド

なし。

例

```
Console> device nvdimm clear
CWZBR02923I: Volume labels cleared
Console>
```

親トピック: [特殊目的コマンド](#)

device clear-intrusion

このコマンドは推奨されません。代わりに、[device intrusion clear](#) を使用してください。

親トピック: [特殊目的コマンド](#)

device restart

目的

アプライアンスをシャットダウンして、リブートします。リブート時に、構成されたすべてのアプリケーションが開始します。これは、電源を入れ直さずにアプライアンスをリブートする、望ましい方法です。

構文

device restart

オプション

なし。

パラメーター

なし。

使用上の注意

ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、シャットダウン中にネットワーク接続が終了します。再始動サイクル全体のすべてのメッセージを表示したい場合には、シリアル・コンソールに接続する必要があります。

関連コマンド

[device shutdown](#) を参照してください。

例

```
Console> device restart
Ok
Console>
Preparing system for shutdown: s1 hb hn 2 0 s2 rs nt hc ns sy ul r6
machine restart
Starting system... dn un nvm 7198 r1 r2 lo wf lu pt tc sr in cd ln mu ku up us
t1 t2 md mm mcg mt ml na fi
login:
```

親トピック: [特殊目的コマンド](#)

device shutdown

目的

アプライアンスをシャットダウンして、電源オフ状態のままにします。自動的にリブートしません。このコマンドの実行後にアプライアンスをリブートしたい場合には、電源ボタンが単一プッシュ・ボタンの場合はボタンを押し、電源ボタンが2ポジションのロッカー・スイッチの場合はオンに切り替えます。これは、電源ボタンのみを使用するのではなくアプライアンスをシャットダウンする、望ましい方法です。

構文

device shutdown

オプション

なし。

パラメーター

なし。

使用上の注意

ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、シャットダウン中にネットワーク接続が終了します。シャットダウン・サイクル全体のすべてのメッセージを表示したい場合には、シリアル・コンソールに接続する必要があります。

関連コマンド

[device restart](#) を参照してください。

例

```
Console> device shutdown
Ok
Console>
Preparing system for shutdown:  s1  hb  hn  2  0  s2  rs  nt  hc  ns  sy  ul  r0
machine shutdown
```

親トピック: [特殊目的コマンド](#)

echo

目的

コマンド行インターフェース (CLI) の出力ストリームに書き込みます。

構文

echo *text*

パラメーター

text

コマンド行インターフェース (CLI) のストリームに書き込むテキスト。このコマンドが発行されたコマンド行インターフェース (CLI) のインスタンスにのみ表示されます。

使用上の注意

echo コマンドは、コマンド行インターフェース (CLI) のスクリプトで役に立ちます。

例

echo コマンドを使用して、**datetime get** 表示の出力を前に付加します。

```
Console> echo Current time is: ; datetime get
Current time is:
GMT: 2012-03-22T15:32:06Z
Local: Mar 22, 2012 3:32:06 PM
Console>
```

親トピック: [特殊目的コマンド](#)

export

目的

アプライアンス構成をローカルに、またはリモート・ファイル・サーバーにアップロードするための一時ロケーションに、エクスポートします。

構文

```
config export
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

使用上の注意

-file フラグを export コマンドに受け渡す必要があります。

例

アプライアンス構成をエクスポートします。

```
config export -file foo.json -silent
```

親トピック: [特殊目的コマンド](#)

file delete

目的

アプリケーションからファイルを削除します。

構文

```
file delete filename
```

パラメーター

filename

削除されるファイル

関連コマンド

[file get](#)、[file list](#)、および [file put](#) を参照してください。

例

platform collect-pd コマンドによって以前に作成されたファイルを削除します。

```
Console> file list  
collect-pd.txt 103576 bytes created Mar 22, 2012 3:38:32 PM  
Console> file delete collect-pd.txt  
Console>
```

親トピック: [特殊目的コマンド](#)

file get

目的

リモート・ロケーションからアプライアンスにファイルをコピーします。

構文

```
file get url localfile
```

パラメーター

url

以下の2つの形式のうちの1つで指定された Uniform Resource Locator。以下の形式では、読みやすくするための空白文字が付いています。コマンドの URL には空白文字を含めないでください。

- プロトコル HTTP および FTP の場合:

```
protocol: // [username [:password] @]hostname [:port]/[%2F abspath/][relpath /] filename
```

ここで、

protocol

FTP または HTTP。

username

リモート・ホストのユーザー名。これは URL のオプション・コンポーネントです。

password

リモート・ホストでの *username* のパスワード。これは URL のオプション・コンポーネントです。

hostname

リモート・ホスト名または IP アドレス。

port

リモート・サーバーの 10 進数のポート番号。これは URL のオプション・コンポーネントです。

abspath

'%2F' または '%2f' の 3 文字の接頭部がついた、リモート・ファイルの絶対パス。これはプロトコル FTP および SCP でサポートされ、URL のオプション・コンポーネントです。

relpath

リモート・ファイルの相対パス。このパスは、リモート・システムのデフォルト・ディレクトリーに対して相対です。*username* が指定されている場合、デフォルト・ディレクトリーは一般にそのユーザーのホーム・ディレクトリーです。このパスは、URL のオプション・コンポーネントです。

filename

リモート・ファイル名。

- プロトコル SCP の場合:

```
protocol: // username@hostname: [%2F abspath/][relpath /] filename
```

ここで、

protocol

SCP。

username

リモート・ホストのユーザー名。これは URL のオプション・コンポーネントです。

hostname

リモート・ホスト名または IP アドレス。IPv6 アドレスに解決するホスト名がサポートされます。しかし、*hostname* がファイル・パスとコロンの区切られるため、コロンを含むリテラル IPv6 アドレスはサポートされません。

abspath

'%2F' または '%2f' の 3 文字の接頭部がついた、リモート・ファイルの絶対パス。これはプロトコル FTP および SCP でサポートされ、URL のオプション・コンポーネントです。

relpath

リモート・ファイルの相対パス。このパスは、リモート・システムのデフォルト・ディレクトリーに対して相

対です。 *username* が指定されている場合、デフォルト・ディレクトリーは一般にそのユーザーのホーム・ディレクトリーです。このパスは、URL のオプション・コンポーネントです。

filename

リモート・ファイル名。

localfile

ファイルのローカル・コピーに割り当てられる名前。絶対パス・エレメント (頭にスラッシュが付いている) またはサブディレクトリー (スラッシュ) は許可されません。相対の、階層なしの名前でなければなりません。

関連コマンド

[file delete](#)、[file list](#)、および [file put](#) を参照してください。

例

リモート・ロケーションからファームウェア・ファイルを取得して、アプライアンスで newfirmware という名前を付けます。

```
Console> file get scp://user1@server1.raleigh.ibm.com:~/dev_bedrock.scrpt2
newfirmware.scrpt2
user1@server1.raleigh.ibm.com's password:
dev_bedrock.scrpt2          100% 136MB 34.0MB/s 00:04
Wrote 142553856 bytes to local storage
Console>
```

親トピック: [特殊目的コマンド](#)

file list

目的

アプライアンス上のファイルをリストします。

構文

file list

パラメーター

なし。

関連コマンド

[file delete](#)、[file get](#)、および [file put](#) を参照してください。

例

現在アプライアンス上にあるファイルを表示します。

```
Console> file list
collect-pd.txt 103794 bytes created Mar 22, 2012 5:30:56 PM
newfirmware 142553856 bytes created Mar 22, 2012 4:41:17 PM
Console>
```

親トピック: [特殊目的コマンド](#)

file put

目的

アプライアンスからリモート・ロケーションにファイルをコピーします。

構文

```
file put localfile url
```

パラメーター

localfile

転送されるローカル・ファイルの名前。このファイルは存在している必要があります。

url

[file get](#) に記載された、Uniform Resource Locator。ただし、**file put** コマンドでは HTTP プロトコルはサポートされません。

関連コマンド

[file delete](#)、[file get](#)、および [file list](#) を参照してください。

例

ローカル・ファイル collect-pd.txt を diag.txt としてリモート・システムにコピーします。

```
Console> file put collect-pd.txt scp://user1@system1.rtp.raleigh.ibm.com:~/diag.txt
user1@system1.rtp.raleigh.ibm.com's password:
collect-pd.txt                100% 101KB 101.4KB/s   00:00
Console>
```

親トピック: [特殊目的コマンド](#)

firmware pristine-install

目的

ファームウェアをインストールし、アプライアンスを工場出荷時に近い状態にリストアします。

構文

firmware pristine-install *image*

パラメーター

image

インストールするファームウェア・イメージ (scrypt2 ファイルの形式)

使用上の注意

- **pristine-install** コマンドは、一般的なファームウェアのアップグレードにコピーされる構成やデータ・ファイルのコピーせずに、新規ファームウェア・イメージをインストールします。
- **pristine-install** コマンドは、アプライアンスに古いバージョンのファームウェアをインストールするのに使用できます。
- **pristine-install** コマンドは、影響がより少ない方法ではうまくいかなかった場合に、エラー状況のリカバリーに使用してみることもできます。
- この操作の一部としてアプライアンスがリブートされるので、ネットワーク接続は中断されます。したがって、通常シリアル・コンソールからこのコマンドを実行することが推奨されます。

関連コマンド

[firmware upgrade](#)、[firmware rollback](#)、および [device RESET](#) を参照してください。

例

ファームウェアを古いファームウェア・レベルに戻します。

```
Console> firmware pristine-install oldfirmware.scrypt2
Upgrading firmware...
Verifying image signature
Executing dynamic loader
Executing dynamic loader
Validating image
Extracting firmware from image
Extracting firmware manifest
Executing pre-installation
Deleting previous installation
Linking common files.
Extracting files
Verifying installation
Copying configuration from existing installation
Switching to new installation
Upgrade or rollback succeeded. Rebooting
```

親トピック: [特殊目的コマンド](#)

firmware rollback

目的

ファームウェア・レベルを、アプライアンスにインストールされたファームウェアの以前のレベルに戻します。

構文

firmware rollback

パラメーター

なし。

使用上の注意

- アプライアンスは最大で 2 レベル (アクティブおよび代替) のファームウェアを保持できます。 **firmware rollback** コマンドを呼び出す都度、2 つのイメージが交互に切り替わります。
- このコマンドは構成をロールバックします。ロールバック後、構成は最後のアップグレード前の状態になります。
- このコマンドはコンポーネント・ファームウェアをロールバックします。各コンポーネントが保持できるコンポーネント・ファームウェアは 1 コピーのみであるため、ロールバックされたバージョンのコンポーネント・ファームウェアがコンポーネントに再インストールされます。
- この操作の一部としてアプライアンスがリブートされるので、ネットワーク接続は中断されます。したがって、通常シリアル・コンソールからこのコマンドを実行することが推奨されます。

関連コマンド

[firmware upgrade](#)、[firmware pristine-install](#)、および [device RESET](#) を参照してください。

例

アプライアンスを、インストールされている以前のバージョンのファームウェアに戻します。

```
Console> firmware rollback
Rolling back firmware...
Upgrade or rollback succeeded. Rebooting...
```

親トピック: [特殊目的コマンド](#)

firmware upgrade

目的

アプライアンスに新規レベルのファームウェアをインストールします。

構文

firmware upgrade *image*

パラメーター

image

インストールするファームウェア・イメージ (scrypt2 ファイルの形式)

使用上の注意

この操作の一部としてアプライアンスがリブートされるので、ネットワーク接続は中断されます。したがって、通常シリアル・コンソールからこのコマンドを実行することが推奨されます。

関連コマンド

[firmware pristine-install](#)、[firmware rollback](#)、および [device RESET](#) を参照してください。

例

アプライアンスに新規レベルのファームウェアをインストールします。

```
Console> firmware upgrade myfirmware.scrypt2
Upgrading firmware...
Verifying image signature
Executing dynamic loader
Validating image
Extracting firmware from image
Extracting firmware manifest
Executing pre-installation
Deleting previous installation
Linking common files
Extracting files
Verifying installation
Copying configuration from existing installation
Switching to new installation
Upgrade or rollback succeeded. Rebooting
```

親トピック: [特殊目的コマンド](#)

force recycle

目的

データを何も保存せずに、WebSphere DataPower XC10 アプライアンスのプロセスを再始動します。データ損失が発生する可能性があるため、データ損失があっても問題ない場合、または device restart コマンドを試してもアプライアンスが使用可能にならなかった場合に限り、このコマンドを実行してください。

構文

force recycle

パラメーター

なし。

関連コマンド

[device restart](#)

例

```
Console> force-recycle
Force Stopped all XC-10 processes
Forced recycle of appliance
```

親トピック: [特殊目的コマンド](#)

get-dns-search

目的

構成されたドメイン・ネーム・サーバーの検索ドメインを表示します。

構文

get-dns-search

パラメーター

なし。

関連コマンド

[set-dns-search](#)、[get-dns-servers](#)、および [set-dns-servers](#) を参照してください。

例

ドメイン・ネーム・サーバーの検索ドメインを構成してから表示します。

```
Console> set-dns-search short.example.com example.com
Ok
Console> get-dns-search
DNS search domains:
    short.example.com
    example.com
Console>
```

親トピック: [特殊目的コマンド](#)

get-dns-servers

目的

構成されたドメイン・ネーム・サーバーの IP アドレスを表示します。

構文

get-dns-servers

パラメーター

なし。

関連コマンド

[set-dns-servers](#)、[get-dns-search](#)、および [set-dns-search](#) を参照してください。

例

ドメイン・ネーム・サーバーを構成してから表示します。

```
Console> set-dns-servers 9.42.106.2 2002:92a:8f7a:106:9:42:106:42
Ok
Console> get-dns-servers
Domain (DNS) servers:
    9.42.106.2
    2002:92a:8f7a:106:9:42:106:42
Console>
```

親トピック: [特殊目的コマンド](#)

get-ntp-servers

目的

構成された Network Time Protocol サーバーを表示します。

構文

get-ntp-servers

パラメーター

なし。

関連コマンド

[set-ntp-servers](#) を参照してください。

例

NTP サーバーを構成してから表示します。

```
Console> set-ntp-servers example.server.com 127.0.0.1
Ok
Console> get-ntp-servers
NTP servers:   example.server.com
               127.0.0.1
Console>
```

親トピック: [特殊目的コマンド](#)

help

目的

コマンドのヘルプ情報を表示します。

構文

- サポートされるコマンドのリストを表示する: **help command**
- 特定のコマンドのヘルプを表示する: **help command**

パラメーター

command

コマンド行インターフェース (CLI) コマンド

例

platform コマンドのサブコマンドを表示します。

```
Console> help platform
The following platform commands are available:
  platform collect-pd [PDFfilename | console]
  platform log-level ...
  platform must-gather filename [PDFfilename]
```

親トピック: [特殊目的コマンド](#)

import

目的

アプライアンス構成をインポートします。

構文

```
config import
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

-truststore

-trustStore フラグの後に、アプライアンスの構成の一部であるトラストストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-keystore

-keyStore フラグの後に、アプライアンスの構成の一部である鍵ストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-noRestart

特定のアプライアンス設定では、アプライアンスを再始動することが要求されます。このフラグを設定すると、アプライアンスはどれも再始動されません。しかし、**-noRestart** パラメーターを設定した場合には、再始動が要求される設定を保存するために、アプライアンスを手動で再始動する必要があります。

-silent

これが指定されると、**config** コマンドの状況メッセージが表示されません。これはデフォルトの動作です。

使用上の注意

-file フラグの付いた **import** コマンドを受け渡す必要があります。

例

アプライアンス構成をインポートします。

```
config import -file foo.json -silent
```

親トピック: [特殊目的コマンド](#)

license accept

目的

ライセンスを表示し、コマンド行インターフェース (CLI) ユーザーがライセンスに同意するか拒否するよう要求します。

構文

license accept

パラメーター

なし。

例

ライセンス同意プロセスを実行します。

```
Console> license accept
Bedrock Project, WebSphere Technical Institute
Copyright 2009-2010, IBM Corporation
```

```
This is your sample license.
```

```
Accept; Reject: accept
Console>
```

親トピック: [特殊目的コマンド](#)

license get

目的

アプライアンスに新規または変更されたライセンスを転送します。

構文

license get *url*

パラメーター

url

取得するライセンスのリモート・ロケーション

使用上の注意

URL は以下のプロトコルのうちの 1 つで指定する必要があります: HTTP、FTP、または SCP。URL 構文の説明については、[URL 構文](#)を参照してください。

例

新規ライセンス lic-test をライセンス・リポジトリに取得します。

```
Console> license get scp://user1@server1.raleigh.ibm.com:~/temp/lic
user1@server1.raleigh.ibm.com's password:
lic                               100%   14     0.0KB/s   00:00
Wrote 14 bytes to local storage
Console>
```

親トピック: [特殊目的コマンド](#)

license reset

目的

ライセンス同意状況をリセットします。

構文

license reset

パラメーター

なし。

使用上の注意

コマンド行インターフェースを次に呼び出すには、ユーザーがライセンスに同意する必要があります。

例

ライセンス同意状況をリセットします。

```
Console> license reset  
Ok
```

親トピック: [特殊目的コマンド](#)

listExports

目的

ローカルに保管された、エクスポートされた構成のリストを表示します。

構文

```
config listExports
```

パラメーター

なし

使用上の注意

なし

例

エクスポートされたアプライアンスの構成をリストします。

```
config listExports
```

親トピック: [特殊目的コマンド](#)

locale get

目的

コマンド行インターフェース (CLI) でメッセージの表示の制御に使用されるロケール設定を取得します。

構文

locale get

パラメーター

なし。

例

ロケール設定を取得します。

```
Console> locale get
Locale is en_US
Console>
```

親トピック: [特殊目的コマンド](#)

locale set

目的

コマンド行インターフェース (CLI) でメッセージの表示に使用されるロケールを設定します。

構文

```
locale set language [country]
```

パラメーター

language

言語の省略形

country

国の省略形

使用上の注意

ロケール設定は、コマンドが発行された CLI ですぐに有効になります。既にアクティブな可能性のある別の CLI インスタンスでは、新規ロケール設定を有効にするために、ユーザーは CLI を終了して再入力する必要があります。

例

ロケールを米国の表記規則を使用して英語に設定します。

```
Console> locale set en US  
Console>
```

親トピック: [特殊目的コマンド](#)

locate-led

目的

場所の特定 LED をオンまたはオフにします。場所の特定 LED は物理アプライアンスの前部にあるライトです。灯台の形のアイコンで表示され、LED の色は青です。場所の特定 LED の目的は、データ・センターの特定のアプライアンスを視覚的にお客様に見つけやすくすることです。場所の特定 LED はデフォルトでオフであり、このコマンドを使用して明示的にオンにする必要があります。LED の状態は持続しないので、アプライアンスがリブートした後で、場所の特定 LED はオフになります。場所の特定 LED は、手動で視覚的に位置を示す以外の機能はありません。これは物理アプライアンスにのみ適用されます。仮想アプライアンスには場所の特定 LED はありません。

構文

```
locate-led { on | off }
```

オプション

なし。

パラメーター

on

場所の特定 LED をオンにします。

off

場所の特定 LED をオフにします。

使用上の注意

場所の特定 LED の状態を照会することはできません。場所の特定 LED の状態を、現在なっている状態に設定することはエラーとは見なされません。

例

```
Console> locate-led on
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

log message

目的

- テキスト・ストリングを診断ログ・ファイルに書き込みます。
- 例えば、このコマンドを使用して、特定のアクティビティの開始または終了を示します。ログ・ファイルの手動による検査や自動化された後処理で、このテキストを見つけることができ、さらに分析するための開始点として使用できます。
- このメッセージは、INFO (通知) レベルで記録されます。

構文

log message *text*

パラメーター

text

ログに書き込むメッセージ・テキスト

例

```
Console> log message About to run the acceptance tests
Console>
```

親トピック: [特殊目的のコマンド](#)

netif

netif コマンドは推奨されず、デフォルトでは存在しなくなりました。

代わりに、ethernet-interface 構成オブジェクトの [共通コマンド](#) に記載されているコマンドを使用してください。

親トピック: [特殊目的コマンド](#)

net-test available

目的

ネットワーク・インターフェースにアクティブなキャリアがあるかどうかをテストします。

構文

net-test available

パラメーター

なし。

関連コマンド

他の診断コマンドについては、[net-test dns](#)、[net-test ping](#)、および [net-test tcp](#) を参照してください。

例

アクティブなネットワーク・インターフェースの使用可能性をテストします。

```
Console> net-test available
Network available
Console>
```

親トピック: [特殊目的コマンド](#)

net-test dns

目的

指定されたホスト名でドメイン・ネーム・サーバー検索を行います。

構文

```
net-test dns hostname
```

パラメーター

hostname

ドメイン・ネーム・サーバー検索で使用するホスト名

関連コマンド

[net-test available](#)、[net-test ping](#)、および [net-test tcp](#) を参照してください。

例

system1.rtp.ibm.com で DNS 検索を行います。

```
Console> net-test dns system1.rtp.raleigh.ibm.com
9.42.77.42
Console>
```

親トピック: [特殊目的コマンド](#)

net-test ping

目的

指定されたホスト名または IP アドレスへの接続をテストします。

構文

```
net-test ping host
```

パラメーター

host

リモート・システムのホスト名または IP アドレス

関連コマンド

他の診断コマンドについては、[net-test available](#)、[net-test dns](#)、および [net-test tcp](#) を参照してください。

例

指定されたホスト名への接続をテストします。リモート・システムに到達できます。

```
Console> net-test ping system1.rtp.raleigh.ibm.com
Ok
Console>
```

指定されたアドレスへの接続をテストします。リモート・システムに到達できません。

```
Console> net-test ping 4.4.4.4
ping failed
Console>
```

親トピック: [特殊目的コマンド](#)

net-test tcp

目的

指定されたホスト名または IP アドレスにあるポートへの TCP 接続を開くことができるかをテストします。

正常に開かれた場合、TCP 接続経由でデータは送信されず、閉じられます。

構文

```
net-test tcp host port
```

パラメーター

host

リモート・システムのホスト名または IP アドレス

port

リモート・システムの TCP ポート。

関連コマンド

他の診断コマンドについては、[net-test available](#)、[net-test dns](#)、および [net-test ping](#) を参照してください。

例

ホスト system1.rtp.raleigh.ibm.com のポート 80 に到達できるかをテストします。

```
Console> net-test tcp system1.rtp.raleigh.ibm.com 80
Ok
```

親トピック: [特殊目的コマンド](#)

net-test traceroute

目的

指定されたホスト名または IP アドレスへのパスをレポートします。

正常に開かれた場合、TCP 接続経由でデータは送信されず、閉じられます。

構文

```
net-test traceroute host
```

パラメーター

host

リモート・システムのホスト名または IP アドレス

関連コマンド

他の診断コマンドについては、[net-test available](#)、[net-test dns](#)、[net-test ping](#)、および [net-test tcp](#) を参照してください。

例

指定されたホストへのルートを判別します。

```
Console> net-test traceroute localhost
localhost (127.0.0.1) 0.097 ms
Ok
Console>
```

```
Console> net-test traceroute 127.0.0.1
localhost (127.0.0.1) 0.099 ms
Ok
Console>
```

指定されたアドレスへの接続をテストします。リモート・システムに到達できません。

```
Console> net-test traceroute host.that.does.not.exist
bad address 'host.that.does.not.exist'
traceroute failed
Console>
Console> net-test traceroute 9.65.47.137
sendto: Network is unreachable
traceroute failed
```

親トピック: [特殊目的コマンド](#)

nodename get

目的

アプライアンスの構成名を取得します。

構文

nodename get

パラメーター

なし。

関連コマンド

[nodename set](#) を参照してください。

例

アプライアンス名を設定してから取得します。

```
Console> nodename set appliance1
Console> nodename get
nodename is appliance1
Console>
```

親トピック: [特殊目的コマンド](#)

nodename set

目的

アプライアンスの構成名を設定します。

構文

```
nodename set name
```

パラメーター

name

アプライアンスの名前

関連コマンド

[nodename get](#) を参照してください。

例

アプライアンス名を設定してから取得します。

```
Console> nodename set appliance1
Console> nodename get
nodename is appliance1
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture clear

目的

指定されたネットワーク・インターフェースのパケット・トレース・キャプチャー・ファイルを削除します。

構文

```
packet-capture clear interface
```

パラメーター

interface

パケット・トレース・ファイルをクリアする必要があるネットワーク・インターフェース

使用上の注意

- インターフェースでパケット・トレースが実行された後、同じインターフェースで別のパケット・キャプチャーを開始するためには、その前に、ファイルをクリアする必要があります。

関連コマンド

[packet-capture list](#)、[packet-capture start](#)、および [packet-capture stop](#) を参照してください。

例

パケット・キャプチャー・アクティビティーを表示し、それからインターフェース eth0 で収集されたファイルをクリアします。

```
Console> packet-capture list
interface      status          file(s)
eth1    Finished      eth1pc1
                               eth1pc2
-----
eth0    Finished      eth0pc
Ok
Console> packet-capture clear eth0
Packet capture dump files cleared for eth0
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture list

目的

アプライアンスのネットワーク・キャプチャー・アクティビティを表示します。

構文

packet-capture list

パラメーター

なし。

関連コマンド

[packet-capture clear](#)、[packet-capture start](#)、および [packet-capture stop](#) を参照してください。

例

アプライアンスのネットワーク・キャプチャー・アクティビティを表示します。この例では、インターフェース **eth0** のネットワーク・キャプチャーが完了しています。インターフェース **mgt0** のトレースは、リング・バッファ・ファイル 3 つにまだアクティブに書き込み中です。

```
Console> packet-capture list
interface      status          file(s)
eth0    Finished      eth0pc
-----
mgt0    Capturing     mgt0pc1
                               mgt0pc2
                               mgt0pc3
-----
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture start

目的

パケット・トレース収集をアクティブ化します。

構文

```
packet-capture start interface filename duration filesize [ ring-buffer-number ] ["filter "]
```

パラメーター

interface

ネットワーク・インターフェース名 (例えば **eth0**)

filename

出力トレース・ファイル名。ring-buffer-number も指定されている場合は、ファイル名に索引番号が追加されず。

duration

パケット・キャプチャーの期間 (単位は秒)。値 **0** は、**packet-capture stop** コマンドが発行されるまでパケット・キャプチャーを続行することを示します。

filesize

トレース・ファイルのサイズ (単位はメガバイト (MB))。値は 0 から 100 の範囲でなければなりません。

ring-buffer-number

指定されると、リング・バッファ機能が使用可能に設定され、リング・バッファ・ファイルの数が設定されます。値は 0 から 10 の範囲でなければなりません。

filter

収集されるパケットを選択するのに使用するフィルター (二重引用符 (" ") で囲む)。拡張フィルター機能がサポートされ、例えばパケット・ソース、宛先、ポート、およびプロトコルのフィルタリングが含まれます。サポートされるフィルター構文の完全な詳細については、インターネットで pcap-filter を検索してください。

使用上の注意

- インターフェースでパケット・キャプチャーを開始するためには、以下の条件が必要です。
 - インターフェースはアクティブであることが必要です。
 - インターフェースで他のパケット・キャプチャーはアクティブにはできません。
 - aggregate-interface にトレースが要求される場合、aggregate-interface 自体でトレースを開始します。メンバー・リンクのパケット・キャプチャーは完全なデータを含まない場合があります。
 - 以前のパケット・キャプチャーからのファイルはすべて、**packet-capture clear** コマンドを使用してクリアする必要があります。
- パケット・キャプチャーは、以下のいずれかの条件が発生した場合に停止します。
 - リング・バッファ機能が使用されていない場合、ファイル・サイズの制限に達したとき。
 - パケット・キャプチャー期間が終了したとき。
 - **packet-capture stop** コマンドが発行されたとき。
- 出力トレース・ファイルは、Wireshark など、libpcap フォーマットのデータを読むことができる市販の既製ツールで分析できます。

関連コマンド

[packet-capture clear](#)、[packet-capture list](#)、および [packet-capture stop](#) を参照してください。

例

インターフェース eth1 でパケット・キャプチャーを開始して 60 秒間行います。トレース・ファイルのサイズは最大 10 MB まで可能です。パケット・キャプチャーを pc11 と pc12 という名前の 2 つのリング・バッファ・ファイルに書き込みます。

```
Console> packet-capture start eth1 pc1 60 10 2 "icmp"
Packet capture started on eth1
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture stop

目的

指定されたネットワーク・インターフェースで進行中のパケット・キャプチャーを停止します。

構文

```
packet-capture stop interface
```

パラメーター

interface

パケット・キャプチャーを停止する必要があるネットワーク・インターフェース

使用上の注意

- パケット・キャプチャー・ファイルは、**packet-capture clear** コマンドが発行されるまで存在し続けます。
- パケット・キャプチャー・ファイルを分析用にアプライアンスの外に転送するには、**file put** を使用します。

関連コマンド

[packet-capture clear](#)、[packet-capture list](#)、および [packet-capture start](#) を参照してください。

例

アクティブなパケット・キャプチャーを表示し、その後インターフェース eth1 のパケット・キャプチャー・アクティビティを停止します。

```
Console> packet-capture list
interface      status      file(s)
eth1    Capturing      eth1pc1
                        eth1pc2
-----
eth0    Finished      eth0pc
-----
Ok
Console> packet-capture stop eth1
Stopping capture on eth1
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

platform collect-pd

目的

問題診断データの標準セットを単一のファイルに収集します。このファイルは、表示または IBM サポートに送信するために、アプライアンスから転送できます。この結果できる出力ファイルは、コマンド行インターフェース (CLI) コマンドの事前定義リストの出力です。このコマンドの目的は、サポート担当者が問題診断の一環としてお客様に実行を依頼する可能性がある多数のコマンド行インターフェース (CLI) コマンドを「記す」ことです。

構文

```
platform collect-pd [ PDfilename ]
```

パラメーター

PDfilename

問題診断コマンドの出力が書き込まれるテキスト・ファイルの名前を指定します。指定されない場合、デフォルト値は collect-pd.txt です。出力ファイルが存在する場合、上書きされます。

使用上の注意

- 出力はお客様が読める平文と、IBM サポート向けのみ意図された隠されたテキスト・データの両方を含みます。
- このコマンドの出力は、**platform must-gather** コマンドにより収集されたデータにも自動的に含まれています。そのためユーザーが **platform must-gather** を実行する場合は、別途 **platform collect-pd** を実行する必要はありません。
- 出力はすべて印刷可能なテキストであるため、出力ファイルの内容は、**show log** コマンドを使用してコマンド行インターフェース (CLI) セッション内で表示することができます。これはアプライアンスがすべてのネットワーク接続を失った場合に役立ちます。

関連コマンド

[platform must-gather](#)、[file put](#)、および [show log](#) を参照してください。

例

診断コマンド・データを収集し、出力をデフォルトの出力ファイルに送信します。

```
Console> platform collect-pd
Console> file list
collect-pd.txt 79336 bytes created Feb 6, 2012 9:32:17 PM
Console>
```

親トピック: [特殊目的コマンド](#)

platform log-level get

このコマンドは推奨されません。代わりに、**show logging-settings** を使用してください。

親トピック: [特殊目的コマンド](#)

platform log-level set

このコマンドは推奨されません。代わりに、**edit logging-settings** を使用してください。

親トピック: [特殊目的コマンド](#)

platform service disable ssh

目的

- SSH プラットフォーム・サービスがセキュリティー・ポリシーに影響する場合に、使用不可に設定します。

構文

```
platform service ssh ... platform service ssh disable
```

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service enable ssh](#)、[platform service enable telnet](#)、[platform service disable telnet](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service ssh ...
CWZBR02901I: platform service ssh disable
```

親トピック: [特殊目的コマンド](#)

platform service disable telnet

目的

- Telnet プラットフォーム・サービス・セキュリティー・プロトコルを使用不可に設定します。

構文

platform service telnet ... platform service telnet disable

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service enable telnet](#)、[platform service enable ssh](#)、[platform service disable ssh](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service telnet ...
CWZBR02901I: platform service telnet disable
```

親トピック: [特殊目的コマンド](#)

platform service enable ssh

目的

- SSH プラットフォーム・サービス・セキュリティー・プロトコルを使用可能に設定します。

構文

platform service ssh ... platform service ssh enable

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service disable ssh](#)、[platform service enable telnet](#)、[platform service disable telnet](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service ssh ...
CWZBR02901I: platform service ssh enable
```

親トピック: [特殊目的コマンド](#)

platform service enable telnet

目的

- Telnet プラットフォーム・サービス・セキュリティー・プロトコルを使用可能に設定します。

構文

platform service telnet ... platform service telnet enable

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service disable telnet](#)、[platform service enable ssh](#)、[platform service disable ssh](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service telnet ...
CWZBR02901I: platform service telnet enable
```

親トピック: [特殊目的コマンド](#)

platform must-gather

目的

IBM サポートに問題を報告するために必要な、集めておくべき診断データを収集します。

構文

```
platform must-gather filename [ PDfilename ]
```

パラメーター

filename

すべての診断データを含む出力ファイル (圧縮 (.tgz) 形式)

pdfilename

問題診断コマンドの出力が書き込まれるテキスト・ファイルの名前。このファイルは、platform collect-pd コマンドにより収集された同じ出力を含みます。指定されない場合、デフォルト値は collect-pd.txt です。出力ファイルが存在する場合、ファイルは上書きされます。

使用上の注意

- 出力は平文と、IBM サポートのみが読むように意図された隠されたデータの両方を含むことがあります。

関連コマンド

[platform collect-pd](#) を参照してください。

例

診断データを problem-data.tgz という名前のファイルに収集します。

```
Console> platform must-gather problem-data.tgz
Console> file list
problem-data.tgz 3449276 bytes created Mar 27, 2012 11:15:48 AM
collect-pd.txt 572878 bytes created Mar 27, 2012 11:15:48 AM
Console>
```

親トピック: [特殊目的コマンド](#)

raid delete

目的

RAID カード・コントローラーの構成を削除し、ボリューム情報とパーティション情報もクリーンアップして、物理ディスクを工場出荷状態に戻します。

構文

raid delete

パラメーター

なし。

使用上の注意

- ディスク上のデータは失われます。
- RAID カード・コントローラーの構成は消去され、次のブートで自動的に再構成されます。
- パーティションは除去され、パーティション情報はディスクから消去されます。
- すべての物理ディスクの状態はリセットされます。
- RAID カード・コントローラーにより管理されていたボリュームのみが削除されます。
- 管理対象でないボリュームは削除されません。

例

RAID コントローラーの構成を削除します。

```
Console> raid delete
Delete volume
Delete volume successfully
Console> status volume
No volume
Console> status raid physical
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 0
  Logical drive ID: 0
  Logical drive name: raid0
  State: unconfigured good drive
...
```

親トピック: [特殊目的コマンド](#)

raid re-use

目的

以前別の RAID カード・コントローラーで使用された物理ディスクを、アプライアンスで使用できるようにします。以前に使用されたディスクがアプライアンスに挿入されると、RAID コントローラーにより外部ディスクと認識されることがあります。このコマンドは、ディスクを RAID コントローラーが使用できるようにして、ディスクを RAID アレイに再ビルドするプロセスを自動的に開始します。

構文

raid re-use

パラメーター

なし。

使用上の注意

- 物理ディスク上の古いデータは失われます。
- このコマンドは、使用された物理ディスクの外部状態をクリアし、さらに再ビルド・プロセスをトリガーします。ビルド・プロセスが完了すると、物理ディスクはオンラインに移行します。
- このコマンドは、1つの物理ディスクのみに機能します。複数の外部物理ディスクが同時に挿入された場合、このコマンドはそれらを RAID 構成に入れることができません。

例

使用された物理ディスクの外部状態をクリーンアップし、再ビルドを開始します。

```
Console> raid re-use
Bring physical drive back online after hot swap
Done
Console> status raid physical
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: rebuild
  Progress percent: 3
...
```

親トピック: [特殊目的コマンド](#)

raid foreign-config import

目的

外部構成のインポートは 9005 でサポートされています。外部構成とは、ドライブの交換セットに既に存在する RAID 構成です。ドライブの外部構成は、クリアするか、またはドライブが取り付けられているアプライアンスにインポートすることができます。デフォルトでは、9005 マシンで外部構成が見つかったら、それはクリアされます。ユーザーは必要に応じて、CLI コマンドまたは C++/Java API を使用した外部構成のインポートを使用可能に設定することができます。

構文

raid foreign-config import

パラメーター

なし。

使用上の注意

交換ディスクの外部構成をインポートするには、以下のステップを順番に行います。

1. コマンド **raid foreign-config import** を実行して、外部構成のインポートを使用可能に設定します。
2. アプライアンスをシャットダウンします。
3. ドライブを、外部構成が既に存在するものに交換します。

アプライアンスを電源オンにした後、`status volume` を実行して RAID ボリュームがオンラインになっていることを確認します。そして外部構成が正常にインポートされたかどうか確認します。デフォルトでは、外部構成のインポートは使用不可に設定されています。使用不可に設定されていると、ディスク上のいかなる外部構成もクリアされます。ディスク上のデータも、RAID 構成がクリアされたときに失われます。

関連コマンド

[raid foreign-config clear](#) を参照してください。

例

RAID コントローラーの外部構成をインポートします。

```
Console> raid foreign-config import
CWZBR02804I: Import of foreign configuration is enabled. Foreign configurations will be
imported at reboot.
```

親トピック: [特殊目的コマンド](#)

raid foreign-config clear

目的

外部構成のインポートを使用不可に設定します。外部構成はリブート時にクリアされます。外部構成のインポートは 9005 でサポートされています。外部構成とは、ドライブの交換セットに既に存在する RAID 構成です。ドライブの外部構成は、クリアするか、またはドライブが取り付けられているアプライアンスにインポートすることができます。デフォルトでは、9005 マシンで外部構成が見つかると、それはクリアされます。

構文

```
raid foreign-config clear
```

パラメーター

なし。

関連コマンド

[raid foreign-config import](#) を参照してください。

例

RAID コントローラーの外部構成のインポートを使用不可に設定します。

```
Console> raid foreign-config clear
CWZBR02805I: Import of foreign configuration is disabled. Foreign configurations will be
cleared at reboot.
```

親トピック: [特殊目的のコマンド](#)

request force-suspend

目的

グリッドへのネットワーク接続を維持しながら、グリッド・アクティビティーを突然中断するよう要求します。ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、ネットワーク接続は使用可能なままです。

構文

request force-suspend

オプション

なし。

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[request resume](#) および [request suspend](#) を参照してください。

例

```
Console> request force-suspend
Force Stopped all XC-10 processes
Forced recycle of appliance
[Mon Feb 25 2013 09:46:59] request force-suspend completed
Console> start-progresss
CWZBR02115E: Unknown command "start-progress"
Console> start-progress
DNS 10.42.229.86 10.42.229.86
DNS 10.1.1.105 10.1.1.105
DNS 9.42.94.15 9.42.94.15
SUSPENDED
```

親トピック: [特殊目的コマンド](#)

request resume

目的

以前に **request suspend** または **request force suspend** コマンドのいずれかでグリッド・アクティビティーの中断を要求した場合、このコマンドはグリッド・アクティビティーをリストアします。

構文

request resume

オプション

なし。

パラメーター

なし。

使用上の注意

このコマンドは、アクティビティーを完全にリストアするのに時間がかかります。ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、ネットワーク接続は使用可能なままです。

関連コマンド

[request suspend](#) および [request force-suspend](#) を参照してください。

例

```
Console> request resume
[Mon Feb 25 2013 09:48:09] request resume completed
Console> start-progress
DNS 10.42.229.86 10.42.229.86
DNS 10.1.1.105 10.1.1.105
DNS 9.42.94.15 9.42.94.15
Volume 1 mounted
Volume 2 mounted
STARTING (4% complete)
```

親トピック: [特殊目的コマンド](#)

request suspend

目的

グリッドへのネットワーク接続を維持しながら、グリッド・アクティビティーを安全に中断するよう要求します。ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、ネットワーク接続は使用可能なままです。

構文

request suspend

オプション

なし。

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[request resume](#) および [request force-suspend](#) を参照してください。

例

```
Console> request suspend
Stopped all XC-10 processes
Recycle of appliance
[Mon Feb 25 2013 09:46:59] request suspend completed
Console> start-progresss
CWZBR02115E: Unknown command "start-progress"
Console> start-progress
DNS 10.42.229.86 10.42.229.86
DNS 10.1.1.105 10.1.1.105
DNS 9.42.94.15 9.42.94.15
SUSPENDED
```

親トピック: [特殊目的コマンド](#)

set-dns-search

目的

ドメイン・ネーム・サーバーの検索ドメインを構成します。

構文

```
set-dns-search [domain [... domain ]]
```

パラメーター

domain

ドメイン・ネーム・サーバーの検索ドメイン。

使用上の注意

- ドメインを指定せずに **set-dns-search** コマンドを呼び出すと、リストがクリアされます。
- それぞれのコマンド呼び出しによって、以前に構成されたドメインが置換されます。

関連コマンド

[get-dns-search](#)、[get-ntp-servers](#)、および [set-dns-servers](#) を参照してください。

例

ドメイン・ネーム・サーバーの検索ドメインを構成してから表示します。

```
Console> set-dns-search rtp.raleigh.ibm.com raleigh.ibm.com ibm.com
Ok
Console> get-dns-search
DNS search domains:
    rtp.raleigh.ibm.com
    raleigh.ibm.com
    ibm.com
Console>
```

親トピック: [特殊目的コマンド](#)

set-dns-servers

目的

1つ以上のドメイン・ネーム・サーバーの IP アドレスを構成します。

構文

```
set-dns-servers [server [... server ]]
```

パラメーター

server

ドメイン・ネーム・サーバーの IP アドレス

使用上の注意

- サーバーを指定せずに **set-dns-servers** コマンドを開始すると、リストがクリアされます。
- それぞれのコマンド呼び出しによって、以前に構成されたサーバーが置換されます。

関連コマンド

[get-dns-servers](#)、[get-dns-search](#)、および [set-dns-search](#) を参照してください。

例

ドメイン・ネーム・サーバーを構成してから表示します。

```
Console> set-dns-servers 9.42.106.2 2002:92a:8f7a:106:9:42:106:42
Ok
Console> get-dns-servers
Domain (DNS) servers:
    9.42.106.2
    2002:92a:8f7a:106:9:42:106:42
Console>
```

親トピック: [特殊目的コマンド](#)

set-ntp-servers

目的

Network Time Protocol サーバーを構成します。

構文

```
set-ntp-servers [server [... server ]]
```

パラメーター

server

Network Time Protocol サーバーの IP アドレスまたはホスト名

使用上の注意

- サーバーを指定せずに **set-ntp-servers** コマンドを開始すると、リストがクリアされます。
- それぞれのコマンド呼び出しによって、以前に構成されたサーバーが置換されます。

関連コマンド

[get-ntp-servers](#) を参照してください。

例

NTP サーバーを構成してから表示します。

```
Console> set-ntp-servers rtpgsa.ibm.com 9.42.106.44
Ok
Console> get-ntp-servers
NTP servers:    rtpgsa.ibm.com
                9.42.106.44
Console>
```

親トピック: [特殊目的コマンド](#)

show commandref

目的

コマンド行インターフェース (CLI) でサポートされるすべてのコマンドの参照リストを表示します。

構文

show commandref

パラメーター

なし。

使用上の注意

- このコマンドからの出力は、プログラムにより構文解析されることを意図しています。
- 出力は非常に長い場合があるので、シリアル・コンソールからコマンドを使用することは推奨されません。

例

コマンドをリストします。

```
Console> show commandref
<reference>

<commands>
  <entry>
    <name>alias</name>
    <brief>alias [name [value]]/brief>
    <detailed/>
  </entry>
  .....

```

親トピック: [特殊目的コマンド](#)

show components

目的

アプライアンス上のコンポーネントのファームウェア・バージョンを表示します。

構文

show components

パラメーター

なし。

使用上の注意

なし。

例

様々なコンポーネントのファームウェア・バージョンを表示します。

```
Console> show components
BMC Firmware Version: 2.53
ServeRaid Firmware Version: 12.12.0-0039
```

親トピック: [特殊目的コマンド](#)

show locales

目的

locale set コマンドで使用可能な、言語、国、および文字セットの値のリストを表示します。言語と国の有効な組み合わせが、その組み合わせのデフォルトの文字セットと共に表示されます。さらに、サポートされる文字セットの全体リストが表示されます。

構文

show locales

パラメーター

なし。

使用上の注意

一部の言語には複数の方言があります。例えば、中国語 (zh) には簡体字 (Hans) と繁体字 (Hant) の方言があります。中国語 (繁体字) を指定するには、言語 zh_Hant を使用します。アプライアンスとの通信に使用されるクライアント・ソフトウェアでサポートされている文字セットを指定するように注意してください。

関連コマンド

なし。

例

locale set コマンドで使用可能なロケール値をリストします。

```
Console> show locales
CWZBR02866I: af
CWZBR02866I: af NA UTF-8
CWZBR02866I: af ZA UTF-8
CWZBR02866I: agq
CWZBR02866I: agq CM UTF-8
...
CWZBR02866I: zh TW BIG5
CWZBR02866I: zu
CWZBR02866I: zu ZA UTF-8
CWZBR02867I: supported charsets: Big5 EUC-JP EUC-KR GB18030 GB2312 IBM00858 IBM850
IBM857 IBM860
IBM861 IBM862 IBM863 IBM864 IBM865 IBM866 IBM868 IBM869 ISO-2022-JP ISO-2022-JP-1 ISO-
2022-JP-2
ISO-2022-KR ISO-8859-1 ISO-8859-10 ISO-8859-13 ISO-8859-15 ISO-8859-2 ISO-8859-3 ISO-
8859-4
ISO-8859-5 ISO-8859-6 ISO-8859-7 ISO-8859-8 ISO-8859-9 KOI8-R KSC_5601 Shift_JIS US-
ASCII UTF-16
UTF-16BE UTF-16LE UTF-32 UTF-7 UTF-8 cp1363 cp851 macintosh x-mac-centraleurroman x-
mac-cyrillic
x-mac-greek x-mac-turkish
```

親トピック: [特殊目的コマンド](#)

show log

目的

ログ・ファイルのリスト、またはログ・ファイルの内容を表示します。

構文

- ログ・ファイルのリストを表示する: **show log list**
- ログ・ファイルの内容を表示する: **show log filename**

パラメーター

filename

コンソールにリストされるファイルの名前

使用上の注意

- IBM サポートに問題を報告するには、[platform must-gather](#) コマンドを使用してください。
- ネットワーク問題が原因で集めておくべき出力を取得できない場合に、このコマンドを使用してログをシリアル・コンソールにリストすることができます。しかし、シリアル・ポートは速度が遅く、ログ・ファイルは非常に大きいことがあるのに注意してください。この方法でリストするには長時間かかることがあります。

例

表示可能なログ・ファイルをリストし、次にデフォルトのログ・ファイルの内容をリストします。

```
Console> show log list
default-log
default-trace
Console> show log default-log
2012-02-17T19:25:43+00:00 CWZBR00028 [info][firmwaremgr(3929)]: The Firmware Manager has
started.
2012-02-17T19:25:43+00:00 CWZBR00031 [debug][firmwaremgr(3929)]: The Firmware Manager was
created.
...
```

親トピック: [特殊目的コマンド](#)

show timezones

目的

timezone set コマンドに使用可能な時間帯のリストを表示します。

構文

show timezones

パラメーター

なし。

例

timezone set コマンドに使用可能な時間帯の値をリストします。

```
Console> show timezones
EST5EDT
PST8PDT
...
```

親トピック: [特殊目的コマンド](#)

show version

目的

アプライアンスでアクティブなファームウェア・バージョンを表示します。

構文

show version

パラメーター

なし。

例

アクティブなファームウェア・バージョンを表示します。

```
Console> show version
Installation date: Mar 9, 2012 8:06:27 PM
Platform version: 4.9.4.0
Platform build ID: build12-20120309-1307
Platform build date: 2012-03-09 18:09:44+00:00
Machine type/model: 923572X
Serial number: 68A0553
Firmware type: Development
Console>
```

親トピック: [特殊目的コマンド](#)

sshkey fingerprint

目的

アプライアンスの SSH サーバーにより使用されるサーバー鍵を表示します。

構文

sshkey fingerprint

パラメーター

なし。

例

サーバーの SSH 鍵を表示します。

```
Console> sshkey fingerprint
SSH RSA Key fingerprint 10:c6:c6:5c:44:72:d1:e9:84:15:df:65:47:96:1a:c1
SSH DSA Key fingerprint cd:a9:8e:a8:0b:d0:45:ce:ac:9e:67:92:06:cf:63:1b
Console>
```

親トピック: [特殊目的コマンド](#)

sshkey reset

目的

アプライアンスの SSH サーバーにより使用されるサーバー鍵を削除して再生成します。

構文

sshkey reset

パラメーター

なし。

関連コマンド

なし。

例

サーバーの SSH 鍵をリセットします。

```
Console> sshkey fingerprint
CWZBR02154I: SSH RSA Key fingerprint 10:c6:c6:5c:44:72:d1:e9:84:15:df:65:47:96:1a:c1
CWZBR02154I: SSH DSA Key fingerprint cd:a9:8e:a8:0b:d0:45:ce:ac:9e:67:92:06:cf:63:1b
Console> sshkey reset
CWZBR02196I: Ok
Console> sshkey fingerprint
CWZBR02154I: SSH RSA Key fingerprint 7c:43:c7:36:2e:4a:ed:2d:a9:af:62:cc:44:f0:a5:7b
CWZBR02154I: SSH DSA Key fingerprint c3:fb:08:aa:c8:81:81:57:f1:ff:9b:c3:70:df:4a:bd
Console>
```

親トピック: [特殊目的コマンド](#)

sshkey verify

目的

アプライアンスの SSH サーバーにより使用されるサーバー鍵が強いか弱いかを判断します。

構文

sshkey verify

パラメーター

なし。

関連コマンド

なし。

例

サーバーの SSH 鍵の強度を検証し、最適な強度でない場合にはリセットします。

```
Console> sshkey verify
CWZBR02797E: SSH keys are not optimally strong
Console> sshkey reset
CWZBR02196I: Ok
Console> sshkey verify
CWZBR02798I: Host SSH keys are strong
Console>
```

親トピック: [特殊目的コマンド](#)

start-progress

目的

始動の進行中にこのコマンドを発行できます。このコマンドは、完了した始動プロセスのパーセントを表示します。

構文

start-progress

なし。

パラメーター

なし。

関連コマンド

なし。

例

```
Console> start-progress
DNS 9.42.139.134 9.42.139.134
DNS 9.42.139.139 9.42.139.139
DNS 9.42.139.152 xsa23e2.rtp.raleigh.ibm.com
DNS 9.42.139.142 localhost
CWXSA0014I: Volume 1 mounted
CWXSA0014I: Volume 2 mounted
CWXSA0006I: Catalog server started.
CWXSA0018I: Grid configuration service started.
CWXSA0017I: Container server 02 started.
CWXSA0017I: Container server 07 started.
CWXSA0017I: Container server 04 started.
CWXSA0017I: Container server 06 started.
CWXSA0017I: Container server 08 started.
CWXSA0017I: Container server 05 started.
CWXSA0017I: Container server 01 started.
CWXSA0017I: Container server 03 started.
CWXSA0002I: Grid administrative service started.
CWXSA0004I: Administrative console started.
STARTED
```

親トピック: [特殊目的コマンド](#)

status battery

目的

予想されるバッテリーの有効期限を表示します。

構文

status battery

パラメーター

なし。

使用上の注意

- バッテリーは、アプライアンスが電源オフで AC 電源に接続していない間、ハードウェア構成と RAID ディスク書き込み可能キャッシュにバックアップ電源を提供します。バッテリーは特定の期間有効です。このコマンドは、その期間の開始と終了を表示します。

関連コマンド

[status battery](#) も参照してください。

例

予想されるバッテリーの有効期限を表示します。

```
Console> status battery
Battery installed: Feb 17, 2012 7:25:45 PM
Battery expires: Feb 16, 2014 7:25:45 PM
Console>
```

親トピック: [特殊目的コマンド](#)

status cpu-usage

このコマンドは推奨されません。代わりに、**status cpu-utilization** を使用してください。

親トピック: [特殊目的コマンド](#)

status cpu-utilization

目的

過去 10 秒間、1 分間、10 分間、1 時間、および 1 日間のインターバルの CPU 使用率を表示します。

構文

```
status cpu-utilization
```

パラメーター

なし。

使用上の注意

なし。

例

CPU 使用率を表示します。

```
Console> status cpu-utilization
CPU utilization over time:
0% over 10 seconds
49% over 1 minute
25% over 10 minutes
14% over 1 hour
6% over 1 day
Console>
```

親トピック: [特殊目的コマンド](#)

status dynamic-tunnels

目的

現在アクティブな動的 IPsec トンネルの詳細を表示します。

構文

```
status dynamic-tunnels [tunnelID]
```

パラメーター

tunnelID

表示する特定の動的トンネルのオプションの ID (Ynnn の形式)。トンネル ID が指定されていない場合、既存の動的 IPsec トンネルすべてが表示されます。

使用上の注意

この状況プロバイダーにより戻されたフィールドについての詳細は、「ネットワーク・セキュリティー構成 (Network Security Configuration)」を参照してください。

例

動的トンネル Y2 の状況を表示します。

```
Console> status dynamic-tunnels y2
Dynamic tunnel name: Y2
  AssociatedFiltCode: n/a
  AssociatedFiltCodeRange: n/a
  AssociatedFiltDestPort: n/a
  AssociatedFiltDestPortRange: n/a
  AssociatedFiltProtocol: ALL(0)
  AssociatedFiltSrcPort: n/a
  AssociatedFiltSrcPortRange: n/a
  AssociatedFiltType: n/a
  AssociatedFiltTypeRange: n/a
  AuthAlgorithm: HMAC-SHA1
  AuthInboundSpi: 4258054942 (0xFDCCC31E)
  AuthOutboundSpi: 796243132 (0x2F75B4BC)
  CurrentTime: 2012/06/01 15:27:14
  DiffieHellmanGroup: 14
  EncryptInboundSpi: 4258054942 (0xFDCCC31E)
  EncryptOutboundSpi: 796243132 (0x2F75B4BC)
  Generation: 1
  HowActivated: OnDemand
  HowToAuth: ESP
  HowToEncap: Transport
  HowToEncrypt: 3DES-CBC
  IKEVersion: 2.0
  InboundBytes: 192
  InboundPackets: 3
  IpFilterRule: tcpcs-ipsec-ifr
  KeyLength: n/a
  Lifesize: 0K
  LifesizeRefresh: 0K
  LifetimeExpires: 2012/06/01 19:17:54
  LifetimeRefresh: 2012/06/01 19:13:05
  LocalAddressBase: 9.42.90.6
  LocalAddressPrefix: n/a
  LocalAddressRange: n/a
  LocalDynVpnRule: n/a
  LocalEndPoint: 9.42.90.6
  OutboundBytes: 192
  OutboundPackets: 3
  PFS: Yes
  ParentIKETunnelID: K1
  PendingNewActivation: n/a
  RemoteAddressBase: 9.42.105.155
  RemoteAddressPrefix: n/a
  RemoteAddressRange: n/a
  RemoteEndPoint: 9.42.105.155
  RmtIpSpecExIDPayload: n/a
```



```
State: DONE  
TunnelID: Y2  
VpnActionName: tcpcs-idva  
Console>
```

親トピック: [特殊目的コマンド](#)

status fan

目的

ファンの速度と状況を表示します。

構文

status fan

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームやブレード・ベースのプラットフォームには適用されません。

例

ファンの状況を表示します。

```
Console> status fan
Fan Speed Tray 1 Fan 1: 4800 (ok)
Fan Speed Tray 1 Fan 2: 5840 (ok)
Fan Speed Tray 1 Fan 3: 0.000000 (alert)
Fan Speed Tray 1 Fan 4: 6960 (ok)
Fan Speed Tray 2 Fan 1: 4720 (ok)
Fan Speed Tray 2 Fan 2: 5520 (ok)
Fan Speed Tray 2 Fan 3: 4720 (ok)
Fan Speed Tray 2 Fan 4: 6080 (ok)
Hard Disk Tray Fan 1: 4720 (ok)
Hard Disk Tray Fan 2: 6000 (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status flash

目的

フラッシュ・ドライブのサイズと使用可能なスペースを表示します。

構文

status flash

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームには適用されません。

例

フラッシュ・メモリー使用状況を表示します。

```
Console> status flash
Flash drive status:
    Total size = 3908 MB, free size = 1894 MB
Console>
```

親トピック: [特殊目的コマンド](#)

status intrusion

目的

物理的なアプライアンスのケースの健全性状況を表示します。

構文

status intrusion

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームやブレード・ベースのプラットフォームには適用されません。

関連コマンド

[device intrusion clear](#) を参照してください。

例

ケースの健全性状況を表示します。

```
Console> status intrusion
Case has not been opened and is secure
Console>
```

親トピック: [特殊目的コマンド](#)

status memory

目的

RAM メモリー使用状況を表示します。

構文

status memory

パラメーター

なし。

例

メモリー使用状況を表示します。

```
Console> status memory
Memory size 8193636K, free 7696564K
Console>
```

親トピック: [特殊目的コマンド](#)

status nvdimm

目的

nvDIMM の状態を表示します。

構文

status nvdimm

パラメーター

なし。

例

nvDIMM 状況を表示します。

```
Console> status nvdimm
CWZBR02911I: All nvDIMMs are ready to go
Console>
```

親トピック: [特殊目的コマンド](#)

status power-supply

目的

電源機構ユニットの状況を表示します。

構文

status power-supply

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームやブレード・ベースのプラットフォームには適用されません。

例

電源機構ユニットの状況を表示します。

```
Console> status power-supply
Power Supply 1 Output Failure: true (alert)
Power Supply 2 Output Failure: false (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status raid all

目的

RAID カード・コントローラーによりコントロールされる物理ディスクと論理ディスクの詳細を表示します。

構文

```
status raid all
```

パラメーター

なし。

使用上の注意

- この論理ディスク情報はまだサポートされないため、物理ディスクの詳細のみが表示されます。

関連コマンド

[status volume](#) および [status raid physical](#) を参照してください。

例

RAID カード・コントローラーによりコントロールされるすべての物理ディスクと論理ディスクを表示します。

```
Console> status raid all
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286968d50000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE2CE1F
Position: HDD0
  Controller ID: 1
  Device ID: 9
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286970090000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE28HV9
Console>
```

親トピック: [特殊目的コマンド](#)

status raid battery

目的

RAID BBU 情報を表示します。

構文

status raid battery

パラメーター

なし。

使用上の注意

BBU 情報は、MegaRAID コントローラーのあるアプライアンスでのみ使用可能で、MPT コントローラーのあるものでは使用できません。

例

RAID BBU 情報を表示します。

```
Console> status raid battery
CWZBR02808I: BBU status for Adapter '0':
CWZBR02809I:  Battery Type                : iBBU
CWZBR02810I:  Voltage                      : 4055 mV
CWZBR02811I:  Current                      : 0 mA
CWZBR02812I:  Temperature                  : 29 C
CWZBR02816I: BBU Firmware Status:
CWZBR02819I:  Charging Status              : None
CWZBR02820I:  Voltage                      : OK
CWZBR02821I:  Temperature                  : OK
CWZBR02822I:  Learn Cycle Requested       : No
CWZBR02823I:  Learn Cycle Active          : No
CWZBR02824I:  Learn Cycle Status         : OK
CWZBR02825I:  Learn Cycle Timeout        : No
CWZBR02826I:  I2c Errors Detected        : No
CWZBR02827I:  Battery Pack Missing       : No
CWZBR02828I:  Battery Replacement required : No
CWZBR02829I:  Remaining Capacity Low     : No
CWZBR02830I:  Periodic Learn Required    : No
CWZBR02831I:  Transparent Learn         : No
CWZBR02832I:  No space to cache offload   : No
CWZBR02833I:  Pack is about to fail. Should be replaced : No
CWZBR02834I:  Cache Offload premium feature required : No
CWZBR02835I:  Module microcode update required : No
CWZBR02847I: BBU Capacity Info:
CWZBR02848I:  Relative State of Charge    : 98 %
CWZBR02849I:  Absolute State of Charge    : 96 %
CWZBR02850I:  Remaining Capacity          : 1163 mAh
CWZBR02851I:  Full Charge Capacity        : 1187 mAh
CWZBR02853I:  Run time to empty (min)          : Battery is not being
discharged
CWZBR02855I:  Average time to empty (min)        : Battery is not being
discharged
CWZBR02857I:  Average Time to full (min)        : Battery is not being
charged
CWZBR02858I:  Cycle Count                      : 21
CWZBR02859I:  Max Error                          : 6 %
CWZBR02860I:  Remaining Capacity Alarm          : 120 mAh
CWZBR02861I:  Remaining Time Alarm             : 10 Min
CWZBR02836I: BBU Design Info:
CWZBR02837I:  Device Name                      : 3150301
CWZBR02838I:  Serial Number                    : 4010
CWZBR02839I:  Manufacturer Name                : LS1121001A
CWZBR02840I:  Date of Manufacture              : 07-03-2010
CWZBR02841I:  Design Capacity                  : 1215 mAh
CWZBR02843I:  Design Voltage                   : 3700 mV
CWZBR02844I:  Transparent Learn                : 0
CWZBR02845I:  Specification Info               : 33
CWZBR02846I:  Pack Stat Configuration          : 0x6490
Console>
```

親トピック: [特殊目的コマンド](#)

status raid physical

目的

RAID カード・コントローラーによりコントロールされる物理ディスクの詳細を表示します。

構文

status raid physical

パラメーター

なし。

使用上の注意

- 表示される詳細情報は、様々な RAID カード・コントローラーによって異なります。
- *ロケーション*または*位置*情報は、アプライアンス上のディスクの物理ロケーションを示します。アプライアンスのハード・ディスク・トレイまたはフロント・パネル上で、このロケーション情報とともにテキスト・マッピングがあります。

関連コマンド

[status volume](#) および [status raid all](#) を参照してください。

例

RAID カード・コントローラーによりコントロールされるすべての物理ディスクを表示します。

```
Console> status raid physical
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286968d50000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE2CE1F
Position: HDD0
  Controller ID: 1
  Device ID: 9
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286970090000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE28HV9
Console>
```

親トピック: [特殊目的コマンド](#)

status voltage

目的

電圧の測定値と状況を表示します。

構文

status voltage

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームには適用されません。

例

電圧の測定値を表示します。

```
Console> status voltage
Voltage +12: 11.8 (ok)
Voltage +5 Standby: 4.90 (ok)
Voltage +5: 4.87 (ok)
Voltage +3.3: 3.33 (ok)
Voltage +1.8: 1.79 (ok)
Voltage +1.5: 1.49 (ok)
Voltage CPU1 Core: 1.07 (ok)
Voltage CPU2 Core: 1.08 (ok)
Voltage Bus Termination: 1.08 (ok)
Voltage Battery: 3.14 (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status volume

目的

システムのすべてのボリュームとパーティション情報を表示します。

構文

status volume

パラメーター

なし。

使用上の注意

- このコマンドは、管理対象および管理対象でない両方のタイプのボリュームとパーティションを表示します。
- 管理対象でないパーティションでは、暗号化情報は表示されません。
- 管理対象でないパーティションは、マウント・ポイントが存在しない場合、オフラインとして表示されます。

関連コマンド

[status raid physical](#) および [status raid all](#) を参照してください。

例

すべてのボリュームの詳細を表示します。

```
Console> status volume
Storage Volume Status:
Volume 'raid0' is [Online] : size = 285148 MB, num partitions = 2
  Partition 1 'raid0-1' is [Online], encryption is [None]
    Total size = 91 MB,   free size = 86 MB
  Partition 2 'raid0-2' is [Online], encryption is [None]
    Total size = 184 MB,  free size = 179 MB
Console>
```

親トピック: [特殊目的コマンド](#)

status temperature

目的

物理ハードウェア・コンポーネントの温度状況を表示します。

構文

status temperature

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- このコマンドは仮想プラットフォームには適用されません。

例

温度状況を表示します。

```
Console> status temperature
Power Supply 1 Over-Temperature: false (ok)
Power Supply 1 High Temperature: false (ok)
Power Supply 2 Over-Temperature: false (ok)
Power Supply 2 High Temperature: false (ok)
Temperature System 1: 30.0 (ok)
Temperature System 2: 25.0 (ok)
Temperature CPU1: 27.0 (ok)
Temperature CPU2: 30.0 (ok)
Temperature Memory 00: 45.0 (ok)
Temperature Memory 10: 41.0 (ok)
Temperature Memory 20: 39.0 (ok)
Temperature Memory 30: 39.0 (ok)
Temperature Ethernet Ambient: 29 (ok)
Temperature Ethernet MAC: 50 (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status uptime

目的

アプライアンスがブートされてからの時間の長さを表示します。

構文

status uptime

パラメーター

なし。

例

アプライアンスが稼働している時間の長さを表示します。

```
Console> status uptime  
13 days 17 hours 13 minutes 15 seconds  
Console>
```

親トピック: [特殊目的コマンド](#)

timezone get

目的

構成された時間帯を表示します。

構文

```
timezone get
```

パラメーター

なし。

使用上の注意

- なし。

関連コマンド

[timezone set](#)、[datetime get](#)、および [datetime set](#) を参照してください。

例

構成された時間帯を表示します。

```
Console> timezone get
Timezone is EST
Console>
```

親トピック: [特殊目的コマンド](#)

timezone set

目的

地方時の時間帯を設定します。

構文

```
timezone set zone
```

パラメーター

zone

時間帯の値

使用上の注意

- なし。

関連コマンド

[timezone get](#)、[datetime get](#)、および [datetime set](#) を参照してください。

例

時間帯を東部標準時 (EST) に設定して、設定を確認します。

```
Console> timezone set EST
Console> timezone get
Timezone is EST
Console>
```

親トピック: [特殊目的コマンド](#)

unalias

目的

構成されたコマンド行インターフェース (CLI) コマンド別名を削除します。

構文

unalias *name*

パラメーター

name

削除されるコマンド行インターフェース (CLI) コマンドの代替名

関連コマンド

[alias](#) を参照してください。

例

構成された別名を表示し、それから test-connectivity という別名を削除します。

```
Console> alias
alias test-connectivity "net-test ping 9.42.106.2"
Console> unalias test-connectivity
Console> alias
No aliases defined
Console>
```

親トピック: [特殊目的コマンド](#)

usage

目的

config コマンドの使用方法の詳細を表示します。

構文

```
config usage
```

パラメーター

なし

使用上の注意

なし

例

config コマンドの詳細を表示します。

```
config usage
```

親トピック: [特殊目的コマンド](#)

user add

目的

新規ユーザーをアプライアンスに対して定義します。

構文

```
user add username [ defaultpass ]
```

パラメーター

username

追加されるユーザー名。ユーザー名は英数字、またはドット(.)、ダッシュ(-)、アンダースコア(_) またはプラス(+)
の特殊文字で構成する必要があります。

defaultpass

新規ユーザー名に割り当てるデフォルトのパスワード

使用上の注意

- 初回のコマンド呼び出しでパスワードが提供されない場合、コマンド行インターフェース (CLI) はパスワードとパスワードの確認を求めるプロンプトを出します。

関連コマンド

[user list](#)、[user delete](#)、[user password](#) を参照してください。

例

ユーザー fred および barney を追加します。

```
Console> user add fred
enter default password:*****
confirm password:*****
Ok
Console> user add barney rubble
Ok
Console> user list
User names:
    admin
    barney
    fred
Console>
```

親トピック: [特殊目的コマンド](#)

user delete

目的

アプリケーションからユーザーを削除します。

構文

```
user delete username
```

パラメーター

username

削除されるユーザー名

関連コマンド

[user add](#) および [user list](#) も参照してください。

例

ユーザー fred を削除します。

```
Console> user list
User names:
    admin
    fred

Console> user delete fred
Ok
Console> user list
User names:
    admin
Console>
```

親トピック: [特殊目的コマンド](#)

user known-hosts list

目的

現行ユーザーの既知のホストをリストします。ユーザーがコマンド `file get|put scp://user@host/dir` を発行すると、ホストの公開鍵がローカル・ファイルに保存されます。このコマンドを使用して、そのファイルに保存されるホストをリストします。

構文

`user known-hosts list`

パラメーター

なし。

関連コマンド

例

現在ログインしているユーザーの既知のホストをリストします。

```
Console> user known-hosts list
aaa.example.com,192.168.10.200
bbb.example.com,192.168.11.201
Console>
```

親トピック: [特殊目的コマンド](#)

user known-hosts delete

目的

現行ユーザーの既知のホスト・ファイルから、指定されたホスト用の保管された公開鍵ファイルを削除します。ユーザーがコマンド `file get|put scp://user@host/dir` を発行すると、ホストの公開鍵がローカル・ファイルに保存され、ホストの ID を後続の `file get|put scp://` コマンドで検証するのに使用されます。ホストが鍵を変更すると、その検証が失敗することがあります。このコマンドを使用して既知のホストに保管された公開鍵を削除し、ホストの新しい鍵を知ることができます。

構文

```
user known-hosts delete hostname
```

パラメーター

hostname

削除される既知のホスト・エントリーのホスト名または IP アドレス

関連コマンド

[user known-hosts list](#) を参照してください。

例

既知のホスト `aaa.example.com` の公開鍵を、現在ログインしているユーザーの既知のホストのリストから削除します。

```
<p>Console> user known-hosts delete aaa.example.com  
CWZBR02196I: Ok  
Console></p>
```

親トピック: [特殊目的コマンド](#)

user list

目的

アプライアンスに定義されたユーザーをリストします。

構文

user list

パラメーター

なし。

関連コマンド

[user add](#) および [user delete](#) を参照してください。

例

現在定義されているユーザーをリストします。

```
Console> user list
User names:
    admin
    fred
Console>
```

親トピック: [特殊目的コマンド](#)

user password

目的

現在アクティブなユーザーのパスワードを変更します。

構文

```
user password oldpass newpass
```

パラメーター

oldpass

古い (現在アクティブな) パスワード

newpass

新規パスワード

使用上の注意

- 初回のコマンド呼び出しで古いパスワードまたは新規パスワードが提供されない場合、コマンド行インターフェース (CLI) は適切なパスワードと新規パスワードの確認を求めるプロンプトを出します。

関連コマンド

[user add](#) を参照してください。

例

パスワードを 3 回変更します。

```
Console> user password aaaa bbbb
Changing password...
Ok
Console> user password bbbb
new password:****
confirm password:****
Changing password...
Ok
Console> user password
current password:****
new password:*****
confirm password:*****
Changing password...
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

user sshkey add

目的

現行ユーザーによる使用のために、指定された URL から SSH 公開鍵ファイルを取得します。

構文

```
user sshkey add url
```

パラメーター

url

公開鍵ファイルの URL

使用上の注意

- 鍵ファイルが保管されると、対応する秘密鍵のある SSH クライアントは、パスワードを指定せずにそのユーザーとして SSH でアクセスすることができます。
- サポートされる公開鍵はアプライアンス・ユーザーごとに 1 つのみです。後続の **user sshkey add** コマンドは、そのアプライアンス・ユーザーの公開鍵ファイルを置換します。

関連コマンド

[user sshkey delete](#) を参照してください。

例

現在ログインしているユーザーの公開鍵を取得します。

```
Console> user sshkey add scp://user1@system1.rtp.raleigh.ibm.com:~/mypublickey
user1@system1.rtp.raleigh.ibm.com's password:
mypublickey                               100% 411      0.4KB/s   00:00
Wrote 411 bytes to local storage
Adding ssh public key file for user wilma
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

user sshkey delete

目的

現行ユーザーの保管された SSH 公開鍵ファイルを削除します。

構文

```
user sshkey delete
```

パラメーター

なし。

関連コマンド

[user sshkey add](#) を参照してください。

例

現在ログインしているユーザーの公開鍵を削除します。

```
Console> user sshkey delete
Deleting ssh public key file for user wilma
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification

Packages	
com.ibm.websphere.objectgrid	These are the main application APIs for the ObjectGrid.
com.ibm.websphere.objectgrid.client	This package contains the primary interfaces and classes for setting behaviors for ObjectGrid clients for this process.
com.ibm.websphere.objectgrid.config	This package contains the interfaces and a factory class for creating ObjectGrid configuration objects programatically.
com.ibm.websphere.objectgrid.continuousquery	
com.ibm.websphere.objectgrid.continuousquery.exception	
com.ibm.websphere.objectgrid.continuousquery.filter	These are the Filter implementations which are used to build continuous query definitions.
com.ibm.websphere.objectgrid.management	This package contains the interfaces for all ObjectGrid MBeans.
com.ibm.websphere.objectgrid.plugins	These are the interfaces for adding plugins to the Grid core framework.
com.ibm.websphere.objectgrid.plugins.builtins	This package contains built-in plugins for ObjectGrid core.
com.ibm.websphere.objectgrid.plugins.index	
com.ibm.websphere.objectgrid.plugins.io.annotations	
com.ibm.websphere.objectgrid	This package has the class MapPermission and class AdminPermission which represents the permissions for to access the ObjectGrid maps and

security	ObjectGrid administration respectively.
com.ibm.websphere.objectgrid.security.config	This package contains the ObjectGrid client security configurations.
com.ibm.websphere.objectgrid.security.plugins	This package contains the interfaces for adding plug-ins to the ObjectGrid security framework and associated Exception classes.
com.ibm.websphere.objectgrid.security.plugins.builtins	This package contains the built-in implementation for the security plugins.
com.ibm.websphere.objectgrid.spring	This package holds the Spring specific APIs for ObjectGrid.
com.ibm.websphere.xs.ra	These are the eXtreme Scale Resource Adapter APIs that allows integration with a Java EE application.
com.ibm.websphere.xsa	

[Overview](#)
[Package](#)
[Class Tree](#)
[Serialized](#)
[Deprecated](#)
[Index](#)
[Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV](#)
[NEXT](#)
[FRAMES](#)
[NO FRAMES](#)
[All Classes](#)

Package com.ibm.websphere.objectgrid

These are the main application APIs for the ObjectGrid.

See:

[Description](#)

Interface Summary	
BackingMap	This is the public interface to the BackingMap.
CatalogDomainInfo	Identifies the configuration attributes of a catalog service domain.
CatalogDomainManager	Provides access to catalog domain configuration information for the current environment.
ClientClusterContext	This interface is a context to represent which cluster/domain the client connected to using one of the ObjectGridManager.connect methods.
ClientReplicableMap	Deprecated. <i>The client replicated map function is deprecated in version 8.6.</i>
IObjectGridException	This interface is used to ensure JDK 1.4 Throwable chaining behavior for all exceptions thrown by ObjectGrid even when an earlier JDK is used (e.g.
JavaMap	This interface is a handle to a named Map.
ObjectGrid	This object is used for creating sessions to the ObjectGrid.
ObjectGridManager	ObjectGridManager is responsible for creating or retrieving local ObjectGrid instances and connecting to distributed ObjectGrid servers.
ObjectMap	This is a handle to a named Map.
PartitionManager	This interface will be used for calculating the proper partition for a given input key.
Session	This interface represents a session container for ObjectMaps.
StateManager	The StateManager can be used to retrieve the availability state of an ObjectGrid.
TxID	This interface is an opaque identifier for a transaction.

Class Summary	
AvailabilityState	Each shard in a distributed ObjectGrid has an availability state associated with it.
ClientReplicableMapMode	Client Replication mode
CopyMode	This class is used to define the "copy" mode when the setCopyMode method of the BackingMap interface is used.
LockStrate	LockStrategy provides an enumerated type idiom for use on the

gy	BackingMap.setLockStrategy(LockStrategy) method.
ObjectGridManagerFactory	This factory class is a high level helper class to get ObjectGridManager instances.
TTLType	Every BackingMap in ObjectGrid has a built in timed based evictor that is referred to as "time to live" evictor or TTL evictor.

Enum Summary

ObjectMap.PutMode	Identifies the operation mode of the ObjectMap.put(Object, Object) , ObjectMap.putAll(Map) , JavaMap.put(Object, Object) and JavaMap.putAll(Map) methods.
Session.TransactionCommitProtocol	The commit protocols that can be used to commit the Session's transaction

Exception Summary

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException	This exception is thrown when a method call to the Client/Server TransactionCallback detects the user is attempting to perform a write against multiple maps in different Map Sets, Partition Sets or Replication groups.
ConnectException	This exception is used to indicate that the client was unable to connect to the server
DuplicateKeyException	A DuplicateKeyException exception is thrown if a key cannot be inserted into a BackingMap because an object with the same key already exists.
KeyNotFoundException	Normally, record not found means a null is returned.
LockDeadlockException	This exception is used by the lock manager to indicate that it detected a deadlock.
LockException	A general locking exception indicating something went wrong with locking operations.
LockInternalFailureException	This exception is used by the lock manager to indicate it detected some internal programming error while processing a lock or unlock request.
LockTimeoutException	This exception is used by the lock manager to indicate that the maximum wait time for a lock has been exceeded.
NoActiveTransactionException	An exception indicating there is no active transaction.
ObjectGridException	Base exception class for all checked exceptions thrown by the ObjectGrid product.
ObjectGrid	

<u>RuntimeException</u>	This exception is the base class for all runtime exceptions thrown by the cache.
<u>ReadOnlyException</u>	This exception is thrown when an attempt is made to modifying operations on a read only maps.
<u>ReplicationVotedToRollbackTransactionException</u>	This exception is thrown when a transaction was rolled back because some/all of the replicas failed to apply the transaction when in synchronous replication mode.
<u>SessionNotReentrantException</u>	A Session object can only be used by a single thread concurrently to perform map operations.
<u>TargetNotAvailableException</u>	A TargetNotAvailableException indicates the ObjectGrid target is not available.
<u>TransactionAffinityException</u>	This exception is thrown for inflight transaction when server fails over.
<u>TransactionAlreadyActiveException</u>	An exception indicating a transaction is already active for the current session.
<u>TransactionException</u>	A general transaction exception indicating something went wrong with a transaction.
<u>TransactionQuiesceException</u>	This exception is thrown when partition/shard/mapset/replication group/replication group member/server/cluster/objectgrid is entered quiesce process for various reasons such as shard movement, partition relocation, system update, server shutdown, and others.
<u>TransactionTimeoutException</u>	This exception is thrown when a transaction exceeds the transaction timeout that was specified on the ObjectGrid or Session.
<u>UnavailableServiceException</u>	This exception is thrown when all servers are dead or when all services are unavailable even though servers are running.
<u>UndefinedMapException</u>	This exception indicates that the map which an application tries to access is not defined in the ObjectGrid.

Package **com.ibm.websphere.objectgrid** Description

These are the main application APIs for the ObjectGrid. The main interface here is the ObjectGrid interface. A JVM needs to create at least one instance.

Introduction

The WebSphere ObjectGrid is designed as a data caching tier that can be used to hold data from multiple sources and then make it available to the clients of the ObjectGrid. The clients access the data through the ObjectGrid APIs. The ObjectGrid is designed to be able to store large quantities of data.

Programming Tutorial

The following sections show snippets on the usage of the ObjectGrid APIs.

Obtaining a ObjectGrid instance.

The application needs to construct an ObjectGrid reference first. An application can choose to make several ObjectGrid instances. Each instance is independent, however, and has its own configuration file. For now, use the following code and programmatically initialize it using the setter methods on the ObjectGrid.

```
ObjectGrid objectGrid = new ObjectGridImpl();
```

The instance can then have a Map defined on it using the following snippet:

```
BackingMap bm = objectGrid.defineMap("TABLE1");
```

Again, setter methods on BackingMap allow it to be configured once it's defined.

Working with an ObjectGrid, Sessions

Each thread that wants to access the ObjectGrid must have its own Session instance. The ObjectGrid class has a getSession method that returns one. Once the thread has a Session then it can obtain ObjectMap instances for manipulating data in the ObjectGrid as well as use the begin/commit/rollback methods on the Session to handle transactions.

```
Session session = objectGrid.getSession();
ObjectMap table1 = session.getMap("TABLE1");
session.begin();
MyData d = (MyData)table1.get("key1");
session.commit();
```

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Package	Next Package	Frames	No Frames	All Classes				

Package com.ibm.websphere.objectgrid.client

This package contains the primary interfaces and classes for setting behaviors for ObjectGrid clients for this process.

See:

[Description](#)

Interface Summary

ClientProperties	The set of properties used to define various preference for ObjectGrid clients.
----------------------------------	---

Package com.ibm.websphere.objectgrid.client Description

This package contains the primary interfaces and classes for setting behaviors for ObjectGrid clients for this process.

Overview

The interfaces in this package should not be implemented directly but are used by the [ClientClusterContext](#) to set default behaviors for application clients for an ObjectGrid instance.

The properties available for use are defined in the [ClientProperties](#) interface.

There are two ways to configure client properties:

1. Create a properties file named `objectGridClient.properties` and store it in the root of your classpath.
2. Create a properties file on your file system in the directory where the client is started from named `objectGridClient.properties`.
3. Create a properties file with any path and name and use the following system property to detect it: `-Dcom.ibm.websphere.objectgrid.ClientProperties=<fileName>`
4. Create a properties file with any path and name and set load it programmatically and pass url to `ClientClusterContext.getClientProperties(ogname, url)`.
5. Programmatically define the properties the `ClientProperties` set methods.

In the following example we set the proximity routing defaults for all clients that use this `ClientClusterContext`:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ClientClusterContext ccc = ogMgr.connect(...);
ClientProperties props = ccc.getClientProperties("myOGName");
props.setPreferLocalHost(true);
props.setPreferLocalProcess(true);
props.setPreferZones(new String[]{"New York", "Texas"});

// The ClientProperties are now applied to the ObjectGrid client connection:
ObjectGrid og=ogMgr.get(ccc, "myOGName");
```

The following example uses a custom client properties file:

```
ClientClusterContext ccc = ogMgr.connect(...);
```

```

URL clientPropsURL = Thread.currentThread().getContextClassLoader().getResource("etc/myObjectGridClient.properties");
ClientProperties props = ccc.setClientProperties("myOGName", clientPropsURL);

// The ClientProperties are now applied to the ObjectGrid client connection:
ObjectGrid og=ogMgr.get(ccc, "myOGName");

```

The following file is an example of a properties file that matches the preceding API:

```

preferLocalProcess=true
preferLocalhost=true
preferZones=New York,Texas

```

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.client

Interface ClientProperties

public interface **ClientProperties**

The set of properties used to define various preference for ObjectGrid clients.

See the [package summary](#) for details on how to use the ClientProperties class and properties file.

Since:

WAS XD 6.1.0.3, XC10

Field Summary

s t a t i c S t r i n g	<p>CLIENT_PROPS_FILE_PATH_KEY The system property key to override the location of the client properties file.</p>
s t a t i c S t r i n g	<p>DEFAULTCLIENTPROPERTYFILE The default name of client property file</p>
s t a t i c S t r i n g	<p>PROP_LISTENER_HOST Listener host property key for the client properties file.</p>
s t a t i c S t r i n g	<p>PROP_LISTENER_PORT Listener port property key for the client properties file.</p>

r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

PROP_MAXIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

PROP_MAXIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.

PROP_MINIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

PROP_MINIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.

PROP_PREFER_LOCAL_HOST

Currently, this property is not used.

PROP_PREFER_LOCAL_PROCESS

Currently, this property is not used.

n
g

s
t
a
t
i
c

[PROP_PREFER_ZONES](#)

Prefer zones property key for the client properties file.

S
t
r
i
n
g

s
t
a
t
i
c

[PROP_REQUEST_RETRY_TIMEOUT](#)

The requestRetryTimeout which indicates how long to retry a request (in milliseconds).

S
t
r
i
n
g

s
t
a
t
i
c

[PROP_SHUFFLE_BOOTSTRAP_ADDRESSES](#)

The shuffleBootstrapAddresses property is used to determine if the catalog service grid addresses should be randomized when used by a client when bootstrapping to the grid.

S
t
r
i
n
g

s
t
a
t
i
c

[PROP_XIO_REQUEST_TIMEOUT](#)

The xioRequestTimeout indicates how long the eXtreme IO transport will wait for cross-process call to complete.

S
t
r
i
n
g

s
t
a
t
i
c

[PROP_XIO_TIMEOUT](#)

Sets the timeout for server requests using the eXtremeIO transport.

S
t
r
i
n
g

Method Summary

S
t
r
i
n
g

[getListenerHost\(\)](#)

Retrieves the host to be used by the ORB.

i n t	getListenerPort() Retrieves the port to be used by the ORB.
i n t	getMaximumXIONetworkThreads() Gets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.
i n t	getMaximumXIOWorkerThreads() Sets the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.
i n t	getMinimumXIONetworkThreads() Gets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.
i n t	getMinimumXIOWorkerThreads() Retrieves the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.
S t r i n g []	getPreferZones() Retrieve the preferred zones.
l o n g	getRequestRetryTimeout() Retrieves the current request retry timeout.
i n t	getXIORequestTimeout() Gets the current timeout that is allowed for requests to servers over the XIO transport to complete.
i n t	getXIOTimeout() Returns the current timeout for server requests using the XIO transport.
b o o l e a n	isBootStrapListShuffled() Retrieves the value of the BootStrapListShuffled property.
b o o l e a n	isPreferLocalHost() This method is reserved for future use.
b o o l e a n	isPreferLocalProcess() This method is reserved for future use.
v o i d	setBootStrapListShuffled(boolean shuffle) Sets the value of the BootStrapListShuffled property.

v o i d	setMaximumXIONetworkThreads (int maxThreads) Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.
v o i d	setMaximumXIOWorkerThreads (int maxThreads) Retrieves the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.
v o i d	setMinimumXIONetworkThreads (int minThreads) Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.
v o i d	setMinimumXIOWorkerThreads (int minThreads) Sets the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.
v o i d	setPreferLocalHost (boolean localHost) This method is reserved for future use.
v o i d	setPreferLocalProcess (boolean localProcess) This method is reserved for future use.
v o i d	setPreferZones (String[] zones) Prefer routing to specific zones.
v o i d	setRequestRetryTimeout (long requestRetryTimeout) Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions.
v o i d	setXIORequestTimeout (int timeout) Sets the current timeout that is allowed for requests to servers over the XIO transport to complete.
v o i d	setXIOTimeout (int timeout) Sets the current timeout for server requests using the XIO transport.

Field Detail

DEFAULTCLIENTPROPERTYFILE

static final [String](#) DEFAULTCLIENTPROPERTYFILE

The default name of client property file

See Also:

[Constant Field Values](#)

CLIENT_PROPS_FILE_PATH_KEY

static final [String](#) CLIENT_PROPS_FILE_PATH_KEY

The system property key to override the location of the client properties file.

Since:

7.0

See Also:

[Constant Field Values](#)

PROP_PREFER_LOCAL_PROCESS

static final [String](#) PROP_PREFER_LOCAL_PROCESS

Currently, this property is not used. It is reserved for future use.

See Also:

[setPreferLocalProcess\(boolean\)](#), [Constant Field Values](#)

PROP_PREFER_LOCAL_HOST

static final [String](#) PROP_PREFER_LOCAL_HOST

Currently, this property is not used. It is reserved for future use.

See Also:

[setPreferLocalHost\(boolean\)](#), [Constant Field Values](#)

PROP_PREFER_ZONES

static final [String](#) PROP_PREFER_ZONES

Prefer zones property key for the client properties file. Each specified zone is separated by a comma in the form: preferZones=ZoneA,ZoneB,ZoneC

See Also:

[setPreferZones\(String\[\]\)](#), [Constant Field Values](#)

PROP_REQUEST_RETRY_TIMEOUT

static final [String](#) PROP_REQUEST_RETRY_TIMEOUT

The requestRetryTimeout which indicates how long to retry a request (in milliseconds). A 0 indicates that the request should fail fast and skip over in internal retry logic. Exceptions that cannot succeed even if tried again such as DuplicateException will be returned immediately.

Since:

7.0

See Also:

[setRequestRetryTimeout\(long\)](#), [Constant Field Values](#)

PROP_XIO_REQUEST_TIMEOUT

static final [String](#) PROP_XIO_REQUEST_TIMEOUT

The xioRequestTimeout indicates how long the eXtreme IO transport will wait for cross-process call to complete. The value is expressed in milliseconds. The default value is 30,000 or 30 seconds. When custom tuning client side retry of eXtreme Scale operations, this value will determine how long a single network operation is given, and then the

PROP_REQUEST_RETRY_TIMEOUT / requestRetryTimeout will control how long those individual network operations are retried.

Since:

8.6.0.2, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_LISTENER_HOST

static final [String](#) PROP_LISTENER_HOST

Listener host property key for the client properties file.

Since:

XS 7.1

See Also:

[getListenerHost\(\)](#), [Constant Field Values](#)

PROP_LISTENER_PORT

static final [String](#) PROP_LISTENER_PORT

Listener port property key for the client properties file.

Since:

XS 7.1

See Also:

[getListenerPort\(\)](#), [Constant Field Values](#)

PROP_SHUFFLE_BOOTSTRAP_ADDRESSES

static final [String](#) PROP_SHUFFLE_BOOTSTRAP_ADDRESSES

The shuffleBootstrapAddresses property is used to determine if the catalog service grid addresses should be randomized when used by a client when bootstrapping to the grid. The default value of the property is true.

Since:

7.1.0.3

See Also:

[Constant Field Values](#)

PROP_XIO_TIMEOUT

static final [String](#) PROP_XIO_TIMEOUT

Sets the timeout for server requests using the eXtremeIO transport. The timeout is set in seconds. The default xioTimeout for server requests is set to 30 seconds. The valid range is timeout >= 1.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MAXIMUM_XIO_NETWORK_THREAD_POOL_SIZE

static final [String](#) PROP_MAXIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MAXIMUM_XIO_WORKER_THREAD_POOL_SIZE

static final [String](#) PROP_MAXIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MINIMUM_XIO_NETWORK_THREAD_POOL_SIZE

static final [String](#) PROP_MINIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MINIMUM_XIO_WORKER_THREAD_POOL_SIZE

static final [String](#) PROP_MINIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

Method Detail

setPreferZones

void setPreferZones([String](#)[] zones)

Prefer routing to specific zones.

When zones are enabled on an ObjectGrid, requests will be routed to the specified zones.

Parameters:

zones - array of zone names. If null or an empty array, then requests are routed to all zones.

setPreferLocalProcess

void **setPreferLocalProcess**(boolean localProcess)

This method is reserved for future use. Calls to the method will not result in any performed operation.

Parameters:

localProcess -

setPreferLocalHost

void **setPreferLocalHost**(boolean localHost)

This method is reserved for future use. Calls to the method will not result in any performed operation.

Parameters:

localHost -

getPreferZones

[String](#)[] **getPreferZones**()

Retrieve the preferred zones.

Returns:

the preferred zones.

isPreferLocalProcess

boolean **isPreferLocalProcess**()

This method is reserved for future use. The returned value should be ignored by the user.

Returns:

false

isPreferLocalHost

boolean **isPreferLocalHost**()

This method is reserved for future use. The returned value should be ignored by the user.

Returns:

false

setRequestRetryTimeout

void **setRequestRetryTimeout**(long requestRetryTimeout)

Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions. A request will timeout when either the request timeout expires or the transaction timeout expires, whichever expires first.

A value of 0 indicates that all requests should fail immediately and avoid any retry logic.

Exceptions that cannot succeed even if tried again such as `DuplicateKeyException` exceptions will be thrown immediately.

A value of -1 indicates that the request retry timeout is not set, meaning that the request duration is governed by the transaction timeout.

The request retry timeout can be overridden using the [Session.setRequestRetryTimeout\(long\)](#) method.

Parameters:

`requestRetryTimeout` - the duration in milliseconds retry a client request, 0 if the request should fail immediately or -1 if the request timeout is not set.

Since:

7.0

See Also:

[Session.setRequestRetryTimeout\(long\)](#), [ObjectGrid.setTxTimeout\(int\)](#)

getRequestRetryTimeout

long `getRequestRetryTimeout()`

Retrieves the current request retry timeout. Returns -1 if it was not set.

Returns:

`requestRetryTimeout` in milliseconds, 0 to fail immediately or -1 if not set.

Since:

7.0

getListenerHost

[String](#) `getListenerHost()`

Retrieves the host to be used by the ORB. The listener host property defaults to 'localhost'. This property can only be set in the client.properties file.

Returns:

The host that the ORB will bind to.

Since:

7.1

getListenerPort

int `getListenerPort()`

Retrieves the port to be used by the ORB. The listener port property defaults to the corbaloc port, 2809. This property can only be set in the client.properties file.

Returns:

The port that the ORB will bind to.

Since:

7.1

isBootStrapListShuffled

boolean `isBootStrapListShuffled()`

Retrieves the value of the `BootStrapListShuffled` property.

Returns:

true if the value of `BootStrapListeShuffled` was set to true. false if the value of `BootStrapListeShuffled` was set to false.

Since:

7.1.0.3

setBootStrapListShuffled

```
void setBootStrapListShuffled(boolean shuffle)
```

Sets the value of the `BootStrapListShuffled` property.

Parameters:

`shuffle` - true the bootstrap list will be shuffled providing each client a random distribution of catalog servers to select from. false the first viable address in the list of catalog servers will be used.

Since:

7.1.0.3

getMinimumXIOWorkerThreads

```
int getMinimumXIOWorkerThreads()
```

Retrieves the minimum number of threads to allocate in the `eXtremeIO` transport request processing thread pool.

Returns:

the minimum number of threads.

Since:

8.6, XC10 2.5

setMinimumXIOWorkerThreads

```
void setMinimumXIOWorkerThreads(int minThreads)
```

Sets the minimum number of threads to allocate in the `eXtremeIO` transport request processing thread pool.

Parameters:

`minThreads` - the minimum number of threads.

Since:

8.6, XC10 2.5

getMaximumXIOWorkerThreads

```
int getMaximumXIOWorkerThreads()
```

Sets the maximum number of threads to allocate in the `eXtremeIO` transport request processing thread pool.

Returns:

the maximum number of threads.

Since:

8.6, XC10 2.5

setMaximumXIOWorkerThreads

```
void setMaximumXIOWorkerThreads(int maxThreads)
```

Retrieves the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.

Parameters:

maxThreads - the maximum number of threads.

Since:

8.6, XC10 2.5

getMinimumXIONetworkThreads

```
int getMinimumXIONetworkThreads()
```

Gets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

Returns:

the minimum number of threads

Since:

8.6, XC10 2.5

setMinimumXIONetworkThreads

```
void setMinimumXIONetworkThreads(int minThreads)
```

Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

Parameters:

minThreads - the minimum number of threads

Since:

8.6, XC10 2.5

getMaximumXIONetworkThreads

```
int getMaximumXIONetworkThreads()
```

Gets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

Returns:

the maximum number of threads.

Since:

8.6, XC10 2.5

setMaximumXIONetworkThreads

```
void setMaximumXIONetworkThreads(int maxThreads)
```

Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

Parameters:

maxThreads - the maximum number of threads.

Since:

8.6, XC10 2.5

getXIOTimeout

int `getXIOTimeout()`

Returns the current timeout for server requests using the XIO transport.

Returns:

the current timeout in seconds

Since:

8.6, XC10 2.5

setXIOTimeout

void `setXIOTimeout(int timeout)`

Sets the current timeout for server requests using the XIO transport. The timeout is set in seconds. The valid range is `timeout >= 1`.

Parameters:

`timeout` - the timeout in seconds

Since:

8.6, XC10 2.5

getXIORequestTimeout

int `getXIORequestTimeout()`

Gets the current timeout that is allowed for requests to servers over the XIO transport to complete. The timeout is set in milliseconds.

Since:

8.6.0.2, XC10 2.5

setXIORequestTimeout

void `setXIORequestTimeout(int timeout)`

Sets the current timeout that is allowed for requests to servers over the XIO transport to complete. The timeout is set in milliseconds. The valid range is greater than 0.

Parameters:

`timeout` - the timeout in milliseconds

Since:

8.6.0.2, XC10 2.5

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package com.ibm.websphere.objectgrid.config

This package contains the interfaces and a factory class for creating ObjectGrid configuration objects programmatically.

See:

[Description](#)

Interface Summary	
BackingMapConfiguration	A BackingMapConfiguration object can be used to override BackingMap settings on the client side.
ConfigProperty	ConfigProperty can be used to attach properties to a Plugin.
ObjectGridConfiguration	An ObjectGridConfiguration object can be used to override ObjectGrid plugins on the client side.
Plugin	This interface represents an ObjectGrid or BackingMap plugin.
PluginType	Every Plugin has a PluginType.

Class Summary	
ConfigPropertyType	ConfigPropertyType is used to set the type of an attribute on a Plugin.
ObjectGridConfigFactory	This is the configuration factory for ObjectGrid configuration entities.
QueryConfig	This QueryConfig represents a schema configuration for a QueryManager.
QueryMapping	A QueryMapping maps a Java class to a BackingMap and allows a map to be included in a query.
QueryRelationship	A QueryRelationship represents a relationship between two BackingMap value classes.

Exception Summary	
ObjectGridConfigurationException	Thrown when a problem with the current configuration is found.

Package com.ibm.websphere.objectgrid.config Description

This package contains the interfaces and a factory class for creating ObjectGrid configuration objects programatically. The main use of this is by the objectgrid client to override serverside configuration.

Overview

ObjectGridManagerFactory has static methods to create the configuration objects. Using these configuration objects in conjunction with ObjectGridManager methods

- `setOverrideObjectGridConfigurations(Map)`
- `putOverrideObjectGridConfigurations(String, List)`

to override client configuration, before connecting to the objectgrid server.

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
ew	ge		ed	ted				
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.config

Class QueryRelationship

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.config.QueryRelationship

All Implemented Interfaces:

[Serializable](#)

```
public class QueryRelationship
extends Object
implements Serializable
```

A QueryRelationship represents a relationship between two BackingMap value classes. A BackingMap must have one class type defined in the value part of the map. A relationship can be established between two maps by mapping the source and target map's value classes.

The cardinality of the relationship is automatically determined by the type of the attribute field.

A relationship requires two classes, a relationship field, and optionally an inverse relationship field.

For example: Two entities; Department and Employee, have the following bi-directional relationship:

1. One department has many employees. the collection field in the Department class is "emps"
2. An employee belongs to one department

```
public class Department {
    private int id;
    private Collection emps;

    public void setEmps(Collection emps) {
        this.emps = emps;
    }

    public Collection getEmps() {
        return emps;
    }
    ...
}

public class Employee {
    private int id;
    private Department dept;

    public void setDept(Department dept) {
        this.dept = dept;
    }

    public Department getDept() {
        return dept;
    }
}
```

```
}
```

Use the following method call to establish this bi-directional relationship.

```
queryConfig.addRelationship(new QueryRelationship(  
    Department.class.getName(), Employee.class.getName(), "emps", "dept"));
```

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary	
QueryRelationship (String sourceClass, String targetClass, String relationshipField, String invRelationshipField)	Constructor for creating a QueryRelationship instance.

Method Summary	
boolean	equals (Object o)
String	getInvRelationshipField () Retrieve the inverse relationship attribute name of a bi-directional relationship.
String	getRelationshipField () Retrieve the name of the attribute in the source class that references the key of the target class.
String	getSourceClass () Retrieve the name of the class representing the source of a relationship.
String	getTargetClass () Retrieve the name of the class representing the target of a relationship.
int	hashCode ()
void	setInvRelationshipField (String invRelationshipField) Set the inverse relationship attribute name of a bi-directional relationship.

v o i d	setRelationshipField (String relationshipField) Set the name of the attribute in the source class that references the key of the target class.
v o i d	setSourceClass (String sourceClass) Set the name of the class representing the source of a relationship.
v o i d	setTargetClass (String targetClass) Set the name of the class representing the target of a relationship.
S t r i n g	toString ()

Methods inherited from class [java.lang.Object](#)

[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

QueryRelationship

```
public QueryRelationship(String sourceClass,
                        String targetClass,
                        String relationshipField,
                        String invRelationshipField)
```

Constructor for creating a QueryRelationship instance.

The sourceClass, targetClass, and relationshipField must not be null.

Parameters:

sourceClass - the source class of the relationship

targetClass - the target class of the relationship

relationshipField - the attribute in the source class that references the key of the target class.

invRelationshipField - the attribute in the target class that references the key of the source class. This value is null if a bi-directional relationship does not exist.

Method Detail

getInvRelationshipField

```
public String getInvRelationshipField()
```

Retrieve the inverse relationship attribute name of a bi-directional relationship.

Returns:

the attribute name of the inverse side of a bi-directional relationship or null if the relationship is uni-directional.

setInvRelationshipField

```
public void setInvRelationshipField(String invRelationshipField)
```

Set the inverse relationship attribute name of a bi-directional relationship.

Parameters:

invRelationshipField - the attribute name of the inverse side of a bi-directional relationship or null if the relationship is uni-directional.

getRelationshipField

```
public String getRelationshipField()
```

Retrieve the name of the attribute in the source class that references the key of the target class.

Returns:

the name of the relationship attribute.

setRelationshipField

```
public void setRelationshipField(String relationshipField)
```

Set the name of the attribute in the source class that references the key of the target class.

Parameters:

relationshipField - the name of the relationship attribute.

getSourceClass

```
public String getSourceClass()
```

Retrieve the name of the class representing the source of a relationship.

Returns:

the source class

setSourceClass

```
public void setSourceClass(String sourceClass)
```

Set the name of the class representing the source of a relationship.

Parameters:

sourceClass - the source class

getTargetClass

```
public String getTargetClass()
```

Retrieve the name of the class representing the target of a relationship.

Returns:

the target class

setTargetClass


```
public void setTargetClass(String targetClass)
```

Set the name of the class representing the target of a relationship.

Parameters:

targetClass - the target class

equals

```
public boolean equals(Object o)
```

Overrides:

[equals](#) in class [Object](#)

See Also:

[Object.equals\(java.lang.Object\)](#)

hashCode

```
public int hashCode()
```

Overrides:

[hashCode](#) in class [Object](#)

See Also:

[Object.hashCode\(\)](#)

toString

```
public String toString()
```

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Class QueryMapping

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.config.QueryMapping

All Implemented Interfaces:

[Serializable](#)

```
public class QueryMapping
extends Object
implements Serializable
```

A QueryMapping maps a Java class to a BackingMap and allows a map to be included in a query. It also indicates whether the query engine should use a getter method or direct field access to access fields in the value class.

For example, class Department is the value class that is stored in the "DepartmentMap" BackingMap and the key is an Integer.

```
public class Department {
    private int id;
    private Collection emps;

    public void setEmps(Collection emps) {
        this.emps = emps;
    }

    public Collection getEmps() {
        return emps;
    }
    ...
}
```

The QueryMapping would be created as follows:

```
...
QueryConfig queryConfig = new QueryConfig();
queryConfig.addMapping(new QueryMapping(
    "DepartmentMap", Department.class.getName(), "id", QueryMapping.PROPERTY_ACCESS)
objectGrid.setQueryConfig(queryConfig);
...
```

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Field Summary

s
t
a

t i c	FIELD_ACCESS This constant indicates to use direct field access to read the field values
i n t	
s t a t i c	PROPERTY_ACCESS This constant indicates to use JavaBean property-style get methods to read the field values from the Java object stored in the BackingMap.
i n t	

Constructor Summary	
i n t	QueryMapping () Default constructor.
i n t	QueryMapping (String mapName, String valueClass, String primaryKeyField) Constructor for creating a basic QueryMapping instance with a default access type of PROPERTY_ACCESS .
i n t	QueryMapping (String mapName, String valueClass, String primaryKeyField, int accessType) Constructor for creating a QueryMapping instance.

Method Summary	
b o o l e a n	equals (Object o)
i n t	getAccessType () Retrieve the method in which the query engine will access the value class object stored in the BackingMap.
S t r i n g	getMapName () Retrieve the BackingMap name associated with this mapping.
S t r i n g	getPrimaryKeyField () Retrieve the name of the attribute in the value class object that is also the primary key of the BackingMap.
S t r i n g	getValueClass () Retrieve the type of object that is stored in the BackingMap.
i n t	hashCode ()

v o i d	setAccessType (int accessType) Set the method in which the query engine will access the value class object stored in the BackingMap.
v o i d	setMapName (String mapName) Set the BackingMap name associated with this mapping.
v o i d	setPrimaryKeyField (String primaryKeyField) Set the name of the attribute in the value class object that is also the primary key of the BackingMap.
v o i d	setValueClass (String valueClass) Set the type of object that is stored in the BackingMap.
S t r i n g	toString ()

Methods inherited from class java.lang.[Object](#)

[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

FIELD_ACCESS

```
public static final int FIELD_ACCESS
```

This constant indicates to use direct field access to read the field values

See Also:

[setAccessType\(int\)](#), [getAccessType\(\)](#), [Constant Field Values](#)

PROPERTY_ACCESS

```
public static final int PROPERTY_ACCESS
```

This constant indicates to use JavaBean property-style get methods to read the field values from the Java object stored in the BackingMap. PROPERTY_ACCESS is the default.

See Also:

[setAccessType\(int\)](#), [getAccessType\(\)](#), [Constant Field Values](#)

Constructor Detail

QueryMapping

```
public QueryMapping()
```

Default constructor.

QueryMapping

```
public QueryMapping(String mapName,  
                   String valueClass,  
                   String primaryKeyField)
```

Constructor for creating a basic QueryMapping instance with a default access type of [PROPERTY_ACCESS](#).

The mapName and valueClass must not be null.

Parameters:

mapName - the name of the BackingMap to map
valueClass - the class of object stored in the BackingMap's value.
primaryKeyField - the optional name of the primary key field of the class.

QueryMapping

```
public QueryMapping(String mapName,  
                   String valueClass,  
                   String primaryKeyField,  
                   int accessType)
```

Constructor for creating a QueryMapping instance. The mapName and valueClass must not be null.

Parameters:

mapName - the name of the BackingMap to map
valueClass - the class of object stored in the BackingMap's value.
primaryKeyField - the optional name of the primary key field of the class.
accessType - the method ([PROPERTY_ACCESS](#) or [FIELD_ACCESS](#)) in which the query engine will access the persistent data in the value object.

Method Detail

getMapName

```
public String getMapName()
```

Retrieve the BackingMap name associated with this mapping.

Returns:

the BackingMap name.

setMapName

```
public void setMapName(String mapName)
```

Set the BackingMap name associated with this mapping.

getValueClass

```
public String getValueClass()
```

Retrieve the type of object that is stored in the BackingMap.

Returns:

the object type that is stored in the BackingMap's value.

setValueClass

```
public void setValueClass(String valueClass)
```

Set the type of object that is stored in the BackingMap.

getAccessType

```
public int getAccessType()
```

Retrieve the method in which the query engine will access the value class object stored in the BackingMap.

Returns:

Returns the accessType.

See Also:

[PROPERTY_ACCESS](#), [FIELD_ACCESS](#)

setAccessType

```
public void setAccessType(int accessType)
```

Set the method in which the query engine will access the value class object stored in the BackingMap.

Parameters:

accessType - the accessType.

See Also:

[PROPERTY_ACCESS](#), [FIELD_ACCESS](#)

getPrimaryKeyField

```
public String getPrimaryKeyField()
```

Retrieve the name of the attribute in the value class object that is also the primary key of the BackingMap.

This value is optional.

Returns:

the primaryKeyField.

setPrimaryKeyField

```
public void setPrimaryKeyField(String primaryKeyField)
```

Set the name of the attribute in the value class object that is also the primary key of the BackingMap.

Parameters:

primaryKeyField - the name of the primary key attribute or null if not set.

equals

```
public boolean equals(Object o)
```

Overrides:

[equals](#) in class [Object](#)

See Also:

[Object.equals\(java.lang.Object\)](#)

hashCode

public int **hashCode**()

Overrides:

[hashCode](#) in class [Object](#)

See Also:

[Object.hashCode\(\)](#)

toString

public [String](#) **toString**()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Class QueryConfig

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```

public class QueryConfig
    extends Object
    implements Serializable
  
```

This QueryConfig represents a schema configuration for a QueryManager. A QueryConfig object contains a set of QueryMapping objects and a set of QueryRelationship objects.

Users use addQuerySchema method to add a QueryMapping object and use addRelationship method to add a QueryRelationship object into this QueryConfig object.

Since:

WAS XD 6.1, XC10

See Also:

[ObjectGrid.setQueryConfig\(QueryConfig\)](#), [QueryMapping](#), [QueryRelationship](#), [Serialized Form](#)

Constructor Summary

[QueryConfig\(\)](#)
 Default constructor.

Method Summary

[void](#) [addQueryMapping](#)([QueryMapping](#) mapping)
 Add a QueryMapping to this QueryConfig

[void](#) [addQueryRelationship](#)([QueryRelationship](#) relation)
 Add a QueryRelationship to this QueryConfig

[QueryMapping](#) [getQueryMappings](#)()
 Retrieve all added QueryMappings

Q
u
e
r
y
R
e
l
a
t
i
o
n
s
h
i
p
[
]

[getQueryRelationships\(\)](#)

Retrieve all added QueryRelationships

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

QueryConfig

public [QueryConfig](#)()

Default constructor.

Method Detail

addQueryMapping

public void [addQueryMapping](#)([QueryMapping](#) mapping)

Add a QueryMapping to this QueryConfig

Parameters:

mapping - the QueryMapping to add.

addQueryRelationship

public void [addQueryRelationship](#)([QueryRelationship](#) relation)

Add a QueryRelationship to this QueryConfig

Parameters:

relation - the QueryRelationship to add.

getQueryMappings

public [QueryMapping](#)[] [getQueryMappings](#)()

Retrieve all added QueryMappings

Returns:

array of QueryMapping

getQueryRelationships

```
public QueryRelationship[] getQueryRelationships()
```

Retrieve all added QueryRelationships

Returns:

array of QueryRelationship

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.config

Interface PluginType

All Superinterfaces:

[Serializable](#)

```
public interface PluginType
extends Serializable
```

Every Plugin has a PluginType. The PluginType is specified during Plugin creation.

ObjectGridConfiguration objects support the following PluginTypes for overriding client-side ObjectGrid plugins:

- PluginType.TRANSACTION_CALLBACK
- PluginType.OBJECTGRID_EVENT_LISTENER
- PluginType.COLLISION_ARBITER

BackingMapConfiguration objects support the following PluginTypes for overriding client-side BackingMap plugins:

- PluginType.EVICTOR
- PluginType.MAP_EVENT_LISTENER

Since:

WAS XD 6.0.1.2, XC10

See Also:

[ObjectGridConfigFactory.createPlugin\(PluginType, String\)](#)

Field Summary

s
t
a
t
i
c
P
L
U
G
I
N
T
Y
P
E

[COLLISION_ARBITER](#)

PluginType.COLLISION_ARBITER can be used to attach a CollisionArbiter plugin to an ObjectGrid.

s
t
a
t
i
c
P
L
U
G
I
N
T
Y
P
E

[EVICTOR](#)

PluginType.EVICTOR can be used to attach an [Evictor](#) to a BackingMap.

q
i
n
T
Y
p
e

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[MAP_EVENT_LISTENER](#)

PluginType.MAP_EVENT_LISTENER can be used to attach a [MapEventListener](#) to a BackingMap.

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[MAP_LIFECYCLE_LISTENER](#)

PluginType.MAP_LIFECYCLE_LISTENER can be used to attach a BackingMapLifecycleListener plugin to an BackingMap

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[MAP_SERIALIZER_PLUGIN](#)

PluginType.MAP_SERIALIZER_PLUGIN can be used to attach a MapSerializerPlugin to a BackingMap to serialize keys.

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[OBJECTGRID_EVENT_LISTENER](#)

PluginType.OBJECTGRID_EVENT_LISTENER can be used to attach an [ObjectGridEventListener](#) plugin to an ObjectGrid

s
t
a
t

i
c
P
L
U
G
I
N
T
Y
P
E

[OBJECTGRID_LIFECYCLE_LISTENER](#)

PluginType.OBJECTGRID_LIFECYCLE_LISTENER can be used to attach an [ObjectGridLifecycleListener](#) plugin to an ObjectGrid

s
t
a
t
i
c
P
L
U
G
I
N
T
Y
P
E

[TRANSACTION_CALLBACK](#)

PluginType.TRANSACTION_CALLBACK can be used to attach a [TransactionCallback](#) plugin to an ObjectGrid.

Field Detail

OBJECTGRID_EVENT_LISTENER

static final [PluginType](#) OBJECTGRID_EVENT_LISTENER

PluginType.OBJECTGRID_EVENT_LISTENER can be used to attach an [ObjectGridEventListener](#) plugin to an ObjectGrid

TRANSACTION_CALLBACK

static final [PluginType](#) TRANSACTION_CALLBACK

PluginType.TRANSACTION_CALLBACK can be used to attach a [TransactionCallback](#) plugin to an ObjectGrid.

COLLISION_ARBITER

static final [PluginType](#) COLLISION_ARBITER

PluginType.COLLISION_ARBITER can be used to attach a CollisionArbiter plugin to an ObjectGrid.

OBJECTGRID_LIFECYCLE_LISTENER

static final [PluginType](#) OBJECTGRID_LIFECYCLE_LISTENER

PluginType.OBJECTGRID_LIFECYCLE_LISTENER can be used to attach an [ObjectGridLifecycleListener](#) plugin to an ObjectGrid

Since:

7.1.1

EVICTOR

static final [PluginType](#) EVICTOR

`PluginType.EVICTOR` can be used to attach an [Evictor](#) to a `BackingMap`.

MAP_EVENT_LISTENER

static final [PluginType](#) MAP_EVENT_LISTENER

`PluginType.MAP_EVENT_LISTENER` can be used to attach a [MapEventListener](#) to a `BackingMap`.

MAP_SERIALIZER_PLUGIN

static final [PluginType](#) MAP_SERIALIZER_PLUGIN

`PluginType.MAP_SERIALIZER_PLUGIN` can be used to attach a `MapSerializerPlugin` to a `BackingMap` to serialize keys.

Since:

7.1.1

MAP_LIFECYCLE_LISTENER

static final [PluginType](#) MAP_LIFECYCLE_LISTENER

`PluginType.MAP_LIFECYCLE_LISTENER` can be used to attach a `BackingMapLifecycleListener` plugin to an `BackingMap`

Since:

7.1.1

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
OD

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Interface Plugin

All Superinterfaces:

[Serializable](#)

```
public interface Plugin
extends Serializable
```

This interface represents an ObjectGrid or BackingMap plugin. An ObjectGridConfiguration object supports the following Plugins:

- PluginType#OBJECTGRID_EVENT_LISTENER
- PluginType#TRANSACTION_CALLBACK

A BackingMapConfiguration object supports the following Plugins:

- PluginType#EVICTOR
- PluginType#MAP_EVENT_LISTENER

A Plugin object has following attributes:

- pluginType: the support PluginTypes are listed above
- className: the plugin implementation class name. This class name will be used to construct the class object. A default constructor must be implemented.
- configProperties: the JavaBean properties for this plugin implementation object.

A plugin object can be created by using ObjectGridConfigFactory.createPlugin(PluginType, String) method. Please refer to ObjectGridConfigFactory for detailed example.

Since:

WAS XD 6.0.1.2, XC10

Method Summary	
V o i d	addConfigProperty (ConfigProperty configProp) Add a ConfigProperty to this object.
S t r i n g	getClassName () Get the String representation of the class name of this Plugin
L i s t	getConfigProperties () Get the ConfigProperty objects that have been set on this object.
S t r	getOSGiService ()

i n g	Get the OSGi service name configured for this Plugin.
P L U G I N T Y P E	getPluginType() Get the PluginType for this Plugin.
v o i d	setClassName(String className) The class name that is set must be an implementor of the PluginTypeImpl for this Plugin.
v o i d	setConfigProperties(List configPropList) Set the ConfigProperties for this object
v o i d	setOSGiService(String osgiService) Set the OSGi service name configured for this Plugin.
v o i d	setPluginType(PluginType pluginType) Set the PluginType for this Plugin.

Method Detail

addConfigProperty

void **addConfigProperty**([ConfigProperty](#) configProp)

Add a ConfigProperty to this object.

Parameters:

configProp - - ConfigProperty to add to this object

setConfigProperties

void **setConfigProperties**([List](#) configPropList)

Set the ConfigProperties for this object

Parameters:

configPropList -

getConfigProperties

[List](#) **getConfigProperties**()

Get the ConfigProperty objects that have been set on this object.

Returns:

a List of the ConfigProperty objects that have been added to this object.

See Also:

[ConfigProperty](#)

getPluginType

[PluginType](#) `getPluginType()`

Get the PluginType for this Plugin.

Returns:
the PluginType for this Plugin

`setPluginType`

`void setPluginType(PluginType pluginType)`

Set the PluginType for this Plugin.

The ObjectGridConfiguration plugins include

- `PluginType.TRANSACTION_CALLBACK`
- `PluginType.OBJECTGRID_EVENT_LISTENER`

The BackingMapConfiguration plugins include

- `PluginType.EVICTOR`
- `PluginType.MAP_EVENT_LISTENER`

Parameters:
pluginType -

`getClassName`

[String](#) `getClassName()`

Get the String representation of the class name of this Plugin

Returns:
the String representation of the class name

`setClassName`

`void setClassName(String className)`

The class name that is set must be an implementor of the PluginTypeImpl for this Plugin. For example, if the type of this Plugin is PluginType.EVICTOR, then the className must be an implementor of the com.ibm.websphere.objectgrid.plugins.Evictor interface.

Parameters:
className - - the class name of the Class that implements the PluginType

`getOSGiService`

[String](#) `getOSGiService()`

Get the OSGi service name configured for this Plugin. If an OSGi service name is configured for this Plugin, the className configured for this Plugin is ignored.

Returns:
the OSGi service name configured for this Plugin.

Since:
7.1.1

`setOSGiService`

`void setOSGiService(String osgiService)`

Set the OSGi service name configured for this Plugin. If an OSGi service name is configured for this Plugin, the className configured for this Plugin is ignored.

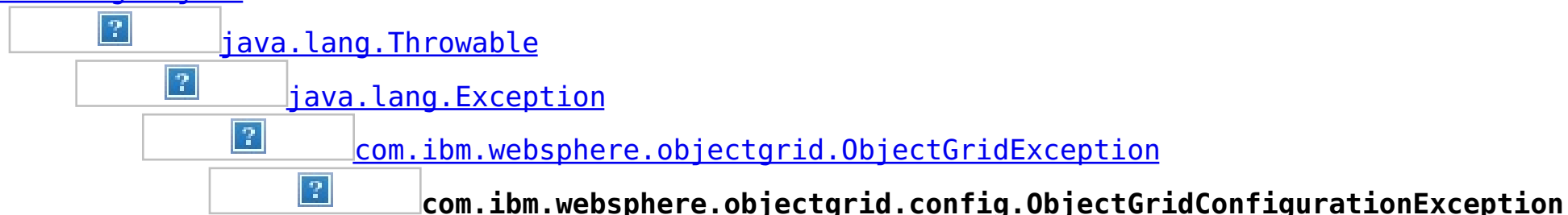
Parameters:
osgiService - the OSGi service name configured for this Plugin.

Since:
7.1.1

com.ibm.websphere.objectgrid.config

Class ObjectGridConfigurationException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class ObjectGridConfigurationException
extends ObjectGridException
```

Thrown when a problem with the current configuration is found. This exception may be thrown when the configuration specified in the deployment policy, ObjectGrid descriptor or security descriptor is incorrect.

Since:

7.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridConfigurationException](#)()

Constructs a new ObjectGridConfigurationException with null as its detail message.

[ObjectGridConfigurationException](#)(String message)

Constructs a new ObjectGridConfigurationException with the specified detail message.

[ObjectGridConfigurationException](#)(String message, [Throwable](#) cause)

Constructs a new ObjectGridConfigurationException with the specified detail message and cause.

[ObjectGridConfigurationException](#)([Throwable](#) cause)

Constructs a new ObjectGridConfigurationException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridConfigurationException

```
public ObjectGridConfigurationException()
```

Constructs a new ObjectGridConfigurationException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ObjectGridConfigurationException

```
public ObjectGridConfigurationException(String message)
```

Constructs a new ObjectGridConfigurationException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridConfigurationException

```
public ObjectGridConfigurationException(Throwable cause)
```

Constructs a new ObjectGridConfigurationException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for ObjectGridConfigurationException that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

ObjectGridConfigurationException

```
public ObjectGridConfigurationException(String message,  
                                       Throwable cause)
```

Constructs a new ObjectGridConfigurationException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in

this ObjectGridConfigurationException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the getMessage method).

cause - the cause (which is saved for later retrieval by the getCause method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Class	Next Class	Frames No Frames All					
			Classes				
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD				

`com.ibm.websphere.objectgrid.config`
Interface ObjectGridConfiguration

public interface **ObjectGridConfiguration**

An ObjectGridConfiguration object can be used to override ObjectGrid plugins on the client side. The `com.ibm.websphere.objectgrid.plugins.ObjectGridEventListener` and the `com.ibm.websphere.objectgrid.plugins.TransactionCallback` Plugins can be overridden.

Since:

WAS XD 6.0.1.2, XC10

See Also:

[ObjectGridEventListener](#), [TransactionCallback](#)

Method Summary	
V o i d	addBackingMapConfiguration (BackingMapConfiguration backingMapConfiguration) Add a BackingMapConfiguration to this ObjectGridConfiguration.
V o i d	addPlugin (Plugin plugin) Add a Plugin to this ObjectGridConfiguration.
L i s t	getBackingMapConfigurations () Get the List of BackingMapConfiguration objects that are attached to this ObjectGridConfiguration object
S t r i n g	getName () Get the name of this ObjectGridConfiguration
L i s t	getPlugins () Get the Plugins that have been attached to this ObjectGridConfiguration.
i n t	getTxIsolation () Retrieves the default transaction isolation level.
i n t	getTxTimeout () Gets the transaction timeout.
V o i d	setBackingMapConfigurations (List backingMapConfigList) Set the BackingMapConfiguration objects for this ObjectGridConfiguration.

void [setPlugins](#)([List](#) pluginList)
Set the Plugins for this ObjectGridConfiguration.

void [setTxIsolation](#)(int level)
Sets the default transaction isolation level for all sessions created by the ObjectGrid.

void [setTxTimeout](#)(int seconds)
Sets the transaction timeout

Method Detail

getName

[String](#) getName()

Get the name of this ObjectGridConfiguration

Returns:
the name of this ObjectGridConfiguration

addBackingMapConfiguration

void [addBackingMapConfiguration](#)([BackingMapConfiguration](#) backingMapConfiguration)

Add a BackingMapConfiguration to this ObjectGridConfiguration.

Parameters:
backingMapConfiguration -

setBackingMapConfigurations

void [setBackingMapConfigurations](#)([List](#) backingMapConfigList)

Set the BackingMapConfiguration objects for this ObjectGridConfiguration. Any BackingMapConfiguration objects that were previously attached to this ObjectGridConfiguration object will be overridden.

Parameters:
backingMapConfigList - - A List of BackingMapConfiguration objects.

See Also:
[BackingMapConfiguration](#)

getBackingMapConfigurations

[List](#) [getBackingMapConfigurations](#)()

Get the List of BackingMapConfiguration objects that are attached to this ObjectGridConfiguration object

Returns:
a List of BackingMapConfiguration objects

See Also:

addPlugin

void **addPlugin**([Plugin](#) plugin)

Add a Plugin to this ObjectGridConfiguration. The Plugins that can be overridden on a client-side ObjectGrid are `com.ibm.websphere.objectgrid.plugins.ObjectGridEventListener` and `com.ibm.websphere.objectgrid.plugins.TransactionCallback`.

Parameters:

plugin -

See Also:

[setPlugins\(List\)](#), [Plugin](#)

setPlugins

void **setPlugins**([List](#) pluginList)

Set the Plugins for this ObjectGridConfiguration. Any Plugins that were previously attached to this ObjectGridConfiguration object will be overridden.

Parameters:

pluginList - - a List of Plugins

See Also:

[addPlugin\(Plugin\)](#), [Plugin](#)

getPlugins

[List](#) **getPlugins**()

Get the Plugins that have been attached to this ObjectGridConfiguration.

Returns:

a List of Plugin objects

See Also:

[Plugin](#)

setTxTimeout

void **setTxTimeout**(int seconds)

Sets the transaction timeout

Parameters:

seconds -

Since:

7.1.0.3

See Also:

[ObjectGrid.setTxTimeout\(int\)](#)

getTxTimeout

int **getTxTimeout**()

Gets the transaction timeout. The value is in seconds.

Returns:
the transaction timeout in seconds

Since:
7.1.0.3

See Also:
[ObjectGrid.getTxTimeout\(\)](#)

setTxIsolation

void **setTxIsolation**(int level)

Sets the default transaction isolation level for all sessions created by the ObjectGrid. The constants defined in the Session interface are the possible transaction isolation levels. The default is [Session.TRANSACTION_REPEATABLE_READ](#).

Parameters:
level - one of the following Session constants: [Session.TRANSACTION_READ_UNCOMMITTED](#), [Session.TRANSACTION_READ_COMMITTED](#) or [Session.TRANSACTION_REPEATABLE_READ](#) or 0 if the TransactionIsolation should not be set.

Since:
7.1.1

getTxIsolation

int **getTxIsolation**()

Retrieves the default transaction isolation level.

Returns:
the current transaction isolation level.

Since:
7.1.1

See Also:
[setTxIsolation\(int\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Class ObjectGridConfigFactory

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory

```
public final class ObjectGridConfigFactory  
extends Object
```

This is the configuration factory for ObjectGrid configuration entities. Users are expected to use static methods of this factory to create ObjectGrid configuration objects.

Here is a list of static methods used to create configuration objects:

- createObjectGridConfiguration(String): create an ObjectGridConfiguration object
- createBackingMapConfiguration(String): create a BackingMapConfiguration object by passing a backing map name
- createConfigProperty(ConfigPropertyType, String, String): create a ConfigProperty object
- createPlugin(PluginType, String): create a Plugin object

Below is an example of creating an ObjectGrid configuration. A Plugin is added to the ObjectGridConfiguration object: com.ibm.websphere.objectgrid.plugins.ObjectGridEventListener plugin.

A BackingMapConfiguration called "myBackingMap" is then created and added to the ObjectGridConfiguration. This BackingMapConfiguration also has an Evictor Plugin configured.

Once the ObjectGridConfiguration object has been created, it can be used to call either of these methods

- com.ibm.websphere.objectgrid.ObjectGridManager.putOverrideObjectGridConfigurations(String, List)
- com.ibm.websphere.objectgrid.ObjectGridManager.setOverrideObjectGridConfigurations(Map)

to set configuration objects, prior to connecting.

```
// Create an ObjectGridConfiguration object  
ObjectGridConfiguration ogConfig = ObjectGridConfigFactory.createObjectGridConfiguration(ogName);  
  
// create ObjectGridEventListener plugin  
Plugin eventListener = ObjectGridConfigFactory.createPlugin(PluginType.OBJECTGRID_EVENT_LISTENER,  
"com.ibm.test.MyOgEventListener");  
  
// Add plugin to ObjectGridConfiguration object  
ogConfig.addPlugin(eventListener);  
  
// Create a BackingMapConfiguration object  
BackingMapConfiguration bmConfig = ObjectGridConfigFactory.createBackingMapConfiguration("mybacki  
ngMap");  
  
// Add BackingMapConfiguration object to ObjectGridConfiguration object  
ogConfig.addBackingMapConfiguration(bmConfig);  
  
// Set the number of buckets to 1000  
bmConfig.setNumberOfBuckets(1000);
```

```

// Create a Evictor plugin for this backing map.
Plugin evictor = ObjectGridConfigFactory.createPlugin(
BackingMapConfiguration.PLUGIN_EVICTOR,
com.acme.myEvictorImpl.class.getName());

// add Evictor Plugin to the BackingMapConfiguration
bmConfig.addPlugin(evictor);

// Create a ConfigProperty for the Evictor plugin
ConfigProperty sizeProperty = ObjectGridConfigFactory.createConfigProperty(
    ConfigPropertyType.INT_PRIM, "size", "153");

// Add the ConfigProperty to the Evictor plugin
evictor.setConfigProperty(sizeProperty);

```

Since:

WAS XD 6.0.1.2, XC10

See Also:

[ObjectGridConfiguration](#), [BackingMapConfiguration](#), [Plugin](#), [ConfigProperty](#)

Constructor Summary

[ObjectGridConfigFactory](#)()

Method Summary

[createBackingMapConfiguration](#)([String](#) backingMapConfigName)
 Create a BackingMapConfiguration object

[createConfigProperty](#)([ConfigPropertyType](#) configPropType, [String](#) name, [String](#) value)
 Create a ConfigPropety object for use on a Plugin.

P
r
o
p
e
r
t
y

s
t
a
t
i
c
O
b
j
e
c
t
G
r
i
d
C
o
n
f
i
g
u
r
e
t
i
o
n

[createObjectGridConfiguration](#)(String objectGridConfigName)
Create an ObjectGridConfiguration object.

s
t
a
t
i
c
P
l
u
g
i
n

[createOSGiServicePlugin](#)(PluginType pluginType, String osgiServiceName)
Create an OSGi Service Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

s
t
a
t
i
c
P
l
u
g
i
n

[createPlugin](#)(PluginType pluginType, String className)
Create a Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridConfigFactory

```
public ObjectGridConfigFactory()
```

Method Detail

createPlugin

```
public static Plugin createPlugin(PluginType pluginType,  
                                   String className)
```

Create a Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

Parameters:

pluginType - the PluginType.
className - of the Plugin implementation class to instantiate

Returns:

a Plugin instance

See Also:

[PluginType.OBJECTGRID_EVENT_LISTENER](#), [PluginType.TRANSACTION_CALLBACK](#),
[PluginType.COLLISION_ARBITER](#), [PluginType.EVICTOR](#), [PluginType.MAP_EVENT_LISTENER](#),
[PluginType.OBJECTGRID_LIFECYCLE_LISTENER](#), [PluginType.MAP_LIFECYCLE_LISTENER](#)

createObjectGridConfiguration

```
public static ObjectGridConfiguration createObjectGridConfiguration(String objectGridConfigName)
```

Create an ObjectGridConfiguration object.

Parameters:

objectGridConfigName - the name that will be assigned to this ObjectGridConfiguration object

Returns:

the ObjectGridConfiguration object

createBackingMapConfiguration

```
public static BackingMapConfiguration createBackingMapConfiguration(String backingMapConfigName)
```

Create a BackingMapConfiguration object

Parameters:

backingMapConfigName - the name to assign to this BackingMapConfiguration

Returns:

the BackingMapConfiguration object

createConfigProperty

```
public static ConfigProperty createConfigProperty(ConfigPropertyType configPropType,  
                                                  String name,  
                                                  String value)
```

Create a ConfigProperty object for use on a Plugin.

The Plugin should have a set method that corresponds to the name of this ConfigProperty. The method must accept a parameter of the ConfigPropertyType that is specified on this ConfigProperty. For example, if the name of this ConfigProperty is set to "size", and the type is ConfigPropertyType.INT_PRIM, then the Plugin must have the method setSize(int). The value of the ConfigProperty will be passed to the setter of the Plugin when an ObjectGrid is created based on this configuration.

Parameters:

configPropType - ConfigPropertyType of the ConfigProperty. Part of the set method's signature, the type of parameter the set method requires. Valid name - of the ConfigProperty. It must correspond to the name of a set method on the Plugin.

value - of the ConfigProperty. This value will be passed to the set method on the Plugin.

Returns:

the ConfigProperty object

See Also:

[ConfigPropertyType.STRING_JAVA_LANG](#), [ConfigPropertyType.BOOLEAN_JAVA_LANG](#), [ConfigPropertyType.BOOLEAN_PRIM](#), [ConfigPropertyType.BYTE_JAVA_LANG](#), [ConfigPropertyType.BYTE_PRIM](#), [ConfigPropertyType.CHARACTER_JAVA_LANG](#), [ConfigPropertyType.CHAR_PRIM](#), [ConfigPropertyType.DOUBLE_JAVA_LANG](#), [ConfigPropertyType.DOUBLE_PRIM](#), [ConfigPropertyType.FLOAT_JAVA_LANG](#), [ConfigPropertyType.FLOAT_PRIM](#), [ConfigPropertyType.INTEGER_JAVA_LANG](#), [ConfigPropertyType.INT_PRIM](#), [ConfigPropertyType.LONG_JAVA_LANG](#), [ConfigPropertyType.LONG_PRIM](#), [ConfigPropertyType.SHORT_JAVA_LANG](#)

createOSGiServicePlugin

```
public static Plugin createOSGiServicePlugin(PluginType pluginType,  
                                             String osgiServiceName)
```

Create an OSGi Service Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

Parameters:

pluginType - the PluginType.
osgiServiceName - the OSGi service name

Returns:

a Plugin instance

Since:

7.1.1

See Also:

[PluginType.OBJECTGRID_EVENT_LISTENER](#), [PluginType.TRANSACTION_CALLBACK](#), [PluginType.COLLISION_ARBITER](#), [PluginType.EVICTOR](#), [PluginType.MAP_EVENT_LISTENER](#), [PluginType.OBJECTGRID_LIFECYCLE_LISTENER](#), [PluginType.MAP_LIFECYCLE_LISTENER](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Class	Next Class	Frames	No Frames	All			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid.config
Class ConfigPropertyType

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public final class ConfigPropertyType
extends Object
implements Serializable
```

ConfigPropertyType is used to set the type of an attribute on a Plugin. The Java primitives, their java.lang counterparts, and java.lang.String are the supported types.

Since:

WAS XD 6.0.1.2, XC10

See Also:

[ObjectGridConfigFactory.createConfigProperty\(ConfigPropertyType, String, String\)](#),
[Plugin.setPluginType\(PluginType\)](#), [Serialized Form](#)

Field Summary

s t a t i c C o n f i g P r o p e r t y T y p e	<p>BOOLEAN_JAVA_LANG ConfigPropertyType.BOOLEAN_JAVA_LANG can be used to set a property of type java.lang.Boolean on a plugin.</p>
s t a t i c C o n	

f
i
j
q
P
r
o
p
e
r
t
y
T
y
p
e

BOOLEAN_PRIM

ConfigPropertyType.BOOLEAN_PRIM can be used to set a property of type boolean on a plugin.

s
t
a
t
i
c
C
o
n
f
i
q
P
r
o
p
e
r
t
y
T
y
p
e

BYTE_JAVA_LANG

ConfigPropertyType.BYTE_JAVA_LANG can be used to set a property of type java.lang.Byte on a plugin.

s
t
a
t
i
c
C
o
n
f
i
q
P
r
o
p
e
r
t
y
T
y
p
e

BYTE_PRIM

ConfigPropertyType.BYTE_PRIM can be used to set a property of type byte on a plugin.

s
t
a
t
i
c
C
o
n
f
i
q
P

CHAR_PRIM

ConfigPropertyType.CHAR_PRIM can be used to set a property of type char on a

r
o
p
e
r
t
i
v
e

plugin.

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
i
v
e

CHARACTER_JAVA_LANG

ConfigPropertyType.CHARACTER_JAVA_LANG can be used to set a property of type java.lang.Character on a plugin.

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
i
v
e

DOUBLE_JAVA_LANG

ConfigPropertyType.DOUBLE_JAVA_LANG can be used to set a property of type java.lang.Double on a plugin.

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p

DOUBLE_PRIM

ConfigPropertyType.DOUBLE_PRIM can be used to set a property of type double on a plugin.

e
r
t
i
c
a
t
e
s

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
y
t
y
p
e

FLOAT_JAVA_LANG

ConfigPropertyType.FLOAT_JAVA_LANG can be used to set a property of type java.lang.Float on a plugin.

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
y
t
y
p
e

FLOAT_PRIM

ConfigPropertyType.FLOAT_PRIM can be used to set a property of type float on a plugin.

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
y
t
y
p
e

INT_PRIM

ConfigPropertyType.INT_PRIM can be used to set a property of type int on a plugin.

I
V
P
E

s
t
a
t
i
c
C
O
N
F
I
G
P
R
O
P
E
R
T
Y
T
Y
P
E

INTEGER_JAVA_LANG

ConfigPropertyType.INTEGER_JAVA_LANG can be used to set a property of type java.lang.Integer on a plugin.

s
t
a
t
i
c
C
O
N
F
I
G
P
R
O
P
E
R
T
Y
T
Y
P
E

LONG_JAVA_LANG

ConfigPropertyType.LONG_JAVA_LANG can be used to set a property of type java.lang.Long on a plugin.

s
t
a
t
i
c
C
O
N
F
I
G
P
R
O
P
E
R
T
Y
T
Y
P
E

LONG_PRIM

ConfigPropertyType.LONG_PRIM can be used to set a property of type long on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
Y
T
Y
P
e

SHORT_JAVA_LANG

ConfigPropertyType.SHORT_JAVA_LANG can be used to set a property of type java.lang.Short on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
Y
T
Y
P
e

SHORT_PRIM

ConfigPropertyType.SHORT_PRIM can be used to set a property of type short on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
Y
T
Y
P
e

STRING_JAVA_LANG

ConfigPropertyType.STRING_JAVA_LANG can be used to set a property of type java.lang.String on a plugin.

Method Summary

b o o l e a n	equals(Object o)
i n t	getId() Get the raw value of this ConfigPropertyType.
i n t	hashCode()
S t r i n g	toString()
s t a t i c C o n f i g P r o p e r t y T y p e	valueOf(byte id) Given the raw value of a ConfigPropertyType, this method returns a ConfigPropertyType object, or null if the raw value does not match an existing type.

Methods inherited from class java.lang.[Object](#)
[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

INTEGER_JAVA_LANG

public static final [ConfigPropertyType](#) INTEGER_JAVA_LANG

ConfigPropertyType.INTEGER_JAVA_LANG can be used to set a property of type java.lang.Integer on a plugin.

INT_PRIM

public static final [ConfigPropertyType](#) INT_PRIM

ConfigPropertyType.INT_PRIM can be used to set a property of type int on a plugin.

BOOLEAN_JAVA_LANG

public static final [ConfigPropertyType](#) **BOOLEAN_JAVA_LANG**

`ConfigPropertyType.BOOLEAN_JAVA_LANG` can be used to set a property of type `java.lang.Boolean` on a plugin.

BOOLEAN_PRIM

public static final [ConfigPropertyType](#) **BOOLEAN_PRIM**

`ConfigPropertyType.BOOLEAN_PRIM` can be used to set a property of type `boolean` on a plugin.

CHARACTER_JAVA_LANG

public static final [ConfigPropertyType](#) **CHARACTER_JAVA_LANG**

`ConfigPropertyType.CHARACTER_JAVA_LANG` can be used to set a property of type `java.lang.Character` on a plugin.

CHAR_PRIM

public static final [ConfigPropertyType](#) **CHAR_PRIM**

`ConfigPropertyType.CHAR_PRIM` can be used to set a property of type `char` on a plugin.

BYTE_JAVA_LANG

public static final [ConfigPropertyType](#) **BYTE_JAVA_LANG**

`ConfigPropertyType.BYTE_JAVA_LANG` can be used to set a property of type `java.lang.Byte` on a plugin.

BYTE_PRIM

public static final [ConfigPropertyType](#) **BYTE_PRIM**

`ConfigPropertyType.BYTE_PRIM` can be used to set a property of type `byte` on a plugin.

SHORT_JAVA_LANG

public static final [ConfigPropertyType](#) **SHORT_JAVA_LANG**

`ConfigPropertyType.SHORT_JAVA_LANG` can be used to set a property of type `java.lang.Short` on a plugin.

SHORT_PRIM

public static final [ConfigPropertyType](#) **SHORT_PRIM**

`ConfigPropertyType.SHORT_PRIM` can be used to set a property of type `short` on a plugin.

LONG_JAVA_LANG

public static final [ConfigPropertyType](#) LONG_JAVA_LANG

`ConfigPropertyType.LONG_JAVA_LANG` can be used to set a property of type `java.lang.Long` on a plugin.

LONG_PRIM

public static final [ConfigPropertyType](#) LONG_PRIM

`ConfigPropertyType.LONG_PRIM` can be used to set a property of type `long` on a plugin.

FLOAT_JAVA_LANG

public static final [ConfigPropertyType](#) FLOAT_JAVA_LANG

`ConfigPropertyType.FLOAT_JAVA_LANG` can be used to set a property of type `java.lang.Float` on a plugin.

FLOAT_PRIM

public static final [ConfigPropertyType](#) FLOAT_PRIM

`ConfigPropertyType.FLOAT_PRIM` can be used to set a property of type `float` on a plugin.

DOUBLE_JAVA_LANG

public static final [ConfigPropertyType](#) DOUBLE_JAVA_LANG

`ConfigPropertyType.DOUBLE_JAVA_LANG` can be used to set a property of type `java.lang.Double` on a plugin.

DOUBLE_PRIM

public static final [ConfigPropertyType](#) DOUBLE_PRIM

`ConfigPropertyType.DOUBLE_PRIM` can be used to set a property of type `double` on a plugin.

STRING_JAVA_LANG

public static final [ConfigPropertyType](#) STRING_JAVA_LANG

`ConfigPropertyType.STRING_JAVA_LANG` can be used to set a property of type `java.lang.String` on a plugin.

Method Detail

valueOf

public static final [ConfigPropertyType](#) `valueOf`(byte id)

Given the raw value of a `ConfigPropertyType`, this method returns a `ConfigPropertyType` object, or null if the raw value does not match an existing type. This method is used to

deserialize this object.

Parameters:

id - the raw value of a ConfigPropertyType

Returns:

the ConfigPropertyType corresponding to the raw input value

Since:

8.6, XC10 2.5

equals

public boolean equals([Object](#) o)

Overrides:

[equals](#) in class [Object](#)

hashCode

public int hashCode()

Overrides:

[hashCode](#) in class [Object](#)

getId

public int getId()

Get the raw value of this ConfigPropertyType. This method is used to serialize this object.

Returns:

the raw value of this ConfigPropertyType.

toString

public [String](#) toString()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Interface ConfigProperty

All Superinterfaces:

[Serializable](#)

```
public interface ConfigProperty  
extends Serializable
```

ConfigProperty can be used to attach properties to a Plugin. A ConfigProperty has the following attributes:

- name: the property name
- value: the property value
- configPropertyType: the configuration property type

This ConfigProperty can be used to set the properties of a plugin. The name of the property should follow JavaBean convention. That is, for every property, there should be a corresponding set method in the plugin class.

Users can use

```
com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory.createConfigProperty(ConfigPropertyType, String, String) to create a ConfigProperty object.
```

```
ConfigProperty evictorNameProp =  
ObjectGridConfigFactory.createConfigProperty(ConfigProperty.STRING_JAVA_LANG, "evictorName",  
"evictor1"); This creates a property "evictorName" with value "evictor1", and the type is  
java.lang.String. Use the  
com.ibm.websphere.objectgrid.config.Plugin#addConfigProperty(ConfigProperty) method to attach a  
ConfigProperty to a Plugin. When the Plugin is created, each ConfigProperty will have its  
corresponding set method called.
```

Continuing with the example above, attach the ConfigProperty to an Evictor Plugin.
evictorPlugin.addConfigProperty(evictorNameProp);

When this Evictor Plugin is created, the setEvictorName(String) method will be called with the value "evictor1".

Since:

WAS XD 6.0.1.2, XC10

Method Summary

C
o
n
f
i
g
P
r
o
p
e
r
t
y

```
getConfigPropertyType\(\)  
Get the ConfigPropertyType of this object.
```

t y p e	
S t r i n g	getName() Get the name of this object.
S t r i n g	getValue() Get the value that is assigned to this ConfigProperty.
V o i d	setConfigPropertyType(ConfigPropertyType configPropType) Set the ConfigPropertyType for this object.
V o i d	setName(String name) Set the name of this object.
V o i d	setValue(String value) Set the value of this ConfigProperty.

Method Detail

setConfigPropertyType

void **setConfigPropertyType**([ConfigPropertyType](#) configPropType)

Set the ConfigPropertyType for this object. The Java primitives, their java.lang counterparts, and java.lang.String are the supported ConfigPropertyTypes.

Parameters:

configPropType -

See Also:

[ConfigPropertyType.INTEGER_JAVA_LANG](#), [ConfigPropertyType.INT_PRIM](#),
[ConfigPropertyType.BOOLEAN_JAVA_LANG](#), [ConfigPropertyType.BOOLEAN_PRIM](#),
[ConfigPropertyType.CHARACTER_JAVA_LANG](#), [ConfigPropertyType.CHAR_PRIM](#),
[ConfigPropertyType.BYTE_JAVA_LANG](#), [ConfigPropertyType.BYTE_PRIM](#),
[ConfigPropertyType.SHORT_JAVA_LANG](#), [ConfigPropertyType.SHORT_PRIM](#),
[ConfigPropertyType.LONG_JAVA_LANG](#), [ConfigPropertyType.LONG_PRIM](#),
[ConfigPropertyType.FLOAT_JAVA_LANG](#), [ConfigPropertyType.FLOAT_PRIM](#),
[ConfigPropertyType.DOUBLE_JAVA_LANG](#), [ConfigPropertyType.DOUBLE_PRIM](#),
[ConfigPropertyType.STRING_JAVA_LANG](#)

getConfigPropertyType

[ConfigPropertyType](#) **getConfigPropertyType**()

Get the ConfigPropertyType of this object.

Returns:

the ConfigPropertyType for this object

setValue

void **setValue**([String](#) value)

Set the value of this ConfigProperty. This String value will be converted to the proper type, based on ConfigPropertyType assigned to this ConfigProperty

Parameters:

value - - will be converted to type and passed to the setter on the plugin

getValue

[String](#) **getValue**()

Get the value that is assigned to this ConfigProperty. This is the value that will be passed to the set method on the plugin.

Returns:

Returns the value.

setName

void **setName**([String](#) name)

Set the name of this object. The Plugin that this ConfigProperty is attached to should have a setter that corresponds to this name. For example, if "size" is passed in as the name, then the Plugin must have a "setSize" method.

Parameters:

name - - name of the property

getName

[String](#) **getName**()

Get the name of this object. The name must have a corresponding set method on the Plugin.

Returns:

Returns the name.

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

Release 2.5 Client API Specification

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

com.ibm.websphere.objectgrid.config

Interface BackingMapConfiguration

public interface **BackingMapConfiguration**

A BackingMapConfiguration object can be used to override BackingMap settings on the client side. The com.ibm.websphere.objectgrid.plugins.Evictor and the com.ibm.websphere.objectgrid.plugins.MapEventListener Plugins can be overridden. Other Evictor related settings can be adjusted as well as client, near-cache specific options.

Use the

com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory.createBackingMapConfiguration(String) method to create a BackingMapConfiguration

Since:

WAS XD 6.0.1.2, XC10

See Also:

[Evictor](#), [MapEventListener](#), [Plugin](#), [ObjectGridConfigFactory](#)

Method Summary

v o i d	<p>addPlugin(Plugin plugin) Add a Plugin to this BackingMapConfiguration.</p>
S t r i n g	<p>getEvictionTriggers() Gets the list of eviction triggers for this BackingMapConfiguration.</p>
c o m . i b m . w e b s p h e r e . o b j e c t	<p>getKeyOutputFormat() Retrieves the data format for all data access APIs that return cache keys.</p>

g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

S
t
r
i
n
g

[getName\(\)](#)
Get the name of this BackingMapConfiguration

i
n
t

[getNumberOfBuckets\(\)](#)
Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

L
i
s
t

[getPlugins\(\)](#)
Get the Plugins that have been attached to this BackingMapConfiguration.

i
n
t

[getTimeToLive\(\)](#)
Gets the "time to live" for each map entry.

I
T
L
E
V
i
c
t
o
r
T
y
p
e

[getTtlEvictorType\(\)](#)
Gets the "time to live" Evictor type for this BackingMapConfiguration.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d

[getValueOutputFormat\(\)](#)
Retrieves the data format for all data access APIs that return cache values.

·
O
u
t
p
u
t
F
o
r
m
a
t

B
o
o
l
e
a
n

[isNearCacheEnabled\(\)](#)

If true, the client local cache is enabled for supported configurations.

B
o
o
l
e
a
n

[isNearCacheInvalidationEnabled\(\)](#)

If true, clients with local caches are automatically invalidated when the data grid map is updated.

B
o
o
l
e
a
n

[isNearCacheLastAccessTTLSyncEnabled\(\)](#)

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

V
o
i
d

[setEvictionTriggers\(String evictionTriggers\)](#)

Sets the eviction triggers for this BackingMapConfiguration.

V
o
i
d

[setKeyOutputFormat\(com.ibm.websphere.objectgrid.OutputFormat dataFormat\)](#)

Sets the data format for all data access APIs that return cache keys.

V
o
i
d

[setNearCacheEnabled\(Boolean nearCacheEnabled\)](#)

Enables or disables the client local cache for supported configurations.

V
o
i
d

[setNearCacheInvalidationEnabled\(Boolean nearCacheInvalidationEnabled\)](#)

Set to true to enable client near cache invalidation.

V
o
i
d

[setNearCacheLastAccessTTLSyncEnabled\(Boolean nearCacheLastAccessTTLSyncEnabled\)](#)

Enables or disables time-to-live access information synchronization to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

V
o
i
d

[setNumberOfBuckets\(int numBuckets\)](#)

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

V
o
i
d

[setPlugins\(List pluginList\)](#)

Set the Plugins for this BackingMapConfiguration.

v o i d	setTimeToLive (int seconds) Sets "time to live" of each BackingMap entry in seconds.
v o i d	setTtlEvictorType (TTLType ttlEvictorType) Set the "time to live" Evictor type for this BackingMapConfiguration.
v o i d	setValueOutputFormat (com.ibm.websphere.objectgrid.OutputFormat dataFormat) Sets the data format for all data access APIs that return cache values.

Method Detail

getName

[String](#) getName()

Get the name of this BackingMapConfiguration

Returns:

The name of this BackingMapConfiguration

addPlugin

void addPlugin([Plugin](#) plugin)

Add a Plugin to this BackingMapConfiguration. The Plugins that can be overridden on a client-side BackingMap are com.ibm.websphere.objectgrid.plugins.Evictor and com.ibm.websphere.objectgrid.plugins.MapEventListener.

Parameters:

plugin -

See Also:

[setPlugins\(List\)](#)

setPlugins

void setPlugins([List](#) pluginList)

Set the Plugins for this BackingMapConfiguration. Any Plugins that were previously attached to this BackingMapConfiguration object will be overridden.

Parameters:

pluginList - - a List of Plugins

See Also:

[addPlugin\(Plugin\)](#)

getPlugins

[List](#) getPlugins()

Get the Plugins that have been attached to this BackingMapConfiguration.

Returns:

getNumberOfBuckets

int `getNumberOfBuckets()`

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Gets the number of buckets defined for this BackingMapConfiguration.

Returns:
the number of buckets defined

setNumberOfBuckets

void `setNumberOfBuckets(int numBuckets)`

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Sets the number of buckets for this BackingMapConfiguration. This will be used by the BackingMap.

The BackingMap implementation uses a hash map for its implementation. If there are a lot of entries in the BackingMap then more buckets means better performance because the risk of collisions is lower as the number of buckets grows. More buckets also means more concurrency.

Parameters:
numBuckets -

See Also:
[BackingMap.setNumberOfBuckets\(int\)](#)

getTimeToLive

int `getTimeToLive()`

Gets the "time to live" for each map entry. The value is in seconds.

Returns:
the "time to live" in seconds

setTimeToLive

void `setTimeToLive(int seconds)`

Sets "time to live" of each BackingMap entry in seconds.

If this method is not called, the lifetime of an entry is forever (or until the application explicitly removes or invalidates the entry, or a user defined Evictor evicts the entry).

Parameters:
seconds -

getTtlEvictorType

[TTLType](#) `getTtlEvictorType()`

Gets the "time to live" Evictor type for this BackingMapConfiguration. If `setTtlEvictorType` was not called, this method will return null and the BackingMap based off this BackingMapConfiguration will use `TTLType.NONE`

Returns:

the "time to live" Evictor type or null if `setTtlEvictorType(TTLType)` was not called

See Also:

[setTimeToLive\(int\)](#)

`setTtlEvictorType`

`void setTtlEvictorType(TTLType ttlEvictorType)`

Set the "time to live" Evictor type for this BackingMapConfiguration. This is used to determine how expiration time of a BackingMap entry is computed.

If this method is not called, `TTLType.NONE` is used to indicate the map entry has no expiration time (e.g. is allowed to live until explicitly removed or invalidated by the application, or evicted by a user defined Evictor).

Parameters:

ttlEvictorType -

See Also:

[BackingMap.setTtlEvictorType\(TTLType\)](#)

`getEvictionTriggers`

`String getEvictionTriggers()`

Gets the list of eviction triggers for this BackingMapConfiguration.

See [BackingMap](#) for a list of valid eviction triggers.

Returns:

a semicolon separated list of eviction triggers or null if `setEvictionTriggers(String)` was not called

Since:

WAS XD 6.1.0.3

`setEvictionTriggers`

`void setEvictionTriggers(String evictionTriggers)`

Sets the eviction triggers for this BackingMapConfiguration. All evictors will use the eviction supplied triggers.

See [BackingMap](#) for a list of valid eviction triggers.

Parameters:

evictionTriggers - a semicolon separated list of eviction triggers

Since:

WAS XD 6.1.0.3

`isNearCacheInvalidationEnabled`

`Boolean isNearCacheInvalidationEnabled()`

If true, clients with local caches are automatically invalidated when the data grid map is updated.

Returns:

true if client near cache invalidation is enabled, false if client near cache invalidation is disabled, or null if the override is not specified.

Since:

8.6, XC10 2.5

setNearCacheInvalidationEnabled

void **setNearCacheInvalidationEnabled**([Boolean](#) nearCacheInvalidationEnabled)

Set to true to enable client near cache invalidation. When enabled, the client will receive events from the remote data grid to invalidate data from the local cache.

Parameters:

nearCacheInvalidationEnabled - If true, the client near cache invalidation is enabled. If false, invalidation is disabled. If null, the override is not specified and the client will use the setting from the remote data grid.

Since:

8.6, XC10 2.5

isNearCacheLastAccessTTLSyncEnabled

[Boolean](#) **isNearCacheLastAccessTTLSyncEnabled**()

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

Returns:

True if last-access time-to-live information is sent to the remote data grid, false if last-access time information is not sent, or null if the override is not specified.

Since:

8.6, XC10 2.5

setNearCacheLastAccessTTLSyncEnabled

void **setNearCacheLastAccessTTLSyncEnabled**([Boolean](#) nearCacheLastAccessTTLSyncEnabled)

Enables or disables time-to-live access information synchronization to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

Parameters:

nearCacheLastAccessTTLSyncEnabled - If true, the last-access time-to-live information is sent to the remote data grid. If false, the last-access information is not sent. If null, the override is not specified and the client will use the setting from the remote data grid.

Since:

8.6, XC10 2.5

isNearCacheEnabled

[Boolean](#) **isNearCacheEnabled**()

If true, the client local cache is enabled for supported configurations. The client near cache can only be enabled when using optimistic locking or when locking is disabled.

Returns:

True if the client near cache is enabled, false if the near cache is disabled, or null if the override is not specified.

Since:

8.6, XC10 2.5

setNearCacheEnabled

void **setNearCacheEnabled**([Boolean](#) nearCacheEnabled)

Enables or disables the client local cache for supported configurations. The client near cache can only be enabled when using optimistic locking or when locking is disabled.

Parameters:

nearCacheEnabled - If true, the client local cache is enabled for supported configurations. If false, the client local cache is disabled. If null, the override is not specified and the client will use the setting from the remote data grid.

getKeyOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getKeyOutputFormat()**

Retrieves the data format for all data access APIs that return cache keys.

Returns:

the data format or null if the default should be used.

Since:

8.6, XC10 2.5

setKeyOutputFormat

void **setKeyOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat dataFormat)

Sets the data format for all data access APIs that return cache keys.

Parameters:

dataFormat - the data format to use or null to use the default.

Since:

8.6, XC10 2.5

getValueOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getValueOutputFormat()**

Retrieves the data format for all data access APIs that return cache values.

Returns:

the data format.

Since:

8.6, XC10 2.5

setValueOutputFormat

void **setValueOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat dataFormat)

Sets the data format for all data access APIs that return cache values.

Parameters:

dataFormat - the data format to use or null to use the default.

Since:

8.6, XC10 2.5

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

PREV CLASS [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#) DETAIL: FIELD | CONSTR | [METHOD](#)

Release 2.5 Client API Specification

Package **com.ibm.websphere.objectgrid.continuousquery**

Interface Summary

ContinuousQueryCache<KeyType, ValueType>	A Continuous Query Cache contains the keys and optionally the values that match a defined continuous query.
ContinuousQueryFilter<KeyType, ValueType, AttributeType, MatchType>	An interface which provides an abstraction to check if an object matches the criteria defined in a filter.
ContinuousQueryListener<KeyType, ValueType>	This interface should be implemented to get a callback for a continuous query.
ContinuousQueryManager	The management interface used to define new continuous queries, retrieve an existing query, or destroy a query.
ContinuousQueryNotificationEvent<KeyType, ValueType>	An interface which defines notification metadata that is sent to continuous query listeners when the result set of a defined continuous query changes.
ContinuousQueryTopic<KeyType, ValueType>	A Continuous Query Topic embodies the client side view of a defined Continuous Query.

Class Summary

ContinuousQueryManagerFactory	A factory class for obtaining the ContinuousQueryManager.
---	---

Enum Summary

ContinuousQueryNotificationEvent.Reason	Indicates the reason a notification event has occurred for this cache entry.
---	--

com.ibm.websphere.objectgrid.continuousquery

Interface ContinuousQueryTopic<KeyType, ValueType>

Type Parameters:

KeyType - Type of the key object for the map being queried
 ValueType - Type of the value object for the map being queried

```
public interface ContinuousQueryTopic<KeyType,ValueType>
```

A Continuous Query Topic embodies the client side view of a defined Continuous Query. The topic gives access to the client-side cached results of the continuous query and allows for management of Continuous Query Listeners for this topic.

Since:

8.6, XC10 2.5

Method Summary	
V o i d	<p>addListener(ContinuousQueryListener<KeyType,ValueType> listener) Add a Continuous Query Listener to this topic.</p>
C o l l e c t i o n < C o n t i n u o u s Q u e r y T o p i c >	<p>getAllListeners() Returns a shallow copy of the current set of listeners registered for this query topic.</p>

Value
Type
,
Value
Type
,
Value
Type
,
>

Continuous
Query
Cache
Object
Methods
<
Key
Value
Type
,
Value
Type
,
>

[getCache\(\)](#)
Return a continuous query cache object with methods to access the keys and values stored in the cache.

Static
Method
<

[getName\(\)](#)
Returns the generated unique name for this continuous query.

com.ibm.websphere

e . o b j e c t g r i d . O u t p u t F o r m a t	<p>getOutputFormat() Return the format this query will use for keys and values.</p>
L i s t < I n t e r >	<p>getPartitions() Return the list of partitions this query is defined against.</p>
b o o l e a n	<p>isKeysOnlyCache() Return true if the cache for this continuous query contains only keys.</p>
b o o l e a n	<p>noCache() Return true if the no caching option is enabled for this Continuous Query.</p>
v o i d	<p>removeAllListeners() Removes all registered listeners from this query topic.</p>
b o o l e a n	<p>removeListener(ContinuousQueryListener<KeyType,ValueType> listener) Remove the listener from this continuous query's the set of listeners.</p>

Method Detail

addListener

```
void addListener(ContinuousQueryListener<KeyTpe,ValueType> listener)
```

Add a Continuous Query Listener to this topic. The listener will be called when this topic receives updates to the query. There is no guarantee of the order that listeners will be called. Multiple adds of the same listener will result in only one call to the listener.

Parameters:

listener - the listener to add to this continuous query's set of listeners.

removeListener

```
boolean removeListener(ContinuousQueryListener<KeyTpe,ValueType> listener)
```

Remove the listener from this continuous query's the set of listeners. In flight notifications to this listener may continue after this method returns.

Parameters:

listener - - the continuous query listener to remove.

Returns:

true if the listener was removed.

getCache

```
ContinuousQueryCache<KeyTpe,ValueType> getCache()
```

Return a continuous query cache object with methods to access the keys and values stored in the cache.

Returns:

a Continuous Query Cache object for this topic.

getName

```
String getName()
```

Returns the generated unique name for this continuous query.

Returns:

a unique string.

isKeysOnlyCache

```
boolean isKeysOnlyCache()
```

Return true if the cache for this continuous query contains only keys. Return false if it also contains the value associated with the key.

Returns:

true if the cache does not contain values.

noCache

```
boolean noCache()
```

Return true if the no caching option is enabled for this Continuous Query.

Returns:

true if no keys or values are being cached.

getOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getOutputFormat()**

Return the format this query will use for keys and values.

Returns:

the OutputFormat specified when this query was defined.

See Also:

OutputFormat

getAllListeners

[Collection](#)<[ContinuousQueryListener](#)<[KeyType](#),[ValueType](#)>> **getAllListeners()**

Returns a shallow copy of the current set of listeners registered for this query topic.

Returns:

a new collection object containing all currently registered listeners.

removeAllListeners

void **removeAllListeners()**

Removes all registered listeners from this query topic. In flight notifications to listeners may continue after this method returns.

getPartitions

[List](#)<[Integer](#)> **getPartitions()**

Return the list of partitions this query is defined against. If the return value is null or an empty list, this query is defined against all partitions of the map.

Returns:

the array of partitions

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.continuousquery

Interface

ContinuousQueryNotificationEvent<KeyType,ValueType>

Type Parameters:

KeyType - Type of the key object in this event, from the key type used in the map being queried

ValueType - Type of the value object in this event, from the value type used in the map being queried

```
public interface ContinuousQueryNotificationEvent<KeyType,ValueType>
```

An interface which defines notification metadata that is sent to continuous query listeners when the result set of a defined continuous query changes.

Since:

8.6, XC10 2.5

Nested Class Summary

S
t
a
t
i
c
c
l
a
s
s

[ContinuousQueryNotificationEvent.Reason](#)

Indicates the reason a notification event has occurred for this cache entry.

Method Summary

K
e
y
T
y
p
e

[getKey\(\)](#)

Retrieve the key of the cache entry that triggered this notification.

C
o
n
t
i
n
u
o
u
s
Q
u
e
r
y

er
y
N
o
t
i
f
i
c
a
t
i
o
n
E
v
e
n
t
R
e
a
s
o
n

[getReason\(\)](#)

Retrieve the reason for this notification event

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
S
e
s
s
i
o
n
H
a
n
d
l
e

[getSessionHandle\(\)](#)

Retrieve a SessionHandle object which can be used to retrieve the cache entry that triggered this notification.

V
a
l
u
e

[getValue\(\)](#)

Retrieve the value of the cache entry that triggered this notification, if the query was

[y](#) configured to return the value object for matches.

[b](#)
[o](#)
[o](#) [isKeysOnly\(\)](#)
[l](#) If true, this notification only includes the key of the matching cache entry.
[e](#)
[a](#)
[n](#)

Method Detail

getKey

[KeyType](#) `getKey()`

Retrieve the key of the cache entry that triggered this notification. This object is a reference to the object stored in the [ContinuousQueryCache](#), if the cache has been enabled. Therefore it should not be modified.

Returns:
the cache entry key

getValue

[ValueType](#) `getValue()`

Retrieve the value of the cache entry that triggered this notification, if the query was configured to return the value object for matches. This object is a reference to the object stored in the [ContinuousQueryCache](#), if the cache has been enabled. Therefore it should not be modified.

Returns:
the cache entry value

getSessionHandle

`com.ibm.websphere.objectgrid.SessionHandle` `getSessionHandle()`

Retrieve a `SessionHandle` object which can be used to retrieve the cache entry that triggered this notification. This method can only be invoked when using a `PER_CONTAINER` placement strategy for the map.

Returns:
the `SessionHandle` for the partition that owns the cache entry.

See Also:
`MapSet.getPlacementStrategy()`, [Session.setSessionHandle\(SessionHandle\)](#)

getReason

[ContinuousQueryNotificationEvent.Reason](#) `getReason()`

Retrieve the reason for this notification event

Returns:

the Reason enum

isKeysOnly

boolean **isKeysOnly**()

If true, this notification only includes the key of the matching cache entry. If false, this notification also includes the value

Returns:

whether or not this notification includes the cache value.

See Also:

[getValue\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help
--------------------------	-------------------------	-------------------------	----------------------------	----------------------------	-----------------------	----------------------

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**


© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Enum ContinuousQueryNotificationEvent.Reason

[java.lang.Object](#)

 [java.lang.Enum](#)<[ContinuousQueryNotificationEvent.Reason](#)>

 [com.ibm.websphere.objectgrid.continuousquery.ContinuousQueryNotificationEvent.Reason](#)

All Implemented Interfaces:

[Serializable](#), [Comparable](#)<[ContinuousQueryNotificationEvent.Reason](#)>

Enclosing interface:

[ContinuousQueryNotificationEvent](#)<[KeyType](#), [ValueType](#)>

```
public static enum ContinuousQueryNotificationEvent.Reason
extends Enum<ContinuousQueryNotificationEvent.Reason>
```

Indicates the reason a notification event has occurred for this cache entry.

Enum Constant Summary

[ADDED](#)

The cache entry was inserted into the grid and matches the query, or an entry which did not match the query was updated such that it now matches the query.

[CLEAR](#)

A cache entry which matched the query was removed from the grid due to a map clear operation.

[REMOVED](#)

The cache entry, which previously matched the query, was removed from the grid or updated such that it no longer matches.

[UPDATED](#)

The cache entry, which previously matched the query, was updated and still matches the query.

Method Summary

s
 t
 a
 t
 i
 c
 C
 o
 n
 t
 i
 n
 u
 o
 u
 s

Q
u
e
r
y
N
o
t
i
f
i
c
a
t
i
o
n
D
e
f
i
n
e
D
e
s
c
r
i
b
e

[valueOf](#)([String](#) name)

Returns the enum constant of this type with the specified name.

s
t
a
t
i
c
C
o
n
s
t
a
n
t
D
e
f
i
n
e
D
e
s
c
r
i
b
e
D
e
f
i
n
e
D
e
s
c
r
i
b
e

[values](#)()

Returns an array containing the constants of this enum type, in the order they are declared.

[
]

Methods inherited from class java.lang.[Enum](#)

[clone](#), [compareTo](#), [equals](#), [finalize](#), [getDeclaringClass](#), [hashCode](#), [name](#), [ordinal](#), [toString](#), [valueOf](#)

Methods inherited from class java.lang.[Object](#)

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Enum Constant Detail

ADDED

public static final [ContinuousQueryNotificationEvent.Reason](#) **ADDED**

The cache entry was inserted into the grid and matches the query, or an entry which did not match the query was updated such that it now matches the query.

REMOVED

public static final [ContinuousQueryNotificationEvent.Reason](#) **REMOVED**

The cache entry, which previously matched the query, was removed from the grid or updated such that it no longer matches.

UPDATED

public static final [ContinuousQueryNotificationEvent.Reason](#) **UPDATED**

The cache entry, which previously matched the query, was updated and still matches the query. Note: The result of `getValue()` for REMOVE events will always be null, even if the query was defined to include values.

See Also:

[ContinuousQueryManager.defineContinuousQuery\(java.lang.String, com.ibm.websphere.objectgrid.continuousquery.ContinuousQueryFilter, boolean, boolean, boolean, java.util.Collection>, boolean, com.ibm.websphere.objectgrid.OutputFormat, java.util.List\)](#)

CLEAR

public static final [ContinuousQueryNotificationEvent.Reason](#) **CLEAR**

A cache entry which matched the query was removed from the grid due to a map clear operation.

Method Detail

values

public static [ContinuousQueryNotificationEvent.Reason](#)[] **values()**

Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows:

```
for (ContinuousQueryNotificationEvent.Reason c : ContinuousQueryNotificationEvent.Reason.values())
    System.out.println(c);
```

Returns:

an array containing the constants of this enum type, in the order they are declared

valueOf

```
public static ContinuousQueryNotificationEvent.Reason valueOf(String name)
```

Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

Parameters:

name - the name of the enum constant to be returned.

Returns:

the enum constant with the specified name

Throws:

[IllegalArgumentException](#) - if this enum type has no constant with the specified name

[NullPointerException](#) - if the argument is null

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All			
SUMMARY: NESTED ENUM			DETAIL: ENUM					
CONSTANTS FIELD METHOD			CONSTANTS FIELD METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Class ContinuousQueryManagerFactory

[java.lang.Object](#)



```
public final class ContinuousQueryManagerFactory  
extends Object
```

A factory class for obtaining the ContinuousQueryManager.

Since:

8.6, XC10 2.5

Constructor Summary

[ContinuousQueryManagerFactory\(\)](#)

Method Summary

s
t
a
t
i
c
C
o
n
t
i
n
u
o
u
s
Q
u
e
r
y
M
a
n
a
g
e
r
F
a
c
t
o
r
y

[getManager\(ObjectGrid objGrid\)](#)

Factory that returns the single instance of a ContinuousQueryManager for the given object grid.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ContinuousQueryManagerFactory

```
public ContinuousQueryManagerFactory()
```

Method Detail

getManager

```
public static ContinuousQueryManager getManager(ObjectGrid objGrid)
```

Factory that returns the single instance of a ContinuousQueryManager for the given object grid.

Parameters:

objGrid - - the ObjectGrid this Continuous Query Manager will define continuous queries for.

Returns:

the ContinuousQueryManager instance.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#) [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.continuousquery

Interface ContinuousQueryManager

public interface **ContinuousQueryManager**

The management interface used to define new continuous queries, retrieve an existing query, or destroy a query. An instance can be retrieved via

[ContinuousQueryManagerFactory.getManager\(com.ibm.websphere.objectgrid.ObjectGrid\)](#)

Since:

8.6, XC10 2.5

Method Summary	
<code><KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType></code>	defineContinuousQuery (String mapName, ContinuousQueryFilter filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated) Define a new continuous query.
<code><KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType></code>	defineContinuousQuery (String mapName, ContinuousQueryFilter filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated, Collection < ContinuousQueryListener <KeyType,ValueType>> continuousQueryListeners, boolean noCache, com.ibm.websphere.objectgrid.OutputFormat format, List < Integer > partitionSubset) Define a new continuous query.
<code><KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType></code>	defineContinuousQuery (String mapName, ContinuousQueryFilter filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated, List < Integer > partitionSubset) Define a new continuous query.
<code>List<ContinuousQueryTopic<?,?>></code>	getDefinedContinuousQueries () Returns a list view of the currently defined Continuous Query Topics.
<code>boolean</code>	removeContinuousQuery (ContinuousQueryTopic <?,?> topic) Remove a defined continuous query.

Method Detail

defineContinuousQuery

`<KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType> defineContinuousQuery(String mapName, ContinuousQueryFilter filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated)`

Updated,

```
ContinuousQueryListener<KeyType,ValueType>> continuousQueryListeners,  
e.objectgrid.OutputFormat format,  
rtitionSubset)  
throws ContinuousQueryIncompatibleDuplicateException,  
UndefinedMapException
```

Define a new continuous query. This query will run in the grid container processes. When cache entries that match the filter criteria are inserted or deleted, the client's [ContinuousQueryCache](#) will be updated and any [ContinuousQueryListener](#) implementations will be invoked. This will also occur if a cache entry is updated such that it now matches, or no longer matches, the filter criteria. Optionally, notifications can be sent for update operations to cache entries that previously matched the criteria, and still match the criteria after the update.

Parameters:

`mapName` - - the name of the map the continuous query will be defined on.
`filter` - - the filter that will determine the contents of the continuous query. Custom filter logic can be implemented by extending [AbstractCQFilter](#).
`keysOnly` - - true if the query should only return the keys of items that satisfy the query; false to also send the values as well as the keys. If false, the [ContinuousQueryCache](#) and [ContinuousQueryNotificationEvent](#) objects will include the value associated with cache entries that match the query.
`returnInitialResultSet` - - true to run the filter on all existing matches in the map at the time of the query definition and return the items to the cache. If false, the client will only be notified of matches which occur after the query is defined. Returning the initial result set will have a high computational requirement for large maps.
`notifyOnUpdated` - - true causes this topic to get called every time an existing item that satisfies this query gets updated and continues to satisfy this query. This will drive [ContinuousQueryListener](#) call backs as well as updated values in the query cache if `keysOnly` is false.
`continuousQueryListeners` - - used to register listeners before the query is defined. These listeners will be guaranteed to be invoked for any entries in the initial result set, as well as changes to the result set which might occur before [ContinuousQueryTopic.addListener\(ContinuousQueryListener\)](#) can be invoked. Can be null.
`noCache` - - true to indicate that no caching of keys or values should occur on the client. Continuous Query listeners will still be notified.
`format` - - if `OutputFormat.RAW` is specified, keys and values returned from the [ContinuousQueryCache](#) and [ContinuousQueryListener](#) will be in `SerializedEntry` format when a `DataSerializer` is defined on the map. - if `OutputFormat.NATIVE` is specified, or when no `DataSerializer` is defined, keys and values will be returned as `Objects`.
`partitionSubset` - - Used to indicate that this continuous query should only be defined on the specified subset of partitions. Passing null or an empty list indicates this continuous query should be defined on all existing and future partitions containing this map. An `IllegalArgumentException` may be thrown if an invalid partition ID is passed.

Returns:

The [ContinuousQueryTopic](#) for this continuous query. If the call to this method would result in an topic which is identical to an existing topic being created, the existing instance is returned.

Throws:

[ContinuousQueryIncompatibleDuplicateException](#)
[UndefinedMapException](#)
[IllegalArgumentException](#)

defineContinuousQuery

```

<KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType> defineContinuousQuery(String mapName,
ContinuousQueryF
ilter filter,
,
boolean keysOnly
boolean returnIn
itialResultSet,
boolean notifyOn
Updated,
List<Integer> pa
rtitionSubset)
throws ContinuousQueryIncompatibleDu
plicateException,
UndefinedMapException

```

Define a new continuous query. This query will run in the grid container processes. When cache entries that match the filter criteria are inserted or deleted, the client's [ContinuousQueryCache](#) will be updated and any [ContinuousQueryListener](#) implementations will be invoked. This will also occur if a cache entry is updated such that it now matches, or no longer matches, the filter criteria. Optionally, notifications can be sent for update operations to cache entries that previously matched the criteria, and still match the criteria after the update.

Parameters:

mapName - - the name of the map the continuous query will be defined on.
filter - - the filter that will determine the contents of the continuous query. Custom filter logic can be implemented by extending [AbstractCQFilter](#).
keysOnly - - true if the query should only return the keys of items that satisfy the query; false to also send the values as well as the keys. If false, the [ContinuousQueryCache](#) and [ContinuousQueryNotificationEvent](#) objects will include the value associated with cache entries that match the query.
returnInitialResultSet - - true to run the filter on all existing matches in the map at the time of the query definition and return the items to the cache. If false, the client will only be notified of matches which occur after the query is defined. Returning the initial result set will have a high computational requirement for large maps.
notifyOnUpdated - - true causes this topic to get called every time an existing item that satisfies this query gets updated and continues to satisfy this query. This will drive [ContinuousQueryListener](#) call backs as well as updated values in the query cache if keysOnly is false.
partitionSubset - - Used to indicate that this continuous query should only be defined on the specified subset of partitions. Passing null or an empty list indicates this continuous query should be defined on all existing and future partitions containing this map. An [IllegalArgumentException](#) may be thrown if an invalid partition ID is passed.

Returns:

The [ContinuousQueryTopic](#) for this continuous query. If the call to this method would result in a topic which is identical to an existing topic being created, the existing instance is returned.

Throws:

[ContinuousQueryIncompatibleDuplicateException](#)
[UndefinedMapException](#)
[IllegalArgumentException](#)

defineContinuousQuery

```

<KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType> defineContinuousQuery(String mapName,
ContinuousQueryF
ilter filter,
,
boolean keysOnly
boolean returnIn
itialResultSet,
boolean notifyOn

```


Updated)

[uplicateException](#),

throws [ContinuousQueryIncompatibleDu](#)

[UndefinedMapException](#)

Define a new continuous query. This query will run in the grid container processes. When cache entries that match the filter criteria are inserted or deleted, the client's [ContinuousQueryCache](#) will be updated and any [ContinuousQueryListener](#) implementations will be invoked. This will also occur if a cache entry is updated such that it now matches, or no longer matches, the filter criteria. Optionally, notifications can be sent for update operations to cache entries that previously matched the criteria, and still match the criteria after the update. This query will be applied to all partitions.

Parameters:

`mapName` - - the name of the map the continuous query will be defined on.
`filter` - - the filter that will determine the contents of the continuous query. Custom filter logic can be implemented by extending [AbstractCQFilter](#).
`keysOnly` - - true if the query should only return the keys of items that satisfy the query; false to also send the values as well as the keys. If false, the [ContinuousQueryCache](#) and [ContinuousQueryNotificationEvent](#) objects will include the value associated with cache entries that match the query.
`returnInitialResultSet` - - true to run the filter on all existing matches in the map at the time of the query definition and return the items to the cache. If false, the client will only be notified of matches which occur after the query is defined. Returning the initial result set will have a high computational requirement for large maps.
`notifyOnUpdated` - - true causes this topic to get called every time an existing item that satisfies this query gets updated and continues to satisfy this query. This will drive [ContinuousQueryListener](#) call backs as well as updated values in the query cache if `keysOnly` is false.

Returns:

The [ContinuousQueryTopic](#) for this continuous query. If the call to this method would result in a topic which is identical to an existing topic being created, the existing instance is returned.

Throws:

[ContinuousQueryIncompatibleDuplicateException](#)
[UndefinedMapException](#)
[IllegalArgumentException](#)

removeContinuousQuery

`boolean removeContinuousQuery(ContinuousQueryTopic<?,?> topic)`

Remove a defined continuous query. Does the opposite of [defineContinuousQuery\(String, ContinuousQueryFilter, boolean, boolean, boolean\)](#). If the number of calls to this method for a given topic equals the number of calls to [defineContinuousQuery](#) that return the specified topic, the topic will cease receiving updates from the server, all listeners of this topic will be removed and the reference to this topic will be invalid.

Parameters:

`topic` - - the topic to remove

Returns:

true if the topic was removed, false otherwise

Throws:

[IllegalArgumentException](#)

getDefinedContinuousQueries

`List<ContinuousQueryTopic<?,?>> getDefinedContinuousQueries()`

Returns a list view of the currently defined Continuous Query Topics.

Returns:
a list of Continuous Query Topics

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Interface

ContinuousQueryListener<KeyType,ValueType>

Type Parameters:

KeyType - Type of the key object for the map being queried

ValueType - Type of the value object for the map being queried

```
public interface ContinuousQueryListener<KeyType,ValueType>
```

This interface should be implemented to get a callback for a continuous query. When a Continuous Query listener is added to a Continuous Query Topic, that listener will be called after every change to the current set of results matching the continuous query.

Since:

8.6, XC10 2.5

Method Summary

v o i d	cacheUpdated (ContinuousQueryNotificationEvent <KeyType,ValueType> event) This method will be called by every ContinuousQueryTopic that this listener is added to.
------------------	--

Method Detail

cacheUpdated

```
void cacheUpdated(ContinuousQueryNotificationEvent<KeyType,ValueType> event)
```

This method will be called by every ContinuousQueryTopic that this listener is added to. The caller will capture any [Throwable](#)s thrown from calls to this method and create an FFDC entry. This [ContinuousQueryListener](#) will not be called again for the same [ContinuousQueryNotificationEvent](#). Additionally, the performance of this [ContinuousQueryListener](#) may make an impact on the speed with which other [ContinuousQueryListeners](#) are called.

Parameters:

event - - [ContinuousQueryNotificationEvent](#) containing information about the change in the query result set

See Also:

[ContinuousQueryTopic.addListener\(ContinuousQueryListener\)](#),
[ContinuousQueryNotificationEvent](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Interface

ContinuousQueryFilter<KeyType,ValueType,AttributeType,MatchType>

Type Parameters:

- KeyType - Type of the key object for the map being queried
- ValueType - Type of the value object for the map being queried
- AttributeType - Type of the attribute referenced by the attribute path
- MatchType - Type of the object being compared to

All Superinterfaces:

[Serializable](#)

All Known Implementing Classes:

[AbstractCQFilter](#), [AndFilter](#), [BinaryLogicalFilter](#), [CompareFilter](#), [EQFilter](#), [FalseFilter](#), [GTEFilter](#), [GTFilter](#), [IsNotNullFilter](#), [IsNullFilter](#), [LTEFilter](#), [LTFilter](#), [MatchFilter](#), [NEQFilter](#), [NotFilter](#), [NotMatchFilter](#), [OrFilter](#), [TrueFilter](#)

```
public interface ContinuousQueryFilter<KeyType,ValueType,AttributeType,MatchType>  
extends Serializable
```

An interface which provides an abstraction to check if an object matches the criteria defined in a filter. Filters are invoked within the eXtreme Scale containers hosting partitions for the map on which the query is defined. All implementations must extend [AbstractCQFilter](#).

Since:

8.6, XC10 2.5

Field Summary	
s t a t i c S t r i n g	POJO_ADDRESSABLEKEYNAME The name used to identify the key objects in POJO maps.
s t a t i c S t r i n g	POJO_PATHSEPARATOR The path separator to use when identifying attribute paths for POJO maps.

Method Summary

`boolean filter(FilterContent<KeyType,ValueType> content)`
Checks if the supplied object passes the filter.

Field Detail

POJO_ADDRESSABLEKEYNAME

static final [String](#) POJO_ADDRESSABLEKEYNAME

The name used to identify the key objects in POJO maps. Not applicable to maps using XDF or a custom MapSerializedPlugin.

See Also:

[CompareFilter.CompareFilter\(String, Object\)](#), [Constant Field Values](#)

POJO_PATHSEPARATOR

static final [String](#) POJO_PATHSEPARATOR

The path separator to use when identifying attribute paths for POJO maps. Not applicable to maps using XDF or a custom MapSerializedPlugin.

See Also:

[CompareFilter.CompareFilter\(String, Object\)](#), [Constant Field Values](#)

Method Detail

filter

boolean **filter**([FilterContent](#)<[KeyType](#),[ValueType](#)> content)
throws [ContinuousQueryException](#)

Checks if the supplied object passes the filter.

Parameters:

content - A representation of the cache entry to be checked

Returns:

true if the object matches the filtering criteria, false otherwise

Throws:

[ContinuousQueryException](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.continuousquery

Interface ContinuousQueryCache<KeyType,ValueType>

Type Parameters:

KeyType - Type of the key object for the map being queried

ValueType - Type of the value object for the map being queried

```
public interface ContinuousQueryCache<KeyType,ValueType>
```

A Continuous Query Cache contains the keys and optionally the values that match a defined continuous query. The contents of this cache arrive asynchronously from the grid for which the query is defined. If the query is defined such that only the keys are stored in the continuous query cache then all operations that return a value will return null.

Since:

8.6, XC10 2.5

Method Summary	
b o o l e a n	<p>containsKey(KeyType key) Returns true if the cache contains the given key.</p>
b o o l e a n	<p>containsValue(ValueType value) Returns true if the cache contains the given value.</p>
V a l u e T y p e	<p>get(KeyType key) Returns the value for the given key.</p>
L i s t < c o m . i b m .	

w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
S
e
s
s
i
o
n
H
a
n
d
l
e
>

[getSessionHandles](#)([KeyType](#) key)

Returns SessionHandle objects that can be used to retrieve all query matches associated with this key.

b
o
o
l
e
a
n

[includesValues](#)()

Returns true if this cache includes the values associated with the keys that have matched the query.

S
e
<
K
e
y
T
y
p
e
>

[keySnapshot](#)()

Returns a Set view of a snapshot of the keys contained in the cache.

i
n
t

[size](#)()

Return the number of keys in the cache.

Method Detail

keySnapshot

[Set](#)<[KeyType](#)> [keySnapshot](#)()

Returns a Set view of a snapshot of the keys contained in the cache. The objects in the set are references to the keys in the query cache, therefore they should not be modified.

Returns:

a set view of the keys in the cache.

size

int **size**()

Return the number of keys in the cache.

Returns:

the number of keys in the cache.

get

[ValueType](#) **get**([KeyType](#) key)

Returns the value for the given key. The object returned is a reference to the value in the query cache, therefore it should not be modified.

Parameters:

key - - the key whose associated value is to be returned

Returns:

the value in the cache for the specified key or null if either the cache does not contain this key or if the cache is configured to not contain values.

getSessionHandles

[List](#)<[com.ibm.websphere.objectgrid.SessionHandle](#)> **getSessionHandles**([KeyType](#) key)

Returns [SessionHandle](#) objects that can be used to retrieve all query matches associated with this key. This method can only be invoked when using a PER_CONTAINER placement strategy for the map.

Parameters:

key - - the key for which [SessionHandles](#) will be returned

Returns:

a List of [SessionHandles](#), one for each instance of this key in the grid.

See Also:

[MapSet.getPlacementStrategy\(\)](#)

containsKey

boolean **containsKey**([KeyType](#) key)

Returns true if the cache contains the given key.

Parameters:

key - - key to check for

Returns:

true if the specified key is in the cache at the time of the request.

containsValue

boolean **containsValue**([ValueType](#) value)

Returns true if the cache contains the given value.

Parameters:

value - - value to check for

Returns:

true if the specified object is in the cache at the time of the request.

includesValues

boolean **includesValues()**

Returns true if this cache includes the values associated with the keys that have matched the query. If true, [containsValue\(Object\)](#) and [get\(Object\)](#) will return usable values, otherwise they will return null.

Returns:

true if this query cache contains values in addition to keys.

See Also:

[ContinuousQueryManager.defineContinuousQuery\(java.lang.String, com.ibm.websphere.objectgrid.continuousquery.ContinuousQueryFilter, boolean, boolean, boolean, java.util.Collection>, boolean, com.ibm.websphere.objectgrid.OutputFormat, java.util.List\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help
--------------------------	-------------------------	-------------------------	----------------------------	----------------------------	-----------------------	----------------------

PREV CLASS [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#) | [OD](#) | [DETAIL](#): FIELD | CONSTR | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

Package **com.ibm.websphere.objectgrid.management**

This package contains the interfaces for all ObjectGrid MBeans.

See:

[Description](#)

Interface Summary	
AgentManagerMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific Agent on a server process.
CatalogServiceManagementMBean	This MBean interface allows user to manipulate the behaviors of heartbeat and leader manager and other catalog service specific actions.
ContainerMBean	This MBean interface allows a client process to perform operations on and get status from an ObjectGrid container running in a dynamic environment.
DynamicServerMBean	This MBean interface allows a client process to access different attributes about a specific server process in a dynamic environment.
HashIndexMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific HashIndex on a server process.
MapMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific map on a server process.
ObjectGridMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific ObjectGrid on a server process.
PlacementMediationServiceMBean	This MBean interface allows a client process to perform operations on and get status from the PlacementMediationService running in a dynamic environment.
PlacementServiceMBean	This MBean interface allows a client process to perform operations on and get status from the PlacementService running in a dynamic environment.
QueryManagerMBean	This MBean interface allows a client process to perform operations on and get status from an ObjectGrid Query Manager running in a dynamic environment.
QuorumManagerMBean	Each catalog service has a QuorumManager and an associated MBean.
ServerMBean	This MBean interface allows a client process to access different attributes about a specific server process.
SessionMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific session.
ShardMBean	This MBean interface allows a client process to perform operations on and get status from a shard running in a dynamic environment.
ThreadPoolMBean	This MBean interface allows user to access the thread pool properties.

Package com.ibm.websphere.objectgrid.management Description

This package contains the interfaces for all ObjectGrid MBeans.

Overview

Each MBean interface has several methods to administer and monitor ObjectGrid services and components.

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.management

Interface ThreadPoolMBean

public interface **ThreadPoolMBean**

This MBean interface allows user to access the thread pool properties. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ThreadPool
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

7.0.0.0 FIX2, XC10

Method Summary	
i n t	getActiveThreadCount() Retrieves the approximate number of active threads in the pool.
i n t	getMaximumSize() Retrieves the maximum number of threads in the pool.
i n t	getMinimumSize() Retrieves the minimum number of threads in the pool.
S t r i n g	getName() Retrieves the name of the ThreadPool.
V o i d	setMaximumSize(int size) Sets the maximum thread pool size.
V o i d	setMinimumSize(int size) Sets the minimum thread pool size.

Method Detail

setMaximumSize

void **setMaximumSize**(int size)

Sets the maximum thread pool size.

Parameters:

size - the maximum number of threads.

getMaximumSize

int **getMaximumSize**()

Retrieves the maximum number of threads in the pool.

Returns:

the maximum number of threads in the pool

setMinimumSize

void **setMinimumSize**(int size)

Sets the minimum thread pool size.

Parameters:

size - the minimum number of threads.

getMinimumSize

int **getMinimumSize**()

Retrieves the minimum number of threads in the pool.

Returns:

the minimum number of threads in the pool

getActiveThreadCount

int **getActiveThreadCount**()

Retrieves the approximate number of active threads in the pool.

Returns:

the number of active threads in the pool in use

getName

[String](#) **getName**()

Retrieves the name of the ThreadPool.

Returns:

the name of the ThreadPool

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management

Interface ShardMBean

public interface **ShardMBean**

This MBean interface allows a client process to perform operations on and get status from a shard running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=Shard,name=<objectgrid>,objectgrid=<objectgrid>,mapset=<mapset>
,partition=<partition id>,container=<container>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary	
s t a t i c S t r i n g	<p>ROLE_SWAP_REQUESTED_WITH_SAME_TYPE Indicates that this shard is the same type of shard as the requested swap type.</p>
s t a t i c S t r i n g	<p>ROLE_SWAP_SUCCESSFUL Indicates that the role swap was executed successfully.</p>
s t a t i c S t r i n g	<p>ROLE_SWAP_TIMEOUT Indicates that this shard has timed out waiting to inherit its requested role</p>
s t	

a t t r i b u t e	<p><u>TYPE_INACTIVE</u> Indicates the shard type is the inactive role.</p>
s t a t i c	<p><u>TYPE_PRIMARY</u> Indicates the shard type is the primary role.</p>
s t a t i c	<p><u>TYPE_REPLICA_ASYNCHRONOUS</u> Indicates the shard type is the asynchronous replica role.</p>
s t a t i c	<p><u>TYPE_REPLICA_SYNCHRONOUS</u> Indicates the shard type is the synchronous replica role.</p>

Method Summary

l o n g	<p><u>getActiveRequestCount()</u> Retrieves the number of requests currently being processed by this shard.</p>
S t r i n g	<p><u>getContainerName()</u> Retrieves the name of the container that is hosting this shard.</p>
S t r i n g	<p><u>getDomainName()</u> Retrieve the name of the catalog server grouping administering this shard.</p>
l o n g	<p><u>getForwardedRequestCount()</u> Retrieves the number of requests that this shard has forwarded since its inception.</p>

S t r i n g	<p>getMapSetName() Retrieve the name of the MapSet in which the shard resides.</p>
S t r i n g	<p>getObjectGridName() Retrieve the name of the ObjectGrid in which the shard resides.</p>
S t r i n g	<p>getPartitionName() Retrieve the name of the partition in which the shard resides.</p>
l o n g	<p>getProcessedRequestCount() Retrieves the number of requests that this shard has processed since its inception.</p>
S t r i n g	<p>getState() Retrieve the state of the shard.</p>
l o n g	<p>getTotalRequestCount() Retrieves the number of requests that this shard has processed or forwarded since its inception.</p>
S t r i n g	<p>getType() Retrieve the type of the shard.</p>
S t r i n g	<p>swapWithPrimary() Causes this shard to swap roles with the primary shard for the partition.</p>

Field Detail

ROLE_SWAP_SUCCESSFUL

static final [String](#) ROLE_SWAP_SUCCESSFUL

Indicates that the role swap was executed successfully.

Since:

7.1.0.0 FIX1

See Also:

[swapWithPrimary\(\)](#), [Constant Field Values](#)

ROLE_SWAP_REQUESTED_WITH_SAME_TYPE

static final [String](#) ROLE_SWAP_REQUESTED_WITH_SAME_TYPE

Indicates that this shard is the same type of shard as the requested swap type. No swap will be executed.

Since:

7.1.0.0 FIX1

See Also:

[swapWithPrimary\(\)](#), [Constant Field Values](#)

ROLE_SWAP_TIMEOUT

static final [String](#) ROLE_SWAP_TIMEOUT

Indicates that this shard has timed out waiting to inherit its requested role

Since:

7.1.0.0 FIX1

See Also:

[swapWithPrimary\(\)](#), [Constant Field Values](#)

TYPE_PRIMARY

static final [String](#) TYPE_PRIMARY

Indicates the shard type is the primary role. This means that this is the shard that handles all updates and coordinates state transitions with the replicas.

See Also:

[Constant Field Values](#)

TYPE_REPLICA_SYNCHRONOUS

static final [String](#) TYPE_REPLICA_SYNCHRONOUS

Indicates the shard type is the synchronous replica role. This means that this shard is receiving state updates from another shard that is acting as primary.

See Also:

[Constant Field Values](#)

TYPE_REPLICA_ASYNCHRONOUS

static final [String](#) TYPE_REPLICA_ASYNCHRONOUS

Indicates the shard type is the asynchronous replica role. This means that this shard is receiving state updates from another shard that is acting as primary.

See Also:

[Constant Field Values](#)

TYPE_INACTIVE

static final [String](#) TYPE_INACTIVE

Indicates the shard type is the inactive role. This means that this shard is not actively enrolled in the partition.

See Also:

[Constant Field Values](#)

Method Detail

getObjectGridName

[String](#) getObjectGridName()

Retrieve the name of the ObjectGrid in which the shard resides.

Returns:

The ObjectGrid name.

getMapSetName

[String](#) getMapSetName()

Retrieve the name of the MapSet in which the shard resides.

Returns:

The MapSet name.

getPartitionName

[String](#) getPartitionName()

Retrieve the name of the partition in which the shard resides.

Returns:

The partition name.

getType

[String](#) getType()

Retrieve the type of the shard.

Returns:

The shard type.

getDomainName

[String](#) getDomainName()

Retrieve the name of the catalog server grouping administering this shard.

Returns:

The domain name.

getState

[String](#) getState()

Retrieve the state of the shard.

Returns:

The shard state.

getTotalRequestCount

long `getTotalRequestCount()`

Retrieves the number of requests that this shard has processed or forwarded since its inception.

Returns:

A count of the total number of requests.

getActiveRequestCount

long `getActiveRequestCount()`

Retrieves the number of requests currently being processed by this shard.

Returns:

A count of the active requests.

getForwardedRequestCount

long `getForwardedRequestCount()`

Retrieves the number of requests that this shard has forwarded since its inception.

Returns:

A count of the total number of forwarded requests.

getProcessedRequestCount

long `getProcessedRequestCount()`

Retrieves the number of requests that this shard has processed since its inception.

Returns:

A count of the total number of processed requests.

getContainerName

[String](#) `getContainerName()`

Retrieves the name of the container that is hosting this shard.

Returns:

The name of the container.

Since:

WAS XD 6.1.0.3

swapWithPrimary

[String](#) `swapWithPrimary()`

Causes this shard to swap roles with the primary shard for the partition. This shard becomes the primary while the shard that was previously the primary inherits this shard's former role.

If the role swap is not complete within 10 seconds, this operation will timeout.

Returns:

String the contains the return code of the operation

Since:

7.1.0.0 FIX1

See Also:

[ROLE_SWAP_SUCCESSFUL](#), [ROLE_SWAP_REQUESTED_WITH_SAME_TYPE](#), [ROLE_SWAP_TIMEOUT](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.management
Interface SessionMBean

public interface **SessionMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific session. The object name pattern for this MBean is:

com.ibm.websphere.objectgrid:type=Session,name=<id>,host=<host>,ogServerName=<server>

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.0.1, XC10

Method Summary	
l o n g	getAccessedSessionsCount() Gets accessed sessions count
l o n g	getAccessToNonExistentSessionCount() Gets access to non-existent session count
l o n g	getActiveSessionsCount() Gets active sessions count
l o n g	getAffinityBreaksCount() Gets affinity breaks count
l o n g	getCacheDiscardsCount() Gets cache discards count
l o n g	getCreatedSessionsCount() Gets the map count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getInvalidatedByTimeoutCount() Gets invalidated by timeout count
l	

o n g	<p>getInvalidatedSessionsCount() Gets invalidated sessions count</p>
l o n g	<p>getMemoryCount() Gets memory count</p>
S t r i n g	<p>getSessionID() Gets the ID of the session instance associated with this MBean.</p>
S t r i n g	<p>getSessionStatsModule() Gets a string representation of the SessionStatsModule attributes loaded up by the retrieveStatsModule() method.</p>
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . s t a t s . S e s s i o n S t a t s M o d u l e	<p>retrieveStatsModule() Gets the SessionStatsModule used to retrieve statistics associated with the session for this MBean.</p>

Method Detail

retrieveStatsModule

`com.ibm.websphere.objectgrid.stats.SessionStatsModule` **retrieveStatsModule()**

Gets the `SessionStatsModule` used to retrieve statistics associated with the session for this MBean.

Returns:

an `SessionStatsModule` for statistics associated with this session

See Also:

`SessionStatsModule`

getSessionID

[String](#) **getSessionID()**

Gets the ID of the session instance associated with this MBean.

Returns:

the ID of the session instance associated with this MBean.

getSessionStatsModule

[String](#) **getSessionStatsModule()**

Gets a string representation of the `SessionStatsModule` attributes loaded up by the `retrieveStatsModule()` method.

Returns:

String form of `SessionStatsModule`

See Also:

[retrieveStatsModule\(\)](#), `SessionStatsModule`

getCreatedSessionsCount

`long` **getCreatedSessionsCount()**

Gets the map count attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the number of entries in the map

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getNumEntries(boolean)`

getInvalidatedSessionsCount

`long` **getInvalidatedSessionsCount()**

Gets invalidated sessions count

Returns:

the count of invalidated Sessions

getActiveSessionsCount

long **getActiveSessionsCount()**
Gets active sessions count
Returns:
the count of active Sessions

getMemoryCount

long **getMemoryCount()**
Gets memory count
Returns:
memory count

getCacheDiscardsCount

long **getCacheDiscardsCount()**
Gets cache discards count
Returns:
cache discards count

getAffinityBreaksCount

long **getAffinityBreaksCount()**
Gets affinity breaks count
Returns:
affinity breaks count

getInvalidatedByTimeoutCount

long **getInvalidatedByTimeoutCount()**
Gets invalidated by timeout count
Returns:
count

getAccessToNonExistentSessionCount

long **getAccessToNonExistentSessionCount()**
Gets access to non-existent session count
Returns:
count

getAccessedSessionsCount

long **getAccessedSessionsCount()**
Gets accessed sessions count
Returns:
count

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

IBM WebSphere® DataPower® XC10
Appliance

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

Release 2.5 Client API Specification

com.ibm.websphere.objectgrid.management

Interface ServerMBean

All Known Subinterfaces:

[DynamicServerMBean](#)

public interface **ServerMBean**

This MBean interface allows a client process to access different attributes about a specific server process.

Since:

WAS XD 6.0.1, XC10

Method Summary

S
t
r
i
n
g

[getServerName\(\)](#)

Gets the name of the server associated with this MBean.

v
o
i
d

[modifyServerTraceSpec\(String spec\)](#)

Deprecated. This is deprecated in version 7.1. See [DynamicServerMBean.setTraceSpec\(String\)](#)

b
o
o
l
e
a
n

[stopServer\(\)](#)

Stops the server associated with this MBean.

Method Detail

getServerName

[String](#) [getServerName\(\)](#)

Gets the name of the server associated with this MBean.

Returns:

the server name

stopServer

boolean [stopServer\(\)](#)

Stops the server associated with this MBean.

Returns:

true if server was stopped, false if not

modifyServerTraceSpec

void **modifyServerTraceSpec**([String](#) spec)

Deprecated. *This is deprecated in version 7.1. See [DynamicServerMBean.setTraceSpec\(String\)](#)*

Modifies the trace spec for the server associated with this MBean.

Parameters:

spec - new trace specification

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid.management

Interface QuorumManagerMBean

public interface **QuorumManagerMBean**

Each catalog service has a QuorumManager and an associated MBean. The QuorumManager monitors and manages the quorum state of the catalog service grid. When quorum is enabled, the QuorumManager for each catalog service process detects when all catalog services in the grid have quorum or not. This MBean allows querying the current quorum state and allows administrators to force quorum when there is a network failure.

com.ibm.websphere.objectgrid:type=QuorumManager

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

The following notifications are available:

com.ibm.websphere.objectgrid.quorum.lost

Description: Catalog service quorum has been lost.

Message: A translated string identifying the number of active catalog servers and the number in quorum.

com.ibm.websphere.objectgrid.quorum.changed

Description: The catalog service has quorum, but the number of catalog servers required for quorum has changed.

Message: A translated string identifying the number of active catalog servers and the number in quorum.

Since:
7.0, XC10

Field Summary	
s t a t i c S t r i n g	QUORUM_CHANGED_NOTIFICATION
s t a t	

`String`
[QUORUM_LOST_NOTIFICATION](#)

Method Summary

`String[]`
[getActiveCatalogServerNames\(\)](#)
Retrieves the names of the known active catalog service processes.

`int`
[getActiveCatalogServers\(\)](#)
Retrieve the known number of active catalog service processes.

`int`
[getQuorumCatalogServers\(\)](#)
Retrieve the number of catalog service processes required for quorum.

`void`
[overrideQuorum\(\)](#)
This operation forces surviving catalog service grid processes to reestablish a quorum.

Field Detail

QUORUM_LOST_NOTIFICATION

static final `String` QUORUM_LOST_NOTIFICATION

See Also:

[Constant Field Values](#)

QUORUM_CHANGED_NOTIFICATION

static final `String` QUORUM_CHANGED_NOTIFICATION

See Also:

[Constant Field Values](#)

Method Detail

overrideQuorum

`void` **overrideQuorum()**
throws [Exception](#)

This operation forces surviving catalog service grid processes to reestablish a quorum.

If a portion the catalog service grid fails or is divided due to a network failure, the grid will lose quorum. Once the administrator identifies the failure and the viable portion of the grid, this operation can be invoked on any of the surviving catalog service processes to reestablish a quorum. Reestablishing a quorum will allow the catalog service to

continue to react to failures and topology changes.

Throws:

[Exception](#)

getActiveCatalogServers

int `getActiveCatalogServers()`

Retrieve the known number of active catalog service processes.

Returns:

the known number of active catalog service processes.

getQuorumCatalogServers

int `getQuorumCatalogServers()`

Retrieve the number of catalog service processes required for quorum.

Returns:

the number of catalog service processes required for quorum.

getActiveCatalogServerNames

[String](#)[] `getActiveCatalogServerNames()`

Retrieves the names of the known active catalog service processes.

Returns:

the names of the known active catalog service processes.

Since:

7.1

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management

Interface QueryManagerMBean

public interface **QueryManagerMBean**

This MBean interface allows a client process to perform operations on and get status from an ObjectGrid Query Manager running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=QueryManager,name=<grid name>,mapset=<mapset name>,partition=<partition number>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1.0.5, XC10

Method Summary	
d o u b l e	<p>getPlanCreationTime(String query) Gets the query's plan creation time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryExecutionCount(String query) Gets the query's execution count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryExecutionTime(String query) Gets the query's execution time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryFailureCount(String query) Gets the query's failure count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryResultCount(String query) Gets the query's result count attribute loaded up by the retrieveStatsModule() method.</p>

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
Q
u
e
r
y
S
t
a
t
s
M
o
d
u
l
e

[retrieveStatsModule\(String query\)](#)

Gets the QueryStatsModule used to retrieve statistics associated with the specified query String

Method Detail

getPlanCreationTime

double **getPlanCreationTime**([String](#) query)

Gets the query's plan creation time attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the plan creation time for this query in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getPlanCreationTime(boolean copy)`

getQueryExecutionTime

double **getQueryExecutionTime**([String](#) query)

Gets the query's execution time attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the execution time for this query in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryExecutionTime(boolean copy)`

getQueryExecutionCount

double `getQueryExecutionCount(String query)`

Gets the query's execution count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the execution count for this query

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryExecutionCount(boolean copy)`

getQueryResultCount

double `getQueryResultCount(String query)`

Gets the query's result count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the result count for this query

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryResultCount(boolean copy)`

getQueryFailureCount

double `getQueryFailureCount(String query)`

Gets the query's failure count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the failure count for this query

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryFailureCount(boolean copy)`

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.QueryStatsModule `retrieveStatsModule(String query)`

Gets the `QueryStatsModule` used to retrieve statistics associated with the specified query String

Returns:

an `QueryStatsModule` for statistics associated with the specified query String

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management

Interface PlacementServiceMBean

All Superinterfaces:

com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean

```
public interface PlacementServiceMBean
extends com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean
```

This MBean interface allows a client process to perform operations on and get status from the PlacementService running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=PlacementService
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary

s t a t i c i n t	<p>ALL</p> <p>Constant representing a all shard types</p>
s t a t i c i n t	<p>ASYNCHRONOUS_REPLICA</p> <p>Constant representing an asynchronous replica shard type.</p>
s t a t i c i n t	<p>PRIMARY</p> <p>Constant representing a primary shard type.</p>
s t a	

t
i
c

i
n
t

[SYNCHRONOUS_REPLICA](#)

Constant representing a synchronous replica shard type.

Fields inherited from interface

com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean

HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE, HEARTBEAT_FREQUENCY_LEVEL_RELAXED,
HEARTBEAT_FREQUENCY_LEVEL_TYPICAL

Method Summary

S
t
r
i
n
g

[balanceShardTypes](#)([String](#) objectGridName, [String](#) mapSetName)

The placement service will examine the distribution of primaries and replicas for a given mapSet and attempt (if zone rules and other balancing constraints allow) to achieve a consistent primary to replica ratio across the set of containers.

S
t
r
i
n
g

[balanceStatus](#)([String](#) objectGridName, [String](#) mapSetName)

Check the balance status (suspended or resumed) for a specified MapSet.

S
t
r
i
n
g

[collectContainerStatus](#)([String](#) objectGridName, [String](#) mapSetName)

Retrieves the container status for all containers in the domain.

b
o
l
e
a
n

[enableForPlacement](#)([String](#) containerName)

Re-enables a disabled container for placement.

S
t
r
i
n
g

[getCoreGroups](#)()

Gets the coregroup status.

T
a
b
u
l
a
r
D
a
t
a

[getDisabledForPlacement](#)()

Retrieves TabularData of containers which have been disabled because of the failure of shard placement operations.

b
o
l
e
a
n

[getInitialized](#)()

Returns whether this instance of the Placement Service finished initialization.

S t r i n g	<p>getObjectGridNames() Gets the names of all ObjectGrids and their mapsets in the domain.</p>
S t r i n g	<p>listCoreGroupMembers(String coreGroupName) List the coregroup members for a given coregroup.</p>
S t r i n g	<p>listObjectGridPlacement(String objectGridName, String mapSetName) List the placement of shards for each container in the domain.</p>
S t r i n g	<p>listObjectGridPlacementStatus(String objectGridName, String mapSetName) List the current placement status.</p>
S t r i n g	<p>listPartition(String objectGridName, String mapSetName, String partitionId) List the partition placement status in the domain.</p>
S t r i n g	<p>listShards(String objectGridName, String mapSetName, String containerName, int mask) List the shard placement status.</p>
S t r i n g	<p>listVerifiedRoutingTable(String objectGridName) Deprecated.</p>
S t r i n g	<p>replaceLostShards(String objectGridName, String mapSetName) Lost shards are placed onto the UNREPAIRED container when autoReplaceLostShards is disabled.</p>
S t r i n g	<p>resumeBalancing(String objectGridName, String mapSetName) Execute balancing operation at next opportunity and allow execution of future balancing attempts for the map set specified.</p>
L i s t	<p>retrieveAllServersJMXAddresses() Retrieves a List of JMX addresses for all servers that have registered with the placement service.</p>
S t r	<p>retrieveMapSetName(String gridName, String mapName)</p>

i n q	Retrieves the name of the MapSet in which the specified map is defined.
L i s t	retrieveServerJMXAddress (String hostName, String serverName) Retrieves a List of JMX address strings for a specific host and server name that has registered with the placement service.
T a b u l a r D a t a	retrieveServerJMXAddressesWithInfo (String hostName, String serverName) Retrieves TabularData of JMX address strings for a specific host and server name that has registered with the placement service.
S t r i n g	suspendBalancing (String objectGridName, String mapSetName) Prevent future balancing attempts for a specific map set.
S t r i n g	tearDownServers (String [] servers) Each of the container servers that are passed into this method will be stopped.
S t r i n g	triggerPlacement (String objectGridName, String mapSetName) Placement normally occurs implicitly after an event such as an ObjectGrid container starting or stopping.

Methods inherited from interface com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean
getHeartBeatFrequencyLevel, setHeartBeatFrequencyLevel

Field Detail

PRIMARY

static final int PRIMARY

Constant representing a primary shard type.

See Also:

[Constant Field Values](#)

SYNCHRONOUS_REPLICA

static final int SYNCHRONOUS_REPLICA

Constant representing a synchronous replica shard type.

See Also:

[Constant Field Values](#)

ASYNCHRONOUS_REPLICA

static final int ASYNCHRONOUS_REPLICA

Constant representing an asynchronous replica shard type.

See Also:

[Constant Field Values](#)

ALL

static final int ALL

Constant representing a all shard types

See Also:

[Constant Field Values](#)

Method Detail

retrieveServerJMXAddress

[List](#) retrieveServerJMXAddress([String](#) hostName,
[String](#) serverName)

Retrieves a List of JMX address strings for a specific host and server name that has registered with the placement service.

Parameters:

hostName - The name of the host to retrieve the JMX addresses.

serverName - The name of the server to retrieve the JMX addresses.

Returns:

the List of all JMX address strings for the specified host and server name.

retrieveServerJMXAddressesWithInfo

[TabularData](#) retrieveServerJMXAddressesWithInfo([String](#) hostName,
[String](#) serverName)

Retrieves TabularData of JMX address strings for a specific host and server name that has registered with the placement service. Null host or null server names can be used to retrieve several results. Using a null host and null server name will retrieve all of the servers. The TabularData contains CompositeData with the following items, where each CompositeData represents a server:

Item Name	Type	Description
------------------	-------------	--------------------

JMXServiceURL	String	JMX Service URL
---------------	--------	-----------------

HostName	String	Host name
----------	--------	-----------

ServerName	String	Server Name
------------	--------	-------------

Parameters:

hostName - The name of the host for which to retrieve JMX addresses. Use null or an empty string to retrieve JMX addresses for any host.

serverName - The name of the server for which to retrieve JMX addresses. Use null or an empty string to retrieve JMX addresses for any server.

Returns:

the TabularData of all JMX address strings for the specified host and server name.

Since:

8.5

retrieveAllServersJMXAddresses

[List](#) retrieveAllServersJMXAddresses()

Retrieves a List of JMX addresses for all servers that have registered with the placement service.

Returns:

the List of all servers' JMX addresses

collectContainerStatus

[String](#) collectContainerStatus([String](#) objectGridName,
[String](#) mapSetName)

Retrieves the container status for all containers in the domain.

The results are returned in the following format:

```
<container name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>">
  <shard type="<type>" partitionName="<partition>"/>
</container>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get container status.

mapSetName - The name of the mapset for which to get the container status.

Returns:

The String status object for all containers in XML form.

listObjectGridPlacement

[String](#) listObjectGridPlacement([String](#) objectGridName,
[String](#) mapSetName)

List the placement of shards for each container in the domain.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset">
  <container name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>">
    <shard type="<type>" partitionName="<partition>" reserved="<true>"/>
  /container>
</objectGrid>
```

NOTE: The default value for the "reserved" attribute is false.

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.

mapSetName - The name of the mapset for which to get the placement status.

Returns:

The placement status in XML form.

listObjectGridPlacementStatus

```
String listObjectGridPlacementStatus(String objectGridName,  
                                     String mapSetName)
```

List the current placement status.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <configuration>  
    <attribute name="<placementStrategy>" value="<strategy>"/>  
    <attribute name="<numInitialContainers>" value="<num>"/>  
    <attribute name="<minSyncReplicas>" value="<min>"/>  
    <attribute name="<developmentMode>" value="<mode>"/>  
  </configuration>  
  <runtime>  
    <attribute name="<numContainers>" value="<num>"/>  
    <attribute name="<numMachines>" value="<num>"/>  
    <attribute name="<numOutstandingWorkItems>" value="<num>"/>  
    <attribute name="<numActiveZones>" value="<num>"/>  
  </runtime>  
</objectGrid>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.

mapSetName - The name of the mapset for which to get the placement status.

Returns:

The placement status in XML form.

getCoreGroups

```
String getCoreGroups()
```

Gets the coregroup status.

The results are returned in the following format:

```
<coreGroup name="<coregroup>">  
  <coreGroupLeader hostName="<host>" serverName="<server>"/>  
  <coreGroupMember hostName="<host>" serverName="<server>"/>  
</coreGroup>
```

Returns:

the coregroup status in XML form.

listCoreGroupMembers

```
String listCoreGroupMembers(String coreGroupName)
```

List the coregroup members for a given coregroup.

The results are returned in the following format:

```
<coreGroup name="<coregroup>">  
  <coreGroupMember hostName="<host>" serverName="<server>"/>  
</coreGroup>
```

Parameters:

coreGroupName - The name of the coregroup for which to get the members.

Returns:

The coregroup members in XML form.

listPartition

```
String listPartition(String objectGridName,  
                    String mapSetName,  
                    String partitionId)
```

List the partition placement status in the domain. The results are returned in the following format:

```
<partition name="<partition>">  
  <shard type="<type>" containerName="<container>" hostName="<host>" serverName="<server>"/>  
</partition>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.
mapSetName - The name of the mapset for which to get the placement status.
partitionId - The name of the partition for which to get the placement status.

Returns:

The partition placement status in the XML form.

listShards

```
String listShards(String objectGridName,  
                 String mapSetName,  
                 String containerName,  
                 int mask)
```

List the shard placement status.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <container name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>">  
    <shard type="<type>" partitionName="<partition>"/>  
  </container>  
</objectGrid>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.
mapSetName - The name of the mapset for which to get the placement status.
containerName - The name of the container for which to get the placement status. If empty string (""), get shard placement for all containers.
mask - The Integer mask to determine for which shard types to get status.

Returns:

The shard placement status in XML form.

See Also:

[ALL](#), [PRIMARY](#), [SYNCHRONOUS_REPLICA](#), [ASYNCHRONOUS_REPLICA](#)

getObjectGridNames

```
String getObjectGridNames()
```

Gets the names of all ObjectGrids and their mapsets in the domain.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>"/>
```

Returns:

the names of all ObjectGrids and their mapsets in the domain in XML form.

replaceLostShards

`String replaceLostShards(String objectGridName,
String mapSetName)`

Lost shards are placed onto the UNREPAIRED container when `autoReplaceLostShards` is disabled. Shards on the UNREPAIRED will not be placed until this method is called.

Calling this method will move shards off the UNREPAIRED container onto the UNASSIGNED container.

Balance and placement operations will be queued up for the MapSet specified. These operations will execute when all outstanding placement work from previous events has completed.

The string returned is an XML representation of the shards that moved as a result of the call to this method.

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <shard type="<type>" partitionName="<partition>">  
    <currentContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />  
    <previousContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />  
  </shard>  
</objectGrid>
```

The returned XML will look as follows when no shards have been moved:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <!-- No shards were moved -->  
</objectGrid>
```

Parameters:

`objectGridName` - replace lost shards for this ObjectGrid

`mapSetName` - replace lost shards for this MapSet

Returns:

An XML String containing shards that have moved

Since:

WAS XD 6.1.0.5

triggerPlacement

`String triggerPlacement(String objectGridName,
String mapSetName)`

Placement normally occurs implicitly after an event such as an ObjectGrid container starting or stopping.

Calling this method will trigger a placement operation for the ObjectGrid and MapSet specified.

Under normal circumstances, the `numInitialContainers` attribute (in the deployment policy) must be met in order for placement to occur. However, when this method is called, the `numInitialContainers` value is ignored.

The string returned is an XML representation of the shards that moved as a result of the call to this method.

```

<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <shard type="<type>" partitionName="<partition>">
    <currentContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
    <previousContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
  </shard>
</objectGrid>

```

The returned XML will look as follows when no shards have been moved:

```

<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <!-- No shards were moved -->
</objectGrid>

```

Parameters:

objectGridName - trigger placement for this ObjectGrid
mapSetName - trigger placement for this MapSet

Returns:

An XML String containing shards that have moved

Since:

WAS XD 6.1.0.5

See Also:

ObjectGridDeployment.addMapSet(com.ibm.websphere.objectgrid.deployment.MapSet),
MapSet.setNumInitialContainers(int)

tearDownServers

[String](#) tearDownServers([String](#)[] servers)

Each of the container servers that are passed into this method will be stopped. If the server cannot be reached, all of the server's artifacts will be removed.

Use this method if servers are found to be in a corrupt state or bindings need to be cleared from the catalog server.

The string returned is an XML representation of the results of the attempt to tear down each of the servers. If the command is successful, the XML will look as follows:

```

<domain name="<domain>">
  <server name="<server>" tearDownSuccessful="true" />
  <server name="<server>" tearDownSuccessful="true" />
</domain>

```

If the command is not successful, the string will look as follows (where the exception element is only present if an exception is part of the failure):

```

<domain name="<domain>">
  <server name="<server>" tearDownSuccessful="false" reason="<String>">
    <exception type="<String>" message="<String>" stack="<String>" />
  </server>
</domain>

```

Parameters:

servers - String array of servers to tear down.

Returns:

An XML String containing the results of tear down attempts.

Since:

WAS XD 6.1.0.5 FIX2

listVerifiedRoutingTable

[@Deprecated](#)

[String](#) listVerifiedRoutingTable([String](#) objectGridName)

Deprecated.

This method is deprecated. The `com.ibm.websphere.objectgrid.client.RouteTableValidation` utility replaces this method.

Calling this method will return an XML string of the current known routing table. The Placement service will contact each shard and return state on whether it was able to verify that's shard's existence. All shards will be included in the XML doc, whether they were reachable or not. The user can use the `reachable` attribute below to filter valid or invalid shards.

```
<objectGrid name="<objectgrid>" name="<name>">
  <primary zone="<zone>"> partition="<partition>"> state="<reachable>"> ipaddress="<ipaddress>">
  <replica zone="<zone>"> partition="<partition>"> state="<reachable>"> ipaddress="<ipaddress>">
  </primary>
</objectGrid>
```

Parameters:

objectGridName - retrieve routing table for this ObjectGrid

Returns:

An XML String containing a pre-verified routing table

Since:

WAS XD 6.1.0.5 FIX2

retrieveMapSetName

[String](#) retrieveMapSetName([String](#) gridName,
[String](#) mapName)

Retrieves the name of the MapSet in which the specified map is defined.

Parameters:

gridName - the name of the ObjectGrid

mapName - the name of the map

Returns:

the name of the MapSet in which the specified map is defined.

Since:

7.0

getInitialized

boolean `getInitialized()`

Returns whether this instance of the Placement Service finished initialization.

Returns:

true if initialized, false if not initialized.

Since:

8.5

balanceShardTypes

[String](#) `balanceShardTypes`([String](#) objectGridName,
[String](#) mapSetName)

The placement service will examine the distribution of primaries and replicas for a given mapSet and attempt (if zone rules and other balancing constraints allow) to achieve a consistent primary to replica ratio across the set of containers.

If the number of primaries or the number of replicas do not divide evenly across the containers, some tolerance must be allowed for the ratio to differ from container to container. However, the difference in the number of primaries from one container to the next will not be greater than 1. Similarly, the difference in the number of replicas from one container to the next will not be greater than 1.

Null arguments are not allowed as input to this method.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <shard type="<type>" partitionName="<partition>">
    <currentContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
    <previousContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
  </shard>
</objectGrid>
```

If no shards were moved or a problem was encountered attempting to execute this method, no shard elements will appear in the XML output. A detail element will appear instead. The message attribute will have further information.

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <detail message="<message>" />
</objectGrid>
```

Parameters:

objectGridName - the grid
mapSetName - the map set within the grid

Returns:

An XML String containing the results of the attempt to redistribute shards for better primary/replica balance

Since:

7.1.1

suspendBalancing

[String](#) `suspendBalancing`([String](#) objectGridName,
[String](#) mapSetName)

Prevent future balancing attempts for a specific map set. Balancing work that is in progress will be allowed to complete.

Other placement activities are allowed to execute while balancing is suspended.

- shard promotion due to container loss
- shard role swap
- shard reservation
- triggerPlacement
- replaceLostShards

Balancing will remain suspended until it is resumed by calling [resumeBalancing\(String, String\)](#).

Null arguments are not allowed as input to this method.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>" />
</objectGrid>
```

Additionally, an optional detail element may be contained within the suspendBalancing element. The detail element will include additional data regarding execution of this method. The XML result will be in the following format when a detail element is included:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>">
    <detail message="<message>" />
  </suspendBalancing/>
</objectGrid>
```

Parameters:

objectGridName - suspend balancing for the map set specified within this ObjectGrid
mapSetName - suspend balancing for this map set

Returns:

An XML String containing the results of the attempt to suspend balancing

Since:

7.1.0.3

See Also:

[resumeBalancing\(String, String\)](#)

resumeBalancing

```
String resumeBalancing(String objectGridName,  
                        String mapSetName)
```

Execute balancing operation at next opportunity and allow execution of future balancing attempts for the map set specified. Balancing is executed in reaction to key placement events. Such events include containers starting and containers stopping.

By default, balancing work is executed unless [suspendBalancing\(String, String\)](#) has been called for the map set.

Null arguments are not allowed as input to this method.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>" />
</objectGrid>
```

Additionally, an optional detail element may be contained within the suspendBalancing element. The detail element will include additional data regarding execution of this method. The XML result will be in the following format when a detail element is included:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>">
    <detail message="<message>" />
  </suspendBalancing/>
</objectGrid>
```

Parameters:

objectGridName - resume balancing for the map set specified within this ObjectGrid
mapSetName - resume balancing for this map set

Returns:

An XML String containing the results of the attempt to resume balancing

Since:

7.1.0.3

See Also:

[suspendBalancing\(String, String\)](#)

balanceStatus

[String](#) **balanceStatus**([String](#) objectGridName,
[String](#) mapSetName)

Check the balance status (suspended or resumed) for a specified MapSet.

Null arguments are not allowed as input to this method.

The string returned is an XML representation of the balance status. The XML will look as follows:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <balanceStatus suspended="<suspended>" />  
</objectGrid>
```

Additionally, an optional detail element may be contained within the balanceStatus element. When balancing has been pre-suspended, the message attribute of the detail element will contain the following message.

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <balanceStatus suspended="true" >  
    <detail message="Balancing has been pre-suspended for this mapSet." />  
  </balanceStatus>  
</objectGrid>
```

Parameters:

objectGridName - check balance status for the map set specified within this ObjectGrid
mapSetName - check balance status for this map set

Returns:

An XML String containing the balance status

Since:

7.1.1

See Also:

[suspendBalancing\(String, String\)](#), [resumeBalancing\(String, String\)](#)

enableForPlacement

boolean **enableForPlacement**([String](#) containerName)

Re-enables a disabled container for placement. A container may become disabled because of a failure to place a shard into the container.

Use the [getDisabledForPlacement\(\)](#) attribute to determine which containers are disabled.

Parameters:

containerName - The name of the container to re-enable.

Returns:

Answers true if the container's status was changed from disabled to enabled, false if

the container was already enabled for placement.

Since:

8.6, XC10 2.5

getDisabledForPlacement

[TabularData](#) `getDisabledForPlacement()`

Retrieves TabularData of containers which have been disabled because of the failure of shard placement operations. The TabularData contains CompositeData with the following items, where each CompositeData represents a container:

Item Name Type Description

Container String A container that has been disabled for placement

Returns:

A TabularData of the names of disabled containers.

Since:

8.6, XC10 2.5

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
ew	ge	ss	ed	ted				
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All			
			Classes					

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid.management

Interface PlacementMediationServiceMBean

public interface PlacementMediationServiceMBean

This MBean interface allows a client process to perform operations on and get status from the PlacementMediationService running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=PlacementMediationService
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

7.1, XC10

Method Summary

C
o
m
p
o
s
i
t
i
o
n
s
D
e
t
a
i
l

[dismissLink](#)(String foreignDomain)

Dismiss a previously established link with the foreign domain specified.

C
o
m
p
o
s
i
t
i
o
n
s
D
e
t
a
i
l

[establishLink](#)(String foreignDomain, String endPoints)

Establish a link between this domain and the foreign domain specified.

T
a
b
l
e
s
D
e
t
a
i
l

[getLinkedDomains](#)()

Retrieve the foreign domains that have an active link with the local domain.

[getLinkedDomainsWithGrids\(\)](#)

Retrieve the foreign domains that have an active link with the local domain and the map sets eligible for linking.

Method Detail

establishLink

[CompositeData](#) **establishLink**([String](#) foreignDomain,
[String](#) endPoints)

Establish a link between this domain and the foreign domain specified. This is functionally equivalent to providing the foreign domain and its end points in the server properties file at server startup time.

Domains that are linked will share placement with each other. When compatible map sets are detected within linked domains, a multi-primary topology will be achieved. Data written to a primary in either domain will be asynchronously replicated to the other domain.

The result is a [CompositeData](#) that includes the following items:

Item Name Type Description

Result	String	The result of the attempt.
StatusBefore	String	The status of the link before the attempt was made to establish the link.
StatusAfter	String	The status of the link after the attempt was made to establish the link.

Parameters:

foreignDomain - the name of the foreign domain
endPoints - end points of the foreign domain

Returns:

[CompositeData](#) representing the status of the attempt to link with the foreign domain

See Also:

[CatalogServerProperties.setForeignDomains\(String\)](#),
[CatalogServerProperties.setDomainEndPoints\(String, String\)](#)

dismissLink

[CompositeData](#) **dismissLink**([String](#) foreignDomain)

Dismiss a previously established link with the foreign domain specified. Any map sets that were participating in a multi-primary topology will be disconnected from each other. Data will no longer be replicated from between domains.

The result is a [CompositeData](#) that includes the following items:

<u>Item Name</u>	<u>Ty</u>	<u>Description</u>
-------------------------	------------------	---------------------------

Result	String	The result of the attempt. Can be one of: SUCCESS, FAILURE, NOP
Status Before	String	The status of the link before the attempt was made to dismiss the link. Can be one of: LINKED, ESTABLISHING_LINK, UNLINKED, DISMISSING_LINK
Status After	String	The status of the link after the attempt was made to dismiss the link. Can be one of: LINKED, ESTABLISHING_LINK, UNLINKED, DISMISSING_LINK

Parameters:

foreignDomain - the name of the foreign domain

Returns:

CompositeData representing the status of the attempt to dismiss the link with the foreign domain

getLinkedDomains

[TabularData](#) `getLinkedDomains()`

Retrieve the foreign domains that have an active link with the local domain.

The result is a TabularData where each row is a CompositeData that includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

Domain	String	The name of the foreign domain linked to the local domain.
--------	--------	--

Returns:

TabularData representing the foreign domains linked to this domain

getLinkedDomainsWithGrids

[TabularData](#) `getLinkedDomainsWithGrids()`

Retrieve the foreign domains that have an active link with the local domain and the map sets eligible for linking.

The result is a TabularData where each row is a CompositeData that includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

Domain	String	The name of the foreign domain linked to the local domain.
ObjectGrid	String	The name of the ObjectGrid that is compatible with the foreign domain.
MapSet	String	The name of the map set that is compatible with the foreign domain.

Returns:

TabularData representing the foreign domains and eligible map sets linked to this domain

Since:

8.6, XC10 2.5

com.ibm.websphere.objectgrid.management
Interface ObjectGridMBean

public interface **ObjectGridMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific ObjectGrid on a server process. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ObjectGrid,name=<objectgrid>,mapset=<mapset>,partition=<partition id>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:
 WAS XD 6.0.1, XC10

Method Summary	
S t r i n g	<p>getContainerName() Gets the name of the container name containing the replication group member for the ObjectGrid associated with this MBean.</p>
l o n g	<p>getCurrentRevision() Retrieves the current revision number of this ObjectGrid shard.</p>
S t r i n g	<p>getDomainName() Retrieves the domain name of this ObjectGrid shard.</p>
I n t e r f a c e	<p>getKnownRevisions() An ObjectGrid shard may exist over several different lifetimes.</p>
S t r i n g	<p>getLifetimeId() Retrieves the lifetime id for this ObjectGrid shard.</p>

n g	
S t r i n g	getObjectGridName() Gets the name of the ObjectGrid associated with this MBean.
l o n g	getOGCount() Gets the ObjectGrid count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGMaxTranTime() Gets the maximum transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
d o u b l e	getOGMeanTranTime() Gets the mean transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGMinTranTime() Gets the minimum transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
S t r i n g	getOGStatsModule() Gets a string representation of the OGStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGTotalTranTime() Gets the total transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGTransPerSecond() Transactions per second attribute loaded up by the retrieveStatsModule call.
T a b l e	getPrimaryShardLinks() Get the shard's list of foreign or domestic linked primaries.
S t r i n g	getServerName() Gets the name of the server containing the replication group member for the ObjectGrid associated with this MBean.
c o m .	

i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
O
G
S
t
a
t
s
M
o
d
u
l
e

[retrieveStatsModule\(\)](#)

Gets the OGStatsModule used to retrieve statistics associated with the ObjectGrid for this MBean.

Method Detail

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.OGStatsModule **retrieveStatsModule()**

Gets the OGStatsModule used to retrieve statistics associated with the ObjectGrid for this MBean.

Returns:

an OGStatsModule for statistics associated with this ObjectGrid

See Also:

OGStatsModule

getObjectGridName

[String](#) getObjectGridName()

Gets the name of the ObjectGrid associated with this MBean.

Returns:

name of the ObjectGrid

getServerName

[String](#) getServerName()

Gets the name of the server containing the replication group member for the ObjectGrid associated with this MBean.

Returns:

the name of server containing the replication group member for the ObjectGrid associated with this MBean.

getContainerName

[String](#) getContainerName()

Gets the name of the container name containing the replication group member for the ObjectGrid associated with this MBean.

Returns:

the name of container containing the replication group member for the ObjectGrid associated with this MBean.

Since:

8.5

getOGStatsModule

[String](#) getOGStatsModule()

Gets a string representation of the OGStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

String form of OGStatsModule

See Also:

[retrieveStatsModule\(\)](#), OGStatsModule

getOGCount

long getOGCount()

Gets the ObjectGrid count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

the number of transactions

See Also:

[retrieveStatsModule\(\)](#), OGStatsModule.getTransactionTime(String, boolean)

getOGMaxTranTime

long getOGMaxTranTime()

Gets the maximum transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

the maximum transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), OGStatsModule.getTransactionTime(String, boolean)

getOGMinTranTime

long **getOGMinTranTime()**

Gets the minimum transaction time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the minimum transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getOGMeanTranTime

double **getOGMeanTranTime()**

Gets the mean transaction time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the mean transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getOGTotalTranTime

long **getOGTotalTranTime()**

Gets the total transaction time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the total transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getOGTransPerSecond

long **getOGTransPerSecond()**

Transactions per second attribute loaded up by the `retrieveStatsModule` call. `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the transactions per second for the ObjectGrid

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getCurrentRevision

long **getCurrentRevision()**

Retrieves the current revision number of this ObjectGrid shard.

Returns:

the current revision number of this ObjectGrid shard.

Since:

7.1

getDomainName

[String](#) **getDomainName()**

Retrieves the domain name of this ObjectGrid shard.

Returns:

the name of the domain name of this ObjectGrid shard.

Since:

7.1

getLifetimeId

[String](#) getLifetimeId()

Retrieves the lifetime id for this ObjectGrid shard.

Returns:

the lifetime id for this ObjectGrid shard.

Since:

7.1

getKnownRevisions

[TabularData](#) getKnownRevisions()

An ObjectGrid shard may exist over several different lifetimes. As such, each shard instance will have a unique lifetime id and revision number associated with it. This method returns a TabularData object representing the known history of revision numbers for each lifetime. Each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Domain	String	The domain name of this ObjectGrid shard.	7.1
Server	String	The name of the server owning the lifetime id.	8.5
LifetimeId	String	The lifetime id of this ObjectGrid shard.	7.1
Revision	Long	The revision of this ObjectGrid shard.	7.1

Returns:

TabularData representing the known lifetimes and revisions of this shard.

Throws:

[OpenDataException](#)

Since:

7.1

See Also:

[TabularData](#)

getPrimaryShardLinks

[TabularData](#) getPrimaryShardLinks()

Get the shard's list of foreign or domestic linked primaries.

This method returns a TabularData object representing the current state of each primary shard link.

Each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
RemoteDomain	String	The catalog service domain name of the remote primary shard..
RemoteContainer	String	The container name of the remote primary shard.
Status	String	The status of the link. Valid states include: online and recovery.

Returns:

TabularData representing the linked primaries

Since:

7.1.1

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

IBM WebSphere® DataPower® XC10
Appliance

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

Release 2.5 Client API Specification

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

com.ibm.websphere.objectgrid.management

Interface MapMBean

public interface **MapMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific map on a server process. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ObjectMap,name=<map>,partition=<partition id>,objectgrid=<objectgrid>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.0.1, XC10

Method Summary

S t r i n g	<p>getContainerName() Gets the name of the container containing the replication group member for the map associated with this MBean.</p>
d o u b l e	<p>getMapBatchUpdateMaxTime() Gets the maximum batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>
d o u b l e	<p>getMapBatchUpdateMeanTime() Gets the mean batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>
d o u b l e	<p>getMapBatchUpdateMinTime() Gets the minimum batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>
d o u b l e	<p>getMapBatchUpdateTotalTime() Gets the total batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>

l o n g	getMapCountStatistic() Gets the map count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getMapGetCountStatistic() Gets the get count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getMapHitCountStatistic() Gets the hit count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
d o u b l e	getMapHitRateStatistic() Gets the hit rate attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
S t r i n g	getMapName() Gets the name of the map associated with this MBean.
S t r i n g	getMapStatsModule() Gets a string representation of the MapStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getMapUsedBytes() Gets the used bytes attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
S t r i n g	getObjectGridName() Gets the name of the ObjectGrid containing the map associated with this MBean.
i n t	getPartitionId() Retrieves the partition identifier for this map instance.
S t r i n g	getServerName() Gets the name of the server containing the replication group member for the map associated with this MBean.
I a b l e	retrieveEntries(String regex) Operation to iterate through all of the entries in this map, convert the key to string form, then match the string against the regular expression if passed, finally return the matching entries.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
M
a
p
S
t
a
t
s
M
o
d
u
l
e

[retrieveStatsModule\(\)](#)

Gets the MapStatsModule used to retrieve statistics associated with the map for this MBean.

Method Detail

retrieveStatsModule

`com.ibm.websphere.objectgrid.stats.MapStatsModule` **retrieveStatsModule()**

Gets the MapStatsModule used to retrieve statistics associated with the map for this MBean.

Returns:

A MapStatsModule for statistics associated with this map.

See Also:

MapStatsModule

getMapName

[String](#) **getMapName()**

Gets the name of the map associated with this MBean.

Returns:

The name of the map.

getObjectGridName

[String](#) getObjectGridName()

Gets the name of the ObjectGrid containing the map associated with this MBean.

Returns:

The name of the ObjectGrid for the map associated with this MBean.

getServerName

[String](#) getServerName()

Gets the name of the server containing the replication group member for the map associated with this MBean.

Returns:

The name of server containing the replication group member for the map associated with this MBean.

getMapStatsModule

[String](#) getMapStatsModule()

Gets a string representation of the MapStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

The String form of MapStatsModule

See Also:

[retrieveStatsModule\(\)](#), MapStatsModule

getMapCountStatistic

long getMapCountStatistic()

Gets the map count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

The number of entries in the map.

See Also:

[retrieveStatsModule\(\)](#), MapStatsModule.getNumEntries(boolean)

getMapHitRateStatistic

double getMapHitRateStatistic()

Gets the hit rate attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

The hit rate for the map.

See Also:

[retrieveStatsModule\(\)](#), MapStatsModule.getHitRate(boolean)

getMapGetCountStatistic

long **getMapGetCountStatistic()**

Gets the get count attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The get count for the map.

Since:

7.1

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getHitRate(boolean)`

getMapUsedBytes

long **getMapUsedBytes()**

Gets the used bytes attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

The used bytes statistics are accurate only when you are using simple objects or the `COPY_TO_BYTES` copy mode.

Returns:

The number of bytes in use by the map.

Since:

7.1

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getUsedBytes(boolean)`

getMapHitCountStatistic

long **getMapHitCountStatistic()**

Gets the hit count attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The hit count for the map.

Since:

7.1

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getHitRate(boolean)`

getMapBatchUpdateMeanTime

double **getMapBatchUpdateMeanTime()**

Gets the mean batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The mean batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getMapBatchUpdateMaxTime

double **getMapBatchUpdateMaxTime()**

Gets the maximum batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The maximum batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getMapBatchUpdateMinTime

double **getMapBatchUpdateMinTime()**

Gets the minimum batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The minimum batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getMapBatchUpdateTotalTime

double **getMapBatchUpdateTotalTime()**

Gets the total batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The total batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getPartitionId

int **getPartitionId()**

Retrieves the partition identifier for this map instance.

Returns:

The partition identifier.

Since:

WAS XD 6.1.0.4

getContainerName

[String](#) **getContainerName()**

Gets the name of the container containing the replication group member for the map associated with this MBean.

Returns:

The name of container containing the replication group member for the map associated with this MBean.

Since:

8.5

retrieveEntries

[TabularData](#) **retrieveEntries([String](#) regex)**

Operation to iterate through all of the entries in this map, convert the key to string form, then match the string against the regular expression if passed, finally return the matching entries. This method could potentially return a very large data structure so care should be taken to ensure the regular expression will reduce the number of keys appropriately.

Each `CompositeData` (row in the `TabularData`) contains the following items:

Item Name	Type	Description
KeyName	String	The domain name of this ObjectGrid shard.
LifetimeIndex	Short	The lifetime index for revisioning.
Revision	Long	The revision number of the last update.

Parameters:

regex - the regular expression to apply to the String form of the key. It should be used in narrowing the entries returned. If null, all entries are returned.

Returns:

A table of entries containing the user readable (String) form of the key and some meta information about the entry.

Since:

7.1.1

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [ALL Classes](#)

Release 2.5 Client API Specification

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.management

Interface HashIndexMBean

public interface **HashIndexMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific HashIndex on a server process. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=HashIndex,name=<index-name>,partition=<partition id>,objectgrid=<objectgrid>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1.0.5, XC10

Method Summary	
d o u b l e	<p>getBatchUpdateCount() Gets the index's batchupdate count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getFindCollisionCount() Gets the index's collision count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getFindCount() Gets the index's find count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getFindDurationTime() Gets the index's find duration time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getFindFailureCount() Gets the index's failure count attribute loaded up by the retrieveStatsModule() method.</p>

d o u b l e	<p>getFindResultCount() Gets the index's result count attribute loaded up by the retrieveStatsModule() method.</p>
S t r i n g	<p>getParentMapName() Gets the index's parent map name</p>
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . s t a t s . H a s h I n d e x S t a t s M o d u l e	<p>retrieveStatsModule() Gets the HashIndexStatsModule used to retrieve statistics associated with this index</p>

Method Detail

getParentMapName

[String](#) getParentMapName()

Gets the index's parent map name

Returns:

the name of the map which this index belongs to

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.HashIndexStatsModule **retrieveStatsModule()**

Gets the HashIndexStatsModule used to retrieve statistics associated with this index

Returns:

an HashIndexStatsModule for statistics associated with this index

See Also:

HashIndexStatsModule

getFindCount

double **getFindCount()**

Gets the index's find count attribute loaded up by the retrieveStatsModule() method.

Returns:

the find operation's invocation count for this index

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindCount(boolean copy)

getFindDurationTime

double **getFindDurationTime()**

Gets the index's find duration time attribute loaded up by the retrieveStatsModule() method.

Returns:

the find call's duration time for this index in milliseconds

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindDurationTime(boolean copy)

getFindResultCount

double **getFindResultCount()**

Gets the index's result count attribute loaded up by the retrieveStatsModule() method.

Returns:

the result count for this index and find operation

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindResultCount(boolean copy)

getFindFailureCount

double **getFindFailureCount()**

Gets the index's failure count attribute loaded up by the retrieveStatsModule() method.

Returns:

the failure count for this index and find operation

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindFailureCount(boolean copy)

getFindCollisionCount

double `getFindCollisionCount()`

Gets the index's collision count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the collision count for this index and find operation

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindCollisionCount(boolean copy)

getBatchUpdateCount

double `getBatchUpdateCount()`

Gets the index's batchupdate count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the doBatchUpdate method's invocation count for this index

See Also:

[retrieveStatsModule\(\)](#), MapIndexPlugin.doBatchUpdate(TxID txid, LogSequence sequence), HashIndexStatsModule.getBatchUpdateCount(boolean copy)

Overview	Package	Classes	TreeSerialized	Deprecated	IndexHelp	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All		
					Classes	
SUMMARY: NESTED FIELD CONSTR METH					DETAIL: FIELD CONSTR METHOD	

com.ibm.websphere.objectgrid.management

Interface DynamicServerMBean

All Superinterfaces:

[ServerMBean](#)

public interface
extends [ServerMBean](#)

This MBean interface allows a client process to access different attributes about a specific server process in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ObjectGridServer,name=<server>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

The following notifications are available:

Description All core group membership changes detected by the server's core group manager.

UserData: The number of members in the core group.

Message: The name of the core group.

Since: 6.1 FIX 3

Descripti on: All log messages detected by the log notification filter. See the [setLogNotificationFilter\(String\)](#) attribute.

UserData A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The log record severity level.	8.6
LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The originating log record class name.	8.6
SourceMethodName	String	The originating log record method name.	8.6
SequenceNumber	Long	The log record sequence number	8.6
ThreadID	Integer	The originating log record thread id.	8.6
ThrownMessage	String	The thrown exception message or empty string, if	8.6

g not available.

Message: The log message.

Since: 8.6

Description: All first-failure data captured by the grid server.

UserData A CompositeData with the following items:
:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
SourceID	String	The source code generating the exception.	8.6
ProbeID	String	The source code location associated with the exception captured.	8.6
ExceptionName	String	The exception captured.	8.6
Count	Integer	The current occurrence count for the specified exception.	8.6
DateOfFirstOccurrence	String	The exception's first occurrence time stamp.	8.6
Label	String	The label associated with the captured exception occurrence.	8.6

Message: A notification was generated on the server for a new exception.

Since: 8.6

Since:

WAS XD 6.1 FIX3, XC10

Field Summary	
s t a t i c S t r i n g	
s t a t i c S t r i n g	! "# \$%& # &&" "
s t a t i c S	"' "(% # ! "

t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

\$\$ # % " # ! !

s
t
a
t
i
c
S
t
r
i
n
g

\$\$ # % % ! %

s
t
a
t
i
c
S
t
r
i
n
g

) ! ")

s
t
a
t
i
c
S
t
r
i
n
g

) " * ") # ! ")

s
t
a
t
i
c
S
t
r
i
n
g

) + # " ! + # % % ! %

The constant identifying the notification type for log messages.

s
t
a
t
i
c
S
t
r
i
n
g

) + # & " & # ! !

i
n
q

s
t
a
t
i
c
S
t
r
i
n
g

) ++" &# ! "

s
t
a
t
i
c
S
t
r
i
n
g

(& "#%

s
t
a
t
i
c
S
t
r
i
n
g

". " "# "&

s
t
a
t
i
c
S
t
r
i
n
g

"&*" &# &"+& (# " "& -%# -! +"

A constant identifying the notification type for all core group membership changes detected by this server.

s
t
a
t
i
c
S
t
r
i
n
g

& "#)! # ! "

s
t
a
t
i
c
S
t
r
i
n
g

& "#%

g s t a t i c S t r i n g	<u>& " # " - # ! "</u>
s t a t i c S t r i n g	<u>- & ! # %</u>
s t a t i c S t r i n g	<u>- & . # " ! + "</u>

Method Summary

v o i d	<u>/ \$\$</u> () Triggers a simulated exception to be captured as a first-failure data capture (FFDC) event (and subsequently broadcasted as a JMX notification), as a means to test and verify the monitoring being enabled on the server.
v o i d	<u>/)</u> () Generates a set of log records with various severity levels, providing a simple means to test and verify when the JMX notification monitoring being enabled on the grid server.
i n t	<u>!</u> () Returns the number of available processors for the JVM hosting this server.
C o m p o s i t i o n S t r i n g	<u>"</u> % () Retrieve the environment information for the server (host name, WebSphere eXtreme Scale version, and additional information).
l o	<u>\$</u> () Returns the available memory in bytes for the JVM hosting this server.

ng

String

getHostName()
Returns the host name for this process.

String

getLogFilter()
Retrieves current regular expression filter being applied while screening log record messages before being broadcasted as JMX notifications by the grid server as a JMX Notification of type:

long

getMaxMemory()
Returns the maximum memory in bytes for the JVM hosting this server.

Iterable

getPrimaryShards()
Provides revisions for all primary shards in the container in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

Iterable

getShards()
Provides revisions for all shards in the server in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

boolean

hasReplicas()
Returns true if a replica exists for each primary hosted on this server.

String

getStatisticsSpec()
Retrieve the current statistics specification for the server.

long

getTotalMemory()
Returns the total memory in bytes for the JVM hosting this server.

String

getTraceSpec()
Retrieve the current trace specification for the server.

S t r i n g	<code>getZoneName()</code> Returns the zone name for this process
v o i d	<code>setRegexFilter(String regexFilter)</code> Sets the regular expression filter to be applied in screening log record messages before being broadcasted by the grid server as a JMX Notification of type:
v o i d	<code>setStatsSpec(String statsSpec)</code> Set the statistics specification for the server.
v o i d	<code>setTraceSpec(String traceSpec)</code> Set the trace specification for the server.

Methods inherited from interface
com.ibm.websphere.objectgrid.management.[ServerMBean](#)
[getServerName](#), [modifyServerTraceSpec](#), [stopServer](#)

Field Detail

SERVER_COREGROUP_MEMBERSHIP_CHANGE

```
static final String SERVER_COREGROUP_MEMBERSHIP_CHANGE = "&*"&#  &" +& (# " "& -%# -! +"
```

A constant identifying the notification type for all core group membership changes detected by this server.

Since:

6.1 FIX 3

See Also:

[Constant Field Values](#)

LOG_MESSAGE_NOTIFICATION

```
static final String LOG_MESSAGE_NOTIFICATION = "+# " !+"#  %$%! %
```

The constant identifying the notification type for log messages.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

FFDC_NOTIFICATION

```
static final String FFDC_NOTIFICATION = "$$ #  %$%! %
```

See Also:

[Constant Field Values](#)

LOG_RECORD_DATA

static final [String](#)) +#&" & # ! !

See Also:

[Constant Field Values](#)

THROWN_MESSAGE

static final [String](#) -& . # " !+"

See Also:

[Constant Field Values](#)

THREAD_ID

static final [String](#) -&"! #%

See Also:

[Constant Field Values](#)

SEQUENCE_NUMBER

static final [String](#) ", " "# "&

See Also:

[Constant Field Values](#)

SOURCE_METHOD_NAME

static final [String](#) & "# " - # ! "

See Also:

[Constant Field Values](#)

SOURCE_CLASS_NAME

static final [String](#) & "#)! # ! "

See Also:

[Constant Field Values](#)

LOGGER_NAME

static final [String](#)) ++" &# ! "

See Also:

[Constant Field Values](#)

LEVEL_NAME

static final [String](#))"*)# ! "

See Also:

[Constant Field Values](#)

FFDC_INCIDENT_DATA

static final [String](#) \$\$ # % " # ! !

See Also:

[Constant Field Values](#)

SOURCE_ID

static final [String](#) & "#%

See Also:

[Constant Field Values](#)

PROBE_ID

static final [String](#) (& "#%

See Also:

[Constant Field Values](#)

LABEL

static final [String](#))! ")

See Also:

[Constant Field Values](#)

DATE_OF_FIRST_OCCURRENCE

static final [String](#) ! "# \$%& # &&" "

See Also:

[Constant Field Values](#)

COUNT

static final [String](#)

See Also:

[Constant Field Values](#)

EXCEPTION_NAME

static final [String](#) " "(% # ! "

See Also:

[Constant Field Values](#)

Method Detail

getAvailableProcessors

int ! ()

Returns the number of available processors for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.availableProcessors\(\)](#)

getFreeMemory

long \$ ()

Returns the available memory in bytes for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.freeMemory\(\)](#)

getMaxMemory

long 0 ()

Returns the maximum memory in bytes for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.maxMemory\(\)](#)

getTotalMemory

long ()

Returns the total memory in bytes for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.totalMemory\(\)](#)

getHostName

[String](#) - ()

Returns the host name for this process.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[InetAddress.getHostName\(\)](#)

getZoneName

[String](#) 1 ()

Returns the zone name for this process

Returns:

the zone name that was included in the properties used to start the server or DefaultZone if no zone name was used

getSafeToShutdown

boolean ()

Returns true if a replica exists for each primary hosted on this server. Returns false if the server has the only copy of data.

Returns:

If server is safe to shutdown.

getStatsSpec

[String](#) ()

Retrieve the current statistics specification for the server.

Returns:

a string representation of the statistics specification.

Since:

7.1

See Also:

StatsSpec

setStatsSpec

void ([String](#) statsSpec)

Set the statistics specification for the server.

Parameters:

statsSpec - the statistics specification string.

Since:

7.1

See Also:

StatsSpec

getTraceSpec

[String](#) ()

Retrieve the current trace specification for the server.

Returns:

the trace specification string.

Since:

7.1

setTraceSpec

void ([String](#) traceSpec)

Set the trace specification for the server.

Parameters:

traceSpec - the statistics specification string.

Since:

7.1

See Also:

[ObjectGridManager.setTraceSpecification\(String\)](#)

getEnvironmentInfo

[CompositeData](#) " % ()

Retrieve the environment information for the server (host name, WebSphere eXtreme Scale version, and additional information). The CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
JMXServicePort	String	JMX Service Port
WASServerName	String	WebSphere Application Server Full Server Name
JVMVersion	String	JAVA Version
JMXConnectorPort	String	JMX Connector Port
IPAddress	String	IP Address
JavaVM	String	JVM Version
WASInstallRoot	String	WebSphere Application Server Product Directory
OSGiFrameworkVersion	String	OSGi Version
ClientPort	String	Client Port
HostName	String	Host name
Timestamp	String	Time Stamp from Server
WASBaseVersion	String	IBM WebSphere Application Server Version
OSName	String	Operating System
XSInstallRoot	String	WebSphere eXtreme Scale Product Directory
OSArch	String	OS Architecture
PeerPort	String	Peer Port
ServerType	String	Server Type
HAManagerPort	String	HAManager Port
JVMInstallPath	String	JAVA Directory
XC10Model	String	Machine Type and Model
JavaRuntimeInfo	String	JVM Runtime Version
xioContainerTCPNonSecure	String	XIO TCP/IP Port
JMXServiceURL	String	JMX Service Port
listenerPort	String	Listener Port (ORB)
ServerName	String	Server Name
WASXDVersion	String	IBM WebSphere Application Server - XD Version
WASExpressVersion	String	IBM WebSphere Application Server - ND Version
JavaBitMode	String	JAVA Bit Mode
XSVersion	String	WebSphere eXtreme Scale Version
JVMVendor	String	JAVA Vendor
OSVersion	String	Operating System Version
xioContainerTCPSecure	String	XIO TCP/IP SSL Port
PID	String	Process ID
WASNDVersion	String	IBM WebSphere Application Server - ND Version
ORB Version	String	ORB Version

Returns:

CompositeData containing the environment information

Since:
8.5, XC10 2.5

setLogNotificationFilter

void () \$ (String regexFilter)

Sets the regular expression filter to be applied in screening log record messages before being broadcasted by the grid server as a JMX Notification of type:

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The logging level.	8.6
LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The class name.	8.6
SourceMethodName	String	The method name.	8.6
SequenceNumber	Long	The sequence number	8.6
ThreadID	Integer	The thread id.	8.6
ThrownMessage	String	The thrown exception message or empty string, if not available.	8.6

Message: The log message.

Since: 8.6

Parameters:

regexFilter - The regular expression filter to applied in screening log record messages.

Since:
8.6, XC10 2.5

getLogNotificationFilter

String () \$ ()

Retrieves current regular expression filter being applied while screening log record messages before being broadcasted as JMX notifications by the grid server as a JMX Notification of type:

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The logging level.	8.6

LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The class name.	8.6
SourceMethodName	String	The method name.	8.6
SequenceNumber	Long	The sequence number	8.6
ThreadID	Integer	The thread id.	8.6
ThrownMessage	String	The thrown exception message or empty string, if not available.	8.6

Message: The log message.

Since: 8.6

Returns:

Current regular expression filter being applied in screening log record messages.

Since:

8.6, XC10 2.5

checkFFDCNotification

void /\$\$ ()

Triggers a simulated exception to be captured as a first-failure data capture (FFDC) event (and subsequently broadcasted as a JMX notification), as a means to test and verify the monitoring being enabled on the server. The JMX notification generated and broadcasted has the following type and content:

Description: All first-failure data captured by the grid server.

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
SourceID	String	The source code generating the exception.	8.6
ProbeID	String	The source code location associated with the exception captured.	8.6
ExceptionName	String	The exception captured.	8.6
Count	Integer	The current occurrence count for the specified exception.	8.6
DateOfFirstOccurrence	String	The exception's first occurrence time stamp.	8.6
Label	String	The label associated with the captured exception occurrence.	8.6

Message: A simulated first-failure data capture (FFDC) exception notification was generated by the server.

Since: 8.6

Since:

8.6, XC10 2.5

checkLoggingNotification

void /) ()

Generates a set of log records with various severity levels, providing a simple means to test and verify when the JMX notification monitoring being enabled on the grid server. Each of the JMX notifications generated and broadcasted has the following type and content:

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The logging level.	8.6
LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The class name.	8.6
SourceMethodName	String	The method name.	8.6
SequenceNumber	Long	The sequence number	8.6
ThreadID	Integer	The thread id.	8.6
ThrownMessage	String	A simulated log info/warning/error notification was generated by the server	8.6

Message: A simulated log info/warning/error notification was generated by the server.

Since: 8.6

Since:
8.6, XC10 2.5

getRevisions

[TabularData](#) & ()

Provides revisions for all shards in the server in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Grid	String	The object grid name.	8.6
MapSet	String	The name of the mapSet.	8.6
PartitionNumber	Integer	The number of a given partition.	8.6
ContainerName	String	Name of the container.	8.6
ShardType	String	Type of the shard.	8.6
RevisionState	TabularData	The lifetimeId / revisionNumber info for a given partition.	8.6

The RevisionState TabularData's CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LifetimeId	Long	The LifetimeId for the shard	8.6
Revision	Long	The revision number	8.6

Returns:

A TabularData object with the revision information.

Since:

8.6, XC10 2.5

getPrimaryRevisions

[TabularData](#) (&)

Provides revisions for all primary shards in the container in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Grid	String	The object grid name.	8.6
MapSet	String	The name of the mapSet.	8.6
PartitionNumber	Integer	The number of a given partition.	8.6
ContainerName	String	Name of the container.	8.6
ShardType	String	Type of the shard.	8.6
RevisionState	TabularData	The lifetimeId / revisionNumber info for a given partition.	8.6

The RevisionState TabularData's CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LifetimeId	Long	The LifetimeId for the shard	8.6
Revision	Long	The revision number	8.6

Returns:

A TabularData object with the revision information.

Since:

8.6, XC10 2.5

<p>Overview Package Classes Serialized Deprecated Index Help</p> <p>PREV CLASS NEXT CLASS</p> <p>SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD OD</p>	<p>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</p>
---	---

com.ibm.websphere.objectgrid.management
Interface ContainerMBean

public interface **ContainerMBean**

This MBean interface allows a client process to perform operations on and get status from an ObjectGrid container running in a dynamic environment. The object name pattern for this MBean is:

com.ibm.websphere.objectgrid:type=ObjectGridContainer,name=<server>,host=<host>,ogServerName=<server>

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary	
s t a t i c S t r i n g	<p>INVALID_PARTITION INVALID_PARTITION indicates that no partition was found for the requested shard.</p>
s t a t i c S t r i n g	<p>MAPSET_UNSUPPORTED_ON_CONTAINER MAPSET_UNSUPPORTED_ON_CONTAINER indicates that an attempt was made to reserve a shard from a map set that is not supported on this container.</p>
s t a t i c S t r i n g	<p>QUIESCE_COMPLETE QUIESCE_COMPLETE is the MBean notification type for a completed quiesce.</p>
s	

statistic

RELEASE_SUCCESSFUL

RELEASE_SUCCESSFUL indicates that the attempt to release the shard was successful.

statistic

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy.

statistic

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT indicates that the attempt to reserve the shard was processed successfully.

statistic

RESERVATION_SUCCESSFUL

RESERVATION_SUCCESSFUL indicates that the attempt to reserve the shard was successful.

statistic

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy.

statistic

SHARD_ALREADY_RESERVED

SHARD_ALREADY_RESERVED indicates that the shard is already reserved elsewhere and cannot be reserved on the specified container.

stat

t
i
c
s
t
r
i
n
g

[SHARD_NOT_RESERVED_ON_CONTAINER](#)

SHARD_NOT_RESERVED_ON_CONTAINER indicates that the attempt to release the shard from the requesting container failed because the specified shard was not found to be reserved by the requesting container.

Method Summary

i
n
t

[getActivatedShardCount\(\)](#)

Retrieve the total number of shards that have been activated for the life of this ObjectGrid container.

i
n
t

[getActiveShardCount\(\)](#)

Retrieve the number of active shards hosted in this ObjectGrid container.

S
t
r
i
n
g

[getContainerName\(\)](#)

Retrieve the name of the container.

i
n
t

[getDeactivatedShardCount\(\)](#)

Retrieve the total number of shards that have been deactivated for the life of this ObjectGrid container.

S
t
r
i
n
g

[getDomainName\(\)](#)

Retrieve the name of the catalog server grouping administering this container.

S
t
r
i
n
g

[getStatus\(\)](#)

Retrieve the status information for the shards in this container.

S
t
r
i
n
g

[getZoneName\(\)](#)

Retrieve the name of the zone grouping that this container belongs to.

i
n
t

[quiesceContainer\(Boolean inQuiesce\)](#)

Prepare the container for a potential shutdown by moving replica shards, verifying that primaries have required sync replicas and preventing the placement of new shards.

S
t
r
i
n
g

[release\(String objectGridName, String mapSetName, String partitionName\)](#)

Release a shard that has been previously reserved by this container.

S
t
r
i
n
g

[reserve\(String objectGridName, String mapSetName, String partitionName, String shardType\)](#)

Reserve a specific shard on this container.

S t r i n g	retrieveStatus (String objectGridName, String mapSetName) Retrieve the status information for the shards in this container, filtered by ObjectGrid and/or mapset.
v o i d	teardown () Tears down and stops the container in a way to allow partitions to be moved to new locations.
v o i d	terminate () Terminates a container without coordinating partition movement, partitions will failover.

Field Detail

QUIESCE_COMPLETE

static final [String](#) QUIESCE_COMPLETE

QUIESCE_COMPLETE is the MBean notification type for a completed quiesce.

See Also:

[quiesceContainer\(Boolean\)](#), [Constant Field Values](#)

RESERVATION_SUCCESSFUL

static final [String](#) RESERVATION_SUCCESSFUL

RESERVATION_SUCCESSFUL indicates that the attempt to reserve the shard was successful. The shard is reserved by the requesting container.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT

static final [String](#) RESERVATION_PRIOR_TO_INITIAL_PLACEMENT

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT indicates that the attempt to reserve the shard was processed successfully. However, since initial placement has not yet occurred, the reserved shard is not immediately moved to the requesting container. The shard will be placed on the container when initial placement is triggered.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

SHARD_ALREADY_RESERVED

static final [String](#) SHARD_ALREADY_RESERVED

SHARD_ALREADY_RESERVED indicates that the shard is already reserved elsewhere

and cannot be reserved on the specified container. The shard must be released from the owning container before it can be reserved again.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

INVALID_PARTITION

static final [String](#) INVALID_PARTITION

INVALID_PARTITION indicates that no partition was found for the requested shard.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER

static final [String](#) RESERVE_UNSUPPORTED_WITH_PER_CONTAINER

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy. Shard reservation is not supported with this placement strategy.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

RELEASE_SUCCESSFUL

static final [String](#) RELEASE_SUCCESSFUL

RELEASE_SUCCESSFUL indicates that the attempt to release the shard was successful. The shard is no longer reserved by this container. The shard is free to migrate, but it is not forced to migrate.

Since:

7.0.0.0 FIX1

See Also:

[release\(String, String, String\)](#), [Constant Field Values](#)

SHARD_NOT_RESERVED_ON_CONTAINER

static final [String](#) SHARD_NOT_RESERVED_ON_CONTAINER

SHARD_NOT_RESERVED_ON_CONTAINER indicates that the attempt to release the shard from the requesting container failed because the specified shard was not found to be reserved by the requesting container. Only the container owning the reservation may release a shard.

Since:

7.0.0.0 FIX1

See Also:

[release\(String, String, String\)](#), [Constant Field Values](#)

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER

static final [String](#) RELEASE_UNSUPPORTED_WITH_PER_CONTAINER

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy. Shard release is not supported with this placement strategy.

Since:

7.0.0.0 FIX1

See Also:

[release\(String, String, String\)](#), [Constant Field Values](#)

MAPSET_UNSUPPORTED_ON_CONTAINER

static final [String](#) MAPSET_UNSUPPORTED_ON_CONTAINER

MAPSET_UNSUPPORTED_ON_CONTAINER indicates that an attempt was made to reserve a shard from a map set that is not supported on this container. Only map sets that were included in the deployment policy at container initialization are supported to run on this container.

Since:

7.1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

Method Detail

teardown

void **teardown**()

Tears down and stops the container in a way to allow partitions to be moved to new locations.

terminate

void **terminate**()

Terminates a container without coordinating partition movement, partitions will failover.

getActiveShardCount

int **getActiveShardCount**()

Retrieve the number of active shards hosted in this ObjectGrid container.

Returns:

The current number of active shards.

getActivatedShardCount

int **getActivatedShardCount**()

Retrieve the total number of shards that have been activated for the life of this

ObjectGrid container.

Returns:

The number of activated shards.

getDeactivatedShardCount

int `getDeactivatedShardCount()`

Retrieve the total number of shards that have been deactivated for the life of this ObjectGrid container.

Returns:

The number of deactivated shards

getDomainName

[String](#) `getDomainName()`

Retrieve the name of the catalog server grouping administering this container.

Returns:

The domain name.

getZoneName

[String](#) `getZoneName()`

Retrieve the name of the zone grouping that this container belongs to.

Returns:

The name of the zone.

quiesceContainer

int `quiesceContainer(Boolean inQuiesce)`

Prepare the container for a potential shutdown by moving replica shards, verifying that primaries have required sync replicas and preventing the placement of new shards.

Parameters:

`inQuiesce` - Initiate quiesce mode (true) or cancel quiesce mode (false)

Returns:

The number of replicas moved off of the ObjectGrid container

See Also:

[QUIESCE_COMPLETE](#)

getStatus

[String](#) `getStatus()`

Retrieve the status information for the shards in this container.

Returns:

The status information for the shards in this container.

retrieveStatus

```
String retrieveStatus(String objectGridName,  
                    String mapSetName)
```

Retrieve the status information for the shards in this container, filtered by ObjectGrid and/or mapset. For example, calling `retrieveStatus` with "og1" and "ms1" as parameters will return the partition status for those partitions in ObjectGrid og1 and mapset ms1. Passing in an empty string ("") objectGridName or mapSetName will return all of the partitions, since the empty string acts as a wildcard. Passing in the empty string for both parameters will return the same status as calling `getStatus()`.

Parameters:

objectGridName - The name of the ObjectGrid for which the status is requested.
mapSetName - The name of the mapset within the ObjectGrid for which the status is requested.

Returns:

The status information for the shards in this container.

reserve

```
String reserve(String objectGridName,  
             String mapSetName,  
             String partitionName,  
             String shardType)
```

Reserve a specific shard on this container. Calling this method will cause the requested shard to move to this container. The shard can be moved to this container only if it is not reserved elsewhere. Calling this method prior to initial placement will pre-reserve the shard so that it will be placed onto this container when initial placement occurs. If non-reserved shard for the same partition is on this container prior to reservation, the non-reserved shard will be moved off the container upon reservation. A reserved shard will not be moved off of this container until it is released or the container is stopped.

Parameters:

objectGridName - the ObjectGrid containing the shard
mapSetName - the map set containing the shard
partitionName - the partition containing the shard
shardType - the type of shard. Currently, only primary shards can be reserved:
[ShardMBean.TYPE_PRIMARY](#)

Returns:

the return code indicating the result of the reserve request

Throws:

[IllegalArgumentException](#) - if any of the arguments are null or the empty String. Also thrown if shardType is not [ShardMBean.TYPE_PRIMARY](#)

Since:

7.0.0.0 FIX1

See Also:

[ShardMBean.TYPE_PRIMARY](#), [release\(String, String, String\)](#), [RESERVATION_SUCCESSFUL](#), [RESERVATION_PRIOR_TO_INITIAL_PLACEMENT](#), [SHARD_ALREADY_RESERVED](#), [INVALID_PARTITION](#), [RESERVE_UNSUPPORTED_WITH_PER_CONTAINER](#)

release

```
String release(String objectGridName,  
             String mapSetName,  
             String partitionName)
```

Release a shard that has been previously reserved by this container. This container can only release shards that it has reserved. Releasing the shard does not guarantee the

shard will be moved. The shard may remain on this container. However, it will not be explicitly bound to this container. Releasing a shard allows the shard to move freely to other containers or to be reserved by another container.

Parameters:

objectGridName - the ObjectGrid containing the shard

mapSetName - the map set containing the shard

partitionName - the partition containing the shard

Returns:

the return code indicating the result of the release request

Throws:

[IllegalArgumentExcpetion](#) - if any of the arguments are null or the empty String

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [RELEASE_SUCCESSFUL](#),
[SHARD_NOT_RESERVED_ON_CONTAINER](#), [RELEASE_UNSUPPORTED_WITH_PER_CONTAINER](#)

getContainerName

[String](#) getContainerName()

Retrieve the name of the container. The container name is based on the server name and includes a suffix which uniquely identifies the container within the server.

Returns:

the name of the container

Since:

7.1

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
ew	ge	ss	ed	ted			
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All		
			Classes				

SUMMARY: NESTED | [FIELD](#) | CONSTR | [METH](#) DETAIL: [FIELD](#) | CONSTR | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid.management

Interface CatalogServiceManagementMBean

public interface **CatalogServiceManagementMBean**

This MBean interface allows user to manipulate the behaviors of heartbeat and leader manager and other catalog service specific actions. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=CatalogService
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

7.1, XC10

Field Summary

s t a t i c	COREGROUP_MEMBERSHIP_CHANGE Constant representing a core group membership change notification.
S t r i n g	
s t a t i c	HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE Constant representing a heartbeat frequency level at an aggressive rate.
i n t	
s t a t i c	HEARTBEAT_FREQUENCY_LEVEL_RELAXED Constant representing a heartbeat frequency level at relaxed rate.
i n t	
s t a t i c	HEARTBEAT_FREQUENCY_LEVEL_TYPICAL Constant representing a heartbeat frequency level at a typical rate.

i n t	
s t a t i c S t r i n g	<u>MANAGEMENT_CONCENTRATOR_STATUS_STARTED</u>
s t a t i c S t r i n g	<u>MANAGEMENT_CONCENTRATOR_STATUS_STOPPED</u>
s t a t i c S t r i n g	<u>ORB</u> Constant representing the ORB or Object Request Broker transport
s t a t i c S t r i n g	<u>SERVER_EVENT_STARTED</u> Constant representing an eXtreme Scale server start notification.
s t a t i c S t r i n g	<u>SERVER_EVENT_STOPPED</u> Constant representing an eXtreme Scale server stop notification.
s t a t i c S t r i n g	<u>XIO</u> Constant representing the eXtremeIO transport

Method Summary

I
a
b
u
l
e
r
D
e
t
a
i
l

[getContainerReplicationState\(\)](#)

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross containers within one domain.

I
a
b
u
l
e
r
D
e
t
a
i
l

[getDomainReplicationState\(\)](#)

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross domains.

i
n
t

[getHeartBeatFrequencyLevel\(\)](#)

Retrieves the heartbeat frequency level.

S
t
r
i
n
g

[getManagementConcentratorStatus\(\)](#)

Returns the String representation of the management concentrator status.

i
n
t

[getNumberOfServers\(\)](#)

Retrieves the number eXtreme Scale servers that are currently registered with the catalog service.

C
o
m
p
o
s
i
t
e
D
e
t
a
i
l

[getServers\(\)](#)

Retrieves a CompositeData of each eXtreme Scale server that is currently registered with the catalog service.

S
t
r
i
n
g

[getTransport\(\)](#)

Returns the transport used by the catalog service domain.

b
o
o
l
e
a
n

[isPrimary\(\)](#)

Provides indication if the catalog is primary.

v o i d	LogMessage (String level, String message) Provides support for logging user messages from processes outside the catalog and/or container servers.
v o i d	startManagementConcentrator () Starts the Management Concentrator.
v o i d	stopManagementConcentrator () Stops the Management Concentrator.

Field Detail

MANAGEMENT_CONCENTRATOR_STATUS_STARTED

static final [String](#) MANAGEMENT_CONCENTRATOR_STATUS_STARTED

See Also:

[Constant Field Values](#)

MANAGEMENT_CONCENTRATOR_STATUS_STOPPED

static final [String](#) MANAGEMENT_CONCENTRATOR_STATUS_STOPPED

See Also:

[Constant Field Values](#)

COREGROUP_MEMBERSHIP_CHANGE

static final [String](#) COREGROUP_MEMBERSHIP_CHANGE

Constant representing a core group membership change notification. The user data associated with this notification is a CompositeData.

The CompositeData includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

MemberName	String	The name of the server that is included in the core group.
------------	--------	--

See Also:

[Constant Field Values](#)

SERVER_EVENT_STARTED

static final [String](#) SERVER_EVENT_STARTED

Constant representing an eXtreme Scale server start notification.

The UserData argument of the Notification includes a TabularData that includes information for each of the servers. Each CompositeData (row in the TabularData) contains the following items:

<u>Description</u>

<u>Item Name</u>	<u>Type</u>	
HAPort	String	The port number of the high availability manager.
Host	String	The host/ip address of the server.
JMXServiceURL	String	The JMX service url used to access the server.
ServerName	String	The name of the server.
ZoneName	String	The name of the zone that the server belongs.

See Also:

[Constant Field Values](#)

SERVER_EVENT_STOPPED

```
static final String SERVER_EVENT_STOPPED
```

Constant representing an eXtreme Scale server stop notification.

The UserData argument of the Notification includes a TabularData instance where each CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

ServerName	String	The name of the server.
------------	--------	-------------------------

See Also:

[Constant Field Values](#)

HEARTBEAT_FREQUENCY_LEVEL_TYPICAL

```
static final int HEARTBEAT_FREQUENCY_LEVEL_TYPICAL
```

Constant representing a heartbeat frequency level at a typical rate.

A typical heartbeat frequency allows reasonable failover detection and resource utilization. This value is the default.

See Also:

[Constant Field Values](#)

HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE

```
static final int HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE
```

Constant representing a heartbeat frequency level at an aggressive rate.

An increased heartbeat frequency allows failures to be detected more quickly, but can also use additional CPU and network resources. This level is more sensitive to missing heartbeats when the server is stressed.

See Also:

[Constant Field Values](#)

HEARTBEAT_FREQUENCY_LEVEL_RELAXED

```
static final int HEARTBEAT_FREQUENCY_LEVEL_RELAXED
```

Constant representing a heartbeat frequency level at relaxed rate.

A decreased heartbeat frequency increases the time to detect failures, but also decreases

CPU and network utilization.

See Also:

[Constant Field Values](#)

ORB

static final [String](#) ORB

Constant representing the ORB or Object Request Broker transport

See Also:

[Constant Field Values](#)

XIO

static final [String](#) XIO

Constant representing the eXtremeIO transport

See Also:

[Constant Field Values](#)

Method Detail

getHeartBeatFrequencyLevel

int `getHeartBeatFrequencyLevel()`

Retrieves the heartbeat frequency level.

Valid values include:

- [HEARTBEAT_FREQUENCY_LEVEL_TYPICAL](#)
- [HEARTBEAT_FREQUENCY_LEVEL_RELAXED](#)
- [HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE](#)

Returns:

the heartbeat frequency level: -1, 0 or 1 as defined by the constants that begin with name HEARTBEAT_FREQUENCY_LEVEL.

getServers

[CompositeData](#) `getServers()`

Retrieves a CompositeData of each eXtreme Scale server that is currently registered with the catalog service.

The CompositeData includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

serverName	String	The name of the server that is registered with the catalog service.
------------	--------	---

Returns:

the CompositeData representing the currently registered eXtreme Scale servers.

getNumberOfServers

int **getNumberOfServers()**

Retrieves the number eXtreme Scale servers that are currently registered with the catalog service.

Returns:

the number of registered eXtreme Scale servers.

logMessage

void **logMessage**([String](#) level,
[String](#) message)

Provides support for logging user messages from processes outside the catalog and/or container servers. Example: XC10 surfaced SNMP trap messages can be flowed from the SNMP agent which throws traps in the console server (sMash) process, not typically an XS catalog/container server.

Parameters:

level - name describing the severity of the event which is compatible with `java.util.logging.Level.parse(String name)` where name may be either level name (ex. "SEVERE") or an integer value (ex. "1000") - @see [Level.parse\(String\)](#)
message - for the end user (already sNLS rendered)

Since:

8.6, XC10 2.5

isPrimary

boolean **isPrimary()**

Provides indication if the catalog is primary.

Returns:

true for primary catalog.

Since:

8.6, XC10 2.5

getContainerReplicationState

[TabularData](#) **getContainerReplicationState()**

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross containers within one domain. For a given container, outstanding out-bound revisions need to be replicated from primary shards located in this container into replicas located in other containers. In similar way, outstanding in-bound revisions need to be replicated from primary shards located in other containers into corresponding replicas located in this container. This operation can be used to check the differences in data revisions between containers within one domain.

The result is a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Container	String	The container ID.	8.6
OutboundRevisions	Long	Number of out-bound revisions for container.	8.6
InboundRevisions	Long	Number of in-bound revisions for container.	8.6

Returns:

A TabularData object with the container replication state information.

Since:

8.6, XC10 2.5

getDomainReplicationState

[TabularData](#) `getDomainReplicationState()`

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross domains. Outstanding out-bound revisions need to be replicated from local primary shards into corresponding remote primary shards. Outstanding in-bound revisions need to be replicated from remote primary shards into corresponding local primary shards. This operation can be used to check the differences in data revisions between different domains linked by MMR replication.

The result is a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Domain	String	The domain name.	8.6
Container	String	The container ID.	8.6
OutboundRevisions	Long	Number of out-bound revisions for container.	8.6
InboundRevisions	Long	Number of in-bound revisions for container.	8.6

Returns:

A TabularData object with the domain replication state information.

Since:

8.6, XC10 2.5

getTransport

[String](#) `getTransport()`

Returns the transport used by the catalog service domain.

Returns:

String containing the transport type

Since:

8.6, XC10 2.5

See Also:

[ORB](#), [XIO](#)

startManagementConcentrator

`void startManagementConcentrator()`

Starts the Management Concentrator. The catalog server will now start listening for log messages.

Since:

8.6.0.2, XC10 2.5

stopManagementConcentrator

`void stopManagementConcentrator()`

Stops the Management Concentrator. The catalog server will no longer listen for log

messages. Stopping the Management Concentrator will also stop the Message Center in the web monitoring console.

Since:

8.6.0.2, XC10 2.5

getManagementConcentratorStatus

[String](#) getManagementConcentratorStatus()

Returns the String representation of the management concentrator status. Status will be either CatalogServiceManagementMBean.MANAGEMENT_CONCENTRATOR_STATUS_STARTED or CatalogServiceManagementMBean.MANAGEMENT_CONCENTRATOR_STATUS_STOPPED.

Returns:

String containing the status

Since:

8.6.0.2, XC10 2.5

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management
Interface AgentManagerMBean

public interface **AgentManagerMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific Agent on a server process. The Agent Manager MBean is scoped at the map level and therefore can access statistical data for every agent run against the specified map. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=AgentManager,name=Agent-<map>,partition=<partition id>,objectgrid=<objectgrid>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:
 WAS XD 6.1.0.5, XC10

Method Summary	
d o u b l e	<p>getAgentInflationTime(String agentClassName) Gets the specified agent's inflation time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getAgentSerializationTime(String agentClassName) Gets the specified agent's serialization time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getFailureCount(String agentClassName) Gets the specified agent's failure count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getInvocationCount(String agentClassName) Gets the specified agent's invocation count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getPartitionCount(String agentClassName) Gets the specified agent's partition count attribute loaded up by the retrieveStatsModule() method.</p>

d
o
u
b
l
e

[getReduceTime](#)(String agentClassName)

Gets the specified agent's total reduce time attribute loaded up by the retrieveStatsModule() method.

d
o
u
b
l
e

[getResultInflationTime](#)(String agentClassName)

Gets the specified agent's result inflation time attribute loaded up by the retrieveStatsModule() method.

d
o
u
b
l
e

[getResultSerializationTime](#)(String agentClassName)

Gets the specified agent's result serialization time attribute loaded up by the retrieveStatsModule() method.

d
o
u
b
l
e

[getTotalDurationTime](#)(String agentClassName)

Gets the specified agent's total run time attribute loaded up by the retrieveStatsModule() method.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
A
g
e
n
t
S
t
a
t
s
M
o

[retrieveStatsModule](#)(String agentClassName)

Gets the AgentStatsModule used to retrieve statistics associated with the specified agent class

Method Detail

getReduceTime

double **getReduceTime**([String](#) agentClassName)

Gets the specified agent's total reduce time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the reduce time for this agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `AgentStatsModule.getReduceTime(boolean copy)`

getTotalDurationTime

double **getTotalDurationTime**([String](#) agentClassName)

Gets the specified agent's total run time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the total run time for this agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `AgentStatsModule.getTotalDurationTime(boolean copy)`

getAgentSerializationTime

double **getAgentSerializationTime**([String](#) agentClassName)

Gets the specified agent's serialization time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to serialize the agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `AgentStatsModule.getAgentSerializationTime(boolean copy)`

getAgentInflationTime

double **getAgentInflationTime**([String](#) agentClassName)

Gets the specified agent's inflation time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to inflate the agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getAgentInflationTime(boolean copy)

getResultInflationTime

double **getResultInflationTime**([String](#) agentClassName)

Gets the specified agent's result inflation time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to inflate the agent results for a given partition in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getResultInflationTime(boolean copy)

getResultSerializationTime

double **getResultSerializationTime**([String](#) agentClassName)

Gets the specified agent's result serialization time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to serialize the agent results for a given partition in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getResultSerializationTime(boolean copy)

getPartitionCount

double **getPartitionCount**([String](#) agentClassName)

Gets the specified agent's partition count attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the number of partitions this agent is sent to

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getPartitionCount(boolean copy)

getFailureCount

double **getFailureCount**([String](#) agentClassName)

Gets the specified agent's failure count attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the failure count for the specified agent

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getFailureCount(boolean copy)

getInvocationCount

double **getInvocationCount**([String](#) agentClassName)

Gets the specified agent's invocation count attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the invocation count for the specified agent

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getInvocationCount(boolean copy)

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.AgentStatsModule **retrieveStatsModule**([String](#) agentClassName)

Gets the AgentStatsModule used to retrieve statistics associated with the specified agent class

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

an AgentStatsModule for statistics associated with the specified agent class

See Also:

AgentStatsModule

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

PREV CLASS [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package **com.ibm.websphere.objectgrid.plugins**

These are the interfaces for adding plugins to the Grid core framework.

See:

[Description](#)

Interface Summary	
BeanFactory	Implement this interface to allow bean factories like Spring or Google guice to be integrated.
CacheEntry	This interface represents a cache entry in an ObjectGrid map.
EvictionEventCallback	An instance of EvictionEventCallback is passed into the Evictor at initialization time.
Evictor	Data contained in a BackingMap are evicted when the map is full.
EvictorData	This interface is optionally used by an implementator of the Evictor interface.
LogElement	LogElements are the individual entries within a LogSequence.
LogSequence	LogSequence is the ordered list of changes performed against a given map for a given transaction.
LogSequenceFilter	This interface can be used to filter a LogSequence.
MapEventListener	This callback interface is implemented by the application when it wants to receive events about a Map such as the eviction of a map entry.
ObjectGridEventGroup	This is a set of single method interfaces for fine grained events delivered for an ObjectGrid.
ObjectGridEventGroup.ShardEvents	These events are fired when a shard is made a primary shard and when the shard is demoted from a primary.
ObjectGridEventGroup.ShardLifecycle	These events are fired when an ObjectGrid shard is initialized and destroyed.
ObjectGridEventGroup.TransactionEvents	These events are called every single transaction.
ObjectGridEventListener	This interface is used to create an implementation of an event listener for an ObjectGrid.
ReplicationMapListener	Deprecated. <i>The client replicated map function is deprecated in version 8.6.</i>
Transaction	Calling methods on a Session will send corresponding events to the

Callback	TransactionCallback.
TransactionCallback.BeforeCommit	The BeforeCommit optional mix-in interface for the TransactionCallback plugin interface allows plug-ins to be notified at the beginning of a Session.commit() .
TransactionCallback.BeforeCommit.TransactionContext	The TransactionContext identifies various information that's available to the beforeCommit() method.
ValueProxyInfo	This interface can be used by value objects to inform a BackingMap or Loader which attribute(s) have been dirtied.

Class Summary	
EvictorData.SpecialEvictorData	Special value class used for representing the key not being found in the BackingMap.
LogElement.Type	The Type class is used to represent a LogElement type.

Exception Summary	
CacheEntryException	This exception indicates an error occurred during a cache entry operation.
LoaderException	This exception is the base exception for any exceptions encountered by a Loader.
TransactionCallbackException	This exception is thrown when a method call to TransactionCallback fails.

Package **com.ibm.websphere.objectgrid.plugins** Description

These are the interfaces for adding plugins to the Grid core framework.

Overview

These plugins can be added into ObjectGrid in several ways such as xml configuration, programmatically adding, or using annotation.

Annotation based callbacks

ObjectGrid when running on Java 5 will start to use an annotated method callback system. This means that objects can be registered as callbacks or listeners. The methods on the object must be annotated as to be invoked for a certain event. Unannotated methods are not invoked. The name of the method is unimportant. The method arguments and return type must be the same as expected for the callback method.

Why?

Usually, callbacks are specified using an interface. This works well but results in a possible performance loss as all methods on the interface will be invoked by the ObjectGrid even

though the application is only interested in a single event. This wastes precious resources. Another issue is when we need to add a new event. Adding a new method to an existing interface breaks back wards compatibility. We can make a new interface extending the old one with the new methods but this is also undesirable as soon there are many interfaces in the hierarchy as new events are added. The annotation system allows the application to only mark methods to be called avoiding the first problem and if new event types are added they have no impact on existing callback objects. Newer applications can add a method and annotate it with the new event annotation to receive the event.

com.ibm.websphere.objectgrid.plugins

Interface ValueProxyInfo

public interface ValueProxyInfo

This interface can be used by value objects to inform a BackingMap or Loader which attribute(s) have been dirtied. This mechanism allows the BackingMap and Loader to interrogate the set of changed attributes in the value object instead of just assuming the whole value object has been updated. For this to be useful, the application must only use the getter and setter methods defined for the value object's interface.

Since:

WAS XD 6.0, XC10

See Also:

[CopyMode.COPY_ON_WRITE](#), [BackingMap.setCopyMode\(CopyMode, Class\)](#), [Loader.batchUpdate\(TxID, LogSequence\)](#)

Method Summary	
V o i d	<p>ibmClearDirtyAttributes() Clears the list of dirty attributes.</p>
L i s t	<p>ibmGetDirtyAttributes() Returns a list of dirty attributes based on the value interface set on the map.</p>
O b j e c t	<p>ibmGetRealValue() Returns the real value object this proxy represents.</p>

Method Detail

ibmGetDirtyAttributes

[List](#) [ibmGetDirtyAttributes\(\)](#)

Returns a list of dirty attributes based on the value interface set on the map.

The attribute name is always starts with an upper case letter. For example, if the setter for the attribute is setPrice then 'Price' is the string returned here. The runtime uses substring(3) of the setter method name as the attribute name.

Returns:

List of attribute names (Strings)

ibmGetRealValue

[Object](#) `ibmGetRealValue()`

Returns the real value object this proxy represents.

Needed internally by the BackingMap to return a separate proxy for each transaction.

Returns:

actual value object.

ibmClearDirtyAttributes

`void ibmClearDirtyAttributes()`

Clears the list of dirty attributes.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Class TransactionCallbackException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[com.ibm.websphere.objectgrid.ClientServerTransactionCallbackException](#),
[ReplicationVotedToRollbackTransactionException](#)

```
public class TransactionCallbackException
extends ObjectGridException
```

This exception is thrown when a method call to TransactionCallback fails.

Since:

WAS XD 6.0, XC10

See Also:

[TransactionCallback](#), [Serialized Form](#)

Constructor Summary

[TransactionCallbackException](#)()

Constructs a new TransactionCallbackException with null as its detail message.

[TransactionCallbackException](#)([String](#) message)

Constructs a new TransactionCallbackException with the specified detail message.

[TransactionCallbackException](#)([String](#) message, [Throwable](#) cause)

Constructs a new TransactionCallbackException with the specified detail message and cause.

[TransactionCallbackException](#)([Throwable](#) cause)

Constructs a new TransactionCallbackException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#),
[printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionCallbackException

```
public TransactionCallbackException()
```

Constructs a new `TransactionCallbackException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionCallbackException

```
public TransactionCallbackException(String message)
```

Constructs a new `TransactionCallbackException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TransactionCallbackException

```
public TransactionCallbackException(String message,  
                                   Throwable cause)
```

Constructs a new `TransactionCallbackException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `TransactionCallbackException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

TransactionCallbackException

```
public TransactionCallbackException(Throwable cause)
```

Constructs a new `TransactionCallbackException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains

the class and detail message of cause). This constructor is useful for TransactionCallbackExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface TransactionCallback

All Known Subinterfaces:

[TransactionCallback.BeforeCommit](#)

All Known Implementing Classes:

[WebSphereTransactionCallback](#)

```
public interface TransactionCallback
```

Calling methods on a `Session` will send corresponding events to the `TransactionCallback`. An `ObjectGrid` can have zero or one `TransactionCallback`. `BackingMaps` defined on an `ObjectGrid` with a `TransactionCallback` should have corresponding `Loaders`.

A `TransactionCallback` works with `Loaders` and place transaction specific objects in slots on the `TxID` object that `Loaders` can obtain. Examples are database connections, prepared statement caches, etc. The `TransactionCallback` should reserve slots in the `TxID` by calling `ObjectGrid.reserveSlot(String)` using the name `TxID.SLOT_NAME`. The `TransactionCallback` can then put an object at that index in the `TxID`. A `Loader` can retrieve the index used by the `TransactionCallback` by calling an internal method on the `TransactionCallback`'s implementation. A reference to the configured `TransactionCallback` can be found using the `TxID.getSession().getObjectGrid().getTransactionCallback()` code sequence.

A `TransactionCallback` implementation that also implements the `ObjectGridLifecycleListener` interface will be automatically added as an `EventListener` on the [ObjectGrid](#) when the callback is set on the object grid.

A `TransactionCallback` may implement the `ObjectGridPlugin` interface in order to receive enhanced `ObjectGrid` plug-in lifecycle method calls. The plug-in is also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Since:

WAS XD 6.0, XC10

See Also:

`Loader`, [ObjectGrid.addEventListener\(EventListener\)](#), [ObjectGrid.getTransactionCallback\(\)](#), [ObjectGrid.reserveSlot\(String\)](#), [ObjectGrid.setTransactionCallback\(TransactionCallback\)](#), [Session.getObjectGrid\(\)](#), [TxID.putSlot\(int, Object\)](#), [TxID.getSlot\(int\)](#), [TxID.getSession\(\)](#)

Nested Class Summary

s
t
a
t
i
c

i
n
t
e
r

[TransactionCallback.BeforeCommit](#)

The `BeforeCommit` optional mix-in interface for the `TransactionCallback` plug-in interface allows plug-ins to be notified at the beginning of a [Session.commit\(\)](#).

Method Summary

void [begin](#)([TxID](#) id)
Invoked when starting a Session transaction.

void [commit](#)([TxID](#) id)
Invoked when committing a Session transaction.

void [initialize](#)([ObjectGrid](#) objectGrid)
Invoked when an ObjectGrid is initialized.

boolean [isExternalTransactionActive](#)([Session](#) session)
Called when an application attempts to use a Session with no transaction active.

void [rollback](#)([TxID](#) id)
Invoked when rolling back a Session transaction.

Method Detail

initialize

void [initialize](#)([ObjectGrid](#) objectGrid)
throws [TransactionCallbackException](#)

Invoked when an ObjectGrid is initialized.

This method is called so this object can do any implementation specific initialization.

Parameters:

objectGrid - A reference to the ObjectGrid.

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[ObjectGrid.reserveSlot\(String\)](#)

begin

void [begin](#)([TxID](#) id)
throws [TransactionCallbackException](#)

Invoked when starting a Session transaction.

A TransactionCallback can communicate the begin processing (along with the TxID) to the appropriate BackingMap and/or Loader. The Loader may use this signal to start a

corresponding transaction on the underlying connection to a database.

Parameters:

id - transaction identifier (TxID)

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#), [TxID](#)

commit

```
void commit(TxID id)
    throws TransactionCallbackException
```

Invoked when committing a Session transaction.

This method should be used to commit any underlying transaction and return any underlying connection back to the pool. The TxID is provided to determine which transaction is being committed

Parameters:

id - transaction identifier (TxID)

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[begin\(TxID\)](#), [Session.commit\(\)](#), [TxID](#)

rollback

```
void rollback(TxID id)
    throws TransactionCallbackException
```

Invoked when rolling back a Session transaction.

This method should be used to roll back any underlying transaction and return any underlying connection back to the pool. The TxID is provided to determine which transaction is being committed

Parameters:

id - transaction identifier (TxID)

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[begin\(TxID\)](#), [Session.rollback\(\)](#), [TxID](#)

isExternalTransactionActive

```
boolean isExternalTransactionActive(Session session)
```

Called when an application attempts to use a Session with no transaction active.

The callback could return true in which case an auto `Session.begin()` is executed. If false is returned, an application exception is thrown indicating no transaction is active. This event is usually used when integrating with a J2EE environment such as WebSphere Application Server.

Parameters:

session - the session which the application is using

Returns:

true if an auto begin should be done, false if this is not the case

See Also:

[Session](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface TransactionCallback.BeforeCommit

All Superinterfaces:

[TransactionCallback](#)

Enclosing interface:

[TransactionCallback](#)

```
public static interface TransactionCallback.BeforeCommit
extends TransactionCallback
```

The BeforeCommit optional mix-in interface for the TransactionCallback plug-in interface allows plug-ins to be notified at the beginning of a [Session.commit\(\)](#). Implementations can use the beforeCommit() method to validate changed data in the transaction and modify the data.

Since:

7.1.1

Nested Class Summary

s
t
a
t
i
c

i
n
t
e
r
f
a
c
e

[TransactionCallback.BeforeCommit.TransactionContext](#)

The TransactionContext identifies various information that's available to the beforeCommit() method.

Nested classes/interfaces inherited from interface

com.ibm.websphere.objectgrid.plugins.[TransactionCallback](#)

[TransactionCallback.BeforeCommit](#)

Method Summary

v
o
i
d

[beforeCommit](#)([TransactionCallback.BeforeCommit.TransactionContext](#) ctx)

Invoked at the beginning of a [Session.commit\(\)](#).

Methods inherited from interface

com.ibm.websphere.objectgrid.plugins.[TransactionCallback](#)

[begin](#), [commit](#), [initialize](#), [isExternalTransactionActive](#), [rollback](#)

Method Detail

beforeCommit

void **beforeCommit**([TransactionCallback.BeforeCommit.TransactionContext](#) ctx)
throws [TransactionCallbackException](#)

Invoked at the beginning of a `Session.commit()`.

Use the `TransactionContext.getLogSequences()` method to retrieve the changes made by this transaction. Use the `TransactionContext.getTxId().getSession()` methods to access the `Session`. The `Session` can be used to access `ObjectMaps` and modify data in the current transaction.

Parameters:

ctx - the context of the transaction.

Throws:

[TransactionCallbackException](#) - if an error occurs during processing. Any exception will roll back the transaction and will be included in the `TransactionException` thrown to the caller.

See Also:

[Session.commit\(\)](#), [TxID](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Interface

TransactionCallback.BeforeCommit.TransactionContext

Enclosing interface:

[TransactionCallback.BeforeCommit](#)

```
public static interface TransactionCallback.BeforeCommit.TransactionContext
```

The TransactionContext identifies various information that's available to the beforeCommit() method.

Since:

7.1.1

Method Summary

C	
o	
j	
r	
e	
c	
i	
t	
i	
j	
o	
g	
<	
T	getLogSequences() The LogSequences that reflect the pending changes to the map.
o	
j	
r	
e	
c	
i	
t	
i	
j	
o	
g	
>	
T	getTxID() Retrieve the TxID for the transaction.
X	
I	
D	

Method Detail

getTxID

[TxID](#) [getTxID\(\)](#)

Retrieve the TxID for the transaction.

Returns:
the TxID for the transaction.

getLogSequences

[Collection<LogSequence>](#) `getLogSequences()`

The LogSequences that reflect the pending changes to the map.

Returns:
the LogSequences.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ReplicationMapListener

Deprecated. *The client replicated map function is deprecated in version 8.6. Use the [ContinuousQueryManager](#) function.*

```
public interface ReplicationMapListener
```

This interface is used to create an implementation of an event listener for client-side maps that are in replication mode. Registered listeners receive notification callbacks for replication start and exit events and data changes.

Listener instances can be registered with a map using the [ClientReplicableMap.enableClientReplication\(com.ibm.websphere.objectgrid.ClientReplicableMap.Mode, int\[\], ReplicationMapListener\)](#) method.

Since:

WAS XD 6.1, XC10

Method Summary

v o i d	onData (LogSequence logSequence) Deprecated. This method is invoked when a data change is replicated to the client replication map.
v o i d	replicationExits () Deprecated. This method is invoked when replication has been disabled for this map.
v o i d	replicationStarts () Deprecated. This method is invoked when a snapshot mode replica has been synchronized with the client or a continuous replica has started replicating.

Method Detail

replicationStarts

```
void replicationStarts()
```

Deprecated.

This method is invoked when a snapshot mode replica has been synchronized with the client or a continuous replica has started replicating.

See also:

[ClientReplicableMap.enableClientReplication\(com.ibm.websphere.objectgrid.ClientReplicableMap.Mode, int\[\], ReplicationMapListener\)](#)

onData

void **onData**([LogSequence](#) logSequence)

Deprecated.

This method is invoked when a data change is replicated to the client replication map.

Parameters:

logSequence - the log sequence containing all of the data changes.

replicationExits

void **replicationExits**()

Deprecated.

This method is invoked when replication has been disabled for this map.

See Also:

[ClientReplicableMap.disableClientReplication\(\)](#)

Overvi	Packa	Cla	TreeSerializ	Depreca	IndexHelp	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
ew	ge	ss	ed	ted		
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All	
			Classes			

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventListener

All Superinterfaces:

com.ibm.websphere.objectgrid.plugins.EventListener

```
public interface ObjectGridEventListener  
extends com.ibm.websphere.objectgrid.plugins.EventListener
```

This interface is used to create an implementation of an event listener for an ObjectGrid. Instances of ObjectGridEventListeners are set on the ObjectGrid interface. Any significant events are communicated to the application using the methods outlined below. When using Java 5, this callback also supports new callback annotation mechanism.

Since:

WAS XD 6.0, XC10

See Also:

[ObjectGrid.addEventListener\(EventListener\)](#), [ObjectGrid.removeEventListener\(EventListener\)](#), [EventListener](#)

Method Summary

v o i d	destroy() Called when the ObjectGrid associated with this listener is destroyed.
v o i d	initialize(Session session) Invoked when an ObjectGrid is initialized.
v o i d	transactionBegin(String txid, boolean isWriteThroughEnabled) Signals the beginning of a Session transaction.
v o i d	transactionEnd(String txid, boolean isWriteThroughEnabled, boolean committed, Collection changes) Signals the ending of a Session transaction.

Method Detail

initialize

```
void initialize(Session session)
```

Invoked when an ObjectGrid is initialized.

A usable Session instance is passed to this listener to provide all of the necessary access

to the various ObjectGrid objects.

Parameters:

session - a Session instance that this listener is associated with.

See Also:

[ObjectGrid.initialize\(\)](#)

transactionBegin

```
void transactionBegin(String txid,  
                    boolean isWriteThroughEnabled)
```

Signals the beginning of a Session transaction.

A stringified version of the TxID is provided for correlating with the end of the transaction, if so desired. The type of transaction is also provided by the isWriteThroughEnabled boolean parameter.

Parameters:

txid - Stringified version of the TxID

isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the Session.beginNoWriteThrough(). method. false is passed if beginNoWriteThrough() was used.

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#)

transactionEnd

```
void transactionEnd(String txid,  
                  boolean isWriteThroughEnabled,  
                  boolean committed,  
                  Collection changes)
```

Signals the ending of a Session transaction.

A string version of the TxID is provided for correlating with the begin of the transaction, if so desired. Map changes are also reported with the collection of LogSequences passed to this method. Typical uses of this event are for customers doing custom peer invalidation or peer commit push. This event listener gives them the changes. Calls to this method are made after commit and are sequenced so that they are delivered one by one, not in parallel. The event order is the commit and rollback order.

For an ObjectGridEventListener receiving changes in an [ObjectMap](#) that is configured to use a OutputFormat.RAW for the keys or values, the keys and values objects in the LogSequences will be SerializedKey or SerializedValue objects respectively. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key or value object.

To override the map's output format configuration, use the PluginOutputFormat annotation in the implementation class.

Parameters:

txid - string version of the TxID

isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the Session.beginNoWriteThrough(). method. false is passed if beginNoWriteThrough() was used.

committed - a boolean flag indicating whether the transaction was committed (true) or rolled back (false)

changes - a Collection of LogSequences representing the changes that were committed or rolled back.

See Also:

[LogSequence.isRollback\(\)](#), [Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#),
[Session.commit\(\)](#), [Session.rollback\(\)](#)

destroy

void **destroy**()

Called when the ObjectGrid associated with this listener is destroyed.

This method is the opposite of the initialize method. When it is called, the listener can free up any resources it uses.

See Also:

[ObjectGrid.destroy\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup

public interface **ObjectGridEventGroup**

This is a set of single method interfaces for fine grained events delivered for an ObjectGrid. Classes implementing these interfaces AND ObjectGridEventListener can receive these events. If an ObjectGridEventListener implements ANY of these interfaces that only the specific methods on the interfaces implemented will be called.

Since:

WAS XD 6.1, XC10

See Also:

[ObjectGridEventListener](#)

Nested Class Summary

s
t
a
t
i
c

i
n
t
e
r
f
a
c
e

s
t
a
t
i
c

i
n
t
e
r
f
a
c
e

s
t
a
t
i
c

i
n
t

[ObjectGridEventGroup.ShardEvents](#)

These events are fired when a shard is made a primary shard and when the shard is demoted from a primary.

[ObjectGridEventGroup.ShardLifecycle](#)

These events are fired when an ObjectGrid shard is initialized and destroyed.

[ObjectGridEventGroup.TransactionEvents](#)

These events are called every single transaction.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
OD

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup.ShardLifecycle

Enclosing interface:

[ObjectGridEventGroup](#)

```
public static interface ObjectGridEventGroup.ShardLifecycle
```

These events are fired when an ObjectGrid shard is initialized and destroyed. A shard can be activated/deactivated multiple times within these two events.

Method Summary

[void](#) [destroy](#)()

Called when the ObjectGrid associated with this listener is destroyed.

[void](#) [initialize](#)([Session](#) session)

Invoked when an ObjectGrid is initialized.

Method Detail

initialize

```
void initialize(Session session)
```

Invoked when an ObjectGrid is initialized.

A usable Session instance is passed to this listener to provide all of the necessary access to the various ObjectGrid objects.

Parameters:

session - a Session instance that this listener is associated with.

See Also:

[ObjectGrid.initialize\(\)](#)

destroy

```
void destroy()
```

Called when the ObjectGrid associated with this listener is destroyed.

This method is the opposite of the initialize method. When it is called, the listener can free up any resources it uses.

See Also:

[ObjectGrid.destroy\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup.ShardEvents

Enclosing interface:

[ObjectGridEventGroup](#)

```
public static interface ObjectGridEventGroup.ShardEvents
```

These events are fired when a shard is made a primary shard and when the shard is demoted from a primary.

Method Summary

[void](#) [shardActivated](#)([ObjectGrid](#) grid)

This is called when a shard is promoted to a primary.

[void](#) [shardDeactivate](#)([ObjectGrid](#) grid)

This is called when a primary shard is demoted to a replica.

Method Detail

shardActivated

```
void shardActivated(ObjectGrid grid)
```

This is called when a shard is promoted to a primary.

Parameters:

grid - This is a local reference to the shard containing the primary data.

shardDeactivate

```
void shardDeactivate(ObjectGrid grid)
```

This is called when a primary shard is demoted to a replica. This can happen is the balancer decides the primary is better placed in a different container. Replication is still active until this method returns to the caller. If any application controlled transactions are in flight then they should be stopped before returning. Once this method returns then any remaining transactions will fail.

Parameters:

grid - A reference to the shard.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup.TransactionEvents

Enclosing interface:

[ObjectGridEventGroup](#)

```
public static interface ObjectGridEventGroup.TransactionEvents
```

These events are called every single transaction. These are primarily used when transaction level listening is required. This is usually for pushing changes or invalidation events to peer caches for simple scenarios.

Method Summary

v o i d	transactionBegin (String txid, boolean isWriteThroughEnabled) Signals the beginning of a Session transaction.
------------------	---

v o i d	transactionEnd (String txid, boolean isWriteThroughEnabled, boolean committed, Collection changes) Signals the ending of a Session transaction.
------------------	---

Method Detail

transactionBegin

```
void transactionBegin(String txid,  
                     boolean isWriteThroughEnabled)
```

Signals the beginning of a Session transaction.

A stringified version of the TxID is provided for correlating with the end of the transaction, if so desired. The type of transaction is also provided by the isWriteThroughEnabled boolean parameter.

Parameters:

txid - Stringified version of the TxID

isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the `Session.beginNoWriteThrough()` method. false is passed if `beginNoWriteThrough()` was used.

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#)

transactionEnd

```
void transactionEnd(String txid,  
                  boolean isWriteThroughEnabled,
```

boolean committed,
[Collection](#) changes)

Signals the ending of a Session transaction.

A string version of the TxID is provided for correlating with the begin of the transaction, if so desired. Map changes are also reported with the collection of LogSequences passed to this method. Typical uses of this event are for customers doing custom peer invalidation or peer commit push. This event listener gives them the changes. Calls to this method are made after commit and are sequenced so that they are delivered one by one, not in parallel. The event order is the commit and rollback order.

Parameters:

- txid - string version of the TxID
- isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the `Session.beginNoWriteThrough()` method. false is passed if `beginNoWriteThrough()` was used.
- committed - a boolean flag indicating whether the transaction was committed (true) or rolled back (false)
- changes - a Collection of LogSequences representing the changes that were committed or rolled back.

See Also:

[LogSequence.isRollback\(\)](#), [Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#), [Session.commit\(\)](#), [Session.rollback\(\)](#)

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All			Classes			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD								
OD								

com.ibm.websphere.objectgrid.plugins

Interface MapEventListener

All Superinterfaces:

com.ibm.websphere.objectgrid.plugins.EventListener

```
public interface MapEventListener  
extends com.ibm.websphere.objectgrid.plugins.EventListener
```

This callback interface is implemented by the application when it wants to receive events about a Map such as the eviction of a map entry.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.addMapEventListener\(EventListener\)](#),
[BackingMap.removeMapEventListener\(EventListener\)](#), EventListener

Method Summary

v o i d	entryEvicted (Object key, Object value) Invoked when the specified entry is evicted from the map.
------------------	---

v o i d	preloadCompleted (Throwable t) Invoked when the preloading of this map has completed.
------------------	---

Method Detail

entryEvicted

```
void entryEvicted(Object key,  
                 Object value)
```

Invoked when the specified entry is evicted from the map.

The eviction could have occurred either by an Evictor's processing or by invoking one of the invalidate methods on the ObjectMap.

For a MapEventListener in an [ObjectMap](#) that is configured to use OutputFormat.RAW for the keys and values, the keys and values objects passed will be SerializedKey or SerializedValue objects respectively. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key or value object.

To override the map's output format configuration, use the PluginOutputFormat annotation in the implementation class.

Parameters:

key - The key for the map entry that was evicted.
value - The value that was in in the map entry evicted. The value object should not be modified.

See Also:

[Evictor](#), [EvictionEventCallback](#), [ObjectMap.invalidate\(Object, boolean\)](#)

preloadCompleted

void `preloadCompleted`([Throwable](#) t)

Invoked when the preloading of this map has completed.

This method is useful to determine when a preload operation finishes if asynchronous preloading is enabled. In addition if any error occurred during synchronous or asynchronous preload, it is reported with the invocation of this method.

Parameters:

t - A Throwable object that indicates if preload completed without any Throwable occurring during the preload of the map. A null reference indicates preload completed without any Throwable objects occurring during the preload of the map.

See Also:

`Loader.preloadMap(Session, BackingMap)`, [BackingMap.setPreloadMode\(boolean\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Interface LogSequenceFilter

public interface **LogSequenceFilter**

This interface can be used to filter a LogSequence. As an operation, such as serialization, needs to know whether a given LogElement should be included or not, this callback object will be used for the boolean check. If the given LogElement should be used in the operation, then "true" should be returned. If the given LogElement should not be used, then "false" should be returned. This interface is primarily used by the `serialize` method of the `LogSequenceTransformer` class.

Since:

WAS XD 6.0, XC10

Method Summary

b o o l e a n	<code>accept</code> (LogElement logElement) Returns true if the given LogElement should be used; false if the given LogElement should not be used.
---------------------------------	--

Method Detail

accept

boolean `accept`([LogElement](#) logElement)

Returns true if the given LogElement should be used; false if the given LogElement should not be used.

Parameters:

logElement - the LogElement to be filtered

Returns:

true if the given LogElement should be used in the operation; false otherwise.

com.ibm.websphere.objectgrid.plugins
Interface LogSequence

All Superinterfaces:
[Serializable](#)

public interface **LogSequence**
 extends [Serializable](#)

LogSequence is the ordered list of changes performed against a given map for a given transaction. These changes are recorded as LogElement objects.

Since:
 WAS XD 6.0, XC10

Method Summary	
Iterator	<p>getAllChanges() Returns an iterator for processing all of the changes for a LogSequence.</p>
Iterator	<p>getChangesByKeys(Collection keys) Returns an iterator for processing the LogElements that have the requested keys.</p>
Iterator	<p>getChangesByTypes(Collection types) Returns an iterator for processing the LogElements that are of the requested type.</p>
String	<p>getMapName() Returns the name of the map that these changes apply to.</p>
String	<p>getObjectGridName() Returns the name of the ObjectGrid that houses the map that these changes apply to.</p>

I t e r a t o r	<p>getPendingChanges() Returns an iterator for processing all of the "pending" changes for a LogSequence (for example, pending inserts, updates, and deletes).</p>
b o o l e a n	<p>isDirty() Returns whether this LogSequence has any LogElements that would "dirty" a Map.</p>
b o o l e a n	<p>isRollback() Returns whether or not this LogSequence was generated to rollback a transaction.</p>
i n t	<p>size() Returns the total number of LogElements within this LogSequence.</p>

Method Detail

size

int **size()**

Returns the total number of LogElements within this LogSequence.

Returns:
total number of LogElements

getPendingChanges

[Iterator](#) **getPendingChanges()**

Returns an iterator for processing all of the "pending" changes for a LogSequence (for example, pending inserts, updates, and deletes).

This method is normally used by a Loader. A pending change is one that has not been written out to a loader yet using a `flush()` operation. Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Returns:
an Iterator for processing the pending LogElement changes

See Also:
[ObjectMap.flush\(\)](#), [Session.flush\(\)](#)

getAllChanges

[Iterator](#) **getAllChanges()**

Returns an iterator for processing all of the changes for a LogSequence.

This method would normally be used by an Evictor and other plugins that want to know all of the changes introduced by this LogSequence. Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Returns:

an Iterator for processing all of the LogElement changes

getChangesByTypes

[Iterator](#) `getChangesByTypes(Collection types)`

Returns an iterator for processing the LogElements that are of the requested type.

Each member of the input Collection should be one of the defined LogElement Types (INSERT, UPDATE, DELETE, FETCH, TOUCH, or EVICT). Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Parameters:

types - A Collection of LogElement Types (INSERT, UPDATE, etc)

Returns:

Iterator for processing all LogElements that support the input Type(s)

Throws:

[IllegalArgumentException](#) - if types is null

See Also:

[LogElement.DELETE](#), [LogElement.EVICT](#), [LogElement.FETCH](#), [LogElement.INSERT](#), [LogElement.TOUCH](#), [LogElement.UPDATE](#), [LogElement.CLEAR](#)

getChangesByKeys

[Iterator](#) `getChangesByKeys(Collection keys)`

Returns an iterator for processing the LogElements that have the requested keys.

Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Parameters:

keys - a collection of key objects

Returns:

an Iterator for processing all LogElements that match the input key(s)

getMapName

[String](#) `getMapName()`

Returns the name of the map that these changes apply to.

The caller can use the return value of this method as input to the `Session.getMap(String)` method.

Returns:

The name of the map that these changes apply to

See Also:

[Session.getMap\(String\)](#)

getObjectGridName

[String](#) `getObjectGridName()`

Returns the name of the ObjectGrid that houses the map that these changes apply to.

Returns:

The name of the ObjectGrid that this LogSequence is associated with

Since:

WAS XD 6.0.1

isDirty

boolean **isDirty()**

Returns whether this LogSequence has any LogElements that would "dirty" a Map.

That is, if it contains any LogElements of any type other than Fetch/Get, it is considered "dirty".

Returns:

true if the LogSequence would modify a Map, if applied; false if the LogSequence would not modify a Map, if applied

isRollback

boolean **isRollback()**

Returns whether or not this LogSequence was generated to rollback a transaction.

Note, depending on when this LogSequence is used, the transaction itself might already be rolled back.

Returns:

true iff this LogSequence was generated to rollback a transaction.

Since:

WAS XD 6.0.1

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins
Interface LogElement

All Superinterfaces:
[Serializable](#)

```
public interface LogElement
extends Serializable
```

LogElements are the individual entries within a LogSequence. A LogElement has attributes such as operation type (delete, insert, update, etc.), current value, last access time, versioned value, etc. A LogElement is created during a transaction to record in-flight operations. For a LogElement on an [ObjectMap](#) that is configured to use OutputFormat.RAW for the keys or values, the keys or values objects in the LogElement will be SerializedKey or SerializedValue objects respectively. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key or value object. To override the map's output format configuration, use the PluginOutputFormat annotation in the caller of the LogElement.

Since:
 WAS XD 6.0, XC10

See Also:
[LogSequence](#)

Nested Class Summary

s t a t i c c l a s s	<p>LogElement.Type</p> <p>The Type class is used to represent a LogElement type.</p>
---	--

Field Summary

s t a t i c f i e l d	<p>CLEAR</p> <p>The Type that represents the CLEAR operation.</p>
---	---

t
I
V
P
E

s
t
a
t
i
c
i
n
t

[CODE_CLEAR](#)

The type code constant for CLEAR.

s
t
a
t
i
c
i
n
t

[CODE_DELETE](#)

The type code constant for DELETE.

s
t
a
t
i
c
i
n
t

[CODE_EVICT](#)

The type code constant for EVICT.

s
t
a
t
i
c
i
n
t

[CODE_FETCH](#)

The type code constant for FETCH.

s
t
a
t
i
c
i
n
t

[CODE_INSERT](#)

The type code constant for INSERT.

s
t
a
t
i
c
i
n
t

[CODE_LOCK](#)

The type code constant for LOCK.

s
t
a
t
i
c
i

[CODE_TOUCH](#)

The type code constant for TOUCH.

n
t

s
t
a
t
i
c
i
n
t

CODE_UNDO_NOT_NEEDED

The code constant for UNDO_NOT_NEEDED.

s
t
a
t
i
c
i
n
t

CODE_UPDATE

The type code constant for UPDATE.

s
t
a
t
i
c
i
n
t

CODE_UPSERT

The type code constant for UPSERT.

s
t
a
t
i
c
L
O
Q
U
E
L
E
M
E
N
T
T
Y
P
E

DELETE

The Type that represents the DELETE operation.

s
t
a
t
i
c
L
O
Q
U
E
L
E
M
E
N
T
T
Y
P
E

EVICT

The Type that represents the EVICT operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T
T
I
P
E

FETCH

The Type that represents the FETCH operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T
T
I
P
E

INSERT

The Type that represents the INSERT operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T
T
I
P
E

LOCK

The Type that represents the LOCK operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T

TOUCH

The Type that represents the TOUCH operation.

e
n
t
i
t
y
p
e

s
t
a
t
i
c
L
o
g
E
l
e
m
e
n
t
i
t
y
p
e

UNDO_NOT_NEEDED

The Type that represents an UNDO action is NOT required for this LogElement.

s
t
a
t
i
c
L
o
g
E
l
e
m
e
n
t
i
t
y
p
e

UPDATE

The Type that represents the UPDATE operation.

s
t
a
t
i
c
L
o
g
E
l
e
m
e
n
t
i
t
y
p
e

UPSERT

The Type that represents the UPSERT operation.

Method Summary

O
b
j
e
c
t

[getAfterImage\(\)](#)

Gets the "after image" value object.

O
b
j
e
c
t

[getBeforeImage\(\)](#)

Gets the "before image" of the value object.

C
a
c
h
e
E
n
t
r
y

[getCacheEntry\(\)](#)

Returns the CacheEntry for this key.

O
b
j
e
c
t

[getCurrentValue\(\)](#)

Gets the value for this LogElement.

O
b
j
e
c
t

[getKey\(\)](#)

Returns the key for this LogElement.

L
o
g

[getLastAccessTime\(\)](#)

Returns the last access time associated with this LogElement.

L
o
g
E
l
e
m
e
n
t
.
T
y
p
e

[getType\(\)](#)

Gets the type of this LogElement.

L
o
g
E
l
e
m
e
n
t
.
T
y
p
e

[getUndoType\(\)](#)

Returns what operation must be performed to "undo" a prior change the transaction made to the map entry.

e	
Object	<p>getVersionedValue() Gets the versioned object at the time the object was first associated with the transaction.</p>
boolean	<p>isCascaded() Answers true if this LogElement is a result of a cascade operation.</p>
boolean	<p>isPending() Answers true if this change has NOT been applied to the loader.</p>
void	<p>setVersionedValue(Object v) Used to update the versioned object after an update of map entry occurs.</p>

Field Detail

CODE_INSERT

static final int **CODE_INSERT**

The type code constant for INSERT.

See Also:

[INSERT](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_UPDATE

static final int **CODE_UPDATE**

The type code constant for UPDATE.

See Also:

[UPDATE](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_DELETE

static final int **CODE_DELETE**

The type code constant for DELETE.

See Also:

[DELETE](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_EVICT

static final int **CODE_EVICT**

The type code constant for EVICT.

See Also:

[EVICT](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_FETCH

static final int **CODE_FETCH**

The type code constant for FETCH.

See Also:

[FETCH](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_TOUCH

static final int **CODE_TOUCH**

The type code constant for TOUCH.

See Also:

[TOUCH](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_CLEAR

static final int **CODE_CLEAR**

The type code constant for CLEAR.

Since:

WAS XD 6.1.0.3

See Also:

[CLEAR](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_LOCK

static final int **CODE_LOCK**

The type code constant for LOCK.

Since:

8.6, XC10 2.5

See Also:

[LOCK](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_UPSERT

static final int **CODE_UPSERT**

The type code constant for UPSERT.

Since:

8.6, XC10 2.5

See Also:

[UPSERT](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_UNDO_NOT_NEEDED

static final int **CODE_UNDO_NOT_NEEDED**

The code constant for UNDO_NOT_NEEDED. Used to indicate no operation is needed to undo the changes for this LogElement since this LogElement was never processed or it was an operation that does not require an undo operation.

See Also:

[Constant Field Values](#)

INSERT

static final [LogElement.Type](#) **INSERT**

The Type that represents the INSERT operation.

UPDATE

static final [LogElement.Type](#) **UPDATE**

The Type that represents the UPDATE operation.

DELETE

static final [LogElement.Type](#) **DELETE**

The Type that represents the DELETE operation.

EVICT

static final [LogElement.Type](#) **EVICT**

The Type that represents the EVICT operation.

FETCH

static final [LogElement.Type](#) **FETCH**

The Type that represents the FETCH operation.

TOUCH

static final [LogElement.Type](#) **TOUCH**

The Type that represents the TOUCH operation.

CLEAR

static final [LogElement.Type](#) **CLEAR**

The Type that represents the CLEAR operation.

Since:

WAS XD 6.1.0.3

LOCK

static final [LogElement.Type](#) LOCK

The Type that represents the LOCK operation.

Since:
8.6, XC10 2.5

UPSERT

static final [LogElement.Type](#) UPSERT

The Type that represents the UPSERT operation.

Since:
8.6, XC10 2.5

UNDO_NOT_NEEDED

static final [LogElement.Type](#) UNDO_NOT_NEEDED

The Type that represents an UNDO action is NOT required for this LogElement.

Method Detail

getType

[LogElement.Type](#) getType()

Gets the type of this LogElement. The type indicates what operation needs to be applied to the map entry.

Returns:
the type of this LogElement.

getCurrentValue

[Object](#) getCurrentValue()

Gets the value for this LogElement.

For a LogElement on an [ObjectMap](#) that is configured to use a ValueSerializerPlugin, the values in the LogSequence will be SerializedValue objects. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original value object.

The original value represents the new value that should be applied to the BackingMap and Loader. This value can be cast to ValueProxyInfo when a value interface is in use in order to determine the set of dirty attributes.

Returns:
the value in case of INSERT, UPDATE, UPSERT, or FETCH, null in the case of DELETE or EVICT.

See Also:
[ValueProxyInfo](#)

getCacheEntry

[CacheEntry](#) `getCacheEntry()`

Returns the CacheEntry for this key. The key, current committed value, etc. can be accessed from the CacheEntry.

Returns:

the entry in the cache that is requested to be updated.

See Also:

[CacheEntry.getCommittedValue\(\)](#), [getKey\(\)](#)

isPending

boolean `isPending()`

Answers true if this change has NOT been applied to the loader.

Changes can previously be applied to a loader using the `ObjectMap.flush()` or `Session.flush()` methods. This method reveals whether the change in this `LogElement` has already been applied to the Loader using one of those methods.

Returns:

true if this change has NOT been applied to the loader.

See Also:

[ObjectMap.flush\(\)](#), [Session.flush\(\)](#)

getVersionedValue

[Object](#) `getVersionedValue()`

Gets the versioned object at the time the object was first associated with the transaction.

For a `LogElement` on an [ObjectMap](#) that is configured to use a `ValueSerializerPlugin`, the versioned object will be returned as an `XsDataInputStream`, read will be `SerializedKey` or `SerializedValue` objects respectively. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original key or value object. For a `LogElement` on an [ObjectMap](#) that is configured to use a `ValueSerializerPlugin` that generates version objects, the version object will be the data stream representing the data.

Returns:

The versioned object.

See Also:

`OptimisticCallback`

setVersionedValue

void `setVersionedValue(Object v)`

Used to update the versioned object after an update of map entry occurs.

The Loader can use this method when it is using an optimistic strategy and uses the `OptimisticCallback.updateVersionedObjectForValue(Object)` method to get an updated version object.

Parameters:

v - The versioned object.

See Also:

`OptimisticCallback.updateVersionedObjectForValue(Object)`

getLastAccessTime

long `getLastAccessTime()`

Returns the last access time associated with this `LogElement`.

Returns:

last access time

getUndoType

[LogElement.Type](#) `getUndoType()`

Returns what operation must be performed to "undo" a prior change the transaction made to the map entry.

Note, an undo type of `UNDO_NOT_NEEDED` is returned if nothing needs to be undone for this `LogElement`.

Returns:

the "undo" type of this `LogElement`. It can be one of: `INSERT`, `UPDATE`, `DELETE` or `UNDO_NOT_NEEDED`

getBeforeImage

[Object](#) `getBeforeImage()`

Gets the "before image" of the value object.

The "before image" is the value object that existed in map entry prior to applying a change to map entry. Note, it is possible for a `null` reference to be returned (e.g. in the case where a new map entry is created).

For a `LogElement` on an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the values, the value will be a `SerializedValue` object. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original value object.

To override the map's output format configuration, use the `PluginOutputFormat` annotation in the caller of the `LogElement`.

Returns:

the value prior to applying the change

getAfterImage

[Object](#) `getAfterImage()`

Gets the "after image" value object.

The "after image" is the value object that existed in map entry after applying a change to the map entry. Note, it is possible for a `null` reference to be returned (e.g. in the case where an existing map entry is removed/evicted).

For a `LogElement` on an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the value, the value will be a `SerializedValue` object. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original value object.

To override the map's output format configuration, use the `PluginOutputFormat` annotation in the caller of the `LogElement`.

Returns:
the value after applying the change

isCascaded

boolean `isCascaded()`

Answers true if this `LogElement` is a result of a cascade operation. This only applies to ObjectGrid EntityManager programming model.

ObjectGrid EntityManager supports cascade operations. For example, when persisting an entity P, if P has a relation to entity C with `CascadeType.PERSIST` enabled, C will also be persisted as a result of the cascade operation. The method `isCascaded()` returns true for the `LogElement` object which represents C, and the method returns false for the `LogElement` object which represents P.

Returns:
true if the `LogElement` object is a result of cascade operation.

Since:
6.1.0.5 FIX1

See Also:
[EntityManager](#)

getKey

[Object](#) `getKey()`

Returns the key for this `LogElement`.

For a `LogElement` on an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the keys, the value will be a `SerializedKey` object. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original key object.

To override the map's output format configuration, use the `PluginOutputFormat` annotation in the caller of the `LogElement`.

This method can be used instead of `LogElement.getCacheEntry().getKey()`.

Returns:
the key for this `LogElement`.

Since:
7.0

See Also:
[CacheEntry.getKey\(\)](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins
Class LogElement.Type

[java.lang.Object](#)



All Implemented Interfaces:

[Comparable](#)

Enclosing interface:

[LogElement](#)

```

public static class LogElement.Type
    extends Object
    implements Comparable
  
```

The Type class is used to represent a LogElement type.

Since:

WAS XD 6.0

Method Summary	
i n t	compareTo (Object object)
i n t	getCode () Gets the type code for this object.
S t r i n g	toString ()
s t a t i c L o g E l e m e n t . T y p e	valueOf (int typeCode) Return the Type for a given type code.

Methods inherited from class java.lang.[Object](#)[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)**Method Detail****getCode**

```
public int getCode()
```

Gets the type code for this object.

Returns:
the type code

compareTo

```
public int compareTo(Object object)
```

Specified by:
[compareTo](#) in interface [Comparable](#)

See Also:
[Comparable.compareTo\(Object\)](#)

toString

```
public String toString()
```

Overrides:
[toString](#) in class [Object](#)

See Also:
[Object.toString\(\)](#)

valueOf

```
public static LogElement.Type valueOf(int typeCode)
```

Return the Type for a given type code.

Parameters:
typeCode - the typecode

Returns:
the Type

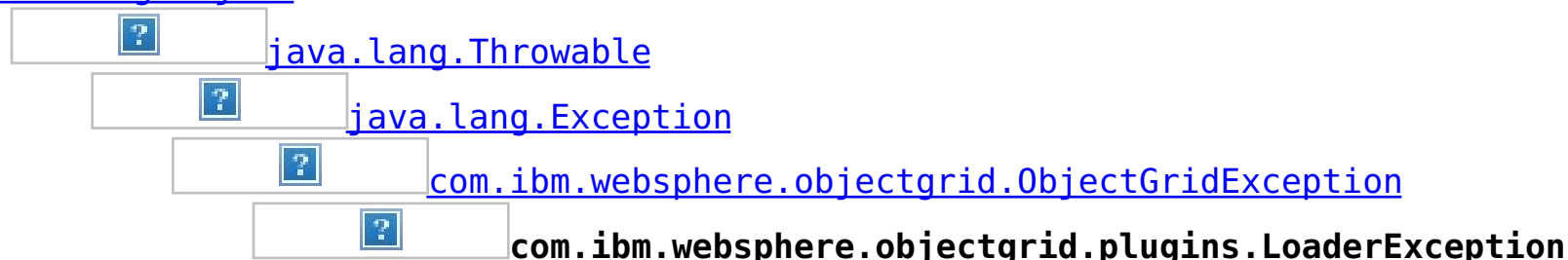
Throws:
[IllegalArgumentException](#) - if the typeCode isn't valid.

Since:
8.6, XC10 2.5

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins Class LoaderException

[java.lang.Object](#)



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:
[UnavailableServiceException](#)

```
public class LoaderException
extends ObjectGridException
```

This exception is the base exception for any exceptions encountered by a Loader.

Since:
WAS XD 6.0, XC10

See Also:
Loader, [Serialized Form](#)

Constructor Summary

- [LoaderException\(\)](#)
Constructs a new LoaderException with null as its detail message.
- [LoaderException\(String message\)](#)
Constructs a new LoaderException with the specified detail message.
- [LoaderException\(String message, Throwable cause\)](#)
Constructs a new LoaderException with the specified detail message and cause.
- [LoaderException\(Throwable cause\)](#)
Constructs a new LoaderException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)
[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)
[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LoaderException

```
public LoaderException()
```

Constructs a new LoaderException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LoaderException

```
public LoaderException(String message)
```

Constructs a new LoaderException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

LoaderException

```
public LoaderException(String message,  
                       Throwable cause)
```

Constructs a new LoaderException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this LoaderException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

LoaderException

```
public LoaderException(Throwable cause)
```

Constructs a new LoaderException with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for LoaderExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for

later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>	
PREV CLASS	NEXT CLASS	FRAMES		NO FRAMES	All			
		Classes						
SUMMARY: NESTED FIELD CONSTR METH OD DETAIL: FIELD CONSTR METHOD								

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface EvictorData

All Known Implementing Classes:

[EvictorData.SpecialEvictorData](#)

```
public interface EvictorData
```

This interface is optionally used by an implementator of the Evictor interface. Application changes applied to a BackingMap are asynchronous from the Evictor activity. The Evictor is not notified of changes to the BackingMap until after application transactions are committed. Consequently, if an Evictor decides to evict a map entry, it is possible that the BackingMap could evict an entry that was different from the original entry it was tracking. For example, consider that an application could execute a transaction that removes a map entry. Before the Evictor is notified of the remove, another transaction inserts a new entry into the BackingMap for the same key as the old entry. Consequently, the Evictor could evict the newly created entry when it meant to evict the old entry. To help close this small timing window, the Evictor can use this interface to associate evictor specific data with a map entry. The Evictor can then do the following:

- store the EvictorData object for a map entry by using the EvictionEventCallback.setEvictorData(Object, EvictorData) method.
- retrieve the EvictorData object for a map entry by using the EvictionEventCallback.getEvictorData(Object) method.
- Conditionally evict a map entry if and only if the cache entry for a specified key has the exact same EvictorData object (the java == operator returns true) associated with it by using the EvictionEventCallback.evictMapEntries(List) method.

Since:

WAS XD 6.0.1, XC10

See Also:

[Evictor](#), [EvictionEventCallback](#)

Nested Class Summary

s
t
a
t
i
c
c
l
a
s
s

[EvictorData.SpecialEvictorData](#)

Special value class used for representing the key not being found in the BackingMap.

Field Summary

s
t
a
t

i
c
E
V
i
c
t
o
r
D
a
t
a

[KEY_NOT_FOUND](#)

A special value indicating that the key was not found.

Method Summary

O
b
j
e
c
t

[getKey\(\)](#)

Retrieves the key object for this EvictorData instance.

Field Detail

KEY_NOT_FOUND

static final [EvictorData](#) KEY_NOT_FOUND

A special value indicating that the key was not found.

Method Detail

getKey

[Object](#) [getKey\(\)](#)

Retrieves the key object for this EvictorData instance.

Returns:

the same key object that was passed to the `EvictionEventCallback.setEvictorData(Object, EvictorData)` method when this `EvictorData` was associated with the map entry with the given key.

See Also:

[EvictionEventCallback.setEvictorData\(Object, EvictorData\)](#)

[Overvi](#) [Packa](#) [Cla](#) [TreeSerializ](#) [Depreca](#) [Index](#) [Help](#)
[ew](#) [ge](#) [ss](#) [ed](#) [ted](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Class EvictorData.SpecialEvictorData

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.plugins.EvictorData.SpecialEvictorData

All Implemented Interfaces:

[EvictorData](#)

Enclosing interface:

[EvictorData](#)

```
public static final class EvictorData.SpecialEvictorData
extends Object
implements EvictorData
```

Special value class used for representing the key not being found in the BackingMap.

Since:

WAS XD 6.0.1

Nested Class Summary

Nested classes/interfaces inherited from interface
com.ibm.websphere.objectgrid.plugins.[EvictorData](#)

[EvictorData.SpecialEvictorData](#)

Field Summary

Fields inherited from interface com.ibm.websphere.objectgrid.plugins.[EvictorData](#)

[KEY_NOT_FOUND](#)

Constructor Summary

[EvictorData.SpecialEvictorData](#)()

Method Summary

[getKey](#)()

Dummy implementation method since this class will not be called.

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

EvictorData.SpecialEvictorData

```
public EvictorData.SpecialEvictorData()
```

Method Detail

getKey

```
public Object getKey()
```

Dummy implementation method since this class will not be called.

Specified by:

[getKey](#) in interface [EvictorData](#)

Returns:

null

See Also:

[EvictionEventCallback.setEvictorData\(Object, EvictorData\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Interface Evictor

All Known Implementing Classes:

[LFUEvictor](#), [LRUEvictor](#)

public interface **Evictor**

Data contained in a BackingMap are evicted when the map is full. This plugin is used by the BackingMap to determine when and what to evict from the map based on some algorithm (LRU, LFU, time based, etc).

An Evictor implementation that also implements the BackingMapLifecycleListener interface will be automatically added as an EventListener on the [BackingMap](#) when the evictor set on the backing map.

An Evictor may also implement the BackingMapPlugin interface in order to receive enhanced BackingMap plug-in lifecycle method calls. The plug-in is then also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.addMapEventListener\(EventListener\)](#), [BackingMap.setEvictor\(Evictor\)](#), [EvictorData](#)

Method Summary

activate()

This method is called to activate the Evictor.

apply([LogSequence](#) sequence)

Called after a transaction has committed to allow the evictor to track object usage in the BackingMap.

deactivate()

This method is called to deactivate the Evictor.

destroy()

Called when the BackingMap associated with this evictor is destroyed.

initialize([BackingMap](#) map, [EvictionEventCallback](#) callback)

Called by a BackingMap instance during the evictor initialization time.

Method Detail

initialize

```
void initialize(BackingMap map,  
               EvictionEventCallback callback)
```

Called by a [BackingMap](#) instance during the evictor initialization time.

The [BackingMap](#) calls this method so the Evictor instance can have references to the [BackingMap](#) and [EvictionEventCallback](#) instances. The evictor can signal events to have specific entries evicted using the [EvictionEventCallback](#).

Parameters:

map - the [BackingMap](#) instance
callback - the [EvictionEventCallback](#) instance

See Also:

[BackingMap](#), [EvictionEventCallback](#)

destroy

```
void destroy()
```

Called when the [BackingMap](#) associated with this evictor is destroyed.

This method is the opposite of the `initialize` method. When it is called, the Evictor can free up any resources it uses.

See Also:

[ObjectGrid.destroy\(\)](#)

apply

```
void apply(LogSequence sequence)
```

Called after a transaction has committed to allow the evictor to track object usage in the [BackingMap](#).

This method also reports any entries that have been successfully evicted. Note, this method is not called for transactions that are rolled back. If there is a need to track object usage for rolled back transactions, the evictor must implement the [RollbackEvictor](#) interface as well.

This method is called after a transaction has completed. Consequently, all transaction locks that were acquired by the completed transaction are no longer held. Potentially, multiple threads could call this method concurrently and each thread would be completing its own transaction. Since transaction locks are already released by the completed transaction, this method must provide its own synchronization to ensure it is thread safe. For an Evictor in an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the keys or values, the keys and values objects in the [LogSequence](#) will be `SerializedKey` or `SerializedValue` objects respectively. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original key or value object. To override the map's output format configuration, use the `PluginOutputFormat` annotation in the implementation class.

Parameters:

sequence - the [LogSequence](#) of changes committed to the map

See Also:

[RollbackEvictor](#)

activate

void **activate**()

This method is called to activate the Evictor. Until this method is called, the Evictor must not use the EvictionEventCallback interface to evict any map entries. If it does use the EvictionEventcallback interface to evict map entries prior to activate being called, an IllegalStateException is thrown.

deactivate

void **deactivate**()

This method is called to deactivate the Evictor. Once this method is called, the Evictor must quit using the EvictionEventCallback interface to evict any map entries. If it does use the EvictionEventcallback interface after this method is called, an IllegalStateException is thrown.

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface EvictionEventCallback

public interface **EvictionEventCallback**

An instance of EvictionEventCallback is passed into the Evictor at initialization time. When an eviction method is called, corresponding methods of EvictionEventCallback will be called so the BackingMap can process evictions.

Since:

WAS XD 6.0, XC10

See Also:

[Evictor](#), [EvictorData](#)

Method Summary

v o i d	evictEntries (List keysToEvictList) If an Evictor chooses not to implement the EvictorData interface, this method can be used to evict a map entry.
v o i d	evictMapEntries (List evictorDataList) This method is the preferred method for the Evictor to use when evicting map entries.
E v i c t o r D a t a	getEvictorData (Object key) Gets the evictor data for a specified BackingMap cache entry.
v o i d	setEvictorData (Object key, EvictorData data) Sets the evictor data for a specified BackingMap cache key.

Method Detail

setEvictorData

void [setEvictorData](#)([Object](#) key,
[EvictorData](#) data)

Sets the evictor data for a specified BackingMap cache key.

This method can be used by an implementor of the `Evictor` interface to keep data that the evictor needs for determining which cache entry to evict.

Parameters:

key - is the key for accessing a `BackingMap` entry.

data - the `EvictorData` object to store as evictor data for a specified key.

Throws:

[IllegalArgumentException](#) - if key is a null reference or there is no `BackingMap` cache entry for this key.

Since:

WAS XD 6.0.1

See Also:

[Evictor](#)

getEvictorData

[EvictorData](#) `getEvictorData(Object key)`

Gets the evictor data for a specified `BackingMap` cache entry.

Parameters:

key - the key for the `BackingMap` entry to set.

Returns:

if the specified key is not found in `BackingMap`, then the special value `EvictorData.KEY_NOT_FOUND` is returned. If the key is found in the `BackingMap`, the same reference that was previously passed to the `setEvictorData(Object, EvictorData)` method of this interface is returned. A null reference is returned if the key is found, but the `setEvictorData` method was not previously called for the specified key.

Throws:

[IllegalArgumentException](#) - if key is a null reference.

Since:

WAS XD 6.0.1

See Also:

[setEvictorData\(Object, EvictorData\)](#), [EvictorData.KEY_NOT_FOUND](#)

evictMapEntries

void `evictMapEntries(List evictorDataList)`
throws [ObjectGridException](#)

This method is the preferred method for the `Evictor` to use when evicting map entries. A list of `EvictorData` objects is passed as an argument to this method. For each `EvictorData` object in the list, the key is obtained from the `EvictorData` object and used to determine which `BackingMap` entry to evict. The `BackingMap` entry is evicted if and only if the cache entry for `BackingMap` entry contains the exact same `EvictorData` object in it. That is, the java `==` operator is used to ensure it is the exact same `EvictorData` object. If the `==` operator indicates a different object, then the map entry is not evicted. For those map entries that are physically evicted from the map, the `Evictor` will receive notification through its `apply` method.

Parameters:

evictorDataList - a list of `EvictorData` objects to process. The caller must guarantee this parameter is not null or contain any null references.

Throws:

[ObjectGridException](#) - if an error occurs during processing

[ClassCastException](#) - if an object in `evictorDataList` does not implement the `EvictorData` interface.

Since:

WAS XD 6.0.1

See Also:

[Evictor.apply\(LogSequence\)](#), [EvictorData.getKey\(\)](#)

evictEntries

void **evictEntries**([List](#) keysToEvictList)
throws [ObjectGridException](#)

If an Evictor chooses not to implement the EvictorData interface, this method can be used to evict a map entry. However, the Evictor must be prepared to handle the exposure of an application removing and recreating a map entry before the Evictor has an opportunity to call this method.

For this method, a list of map keys is passed. The list is evaluated and an eviction is conducted on the list. When the entries are physically evicted from the map, the Evictor will receive notification through its apply method.

Parameters:

keysToEvictList - List of keys to evict from the map. The caller must guarantee this parameter is not null or contain any null references.

Throws:

[ObjectGridException](#) - if an error occurs during processing

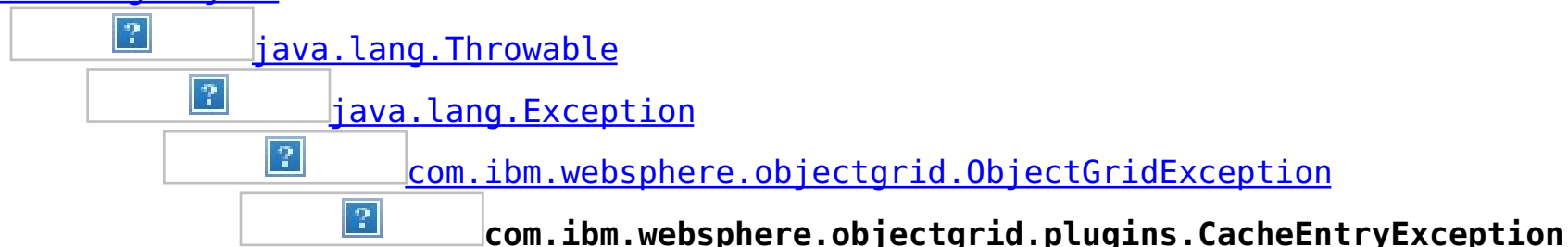
See Also:

[Evictor.apply\(LogSequence\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid.plugins
Class CacheEntryException

[java.lang.Object](#)



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

```
public class CacheEntryException
extends ObjectGridException
```

This exception indicates an error occurred during a cache entry operation.

Since:
WAS XD 6.0, XC10

See Also:
[Serialized Form](#)

Constructor Summary

CacheEntryException ()	Constructs a new CacheEntryException with null as its detail message.
CacheEntryException (String message)	Constructs a new CacheEntryException with the specified detail message.
CacheEntryException (String message, Throwable cause)	Constructs a new CacheEntryException with the specified detail message and cause.
CacheEntryException (Throwable cause)	Constructs a new CacheEntryException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

CacheEntryException

```
public CacheEntryException()
```

Constructs a new CacheEntryException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

CacheEntryException

```
public CacheEntryException(String message)
```

Constructs a new CacheEntryException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

CacheEntryException

```
public CacheEntryException(String message,
                           Throwable cause)
```

Constructs a new CacheEntryException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this CacheEntryException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

CacheEntryException

```
public CacheEntryException(Throwable cause)
```

Constructs a new CacheEntryException with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for CacheEntryExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

com.ibm.websphere.objectgrid.plugins
Interface CacheEntry

All Superinterfaces:
[Serializable](#)

public interface **CacheEntry**
 extends [Serializable](#)

This interface represents a cache entry in an ObjectGrid map.

Since:
 WAS XD 6.0, XC10

Method Summary	
Object	getCommittedValue() Returns the committed value for this entry.
Object	getKey() Returns the key for this entry.
long	getLastAccessTime() Returns the last time this entry was accessed.
long	getTTL() Returns the time-to-live value for this entry.
boolean	isInBackingMap() Indicates whether this entry is in the BackingMap

Method Detail

isInBackingMap

boolean **isInBackingMap()**

Indicates whether this entry is in the BackingMap

Returns:

Returns true if this element is in the BackingMap

getKey

[Object](#) getKey()

Returns the key for this entry.

For a CacheEntry on an [ObjectMap](#) that is configured to use a KeySerializerPlugin, the value will be a SerializedKey object. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key object.

Returns:

the key

getCommittedValue

[Object](#) getCommittedValue()

Returns the committed value for this entry.

The type of the object returned from the getCommittedValue() method depends on various configuration and storage options used by the [ObjectMap](#) that holds the CacheEntry. In the default case, getCommittedValue() returns the Java object of the same type that was put into the map. For an [ObjectMap](#) that is configured to use a ValueSerializerPlugin, the committed value depends on the underlying storage mechanism, typically represented as an array of bytes.

Returns:

the committed value

getTTL

long getTTL()

Returns the time-to-live value for this entry.

Returns:

the time-to-live value

getLastAccessTime

long getLastAccessTime()

Returns the last time this entry was accessed.

Returns:

last access time.

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#) [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins
Interface BeanFactory

public interface **BeanFactory**

Implement this interface to allow bean factories like Spring or Google guice to be integrated. This allows ObjectGrid to delegate to an external Bean Factory to instantiate beans needed by ObjectGrid.

Since:
WAS XD 6.1 FIX3, XC10

Method Summary

ObjectGrid	getBean (String name) This returns an instance of the bean with the specified name.
------------	--

Method Detail

getBean

[Object](#) **getBean**([String](#) name)

This returns an instance of the bean with the specified name.

Parameters:

name - The name of the bean instance to return.

Returns:

the bean instance.

Package com.ibm.websphere.objectgrid.security

This package has the class MapPermission and class AdminPermission which represents the permissions for to access the ObjectGrid maps and ObjectGrid administration respectively.

See:

[Description](#)

Class Summary	
SecurityConstants	This class contains the constants used for security configuration.

Exception Summary	
ObjectGridSecurityException	This exception represents a general ObjectGrid security exception.

Package com.ibm.websphere.objectgrid.security Description

This package has the class MapPermission and class AdminPermission which represents the permissions for to access the ObjectGrid maps and ObjectGrid administration respectively.

MapPermission action types.

The ObjectGrid defines 5 permission actions that are used to authorize accesses to the maps. These permissions allow access to maps to be controlled by an administrator. Objects within the ObjectGrid use a simple naming scheme. Each Map is named using the convention of the ObjectGrid name followed by a period followed by the Map name. For example, if the object grid name is "myObjectGrid" and the map name is "myMap", then the map name used in the permission is "myobjectgrid.mymap".

Wildcards can be used on names with some restrictions. A wild card "*" can be used to replace the map name or the object grid name, but not partially. For example, "myObjectGrid.*", ".*myMap", and ".*" are valid names, but "myObject*.*" is not valid.

There are five actions with the permission object ObjectMapPermission.

- Read
This action allows get operations to be issued against a Map.
- Write
This action allows put operations to be issued against a Map. It allows existing entries to be updated.
- Remove
This action allows entries to be removed from the Map.
- Insert
This action allows clients to add entries to a Map.

- Invalidate
This action allows clients to invalidate entries from the Map.

	com.ibm.websphere.objectgrid.ObjectMap/ com.ibm.websphere.objectgrid.JavaMap
Read	boolean containsKey(Object)
	boolean equals(Object)
	Object get(Object)
	Object get(Object, Serializable)
	List getAll(List)
	List getAll(List keyList, Serializable)
	List getAllForUpdate(List)
	List getAllForUpdate(List, Serializable)
	Object getForUpdate(Object)
	Object getForUpdate(Object, Serializable)
write	Object put(Object key, Object value)
	void put(Object, Object, Serializable)
	void putAll(Map)
	void putAll(Map, Serializable)
	void update(Object, Object)
	void update(Object, Object, Serializable)
insert	public void insert(Object, Object)
	void insert(Object, Object, Serializable)
	remove Object remove(Object)
	void removeAll(Collection)
invalidate	public void invalidate(Object, boolean)
	void invalidateAll(Collection, boolean)
	int setTimeToLive(int)

An authroizationMechanism setting of the ObjectGrid has two possible values: JAAS and custom. Users can also use API [ObjectGrid.setAuthorizationMechanism\(int\)](#) to set which authorization mechanism the object grid will use.

A value "JAAS" means ObjectGrid will rely on JAAS authorization mechanism to handle the authorization. A JAAS policy file should be configured to associate permissions with a set of credentials and/or groups of credentials. We recommend that groups should be used as then new users can be added to groups without modifying the policy file.

A value "custom" means ObjectGrid will rely on custom authorization mechanism to handle the authorization. Users can set call [ObjectGrid.setObjectGridAuthorization\(com.ibm.websphere.objectgrid.security.plugins.ObjectGridAuth orization ogAuthorization\)](#) to set their custom authorization plug-in. Users can also configure the objectgrid.xml to achieve the same result.

AdminPermission types

An AdminPermission has two types: ADMIN and MONITOR. An AdminPermission with ADMIN name grants permissions to access all the ManagementMBean methods. An AdminPermission with MONITOR name grants permissions to access the ManagementMBean read-only methods. Therefore, ADMIN permission implies MONITOR permission.

The detailed operations granted to users with different permissions are listed in the following

table. These operations correspond to the methods in the ManagementMBean interface:

operations	admin	monitor
startServer	Y	N
stopServer	Y	N
forceStopServer	Y	N
setServerTrace	Y	N
retrieveServerStatus	Y	Y
getMapStats	Y	Y
getOGStats	Y	Y
getReplicationStats	Y	Y

The table can read like this: If the client has admin permission, it can execute "startServer" task; if the client has monitor permission, it cannot execute "startServer" task.

AgentPermission types

An AgentPermission represents permissions to the datagrid agents. The name of the permission is the full name of the ObjectGrid map, and the action is a "," delimited string of agent implementation class names or package names.

The following methods in the class AgentManager requires AgentPermission:

- AgentManager.callMapAgent(MapGridAgent, Collection)
- AgentManager.callMapAgent(MapGridAgent)
- AgentManager.callReduceAgent(ReduceGridAgent, Collection)
- AgentManager.callReduceAgent(ReduceGridAgent, Collection)

ObjectGridPermission types

An ObjectGridPermission represents permissions to an ObjectGrid. The name of the permission is the ObjectGrid name, and the action is either "query" or "dynamicmap".

The detailed methods which require different permissions are listed in the following table:

methods	action
Session.createObjectQuery(String)	query
EntityManager.createQuery(String)	query
Session.getMap(String)	dynamicmap

ServerMapPermission types

An ServerMapPermission represents permissions to an ObjectMap hosted in a server. The name of the permission is the full name of the ObjectGrid map name, and the action is either "replicate" or "dynamicIndex".

The detailed methods which require different ServerMapPermission are listed in the following table:

methods	action
ClientReplicableMap.enableClientReplication(Mode, int[], ReplicationMapListener)	replicate
BackingMap.createDynamicIndex(String, boolean, String, DynamicIndexCallback)	dynamicIndex
BackingMap.removeDynamicIndex(String)	dynamicIndex

SecurityConstants

SecurityConstants class contains constants used for representing the security parameters.

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.security
Class SecurityConstants

[java.lang.Object](#)



```
public class SecurityConstants
extends Object
```

This class contains the constants used for security configuration.

Since:
 WAS XD 6.0, XC10

Field Summary

s t a t i c i n t	<p>ACCESS_BY_CREATOR_ONLY_COMPLEMENT The access by creator only authorization is enabled to complement the ObjectGrid map authorization.</p>
s t a t i c i n t	<p>ACCESS_BY_CREATOR_ONLY_DISABLED The access by creator only authorization is disabled.</p>
s t a t i c i n t	<p>ACCESS_BY_CREATOR_ONLY_SUPERSEDE The access by creator only authorization is enabled to supersede the ObjectGrid map authorization.</p>
s t a t i c i n t	<p>AUTHORIZATION_MECHANISM_CUSTOM Constant representing custom authorization</p>
s t	

a t t i c i n t	<u>AUTHORIZATION_MECHANISM_JAAS</u> Constant representing JAAS authorization
s t a t i c i n t	<u>CLIENT_CERTIFICATE_AUTHENTICATION_NEVER</u> Constant indicating client certificate authentication is not supported.
s t a t i c i n t	<u>CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED</u> Constant indicating client certificate authentication is required.
s t a t i c i n t	<u>CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED</u> Constant indicating client certificate authentication is supported.
s t a t i c i n t	<u>CREDENTIAL_AUTHENTICATION_NEVER</u> Constant indicating credential authentication is not supported.
s t a t i c i n t	<u>CREDENTIAL_AUTHENTICATION_REQUIRED</u> Constant indicating credential authentication is required.
s t a t i c i n t	<u>CREDENTIAL_AUTHENTICATION_SUPPORTED</u> Constant indicating credential authentication is supported.
s t a t i c	<u>NEVER_STRING</u>

String

String representation for value Never indicating an option is not supported.

statistic
String

NEW_SECURE_TOKEN_MANAGER_STRING

String representation for "autoSecret" type of the secure token manager.

statistic
String

REQUIRED_STRING

String representation for value Required indicating an option is required.

statistic
String

SECURE_TOKEN_MANAGER_CUSTOM_STRING

String representation for "custom" type of the secure token manager.

statistic
String

SECURE_TOKEN_MANAGER_DEFAULT_STRING

String representation for "default" type of the secure token manager.

statistic
String

SECURE_TOKEN_MANAGER_NONE_STRING

String representation for "none" type of the secure token manager.

statistic
i

SSL_REQUIRED

Constant indicating SSL transport is required.

n t	
s t a t i c S t r i n g	<p><u>SSL_REQUIRED_STRING</u> String representation for value SSL-Required indicating SSL transport type is required.</p>
s t a t i c i n t	<p><u>SSL_SUPPORTED</u> Constant indicating SSL transport is supported.</p>
s t a t i c S t r i n g	<p><u>SSL_SUPPORTED_STRING</u> String representation for value SSL-Supported indicating SSL transport type is supported.</p>
s t a t i c S t r i n g	<p><u>SUPPORTED_STRING</u> String representation for value Supported indicating an option is supported.</p>
s t a t i c i n t	<p><u>TCP_IP</u> Constant indicating TCP/IP is the only supported transport.</p>
s t a t i c S t r i n g	<p><u>TCPIP_STRING</u> String representation for value TCP/IP indicating a transport type of TCP/IP is used.</p>

Constructor Summary

[SecurityConstants\(\)](#)

Method Summary

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Field Detail

AUTHORIZATION_MECHANISM_JAAS

```
public static final int AUTHORIZATION_MECHANISM_JAAS
```

Constant representing JAAS authorization

See Also:

[ObjectGrid.setAuthorizationMechanism\(int\)](#), [Constant Field Values](#)

AUTHORIZATION_MECHANISM_CUSTOM

```
public static final int AUTHORIZATION_MECHANISM_CUSTOM
```

Constant representing custom authorization

See Also:

[ObjectGrid.setAuthorizationMechanism\(int\)](#), [Constant Field Values](#)

TCP_IP

```
public static final int TCP_IP
```

Constant indicating TCP/IP is the only supported transport.

If the client's transport type is set to this value, TCP/IP is the only supported transport type. If the server requires SSL, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setTransportType\(int\)](#), [Constant Field Values](#)

SSL_SUPPORTED

```
public static final int SSL_SUPPORTED
```

Constant indicating SSL transport is supported.

If the client's transport type is set to this value, the client supports both TCP/IP and SSL. SSL will be used if both sides side supports SSL. Otherwise, TCP/IP will be used.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setTransportType\(int\)](#), [Constant Field Values](#)

SSL_REQUIRED

```
public static final int SSL_REQUIRED
```

Constant indicating SSL transport is required.

If the client's transport type is set to this value, SSL is the only supported transport type. If the server requires TCP/IP, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setTransportType\(int\)](#), [Constant Field Values](#)

CREDENTIAL_AUTHENTICATION_NEVER

```
public static final int CREDENTIAL_AUTHENTICATION_NEVER
```

Constant indicating credential authentication is not supported.

If the credential authentication type is set to this value, no credential authentication will be enforced. If the server requires credential authentication, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setCredentialAuthenticationType\(int\)](#), [Constant Field Values](#)

CREDENTIAL_AUTHENTICATION_SUPPORTED

```
public static final int CREDENTIAL_AUTHENTICATION_SUPPORTED
```

Constant indicating credential authentication is supported.

If the credential authentication type is set to this value, credential authentication will be enforced if and only if both client and server support credential authentication.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setCredentialAuthenticationType\(int\)](#), [Constant Field Values](#)

CREDENTIAL_AUTHENTICATION_REQUIRED

```
public static final int CREDENTIAL_AUTHENTICATION_REQUIRED
```

Constant indicating credential authentication is required.

If the credential authentication type is set to this value, credential authentication will be enforced. If the server doesn't support credential authentication, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setCredentialAuthenticationType\(int\)](#), [Constant Field Values](#)

CLIENT_CERTIFICATE_AUTHENTICATION_NEVER

```
public static final int CLIENT_CERTIFICATE_AUTHENTICATION_NEVER
```

Constant indicating client certificate authentication is not supported.

If the client certificate authentication type is set to this value, no client certificate authentication will be enforced. If the server doesn't support client certificate authentication, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setClientCertificateAuthentication\(int\)](#), [Constant Field Values](#)

CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED

```
public static final int CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED
```

Constant indicating client certificate authentication is supported.

If the client certificate authentication type is set to this value, client certificate authentication will be enforced when the following conditions are met:

- both client and server supports or requires client certificate authentication;
- the transport protocol to use is SSL;
- no credential authentication will be done.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setClientCertificateAuthentication\(int\)](#), [Constant Field Values](#)

CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED

```
public static final int CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED
```

Constant indicating client certificate authentication is required. If the client certificate authentication type is set to this value, client certificate authentication will be enforced if the following conditions are met:

- both client and server supports or requires client certificate authentication;
- the transport protocol to use is SSL;
- no credential authentication will be done.

If the server doesn't support client certificate authentication and no credential authentication will be done, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setClientCertificateAuthentication\(int\)](#), [Constant Field Values](#)

NEVER_STRING

```
public static final String NEVER_STRING
```


String representation for value Never indicating an option is not supported.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration settings "clientCertificateAuthentication" and "credentialAuthentication".

Since:

WAS XD 6.0.1

See Also:

[CLIENT_CERTIFICATE_AUTHENTICATION_NEVER](#), [CREDENTIAL_AUTHENTICATION_NEVER](#), [Constant Field Values](#)

SUPPORTED_STRING

```
public static final String SUPPORTED_STRING
```

String representation for value Supported indicating an option is supported.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration settings "clientCertificateAuthentication" and "credentialAuthentication".

Since:

WAS XD 6.0.1

See Also:

[CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED](#), [CREDENTIAL_AUTHENTICATION_SUPPORTED](#), [Constant Field Values](#)

REQUIRED_STRING

```
public static final String REQUIRED_STRING
```

String representation for value Required indicating an option is required.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration settings "clientCertificateAuthentication" and "credentialAuthentication".

Since:

WAS XD 6.0.1

See Also:

[CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED](#), [CREDENTIAL_AUTHENTICATION_REQUIRED](#), [Constant Field Values](#)

TCPIP_STRING

```
public static final String TCPIP_STRING
```

String representation for value TCP/IP indicating a transport type of TCP/IP is used.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration setting "transportType".

Since:

WAS XD 6.0.1

See Also:

[TCP_IP](#), [Constant Field Values](#)

SSL_SUPPORTED_STRING

```
public static final String SSL_SUPPORTED_STRING
```

String representation for value SSL-Supported indicating SSL transport type is supported.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration setting "transportType".

Since:

WAS XD 6.0.1

See Also:

[SSL_SUPPORTED](#), [Constant Field Values](#)

SSL_REQUIRED_STRING

```
public static final String SSL_REQUIRED_STRING
```

String representation for value SSL-Required indicating SSL transport type is required.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration setting "transportType".

Since:

WAS XD 6.0.1

See Also:

[SSL_REQUIRED](#), [Constant Field Values](#)

SECURE_TOKEN_MANAGER_NONE_STRING

```
public static final String SECURE_TOKEN_MANAGER_NONE_STRING
```

String representation for "none" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType".

Since:

WAS XD 6.0.1

See Also:

[Constant Field Values](#)

SECURE_TOKEN_MANAGER_DEFAULT_STRING

```
public static final String SECURE_TOKEN_MANAGER_DEFAULT_STRING
```

String representation for "default" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType". This value requires users to provide the secure token key store settings.

Since:

WAS XD 6.0.1

See Also:

[Constant Field Values](#)

SECURE_TOKEN_MANAGER_CUSTOM_STRING

```
public static final String SECURE_TOKEN_MANAGER_CUSTOM_STRING
```

String representation for "custom" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType". This value requires users to provide the SecureTokenManager implementation class name using the "customSecureTokenManagerClass" configuration setting.

Since:

WAS XD 6.0.1

See Also:

[Constant Field Values](#)

NEW_SECURE_TOKEN_MANAGER_STRING

```
public static final String NEW_SECURE_TOKEN_MANAGER_STRING
```

String representation for "autoSecret" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType". This value does not require users to provide other settings.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

ACCESS_BY_CREATOR_ONLY_DISABLED

```
public static final int ACCESS_BY_CREATOR_ONLY_DISABLED
```

The access by creator only authorization is disabled.

The access by creator authorization ensures that only the user (represented by the Principals associated with it), who inserts the data entry into the map, can access the data. Here the access means read, update, invalidate, and remove.

Since:

WAS XD 6.1 FIX3

See Also:

[Constant Field Values](#)

ACCESS_BY_CREATOR_ONLY_COMPLEMENT

```
public static final int ACCESS_BY_CREATOR_ONLY_COMPLEMENT
```

The access by creator only authorization is enabled to complement the ObjectGrid map authorization.

The access by creator authorization ensures that only the user (represented by the Principals associated with it), who inserts the data entry into the map, can access the data. Here the access means read, update, invalidate, and remove.

If this constant is used, both map authorization and access by creator only authorization will take effect. Therefore, you can further limit the operations to the data entries. For example, you can restrict the creator from invalidating the data entries.

Since:

WAS XD 6.1 FIX3

See Also:

[Constant Field Values](#)

ACCESS_BY_CREATOR_ONLY_SUPERSEDE

```
public static final int ACCESS_BY_CREATOR_ONLY_SUPERSEDE
```

The access by creator only authorization is enabled to supersede the ObjectGrid map authorization.

The access by creator authorization ensures that only the user (represented by the Principals associated with it), who inserts the data entry into the map, can access the data. Here the access means read, update, invalidate, and remove.

If this constant is used, the access by creator only authorization will supersede the map authorization; no map authorization will be done.

Since:

WAS XD 6.1 FIX3

See Also:

[Constant Field Values](#)

Constructor Detail

SecurityConstants

```
public SecurityConstants()
```

[Overview](#)
[Package](#)
[Classes](#)
[TreeSerialized](#)
[Deprecated](#)
[Index](#)
[Help](#)

[PREV CLASS](#)
[NEXT CLASS](#)
[FRAMES](#)
[NO FRAMES](#)
[All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#)
 DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification

com.ibm.websphere.objectgrid.security

Class ObjectGridSecurityException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[CannotGenerateCredentialException](#), [ExpiredCredentialException](#),
[InvalidCredentialException](#)

```
public class ObjectGridSecurityException
extends ObjectGridException
```

This exception represents a general ObjectGrid security exception.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridSecurityException](#)()

Constructs a new ObjectGridSecurityException with null as its detail message.

[ObjectGridSecurityException](#)(String message)

Constructs a new ObjectGridSecurityException with the specified detail message.

[ObjectGridSecurityException](#)(String message, Throwable cause)

Constructs a new ObjectGridSecurityException with the specified detail message and cause.

[ObjectGridSecurityException](#)(Throwable cause)

Constructs a new ObjectGridSecurityException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#),
[printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridSecurityException

```
public ObjectGridSecurityException()
```

Constructs a new `ObjectGridSecurityException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ObjectGridSecurityException

```
public ObjectGridSecurityException(String message)
```

Constructs a new `ObjectGridSecurityException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridSecurityException

```
public ObjectGridSecurityException(String message,
                                   Throwable cause)
```

Constructs a new `ObjectGridSecurityException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ObjectGridSecurityException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ObjectGridSecurityException

```
public ObjectGridSecurityException(Throwable cause)
```

Constructs a new `ObjectGridSecurityException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for

ObjectGridSecurityExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

PREV CLASS [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | [CONSTR](#) | [METH](#) DETAIL: FIELD | [CONSTR](#) | METHOD
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

Package com.ibm.websphere.objectgrid.spring

This package holds the Spring specific APIs for ObjectGrid.

See:

[Description](#)

Interface Summary

SpringLocalTransactionManager	This interface has the methods for interacting with the ObjectGrid Spring LocalTransactionManager.
---	--

Class Summary

ObjectGridCache	This class is a WebSphere eXtreme Scale implementation of the Spring Framework's Cache interface.
ObjectGridCatalogServiceDomainBean	This class is a Spring bean representing a Catalog Service Domain for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.
ObjectGridClientBean	This class is a Spring bean representing an ObjectGrid client instance for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.
ObjectGridSpringFactory	This class serves as a factory to construct instances of the various Spring specific APIs.

Exception Summary

CannotGetObjectGridSessionException	This can be thrown by getSession if a new Session cannot be obtained for any reason.
ObjectGridTransactionException	The ObjectGrid platform transaction manager can throw this unchecked exception whenever errors occur.

Package com.ibm.websphere.objectgrid.spring Description

This package holds the Spring specific APIs for ObjectGrid.

Local Transaction Support

ObjectGrid has implemented a Spring PlatformTransactionManager. This allows Spring to manage local transactions using a single ObjectGrid session. Spring can then be used to annotate POJOs with container managed transaction semantics much like a J2EE application server does using J2EE CMT. An application should instantiate a SpringLocalTxManager using the appropriate factory method on ObjectGridSpringFactory and then wire a reference to that object in to all POJOs that use Spring CMT. This instance has a getSession method to obtain the correct Session for that POJO. The application must call one of the SpringLocalTxManager#setObjectGridForThread methods before invoking a managed POJO to specify which ObjectGrid instance should be used for any CMT on this thread.

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV PACKAGE](#) [NEXT PACKAGE](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.spring

Interface SpringLocalTxManager

public interface **SpringLocalTxManager**

This interface has the methods for interacting with the ObjectGrid Spring LocalTransactionManager. It also allows the desired partition to use with this thread to be specified.

Since:

WAS XD 6.1 FIX3, XC10

See Also:

[ObjectGridSpringFactory.getLocalPlatformTransactionManager\(\)](#)

Method Summary

S e s s i o n	getSession() This returns a managed session for the ObjectGrid associated with this thread.
V o i d	setObjectGridForThread(ObjectGrid grid) This indicates the ObjectGrid to use on this thread when a session is requested.

Method Detail

setObjectGridForThread

void **setObjectGridForThread**([ObjectGrid](#) grid)

This indicates the ObjectGrid to use on this thread when a session is requested. This replaces any previously associated ObjectGrid, i.e. only a single grid instance can be associated with a thread at a time.

Parameters:

grid - the ObjectGrid to set on this thread.

See Also:

[getSession\(\)](#)

getSession

[Session](#) **getSession()**

This returns a managed session for the ObjectGrid associated with this thread.

Do not call `begin()`, `commit()` or `rollback()` directly on the session. Spring manages the transaction automatically.

Returns:

A managed Session to use with this thread.

Throws:

[CannotGetObjectGridSessionException](#) - thrown when an ObjectGrid session can't be retrieved.

See Also:

[setObjectGridForThread\(ObjectGrid\)](#)

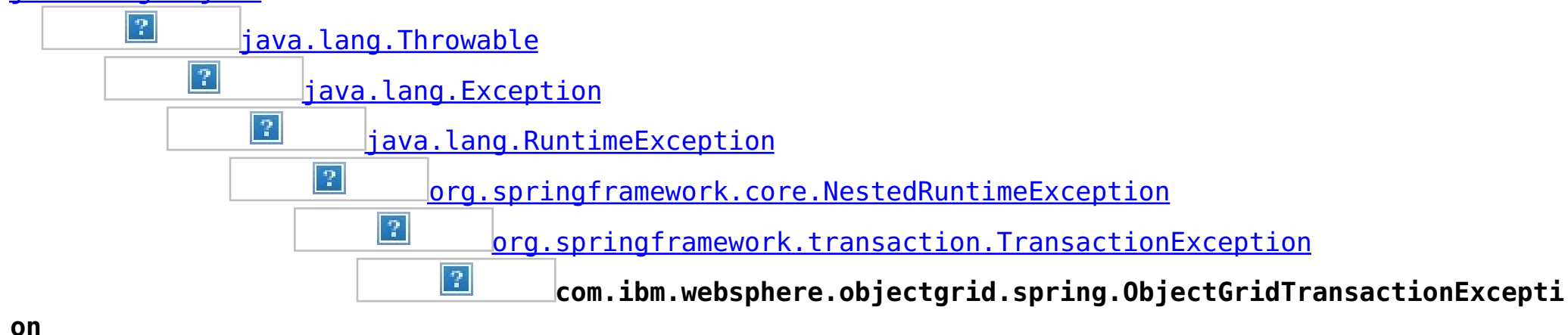
Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.spring

Class ObjectGridTransactionException

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public class ObjectGridTransactionException
extends TransactionException
```

The ObjectGrid platform transaction manager can throw this unchecked exception whenever errors occur.

Since:

WAS XD 6.1 FIX3, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridTransactionException](#)([String](#) message)

Constructs a new ObjectGridTransactionException with the specified detail message.

[ObjectGridTransactionException](#)([String](#) message, [Throwable](#) cause)

Constructs a new ObjectGridTransactionException with the specified detail message and cause.

Method Summary

Methods inherited from class org.springframework.core.[NestedRuntimeException](#)

[contains](#), [getMessage](#), [getMostSpecificCause](#), [getRootCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getCause](#), [getLocalizedMessage](#), [getStackTrace](#), [initCause](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridTransactionException

```
public ObjectGridTransactionException(String message,  
                                     Throwable cause)
```

Constructs a new ObjectGridTransactionException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ObjectGridTransactionException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[Throwable.getCause\(\)](#), [NestedRuntimeException.getMessage\(\)](#)

ObjectGridTransactionException

```
public ObjectGridTransactionException(String message)
```

Constructs a new ObjectGridTransactionException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[Throwable.initCause\(Throwable\)](#), [NestedRuntimeException.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.spring

Class ObjectGridSpringFactory

[java.lang.Object](#)



```
public final class ObjectGridSpringFactory
extends Object
```

This class serves as a factory to construct instances of the various Spring specific APIs.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary

s t a t i c	<p>SCOPE_SHARD Scope identifier for shard scope: "shard".</p>
----------------------------	---

Constructor Summary

<p>ObjectGridSpringFactory()</p>
--

Method Summary

s t a t i c	<p>getBeanFactoryForObjectGrid(String objectGridName) This returns the currently registered external bean factory for a named object grid.</p>
s t a	

t
i
c
O
b
j
e
c
t

[getBeanInShardScope](#)([ObjectGrid](#) shard, [String](#) beanName)

This returns an instance of the named Spring bean with the current shard scope using the specified ObjectGrid instance.

s
t
a
t
i
c
S
p
r
i
n
g
L
o
c
a
l
T
r
a
n
s
a
c
t
i
o
n
M
a
n
a
g
e
r

[getLocalPlatformTransactionManager](#)()

This returns an ObjectGrid PlatformLocalTransactionManager.

s
t
a
t
i
c
v
o
i
d

[registerSpringBeanFactoryAdapter](#)([String](#) objectGridName, [Object](#) springBeanFactory)

This returns an adapter for a Spring based bean factory.

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Field Detail

SCOPE_SHARD

public static final [String](#) SCOPE_SHARD

Scope identifier for shard scope: "shard".

See Also:

[getBeanInShardScope\(ObjectGrid, String\)](#), [Constant Field Values](#)

Constructor Detail

ObjectGridSpringFactory

public [ObjectGridSpringFactory](#)()

Method Detail

getLocalPlatformTransactionManager

```
public static SpringLocalTxManager getLocalPlatformTransactionManager()
```

This returns an ObjectGrid PlatformLocalTransactionManager.

Returns:

the PlatformLocalTransactionManager instance.

registerSpringBeanFactoryAdapter

```
public static void registerSpringBeanFactoryAdapter(String objectGridName,  
                                                    Object springBeanFactory)  
    throws ClassCastException
```

This returns an adapter for a Spring based bean factory. We use an Object type here to avoid making ObjectGrid dependent on Spring classes being present. A ClassCastException exception is thrown if the supplied factory isn't a Spring BeanFactory instance.

Parameters:

objectGridName - the name of the ObjectGrid
springBeanFactory - A Spring BeanFactory instance.

Throws:

[ClassCastException](#) - thrown when the Object type is not a BeanFactory instance.

getBeanFactoryForObjectGrid

```
public static BeanFactory getBeanFactoryForObjectGrid(String objectGridName)
```

This returns the currently registered external bean factory for a named object grid. If no factory has been registered then it attempts to construct a Spring BeanFactory using the xml resource on the class path @ "/X_spring.xml" and /META-INF/X_spring.xml where X is the name of the ObjectGrid. If the xml file is on the class path then the ObjectGrid name MUST be a valid resource name.

Parameters:

objectGridName - The name of the ObjectGrid

Returns:

The BeanFactory instance or null if there were none registered

getBeanInShardScope

```
public static Object getBeanInShardScope(ObjectGrid shard,  
                                          String beanName)
```

This returns an instance of the named Spring bean with the current shard scope using the specified ObjectGrid instance. This allows shard scoped beans to be obtained.

Parameters:

shard - The ObjectGrid instance to use to scope Spring beans using "shard" as scope.
beanName - The bean to return

Returns:

The bean instance if it exists

See Also:

[SCOPE_SHARD](#)

com.ibm.websphere.objectgrid.spring
Class ObjectGridClientBean

[java.lang.Object](#)



All Implemented Interfaces:
[InitializingBean](#)

```
public final class ObjectGridClientBean
extends Object
implements InitializingBean
```

This class is a Spring bean representing an ObjectGrid client instance for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.

Users must provide a [ObjectGridCatalogServiceDomainBean](#) to configure this client. The ObjectGrid name is optional when using the provided XML configuration files.

Since:
8.5, XC10

Constructor Summary

[ObjectGridClientBean\(\)](#)

Method Summary

V O I D	afterPropertiesSet() Initializes the client bean.
V O I D	setCatalogServiceDomain(ObjectGridCatalogServiceDomainBean catalogServiceDomain) Sets the ObjectGridCatalogServiceDomainBean used by the client.
V O I D	setObjectGridName(String objectGridName) Sets the name of the ObjectGrid for the client.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridClientBean

```
public ObjectGridClientBean()
```

Method Detail

setObjectGridName

```
public void setObjectGridName(String objectGridName)
```

Sets the name of the ObjectGrid for the client. This is optional when using the provided XML configuration files.

Parameters:

objectGridName - The name of the ObjectGrid to connect to.

setCatalogServiceDomain

```
public void setCatalogServiceDomain(ObjectGridCatalogServiceDomainBean catalogServiceDomain)
```

Sets the [ObjectGridCatalogServiceDomainBean](#) used by the client.

Parameters:

catalogServiceDomain - The [ObjectGridCatalogServiceDomainBean](#) for the client.

afterPropertiesSet

```
public void afterPropertiesSet()  
    throws Exception
```

Initializes the client bean. This method is to be called after all setter methods have been called and will throw an [IllegalArgumentException](#) if [setCatalogServiceDomain\(ObjectGridCatalogServiceDomainBean\)](#) has not been called prior.

Specified by:

[afterPropertiesSet](#) in interface [InitializingBean](#)

Throws:

[Exception](#) - if [setCatalogServiceDomain\(ObjectGridCatalogServiceDomainBean\)](#) has not been called prior or an error occurs retrieving the ObjectGrid.

See Also:

[InitializingBean.afterPropertiesSet\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.spring

Class ObjectGridCatalogServiceDomainBean

[java.lang.Object](#)



All Implemented Interfaces:

[InitializingBean](#)

```
public final class ObjectGridCatalogServiceDomainBean
extends Object
implements InitializingBean
```

This class is a Spring bean representing a Catalog Service Domain for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.

Users must provide the catalog service endpoints used to connect to the eXtreme Scale cluster/domain. Users optionally may provide a client override XML and/or a client security configuration.

Since:

8.5, XC10

Constructor Summary

[ObjectGridCatalogServiceDomainBean\(\)](#)

Method Summary

void [afterPropertiesSet\(\)](#)
Initializes the connection to the Catalog Service Domain.

void [setCatalogServiceEndpoints\(String catalogServiceEndpoints\)](#)
Sets the catalog service endpoints used to connect to the cluster/domain.

void [setClientOverrideXml\(Resource clientOverrideXml\)](#)
Sets the location of the client override XML.

void [setClientSecurityConfig\(Resource clientSecurityConfiguration\)](#)
Sets the location of the client security configuration.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridCatalogServiceDomainBean

```
public ObjectGridCatalogServiceDomainBean()
```

Method Detail

setCatalogServiceEndpoints

```
public void setCatalogServiceEndpoints(String catalogServiceEndpoints)
```

Sets the catalog service endpoints used to connect to the cluster/domain.

Parameters:

catalogServiceEndpoints - The catalog service endpoints to connect to the cluster/domain

setClientOverrideXml

```
public void setClientOverrideXml(Resource clientOverrideXml)
```

Sets the location of the client override XML.

Parameters:

clientOverrideXml - The location of the client override XML.

setClientSecurityConfig

```
public void setClientSecurityConfig(Resource clientSecurityConfiguration)
```

Sets the location of the client security configuration.

Parameters:

clientSecurityConfiguration - The location of the client security configuration.

afterPropertiesSet

```
public void afterPropertiesSet()  
    throws Exception
```

Initializes the connection to the Catalog Service Domain. This method is to be called after all setter methods have been called and will throw an [IllegalArgumentException](#) if [setCatalogServiceEndpoints\(String\)](#) have not been called prior.

Specified by:

[afterPropertiesSet](#) in interface [InitializingBean](#)

Throws:

[Exception](#) - if [setCatalogServiceEndpoints\(String\)](#) have not been called prior or an error occurs initializing the connection.

See Also:

[InitializingBean.afterPropertiesSet\(\)](#)

com.ibm.websphere.objectgrid.spring

Class ObjectGridCache

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.spring.ObjectGridCache

All Implemented Interfaces:

[InitializingBean](#), [Cache](#)

```
public final class ObjectGridCache
extends Object
implements Cache, InitializingBean
```

This class is a WebSphere eXtreme Scale implementation of the Spring Framework's [Cache](#) interface.

Users must provide a name and a [ObjectGridClientBean](#) to configure this cache. The ObjectMap name is optional when using the provided XML configuration files.

This implementation allows for the storage of null values and does not support null keys.

The following Spring Inversion of Control (IoC) container configuration snippet creates two caches, named default and books hosted by the catalog service domain with connection endpoints of host1:2809,host2:2809.

```
<bean id="wxsCSDomain" class="com.ibm.websphere.objectgrid.spring.ObjectGridCatalogServiceDomainBean"
  p:catalog-service-endpoints="host1:2809,host2:2809" />

<bean id="wxsGridClient" class="com.ibm.websphere.objectgrid.spring.ObjectGridClientBean"
  p:catalog-service-domain-ref="wxsCSDomain" />

<bean id="cacheManager" class="org.springframework.cache.support.SimpleCacheManager">
  <property name="caches">
    <set>
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="default"
        p:object-grid-client-ref="wxsGridClient" />
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="books"
        p:object-grid-client-ref="wxsGridClient" />
    </set>
  </property>
</bean>
```

Since:

8.5, XC10

Nested Class Summary

Nested classes/interfaces inherited from interface

[org.springframework.cache.Cache](#)

[Cache.ValueWrapper](#)

Constructor Summary

[ObjectGridCache\(\)](#)

Method Summary

void [afterPropertiesSet\(\)](#)
Initializes this cache.

void [clear\(\)](#)
Clears all entries from the cache.

void [evict\(Object key\)](#)
Evicts the entry at the given key.

Cache.ValueWrapper [get\(Object key\)](#)
Retrieves the object from the cache at the given key.

String [getName\(\)](#)
Returns the name of the cache.

Object [getNativeCache\(\)](#)
This implementation returns null.

void [put\(Object key, Object value\)](#)
Creates an entry in the cache.

void [setMapName\(String mapName\)](#)
Sets the ObjectMap name.

```
void setName(String name)
    Sets the cache name.
```

```
void setObjectClient(ObjectGridClientBean objectGridClient)
    Sets the ObjectGridClientBean to use.
```

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridCache

```
public ObjectGridCache()
```

Method Detail

afterPropertiesSet

```
public void afterPropertiesSet()
    throws Exception
```

Initializes this cache. This method is to be called after all setter methods have been called and will throw an [IllegalArgumentException](#) if [setObjectGridClient\(ObjectGridClientBean\)](#) or [setName\(String\)](#) have not been called prior.

Specified by:

[afterPropertiesSet](#) in interface [InitializingBean](#)

Throws:

[Exception](#) - if [setObjectGridClient\(ObjectGridClientBean\)](#) or [setName\(String\)](#) have not been called prior or an error occurs initializing the cache.

See Also:

[InitializingBean.afterPropertiesSet\(\)](#)

getName

```
public String getName()
```

Returns the name of the cache.

Specified by:

[getName](#) in interface [Cache](#)

Returns:

The name of the cache.

See Also:

[Cache.getName\(\)](#)

setName

```
public void setName(String name)
```

Sets the cache name.

Parameters:

name - The cache name.

See Also:

[getName\(\)](#)

setMapName

```
public void setMapName(String mapName)
```

Sets the ObjectMap name. This is optional when using the provided XML configuration files.

Parameters:

mapName - The name of the ObjectMap

setObjectGridClient

```
public void setObjectGridClient(ObjectGridClientBean objectGridClient)
```

Sets the [ObjectGridClientBean](#) to use.

Parameters:

objectGridClient - The [ObjectGridClientBean](#) to use

getNativeCache

```
public Object getNativeCache()
```

This implementation returns null.

Specified by:

[getNativeCache](#) in interface [Cache](#)

Returns:

This implementation returns null.

See Also:

[Cache.getNativeCache\(\)](#)

get

```
public Cache.ValueWrapper get(Object key)
```

Retrieves the object from the cache at the given key. Returns null if there is no mapping for the key or if the key is null

Specified by:

[get](#) in interface [Cache](#)

Returns:

the object from the cache at the given key or null if there is no mapping or the key is null

See Also:

[Cache.get\(java.lang.Object\)](#)

put

```
public void put(Object key,  
               Object value)
```

Creates an entry in the cache. Overwrites the value if an entry for the given key exists. This method does not create an entry if the given key is null.

Specified by:

[put](#) in interface [Cache](#)

See Also:

[Cache.put\(java.lang.Object, java.lang.Object\)](#)

evict

public void **evict**([Object](#) key)

Evicts the entry at the given key. If the provided key is null this method does not evict any entries.

Specified by:

[evict](#) in interface [Cache](#)

See Also:

[Cache.evict\(java.lang.Object\)](#)

clear

public void **clear**()

Clears all entries from the cache.

Specified by:

[clear](#) in interface [Cache](#)

See Also:

[Cache.clear\(\)](#)

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All Classes			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

PREV CLASS [NEXT CLASS](#)

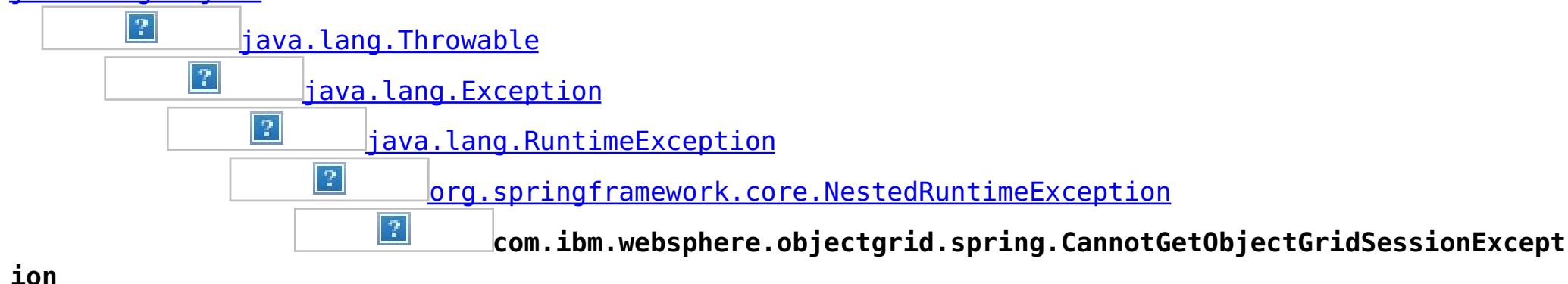
[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

com.ibm.websphere.objectgrid.spring

Class CannotGetObjectGridSessionException

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public class CannotGetObjectGridSessionException
extends NestedRuntimeException
```

This can be thrown by getSession if a new Session cannot be obtained for any reason.

Since:

WAS XD 6.1 FIX3, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[CannotGetObjectGridSessionException](#)([String](#) message)

Constructs a new CannotGetObjectGridSessionException with the specified detail message.

[CannotGetObjectGridSessionException](#)([String](#) message, [Throwable](#) cause)

Constructs a new CannotGetObjectGridSessionException with the specified detail message and cause.

Method Summary

Methods inherited from class [org.springframework.core.NestedRuntimeException](#)

[contains](#), [getMessage](#), [getMostSpecificCause](#), [getRootCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getCause](#), [getLocalizedMessage](#), [getStackTrace](#), [initCause](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

CannotGetObjectGridSessionException

```
public CannotGetObjectGridSessionException(String message,  
                                           Throwable cause)
```

Constructs a new `CannotGetObjectGridSessionException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `CannotGetObjectGridSessionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[Throwable.getCause\(\)](#), [NestedRuntimeException.getMessage\(\)](#)

CannotGetObjectGridSessionException

```
public CannotGetObjectGridSessionException(String message)
```

Constructs a new `CannotGetObjectGridSessionException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[Throwable.initCause\(Throwable\)](#), [NestedRuntimeException.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

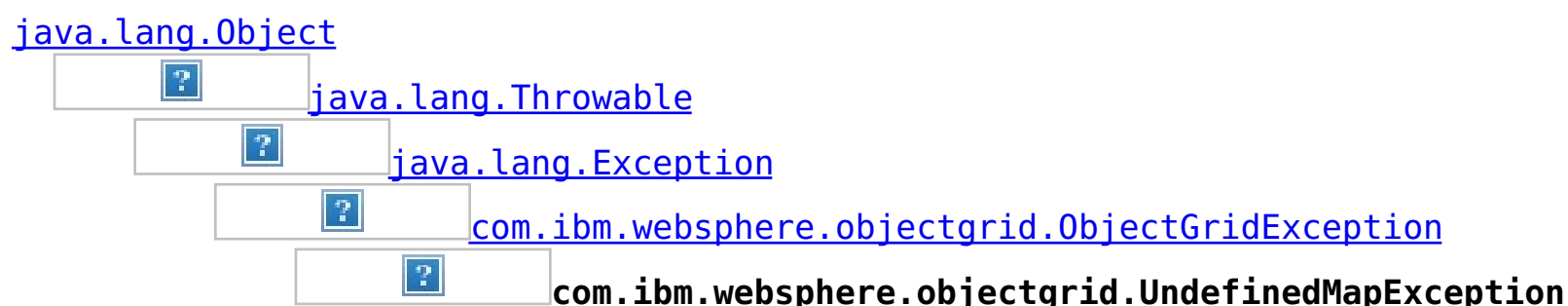
PREV CLASS [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: NESTED | FIELD | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL](#): FIELD | [CONSTR](#) | METHOD

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid
Class UndefinedMapException



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

```
public class UndefinedMapException
extends ObjectGridException
```

This exception indicates that the map which an application tries to access is not defined in the ObjectGrid.

Since:
WAS XD 6.0, XC10

See Also:
[Serialized Form](#)

Constructor Summary	
UndefinedMapException ()	Constructs a new UndefinedMapException with null as its detail message.
UndefinedMapException (String message)	Constructs a new UndefinedMapException with the specified detail message.
UndefinedMapException (String message, Throwable cause)	Constructs a new UndefinedMapException with the specified detail message and cause.
UndefinedMapException (Throwable cause)	Constructs a new UndefinedMapException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid. ObjectGridException
getCause , initCause

Methods inherited from class java.lang. Throwable
fillInStackTrace , getLocalizedMessage , getMessage , getStackTrace , printStackTrace , printStackTrace , printStackTrace , setStackTrace , toString

Methods inherited from class java.lang. Object
--

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

UndefinedMapException

```
public UndefinedMapException()
```

Constructs a new UndefinedMapException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

UndefinedMapException

```
public UndefinedMapException(String message)
```

Constructs a new UndefinedMapException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

UndefinedMapException

```
public UndefinedMapException(String message,  
                             Throwable cause)
```

Constructs a new UndefinedMapException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this UndefinedMapException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

UndefinedMapException

```
public UndefinedMapException(Throwable cause)
```

Constructs a new UndefinedMapException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for UndefinedMapExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that

the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

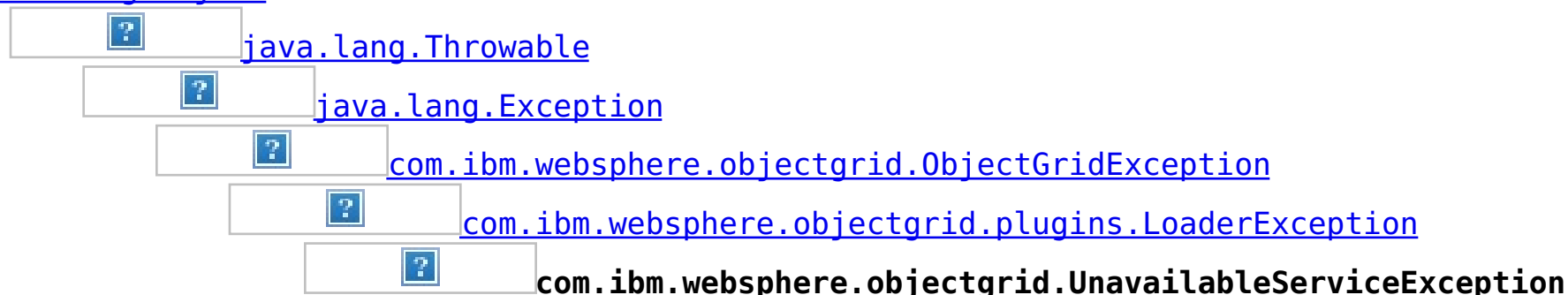
**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class UnavailableServiceException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[TransactionAffinityException](#), [TransactionQuiesceException](#)

```
public class UnavailableServiceException
extends LoaderException
```

This exception is thrown when all servers are dead or when all services are unavailable even though servers are running.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[UnavailableServiceException](#)()

Constructs a new `UnavailableServiceException` with `null` as its detail message.

[UnavailableServiceException](#)([String](#) message)

Constructs a new `UnavailableServiceException` with the specified detail message.

[UnavailableServiceException](#)([String](#) message, [Throwable](#) cause)

Constructs a new `UnavailableServiceException` with the specified detail message and cause.

[UnavailableServiceException](#)([Throwable](#) cause)

Constructs a new `UnavailableServiceException` with a specified cause.

Method Summary

[getReplicationGroup](#)()

Returns the replication group identifier for this exception.

[setReplicationGroup](#)(int replicationGroup)

Sets the replication group identifier for this exception.

Methods inherited from class [com.ibm.websphere.objectgrid.ObjectGridException](#)[getCause](#), [initCause](#)**Methods inherited from class [java.lang.Throwable](#)**[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)**Methods inherited from class [java.lang.Object](#)**[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

UnavailableServiceException

```
public UnavailableServiceException()
```

Constructs a new `UnavailableServiceException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:[ObjectGridException.initCause\(Throwable\)](#)

UnavailableServiceException

```
public UnavailableServiceException(String message)
```

Constructs a new `UnavailableServiceException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

UnavailableServiceException

```
public UnavailableServiceException(String message,  
                                  Throwable cause)
```

Constructs a new `UnavailableServiceException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `UnavailableServiceException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

UnavailableServiceException

public **UnavailableServiceException**([Throwable](#) cause)

Constructs a new UnavailableServiceException with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for UnavailableServiceExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Method Detail

getReplicationGroup

public int **getReplicationGroup**()

Returns the replication group identifier for this exception.

Returns:

the argument that was passed to the `setReplicationGroup(int)` method of this class or 0 if the `setReplicationGroup` method was not previously called for this object.

See Also:

[setReplicationGroup\(int\)](#)

setReplicationGroup

public void **setReplicationGroup**(int replicationGroup)

Sets the replication group identifier for this exception.

Parameters:

replicationGroup - The replication group identifier

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface TxID

All Superinterfaces:

[Serializable](#)

```
public interface TxID
extends Serializable
```

This interface is an opaque identifier for a transaction. Context information can be stored and retrieved in multiple slots on this object. This mechanism allows a TransactionCallback and Loader, for example, to share state information with each other in the context of a specific session transaction.

The TxID.toString() output can be used to determine whether the originating Session transaction is a single partition transaction or a multi-partition transaction. If the String output begins with the keyword Local then this indicates a single partition transaction, for example: Local-40000139-72B2-C037-E000-1C271366B073

If the String output begins with the keyword WXS then this indicates a multi-partition transaction, for example: WXS-40000139-72B2-BD3A-E000-1C271366B073

Since:

WAS XD 6.0, XC10

See Also:

Loader, [ObjectGrid.reserveSlot\(String\)](#), [Session](#), [TransactionCallback](#)

Field Summary

s t a t i c S t r i n g	<p>SLOT_NAME</p> <p>All slots should be reserved using this name.</p>
--	---

Method Summary

b o o l e a n	<p>equals(TxID o)</p> <p>Checks for equality between two TxID objects.</p>
S e	

S S i o n	getSession() Returns the Session that owns this TxID.
O b j e c t	getSlot(int slotNumber) Gets the context information currently associated with this transaction.
i n t	hashCode() Returns the hashcode of the Tx identifier.
V o i d	putSlot(int slotNumber, Object o) Sets some context information to be associated with this transaction.

Field Detail

SLOT_NAME

static final [String](#) SLOT_NAME

All slots should be reserved using this name.

See Also:

[ObjectGrid.reserveSlot\(String\)](#), [Constant Field Values](#)

Method Detail

equals

boolean **equals**([TxID](#) o)

Checks for equality between two TxID objects.

Parameters:

o - Input TxID to check for equality against

Returns:

true, if they are equal; false, if they not equal

hashCode

int **hashCode**()

Returns the hashcode of the Tx identifier.

Overrides:

[hashCode](#) in class [Object](#)

Returns:

hashcode

getSlot

[Object](#) `getSlot(int slotNumber)`

Gets the context information currently associated with this transaction.

Parameters:

slotNumber - the slot number for the context information being requested

Returns:

Object the current context information for the slot number

See Also:

[putSlot\(int, Object\)](#), [ObjectGrid.reserveSlot\(String\)](#)

putSlot

void `putSlot(int slotNumber,`
 [Object](#) o)

Sets some context information to be associated with this transaction.

Parameters:

slotNumber - the slot number

o - Object to be put into the TxID slot

See Also:

[getSlot\(int\)](#), [ObjectGrid.reserveSlot\(String\)](#)

getSession

[Session](#) `getSession()`

Returns the Session that owns this TxID.

Returns:

a Session object to use.

See Also:

[Session](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class TransactionTimeoutException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionTimeoutException
extends ObjectGridRuntimeException
```

This exception is thrown when a transaction exceeds the transaction timeout that was specified on the ObjectGrid or Session.

Since:

WAS XD 6.0.1, XC10

See Also:

[ObjectGrid.setTxTimeout\(int\)](#), [Session.setTransactionTimeout\(int\)](#), [Serialized Form](#)

Constructor Summary

- [TransactionTimeoutException](#)([String](#) message, [String](#) txIdString)
Constructs a new TransactionTimeoutException with the specified detail message.
- [TransactionTimeoutException](#)([String](#) message, [Throwable](#) cause)

Method Summary

- | | |
|----------------------------|---|
| S
t
r
i
n
g | getTxIDString ()
Get the String representation of the TXID for the transaction that timed out. |
| D
a
t
e | whenOccurred ()
Gives the time when this TransactionTimeoutException was created. |

Methods inherited from class

com.ibm.websphere.objectgrid.[ObjectGridRuntimeException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionTimeoutException

```
public TransactionTimeoutException(String message,  
                                   Throwable cause)
```

TransactionTimeoutException

```
public TransactionTimeoutException(String message,  
                                   String txIdString)
```

Constructs a new TransactionTimeoutException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

`txIdString` - the result of `TxID.toString()` for the transaction that timed out.

See Also:

[ObjectGridRuntimeException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

Method Detail

whenOccurred

```
public Date whenOccurred()
```

Gives the time when this TransactionTimeoutException was created.

Returns:

Date object that represents the instant in time when this exception object was created.

getTxIDString

```
public String getTxIDString()
```

Get the String representation of the TxID for the transaction that timed out.

Returns:

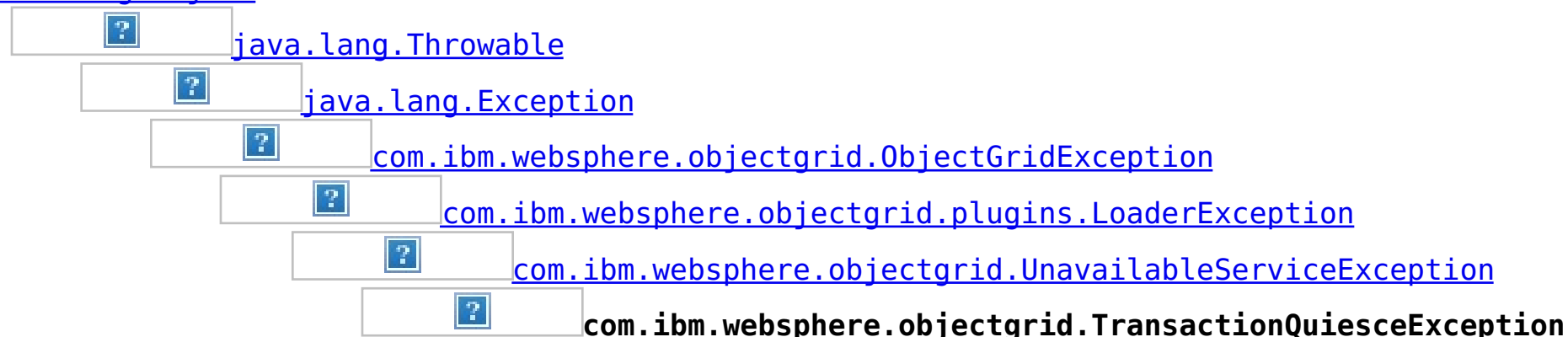
String value of TxID of transaction that timed out.

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class TransactionQuiesceException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionQuiesceException
extends UnavailableServiceException
```

This exception is thrown when partition/shard/mapset/replication group/ replication group member/server/cluster/objectgrid is entered quiesce process for various reasons such as shard movement, partition relocation, system update, server shutdown, and others. Quiesce process ensures data integrity and transaction integrity. This exception is thrown only for new start of a new transaction; it will not impact old transaction requests that are allowed to finish.

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[TransactionQuiesceException](#)()

Constructs a new TransactionQuiesceException with null as its detail message.

[TransactionQuiesceException](#)(String message)

Constructs a new TransactionQuiesceException with the specified detail message.

Method Summary

Methods inherited from class

com.ibm.websphere.objectgrid.[UnavailableServiceException](#)

[getReplicationGroup](#), [setReplicationGroup](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionQuiesceException

```
public TransactionQuiesceException()
```

Constructs a new TransactionQuiesceException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionQuiesceException

```
public TransactionQuiesceException(String message)
```

Constructs a new TransactionQuiesceException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

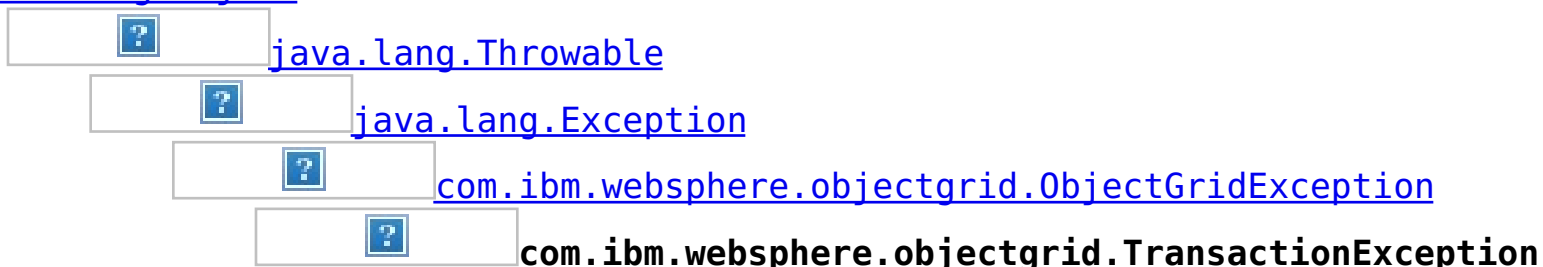
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class TransactionException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[SessionNotReentrantException](#), [TransactionAlreadyActiveException](#)

```
public class TransactionException
extends ObjectGridException
```

A general transaction exception indicating something went wrong with a transaction. The `isTransactionActive()` and `wasTransactionRolledBack()` methods can be used to determine whether transaction is still active or was rolled back as a result of this exception occurring.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Field Summary

protected

[ivTransactionRolledBack](#)

Indicates whether the transaction was rolled back or not.

Constructor Summary

[TransactionException](#)([String](#) message, boolean rolledBack)

Constructs a new TransactionException with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([String](#) message, [Throwable](#) cause, boolean rolledBack)

Constructs a new TransactionException with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([String](#) message, [TransactionException](#) cause, boolean rolledBack)

Constructs a new TransactionException with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([Throwable](#) cause, boolean rolledBack)

Constructs a new TransactionException with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([TransactionException](#) cause, boolean rolledBack)

Constructs a new TransactionException with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception.

Method Summary

[boolean](#)
[isTransactionActive](#)()
Returns true if the transaction is active.

[boolean](#)
[wasTransactionRolledBack](#)()
Returns true if the transaction was rolled back.

Methods inherited from class [com.ibm.websphere.objectgrid.ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

ivTransactionRolledBack

protected boolean **ivTransactionRolledBack**

Indicates whether the transaction was rolled back or not.

Constructor Detail

TransactionException

```
public TransactionException(String message,  
                           boolean rolledBack)
```

Constructs a new `TransactionException` with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.
`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(Throwable cause,  
                           boolean rolledBack)
```

Constructs a new `TransactionException` with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of cause). This constructor is useful for as a wrapper for other `Throwable` objects that occur.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.
`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.getCause\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(TransactionException cause,  
                           boolean rolledBack)
```

Constructs a new `TransactionException` with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of cause). This constructor is useful for as a wrapper for other `Throwable` objects that occur.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.
`rolledBack` - A value of `true` indicates the transaction was rolled back.

Since:

WAS XD 6.1 IFIX1

See Also:

[ObjectGridException.getCause\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(String message,  
                           Throwable cause,  
                           boolean rolledBack)
```

Constructs a new `TransactionException` with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `TransactionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(String message,  
                             TransactionException cause,  
                             boolean rolledBack)
```

Constructs a new `TransactionException` with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `TransactionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

`rolledBack` - A value of `true` indicates the transaction was rolled back.

Since:

WAS XD 6.1 IFIX1

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#), [wasTransactionRolledBack\(\)](#)

Method Detail

isTransactionActive

```
public boolean isTransactionActive()
```

Returns `true` if the transaction is active. Otherwise, `false` is returned to indicate either the transaction never started or was completed.

Returns:

`true` if the transaction is active

wasTransactionRolledBack

```
public boolean wasTransactionRolledBack()
```

Returns `true` if the transaction was rolled back.

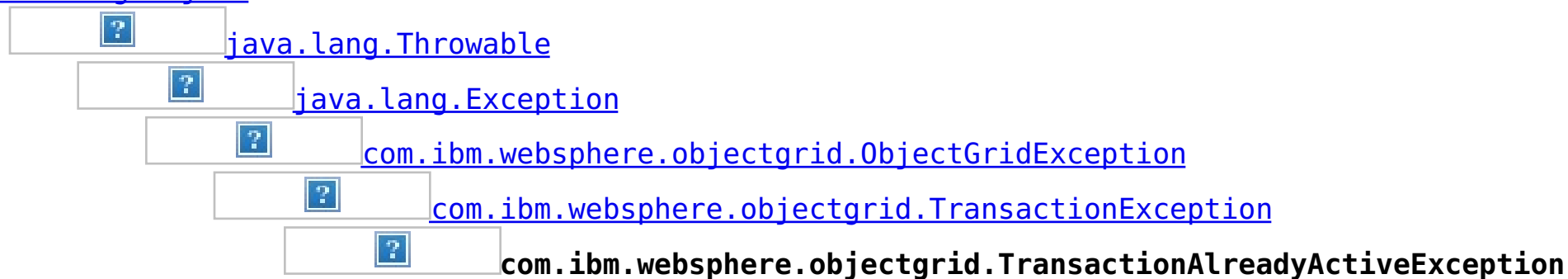
Returns:

`true` if the transaction was rolled back

com.ibm.websphere.objectgrid

Class TransactionAlreadyActiveException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionAlreadyActiveException
extends TransactionException
```

An exception indicating a transaction is already active for the current session. This exception does not cause the current active transaction to be rolled back, so the `isTransactionActive` method will return `true`.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Field Summary

Fields inherited from class `com.ibm.websphere.objectgrid.TransactionException`

[ivTransactionRolledBack](#)

Constructor Summary

[TransactionAlreadyActiveException](#)()

Constructs a new `TransactionAlreadyActiveException` with `null` as its detail message.

[TransactionAlreadyActiveException](#)([String](#) message)

Constructs a new `TransactionAlreadyActiveException` with the specified detail message.

[TransactionAlreadyActiveException](#)([String](#) message, [Throwable](#) cause)

Constructs a new `TransactionAlreadyActiveException` with the specified detail message and cause.

[TransactionAlreadyActiveException](#)([Throwable](#) cause)

Constructs a new `TransactionAlreadyActiveException` with a specified cause.

Method Summary

Methods inherited from class `com.ibm.websphere.objectgrid.TransactionException`

[isTransactionActive](#), [wasTransactionRolledBack](#)

Methods inherited from class `com.ibm.websphere.objectgrid.ObjectGridException`

[getCause](#), [initCause](#)

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException()
```

Constructs a new `TransactionAlreadyActiveException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException(String message)
```

Constructs a new `TransactionAlreadyActiveException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException(Throwable cause)
```

Constructs a new `TransactionAlreadyActiveException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of cause). This constructor is useful for `TransactionAlreadyActiveExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException(String message,  
                                         Throwable cause)
```

Constructs a new TransactionAlreadyActiveException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this TransactionAlreadyActiveException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class TransactionAffinityException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionAffinityException
extends UnavailableServiceException
```

This exception is thrown for inflight transaction when server fails over. We suggest applications to retry transaction.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[TransactionAffinityException](#)()

Constructs a new TransactionAffinityException with null as its detail message.

[TransactionAffinityException](#)(String message)

Constructs a new TransactionAffinityException with the specified detail message.

[TransactionAffinityException](#)(String message, Throwable cause)

Constructs a new TransactionAffinityException with the specified detail message and cause.

[TransactionAffinityException](#)(Throwable cause)

Constructs a new TransactionAffinityException with a specified cause.

Method Summary

Methods inherited from class

com.ibm.websphere.objectgrid.[UnavailableServiceException](#)

[getReplicationGroup](#), [setReplicationGroup](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionAffinityException

```
public TransactionAffinityException()
```

Constructs a new TransactionAffinityException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionAffinityException

```
public TransactionAffinityException(String message)
```

Constructs a new TransactionAffinityException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TransactionAffinityException

```
public TransactionAffinityException(String message,  
                                   Throwable cause)
```

Constructs a new TransactionAffinityException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this TransactionAffinityException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

TransactionAffinityException

```
public TransactionAffinityException(Throwable cause)
```

Constructs a new TransactionAffinityException with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for TransactionAffinityExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes						
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD								
OD								

com.ibm.websphere.objectgrid

Class TargetNotAvailableException

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public class TargetNotAvailableException
extends RuntimeException
```

A TargetNotAvailableException indicates the ObjectGrid target is not available. This could be due to the fact that ObjectGrid servers are not available or the ObjectGrid placement has not finished.

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[TargetNotAvailableException](#)()

Constructs a new TargetNotAvailableException with null as its detail message.

[TargetNotAvailableException](#)([String](#) message)

Constructs a new TargetNotAvailableException with the specified detail message.

[TargetNotAvailableException](#)([String](#) message, [Throwable](#) cause)

Constructs a new TargetNotAvailableException with the specified detail message and cause.

[TargetNotAvailableException](#)([Throwable](#) cause)

Constructs a new TargetNotAvailableException with a specified cause.

Method Summary

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getCause](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [initCause](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TargetNotAvailableException

```
public TargetNotAvailableException()
```

Constructs a new `TargetNotAvailableException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[Throwable.initCause\(Throwable\)](#)

TargetNotAvailableException

```
public TargetNotAvailableException(String message)
```

Constructs a new `TargetNotAvailableException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[Throwable.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TargetNotAvailableException

```
public TargetNotAvailableException(Throwable cause)
```

Constructs a new `TargetNotAvailableException` with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for `TargetNotAvailableExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[Throwable.getCause\(\)](#)

TargetNotAvailableException

```
public TargetNotAvailableException(String message,
                                   Throwable cause)
```

Constructs a new `TargetNotAvailableException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `TargetNotAvailableException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[Throwable.getCause\(\)](#), [Throwable.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class TTLType

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```

public class TTLType
    extends Object
    implements Serializable
  
```

Every BackingMap in ObjectGrid has a built in timed based evictor that is referred to as "time to live" evictor or TTL evictor. Each BackingMap entry has an expiration time that determines how long the entry is allowed to live in the BackingMap. When the expiration time is reached, the TTL evictor causes the expired entry to be evicted from the BackingMap. This class is used to define the TTLType value constants that determine how the the expiration time is computed for a map entry.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.setTtlEvictorType\(TTLType\)](#), [Serialized Form](#)

Field Summary

s t a t i c	<p>CREATION_TIME</p> <p>A TTLType.CREATION_TIME indicates an entry expiration time is the sum of the creation time of the entry plus the "time to live" value.</p>
I T T L T Y P E	
s t a t i c	<p>LAST_ACCESS_TIME</p> <p>A TTLType.LAST_ACCESS_TIME indicates an entry expiration time is the sum of the last access time of the entry plus the "time to live" value.</p>
I T T L T Y P E	
s t a	

t
i
c
T
T
L
T
Y
P
E

[LAST_UPDATE_TIME](#)

A `TTLType.LAST_UPDATE_TIME` indicates an entry expiration time is the sum of the last update time of the entry plus the "time to live" value.

s
t
a
t
i
c
T
T
L
T
Y
P
E

[NONE](#)

A `TTLType.NONE` indicates an entry has no expiration time and is allowed to live in the `BackingMap` until the application explicitly removes or invalidates the entry or a user defined evictor evicts it.

Method Summary

b
y
t
e

[getId\(\)](#)

Get the raw value of this `TTLType`.

S
t
r
i
n
g

[toString\(\)](#)

Returns a string representation of the `TTLType`.

s
t
a
t
i
c
T
T
L
T
Y
P
E

[valueOf\(byte id\)](#)

Given the raw value of a `TTLType`, this method returns a `TTLType` object, or null if the raw value does not match an existing type.

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

NONE

```
public static final TTLType NONE
```

A `TTLType.NONE` indicates an entry has no expiration time and is allowed to live in the `BackingMap` until the application explicitly removes or invalidates the entry or a user defined evictor evicts it.

CREATION_TIME

public static final [TTLType](#) CREATION_TIME

A `TTLType.CREATION_TIME` indicates an entry expiration time is the sum of the creation time of the entry plus the "time to live" value. The "time to live" value is set using the `BackingMap.setTimeToLive(int)` method and is the same for every entry and can **not** be changed by the application by using the `ObjectMap.setTimeToLive(int)` method. It can only be set prior to `ObjectGrid` initialization by use of the `BackingMap.setTimeToLive(int)` method.

See Also:

[BackingMap.setTimeToLive\(int\)](#)

LAST_ACCESS_TIME

public static final [TTLType](#) LAST_ACCESS_TIME

A `TTLType.LAST_ACCESS_TIME` indicates an entry expiration time is the sum of the last access time of the entry plus the "time to live" value. By default, the time to live value is set using the `BackingMap.setTimeToLive(int)` method and the default can be overridden by the application by using the `ObjectMap.setTimeToLive(int)` method.

See Also:

[BackingMap.setTimeToLive\(int\)](#), [ObjectMap.setTimeToLive\(int\)](#)

LAST_UPDATE_TIME

public static final [TTLType](#) LAST_UPDATE_TIME

A `TTLType.LAST_UPDATE_TIME` indicates an entry expiration time is the sum of the last update time of the entry plus the "time to live" value. By default, the time to live value is set using the `BackingMap.setTimeToLive(int)` method and the default can be overridden by the application by using the `ObjectMap.setTimeToLive(int)` method. The difference between this `TTLType` and `LAST_ACCESS_TIME` is that fetch operations do not cause the entry expiration time to be updated.

Since:

7.1

See Also:

[BackingMap.setTimeToLive\(int\)](#), [ObjectMap.setTimeToLive\(int\)](#)

Method Detail

valueOf

public static final [TTLType](#) valueOf(byte id)

Given the raw value of a `TTLType`, this method returns a `TTLType` object, or null if the raw value does not match an existing type. This method is used to deserialize this object.

Parameters:

id - the raw value of a `TTLType`

Returns:

the `TTLType` corresponding to the raw input value

Since:

8.6, XC10 2.5

getId

public byte **getId()**

Get the raw value of this TTLType. This method is used to serialize this object.

Returns:

the raw value of this TTLType.

Since:

8.6, XC10 2.5

toString

public [String](#) **toString()**

Returns a string representation of the TTLType.

Overrides:

[toString](#) in class [Object](#)

Returns:

a string representation of the TTLType.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface StateManager

public interface **StateManager**

The StateManager can be used to retrieve the availability state of an ObjectGrid. Use the StateManagerFactory.getStateManager() method to retrieve a StateManager instance.

Since:

WAS XD 6.1.0.3, XC10

Method Summary

AvailabilityState	getObjectGridState (ObjectGrid objectGrid) Get the AvailabilityState of an ObjectGrid.
ObjectGrid	setObjectGridState (AvailabilityState state, ObjectGrid objectGrid) Set the AvailabilityState for an ObjectGrid.

Method Detail

getObjectGridState

[AvailabilityState](#) getObjectGridState([ObjectGrid](#) objectGrid)

Get the AvailabilityState of an ObjectGrid. A random shard within the ObjectGrid is chosen for reporting availability state.

Parameters:

objectGrid - the availability state of the specified remote ObjectGrid will be retrieved

Returns:

the AvailabilityState of the remote ObjectGrid

Throws:

IllegalArgumentException. - If parameter objectGrid, is either null or it is of type 'LOCAL'. See [ObjectGrid.getObjectGridType\(\)](#).

[TargetNotAvailableException](#) - if there are no active shards for the ObjectGrid specified.

setObjectGridState

```
void setObjectGridState(AvailabilityState state,  
                        ObjectGrid objectGrid)
```

Set the AvailabilityState for an ObjectGrid. Each shard in the ObjectGrid will be transitioned to the state specified. This method does not return until each shard in the ObjectGrid has transitioned to the AvailabilityState specified or if it times-out.

Parameters:

state - the AvailabilityState to transition to.

objectGrid - the ObjectGrid to transaction to the specified AvailabilityState.

Throws:

IllegalArgumentException. -

1. If parameter ObjectGrid. is either null or is of type 'LOCAL'. See [ObjectGrid.getObjectGridType\(\)](#).
2. If parameter AvailabilityState is null.

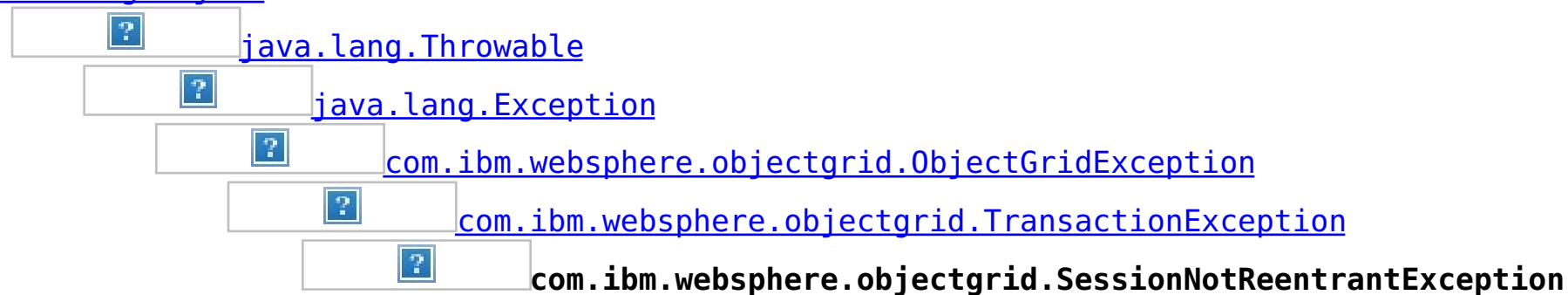
Overview	Package	Classes	Serialized	Deprecated	Index	Help
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes				
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD						
OD						

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class SessionNotReentrantException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class SessionNotReentrantException
extends TransactionException
```

A Session object can only be used by a single thread concurrently to perform map operations. If a thread tries to execute a map operation (for example, call a method on ObjectMap interface) while another thread is already executing a map operation for the Session, then this exception is thrown.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Field Summary

Fields inherited from class com.ibm.websphere.objectgrid.[TransactionException](#)

[ivTransactionRolledBack](#)

Constructor Summary

[SessionNotReentrantException](#)([String](#) message, boolean rolledBack)

Constructs a new SessionNotReentrantException with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[TransactionException](#)

[isTransactionActive](#), [wasTransactionRolledBack](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

SessionNotReentrantException

```
public SessionNotReentrantException(String message,  
                                   boolean rolledBack)
```

Constructs a new SessionNotReentrantException with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [TransactionException.wasTransactionRolledBack\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid
Interface Session

public interface **Session**

This interface represents a session container for ObjectMaps. A thread must get its own Session object to interact with ObjectGrid. You can think of this interface as a session that can only be used by a single thread at a time. A Session itself is shareable across threads so long as only one thread uses it at a time. However, if a J2EE connection/transaction infrastructure is being used, that won't be shareable across threads and will prevent the Session object from being shared across threads. A good analogy for this object is a JDBC connection to a database. For best performance, use the [close\(\)](#) method to close the session once it is no longer required.

Since:

WAS XD 6.0, XC10

See Also:

[ObjectGrid.getSession\(\)](#), [ObjectGrid.getSession\(Subject\)](#),
[ObjectGrid.getSession\(CredentialGenerator\)](#)

Nested Class Summary

s t a t i c c l a s s	<p>Session.TxCommitProtocol The commit protocols that can be used to commit the Session's transaction</p>
---	---

Field Summary

s t a t i c	<p>DEFAULT_RETRY_TIMEOUT</p>
s t a t i c	<p>TRANSACTION_NO_TIMEOUT A special value for the timeout parameter of the setTransactionTimeout(int) method.</p>

s t a t i c i n t	<p>TRANSACTION_READ_COMMITTED</p> <p>A transaction isolation level constant indicating that dirty reads are prevented; non-repeatable reads and phantom reads can occur.</p>
s t a t i c i n t	<p>TRANSACTION_READ_UNCOMMITTED</p> <p>A transaction isolation level constant indicating that dirty reads, non-repeatable reads and phantom reads can occur.</p>
s t a t i c i n t	<p>TRANSACTION_REPEATABLE_READ</p> <p>A transaction isolation level constant indicating that dirty reads and non-repeatable reads are prevented; phantom reads can occur.</p>
s t a t i c S t r i n g	<p>TRANSACTION_TYPE_DEFAULT</p> <p>A string indicating the default transaction type</p>

Method Summary

v o i d	<p>begin()</p> <p>Begins a new transaction.</p>
v o i d	<p>beginNoWriteThrough()</p> <p>Starts a new transaction that does not write changes through to a Loader or ObjectGrid server.</p>
v o i d	<p>close()</p> <p>Closes this session, freeing all resources that are held.</p>
v o i d	<p>commit()</p> <p>Commits a transaction.</p>
c o m . i b	

m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
q
u
e
r
y
.
O
b
j
e
c
t
Q
u
e
r
y

[createObjectQuery\(String qlString\)](#)

Creates an instance of an object query for executing a query over the ObjectMaps visible to this session.

v
o
i
d

[flush\(\)](#)

Forces the current changes in the Session to the Loader or ObjectGrid server.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
e
m

[getEntityManager\(\)](#)

Retrieve the EntityManager associated with this Session.

EntityManager

ObjectMap

[getMap\(String cacheName\)](#)
Returns the ObjectMap for the specified name.

ObjectGrid

[getObjectGrid\(\)](#)
Returns the ObjectGrid that owns this session.

long

[getRequestRetryTimeout\(\)](#)
Retrieves the current request retry timeout for this session.

com.ibm.websphere.objgrid.Session

[getSessionHandle\(\)](#)
Retrieves a handle for this session.

o
n
H
a
n
d
l
e

i
n
t

[getTransactionIsolation\(\)](#)
Retrieves the current transaction isolation level for this session.

i
n
t

[getTransactionTimeout\(\)](#)
Gets current transaction timeout for this Session.

S
t
r
i
n
g

[getTransactionType\(\)](#)
Retrieves the transaction type that is set with the [setTransactionType\(String\)](#) method.

S
e
s
s
i
o
n
.
T
x
C
o
m
m
i
t
P
r
o
t
o
c
o
l

[getTxCommitProtocol\(\)](#)
Retrieve the current commit protocol for this Session.

T
x
I
D

[getTxID\(\)](#)
Gets the TxID transaction identifier, if a transaction is active.

b
o
o
l
e
a
n

[isCommitting\(\)](#)
Returns whether the current session transaction is performing a session `commit()` operation.

b
o
o
l
e
a
n

[isFlushing\(\)](#)
Returns whether the current session transaction is performing a session `flush()` operation.

b
o
o

[isMarkedRollbackOnly\(\)](#)
Returns whether or not the current active session transaction is marked as being

l e a n	rollback only as a result of a prior call to the <code>markRollbackOnly(Throwable)</code> method on this Session.
b o o l e a n	<code>isSessionHandleSet()</code> Determines if a SessionHandle is currently set on this Session.
b o o l e a n	<code>isTransactionActive()</code> Determines if a transaction is currently active.
b o o l e a n	<code>isWriteThroughEnabled()</code> Returns whether the current session transaction is writing through to the back end Loader or ObjectGrid server(<code>true</code>), or if the changes are only applying to the BackingMap (<code>false</code>) or client respectively.
v o i d	<code>markRollbackOnly(Throwable t)</code> Marks the current transaction as being rollback only.
v o i d	<code>processLogSequence(LogSequence logSequence)</code> Processes a LogSequence.
v o i d	<code>rollback()</code> Rolls back a transaction.
v o i d	<code>setRequestRetryTimeout(long requestRetryTimeout)</code> Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions.
v o i d	<code>setSessionHandle(com.ibm.websphere.objectgrid.SessionHandle target)</code> Apply a SessionHandle to this session.
v o i d	<code>setTransactionIsolation(int level)</code> Attempts to change the transaction isolation level for this session.
v o i d	<code>setTransactionTimeout(int timeout)</code> Sets the transaction timeout for the next transaction started by this Session object to a specified number of seconds.
v o i d	<code>setTransactionType(String tranType)</code> Sets the transaction type for future transactions.
v o	<code>setTxCommitProtocol(Session.TxCommitProtocol protocol)</code>

i d	Set the commit protocol to be used when committing this Session's transaction.
b o o l l e a n	transactionTimedOut() Determines whether the current session transaction has timed out.

Field Detail

TRANSACTION_TYPE_DEFAULT

static final [String](#) TRANSACTION_TYPE_DEFAULT

A string indicating the default transaction type

See Also:

[Constant Field Values](#)

TRANSACTION_NO_TIMEOUT

static final int TRANSACTION_NO_TIMEOUT

A special value for the timeout parameter of the `setTransactionTimeout(int)` method. This special value is used to indicate that the next transaction started by this Session is allowed unlimited amount of time.

See Also:

[setTransactionTimeout\(int\)](#), [Constant Field Values](#)

DEFAULT_RETRY_TIMEOUT

static final long DEFAULT_RETRY_TIMEOUT

See Also:

[Constant Field Values](#)

TRANSACTION_REPEATABLE_READ

static final int TRANSACTION_REPEATABLE_READ

A transaction isolation level constant indicating that dirty reads and non-repeatable reads are prevented; phantom reads can occur. This level prohibits a transaction from reading an uncommitted cache entry, and it also prohibits the situation where one transaction reads an entry, a second transaction alters the entry, and the first transaction rereads the entry, getting different values the second time (a "non-repeatable read").

Since:

WAS XD 6.1.0.1

See Also:

[setTransactionIsolation\(int\)](#), [Constant Field Values](#)

TRANSACTION_READ_COMMITTED

static final int TRANSACTION_READ_COMMITTED

A transaction isolation level constant indicating that dirty reads are prevented; non-repeatable reads and phantom reads can occur. This level only prohibits a transaction from reading a cache entry with uncommitted changes in it.

Since:

WAS XD 6.1.0.1

See Also:

[setTransactionIsolation\(int\)](#), [Constant Field Values](#)

TRANSACTION_READ_UNCOMMITTED

static final int TRANSACTION_READ_UNCOMMITTED

A transaction isolation level constant indicating that dirty reads, non-repeatable reads and phantom reads can occur. This level allows a cache entry changed by one transaction to be read by another transaction before any changes in that entry have been committed (a "dirty read"). If any of the changes are rolled back, the second transaction will have retrieved an invalid entry.

Since:

WAS XD 6.1.0.1

See Also:

[setTransactionIsolation\(int\)](#), [Constant Field Values](#)

Method Detail

beginNoWriteThrough

```
void beginNoWriteThrough()  
    throws TransactionAlreadyActiveException,  
           TransactionException
```

Starts a new transaction that does not write changes through to a Loader or ObjectGrid server.

Changes made by the session transaction started by this method are only applied to the BackingMap and not given to the Loader. This method can be used to apply changes made in a peer cache to the local BackingMap only. In addition, with a distributed map, this method can be used to start a session transaction which changes will only be applied to the client BackingMap, but not the BackingMap on the server side.

Throws:

[TransactionAlreadyActiveException](#) - if there is already an active transaction
[TransactionException](#) - a TransactionCallbackException occurred or some other error occurred starting a new transaction

getMap

```
ObjectMap getMap(String cacheName)  
    throws UndefinedMapException
```

Returns the ObjectMap for the specified name.

The ObjectMap is used to retrieve and modify values in the BackingMap. Multiple invocations of this method on the same Session object will always return the same object.

This method can also be used to create a BackingMap and its associated ObjectGrid after ObjectGrid initialization. If cacheName does not match the name of a previously created

map, a name comparison will be executed against template maps that have been configured. The ObjectMap and BackingMap will be created if the name matches the regular expression of a template.

Required Client Permission: `ObjectGridPermission.DYNAMIC_MAP` (when creating a new map from a template)

Parameters:

cacheName - name of desired map

Returns:

ObjectMap the transactional interface to modify values in the map

Throws:

[UndefinedMapException](#) - if the map is not defined.

See Also:

[ObjectGrid.defineMap\(String\)](#), [ObjectMap](#)

begin

```
void begin()  
    throws TransactionAlreadyActiveException,  
           TransactionException
```

Begins a new transaction.

Throws:

[TransactionAlreadyActiveException](#) - if this method is invoked with an active transaction
[TransactionException](#) - a TransactionCallbackException occurred or some other error occurred starting a new transaction

commit

```
void commit()  
    throws NoActiveTransactionException,  
           TransactionException
```

Commits a transaction.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction
[TransactionException](#) - if an error occurred during commit processing, see the caused by to determine the root error

See Also:

[markRollbackOnly\(Throwable\)](#)

rollback

```
void rollback()  
    throws NoActiveTransactionException,  
           TransactionException
```

Rolls back a transaction.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction
[TransactionException](#) - if an error occurred during rollback processing, see the caused by to determine the root error

flush

void **flush()**
throws [NoActiveTransactionException](#),
[TransactionException](#)

Forces the current changes in the Session to the Loader or ObjectGrid server. This method does not commit the changes, it just applies the changes.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction
[TransactionException](#) - if an error occurred during flush processing, see the caused by to determine the root error

getObjectGrid

[ObjectGrid](#) getObjectGrid()

Returns the ObjectGrid that owns this session.

Returns:

the owning ObjectGrid instance.

isTransactionActive

boolean isTransactionActive()

Determines if a transaction is currently active.

Returns:

true if a transaction is currently active for this session.

Since:

WAS XD 6.1 FIX3

getTxID

[TxID](#) getTxID()
throws [NoActiveTransactionException](#)

Gets the TxID transaction identifier, if a transaction is active.

Returns:

The current TxID object.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction

isWriteThroughEnabled

boolean isWriteThroughEnabled()

Returns whether the current session transaction is writing through to the back end Loader or ObjectGrid server(true), or if the changes are only applying to the BackingMap (false) or client respectively.

Returns:

true, if write through is enabled

See Also:

[begin\(\)](#), [beginNoWriteThrough\(\)](#)

setTransactionType

void **setTransactionType**([String](#) tranType)

Sets the transaction type for future transactions.

After this method is called, all future transactions will have the same type until another transaction type is set. If no transaction type is set, the default transaction type TRANSACTION_TYPE_DEFAULT will be used.

Transaction types are used mainly for statistical data tracking purpose. Users can predefine types of transactions that will be executed in an application. The idea is to categorize transactions with the same characteristics to one category (type), so one transaction response time statistics can be used to track each transaction type. This approach is useful when your application has different types of transactions. Some types of transactions, such as update transactions, process longer than others transactions, such as read-only transactions. By using the transaction type, different transactions are tracked by different statistics, so the statistics can be more useful.

Parameters:

tranType - the transaction type for future transactions.

See Also:

[TRANSACTION_TYPE_DEFAULT](#)

getTransactionType

[String](#) **getTransactionType**()

Retrieves the transaction type that is set with the [setTransactionType\(String\)](#) method.

Returns:

the transaction type for the session.

Since:

7.1.1.1

processLogSequence

void **processLogSequence**([LogSequence](#) logSequence)
throws [NoActiveTransactionException](#),
[UndefinedMapException](#),
[ObjectGridException](#)

Processes a LogSequence.

Each LogElement within the LogSequence will be examined and the appropriate operation (insert, update, invalidate, etc) will be performed against the BackingMap identified by the LogSequence's map name. An ObjectGrid Session must be active before this method is invoked. The caller is responsible for issuing the appropriate commit or rollback invocation to complete the Session. Autocommit processing is not available for this method invocation.

The main use of this method is for processing a LogSequence that was received by a remote JVM. For example, using the Distributed Commit support, the LogSequences associated with a given committed Session are distributed to other listening ObjectGrids in other JVMs. After receiving the LogSequences at the remote JVM, the listener could start a Session using beginNoWriteThrough(), invoke this method, and commit the Session transaction.

Parameters:

logSequence - LogSequence of changes to be applied to an active transaction

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction

[UndefinedMapException](#) - if the map referenced by the LogSequence cannot be found

[ObjectGridException](#) - if the LogSequence elements cannot be processed

See Also:

[beginNoWriteThrough\(\)](#), [LogSequence](#), [ObjectGridEventListener](#)

isFlushing

boolean **isFlushing**()

Returns whether the current session transaction is performing a session `flush()` operation. It is helpful to know if a session `flush()` is active (`true`), or if only an `ObjectMap.flush()` is in progress (returns `false` in this case).

Returns:

`true`, if the session is executing a session `flush()` call.

Since:

WAS XD 6.0.1

See Also:

[flush\(\)](#), [ObjectMap.flush\(\)](#)

isCommitting

boolean **isCommitting**()

Returns whether the current session transaction is performing a session `commit()` operation. It is helpful to know if a session `commit` is active (`true`), or if an `ObjectMap.flush()` or session `flush()` is in progress (returns `false` in these cases).

Returns:

`true`, if session is executing a session `commit()` call.

Since:

WAS XD 6.0.1

See Also:

[commit\(\)](#), [flush\(\)](#), [ObjectMap.flush\(\)](#)

markRollbackOnly

void **markRollbackOnly**([Throwable](#) t)
throws [NoActiveTransactionException](#)

Marks the current transaction as being rollback only.

Marking a transaction rollback only ensures that even if the `commit()` method is called for this session transaction, the transaction is rolled back. A rollback is typically done when either ObjectGrid itself or the application knows that data corruption could occur if the `commit()` method was allowed to commit the transaction. Once this method is called, the `Throwable` object that is passed to it is chained to the `TransactionException` that is thrown if the `commit` method is ever called. Any subsequent calls to this method for the current active transaction is ignored (e.g. only the first call that passes a non null `Throwable` reference is used). Once the transaction is completed, the rollback only mark is removed so that the next transaction that is started using this session can be committed.

Parameters:

t - the `Throwable` that caused this method to be called.

Throws:

[NoActiveTransactionException](#) - if there is no active transaction for this Session.

Since:

WAS XD 6.0.1

See Also:

[commit\(\)](#), [TransactionException](#)

isMarkedRollbackOnly

boolean `isMarkedRollbackOnly()`

Returns whether or not the current active session transaction is marked as being rollback only as a result of a prior call to the `markRollbackOnly(Throwable)` method on this Session.

Returns:

true if and only if current session transaction is marked rollback only.

Since:

WAS XD 6.0.1

See Also:

[markRollbackOnly\(Throwable\)](#)

setTransactionTimeout

void `setTransactionTimeout(int timeout)`

Sets the transaction timeout for the next transaction started by this Session object to a specified number of seconds.

This method does not affect the transaction timeout of any transactions previously started by this Session. It only affects transactions that are started after this method is called. If this method is never called, the ObjectGrid configured transaction timeout value is used.

Parameters:

`timeout` - is the transaction timeout value in seconds. Use the special value `TRANSACTION_NO_TIMEOUT` if transaction is allowed unlimited amount of time and no transaction timeout should occur.

Since:

WAS XD 6.0.1

See Also:

[TRANSACTION_NO_TIMEOUT](#), [ObjectGrid.setTxTimeout\(int\)](#), [TransactionTimeoutException](#)

getTransactionTimeout

int `getTransactionTimeout()`

Gets current transaction timeout for this Session.

The transaction timeout value returned is the value that was configured for the ObjectGrid using `ObjectGrid.setTxTimeout(int)` or the value passed to `setTransactionTimeout(int)` to override the value configured on ObjectGrid. The return value is in seconds.

Returns:

timeout value in seconds.

Since:

WAS XD 6.0.1

See Also:

[setTransactionTimeout\(int\)](#), [ObjectGrid.setTxTimeout\(int\)](#)

getTransactionIsolation

int `getTransactionIsolation()`

Retrieves the current transaction isolation level for this session.

Returns:

one of the following Session constants: [TRANSACTION_READ_UNCOMMITTED](#), [TRANSACTION_READ_COMMITTED](#) or [TRANSACTION_REPEATABLE_READ](#)

Since:

WAS XD 6.1.0.1

transactionTimedOut

boolean **transactionTimedOut**()

Determines whether the current session transaction has timed out.

Returns:

true if and only if transaction has timed out.

Since:

WAS XD 6.0.1

See Also:

[setTransactionTimeout\(int\)](#)

createObjectQuery

com.ibm.websphere.objectgrid.query.ObjectQuery **createObjectQuery**([String](#) qlString)
throws com.ibm.websphere.objectgrid.query.ObjectQueryException

Creates an instance of an object query for executing a query over the ObjectMaps visible to this session.

When ObjectGrid security is enabled, this method requires an `com.ibm.websphere.objectgrid.security.ObjectGridPermission` with action "query".

Required Client Permission: `ObjectGridPermission.QUERY`

Parameters:

qlString - a query string

Returns:

the new query instance.

Throws:

`com.ibm.websphere.objectgrid.query.ObjectQueryException` - if an error occurs creating the object query.

Since:

WAS XD 6.1

getEntityManager

com.ibm.websphere.objectgrid.em.EntityManager **getEntityManager**()

Retrieve the EntityManager associated with this Session. Each session is associated with a single EntityManager instance. Repeated calls to this method on the same Session instance will result in the same EntityManager instance.

Returns:

this session's EntityManager instance.

Since:

WAS XD 6.1

setTransactionIsolation

void **setTransactionIsolation**(int level)

Attempts to change the transaction isolation level for this session. The constants defined in the Session interface are the possible transaction isolation levels.

This method should normally be invoked prior to beginning a transaction. Invoking after a transaction has started may result in an exception.

Parameters:

level - one of the following Session constants: [TRANSACTION_READ_UNCOMMITTED](#), [TRANSACTION_READ_COMMITTED](#) or [TRANSACTION_REPEATABLE_READ](#)

Since:

WAS XD 6.1.0.1

getSessionHandle

com.ibm.websphere.objectgrid.SessionHandle **getSessionHandle**()

Retrieves a handle for this session.

A SessionHandle contains partition information for the current session and can be re-applied to a new session using the [setSessionHandle\(SessionHandle\)](#) method. A SessionHandle is only applicable for ObjectGrids using per-container partition placement. If [setSessionHandle\(SessionHandle\)](#) is not called before invoking this method, a Session Handle is selected using the properties configured in the [ClientProperties](#). If there are no per-container partition placement mapsets or more than one in the ObjectGrid, an `IllegalStateException` is thrown.

Returns:

the SessionHandle for this session

Throws:

[IllegalStateException](#) - if this method is called in an invalid environment.

Since:

WAS XD 6.1.0.3

setSessionHandle

void **setSessionHandle**(com.ibm.websphere.objectgrid.SessionHandle target)
throws [TargetNotAvailableException](#)

Apply a SessionHandle to this session.

Parameters:

target - the SessionHandle to apply or null to disassociate a SessionHandle from this session.

Throws:

[TargetNotAvailableException](#) - when the target is no longer available.

[IllegalStateException](#) - if the Session has modified some maps already and the SessionHandle has already been set or if this method is called in an invalidate environment.

Since:

WAS XD 6.1.0.3

setRequestRetryTimeout

void **setRequestRetryTimeout**(long requestRetryTimeout)

Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions. A request will timeout

when either the request timeout expires or the transaction timeout expires, whichever expires first.

A value of 0 indicates that all requests should fail immediately and avoid any retry logic. Exceptions that cannot succeed even if tried again such as `DuplicateKeyException` exceptions will be thrown immediately.

A value of -1 indicates that the request retry timeout is not set, meaning that the request duration is governed by the request retry timeout set on the `ClientProperties`. If the `ClientProperties` is also set to -1, then the request retry timeout is governed by the transaction timeout.

Parameters:

`requestRetryTimeout` - the duration in milliseconds retry a client request, 0 if the request should fail immediately or -1 if the request timeout is not set.

Since:

7.0

See Also:

[ClientProperties.setRequestRetryTimeout\(long\)](#), [setTransactionTimeout\(int\)](#)

getRequestRetryTimeout

long `getRequestRetryTimeout()`

Retrieves the current request retry timeout for this session. Returns -1 if it was not set.

Returns:

the duration in milliseconds retry a client request, 0 if the request should fail immediately or -1 if the request timeout is not set.

Since:

7.0

isSessionHandleSet

boolean `isSessionHandleSet()`

Determines if a `SessionHandle` is currently set on this `Session`.

Returns:

true if a `SessionHandle` is currently set on this session.

Since:

7.1

close

void `close()`

Closes this session, freeing all resources that are held. Once closed, this session must be discarded. Use one of the [ObjectGrid.getSession\(\)](#) methods to retrieve a new session. If the session has an active transaction, the transaction will be rolled back and the session resources are not freed.

Throws:

[ObjectGridRuntimeException](#) - thrown if there is a problem releasing resources held by this session.

Since:

7.1.1

setTxCommitProtocol

void **setTxCommitProtocol**([Session.TxCommitProtocol](#) protocol)

Set the commit protocol to be used when committing this Session's transaction. The constants defined in the TxCommitProtocol enum are the possible commit protocols.

This method should normally be invoked prior to beginning a transaction. Invoking after a transaction has started will result in an exception.

Parameters:

protocol - one of the following constants TxCommitProtocol.ONEPHASE or TxCommitProtocol.TWOPHASE

Since:

8.6, XC10 2.5

getTxCommitProtocol

[Session.TxCommitProtocol](#) **getTxCommitProtocol**()

Retrieve the current commit protocol for this Session.

Returns:

one of the following constants TxCommitProtocol.ONEPHASE or TxCommitProtocol.TWOPHASE

Since:

8.6, XC10 2.5

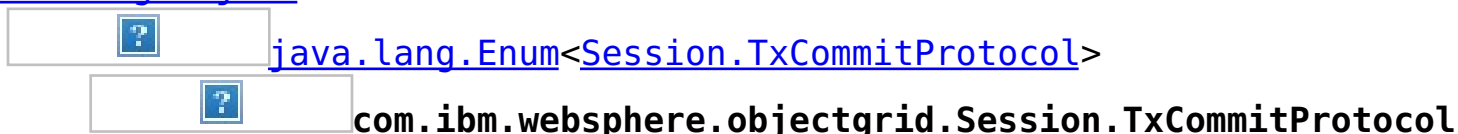
Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

[OD](#)

com.ibm.websphere.objectgrid

Enum Session.TxCommitProtocol

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#), [Comparable](#)<[Session.TxCommitProtocol](#)>

Enclosing interface:

[Session](#)

```
public static enum Session.TxCommitProtocol  
extends Enum<Session.TxCommitProtocol>
```

The commit protocols that can be used to commit the Session's transaction

Since:

8.6, XC10 2.5

Enum Constant Summary

[ONEPHASE](#)

A commit protocol constant indicating that the Session transaction can read from multiple partitions but can only write to a single partition.

[TWOPHASE](#)

A commit protocol constant indicating that the Session transaction can read and write from multiple partitions.

Method Summary

s
t
a
t
i
c
S
e
s
s
i
o
n
.
T
x
C
o
m
m
i
t

[valueOf](#)([String](#) name)

Returns the enum constant of this type with the specified name.

P
r
o
t
o
c
o
l

S
t
a
t
i
c
S
e
s
s
i
o
n
.
T
x
C
o
m
m
i
t
P
r
o
t
o
c
o
l
[
]

[values\(\)](#)

Returns an array containing the constants of this enum type, in the order they are declared.

Methods inherited from class [java.lang.Enum](#)

[clone](#), [compareTo](#), [equals](#), [finalize](#), [getDeclaringClass](#), [hashCode](#), [name](#), [ordinal](#), [toString](#), [valueOf](#)

Methods inherited from class [java.lang.Object](#)

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Enum Constant Detail

ONEPHASE

public static final [Session.TxCommitProtocol](#) **ONEPHASE**

A commit protocol constant indicating that the Session transaction can read from multiple partitions but can only write to a single partition. A `TransactionException` is thrown if the transaction writes to multiple partitions. The transaction is committed using the one-phase commit protocol.

TWOPHASE

public static final [Session.TxCommitProtocol](#) **TWOPHASE**

A commit protocol constant indicating that the Session transaction can read and write from multiple partitions. The transaction is committed using the two-phase commit protocol. If the transaction only writes to a single partition then the transaction is

committed using the one-phase commit protocol.

Method Detail

values

```
public static Session.TxCommitProtocol[] values()
```

Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows:

```
for (Session.TxCommitProtocol c : Session.TxCommitProtocol.values())  
    System.out.println(c);
```

Returns:

an array containing the constants of this enum type, in the order they are declared

valueOf

```
public static Session.TxCommitProtocol valueOf(String name)
```

Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

Parameters:

name - the name of the enum constant to be returned.

Returns:

the enum constant with the specified name

Throws:

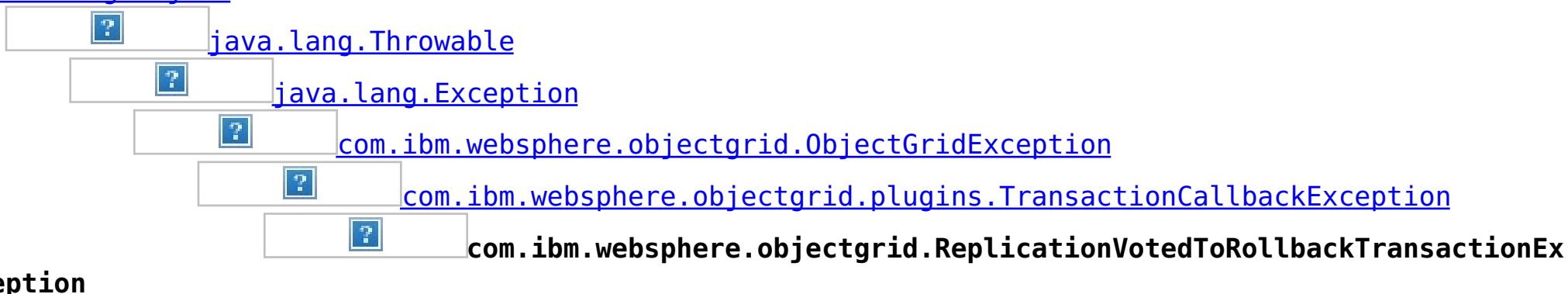
[IllegalArgumentException](#) - if this enum type has no constant with the specified name
[NullPointerException](#) - if the argument is null

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All			
SUMMARY: NESTED ENUM			DETAIL: ENUM					
CONSTANTS FIELD METHOD			CONSTANTS FIELD METHOD					

com.ibm.websphere.objectgrid

Class **ReplicationVotedToRollbackTransactionException**

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class ReplicationVotedToRollbackTransactionException
extends TransactionCallbackException
```

This exception is thrown when a transaction was rolled back because some/all of the replicas failed to apply the transaction when in synchronous replication mode.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ReplicationVotedToRollbackTransactionException](#)()

Constructs a new `ReplicationVotedToRollbackTransactionException` with `null` as its detail message.

[ReplicationVotedToRollbackTransactionException](#)([String](#) message)

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message.

[ReplicationVotedToRollbackTransactionException](#)([String](#) message, [Throwable](#) cause)

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message and cause.

[ReplicationVotedToRollbackTransactionException](#)([Throwable](#) cause)

Constructs a new `ReplicationVotedToRollbackTransactionException` with a specified cause.

Method Summary

Methods inherited from class `com.ibm.websphere.objectgrid.ObjectGridException`

[getCause](#), [initCause](#)

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException()
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException(String message)
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException(String message,
                                                       Throwable cause)
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ReplicationVotedToRollbackTransactionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException(Throwable cause)
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ReplicationVotedToRollbackTransactionExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

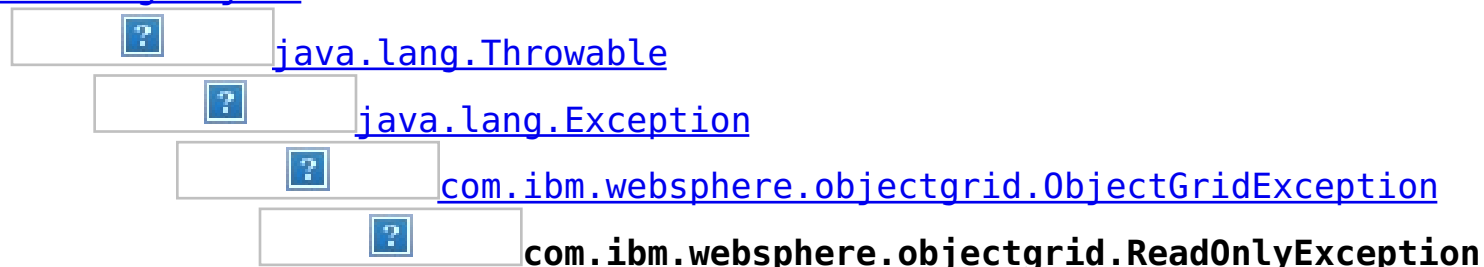
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class ReadOnlyException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class ReadOnlyException
extends ObjectGridException
```

This exception is thrown when an attempt is made to modifying operations on a read only maps.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ReadOnlyException](#)()

Constructs a new ReadOnlyException with null as its detail message.

[ReadOnlyException](#)([String](#) message)

Constructs a new ReadOnlyException with the specified detail message.

[ReadOnlyException](#)([String](#) message, [Throwable](#) cause)

Constructs a new ReadOnlyException with the specified detail message and cause.

[ReadOnlyException](#)([Throwable](#) cause)

Constructs a new ReadOnlyException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ReadOnlyException

```
public ReadOnlyException()
```

Constructs a new `ReadOnlyException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ReadOnlyException

```
public ReadOnlyException(String message)
```

Constructs a new `ReadOnlyException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ReadOnlyException

```
public ReadOnlyException(String message,
                          Throwable cause)
```

Constructs a new `ReadOnlyException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ReadOnlyException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ReadOnlyException

```
public ReadOnlyException(Throwable cause)
```

Constructs a new `ReadOnlyException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ReadOnlyExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Interface PartitionManager

public interface **PartitionManager**

This interface will be used for calculating the proper partition for a given input key. The set of partitions is determined by the BackingMap configuration.

Since:

WAS XD 6.0.1, XC10

See Also:

[BackingMap.getPartitionManager\(\)](#)

Method Summary

int	getNumOfPartitions() Returns the number of configured partitions for this PartitionManager.
int	getPartition(Object key) Obtains a 0-based partition number determined by the input parameter's hashCode() method.
List	getPartitionLists(List keyList) This method is very similar to getPartitions(List), except it returns the keys organized by the partition identifiers.
List	getPartitions(List keyList) Obtains the 0-based partition numbers for each of the keys in the input List of keys.
List	partitionLogSequence(LogSequence ls) Partitions a LogSequence based on the partitioning algorithm for the Map.

Method Detail

getPartition

int **getPartition**([Object](#) key)

Obtains a 0-based partition number determined by the input parameter's hashCode() method.

Parameters:

key - Individual key used to determine partition (can not be null)

Returns:

int 0-based partition number

getPartitions

[List](#) `getPartitions(List keyList)`

Obtains the 0-based partition numbers for each of the keys in the input `List` of keys. Each object in the returned list of partition identifiers is an instance of `java.lang.Integer`.

Parameters:

`keyList` - Ordered list of keys

Returns:

List of partition identifiers that corresponds to the input list of keys

getPartitionLists

[List](#) `getPartitionLists(List keyList)`

This method is very similar to `getPartitions(List)`, except it returns the keys organized by the partition identifiers. The return value is a `List` of `Lists`. The outer `List` is an ordered `List` of the partition numbers, with the first entry in the `List` corresponding to partition 0. The inner `Lists` contain the keys from the input parameter that correspond to that partition identifier.

The return value will always contain a `List` object. Either the outer or the inner `Lists` may contain zero elements, but the `List` objects themselves will not be `null`.

Parameters:

`keyList` - Ordered list of keys

Returns:

List of `Lists` that correspond to the 0-based partition numbers, with each inner `List` containing the set of keys that parse to that partition number.

partitionLogSequence

[List](#) `partitionLogSequence(LogSequence ls)`

Partitions a `LogSequence` based on the partitioning algorithm for the `Map`.

Parameters:

`ls` - `LogSequence` that needs to be partitioned

Returns:

List of partitioned `LogSequences`. The first `LogSequence` in the `List` corresponds to the first partition, etc.

getNumOfPartitions

`int` `getNumOfPartitions()`

Returns the number of configured partitions for this `PartitionManager`.

Returns:

the number of configured partitions

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) | DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid

Interface ObjectMap

```
public interface ObjectMap
```

This is a handle to a named Map. Maps should have homogeneous keys and values. An instance of this ObjectMap can only be used by the thread that is currently associated with the Session that was used to get this ObjectMap instance. Both Session and ObjectMap objects are not allowed to be shared by multiple threads concurrently.

The CopyMode setting on the map determines whether or not a copy of the value is returned by get methods. It also determines whether or not a copy of the committed value is made at commit time. The LockStrategy setting for the map determines whether or not a lock is obtained for each map entry accessed by the transaction, the lock mode of the lock obtained, and when the lock is obtained.

Each data access method includes a "Specification details" table that includes the following information:

Required permission: The permission required to use the API.

Pessimistic read lock acquired: The type of lock that is acquired when using pessimistic locking with repeatable read transaction isolation.

Pessimistic read lock held: The type of lock that is held for the duration of the transaction when using pessimistic locking with repeatable read transaction isolation. Locks can be upgraded but not demoted during a transaction.

Transaction: The state of the transaction when invoking the API.

- **Manual** - The transaction is explicitly demarcated using the associated [Session](#) and the specified API is used within the scope of that transaction.
- **Automatic** - The transaction is automatically demarcated. This is also referred to as an auto-commit transaction or API which the associated [Session.isTransactionActive\(\)](#) is false when invoking the API.

Cache tier: Identifies the map cache tiers that are included when fetching or updating cache entries in the call and under what circumstances.
The following tiers are available for client maps:

- Transactional Cache
- Near Cache (if enabled)
- Server Cache
- Loader (if enabled)

The following tiers are available for local maps:

- Transactional Cache
- Local Cache
- Loader (if enabled)

Since:

WAS XD 6.0, XC10

See Also:

[Session.getMap\(String\)](#), [BackingMap.setCopyMode\(CopyMode, Class\)](#),
[BackingMap.setLockStrategy\(LockStrategy\)](#)

Nested Class Summary

s
t
a
t
i
c
c
l
a
s
s

[ObjectMap.PutMode](#)

Identifies the operation mode of the [put\(Object, Object\)](#), [putAll\(Map\)](#), [JavaMap.put\(Object, Object\)](#) and [JavaMap.putAll\(Map\)](#) methods.

Field Summary

s
t
a
t
i
c
l
o
n
g

[QUEUE_TIMEOUT_INFINITE](#)

Used as a parameter on the [getNextKey\(long\)](#) method, specifies the method should block until a key becomes available.

s
t
a
t
i
c
l
o
n
g

[QUEUE_TIMEOUT_NONE](#)

Used as a parameter on the [getNextKey\(long\)](#) method, specifies to return a null value if the map is empty.

s
t
a
t
i
c
i
n
t

[TTL_FOREVER](#)

A constant indicating the time-to-live value is "forever".

s
t
a
t
i
c
i
n
t

[USE_DEFAULT](#)

A constant indicating the time-to-live value or lock timeout value is the default setting.

Method Summary

v

[clear\(\)](#)

o i d	Clear all keys from the Map.
v o i d	clearCopyMode() Resets the CopyMode back to the one in the BackingMap.
b o o l l e a n	containsKey(Object key) Returns true if this map contains a mapping for the specified key.
v o i d	flush() Pushes the current set of changes for the ObjectMap instance to the Loader without committing the changes.
Object	get(Object key) Retrieves the object from the cache at the given key.
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . d a t a g r i d . A g e n t M a n	getAgentManager() Returns the Agent Manager that allows DataGrid operations to be performed on the objects within this Map.

a
g
e
r

[List](#)

[getAll](#)([List](#) keyList)
Gets a list of entries from the map.

[List](#)

[getAllForUpdate](#)([List](#) keyList)
Same as the `getAll(List)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
p
r
o
j
e
c
t
o
r
.
m
d
.
E
n
t
i
t
y
M
e
t
a
d
a
t
a

[getEntityMetadata](#)()
Retrieve the metadata for the entity associated with this map.

[Object](#)

[getForUpdate](#)([Object](#) key)
Same as `get(Object)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry.

[Object](#)

[getIndex](#)([String](#) name)
Returns a reference to the named index that can be used with this Map.

O
b
j
e
c
t

[getIndex](#)(String name, boolean forUpdate)

Returns a reference to the named index that can be used with this Map.

M
a
p

[getJavaMap](#)()

Returns an implementation of java.util.Map that is backed by this ObjectMap.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

[getKeyOutputFormat](#)()

Retrieves the data format for all data access APIs that return cache keys for the current session.

i
n
t

[getMapType](#)()

Returns the type of the underlying BackingMap.

S
t
r
i
n
g

[getName](#)()

Returns the name of the ObjectMap as defined by the configuration.

O
b
j
e
c
t

[getNextKey](#)(long timeout)

Retrieves a key off the map in first-in-first-out (FIFO) insert order.

O
b

j
e
c
t
M
a
p
.
P
u
t
M
o
d
e

[getPutMode\(\)](#)

Retrieve the current PutMode set for this ObjectMap instance.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

[getValueOutputFormat\(\)](#)

Retrieves the data format for all data access APIs that return cache values for the current session.

v
o
i
d

[insert\(Object key, Object value\)](#)

Performs an explicit insert of a given entry.

v
o
i
d

[invalidate\(Object key, boolean isGlobal\)](#)

Invalidates an entry in the cache based on the key parameter.

v
o
i
d

[invalidateAll\(Collection keyList, boolean isGlobal\)](#)

Invalidate a set of cache entries based on the Collection of keys provided.

b
o

o l e a n	lock (Object key, com.ibm.websphere.objectgrid.LockMode lockMode) Obtains a lock for the given key.
L i s t < B o o l e a n >	lockAll (List keyList, com.ibm.websphere.objectgrid.LockMode lockMode) Obtains locks for the given keys.
O b j e c t	put (Object key, Object value) Puts the Object value into the cache at location represented by key.
v o i d	putAll (Map map) Puts each of the Object value into the cache at location represented by key contained in the Map.
O b j e c t	remove (Object key) Removes the Object value from the cache represented by key.
v o i d	removeAll (Collection keyList) Batch remove from the Map.
v o i d	setCopyMode (CopyMode copyMode, Class valueInterface) Allows the CopyMode for the Map to be overridden on this map on this session only.
v o i d	setKeyOutputFormat (com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data format for all data access APIs that return cache keys for the current session.
v o i d	setLockTimeout (int seconds) Overrides the BackingMap's lock timeout for this ObjectMap.
v o i d	setPutMode (ObjectMap.PutMode putMode) Allows the default operation for the put(Object, Object) and putAll(Map) methods to be changed, allowing the put to use the optimized upsert(Object, Object) and upsertAll(LinkedHashMap) implementations.
i n t	setTimeToLive (int ttl) Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL.
v o i d	setValueOutputFormat (com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data format for all data access APIs that return cache values for the current session.

v o i d	<p>touch(Object key)</p> <p>Updates the last access time in the BackingMap without retrieving the value to the ObjectMap.</p>
v o i d	<p>update(Object key, Object value)</p> <p>Performs an explicit update of a given entry.</p>
v o i d	<p>upsert(Object key, Object value)</p> <p>Puts the Object value into the cache at location represented by key.</p>
v o i d	<p>upsertAll(LinkedHashMap map)</p> <p>Puts each of the Object value into the cache at location represented by key contained in the Map.</p>

Field Detail

TTL_FOREVER

static final int **TTL_FOREVER**

A constant indicating the time-to-live value is "forever".

See Also:

[Constant Field Values](#)

USE_DEFAULT

static final int **USE_DEFAULT**

A constant indicating the time-to-live value or lock timeout value is the default setting.

The default setting is to retain the time-to-live value for any existing map entry and to use the default value from BackingMap setting if a new map entry is being created.

For lock timeout override the default setting is to use the value defined on the BackingMap

See Also:

[setLockTimeout\(int\)](#), [setTimeToLive\(int\)](#), [BackingMap.setTimeToLive\(int\)](#), [BackingMap.getTimeToLive\(\)](#), [BackingMap.setLockTimeout\(int\)](#), [Constant Field Values](#)

QUEUE_TIMEOUT_NONE

static final long **QUEUE_TIMEOUT_NONE**

Used as a parameter on the [getNextKey\(long\)](#) method, specifies to return a null value if the map is empty.

See Also:

[Constant Field Values](#)

QUEUE_TIMEOUT_INFINITE

static final long `QUEUE_TIMEOUT_INFINITE`

Used as a parameter on the [getNextKey\(long\)](#) method, specifies the method should block until a key becomes available.

See Also:

[Constant Field Values](#)

Method Detail

getName

[String](#) `getName()`

Returns the name of the ObjectMap as defined by the configuration.

Returns:

name of ObjectMap

get

[Object](#) `get(Object key)`
throws [ObjectGridException](#)

Retrieves the object from the cache at the given key.

Whether or not a copy of the object is returned is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode. If the key cannot be found in the map, a null value will be returned. A null value is also returned if a value is null and this map allows null values. To distinguish the two, use the `containsKey` method.

The return value is a SerializedValue when using the [CopyMode.COPY_TO_BYTES_RAW](#) CopyMode or `OutputFormat.RAW` OutputFormat with a ValueSerializerPlugin plug-in defined on the [BackingMap](#). The SerializedValue allows access to the value in its serialized form, or its native Java Object form.

The return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

Specification details:

Required permission:	MapPermission.READ
Pessimistic read lock acquired:	LockMode.SHARED
Pessimistic read lock held:	Yes
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found.

Parameters:

key - The entry to fetch

Returns:

the value, null, SerializedValue OR Tuple

Throws:

[IllegalArgumentException](#) - if key is null
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[containsKey\(Object\)](#), [getForUpdate\(Object\)](#), [CopyMode](#)

put

[Object](#) put([Object](#) key,
[Object](#) value)
throws [ObjectGridException](#)

Puts the Object value into the cache at location represented by key.

The value will be pushed down to the BackingMap/Loader at commit time and has two behaviors, which can be altered using the [setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSERTUPDATE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSERT](#) The value is put into the map using the specification of the [upsert\(Object, Object\)](#).

Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See [CopyMode](#) for a description of each possible copy mode.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an [EntityManager](#) API entity is associated with the [BackingMap](#).

Specification details:

Required [MapPermission.WRITE](#)
permission:

Transaction: Automatic or manual

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - The entry to put into the map
value - The value to put into the map using the key

Returns:

If [ObjectMap.PutMode.INSERTUPDATE](#) is set, return the previous value in this transaction.
If [ObjectMap.PutMode.UPSERT](#) is set, the return value is null.

Throws:

[IllegalArgumentException](#) - if key is null, or if the map does not allow null values and value is null
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[CopyMode](#)

getForUpdate

[Object](#) getForUpdate([Object](#) key)
throws [ObjectGridException](#)

Same as `get(Object)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry. See `LockStrategy.PESSIMISTIC` for additional information. Whether or not a copy of the object is returned is determined by the `CopyMode` setting for this map. See `CopyMode` for a description of each possible `CopyMode`. If the key cannot be found in the map, a `null` value will be returned. A `null` value is also returned if the value is `null` and this map allows `null` values. To distinguish the two, use the `containsKey` method.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

The return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

Specification details:

Required permission:	<code>MapPermission.READ</code>
Pessimistic read lock acquired:	<code>LockMode.UPGRADABLE</code> Note: Transaction isolation is ignored.
Pessimistic locks held:	Yes
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found and the appropriate lock is acquired.

Parameters:

key - The entry to fetch

Returns:

the value, `null`, `SerializedValue` OR `Tuple`

Throws:

[IllegalArgumentException](#) - if key is `null`
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[containsKey\(Object\)](#), [get\(Object\)](#), [CopyMode](#), [LockStrategy.PESSIMISTIC](#)

remove

`Object` `remove(Object key)`
throws [ObjectGridException](#)

Removes the Object value from the cache represented by key.

This removal will be pushed down to the `BackingMap/Loader` at commit time. If the key cannot be found in the map, a `null` value will be returned.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

The return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

Specification details:

Required `MapPermission.REMOVE`
permissi

on:

Transact Automatic or manual

ion:

Cache Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the
tier: operation to the Client Cache tier for client maps, or the Server Cache tier for
local or shard maps.

Parameters:

key - The entry to remove

Returns:

the current value at invocation time

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not
have the appropriate permission.

getAll

[List](#) `getAll(List keyList)`
throws [ObjectGridException](#)

Gets a list of entries from the map.

If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a SerializedValue when using the [CopyMode.COPY_TO_BYTES_RAW](#) CopyMode or OutputFormat.RAW OutputFormat with a ValueSerializerPlugin plug-in defined on the [BackingMap](#). The SerializedValue allows access to the value in its serialized form, or its native Java Object form.

A return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

Specification details:

Required permission:	MapPermission.READ
Pessimistic read lock acquired:	LockMode.SHARED
Pessimistic read lock held:	Yes
Transaction:	Automatic or manual
Cache tier:	For each key, progresses to all tiers until the key is found.

Parameters:

keyList - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not
have the appropriate permission.

See Also:

[get\(Object\)](#)

getAllForUpdate

[List](#) `getAllForUpdate(List keyList)`
throws [ObjectGridException](#)

Same as the `getAll(List)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries. See `LockStrategy.PESSIMISTIC` for additional information. If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

A return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

Specification details:

Required permission: `MapPermission.READ`
Pessimistic read lock acquired: `LockMode.UPGRADABLE`
Note: Transaction isolation is ignored.
Pessimistic locks held: Yes
Transaction: Automatic or manual
Cache tier: Progresses to all tiers until the keys are found and the appropriate locks are acquired.

Parameters:

`keyList` - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if `keyList` is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[getAll\(List\)](#), [getForUpdate\(Object\)](#), [LockStrategy.PESSIMISTIC](#)

removeAll

`void removeAll(Collection keyList)`
throws [ObjectGridException](#)

Batch remove from the Map. If a key in the list cannot be found, it will be ignored.

Specification details:

Required permission: `MapPermission.REMOVE`
Transaction: Automatic or manual
Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

`keyList` - A list of keys for identifying which entries to remove

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[remove\(Object\)](#)

putAll

```
void putAll(Map map)
    throws ObjectGridException
```

Puts each of the Object value into the cache at location represented by key contained in the Map.

The values will be pushed down to the BackingMap/Loader at commit time and has two behaviors, which can be altered using the [setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSE](#) The values are put into the map using the specification of the [upsertAll\(LinkedHashMap\)](#).

Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See CopyMode for a description of each possible copy mode.

An existing Map object will be passed in to use for obtaining the keys and values to be inserted or updated into the existing Map.

Specification details:

Required MapPermission.WRITE
permission:

Transact Automatic or manual
ion:

Cache Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the
tier: operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

map - The key/values to be put into the map.

Throws:

[IllegalArgumentException](#) - if map is null or contains a null key or if null values are not allowed and map contains a null value.

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[put\(Object, Object\)](#)

invalidate

```
void invalidate(Object key,
    boolean isGlobal)
    throws ObjectGridException
```

Invalidates an entry in the cache based on the key parameter.

If the key's value has changes pending in the ObjectMap, it is the application's responsibility to flush these changes to the Loader before invalidation. If a flush is not performed prior to invoking the invalidate operation, all pending changes for this key will be removed from the ObjectMap. If the key cannot be found in the map, it will be ignored.

The isGlobal parameter is used to indicate which cache level is used to invalidate the entries. If isGlobal is true, when the transaction is committed, the key is removed from the BackingMap also. If a subsequent get operation is performed, the BackingMap will be skipped and the Loader will be used to get the data. If isGlobal is false, the entry is only invalidated in the ObjectMap (transactional cache). If a subsequent get operation is performed, the BackingMap can be used; and, if it's not in the BackingMap, the Loader will be used to get the data.

A typical use of isGlobal being false is when a large number of records are touched in a transaction and the application wants to evict records that are no longer used in the cache.

Specification details:

Requires MapPermission.INVALIDATE

red

permission:

ssion:

Transaction Automatic or manual

action:

n:

Cache Applied to all tiers during commit except the Loader. Use

the tier: [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps. Set the isGlobal parameter to false to limit the operation to the transaction cache tier.

Parameters:

key - Object representing the key to be used for cache entry invalidation

isGlobal - Indicates whether to remove the entry from the BackingMap (true) or just the ObjectMap (false).

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

invalidateAll

```
void invalidateAll(Collection keyList,  
                  boolean isGlobal)  
    throws ObjectGridException
```

Invalidate a set of cache entries based on the Collection of keys provided. If a key in the collection cannot be found, it will be ignored.

Specification details:

Requires MapPermission.INVALIDATE

red

permission:

ssion:

Transaction Automatic or manual

action:

n:

Cache tier: Applied to all tiers during commit except the Loader. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps. Set the `isGlobal` parameter to `false` to limit the operation to the transaction cache tier.

Parameters:

`keyList` - A Collection of keys representing the entries to be invalidated
`isGlobal` - Indicates whether to remove the entry from the BackingMap (true) or just the ObjectMap (false).

Throws:

[IllegalArgumentException](#) - if `keyList` is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[invalidate\(Object, boolean\)](#)

setTimeToLive

```
int setTimeToLive(int ttl)
```

Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this ObjectMap, any previous value set by the BackingMap.`setTimeToLive(int)` method is overridden for this ObjectMap. If this method is never called on the ObjectMap, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from BackingMap setting if a new map entry is being created. If TTL is never set on the BackingMap, the cache entry can live "forever".

This method can only be used when the `TTLType` is set to `LAST_ACCESS_TIME` or `LAST_UPDATE_TIME` on the BackingMap. If this method is called on the ObjectMap and the `TTLType` is something other than `LAST_ACCESS_TIME` or `LAST_UPDATE_TIME`, an `IllegalStateException` is thrown.

Required permission: `MapPermission.INVALIDATE`

Parameters:

`ttl` - is the time-to-live value in seconds. The value must be ≥ -1 . A value of 0 is used to indicate the cache entry can live "forever" and -1 to indicate to use default setting. Use of the constant `TTL_FOREVER` is recommended when "forever" is desired and the constant `USE_DEFAULT` is recommended when "use default" setting is desired.

Returns:

previous time-to-live value in seconds. The constant `TTL_FOREVER` and constant `USE_DEFAULT` can be used to determine if the previous TTL is one of the special values.

Throws:

[IllegalArgumentException](#) - if seconds argument is < -1 .
[IllegalStateException](#) - if `BackingMap.getTtlEvictorType()` returns anything other than `TTLType.LAST_ACCESS_TIME` or `TTLType.LAST_UPDATE_TIME`.
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[TTL_FOREVER](#), [USE_DEFAULT](#), [BackingMap.setTimeToLive\(int\)](#), [TTLType.LAST_ACCESS_TIME](#), [TTLType.LAST_UPDATE_TIME](#)

update


```
void update(Object key,  
           Object value)  
    throws KeyNotFoundException,  
           ObjectGridException
```

Performs an explicit update of a given entry.

A get operation is not required prior to invoking the update method (unlike the put method). Also, an update invocation will never insert a new record. If a the map's LockStrategy is LockStrategy.OPTIMISTIC this method will implicitly get the entry so as to have the version value of the object for when this method was invoked. Whether or not a copy of the object is made when transaction is committed is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode.

If a key cannot be found in the map during commit, a TransactionException will be thrown.

Specification details:

Required MapPermission.WRITE

permissi
on:

Transact Automatic or manual
ion:

Cache Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the
tier: operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - Identifies the entry to be updated
value - The updated value for this entry

Throws:

[IllegalArgumentException](#) - if key is null or if the map does not allow null values and value is null.

[KeyNotFoundException](#) - if the key cannot be found in the map

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[insert\(Object, Object\)](#), [put\(Object, Object\)](#), [CopyMode](#), [LockStrategy.OPTIMISTIC](#)

insert

```
void insert(Object key,  
           Object value)  
    throws DuplicateKeyException,  
           ObjectGridException
```

Performs an explicit insert of a given entry.

The key must not exist before executing this method. Also, an insert invocation will never update an existing record. Whether or not a copy of the object is made when a transaction is committed is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode.

If the key is already in the map, a TransactionException will be thrown during commit.

Specification details:

Required MapPermission.INSERT

permission:

Transaction: Automatic or manual

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - Identifies the entry to be inserted
value - The value for this entry

Throws:

[IllegalArgumentException](#) - if key is null or if the map does not allow null values and value is null.
[DuplicateKeyException](#) - if this entries already exists in the map
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[put\(Object, Object\)](#), [update\(Object, Object\)](#), [CopyMode](#)

getIndex

[Object](#) [getIndex](#)([String](#) name)
throws [com.ibm.websphere.objectgrid.IndexUndefinedException](#),
[com.ibm.websphere.objectgrid.IndexNotReadyException](#)

Returns a reference to the named index that can be used with this Map. This index cannot be shared between threads and works on the same rules as Session. The returned value should be cast to the right index interface such as [MapIndex](#), [MapRangeIndex](#) or a custom index interface such as a geo spatial index.

Parameters:

name - The index name

Returns:

A reference to the index, it must be cast to the appropriate index interface.

Throws:

[IndexUndefinedException](#) - if the index is not defined on the [BackingMap](#)
[IndexNotReadyException](#) - if the index is a dynamic index and it is not ready

Since:

WAS XD 6.0.1

getIndex

[Object](#) [getIndex](#)([String](#) name,
boolean forUpdate)
throws [com.ibm.websphere.objectgrid.IndexUndefinedException](#),
[com.ibm.websphere.objectgrid.IndexNotReadyException](#)

Returns a reference to the named index that can be used with this Map. This index cannot be shared between threads and works on the same rules as Session. The returned value should be cast to the right index interface such as [MapIndex](#), [MapRangeIndex](#) or a custom index interface such as a geo spatial index.

Parameters:

name - The index name
forUpdate - if true, the returned index will always operate with forUpdate intent.

Returns:

A reference to the index, it must be cast to the appropriate index interface.

Throws:

`IndexUndefinedException` - if the index is not defined on the `BackingMap`
`IndexNotReadyException` - if the index is a dynamic index and it is not ready

Since:

WAS XD 6.1.0.1

flush

void **flush**()
throws [ObjectGridException](#)

Pushes the current set of changes for the `ObjectMap` instance to the `Loader` without committing the changes. The changes are not propagated to the `BackingMap` either. This is useful for re-priming the `Loader`'s data without committing the current transaction and starting over.

Throws:

[ObjectGridException](#) - if an error occurs during processing

See Also:

[Session.flush\(\)](#)

containsKey

boolean **containsKey**([Object](#) key)
throws [ObjectGridException](#)

Returns `true` if this map contains a mapping for the specified key. `ObjectGrid` does not support null keys. If you configured the map to support `null` values, this method can be used to determine whether a key is contained in the map or not.

This API does not hold any locks when using pessimistic locking.

Specification details:

Required permission:	<code>MapPermission.READ</code>
Pessimistic read lock acquired:	<code>LockMode.SHARED</code>
Pessimistic locks held:	None
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found.

Parameters:

key - key whose presence in this map is to be tested.

Returns:

`true` if this map contains a mapping for the specified key.

Throws:

[IllegalArgumentException](#) - if `null` key parameter is passed in
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

getJavaMap

[Map](#) **getJavaMap**()

Returns an implementation of `java.util.Map` that is backed by this `ObjectMap`.

The returned `java.util.Map` implementation can be cast to `com.ibm.websphere.objectgrid.JavaMap` to be able to use the rest of the `ObjectGrid`

programming model, but with `java.util.Map`'s use of `RuntimeExceptions` instead of checked `ObjectGridExceptions`.

Returns:

a `java.util.Map` backed by this `ObjectMap`

See Also:

[Map](#), [JavaMap](#)

touch

```
void touch(Object key)
    throws ObjectGridException
```

Updates the last access time in the `BackingMap` without retrieving the value to the `ObjectMap`.

The last access time is updated during commit. If the key does not exist in the `BackingMap`, a `TransactionException` will be returned during commit processing.

Specification details:

Required None

permission:

Transaction: Automatic or manual

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - key to be touched

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

setCopyMode

```
void setCopyMode(CopyMode copyMode,
    Class valueInterface)
    throws TransactionAlreadyActiveException,
    ObjectGridException
```

Allows the `CopyMode` for the Map to be overridden on this map on this session only.

This method allows an application to use an optimal `CopyMode` TRANSACTION by TRANSACTION as its needs dictate. The `CopyMode` cannot be changed during a transaction. There must be no active transaction when this method is called.

Parameters:

copyMode - must be one of the final static variables defined in `CopyMode`. See `CopyMode` class for an explanation of each mode and how the valueInterface is used for `CopyMode.COPY_ON_WRITE`.

valueInterface - the value interface `Class` object. Specify null in version 7.1 and later.

Throws:

[IllegalArgumentException](#) - if copyMode is null or COPY_ON_WRITE CopyMode is specified and the required value interface parameter is null

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

[ObjectGridException](#) - if an error occurs during processing

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\), CopyMode](#)

clearCopyMode

void **clearCopyMode**()
throws [TransactionAlreadyActiveException](#)

Resets the CopyMode back to the one in the BackingMap.

This method is used to reverse a previous setCopyMode method call for this ObjectMap. This method can only be called when no transaction is active on the associated session.

Throws:

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

See Also:

[setCopyMode\(CopyMode, Class\)](#)

getNextKey

[Object](#) **getNextKey**(long timeout)
throws [ObjectGridException](#)

Retrieves a key off the map in first-in-first-out (FIFO) insert order.

The entry is locked by the session such that other calls to getNextKey will not return the same key. The key can be used to remove or manipulate the value although leaving the entry will result in the key remaining at the beginning of the queue. This order is optimized for performance and is not guaranteed especially across partitions or in highly concurrent environments.

The return value is a SerializedKey when OutputFormat.RAW is set for the keys. The default key output format for maps that are associated with a KeySerializerPlugin is OutputFormat.RAW. The SerializedKey allows access to the value in its serialized form, or its native Java Object form.

The return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

Specification details:

Required permission:	MapPermission.READ
Pessimistic read lock acquired:	LockMode.EXCLUSIVE
Pessimistic read lock held:	Yes
Transaction:	Automatic or manual

Parameters:

timeout - The period of time in milliseconds to wait for an entry to become available on the queue.

Returns:

the next key

Throws:

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

WAS XD 6.1

See Also:

getEntityMetadata

com.ibm.websphere.projector.md.EntityMetadata **getEntityMetadata()**

Retrieve the metadata for the entity associated with this map.

Returns:

the EntityMetadata if an entity is associated with this map or null if there is no entity associated with this map.

Since:

WAS XD 6.1

getMapType

int **getMapType()**

Returns the type of the underlying BackingMap.

The return value is equivalent to one of the constants declared on the BackingMap interface, [BackingMap.LOCAL](#), [BackingMap.SERVER](#), or [BackingMap.CLIENT](#).

Returns:

the BackingMap type

Since:

WAS XD 6.1

getAgentManager

com.ibm.websphere.objectgrid.datagrid.AgentManager **getAgentManager()**

Returns the Agent Manager that allows DataGrid operations to be performed on the objects within this Map.

This method should only be called on a client ObjectGrid. If called on a non client ObjectGrid an `IllegalStateException` will be thrown

Returns:

AgentManager

Throws:

[IllegalStateException](#) - if this method is invoked on a non client ObjectGrid

Since:

WAS XD 6.1

setLockTimeout

void **setLockTimeout**(int seconds)

Overrides the BackingMap's lock timeout for this ObjectMap.

Establishes the number of seconds that any given fetch (get, getForUpdate, find, findForUpdate) of a cache entry will wait to get a lock. When the lock strategy is `LockStrategy.NONE`, no lock manager is used by this map. In this case, a call to this method does nothing.

Parameters:

seconds - is the lock timeout in seconds. The value must be ≥ -1 . A value of -1 is

used to indicate to use the default setting. Use of the constant `USE_DEFAULT` is recommended when "use default" setting is desired. A value of 0 indicates that if a lock cannot be retrieved immediately to time out without waiting for any period of time for the lock to be released and made available.

Throws:

[IllegalArgumentOutOfRangeException](#) - if seconds argument is less than -1 (`USE_DEFAULT`)

Since:

WAS XD 6.1

See Also:

[USE_DEFAULT](#), [BackingMap.setLockTimeout\(int\)](#), [BackingMap.setLockStrategy\(LockStrategy\)](#), [LockStrategy.OPTIMISTIC](#), [LockStrategy.PESSIMISTIC](#)

clear

```
void clear()  
    throws ObjectGridException
```

Clear all keys from the Map.

This method is an automatic transaction call. The [Session.isTransactionActive\(\)](#) must answer false prior to invoking this method.

Specification details:

Required permission:	<code>MapPermission.REMOVE</code>
Pessimistic read lock acquired:	Map exclusive lock is acquired.
Transaction:	Hybrid. Each shard is cleared in a separate transaction.
Cache tier:	Applies to all cache tiers except the Loader.

Throws:

[ObjectGridException](#) - if an error occurs during processing
[TransactionAlreadyActiveException](#) - if a transaction is already started.
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

WAS XD 6.1.0.3

lock

```
boolean lock(Object key,  
            com.ibm.websphere.objectgrid.LockMode lockMode)  
    throws ObjectGridException
```

Obtains a lock for the given key. **Specification details:**

Required permission:	<code>MapPermission.READ</code>
Pessimistic read lock acquired:	<code>LockMode.SHARED</code> , <code>LockMode.UPGRADABLE</code> OR <code>LockMode.EXCLUSIVE</code> Note: Transaction isolation is ignored.
Pessimistic locks held:	Yes
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found and the appropriate lock is acquired.

Parameters:

key - the key to lock
lockMode - the lockMode to obtain for the given key

Returns:

true if the entry exists in the grid or Loader (if one is defined)

Throws:

[IllegalArgumentException](#) - if key is null
[IllegalStateException](#) - if this map is not using [LockStrategy.PESSIMISTIC](#) LockStrategy
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:
8.6, XC10 2.5

lockAll

[List<Boolean>](#) **lockAll**([List](#) keyList,
com.ibm.websphere.objectgrid.LockMode lockMode)
throws [ObjectGridException](#)

Obtains locks for the given keys. **Specification details:**

Required permission: MapPermission.READ
Pessimistic read lock acquired: LockMode.SHARED, LockMode.UPGRADABLE OR LockMode.EXCLUSIVE
Note: Transaction isolation is ignored.
Pessimistic locks held: Yes
Transaction: Automatic or manual
Cache tier: Progresses to all tiers until the keys are found and the appropriate locks are acquired.

Parameters:

keyList - the keys to lock
lockMode - the lockMode to obtain for the given keys

Returns:

a List of Booleans indicating whether the entries exists in the grid or Loader (if one is defined)

Throws:

[IllegalStateException](#) - if this map is not using [LockStrategy.PESSIMISTIC](#) LockStrategy
[IllegalArgumentException](#) - if keyList is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:
8.6, XC10 2.5

upsert

void **upsert**([Object](#) key,
[Object](#) value)
throws [ObjectGridException](#)

Puts the Object value into the cache at location represented by key.

The value will be pushed down to the BackingMap/Loader at commit time. The semantics of this method are that the Loader will receive a [LogElement.UPSERT](#) operation and the map will either do an insert or an update to cause the map to contain this updated value. Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See CopyMode for a description of each possible copy mode.

Specification details:

Required MapPermission.WRITE
permissi
on:
Transact Automatic or manual

ion:

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - The entry to insert or update in the map

value - The value to insert or update in the map using the key

Throws:

[IllegalArgumentException](#) - if key is null, or if the map does not allow null values and value is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

8.6, XC10 2.5

See Also:

[CopyMode](#)

upsertAll

```
void upsertAll(LinkedHashMap map)
    throws ObjectGridException
```

Puts each of the Object value into the cache at location represented by key contained in the Map. The values will be pushed down to the BackingMap/Loader at commit time. The semantics of this method are that the Loader will receive a [LogElement.UPSERT](#) operation and the map will either do an insert or an update to cause the map to contain this updated value. Whether or not a copy of the objects is made when transaction is committed is determined by the copy mode setting for this map. See [CopyMode](#) for a description of each possible copy mode.

Specification details:

Required `MapPermission.WRITE`

permissi

on:

Transact Automatic or manual

ion:

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

map - The key/values to be inserted or updated in the map. The type is `LinkedHashMap` so that the order can be controlled to avoid deadlocks.

Throws:

[IllegalArgumentException](#) - if map is null or contains a null key or if null values are not allowed and map contains a null value.

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

8.6, XC10 2.5

See Also:

[CopyMode](#)

getKeyOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getKeyOutputFormat()**

Retrieves the data format for all data access APIs that return cache keys for the current session.

Returns:

the data output format or null if the default should be used.

Since:

8.6, XC10 2.5

setKeyOutputFormat

void **setKeyOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)
throws [TransactionAlreadyActiveException](#)

Sets the data format for all data access APIs that return cache keys for the current session.

This method supports map configurations with a KeyDataSerializer plug-in defined, or with eXtreme Data Format enabled.

Parameters:

outputFormat - the data output format to use or OutputFormat.UNDEFINED to use the default defined on the parent [BackingMap](#).

Throws:

[IllegalArgumentException](#) - thrown when the data format is not valid for the current configuration.

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

Since:

8.6, XC10 2.5

See Also:

[BackingMap.getKeyOutputFormat\(\)](#)

getValueOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getValueOutputFormat()**

Retrieves the data format for all data access APIs that return cache values for the current session.

Returns:

the data output format or null if the default should be used.

Since:

8.6, XC10 2.5

setValueOutputFormat

void **setValueOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)
throws [TransactionAlreadyActiveException](#)

Sets the data format for all data access APIs that return cache values for the current session.

This method is functionally equivalent to the [setCopyMode\(CopyMode, Class\)](#) with the [CopyMode.COPY_TO_BYTES](#) and [CopyMode.COPY_TO_BYTES_RAW](#) values when used with a ValueDataSerializer or eXtreme Data Format.

This method supports map configurations with a ValueDataSerializer plug-in defined, or with eXtreme Data Format enabled.

Parameters:

outputFormat - the data output format to use or `OutputFormat.UNDEFINED` to use the default defined on the parent [BackingMap](#).

Throws:

[IllegalArgumentException](#) - thrown when the data format is not valid for the current configuration.

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

Since:

8.6, XC10 2.5

setPutMode

```
void setPutMode(ObjectMap.PutMode putMode)
    throws TransactionAlreadyActiveException,
           ObjectGridException
```

Allows the default operation for the [put\(Object, Object\)](#) and [putAll\(Map\)](#) methods to be changed, allowing the put to use the optimized [upsert\(Object, Object\)](#) and [upsertAll\(LinkedHashMap\)](#) implementations.

The PutMode cannot be changed during a transaction. There must be no active transaction when this method is called.

Parameters:

putMode - the mode in which the put methods operate.

Throws:

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

[ObjectGridException](#) - if an error occurs during processing

Since:

8.6, XC10 2.5

getPutMode

```
ObjectMap.PutMode getPutMode()
```

Retrieve the current PutMode set for this ObjectMap instance.

Returns:

the current PutMode.

Since:

8.6, XC10 2.5

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Enum ObjectMap.PutMode

[java.lang.Object](#)

 [java.lang.Enum](#)<[ObjectMap.PutMode](#)>

 [com.ibm.websphere.objectgrid.ObjectMap.PutMode](#)

All Implemented Interfaces:

[Serializable](#), [Comparable](#)<[ObjectMap.PutMode](#)>

Enclosing interface:

[ObjectMap](#)

```
public static enum ObjectMap.PutMode
extends Enum<ObjectMap.PutMode>
```

Identifies the operation mode of the [ObjectMap.put\(Object, Object\)](#), [ObjectMap.putAll\(Map\)](#), [JavaMap.put\(Object, Object\)](#) and [JavaMap.putAll\(Map\)](#) methods.

Since:

8.6, XC10 2.5

See Also:

[ObjectMap.setPutMode\(PutMode\)](#)

Enum Constant Summary

[INSERTUPDATE](#)

Deprecated. *Deprecated in 8.6. Use the [UPSERT](#) enumeration.*

[UPSERT](#)

The put API behaves like the upsert method.

Method Summary

s
t
a
t
i
c

O
b
j
e
c
t
M
a
p
.
P
u
t
M
o
d
e

[valueOf](#)([String](#) name)

Returns the enum constant of this type with the specified name.

d
e
s
t
a
t
i
c
O
b
j
e
c
t
M
e
t
h
o
d
s
[
]

[values\(\)](#)

Returns an array containing the constants of this enum type, in the order they are declared.

Methods inherited from class [java.lang.Enum](#)

[clone](#), [compareTo](#), [equals](#), [finalize](#), [getDeclaringClass](#), [hashCode](#), [name](#), [ordinal](#), [toString](#), [valueOf](#)

Methods inherited from class [java.lang.Object](#)

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Enum Constant Detail

INSERTUPDATE

```
public static final ObjectMap.PutMode INSERTUPDATE
```

Deprecated. *Deprecated in 8.6. Use the [UPSERT](#) enumeration.*
The put API behaves like an insert or update.

UPSERT

```
public static final ObjectMap.PutMode UPSERT
```

The put API behaves like the upsert method.

Method Detail

values

```
public static ObjectMap.PutMode[] values()
```

Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows:

```
for (ObjectMap.PutMode c : ObjectMap.PutMode.values())  
    System.out.println(c);
```

Returns:

an array containing the constants of this enum type, in the order they are declared

valueOf

```
public static ObjectMap.PutMode valueOf(String name)
```

Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

Parameters:

name - the name of the enum constant to be returned.

Returns:

the enum constant with the specified name

Throws:

[IllegalArgumentException](#) - if this enum type has no constant with the specified name

[NullPointerException](#) - if the argument is null

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
SUMMARY: NESTED ENUM		DETAIL: ENUM					
CONSTANTS FIELD METHOD		CONSTANTS FIELD METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class ObjectGridRuntimeException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[TransactionTimeoutException](#)

```
public class ObjectGridRuntimeException
extends RuntimeException
implements IObjectGridException
```

This exception is the base class for all runtime exceptions thrown by the cache.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

ObjectGridRuntimeException()	Constructs a new ObjectGridRuntimeException with null as its detail message.
ObjectGridRuntimeException(String message)	Constructs a new ObjectGridRuntimeException with the specified detail message.
ObjectGridRuntimeException(String message, Throwable cause)	Constructs a new ObjectGridRuntimeException with the specified detail message and cause.
ObjectGridRuntimeException(Throwable cause)	Constructs a new ObjectGridRuntimeException with a specified cause.

Method Summary

I h r o w a b l e	<p>getCause()</p> <p>Returns the cause of this ObjectGridRuntimeException or null if the cause is nonexistent or unknown.</p>
---	---

[initCause](#)([Throwable](#) cause)

Initializes the *cause* of this `ObjectGridRuntimeException` to the specified value.

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridRuntimeException

```
public ObjectGridRuntimeException()
```

Constructs a new `ObjectGridRuntimeException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[initCause\(Throwable\)](#)

ObjectGridRuntimeException

```
public ObjectGridRuntimeException(String message)
```

Constructs a new `ObjectGridRuntimeException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridRuntimeException

```
public ObjectGridRuntimeException(Throwable cause)
```

Constructs a new `ObjectGridRuntimeException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ObjectGridRuntimeException`s that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[getCause\(\)](#)

ObjectGridRuntimeException

```
public ObjectGridRuntimeException(String message,  
                                Throwable cause)
```

Constructs a new ObjectGridRuntimeException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ObjectGridRuntimeException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A `null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[getCause\(\)](#), [Throwable.getMessage\(\)](#)

Method Detail

getCause

```
public Throwable getCause()
```

Returns the cause of this ObjectGridRuntimeException or `null` if the cause is nonexistent or unknown. (The cause is the throwable that caused this ObjectGridRuntimeException to get thrown.)

This implementation returns the cause that was supplied via one of the constructors requiring a `Throwable`, or that was set after creation with the `initCause(Throwable)` method. While it is typically unnecessary to override this method, a subclass can override it to return a cause set by some other means. This is appropriate for a "legacy chained throwable" that predates the addition of chained exceptions to `Throwable`. Note that it is *not* necessary to override any of the `PrintStackTrace` methods, all of which invoke the `getCause` method to determine the cause of an ObjectGridRuntimeException

Specified by:

[getCause](#) in interface [IObjectGridException](#)

Overrides:

[getCause](#) in class [Throwable](#)

Returns:

the cause of this ObjectGridRuntimeException or `null` if the cause is nonexistent or unknown.

See Also:

[ObjectGridRuntimeException\(String, Throwable\)](#), [ObjectGridRuntimeException\(Throwable\)](#), [initCause\(Throwable\)](#)

initCause

```
public Throwable initCause(Throwable cause)
```

Initializes the *cause* of this ObjectGridRuntimeException to the specified value. (The cause is the throwable that caused this ObjectGridRuntimeException to get thrown.)

This method can be called at most once. It is generally called from within the constructor, or immediately after creating the ObjectGridRuntimeException. If this

ObjectGridRuntimeException was created with ObjectGridRuntimeException(Throwable) or ObjectGridRuntimeException(String,Throwable), this method cannot be called even once.

Specified by:

[initCause](#) in interface [IObjectGridException](#)

Overrides:

[initCause](#) in class [Throwable](#)

Parameters:

cause - the cause (which is saved for later retrieval by the `getCause()` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown.)

Returns:

a reference to this ObjectGridRuntimeException instance.

Throws:

[IllegalArgumentException](#) - if cause is this ObjectGridRuntimeException. (An ObjectGridRuntimeException cannot be its own cause.)

[IllegalStateException](#) - if this ObjectGridRuntimeException was created with ObjectGridRuntimeException(Throwable) or ObjectGridRuntimeException(String,Throwable), or this method has already been called on this ObjectGridRuntimeException.

See Also:

[ObjectGridRuntimeException\(String, Throwable\)](#), [ObjectGridRuntimeException\(Throwable\)](#), [getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class ObjectGridManagerFactory

[java.lang.Object](#)



```
public final class ObjectGridManagerFactory
extends Object
```

This factory class is a high level helper class to get ObjectGridManager instances.

Since:

WAS XD 6.0, XC10

Constructor Summary

[ObjectGridManagerFactory\(\)](#)

Method Summary

s
t
a
t
i
c
O
b
j
e
c
t
G
r
i
d
M
a
n
a
g
e
r

[getObjectGridManager\(\)](#)

Returns the ObjectGridManager singleton.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridManagerFactory

```
public ObjectGridManagerFactory()
```

Method Detail

getObjectGridManager

```
public static final ObjectGridManager getObjectGridManager()
```

Returns the ObjectGridManager singleton.

Returns:

The ObjectGridManager singleton

See Also:

[ObjectGridManager](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Interface ObjectGridManager

```
public interface ObjectGridManager
```

ObjectGridManager is responsible for creating or retrieving local ObjectGrid instances and connecting to distributed ObjectGrid servers. Use the [ObjectGridManagerFactory](#) to retrieve an ObjectGridManager.

Use the createObjectGrid methods to create a local, in-memory ObjectGrid instance. The createObjectGrid methods give you the choice of caching the created ObjectGrid instance. If you choose to cache the instance, you cannot create an ObjectGrid with the same name unless you remove the previously created ObjectGrid using the removeObjectGrid(String) method. A cached ObjectGrid instance can later be retrieved using the getObjectGrid(String) method.

An example of creating a local in-memory ObjectGrid programmatically:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ObjectGrid grid = ogMgr.createObjectGrid("LocalBookStoreGrid");
grid.defineMap("Orders");
grid.defineMap("Books");
grid.initialize();
...
grid.destroy();
```

An example of creating a local in-memory ObjectGrid using an ObjectGrid descriptor XML file:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
URL objectgridXML = Thread.currentThread().getContextClassLoader().getResource("configs/objectgrid.xml");
ObjectGrid grid = ogMgr.createObjectGrid("LocalBookStoreGrid", objectgridXML);
grid.initialize();
...
ogMgr.destroy();
```

Use the connect methods to connect to a distributed ObjectGrid. The connect methods return a ClientClusterContext that can then be passed to one of the getObjectGrid methods, which will in turn retrieve a client ObjectGrid instance.

An example to connect to a dynamic, distributed ObjectGrid using a catalog server cluster:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ClientClusterContext ccc = ogMgr.connect("catserver1:2809,catserver2:2809", null, null);
ObjectGrid grid = ogMgr.getObjectGrid(ccc, "BookStoreGrid");
...
ogMgr.disconnect(ccc);
```

An example to connect to an embedded ObjectGrid server (a server running in the current process):

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ClientClusterContext ccc = ogMgr.connect((ClientSecurityConfiguration) null, (URL) null);
ObjectGrid grid = ogMgr.getObjectGrid(ccc, "BookStoreGrid");
...
```

```
ogMgr.disconnect(ccc);
```

This interface also allows ObjectGrid trace to be disabled completely for performance improvements especially on a processor with a smaller L2 cache.

Since:

WAS XD 6.0, XC10

Method Summary	
ClientSecurityConfiguration	<p>connect(ClientSecurityConfiguration securityProps, URL overRideObjectGridXml) Use this method to connect to an embedded ObjectGrid server.</p>
ClientSecurityConfiguration	<p>connect(String catalogServerEndpoints, ClientSecurityConfiguration securityProps, URL overRideObjectGridXml) Connects a client process to a remote ObjectGrid catalog server or catalog server cluster (a dynamic ObjectGrid deployment topology).</p>
ObjectGrid	<p>createObjectGrid() Creates a local, in-memory instance of an ObjectGrid.</p>
ObjectGrid	<p>createObjectGrid(String objectGridName) Creates and caches a local, in-memory ObjectGrid instance with the specified name.</p>

j d	
O b j e c t G r i d	<p>createObjectGrid(String objectGridName, boolean cacheInstance) Creates a local, in-memory instance of an ObjectGrid with the specified name.</p>
O b j e c t G r i d	<p>createObjectGrid(String objectGridName, URL xmlFile) Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.</p>
O b j e c t G r i d	<p>createObjectGrid(String objectGridName, URL xmlFile, boolean cacheInstance) Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.</p>
O b j e c t G r i d	<p>createObjectGrid(String objectGridName, URL xmlFile, boolean enableXmlValidation, boolean cacheInstance) Deprecated. <i>Deprecated in version 8.6. XML validation is always enabled. Use instead.</i></p>
L i s t	<p>createObjectGrids(URL xmlFile) Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.</p>
L i s t	<p>createObjectGrids(URL xmlFile, boolean cacheInstances) Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.</p>
L i s t	<p>createObjectGrids(URL xmlFile, boolean enableXmlValidation, boolean cacheInstances) Deprecated. <i>Deprecated in version 8.6. XML validation is always enabled. Use createObjectGrids(URL, boolean) instead.</i></p>
b o l e a n	<p>disconnect(ClientClusterContext context) Disconnects a client process from an ObjectGrid server</p>
C a	

getCatalogDomainManager()

Retrieve the CatalogDomainManager for this ObjectGridManager.

getObjectGrid(ClientClusterContext context, String objectGridName)

Returns a client ObjectGrid instance corresponding to an ObjectGrid in an ObjectGrid cluster.

getObjectGrid(ClientClusterContext context, String objectGridName, ObjectGridConfiguration overrideOGConfig)

Returns a client ObjectGrid instance corresponding to an ObjectGrid in an ObjectGrid cluster.

getObjectGrid(String objectGridName)

Returns a cached local, in-memory ObjectGrid instance by name.

getObjectGrids()

Gets a List of the local, in-memory ObjectGrid instances that have been previously cached.

getOverrideObjectGridConfigurations()

Deprecated. This method is replaced in 7.1.1 with the getObjectGrid method that takes a ObjectGridConfigurations.

getTraceSpecification()

Retrieves the trace specification for the current JVM.

putOverrideObjectGridConfigurations(String clusterName, List objectGridConfigList)

Deprecated. This method is replaced in 7.1.1 with the getObjectGrid method that takes a ObjectGridConfigurations.

v o i d	removeObjectGrid (String objectGridName) Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances.
v o i d	removeObjectGrid (String objectGridName, boolean destroy) Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances and optionally destroys its associated resources.
v o i d	setOverrideObjectGridConfigurations (Map overrideMap) Deprecated. <i>This method is replaced in 7.1.1 with the getObjectGrid method that takes a ObjectGridConfigurations.</i>
v o i d	setTraceEnabled (boolean enabledFlag) Enables or disables ObjectGrid trace for the JVM.
v o i d	setTraceFileName (String traceFileName) Sets the trace output to go to a file instead of System.out.
v o i d	setTraceSpecification (String traceSpec) Set the trace specification for the current JVM.

Method Detail

createObjectGrid

[ObjectGrid](#) createObjectGrid()
throws [ObjectGridException](#)

Creates a local, in-memory instance of an ObjectGrid.

The created ObjectGrid returned by this method is assigned a unique name is **not** cached by the ObjectGridManager. Use the ObjectGrid.setName(String) method to change the ObjectGrid name.

Returns:

an instance of ObjectGrid with a unique name assigned

Throws:

[ObjectGridException](#) - if any error occurs during the ObjectGrid creation

See Also:

[ObjectGrid](#), [ObjectGrid.setName\(String\)](#)

createObjectGrid

[ObjectGrid](#) createObjectGrid([String](#) objectGridName,
boolean cacheInstance)
throws [ObjectGridException](#)

Creates a local, in-memory instance of an ObjectGrid with the specified name.

The instance of ObjectGrid created can optionally be cached. If an ObjectGrid with the specified name was previously created and cached, an ObjectGridException will be thrown.

Parameters:

objectGridName - the name of the ObjectGrid to be created. Must not be null.
cacheInstance - true, if the ObjectGrid instance should be cached

Returns:

an ObjectGrid instance with the specified name.

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with this name has already been cached or any error occurs during the ObjectGrid creation.

See Also:

[ObjectGrid](#)

createObjectGrid

[ObjectGrid](#) createObjectGrid([String](#) objectGridName)
throws [ObjectGridException](#)

Creates and caches a local, in-memory ObjectGrid instance with the specified name.

The ObjectGrid instance created by this method will be cached. Invoking this method is equivalent to invoke createObjectGrid(String, true)

Parameters:

objectGridName - the Name of the ObjectGrid instance to be created. Must not be null.

Returns:

an ObjectGrid instance with the specified name

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with this name has already been cached, or any error occurs during the ObjectGrid creation

See Also:

[createObjectGrid\(String, boolean\)](#), [ObjectGrid](#)

createObjectGrid

[@Deprecated](#)

[ObjectGrid](#) createObjectGrid([String](#) objectGridName,
[URL](#) xmlFile,
boolean enableXmlValidation,
boolean cacheInstance)
throws [ObjectGridException](#)

Deprecated. *Deprecated in version 8.6. XML validation is always enabled. Use [instead](#).*

Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

An ObjectGrid instance is created for the ObjectGrid configuration in the XML file corresponding to the specified ObjectGrid name. If the specified ObjectGrid name cannot be found in the XML file, an exception will be thrown.

This returned ObjectGrid instance optionally can be cached.

If the URL is null, it will be simply ignored. In this case, this method behaves the same as the createObjectGrid(String, boolean).

Parameters:

objectGridName - the Name of the ObjectGrid instance to be returned. Must not be null.

xmlFile - a URL to a well formed xml file based on the ObjectGrid schema.

enableXmlValidation - if true the XML is validated

cacheInstance - a boolean value indicating whether the returned ObjectGrid instance

defined in the XML will be cached or not. If true, the instance will be cached.

Returns:

an ObjectGrid instance

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with the same name has been previously cached, an ObjectGrid configuration with the specified name can be found in the xml file, or any other error occurs during ObjectGrid creation.

See Also:

[createObjectGrid\(String, boolean\)](#), [ObjectGrid](#)

createObjectGrid

[ObjectGrid](#) createObjectGrid([String](#) objectGridName,
[URL](#) xmlFile,
boolean cacheInstance)
throws [ObjectGridException](#)

Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

An ObjectGrid instance is created for the ObjectGrid configuration in the XML file corresponding to the specified ObjectGrid name. If the specified ObjectGrid name cannot be found in the XML file, an exception will be thrown.

This returned ObjectGrid instance optionally can be cached.

If the URL is null, it will be simply ignored. In this case, this method behaves the same as the createObjectGrid(String, boolean).

Parameters:

objectGridName - the Name of the ObjectGrid instance to be returned. Must not be null.

xmlFile - a URL to a well formed xml file based on the ObjectGrid schema.

cacheInstance - a boolean value indicating whether the returned ObjectGrid instance defined in the XML will be cached or not. If true, the instance will be cached.

Returns:

an ObjectGrid instance

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with the same name has been previously cached, an ObjectGrid configuration with the specified name can be found in the xml file, or any other error occurs during ObjectGrid creation.

Since:

8.6, XC10 2.5

See Also:

[createObjectGrid\(String, boolean\)](#), [ObjectGrid](#)

createObjectGrids

[List](#) createObjectGrids([URL](#) xmlFile,
boolean enableXmlValidation,
boolean cacheInstances)
throws [ObjectGridException](#)

Deprecated. *Deprecated in version 8.6. XML validation is always enabled. Use [createObjectGrids\(URL, boolean\)](#) instead.*

Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

The returned ObjectGrid instances can be cached. An ObjectGridException will be thrown when attempting to cache a newly created ObjectGrid that has the same name as an ObjectGrid that has already been cached.

Parameters:

xmlFile - the file that defines an ObjectGrid or multiple ObjectGrids

cacheInstances - set to true to cache all ObjectGrid instances created based on the file

Returns:

a list of ObjectGrid instances

Throws:

[ObjectGridException](#) - if attempting to create and cache an ObjectGrid with the same name as an ObjectGrid that has already been cached, or any other error occurs during ObjectGrid creation

See Also:

[ObjectGrid](#)

createObjectGrids

[List](#) createObjectGrids([URL](#) xmlFile,
boolean cacheInstances)
throws [ObjectGridException](#)

Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

The returned ObjectGrid instances can be cached. An ObjectGridException will be thrown when attempting to cache a newly created ObjectGrid that has the same name as an ObjectGrid that has already been cached.

Parameters:

xmlFile - the file that defines an ObjectGrid or multiple ObjectGrids

cacheInstances - set to true to cache all ObjectGrid instances created based on the file

Returns:

a list of ObjectGrid instances

Throws:

[ObjectGridException](#) - if attempting to create and cache an ObjectGrid with the same name as an ObjectGrid that has already been cached, or any other error occurs during ObjectGrid creation

Since:

8.6, XC10 2.5

See Also:

[ObjectGrid](#)

createObjectGrids

[List](#) createObjectGrids([URL](#) xmlFile)
throws [ObjectGridException](#)

Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

The XML file will be validated against the schema and each ObjectGrid instance that is created will be cached. An ObjectGridException will be thrown when attempting to cache a newly created ObjectGrid that has the same name as an ObjectGrid that has already been cached. Using this method is equivalent to calling the createObjectGrids(URL, true, true) method.

Parameters:

xmlFile - The XML file to process. ObjectGrid(s) will be created based on the configurations what is in the file.

Returns:

A list of ObjectGrid instances that have been created.

Throws:

[ObjectGridException](#) - if attempting to create and cache an ObjectGrid with the same name as an ObjectGrid that has already been cached, or any other error occurs during ObjectGrid creation

See Also:

[createObjectGrids\(URL, boolean, boolean\)](#), [ObjectGrid](#)

createObjectGrid

```
ObjectGrid createObjectGrid(String objectGridName,  
                             URL xmlFile)  
throws ObjectGridException
```

Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

If there is no ObjectGrid with this name defined in the XML file, an ObjectGridException will be thrown. The XML file will be validated against the schema and the ObjectGrid instance created will be cached. Using this method is equivalent to calling the createObjectGrid(String, URL, true, true) method.

Parameters:

objectGridName - name of the ObjectGrid to create. This ObjectGrid should be defined in the XML file. Must not be null.
xmlFile - the XML file to process

Returns:

A newly created ObjectGrid

Throws:

[IllegalArgumentException](#) - if objectGridName is null
[ObjectGridException](#) - if an ObjectGrid with the same name has been previously cached, an ObjectGrid configuration with the specified name can be found in the xml file, or any other error occurs during ObjectGrid creation.

See Also:

[createObjectGrid\(String, URL, boolean, boolean\)](#), [ObjectGrid](#)

removeObjectGrid

```
void removeObjectGrid(String objectGridName)  
throws ObjectGridException
```

Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances.

Invoking this method is equivalent to calling removeObjectGrid(String, false)

Parameters:

objectGridName - the name of the ObjectGrid instance to remove from the cache

Throws:

[ObjectGridException](#) - if an ObjectGrid with the objectGridName was not found in the cache

See Also:

[removeObjectGrid\(String, boolean\)](#)

removeObjectGrid

```
void removeObjectGrid(String objectGridName,  
                     boolean destroy)  
throws ObjectGridException
```

Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances and optionally destroys its associated resources.

Parameters:

objectGridName - the name of the ObjectGrid instance to remove from the cache
destroy - if true, destroy the objectgrid instance and its associated resources

Throws:

[ObjectGridException](#) - if an ObjectGrid with the objectGridName was not found in the cache

See Also:

[ObjectGrid.destroy\(\)](#)

getObjectGrids

[List](#) getObjectGrids()

Gets a List of the local, in-memory ObjectGrid instances that have been previously cached.

This method returns null if no ObjectGrid instances have been cached.

Returns:

a list of ObjectGrid instances that have been previously cached or null if there are no cached ObjectGrid instances

getObjectGrid

[ObjectGrid](#) getObjectGrid([String](#) objectGridName)

Returns a cached local, in-memory ObjectGrid instance by name.

This method returns null if no ObjectGrid with the specified name is currently cached.

Parameters:

objectGridName - the cached objectgrid name.

Returns:

a cached ObjectGrid which currently exists.

setTraceSpecification

void setTraceSpecification([String](#) traceSpec)

Set the trace specification for the current JVM.

This operation is a replace operation, not an append operation. The specification should be of the form:

```
TraceString := <ComponentString>(:<ComponentString>)* ComponentString := <ComponentName>=<type>=<state>(,<type>=<state>)*  
ComponentName := a java String state := [enabled|disabled] type := [all|debug|event|entryExit]
```

For example, ObjectGrid=all=enabled

Parameters:

traceSpec - the new trace specification

getTraceSpecification

[String](#) getTraceSpecification()

Retrieves the trace specification for the current JVM.

Since:

7.1

See Also:

[setTraceSpecification\(String\)](#)

setTraceFileName

void **setTraceFileName**([String](#) traceFileName)

Sets the trace output to go to a file instead of System.out.

The supplied file name can be relative to the working directory or a fully-qualified file name.

Parameters:

traceFileName - Name of trace file

setTraceEnabled

void **setTraceEnabled**(boolean enabledFlag)

Enables or disables ObjectGrid trace for the JVM.

Disabling trace improves the performance when ObjectGrid runs on processors whose L2 caches are not large enough to contain the trace enabled code paths. If this is set to false, ObjectGrid trace is suppressed even if it is enabled using [setTraceSpecification\(String\)](#). By default ObjectGrid trace is enabled.

Parameters:

enabledFlag - true to enable trace

Since:

WAS XD 6.0.1

See Also:

[setTraceSpecification\(String\)](#)

connect

[ClientClusterContext](#) **connect**([ClientSecurityConfiguration](#) securityProps,
[URL](#) overrideObjectGridXml)
throws [ConnectException](#)

Use this method to connect to an embedded ObjectGrid server. An embedded ObjectGrid server is typically started in an application server process such as IBM WebSphere Application Server. This method allows connecting to the in-memory ObjectGrid server instance without specifying connection information.

This method can be used to connect to both dynamic and static ObjectGrid server deployments.

Parameters:

securityProps - client security configuration. The value can be null if not running in secure mode.

overrideObjectGridXml - This parameter can be null. If it is not null, the client side configuration of the ObjectGrid plugins are overridden with the ObjectGrid configuration using this URL. If this parameter is null, client side ObjectGrid plugins can be overridden by providing an overrideMap to [setOverrideObjectGridConfigurations\(Map\)](#) or by calling [putOverrideObjectGridConfigurations\(String, List\)](#).

In cases where this parameter is not null, and overriding configuration objects have been provided to the ObjectGridManager by `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)`, the configuration based on the XML file will be used to override ObjectGrid settings. Overriding objects provided to `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)` will be ignored.

Not all plugins can be overridden. For details please see the ObjectGrid documentation.

Returns:

a `ClientClusterContext` representing the cluster ObjectGrid configuration to which the client is connected.

Throws:

[ConnectException](#) - if any error occurs connecting to the server

Since:

WAS XD 6.0.1

See Also:

[ClientClusterContext](#), [ClientSecurityConfiguration](#)

connect

`ClientClusterContext connect(String catalogServerEndpoints, ClientSecurityConfiguration securityProps, URL overrideObjectGridXml)`
throws [ConnectException](#)

Connects a client process to a remote ObjectGrid catalog server or catalog server cluster (a dynamic ObjectGrid deployment topology). The result `ClientClusterContext` can then be used to get any ObjectGrid reference managed by that catalog server cluster.

Parameters:

`catalogServerEndpoints` - A concatenated list of host/port pairs belonging to the catalog servers in the form "host:port<,host:port>". This list can be arbitrarily long and is used for bootstrapping only. The first viable address will be used.

`securityProps` - This parameter may be null if the client does not wish to establish a secure connection with the server.

`overrideObjectGridXml` - This parameter can be null. If it is not null, the client side configuration of the ObjectGrid plugins are overridden with the ObjectGrid configuration using this URL. If this parameter is null, client side ObjectGrid plugins can be overridden by providing an `overrideMap` to `setOverrideObjectGridConfigurations(Map)` or by calling `putOverrideObjectGridConfigurations(String, List)`.

In cases where this parameter is not null, and overriding configuration objects have been provided to the ObjectGridManager by `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)`, the configuration based on the XML file will be used to override ObjectGrid settings. Overriding objects provided to `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)` will be ignored.

Not all plugins can be overridden. For details please see the ObjectGrid documentation.

Returns:

a `ClientClusterContext` representing the cluster ObjectGrid configuration to which the client is connected.

Throws:

[ConnectException](#) - If there is a problem connecting to the addresses given.

disconnect

boolean **disconnect**([ClientClusterContext](#) context)

Disconnects a client process from an ObjectGrid server

Parameters:

context - the [ClientClusterContext](#) object returned from a previous call to one of the connect methods The caller must guarantee this parameter is not null.

Returns:

true if the disconnect was successful, false if the supplied context was not connected

Throws:

[IllegalArgumentException](#) - if the [ClientClusterContext](#) is null

Since:

WAS XD 6.0.1

See Also:

[ClientClusterContext](#)

getObjectGrid

[ObjectGrid](#) **getObjectGrid**([ClientClusterContext](#) context,
[String](#) objectGridName)

Returns a client [ObjectGrid](#) instance corresponding to an [ObjectGrid](#) in an [ObjectGrid](#) cluster.

This method is equivalent to calling `getObjectGrid(context, objectGridName, null)`

Parameters:

context - the [ClientClusterContext](#) object returned from a previous call to one of the connect methods The caller must guarantee this parameter is not null.

objectGridName - the name of the requested client [ObjectGrid](#)

Returns:

a client [ObjectGrid](#) instance corresponding to a remote [ObjectGrid](#)

Throws:

[IllegalArgumentException](#) - if either provided parameter is null

[ObjectGridRuntimeException](#) - if the [ObjectGrid](#) with the specified name is not hosted in any eXtreme Scale servers managed by the catalog server

Since:

WAS XD 6.0.1

See Also:

[ClientClusterContext](#), [ObjectGrid](#), [getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

getObjectGrid

[ObjectGrid](#) **getObjectGrid**([ClientClusterContext](#) context,
[String](#) objectGridName,
[ObjectGridConfiguration](#) overrideOGConfig)

Returns a client [ObjectGrid](#) instance corresponding to an [ObjectGrid](#) in an [ObjectGrid](#) cluster.

This method replaces the `get/set/putOverrideObjectGridConfigurations` methods. Those methods had thread safety issues. In addition they were global in nature so we end up having configuration override happen for all client connections unless it was managed correctly. If [ClientClusterContext](#) was used previously to get an [ObjectGrid](#) for the given name, the same [ObjectGrid](#) instance is returned even if the `overrideOGConfig` parameter is different.

Parameters:

context - the `ClientClusterContext` object returned from a previous call to one of the connect methods. The caller must guarantee this parameter is not `null`.
objectGridName - the name of the requested client `ObjectGrid`
overrideOGConfig - This parameter can be `null`. If it is not `null`, the client side configuration of the `ObjectGrid` plugins are overridden with the `ObjectGridConfiguration` provided. The provided override configuration takes precedence over any other provided override configuration for the requested `ObjectGrid` name provided by the `connect`, `putOverrideObjectGridConfigurations`, and `putOverrideObjectGridConfigurations` methods.

Not all plugins can be overridden. For details please see the `ObjectGrid` documentation.

Returns:

a client `ObjectGrid` instance corresponding to a remote `ObjectGrid`

Since:

7.1.1

See Also:

[ObjectGridConfiguration](#), [ObjectGridConfigFactory](#)

setOverrideObjectGridConfigurations

```
void setOverrideObjectGridConfigurations(Map overrideMap)
```

Deprecated. *This method is replaced in 7.1.1 with the `getObjectGrid` method that takes a `ObjectGridConfigurations`.*

Override `ObjectGrid` settings on client side `ObjectGrids` by passing in a `Map` where each key corresponds to a cluster name or domain name and each value is a `List` of `ObjectGridConfiguration` objects to be overridden.

Client side configuration of `ObjectGrid` and `BackingMap` plugins are overridden using the `ObjectGridConfiguration` values provided in the `List`. To override a `Plugin`, each `ObjectGridConfiguration` object must have a name that matches the name of the `ObjectGrid` to be overridden. `BackingMapConfiguration` objects must have the same name as a `BackingMap` and be associated with the proper `ObjectGridConfiguration`.

Not all plugins can be overridden. `ObjectGrid` plugins that can be overridden on the client side are `TransactionCallback` and `ObjectGridEventListener`. `BackingMap` plugins that can be overridden on the client side are `Evictor` and `MapEventListener`. Settings for the builtin `Evictor` can also be altered on the `BackingMap`. These settings include `numberOfBuckets`, `timeToLive`, and `ttlEvictorType`.

Parameters:

overrideMap - a `Map` that will be used to override `ObjectGrid` settings on the client side. To override client side settings, each key of the `Map` must be a `String` with a value that corresponds to a cluster name or domain name. Each value of the `overrideMap` must be a `java.util.List`. The `List` elements must be `ObjectGridConfiguration` objects. Each call to a connect method with a `clusterName` that matches a key in the `overrideMap` will result in the client side settings being overridden using the `List` of `ObjectGridConfiguration` objects provided in the key's corresponding value. Pass in `null` to clear an `overrideMap` that was previously set and thereby remove any overriding settings from future connect calls.

Throws:

[IllegalArgumentException](#) - if any keys or values are `null` or if keys or values are of the wrong type

Since:

WAS XD 6.0.1.2

See Also:

[connect\(String, ClientSecurityConfiguration, URL\)](#), [connect\(ClientSecurityConfiguration,](#)

[URL](#)), [ObjectGridConfiguration](#), [BackingMapConfiguration](#),
[getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

putOverrideObjectGridConfigurations

```
void putOverrideObjectGridConfigurations(String clusterName,  
                                         List objectGridConfigList)
```

Deprecated. *This method is replaced in 7.1.1 with the `getObjectGrid` method that takes a `ObjectGridConfigurations`.*

Put an entry into the Map that is used to override client side ObjectGrid and BackingMap plugins.

Parameters:

`clusterName` - to be used as a key in the Map used to override client side ObjectGrid plugins. If a connect method is called with a matching `clusterName`, the client side ObjectGrid and BackingMap plugins can be overridden using the `objectGridConfigList`. In the dynamic environment, use the domain name to override ObjectGrid settings.

`objectGridConfigList` - a List of ObjectGridConfiguration objects that will be used to override client side ObjectGrid settings if a connect method is called with a cluster name that matches the `clusterName` on this method

Throws:

[IllegalArgumentException](#) - if the `clusterName` or `objectGridConfigList` is null

See Also:

[getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

getOverrideObjectGridConfigurations

```
Map getOverrideObjectGridConfigurations()
```

Deprecated. *This method is replaced in 7.1.1 with the `getObjectGrid` method that takes a `ObjectGridConfigurations`.*

Get the Map that is used to override client side ObjectGrid and BackingMap plugins.

Returns:

the Map that was set by the call to `setOverrideObjectGridConfigurations`. The Map may also have entries that were put there using the `putOverrideObjectGridConfigurations` method.

See Also:

[getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

getCatalogDomainManager

```
CatalogDomainManager getCatalogDomainManager()
```

Retrieve the CatalogDomainManager for this ObjectGridManager.

Returns:

the CatalogDomainManager, if available. Returns null if a CatalogDomainManager is not supported in the current runtime environment.

Since:

8.5

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class ObjectGridException

[java.lang.Object](#)

 [java.lang.Throwable](#)

 [java.lang.Exception](#)

 [com.ibm.websphere.objectgrid.ObjectGridException](#)

All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[CacheEntryException](#), [ConnectException](#), [ContinuousQueryException](#),
[DuplicateKeyException](#), [KeyNotFoundException](#), [LoaderException](#), [LockException](#),
[NoActiveTransactionException](#), [ObjectGridConfigurationException](#),
[ObjectGridSecurityException](#), [ReadOnlyException](#), [TransactionCallbackException](#),
[TransactionException](#), [UndefinedMapException](#)

```
public class ObjectGridException  
extends Exception  
implements IObjectGridException
```

Base exception class for all checked exceptions thrown by the ObjectGrid product.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridException](#)()

Constructs a new ObjectGridException with null as its detail message.

[ObjectGridException](#)([String](#) message)

Constructs a new ObjectGridException with the specified detail message.

[ObjectGridException](#)([String](#) message, [Throwable](#) cause)

Constructs a new ObjectGridException with the specified detail message and cause.

[ObjectGridException](#)([Throwable](#) cause)

Constructs a new ObjectGridException with a specified cause.

Method Summary

[I](#)
[h](#)
[r](#)
[o](#)
[w](#)
[a](#)
[b](#)

[getCause](#)()

Returns the cause of this ObjectGridException or null if the cause is nonexistent or unknown.

l
e

I
h
r
o
w
a
b
l
e

[initCause\(Throwable cause\)](#)

Initializes the *cause* of this ObjectGridException to the specified value.

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridException

```
public ObjectGridException()
```

Constructs a new ObjectGridException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[initCause\(Throwable\)](#)

ObjectGridException

```
public ObjectGridException(String message)
```

Constructs a new ObjectGridException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridException

```
public ObjectGridException(Throwable cause)
```

Constructs a new ObjectGridException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for ObjectGridExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:
[getCause\(\)](#)

ObjectGridException

```
public ObjectGridException(String message,  
                           Throwable cause)
```

Constructs a new ObjectGridException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ObjectGridException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[getCause\(\)](#), [Throwable.getMessage\(\)](#)

Method Detail

getCause

```
public Throwable getCause()
```

Returns the cause of this ObjectGridException or `null` if the cause is nonexistent or unknown. (The cause is the throwable that caused this ObjectGridException to get thrown.)

This implementation returns the cause that was supplied via one of the constructors requiring a `Throwable`, or that was set after creation with the `initCause(Throwable)` method. While it is typically unnecessary to override this method, a subclass can override it to return a cause set by some other means. This is appropriate for a "legacy chained throwable" that predates the addition of chained exceptions to `Throwable`. Note that it is *not* necessary to override any of the `PrintStackTrace` methods, all of which invoke the `getCause` method to determine the cause of an ObjectGridException

Specified by:

[getCause](#) in interface [IObjectGridException](#)

Overrides:

[getCause](#) in class [Throwable](#)

Returns:

the cause of this ObjectGridException or `null` if the cause is nonexistent or unknown.

See Also:

[ObjectGridException\(String, Throwable\)](#), [ObjectGridException\(Throwable\)](#),
[initCause\(Throwable\)](#)

initCause

```
public Throwable initCause(Throwable cause)
```

Initializes the *cause* of this ObjectGridException to the specified value. (The cause is the throwable that caused this ObjectGridException to get thrown.)

This method can be called at most once. It is generally called from within the constructor, or immediately after creating the ObjectGridException. If this

ObjectGridException was created with ObjectGridException(Throwable) or ObjectGridException(String,Throwable), this method cannot be called even once.

Specified by:

[initCause](#) in interface [IObjectGridException](#)

Overrides:

[initCause](#) in class [Throwable](#)

Parameters:

cause - the cause (which is saved for later retrieval by the `getCause()` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown.)

Returns:

a reference to this ObjectGridException instance.

Throws:

[IllegalArgumentException](#) - if cause is this ObjectGridException. (An ObjectGridException cannot be its own cause.)

[IllegalStateException](#) - if this ObjectGridException was created with ObjectGridException(Throwable) OR ObjectGridException(String,Throwable), or this method has already been called on this ObjectGridException.

See Also:

[ObjectGridException\(String, Throwable\)](#), [ObjectGridException\(Throwable\)](#), [getCause\(\)](#)

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
Prev Class	Next Class	Frames	No Frames	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

com.ibm.websphere.objectgrid

Interface ObjectGrid

public interface **ObjectGrid**

This object is used for creating sessions to the ObjectGrid. It is the central core of the ObjectGrid framework. Besides creating sessions, it is also responsible for defining BackingMaps, setting a TransactionCallback, adding event listeners, and managing the security settings.

Since:
WAS XD 6.0, XC10

Field Summary	
s t a t i c i n t	<p>CLIENT Indicates the ObjectGrid is a client-side distributed ObjectGrid.</p>
s t a t i c i n t	<p>DEFAULT_TX_TIMEOUT_VALUE The default transaction time out value of 10 minutes if no transaction time out value is set.</p>
s t a t i c i n t	<p>LOCAL Indicates the ObjectGrid is a local ObjectGrid.</p>
s t a t i c i n t	<p>SERVER Indicates the ObjectGrid is a server-side distributed ObjectGrid.</p>

Method Summary

v
o
i
d

[addEventListener](#)(com.ibm.websphere.objectgrid.plugins.EventListener listener)
Adds an EventListener.

v
o
i
d

[addEventListener](#)(ObjectGridEventListener listener)
Deprecated. *This method is deprecated in version 7.1.1, use the [addEventListener\(EventListener\)](#) method.*

B
a
c
k
i
n
g
M
a
p

[createMap](#)(String name)
Creates a BackingMap, but does not associate it with this ObjectGrid.

B
a
c
k
i
n
g
M
a
p

[defineMap](#)(String name)
Defines a BackingMap that will be used by the application.

v
o
i
d

[destroy](#)()
Destroys this instance.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
r
e
v
i
s
i
o
n

[getCollisionArbiter](#)()
Retrieves the CollisionArbiter that this grid is using to resolve revision collisions.

C
o
l
l
i
s
i
o
n
A
r
b
i
t
r

L
i
s
t

[getEventListeners\(\)](#)
Returns the current list of EventListeners.

L
i
s
t

[getListOfMapNames\(\)](#)
Gets the list of map names currently defined for this ObjectGrid instance.

B
a
c
k
i
n
g
M
a
p

[getMap\(String name\)](#)
Returns a BackingMap previously configured by calling the `defineMap(String)` or `setMaps(List)` method.

S
t
r
i
n
g

[getName\(\)](#)
Gets the name of this ObjectGrid.

i
n
t

[getObjectGridType\(\)](#)
Returns the type of ObjectGrid.

S
e
s
s
i
o
n

[getSession\(\)](#)
Gets a Session object that can be used by a single thread at a time.

S
e
s
s
i
o
n

[getSession\(CredentialGenerator credGen\)](#)
Get a session using a CredentialGenerator.

S
e
s
s
i
o
n

[getSession\(Subject subject\)](#)
Allows the use of a specific Subject rather than use the SubjectSource configured on the ObjectGrid to get a Session.

[getState\(\)](#)

Retrieve the current life cycle state of this ObjectGrid.

a c t i o n C a l l b a c k	getTransactionCallback() Retrieves the TransactionCallback object.
i n t	getTxIsolation() Retrieves the default transaction isolation level.
i n t	getTxTimeout() Gets transaction timeout setting for this ObjectGrid instance.
v o i d	initialize() Begins the bootstrapping of the ObjectGrid and Session instances.
b o o l e a n	isSecurityEnabled() Checks whether security is enabled on this ObjectGrid or not.
v o i d	registerEntities(Class[] entities) Register one or more entities based on the class metadata.
v o i d	registerEntities(URL entityXML) Registers one ore more entities from an entity XML file.
v o i d	removeEventListener(com.ibm.websphere.objectgrid.plugins.EventListener listener) Removes an EventListener.
v o i d	removeEventListener(ObjectGridEventListener listener) Deprecated. <i>This method is deprecated in version 7.1.1, use the removeEventListener(EventListener) method.</i>
i n t	reserveSlot(String containerName) Allows plugins on this ObjectGrid to reserve slots for use to store context data.
v o i d	setAccessByCreatorOnlyMode(int accessByCreatorOnlyMode) Set the "access by creator only" mode.
v o i d	setAuthorizationMechanism(int authMechanism) Sets the authorization mechanism.
v	setCollisionArbiter(com.ibm.websphere.objectgrid.revision.CollisionArbiter arbiter)

o i d	Sets the CollisionArbiter that is responsible for arbitration of revision conflicts.
v o i d	setEventListeners (List listeners) Deprecated. <i>This method is deprecated in version 7.1.1. Use the addEventListener(EventListener) or removeEventListener(EventListener) methods. Plug-ins that implement the ObjectGridLifecycleListener interface are automatically registered with the grid. Using this method will remove those automatically added listeners.</i>
v o i d	setMaps (List mapList) Clears any BackingMaps that have been previously defined on this ObjectGrid and replaces them with the List of BackingMaps provided.
v o i d	setName (String gridName) Sets the name of this ObjectGrid.
v o i d	setObjectGridAuthorization (com.ibm.websphere.objectgrid.security.plugins.ObjectGridAuthorization ogAuthorization) Sets the ObjectGridAuthorization for this ObjectGrid instance.
v o i d	setPermissionCheckPeriod (int period) Sets the permission check period.
v o i d	setQueryConfig (QueryConfig queryConfig) Set the QueryConfig object for this ObjectGrid.
v o i d	setSecurityEnabled () Enables the ObjectGrid security.
v o i d	setSubjectSource (com.ibm.websphere.objectgrid.security.plugins.SubjectSource source) Sets the SubjectSource plugin.
v o i d	setSubjectValidation (com.ibm.websphere.objectgrid.security.plugins.SubjectValidation subjectValidation) Sets the SubjectValidation for this ObjectGrid instance.
v o i d	setTransactionCallback (TransactionCallback callback) Sets the TransactionCallback object.
v o i d	setTxIsolation (int level) Sets the default transaction isolation level for all sessions created by the ObjectGrid.
v o i d	setTxTimeout (int timeout) Sets the transaction timeout value to a specified number of seconds.

Field Detail

DEFAULT_TX_TIMEOUT_VALUE

static final int **DEFAULT_TX_TIMEOUT_VALUE**

The default transaction time out value of 10 minutes if no transaction time out value is set.

Since:

WXS 7.1.0.0 FIX1

See Also:

[Constant Field Values](#)

LOCAL

static final int **LOCAL**

Indicates the ObjectGrid is a local ObjectGrid.

See Also:

[Constant Field Values](#)

SERVER

static final int **SERVER**

Indicates the ObjectGrid is a server-side distributed ObjectGrid.

See Also:

[Constant Field Values](#)

CLIENT

static final int **CLIENT**

Indicates the ObjectGrid is a client-side distributed ObjectGrid.

See Also:

[Constant Field Values](#)

Method Detail

getSession

[Session](#) getSession()

throws [ObjectGridException](#),
[TransactionCallbackException](#)

Gets a Session object that can be used by a single thread at a time.

It is not allowed to share this Session object between threads without placing a critical section around it. While the core framework allows the object to move between threads, the TransactionCallback and Loader may prevent this usage, especially in J2EE environments.

When the ObjectGrid is a local ObjectGrid, and its security is enabled, this method will use the SubjectSource to get a Subject object and then associate the Subject object with this session .

When the ObjectGrid is a distributed ObjectGrid (client server mode), and its security is

enabled, this method will utilize the client server security infrastructure to get a secure session.

If the `initialize()` method has not been invoked prior to the first `getSession` invocation, an implicit initialization will occur. This ensures that all of the configuration is complete before any runtime usage is required.

Returns:

An instance of `Session`

Throws:

[ObjectGridException](#) - if an error occurs during processing

[TransactionCallbackException](#) - if the `TransactionCallback` throws an exception

[IllegalStateException](#) - if this method is called after the `destroy()` method is called.

See Also:

[destroy\(\)](#), [initialize\(\)](#), [Session](#), `SubjectSource`

getSession

`Session getSession(Subject subject)`
throws [ObjectGridException](#),
[TransactionCallbackException](#),
`com.ibm.websphere.objectgrid.security.plugins.InvalidSubjectException`

Allows the use of a specific `Subject` rather than use the `SubjectSource` configured on the `ObjectGrid` to get a `Session`.

This method should only be used when `ObjectGrid` security is enabled. If the `ObjectGrid` security is disabled, the provided `Subject` object will not be used.

If the `initialize()` method has not been invoked prior to the first `getSession` invocation, an implicit initialization will occur. This ensures that all of the configuration is complete before any runtime usage is required.

Parameters:

`subject` - `Subject` to associate with the returned `Session`

Returns:

An instance of `Session`

Throws:

[ObjectGridException](#) - if an error occurs during processing

[TransactionCallbackException](#) - if the `TransactionCallback` throws an exception

`com.ibm.websphere.objectgrid.security.plugins.InvalidSubjectException` - the `subject` passed in is invalid based on the `SubjectValidation` mechanism.

[IllegalStateException](#) - if this method is called after the `destroy()` method is called.

See Also:

[destroy\(\)](#), [initialize\(\)](#), [Session](#), `SubjectValidation`

setTransactionCallback

`void setTransactionCallback(TransactionCallback callback)`

Sets the `TransactionCallback` object.

A single cache is a single domain. All `Loaders` defined for `BackingMaps` in an `ObjectGrid` will normally cooperate, thus a corresponding `TransactionCallback` object needs to be set on the `ObjectGrid`.

A `TransactionCallback` that implements the `ObjectGridLifecycleListener` interface is automatically added as if the [addEventListener\(EventListener\)](#) method was called. Any previous callback which implements `ObjectGridLifecycleListener` interface is removed as if the [removeEventListener\(EventListener\)](#) method was called.

A TransactionCallback may implement the ObjectGridPlugin interface in order to receive enhanced ObjectGrid plug-in lifecycle method calls. The plug-in is also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

`callback` - An instance of a TransactionCallback

Throws:

[IllegalArgumentException](#) - if `callback` is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#), [TransactionCallback](#)

getTransactionCallback

[TransactionCallback](#) `getTransactionCallback()`

Retrieves the TransactionCallback object.

The TransactionCallback can be used in conjunction with the TXID to house transaction-specific context data, such as the connection to the database.

Returns:

the argument that was passed to the `setTransactionCallback(TransactionCallback)` method of this interface or a default TransactionCallback object if `setTransactionCallback` was not previously called for this ObjectGrid.

See Also:

[setTransactionCallback\(TransactionCallback\)](#), [TransactionCallback](#)

setCollisionArbiter

`void setCollisionArbiter(com.ibm.websphere.objectgrid.revision.CollisionArbiter arbiter)`

Sets the CollisionArbiter that is responsible for arbitration of revision conflicts.

A CollisionArbiter that implements the ObjectGridLifecycleListener interface is automatically added as if the [addEventListener\(EventListener\)](#) method was called. Any previous arbiter which implements ObjectGridLifecycleListener interface is removed as if the [removeEventListener\(EventListener\)](#) method was called.

A CollisionArbiter may implement the ObjectGridPlugin interface in order to receive enhanced ObjectGrid plug-in lifecycle method calls. The plug-in is also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

`arbiter` - The arbitration logic that will be used to resolve collisions.

Since:

7.1

getCollisionArbiter

`com.ibm.websphere.objectgrid.revision.CollisionArbiter getCollisionArbiter()`

Retrieves the CollisionArbiter that this grid is using to resolve revision collisions.

Returns:

The arbitration logic that is responsible for resolving revision collisions.

Since:

defineMap

[BackingMap](#) `defineMap(String name)`

Defines a BackingMap that will be used by the application.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

name - the name of the map being defined.

Returns:

a BackingMap reference

Throws:

[IllegalArgumentException](#) - if name is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#), [BackingMap](#)

createMap

[BackingMap](#) `createMap(String name)`

Creates a BackingMap, but does not associate it with this ObjectGrid.

This method is to be used in tandem with the `setMaps(List)` method, which will associate BackingMaps with this ObjectGrid. These methods are for use when configuring an ObjectGrid with the Spring Framework.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

name - the name of the map being defined.

Returns:

a BackingMap reference

Throws:

[IllegalArgumentException](#) - if name is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

Since:

WAS XD 6.0.1

See Also:

[initialize\(\)](#), [setMaps\(List\)](#)

setMaps

void `setMaps(List mapList)`

Clears any BackingMaps that have been previously defined on this ObjectGrid and replaces them with the List of BackingMaps provided.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

mapList - a list of BackingMaps to set on this ObjectGrid.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.0.1

See Also:

[createMap\(String\)](#), [initialize\(\)](#)

getListOfMapNames

[List](#) `getListOfMapNames()`

Gets the list of map names currently defined for this ObjectGrid instance.

Note, once the initialize() method is called, the List returned will not change. However, it could change if called prior to initialization. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Returns:

a List of String objects, one String per map that was previously configured by the defineMap(String) or setMaps(List) method. An empty List is returned if no maps are currently defined.

See Also:

[defineMap\(String\)](#), [initialize\(\)](#), [setMaps\(List\)](#)

getMap

[BackingMap](#) `getMap(String name)`

Returns a BackingMap previously configured by calling the defineMap(String) or setMaps(List) method.

Parameters:

name - the same name that was used as an argument to the defineMap(String) or createMap(String) method. A null reference is returned if a map is not associated with this ObjectGrid for the specified map name.

Returns:

the BackingMap instance

See Also:

[createMap\(String\)](#), [defineMap\(String\)](#), [setMaps\(List\)](#)

initialize

void `initialize()`

throws [ObjectGridException](#)

Begins the bootstrapping of the ObjectGrid and Session instances.

After this method has been invoked, the configuration of the ObjectGrid is considered complete and is ready for runtime usage. Any additional configuration method invocations, such as defineMap(String), will result in an exception. This method is considered optional since the first call to one of the getSession methods will perform an implicit initialization.

Throws:

[ObjectGridException](#) - if an error occurs during processing

addEventListener

void `addEventListener(com.ibm.websphere.objectgrid.plugins.EventListener listener)`

Adds an EventListener.

Significant events will be communicated to interested listeners through the ObjectGridEventListener and ObjectGridLifecycleListener callback interface. Multiple event listeners are allowed to be registered, with no implied ordering of event notifications.

Note, this method is allowed to be invoked before and after the initialize() method.

Object grid plug-ins (TransactionCallback, CollisionArbiter) that implement the ObjectGridLifecycleListener are automatically added as lifecycle listeners when added to the ObjectGrid.

Parameters:

listener - An instance of ObjectGridEventListener or ObjectGridLifecycleListener

Throws:

[IllegalArgumentException](#) - if listener is null or not an instance of ObjectGridEventListener, ObjectGridLifecycleListener.

[IllegalStateException](#) - if this method is called during initialization by one of the configured plugins and the ObjectGrid runtime is not in a usable state to initialize the ObjectGridEventListener.

See Also:

[ObjectGridEventListener](#), ObjectGridLifecycleListener, EventListener

addEventListener

void **addEventListener**([ObjectGridEventListener](#) listener)

Deprecated. *This method is deprecated in version 7.1.1, use the [addEventListener\(EventListener\)](#) method.*

Provided for compatibility with old releases, use the [addEventListener\(EventListener\)](#) method.

Parameters:

listener -

removeEventListener

void **removeEventListener**(com.ibm.websphere.objectgrid.plugins.EventListener listener)

Removes an EventListener.

This method removes an ObjectGridEventListener or ObjectGridLifecycleListener that was previously added to this object using the addEventListener(ObjectGridEventListener) or setEventListeners(List) method. If the desired ObjectGridEventListener is not found, no error will be returned.

Note, this method is allowed to be invoked before and after the initialize() method. Object grid plug-ins (TransactionCallback, CollisionArbiter) that implement the ObjectGridLifecycleListener are automatically removed as lifecycle listeners when removed from the ObjectGrid.

Parameters:

listener - An instance of ObjectGridEventListener or ObjectGridLifecycleListener

Throws:

[IllegalArgumentException](#) - if listener is null or not an instance of ObjectGridEventListener, ObjectGridLifecycleListener

See Also:

[addEventListener\(EventListener\)](#), [ObjectGridEventListener](#), EventListener

removeEventListener

void **removeEventListener**([ObjectGridEventListener](#) listener)

Deprecated. *This method is deprecated in version 7.1.1, use the [removeEventListener\(EventListener\)](#) method.*

Provided for compatibility with old releases, use the [removeEventListener\(EventListener\)](#) method.

Parameters:

listener -

setEventListeners

[@Deprecated](#)

void **setEventListeners**([List](#) listeners)

Deprecated. *This method is deprecated in version 7.1.1. Use the [addEventListener\(EventListener\)](#) or [removeEventListener\(EventListener\)](#) methods. Plug-ins that implement the `ObjectGridLifecycleListener` interface are automatically registered with the grid. Using this method will remove those automatically added listeners.*

This overwrites the current list of EventListeners and replaces it with the supplied List of EventListeners

Note, this method is allowed to be invoked before and after the `initialize()` method.

Parameters:

listeners - List of `ObjectGridEventListeners` and `ObjectGridLifecycleListener` instances

Throws:

[ClassCastException](#) - if one of the elements in the provided list is not an instance of `ObjectGridEventListener`

[IllegalArgumentException](#) - if listeners is null, contains a null reference, or contains an instance of a type other than `ObjectGridEventListener` and `ObjectGridLifecycleListener`

[IllegalStateException](#) - if this method is called during initialization by one of the configured plugins and the `ObjectGrid` runtime is not in a usable state to initialize the `ObjectGridEventListener` objects.

See Also:

`EventListener`, [addEventListener\(EventListener\)](#), [removeEventListener\(EventListener\)](#)

getEventListeners

[List](#) `getEventListeners()`

Returns the current list of EventListeners.

Returns:

The current list of EventListeners.

See Also:

[addEventListener\(EventListener\)](#), `EventListener`, [ObjectGridEventListener](#), `ObjectGridLifecycleListener`

getName

[String](#) `getName()`

Gets the name of this `ObjectGrid`.

This method is useful for authorization as all Maps are prefixed with the `ObjectGrid` name.

Returns:

The name of the `ObjectGrid`.

See Also:

[setName\(String\)](#)

setName

void `setName(String gridName)`

Sets the name of this `ObjectGrid`. Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

gridName - The `ObjectGrid` name to use.

Throws:

[IllegalArgumentException](#) - if gridName is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

reserveSlot

int `reserveSlot(String containerName)`

Allows plugins on this `ObjectGrid` to reserve slots for use to store context data.

Currently the TxID object is the only object that uses slots for storing context data. TxID slots are used for storing transactional context data.

Once a slot is reserved, the slot assignment is permanent and cannot be given back. Note, this method is allowed to be invoked before and after the initialize() method.

Parameters:

containerName - The name of the Object with the slots. Currently TxID.SLOT_NAME is the only supported value for this argument.

Returns:

The slot index to use.

Throws:

[IllegalArgumentException](#) - if containerName is not TxID.SLOT_NAME.

See Also:

[TxID.SLOT_NAME](#), [TxID.getSlot\(int\)](#), [TxID.putSlot\(int, Object\)](#)

setSubjectValidation

void **setSubjectValidation**(com.ibm.websphere.objectgrid.security.plugins.SubjectValidation subjectValidation)

Sets the SubjectValidation for this ObjectGrid instance.

Passing null to this method removes a previously set SubjectValidation object from an earlier invocation of this method and indicates that this ObjectGrid is not associated with a SubjectValidation object.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provided SubjectValidation object will not be used.

A SubjectValidation plugin can be used to validate the Subject object passed in is a valid Subject. Please refer to SubjectValidation for more details.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:

subjectValidation - the SubjectValidation plugin

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

See Also:

[getSession\(Subject\)](#), [initialize\(\)](#), SubjectValidation

setAuthorizationMechanism

void **setAuthorizationMechanism**(int authMechanism)

Sets the authorization mechanism.

If this method is not invoked, the default authorization mechanism is SecurityConstants.AUTHORIZATION_MECHANISM_JAAS.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provide authorization mechanism will not be used.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:

authMechanism - the authorization mechanism, must be one of the final static variable on the SecurityConstants class.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

See Also:

[initialize\(\)](#), [SecurityConstants.AUTHORIZATION_MECHANISM_CUSTOM](#), [SecurityConstants.AUTHORIZATION_MECHANISM_JAAS](#)

setSecurityEnabled

void **setSecurityEnabled()**

Enables the ObjectGrid security.

Security on the ObjectGrid level refers to ObjectGrid authorizations.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Throws:

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#)

isSecurityEnabled

boolean **isSecurityEnabled()**

Checks whether security is enabled on this ObjectGrid or not.

Security on the ObjectGrid level refers to ObjectGrid authorizations. Security is disabled by default.

Returns:

true if security is enabled on this ObjectGrid; false otherwise.

See Also:

[setSecurityEnabled\(\)](#)

setPermissionCheckPeriod

void **setPermissionCheckPeriod**(int period)

Sets the permission check period.

This method takes a single parameter indicating how often the customer wants to check the permission used to allow a client access. If the parameter is 0 then every single authorized operation call will ask the authorization mechanism, either JAAS authorization or custom authorization to check if the current Subject has permission. This approach may be prohibitively expensive from a performance point of view depending on the authorization implementation, but if it is required then you can do it. Alternatively, if the parameter is > 0 then it indicates the number of seconds to cache a set of permissions before returning to the authorization mechanism to refresh them. This mechanism provides much better performance, but you run the risk that if the back-end permissions are changed during this time, the ObjectGrid will possibly allow or prevent access even though the back-end security provider has been modified.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

period - the permission check period in seconds.

Throws:

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#)

setSubjectSource

void **setSubjectSource**(com.ibm.websphere.objectgrid.security.plugins.SubjectSource source)

Sets the SubjectSource plugin.

Passing null to this method removes a previously set SubjectSource object from an earlier invocation of this method and indicates that this ObjectGrid is not associated with a SubjectSource object.

A SubjectSource plugin can be used to get a Subject object from the environment to represent the ObjectGrid client.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provided SubjectSource object will not be used.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:

source - the SubjectSource plugin

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

See Also:

[initialize\(\)](#), SubjectSource

setTxTimeout

void **setTxTimeout**(int timeout)

Sets the transaction timeout value to a specified number of seconds.

Any transaction that is started by use of a Session returned by one of the getSession methods on this interface must complete within the number of seconds specified by the transaction timeout parameter of this method. The timeout value is the maximum number of seconds the transaction is allowed to execute. If a transaction executes longer than this amount, a TransactionTimeoutException is thrown and the transaction is rolled back even if commit is requested.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

The transaction timeout is used by any transaction started by a Session that is returned by the getSession methods of this interface. Since this method must be called prior to getSession method to avoid IllegalStateException, this method only affects transactions that are started after this method is called. If this method is never called, the transaction is allowed unlimited amount of time to complete.

Parameters:

timeout - is the transaction timeout value in seconds. Use a value of 0 to indicate a transaction is allowed unlimited amount of time so that no TransactionTimeoutException ever occurs.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.0.1

See Also:

[initialize\(\)](#), [Session.TRANSACTION_NO_TIMEOUT](#), [Session.setTransactionTimeout\(int\)](#), [TransactionTimeoutException](#)

getTxTimeout

int **getTxTimeout**()

Gets transaction timeout setting for this ObjectGrid instance.

Returns:

timeout value that was passed to the setTxTimeout(int) method or 0 if setTxTimeout was never called.

Since:

WAS XD 6.0.1

See Also:

[setTxTimeout\(int\)](#)

setTxIsolation

void **setTxIsolation**(int level)

Sets the default transaction isolation level for all sessions created by the ObjectGrid. The constants defined in the Session interface are the possible transaction isolation levels. The default is [Session.TRANSACTION_REPEATABLE_READ](#).

Parameters:

level - one of the following Session constants: [Session.TRANSACTION_READ_UNCOMMITTED](#), [Session.TRANSACTION_READ_COMMITTED](#) or [Session.TRANSACTION_REPEATABLE_READ](#)

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

[IllegalArgumentException](#) - if this method includes an invalid transaction isolation level.

Since:

7.1.1

getTxIsolation

int **getTxIsolation**()

Retrieves the default transaction isolation level.

Returns:

the current transaction isolation level.

Since:

7.1.1

See Also:

[setTxIsolation\(int\)](#)

destroy

void **destroy**()

Destroys this instance.

This method should be invoked when the ObjectGrid is no longer being used. When this method is called, the ObjectGrid can free up any resources it is using. No new Sessions can be created or used after the destroy() has been invoked. Any in-flight Sessions will be allowed to continue, if the resources are still available to complete processing.

getSession

[Session](#) **getSession**([CredentialGenerator](#) credGen)
throws [ObjectGridException](#),
[TransactionCallbackException](#)

Get a session using a CredentialGenerator.

This method can only be called by the ObjectGrid client in an ObjectGrid client server environment. If ObjectGrid is used in a local model, that is, within the same JVM with no client or server existing, getSession(Subject) or the SubjectSource plugin should be used to secure the ObjectGrid.

If the initialize() method has not been invoked prior to the first getSession invocation, an implicit initialization will occur. This ensures that all of the configuration is complete before any runtime usage is required.

Parameters:

credGen - A CredentialGenerator for generating a credential for the session returned.

Returns:

An instance of Session

Throws:

[ObjectGridException](#) - if an error occurs during processing

[TransactionCallbackException](#) - if the TransactionCallback throws an exception
[IllegalStateException](#) - if this method is called after the destroy() method is called.

Since:

WAS XD 6.0.1

See Also:

[destroy\(\)](#), [initialize\(\)](#), [CredentialGenerator](#), [Session](#)

setQueryConfig

void **setQueryConfig**([QueryConfig](#) queryConfig)

Set the QueryConfig object for this ObjectGrid. A QueryConfig object provides query configurations for executing object queries over the maps in this ObjectGrid.

Parameters:

queryConfig - The QueryConfig to associate with this ObjectGrid instance.

Throws:

[IllegalArgumentException](#) - if queryConfig is null.

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.1

See Also:

[QueryConfig](#)

registerEntities

void **registerEntities**([URL](#) entityXML)

Registers one or more entities from an entity XML file.

Entity registration is required prior to ObjectGrid initialization to bind an Entity with a BackingMap and any defined indices.

This method may be called multiple times.

Parameters:

entityXML - the URL of the entity XML that defines the entities.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.1

registerEntities

void **registerEntities**([Class](#)[] entities)

Register one or more entities based on the class metadata.

Entity registration is required prior to ObjectGrid initialization to bind an Entity with a BackingMap and any defined indices.

This method may be called multiple times.

Parameters:

entities - one or more annotated entity classes to register as entities.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.1

getObjectGridType

int **getObjectGridType**()

Returns the type of ObjectGrid.

The return value is equivalent to one of the constants declared on this interface, [LOCAL](#), [SERVER](#), or [CLIENT](#).

Returns:
the ObjectGrid type

Since:
WAS XD 6.1

setObjectGridAuthorization

void **setObjectGridAuthorization**(com.ibm.websphere.objectgrid.security.plugins.ObjectGridAuthorization ogAuthorization)

Sets the ObjectGridAuthorization for this ObjectGrid instance.

Passing null to this method removes a previously set ObjectGridAuthorization object from an earlier invocation of this method and indicates that this ObjectGrid is not associated with a ObjectGridAuthorization object.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provided ObjectGridAuthorization object will not be used.

A ObjectGridAuthorization plugin can be used to authorize access to the ObjectGrid and maps. Please refer to ObjectGridAuthorization for more details.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:
ogAuthorization - the ObjectGridAuthorization plugin

Throws:
[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:
WAS XD 6.1

See Also:
[initialize\(\)](#), ObjectGridAuthorization

setAccessByCreatorOnlyMode

void **setAccessByCreatorOnlyMode**(int accessByCreatorOnlyMode)

Set the "access by creator only" mode.

Enabling "access by creator only" mode ensures that only the user (represented by the Principals associated with it), who inserts the record into the map, can access (read, update, invalidate, and remove) the record.

The "access by creator only" mode can be disabled, or can complement the ObjectGrid authorization model, or it can supersede the ObjectGrid authorization model. The default value is disabled: [SecurityConstants.ACCESS_BY_CREATOR_ONLY_DISABLED](#).

Parameters:
accessByCreatorOnlyMode - the access by creator mode.

Since:
WAS XD 6.1 FIX3

See Also:
[SecurityConstants.ACCESS_BY_CREATOR_ONLY_DISABLED](#),
[SecurityConstants.ACCESS_BY_CREATOR_ONLY_COMPLEMENT](#),
[SecurityConstants.ACCESS_BY_CREATOR_ONLY_SUPERSEDE](#)

getState

com.ibm.websphere.objectgrid.plugins.ObjectGridLifecycleListener.State **getState**()

Retrieve the current life cycle state of this ObjectGrid.

Returns:

the current state.
Since:
7.1.1

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

Release 2.5 Client API Specification

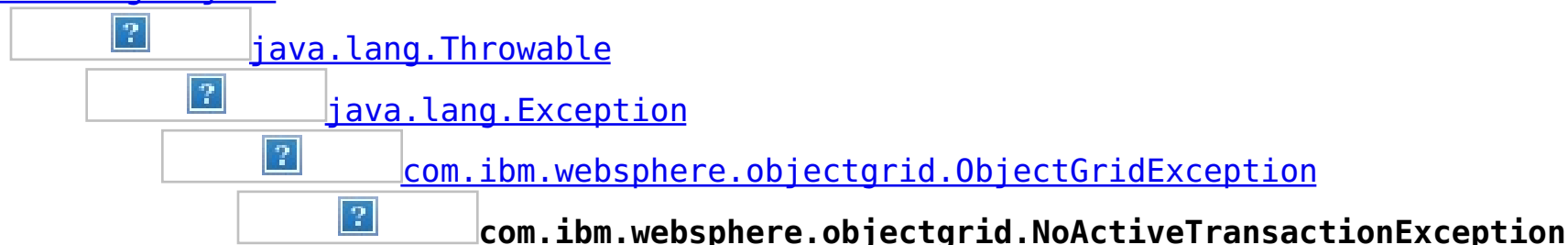
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class NoActiveTransactionException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class NoActiveTransactionException
extends ObjectGridException
```

An exception indicating there is no active transaction.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[NoActiveTransactionException](#)()

Constructs a new NoActiveTransactionException with null as its detail message.

[NoActiveTransactionException](#)(String message)

Constructs a new NoActiveTransactionException with the specified detail message.

[NoActiveTransactionException](#)(String message, Throwable cause)

Constructs a new NoActiveTransactionException with the specified detail message and cause.

[NoActiveTransactionException](#)(Throwable cause)

Constructs a new NoActiveTransactionException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

NoActiveTransactionException

```
public NoActiveTransactionException()
```

Constructs a new NoActiveTransactionException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

NoActiveTransactionException

```
public NoActiveTransactionException(String message)
```

Constructs a new NoActiveTransactionException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

NoActiveTransactionException

```
public NoActiveTransactionException(String message,  
                                   Throwable cause)
```

Constructs a new NoActiveTransactionException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this NoActiveTransactionException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

NoActiveTransactionException

```
public NoActiveTransactionException(Throwable cause)
```

Constructs a new NoActiveTransactionException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for NoActiveTransactionExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

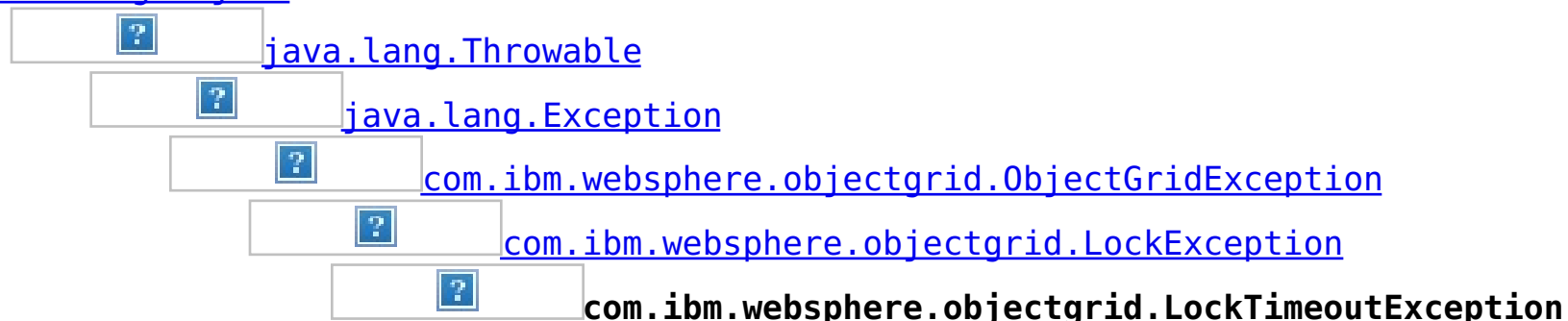
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class LockTimeoutException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[LockDeadlockException](#), [LockInternalFailureException](#)

```
public class LockTimeoutException
extends LockException
```

This exception is used by the lock manager to indicate that the maximum wait time for a lock has been exceeded. The timeout may or may not be the result of a deadlock. If it is a deadlock, the timeout is used to break the deadlock.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

LockTimeoutException ()	Constructs a new LockTimeoutException with null as its detail message.
LockTimeoutException (String message)	Constructs a new LockTimeoutException with the specified detail message.
LockTimeoutException (String message, Throwable cause)	Constructs a new LockTimeoutException with the specified detail message and cause.
LockTimeoutException (Throwable cause)	Constructs a new LockTimeoutException with a specified cause.

Method Summary

v o i d	forceJavaCore ()
S t	getLockRequestQueueDetails ()

r i n g	Provides detailed information about the state of the lock queue when the lock timeout occurred.
S t r i n g	getMessage() Returns the detail message string of this exception.
v o i d	setLockRequestQueueDetails(String string) Sets the details of the lock requests on the lock request queue at the time the lock timeout occurred.

Methods inherited from class [com.ibm.websphere.objectgrid.ObjectGridException](#)
[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)
[fillInStackTrace](#), [getLocalizedMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)
[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockTimeoutException

```
public LockTimeoutException()
```

Constructs a new LockTimeoutException with null as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockTimeoutException

```
public LockTimeoutException(String message)
```

Constructs a new LockTimeoutException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [getMessage\(\)](#)

LockTimeoutException

```
public LockTimeoutException(String message,
                             Throwable cause)
```


Constructs a new `LockTimeoutException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `LockTimeoutException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [getMessage\(\)](#)

LockTimeoutException

```
public LockTimeoutException(Throwable cause)
```

Constructs a new `LockTimeoutException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `LockTimeoutExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Method Detail

getLockRequestQueueDetails

```
public String getLockRequestQueueDetails()
```

Provides detailed information about the state of the lock queue when the lock timeout occurred.

Returns:

the argument that was passed to the `setLockRequestQueueDetails(String)` method of this class or `null` if the `setLockRequestQueueDetails` method was not previously called for this object.

forceJavaCore

```
public void forceJavaCore()
```

setLockRequestQueueDetails

```
public void setLockRequestQueueDetails(String string)
```

Sets the details of the lock requests on the lock request queue at the time the lock timeout occurred.

Parameters:

`string` - the details of lock requests on the lock request queue at the time the lock timeout occurred.

getMessage

public [String](#) getMessage()

Returns the detail message string of this exception. The returned String includes the request queue details as well as the message provided to the constructor.

Overrides:

[getMessage](#) in class [Throwable](#)

Returns:

the detail message string of this object instance

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All				
			Classes					
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD					

[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid
Class LockStrategy

[java.lang.Object](#)



```
public final class LockStrategy
extends Object
```

LockStrategy provides an enumerated type idiom for use on the BackingMap.setLockStrategy(LockStrategy) method. It determines whether or not a lock manager is needed for a BackingMap and if so, whether to use an optimistic or pessimistic locking strategy.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.setLockStrategy\(LockStrategy\)](#)

Field Summary

s t a t i c L O C K S T R A T E G Y	<p>NONE</p> <p>NONE indicates internal LockManager use is not needed since concurrency control is provided outside of ObjectGrid either by a persistence manager using objectgrid as a side cache, the application, or by a Loader plugin (for example, uses database locks to control concurrency).</p>
s t a t i c L O C K S T R A T E G Y	<p>OPTIMISTIC</p> <p>OPTIMISTIC is typically used for a map that does not have a Loader plugin, the map is read mostly, and locking is neither provided by a persistence manager using ObjectGrid as a side cache or by the application itself.</p>
s	

t
a
t
i
c
L
o
c
k
S
t
r
a
t
e
g
y

PESSIMISTIC

PESSIMISTIC is typically used for a map that does not have a Loader plugin and locking is neither provided by a persistence manager using ObjectGrid as a side cache, by a Loader plugin, or by the application itself.

Method Summary

S
t
r
i
n
g

toString()

Returns a string representation of the LockStrategy.

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

NONE

```
public static final LockStrategy NONE
```

NONE indicates internal LockManager use is not needed since concurrency control is provided outside of ObjectGrid either by a persistence manager using objectgrid as a side cache, the application, or by a Loader plugin (for example, uses database locks to control concurrency).

OPTIMISTIC

```
public static final LockStrategy OPTIMISTIC
```

OPTIMISTIC is typically used for a map that does not have a Loader plugin, the map is read mostly, and locking is neither provided by a persistence manager using ObjectGrid as a side cache or by the application itself. For this strategy, an exclusive lock is obtained on a map entry being inserted, updated, or removed at commit time. The lock ensures version information cannot be changed by another transaction while the transaction being committed is performing an optimistic version check.

PESSIMISTIC

```
public static final LockStrategy PESSIMISTIC
```

PESSIMISTIC is typically used for a map that does not have a Loader plugin and locking is neither provided by a persistence manager using ObjectGrid as a side cache, by a Loader plugin, or by the application itself. It is typically used when optimistic approach fails too often since there are update transactions that frequently collide on the same

map entry (e.g. not a read mostly map or large number of clients accessing a small map).

Method Detail

toString

```
public String toString()
```

Returns a string representation of the LockStrategy.

Overrides:

[toString](#) in class [Object](#)

Returns:

a string representation of the LockStrategy.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class LockInternalFailureException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class LockInternalFailureException
extends LockTimeoutException
```

This exception is used by the lock manager to indicate it detected some internal programming error while processing a lock or unlock request.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[LockInternalFailureException](#)()

Constructs a new LockInternalFailureException with null as its detail message.

[LockInternalFailureException](#)(String message)

Constructs a new LockInternalFailureException with the specified detail message.

[LockInternalFailureException](#)(String message, Throwable cause)

Constructs a new LockInternalFailureException with the specified detail message and cause.

[LockInternalFailureException](#)(Throwable cause)

Constructs a new LockInternalFailureException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[LockTimeoutException](#)

[forceJavaCore](#), [getLockRequestQueueDetails](#), [getMessage](#), [setLockRequestQueueDetails](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockInternalFailureException

```
public LockInternalFailureException()
```

Constructs a new `LockInternalFailureException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockInternalFailureException

```
public LockInternalFailureException(String message)
```

Constructs a new `LockInternalFailureException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [LockTimeoutException.getMessage\(\)](#)

LockInternalFailureException

```
public LockInternalFailureException(String message,  
                                   Throwable cause)
```

Constructs a new `LockInternalFailureException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `LockInternalFailureException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [LockTimeoutException.getMessage\(\)](#)

LockInternalFailureException

public **LockInternalFailureException**([Throwable](#) cause)

Constructs a new **LockInternalFailureException** with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for **LockInternalFailureExceptions** that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A null value is permitted and indicates that the cause is nonexistent or is unknown.

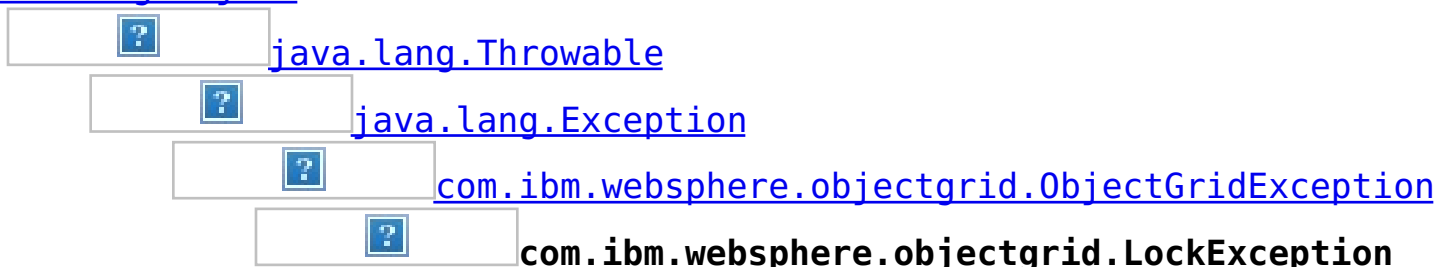
See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid
Class LockException

[java.lang.Object](#)



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:
[LockTimeoutException](#)

```
public class LockException
extends ObjectGridException
```

A general locking exception indicating something went wrong with locking operations.

Since:
WAS XD 6.0, XC10

See Also:
[LockTimeoutException](#), [Serialized Form](#)

Constructor Summary

- [LockException](#)()
Constructs a new LockException with null as its detail message.
- [LockException](#)(String message)
Constructs a new LockException with the specified detail message.
- [LockException](#)(String message, Throwable cause)
Constructs a new LockException with the specified detail message and cause.
- [LockException](#)(Throwable cause)
Constructs a new LockException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)
[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)
[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockException

```
public LockException()
```

Constructs a new LockException with null as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockException

```
public LockException(String message)
```

Constructs a new LockException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

LockException

```
public LockException(String message,  
                    Throwable cause)
```

Constructs a new LockException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this LockException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

LockException

```
public LockException(Throwable cause)
```

Constructs a new LockException with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of cause). This constructor is useful for LockExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for

later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							

[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class LockDeadlockException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class LockDeadlockException
extends LockTimeoutException
```

This exception is used by the lock manager to indicate that it detected a deadlock. It prevents the deadlock by throwing this exception. Typically, this deadlock is a result of the following scenario: one transaction owns a weaker lock as a result of getting a map entry, and then, at commit time, the transaction attempts to promote the weaker lock to a stronger lock in order to apply the changes to the data store. For example, two transactions try to promote from shared locks to exclusive locks but each transaction already owns a shared lock.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[LockDeadlockException](#)()

Constructs a new LockDeadlockException with null as its detail message.

[LockDeadlockException](#)(String message)

Constructs a new LockDeadlockException with the specified detail message.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[LockTimeoutException](#)

[forceJavaCore](#), [getLockRequestQueueDetails](#), [getMessage](#), [setLockRequestQueueDetails](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockDeadlockException

```
public LockDeadlockException()
```

Constructs a new LockDeadlockException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockDeadlockException

```
public LockDeadlockException(String message)
```

Constructs a new LockDeadlockException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [LockTimeoutException.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

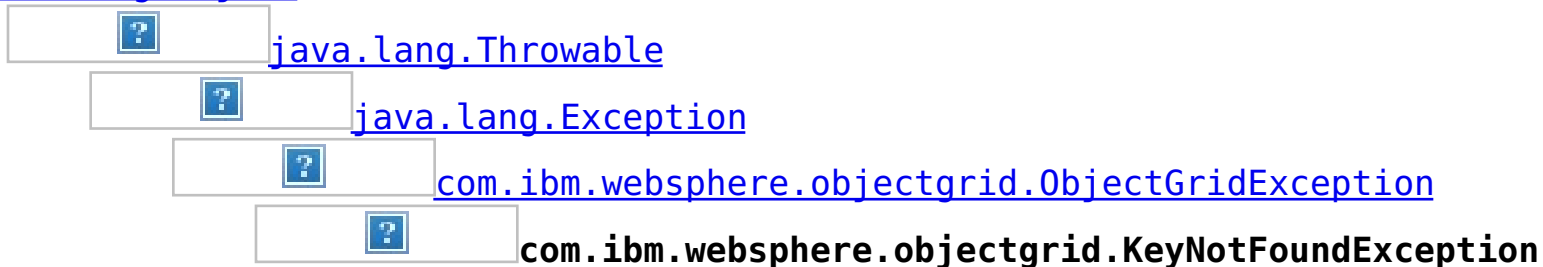
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class KeyNotFoundException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class KeyNotFoundException  
extends ObjectGridException
```

Normally, record not found means a null is returned. However, sometimes on the explicit operation methods like update methods, we can figure that the record isn't there and then we throw this exception.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[KeyNotFoundException](#)()

Constructs a new KeyNotFoundException with null as its detail message.

[KeyNotFoundException](#)([String](#) message)

Constructs a new KeyNotFoundException with the specified detail message.

[KeyNotFoundException](#)([String](#) message, [Throwable](#) cause)

Constructs a new KeyNotFoundException with the specified detail message and cause.

[KeyNotFoundException](#)([Throwable](#) cause)

Constructs a new KeyNotFoundException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

KeyNotFoundException

```
public KeyNotFoundException()
```

Constructs a new KeyNotFoundException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

KeyNotFoundException

```
public KeyNotFoundException(String message)
```

Constructs a new KeyNotFoundException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

KeyNotFoundException

```
public KeyNotFoundException(String message,  
                             Throwable cause)
```

Constructs a new KeyNotFoundException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this KeyNotFoundException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

KeyNotFoundException

```
public KeyNotFoundException(Throwable cause)
```

Constructs a new KeyNotFoundException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for KeyNotFoundExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that

the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid Interface **JavaMap**

All Superinterfaces:

[Map](#)

```
public interface JavaMap  
extends Map
```

This interface is a handle to a named Map. Maps should have homogeneous keys and values. An instance of this JavaMap can only be used by the thread that is currently associated with the Session that was used to get this JavaMap instance. Both Session and JavaMap objects are not allowed to be shared by multiple threads concurrently.

Users can get an instance of JavaMap from an instance of ObjectMap by calling `ObjectMap.getJavaMap()`. There are two main differences between JavaMap and ObjectMap:

- JavaMap extends `java.util.Map`. Therefore, users can cast an instance of JavaMap to `java.util.Map` if they want.
- The methods in JavaMap are defined to throw Exceptions similar to those defined on the `java.util.Map` interface that is `ObjectGridRuntimeException`, which is a subclass of `java.lang.RuntimeException` is used for error conditions. The methods in ObjectMap are defined to throw `ObjectGridExceptions`, which are checked exceptions.

The only methods that are supported from the `java.util.Map` interface are:

- `containsKey(Object)`
- `get(Object)`
- `put(Object, Object)`
- `putAll(Map)`
- `remove(Object)`
- `clear()`

All other methods on the `java.util.Map` interface will throw `java.lang.UnsupportedOperationException`.

Since:

WAS XD 6.0, XC10

See Also:

[ObjectMap](#), [ObjectMap.getJavaMap\(\)](#), [Map](#), [BackingMap.setCopyMode\(CopyMode, Class\)](#), [BackingMap.setLockStrategy\(LockStrategy\)](#)

Nested Class Summary

Nested classes/interfaces inherited from interface `java.util.Map`

[Map.Entry<K, V>](#)

Method Summary

v o i d	<p>clear() Clear all keys from the Map.</p>
v o i d	<p>clearCopyMode() Resets the CopyMode back to the one in the BackingMap.</p>
b o o l e a n	<p>containsKey(Object key) Returns true if this map contains a mapping for the specified key.</p>
b o o l e a n	<p>containsValue(Object value) This method of the java.util.Map interface is not supported.</p>
S e t	<p>entrySet() This method of the java.util.Map interface is not supported.</p>
v o i d	<p>flush() Pushes the current set of changes for the JavaMap instance to the Loader without committing the changes.</p>
O b j e c t	<p>get(Object key) Retrieves the object from the cache at the given key.</p>
L i s t	<p>getAll(List keyList) Gets a list of entries from the map.</p>
L i s t	<p>getAllForUpdate(List keyList) Same as the getAll(List) method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries.</p>
c o m . i b m . w e b s p h e r e . p r o	

j
e
c
t
o
r
.
m
d
.
E
n
t
i
t
y
M
e
t
a
d
a
t
a

[getEntityMetadata\(\)](#)

Retrieve the metadata for the entity associated with this map.

O
b
j
e
c
t

[getForUpdate\(Object key\)](#)

Same as `get(Object)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry.

O
b
j
e
c
t

[getIndex\(String name\)](#)

Returns a reference to the named index that can be used with this Map.

S
t
r
i
n
g

[getName\(\)](#)

Returns the name of the JavaMap as defined by the configuration.

O
b
j
e
c
t

[getNextKey\(long timeout\)](#)

Retrieves a key off the map in first-in-first-out (FIFO) insert order.

V
o
i
d

[insert\(Object key, Object value\)](#)

Performs an explicit insert of a given entry.

V
o
i
d

[invalidate\(Object key, boolean isGlobal\)](#)

Invalidates an entry in the cache based on the key parameter.

V
o
i
d

[invalidateAll\(Collection keyList, boolean isGlobal\)](#)

Invalidate a set of cache entries based on the Collection of keys provided.

b
o
o
l
e

[isEmpty\(\)](#)

This method of the `java.util.Map` interface is not supported.

a n	
S e t	keySet() This method of the java.util.Map interface is not supported.
O b j e c t	put(Object key, Object value) Puts the Object value into the cache at location represented by key.
v o i d	putAll(Map map) Puts each of the Object values into the cache at location represented by the corresponding key contained in the Map.
O b j e c t	remove(Object key) Removes the Object value from the cache represented by key.
v o i d	removeAll(Collection keyList) Batch remove from the Map.
v o i d	setCopyMode(CopyMode copyMode, Class valueInterface) Allows the CopyMode for the Map to be overridden on this map on this session only.
i n t	setTimeToLive(int ttl) Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL.
i n t	size() This method of the java.util.Map interface is not supported.
v o i d	touch(Object key) Updates the last access time in the BackingMap without retrieving the value to the JavaMap.
v o i d	update(Object key, Object value) Performs an explicit update of a given entry.
C o l l e c t i o n	values() This method of the java.util.Map interface is not supported.

Methods inherited from interface java.util.[Map](#)

[equals](#), [hashCode](#)

Method Detail

getName

[String](#) getName()

Returns the name of the JavaMap as defined by the configuration.

Returns:

name of JavaMap

getForUpdate

[Object](#) getForUpdate([Object](#) key)
throws [ObjectGridRuntimeException](#)

Same as get(Object) method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry. See [LockStrategy.PESSIMISTIC](#) for additional information. Whether or not a copy of the object is returned is determined by the [CopyMode](#) setting for this map. See [CopyMode](#) for a description of each possible [CopyMode](#). If the key cannot be found in the map, a null value will be returned. A null value is also returned if the value is null and this map allows null values. To distinguish the two, use the containsKey method.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an an [EntityManager](#) API entity is associated with the [BackingMap](#).

See [ObjectMap.getForUpdate\(Object\)](#) for additional specification details.

Parameters:

key - The entry to fetch

Returns:

the value retrieved for update or null

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[containsKey\(Object\)](#), [get\(Object\)](#), [CopyMode](#), [LockStrategy.PESSIMISTIC](#), [ObjectMap.getForUpdate\(Object\)](#)

getAll

[List](#) getAll([List](#) keyList)
throws [ObjectGridRuntimeException](#)

Gets a list of entries from the map.

If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

A return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

See [ObjectMap.getAll\(List\)](#) for additional specification details.

Parameters:

keyList - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[get\(Object\)](#), [ObjectMap.getAll\(List\)](#)

getAllForUpdate

[List](#) `getAllForUpdate(List keyList)`
throws [ObjectGridRuntimeException](#)

Same as the `getAll(List)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries. See `LockStrategy.PESSIMISTIC` for additional information. If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

A return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

See [ObjectMap.getAllForUpdate\(List\)](#) for additional specification details.

Parameters:

keyList - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[getAll\(List\)](#), [getForUpdate\(Object\)](#), [LockStrategy.PESSIMISTIC](#),
[ObjectMap.getAllForUpdate\(List\)](#)

removeAll

`void removeAll(Collection keyList)`
throws [ObjectGridRuntimeException](#)

Batch remove from the Map. If a key in the list cannot be found, it will be ignored.

See [ObjectMap.removeAll\(Collection\)](#) for additional specification details.

Parameters:

keyList - A list of keys for identifying which entries to remove

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[remove\(Object\)](#), [ObjectMap.removeAll\(Collection\)](#)

invalidate

```
void invalidate(Object key,  
              boolean isGlobal)  
              throws ObjectGridRuntimeException
```

Invalidates an entry in the cache based on the key parameter.

If the key's value has changes pending in the JavaMap, it is the application's responsibility to flush these changes to the Loader before invalidation. If a flush is not performed prior to invoking the invalidate operation, all pending changes for this key will be removed from the JavaMap. If the key cannot be found in the map, it will be ignored.

The isGlobal parameter is used to indicate which cache level is used to invalidate the entries. If isGlobal is true, when the transaction is committed, the key is removed from the BackingMap also. If a subsequent get operation is performed, the BackingMap will be skipped and the Loader will be used to get the data. If isGlobal is false, the entry is only invalidated in the JavaMap (transactional cache). If a subsequent get operation is performed, the BackingMap can be used; and, if it's not in the BackingMap, the Loader will be used to get the data.

A typical use of isGlobal being false is when a large number of records are touched in a transaction and the application wants to evict records that are no longer used in the cache.

See [ObjectMap.invalidate\(Object, boolean\)](#) for additional specification details.

Parameters:

key - Object representing the key to be used for cache entry invalidation
isGlobal - Indicates whether to remove the entry from the BackingMap (true) or just the JavaMap (false).

Throws:

[IllegalArgumentException](#) - if key is null
[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.invalidate\(Object, boolean\)](#)

invalidateAll

```
void invalidateAll(Collection keyList,  
                 boolean isGlobal)  
                 throws ObjectGridRuntimeException
```

Invalidate a set of cache entries based on the Collection of keys provided. If a key in the collection cannot be found, it will be ignored.

See [ObjectMap.invalidateAll\(Collection, boolean\)](#) for additional specification details.

Parameters:

keyList - A Collection of keys representing the entries to be invalidated
isGlobal - Indicates whether to remove the entry from the BackingMap (true) or just the JavaMap (false).

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.
[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[invalidate\(Object, boolean\)](#), [ObjectMap.invalidateAll\(Collection, boolean\)](#)

setTimeToLive

```
int setTimeToLive(int ttl)
```

Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this `JavaMap`, any previous value set by the `BackingMap.setTimeToLive(int)` method is overridden for this `JavaMap`. If this method is never called on the `JavaMap`, the TTL value from the `BackingMap` setting is used by default. If TTL is never set on the `BackingMap`, the cache entry can live "forever".

This method can only be used when the `TTLType` is set to `LAST_ACCESS_TIME` on the `BackingMap`. If this method is called on the `JavaMap` and the `TTLType` is something other than `LAST_ACCESS_TIME`, an `IllegalStateException` is thrown.

Parameters:

`ttl` - is the time-to-live value in seconds. The value must be ≥ 0 . A value of 0 is used to indicate the cache entry can live "forever". Use of the constant `ObjectMap.TTL_FOREVER` is recommended when "forever" is desired.

Returns:

previous time-to-live value in seconds. The constant `ObjectMap.TTL_FOREVER` can be used to determine if the previous TTL was set to "forever".

Throws:

[IllegalArgumentException](#) - if seconds argument is < 0 .

[IllegalStateException](#) - if `BackingMap.getTtlEvictorType()` returns anything other than `TTLType.LAST_ACCESS_TIME`.

See Also:

[BackingMap.setTimeToLive\(int\)](#), [ObjectMap.setTimeToLive\(int\)](#), [TTLType.LAST_ACCESS_TIME](#)

update

```
void update(Object key,  
            Object value)  
    throws ObjectGridRuntimeException
```

Performs an explicit update of a given entry.

A get operation is not required prior to invoking the update method (unlike the `put` method). Also, an update invocation will never insert a new record. If a the map's `LockStrategy` is `LockStrategy.OPTIMISTIC` this method will implicitly get the entry so as to have the version value of the object for when this method was invoked. Whether or not a copy of the object is made when transaction is committed is determined by the `CopyMode` setting for this map. See `CopyMode` for a description of each possible `CopyMode`.

If a key cannot be found in the map during commit, a `TransactionException` will be thrown.

See [ObjectMap.update\(Object, Object\)](#) for additional specification details.

Parameters:

`key` - Identifies the entry to be updated
`value` - The updated value for this entry

Throws:

[IllegalArgumentException](#) - if `key` is `null` or if the map does not allow `null` values and `value` is `null`.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[insert\(Object, Object\)](#), [put\(Object, Object\)](#), [CopyMode](#), [LockStrategy.OPTIMISTIC](#),

insert

```
void insert(Object key,  
           Object value)  
    throws ObjectGridRuntimeException
```

Performs an explicit insert of a given entry.

The key must not exist before executing this method. Also, an insert invocation will never update an existing record. Whether or not a copy of the object is made when a transaction is committed is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode.

If the key is already in the map, a TransactionException will be thrown during commit.

See [ObjectMap.insert\(Object, Object\)](#) for additional specification details.

Parameters:

key - Identifies the entry to be inserted
value - The value for this entry

Throws:

[IllegalArgumentException](#) - if key is null or if the map does not allow null values and value is null.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[put\(Object, Object\)](#), [update\(Object, Object\)](#), [CopyMode](#), [ObjectMap.insert\(Object, Object\)](#)

getIndex

```
Object getIndex(String name)  
    throws com.ibm.websphere.objectgrid.IndexUndefinedException,  
           com.ibm.websphere.objectgrid.IndexNotReadyException,  
           UnsupportedOperationException
```

Returns a reference to the named index that can be used with this Map. This index cannot be shared between threads and works on the same rules as Session. The returned value should be cast to the right index interface such as MapIndex, MapRangeIndex or a custom index interface such as a geo spatial index.

Parameters:

name - The index name

Returns:

A reference to the index, it must be cast to the appropriate index interface.

Throws:

[IndexUndefinedException](#) - if the index is not defined on the BackingMap

[IndexNotReadyException](#) - if the index is a dynamic index and it is not ready

[UnsupportedOperationException](#) - if the map is a distributed map

Since:

WAS XD 6.0.1

flush

```
void flush()  
    throws ObjectGridRuntimeException
```

Pushes the current set of changes for the JavaMap instance to the Loader without committing the changes. The changes are not propagated to the BackingMap either. This is useful for re-priming the Loader's data without committing the current transaction and

starting over.

Throws:

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[Session.flush\(\)](#), [ObjectMap.flush\(\)](#)

size

int `size()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[size](#) in interface [Map](#)

Returns:

the number of key-value mappings in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

isEmpty

boolean `isEmpty()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[isEmpty](#) in interface [Map](#)

Returns:

true if this map contains no key-value mappings.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

containsKey

boolean `containsKey(Object key)`

Returns true if this map contains a mapping for the specified key. ObjectGrid does not support null keys. If you configured the map to support null values, this method can be used to determine whether a key is contained in the map or not.

This API does not hold any locks when using pessimistic locking.

See [ObjectMap.containsKey\(Object\)](#) for additional specification details.

Specified by:

[containsKey](#) in interface [Map](#)

Parameters:

key - key whose presence in this map is to be tested.

Returns:

true if this map contains a mapping for the specified key.

Throws:

[IllegalArgumentException](#) - if null key parameter is passed in

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.containsKey\(Object\)](#)

containsValue

boolean `containsValue(Object value)`

This method of the `java.util.Map` interface is not supported.

Specified by:

[containsValue](#) in interface [Map](#)

Parameters:

value - value whose presence in this map is to be tested.

Returns:

true if this map maps one or more keys to the specified value.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

get

`Object get(Object key)`

Retrieves the object from the cache at the given key.

Whether or not a copy of the object is returned is determined by the `CopyMode` setting for this map. See `CopyMode` for a description of each possible `CopyMode`. If the key cannot be found in the map, a `null` value will be returned. A `null` value is also returned if a value is `null` and this map allows `null` values. To distinguish the two, use the `containsKey` method.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

The return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

See [ObjectMap.get\(Object\)](#) for additional specification details.

Specified by:

[get](#) in interface [Map](#)

Parameters:

key - The entry to fetch

Returns:

the value, `null`, `SerializedValue` OR `Tuple`

Throws:

[IllegalArgumentException](#) - if key is `null`

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[containsKey\(Object\)](#), [getForUpdate\(Object\)](#), [CopyMode](#), [ObjectMap.get\(Object\)](#)

put

`Object put(Object key,
Object value)`

Puts the Object value into the cache at location represented by key.

The values will be pushed down to the `BackingMap/Loader` at commit time and has two behaviors, which can be altered using the [ObjectMap.setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSERTUPDATE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSERT](#) The values are put into the map using the specification of the [ObjectMap.putAll\(LinkedHashMap\)](#).

Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See [CopyMode](#) for a description of each possible copy mode.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an an [EntityManager](#) API entity is associated with the [BackingMap](#).

See [ObjectMap.put\(Object, Object\)](#) for additional specification details.

Specified by:

[put](#) in interface [Map](#)

Parameters:

key - The entry to put into the map

value - The value to put into the map using the key

Returns:

If [ObjectMap.PutMode.INSERTUPDATE](#) is set, return the previous value in this transaction.

If [ObjectMap.PutMode.UPSERT](#) is set, the return value is null.

Throws:

[IllegalArgumentException](#) - if key is null, or if the map does not allow null values and value is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[CopyMode](#), [ObjectMap.put\(Object, Object\)](#)

remove

[Object](#) [remove\(Object key\)](#)

Removes the Object value from the cache represented by key.

This removal will be pushed down to the [BackingMap/Loader](#) at commit time. If the key cannot be found in the map, a null value will be returned.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an an [EntityManager](#) API entity is associated with the [BackingMap](#).

See [ObjectMap.remove\(Object\)](#) for additional specification details.

Specified by:

[remove](#) in interface [Map](#)

Parameters:

key - The entry to remove

Returns:

the current value at invocation time

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.remove\(Object\)](#)

putAll

void **putAll**([Map](#) map)

Puts each of the Object values into the cache at location represented by the corresponding key contained in the Map.

The value will be pushed down to the BackingMap/Loader at commit time and has two behaviors, which can be altered using the [ObjectMap.setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSE](#) The value is put into the map using the specification of the [ObjectMap.upsert\(Object, Object\)](#).

Whether or not a copy of the objects contained in the map is made when transaction is committed is determined by the copy mode setting for this map. See CopyMode for a description of each possible copy mode.

An existing Map object will be passed in to use for obtaining the keys and values to be inserted or updated into the existing Map.

See [ObjectMap.putAll\(Map\)](#) for additional specification details.

Specified by:

[putAll](#) in interface [Map](#)

Parameters:

map - The key/values to be put into the map.

Throws:

[IllegalArgumentException](#) - if map is null or contains a null key or if null values are not allowed and map contains a null value.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[put\(Object, Object\)](#), [ObjectMap.putAll\(Map\)](#)

clear

void **clear**()

Clear all keys from the Map.

This method is an auto-commit call, so a session should not be explicitly begun or committed when calling clear on the Map.

Specified by:

[clear](#) in interface [Map](#)

Throws:

[ObjectGridRuntimeException](#) - if an error occurs during processing

Since:

keySet

[Set](#) `keySet()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[keySet](#) in interface [Map](#)

Returns:

a set view of the keys contained in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

values

[Collection](#) `values()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[values](#) in interface [Map](#)

Returns:

a collection view of the values contained in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

entrySet

[Set](#) `entrySet()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[entrySet](#) in interface [Map](#)

Returns:

a set view of the mappings contained in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

touch

`void touch(Object key)`

Updates the last access time in the `BackingMap` without retrieving the value to the `JavaMap`.

The last access time is updated during commit. If the key does not exist in the `BackingMap`, a `TransactionException` will be returned during commit processing.

See [ObjectMap.touch\(Object\)](#) for additional specification details.

Parameters:

key - key to be touched

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.touch\(Object\)](#)

setCopyMode

```
void setCopyMode(CopyMode copyMode,  
                Class valueInterface)  
    throws ObjectGridRuntimeException
```

Allows the CopyMode for the Map to be overridden on this map on this session only.

This method allows an application to use an optimal CopyMode TRANSACTION by TRANSACTION as its needs dictate. The CopyMode cannot be changed during a transaction. There must be no active transaction when this method is called.

Parameters:

copyMode - must be one of the final static variables defined in CopyMode. See CopyMode class for an explanation of each mode and how the valueInterface is used for CopyMode.COPY_ON_WRITE .

valueInterface - the value interface Class object. Specify null in version 7.1 and later.

Throws:

[IllegalArgumentException](#) - if copyMode is null or COPY_ON_WRITE CopyMode is specified and the required value interface parameter is null

[ObjectGridRuntimeException](#) - if a transaction is active and this map has already been used in the transaction or an error occurs during processing

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\)](#), [CopyMode](#), [ObjectMap.setCopyMode\(CopyMode, Class\)](#)

clearCopyMode

```
void clearCopyMode()  
    throws ObjectGridRuntimeException
```

Resets the CopyMode back to the one in the BackingMap.

This method is used to reverse a previous setCopyMode method call for this JavaMap. This method can only be called when no transaction is active on the associated session.

Throws:

[ObjectGridRuntimeException](#) - if a transaction is active and this map has already been used in the transaction or an error occurs during processing

See Also:

[setCopyMode\(CopyMode, Class\)](#), [ObjectMap.clearCopyMode\(\)](#)

getNextKey

```
Object getNextKey(long timeout)
```

Retrieves a key off the map in first-in-first-out (FIFO) insert order. The entry is locked by the session such that other calls to getNextKey will not return the same key. The key can be used to remove or manipulate the value although leaving the entry will result in the key remaining at the beginning of the queue. This order is optimized for performance and is not guaranteed especially across partitions or in highly concurrent environments.

The return value is a SerializedKey when OutputFormat.RAW is set for the keys. The default key output format for maps that are associated with a KeySerializerPlugin is OutputFormat.RAW. The SerializedKey allows access to the value in its serialized form, or

its native Java Object form.

The return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

See [ObjectMap.getNextKey\(long\)](#) for additional specification details.

Parameters:

timeout - The period of time to wait for an entry to become available on the queue.

Returns:

The next available key in the map.

See Also:

[ObjectMap.getNextKey\(long\)](#)

getEntityMetadata

com.ibm.websphere.projector.md.EntityMetadata **getEntityMetadata()**

Retrieve the metadata for the entity associated with this map.

Returns:

the EntityMetadata if an entity is associated with this map or null if there is no entity associated with this map.

Since:

WAS XD 6.1

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
ew	ge	ss	ed	ted			
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All		
			Classes				

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid

Interface IObjectGridException

All Known Implementing Classes:

[CacheEntryException](#), [CannotGenerateCredentialException](#),
[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#),
 com.ibm.websphere.objectgrid.ClientServerTransactionCallbackException,
[ConnectException](#), [ContinuousQueryAttributePathException](#), [ContinuousQueryException](#),
[ContinuousQueryGetValueException](#), [ContinuousQueryIncompatibleDuplicateException](#),
[DuplicateKeyException](#), [ExpiredCredentialException](#), [InvalidCredentialException](#),
[KeyNotFoundException](#), [LoaderException](#), [LockDeadlockException](#), [LockException](#),
[LockInternalFailureException](#), [LockTimeoutException](#), [NoActiveTransactionException](#),
[ObjectGridConfigurationException](#), [ObjectGridException](#), [ObjectGridRuntimeException](#),
[ObjectGridSecurityException](#), [ReadOnlyException](#),
[ReplicationVotedToRollbackTransactionException](#), [SessionNotReentrantException](#),
[TransactionAffinityException](#), [TransactionAlreadyActiveException](#),
[TransactionCallbackException](#), [TransactionException](#), [TransactionQuiesceException](#),
[TransactionTimeoutException](#), [UnavailableServiceException](#), [UndefinedMapException](#)

public interface **IObjectGridException**

This interface is used to ensure JDK 1.4 Throwable chaining behavior for all exceptions thrown by ObjectGrid even when an earlier JDK is used (e.g. JDK 1.3.1).

Since:

WAS XD 6.0.1, XC10

Method Summary

I h r o w a b l e	<p>getCause() Provides JDK 1.4 Throwable.getCause() behavior.</p>
I h r o w a b l e	<p>initCause(Throwable cause) Provides JDK 1.4 Throwable.initCause() behavior.</p>

Method Detail

getCause

[Throwable](#) `getCause()`

Provides JDK 1.4 `Throwable.getCause()` behavior.

Returns the cause of this throwable or `null` if the cause is nonexistent or unknown. (The cause is the throwable that caused this throwable to get thrown.)

This implementation returns the cause that was supplied via one of the constructors requiring a `Throwable`, or that was set after creation with the `initCause(Throwable)` method. While it is typically unnecessary to override this method, a subclass can override it to return a cause set by some other means. This is appropriate for a "legacy chained throwable" that predates the addition of chained exceptions to `Throwable`. Note that it is *not* necessary to override any of the `PrintStackTrace` methods, all of which invoke the `getCause` method to determine the cause of a throwable.

Returns:

the cause of this throwable or `null` if the cause is nonexistent or unknown.

See Also:

[initCause\(Throwable\)](#)

initCause

[Throwable](#) `initCause(Throwable cause)`
throws [IllegalArgumentException](#),
[IllegalStateException](#)

Provides JDK 1.4 `Throwable.initCause()` behavior.

Initializes the *cause* of this throwable to the specified value. (The cause is the throwable that caused this throwable to get thrown.)

This method can be called at most once. It is generally called from within the constructor, or immediately after creating the throwable. If this throwable was created with `Throwable(Throwable)` or `Throwable(String,Throwable)`, this method cannot be called even once.

Parameters:

cause - the cause (which is saved for later retrieval by the `getCause()` method). (A `null` value is permitted, and indicates that the cause is nonexistent or unknown.)

Returns:

a reference to this `Throwable` instance.

Throws:

[IllegalArgumentException](#) - if cause is this throwable. (A throwable cannot be its own cause.)

[IllegalStateException](#) - if this throwable was created with `Throwable(Throwable)` or `Throwable(String,Throwable)`, or this method has already been called on this throwable.

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

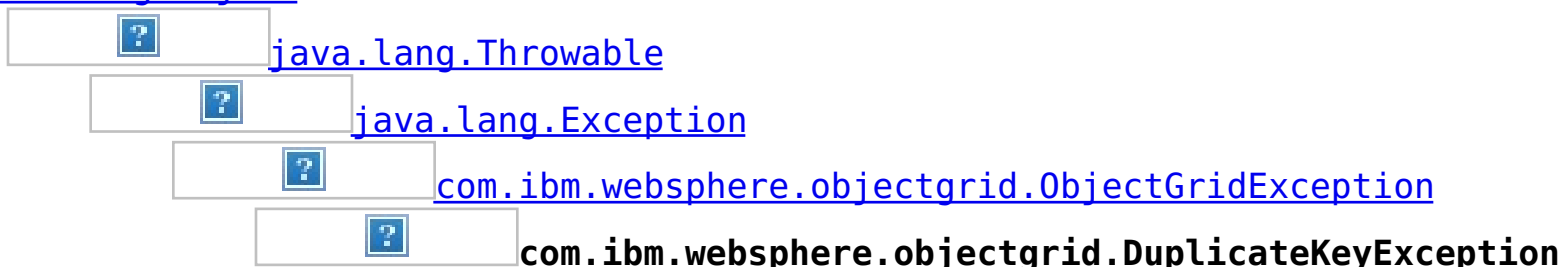
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class DuplicateKeyException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class DuplicateKeyException
extends ObjectGridException
```

A DuplicateKeyException exception is thrown if a key cannot be inserted into a BackingMap because an object with the same key already exists.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[DuplicateKeyException](#)()

Constructs a new DuplicateKeyException with null as its detail message.

[DuplicateKeyException](#)(String message)

Constructs a new DuplicateKeyException with the specified detail message.

[DuplicateKeyException](#)(String message, Throwable cause)

Constructs a new DuplicateKeyException with the specified detail message and cause.

[DuplicateKeyException](#)(Throwable cause)

Constructs a new DuplicateKeyException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

DuplicateKeyException

```
public DuplicateKeyException()
```

Constructs a new DuplicateKeyException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

DuplicateKeyException

```
public DuplicateKeyException(String message)
```

Constructs a new DuplicateKeyException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

DuplicateKeyException

```
public DuplicateKeyException(String message,
                             Throwable cause)
```

Constructs a new DuplicateKeyException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this DuplicateKeyException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

DuplicateKeyException

```
public DuplicateKeyException(Throwable cause)
```

Constructs a new DuplicateKeyException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for DuplicateKeyExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid
Class CopyMode

[java.lang.Object](#)



```
public final class CopyMode
extends Object
```

This class is used to define the "copy" mode when the setCopyMode method of the BackingMap interface is used. The application is expected to pass one of the final static variables that are defined in this class to the setCopyMode method.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\)](#), [ObjectTransformer.copyValue\(Object\)](#)

Field Summary

s t a t i c C O P Y M O D E	<p>COPY_ON_READ</p> <p>The COPY_ON_READ mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when a transaction is committed.</p>
s t a t i c C O P Y M O D E	<p>COPY_ON_READ_AND_COMMIT</p> <p>The COPY_ON_READ_AND_COMMIT mode is the default mode.</p>
s t a t i c C O P Y M O D E	<p>COPY_ON_WRITE</p> <p>The COPY_ON_WRITE mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when ObjectMap.get is called for the first time by a transaction for a given key.</p>

M
o
d
e

s
t
a
t
i
c
C
o
p
y
M
o
d
e

[COPY_TO_BYTES](#)

The COPY_TO_BYTES mode is similar to the COPY_ON_READ_AND_COMMIT mode in that it ensures that an application never has a reference to the value object that is in the BackingMap.

s
t
a
t
i
c
C
o
p
y
M
o
d
e

[COPY_TO_BYTES_RAW](#)

When set, all ObjectMap APIs that return a SerializedValue rather than the original Java Object, allowing access to the serialized form of the data, preventing inflation of object into Java Object form.

s
t
a
t
i
c
C
o
p
y
M
o
d
e

[NO_COPY](#)

The NO_COPY mode allows an application to promise that it will never modify a value object obtained using an ObjectMap.get method in exchange for performance improvements.

Method Summary

b
o
o
l
e
a
n

[isBytes\(\)](#)

Is the copy mode one of the copy modes that indicate copy to bytes?

S
t
r
i
n
g

[toString\(\)](#)

Returns a string representation of the CopyMode.

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

COPY_ON_READ_AND_COMMIT

```
public static final CopyMode COPY_ON_READ_AND_COMMIT
```

The COPY_ON_READ_AND_COMMIT mode is the default mode. This mode ensures that an application never has a reference to the value object that is in the BackingMap, and instead the application is always working with a copy of the value that is in the BackingMap. The copy ensures the application can never inadvertently corrupt the data that is cached in the BackingMap. When an application transaction calls an `ObjectMap.get` method for a given key, and it is the first access of the `ObjectMap` entry for that key, a copy of the value is returned. When the transaction is committed, any changes the application committed are copied to the BackingMap to ensure that the application does not have reference to the committed value in the BackingMap.

COPY_ON_READ

```
public static final CopyMode COPY_ON_READ
```

The COPY_ON_READ mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when a transaction is committed. To preserve integrity of BackingMap data, the application promises to destroy every reference it has to an entry once the transaction is committed. This mode results in a `ObjectMap.get` method returning a copy of the value rather than a reference to the value to ensure that changes made by the application to the value does not affect the BackingMap value until the transaction is committed. However, when the transaction does commit, a copy of changes is not made. Instead, the reference to the copy that was returned by `ObjectMap.get` is stored in the BackingMap. This is the reason the application must agree to destroy all map entry references once the transaction is committed. If application fails to keep its promise, the application could cause the data cached in BackingMap to become corrupted. If an application is using this mode and it is having problems, then switch to the COPY_ON_READ_AND_COMMIT mode to see if the problem still exists. If the problem goes away, then more than likely the application is failing to destroy all of its references after the transaction has committed.

COPY_ON_WRITE

```
public static final CopyMode COPY_ON_WRITE
```

The COPY_ON_WRITE mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when `ObjectMap.get` is called for the first time by a transaction for a given key. Instead, the `ObjectMap.get` method returns a proxy to the value rather than a direct reference to the value object itself. The proxy ensures that a copy of the value is not made unless the application calls a set method on the value interface that is passed on the `BackingMap.setCopyMode(CopyMode, Class)` method. Thus, the proxy provides a "copy on write" implementation. When a transaction commits, the BackingMap examines the proxy to determine if any copy was made as a result of a set method being called. If a copy was made, then the reference to that copy is stored in the BackingMap. The big advantage of this mode is a value is never copied on read or at commit when the transaction never calls a set method to mutate the value.

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\)](#)

NO_COPY

```
public static final CopyMode NO_COPY
```

The NO_COPY mode allows an application to promise that it will never modify a value

object obtained using an `ObjectMap.get` method in exchange for performance improvements. If this mode is used, no copy of the value is ever made. If the application breaks its promise and does modify values, then data in the `BackingMap` will be corrupted. This mode is primarily useful for read only maps where data is never modified by the application. If the application is using this mode and it is having problems, then switch to `COPY_ON_READ_AND_COMMIT` mode to see if the problem still exists. If the problem goes away, then more than likely the application is not keeping its promise and is modifying the value returned by `ObjectMap.get` method (either during transaction or after transaction has committed).

COPY_TO_BYTES

```
public static final CopyMode COPY_TO_BYTES
```

The `COPY_TO_BYTES` mode is similar to the `COPY_ON_READ_AND_COMMIT` mode in that it ensures that an application never has a reference to the value object that is in the `BackingMap`. The value that the application works with is a newly inflated version of the serialized version that is in the `BackingMap`. The copy ensures the application can never inadvertently corrupt the data that is cached in the `BackingMap` since a byte form of the value is what is stored in the `BackingMap` instead of the Object form.

A copy of the value is returned when an application transaction calls an `ObjectMap.get` method for a given key, and it is the first time that the `ObjectMap` entry is accessed for that key. When the transaction is committed, any changes the application committed are copied to bytes in the `BackingMap` to ensure that the application does not have reference to the committed value in the `BackingMap`.

Since:
7.0

COPY_TO_BYTES_RAW

```
public static final CopyMode COPY_TO_BYTES_RAW
```

When set, all `ObjectMap` APIs that return a `SerializedValue` rather than the original Java Object, allowing access to the serialized form of the data, preventing inflation of object into Java Object form.

Since:
7.1.1
See Also:
`ValueDataSerializer`

Method Detail

toString

```
public String toString()
```

Returns a string representation of the `CopyMode`.

Overrides:
[toString](#) in class [Object](#)

Returns:
a string representation of the `CopyMode`.

isBytes

public boolean **isBytes()**

Is the copy mode one of the copy modes that indicate copy to bytes?

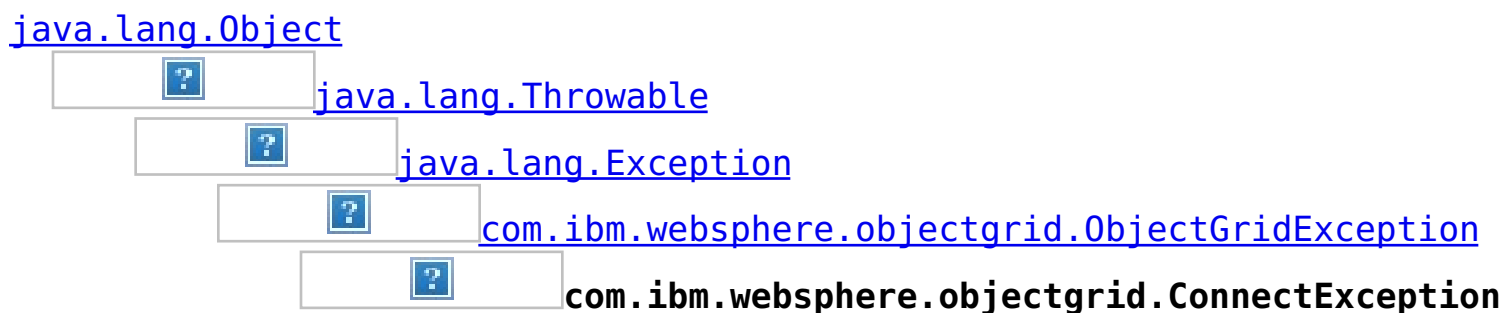
Returns:

boolean indicating if copy mode is one of COPY_TO_BYTES or COPY_TO_BYTES_RAW.

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid
Class ConnectException



All Implemented Interfaces:
[IObjectGridException](#), [Serializable](#)

```
public class ConnectException
extends ObjectGridException
```

This exception is used to indicate that the client was unable to connect to the server

Since:
 WAS XD 6.0.1, XC10

See Also:
[Serialized Form](#)

Field Summary	
s t a t i c i n t	<p>BAD_CONFIGURATION This failure code indicates the provided configuration was corrupt.</p>
s t a t i c i n t	<p>CONNECTION_REFUSED This failure code indicates that the server may not be available.</p>
s t a t i c i n t	<p>FAILED_SECURITY This failure code indicates the a failure to authenticate.</p>
s	

t a t i c i n t	<p>SERVER_DEFINITION_NOT_FOUND This failure code indicates the definition of cluster cannot be accessed.</p>
s t a t i c i n t	<p>UNKNOWN This failure code indicates the reason for the connect failure is unknown.</p>

Constructor Summary

<p>ConnectException() Constructs a new <code>ConnectException</code> with <code>null</code> as its detail message.</p>
<p>ConnectException(String message) Constructs a new <code>ConnectException</code> with the specified detail message.</p>
<p>ConnectException(String message, int failureCode) Constructs a new <code>ConnectException</code> with the specified detail message.</p>
<p>ConnectException(String message, Throwable cause) Constructs a new <code>ConnectException</code> with the specified detail message and cause.</p>
<p>ConnectException(String message, Throwable cause, int failureCode) Constructs a new <code>ConnectException</code> with the specified detail message and cause.</p>
<p>ConnectException(Throwable cause) Constructs a new <code>ConnectException</code> with a specified cause.</p>

Method Summary

i n t	<p>getFailureCode() Returns the failure code that was set by one of the constructors that accepts a failure code, or <code>UNKNOWN</code> if one of the other constructors was called.</p>
-------------	--

<p>Methods inherited from class com.ibm.websphere.objectgrid.ObjectGridException getCause, initCause</p>
--

<p>Methods inherited from class java.lang.Throwable fillInStackTrace, getLocalizedMessage, getMessage, getStackTrace, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString</p>
--

<p>Methods inherited from class java.lang.Object clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait</p>

Field Detail

BAD_CONFIGURATION

```
public static final int BAD_CONFIGURATION
```

This failure code indicates the provided configuration was corrupt.

See Also:

[Constant Field Values](#)

UNKNOWN

```
public static final int UNKNOWN
```

This failure code indicates the reason for the connect failure is unknown.

See Also:

[Constant Field Values](#)

FAILED_SECURITY

```
public static final int FAILED_SECURITY
```

This failure code indicates the a failure to authenticate.

See Also:

[Constant Field Values](#)

CONNECTION_REFUSED

```
public static final int CONNECTION_REFUSED
```

This failure code indicates that the server may not be available.

See Also:

[Constant Field Values](#)

SERVER_DEFINITION_NOT_FOUND

```
public static final int SERVER_DEFINITION_NOT_FOUND
```

This failure code indicates the definition of cluster cannot be accessed.

See Also:

[Constant Field Values](#)

Constructor Detail

ConnectException

```
public ConnectException()
```

Constructs a new `ConnectException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method. In addition the failure code is initialized to `UNKNOWN`.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [UNKNOWN](#)

ConnectException

```
public ConnectException(String message)
```

Constructs a new `ConnectException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method. In addition the failure code is initialized to `UNKNOWN`.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [UNKNOWN](#)

ConnectException

```
public ConnectException(String message,  
                        int failureCode)
```

Constructs a new `ConnectException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

`failureCode` - the failure code which should be one of the constants of this exception class.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [getFailureCode\(\)](#)

ConnectException

```
public ConnectException(Throwable cause)
```

Constructs a new `ConnectException` with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ConnectExceptions` that are little more than wrappers for other throwables. The failure code is initialized to `UNKNOWN`.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#), [UNKNOWN](#)

ConnectException

```
public ConnectException(String message,  
                        Throwable cause)
```

Constructs a new `ConnectException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ConnectException`'s detail message. The failure code is initialized to `UNKNOWN`.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

ConnectException

```
public ConnectException(String message,  
                       Throwable cause,  
                       int failureCode)
```

Constructs a new ConnectException with the specified detail message and cause.

Note that the detail message associated with *cause* is *not* automatically incorporated in this ConnectException's detail message. The failure code is initialized to UNKNOWN.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A `null` value is permitted, and indicates that the cause is nonexistent or unknown).

failureCode - the failure code which should be one of the constants of this exception class.

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#), [getFailureCode\(\)](#)

Method Detail

getFailureCode

```
public int getFailureCode()
```

Returns the failure code that was set by one of the constructors that accepts a failure code, or UNKNOWN if one of the other constructors was called.

Returns:

the failure code. One of the constants of this exception class.

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

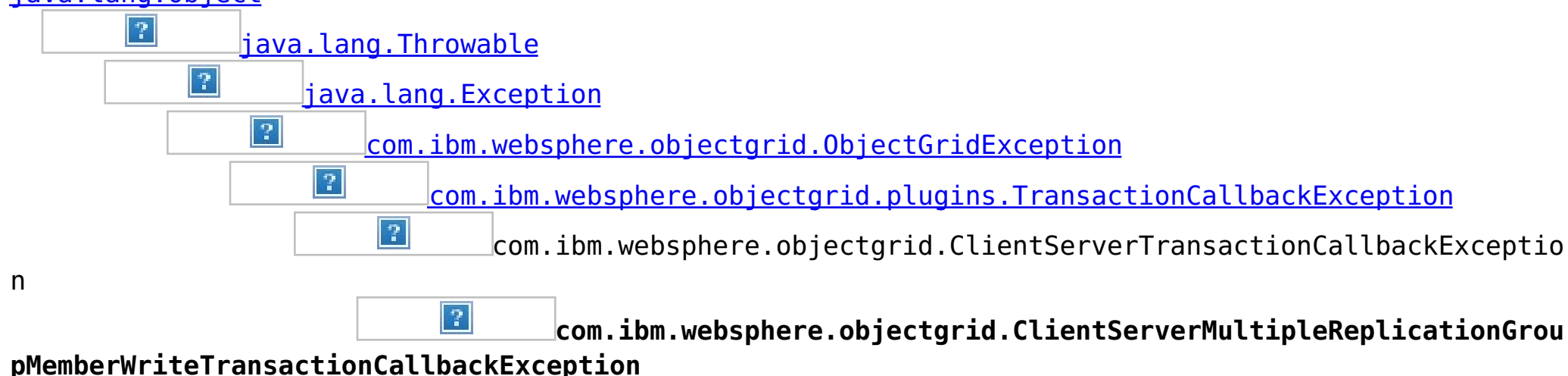
**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException
extends com.ibm.websphere.objectgrid.ClientServerTransactionCallbackException
```

This exception is thrown when a method call to the Client/Server TransactionCallback detects the user is attempting to perform a write against multiple maps in different Map Sets, Partition Sets or Replication groups. This is not allowed.

Since:

WAS XD 6.0.1, XC10

See Also:

[TransactionCallback](#), [Serialized Form](#)

Constructor Summary

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)()

Constructs a new `ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException` with `null` as its detail message.

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)([String](#) message)

Constructs a new `ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException` with the specified detail message.

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)([String](#) message, [Throwable](#) cause)

Constructs a new `ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException` with the specified detail message and cause.

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)([Throwable](#) cause)

Constructs a new

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with a specified cause.

Method Summary

Methods inherited from class `com.ibm.websphere.objectgrid.ObjectGridException`

[getCause](#), [initCause](#)

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException()
```

Constructs a new ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with null as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException(String message)
```

Constructs a new ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException(String message, Throwable cause)
```

Constructs a new

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the getMessage method).

cause - the cause (which is saved for later retrieval by the getCause method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException(Throwable cause)
```

Constructs a new ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the getCause() method. A null value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface ClientReplicableMap

All Known Subinterfaces:

[BackingMap](#)

Deprecated. *The client replicated map function is deprecated in version 8.6. Use the [ContinuousQueryManager](#) function.*

```
public interface ClientReplicableMap
```

This interface represents a replicable client map. A replicable client map can be a continuous replica or a snapshot replica of the server map.

If the client is a continuous replica of the server map, the data in the server will be replicated to the client continuously in an asynchronous manner.

If the client is a snapshot replica of the server map, a snapshot on the data in the server will be taken and the snapshot will be replicated to the client in an asynchronous manner. A snapshot replication is a one-time replication.

A ReplicationMapListener can be used to listen for the data changes as well as the replication lifecycle events.

Since:

WAS XD 6.1, XC10

See Also:

[ReplicationMapListener](#)

Nested Class Summary

s t a t i c c l a s s	ClientReplicableMap.Mode Deprecated. Client Replication mode
---	--

Field Summary

s
t
a
t
i
c

C
l
i
e

n
t
R
e
p
l
i
c
a
b
l
e
M
a
p
.
M
o
d
e

CONTINUOUS_REPLICATION

Deprecated. Full replication mode.

s
t
a
t
i
c
l
i
e
n
t
R
e
p
l
i
c
a
b
l
e
M
a
p
.
M
o
d
e

NONE

Deprecated. No replication mode, aka normal mode.

s
t
a
t
i
c
l
i
e
n
t
R
e
p
l
i
c
a
b
l
e
M
a
p
.
M
o
d
e

SNAPSHOT_REPLICATION

Deprecated. Snapshot replication mode.

Method Summary

[disableClientReplication\(\)](#)

Deprecated. Disables the replication for this client.

[enableClientReplication\(ClientReplicableMap.Mode mode, int\[\] partitions, ReplicationMapListener listener\)](#)

Deprecated. Make the client map a replica of the server side map.

[getReplicationMode\(\)](#)

Deprecated. Returns the current replication mode

Field Detail

NONE

static final [ClientReplicableMap.Mode](#) NONE

Deprecated.

No replication mode, aka normal mode.

CONTINUOUS_REPLICATION

static final [ClientReplicableMap.Mode](#) CONTINUOUS_REPLICATION

Deprecated.

Full replication mode. Data in the server map will be replicated to the client continuously.

SNAPSHOT_REPLICATION

static final [ClientReplicableMap.Mode](#) SNAPSHOT_REPLICATION

Deprecated.

Snapshot replication mode. A snapshot on the data in the server will be taken and the snapshot will be replicated to the client. A snapshot replication is a one-time replication.

Method Detail

enableClientReplication

```
void enableClientReplication(ClientReplicableMap.Mode mode,  
                             int[] partitions,  
                             ReplicationMapListener listener)  
    throws ObjectGridException
```

Deprecated.

Make the client map a replica of the server side map.

When security is enabled, this method requires a `ServerMapPermission` with action "replicate". Refer to `ServerMapPermission` for more permission details.

Required Client Permission: `ServerMapPermission.REPLICATE`

Parameters:

mode - The replication mode.

partitions - The array of partition IDs represent which partitions the data should be replicated from. If the value is null or an empty array, it indicates the data should be replicated from all partitions.

listener - a listener to receive client replication events

Throws:

[IllegalArgumentException](#) - if mode is not [CONTINUOUS_REPLICATION](#) or [SNAPSHOT_REPLICATION](#) or the map isn't currently in the mode specified or is not in [NONE](#) mode

[IllegalStateException](#) - if this method is invoked on a map other than a client map

[ObjectGridException](#) - if an error occurs during processing this request

See Also:

[ReplicationMapListener](#), [CONTINUOUS_REPLICATION](#), [SNAPSHOT_REPLICATION](#), [getReplicationMode\(\)](#), [BackingMap.CLIENT](#)

getReplicationMode

```
ClientReplicableMap.Mode getReplicationMode()
```

Deprecated.

Returns the current replication mode

Returns:

the replication mode

See Also:

[NONE](#), [CONTINUOUS_REPLICATION](#), [SNAPSHOT_REPLICATION](#)

disableClientReplication

```
void disableClientReplication()  
    throws ObjectGridException
```

Deprecated.

Disables the replication for this client. If it is not in a replication mode, this method will be a no-op.

When security is enabled, this method requires a `ServerMapPermission` with action "replicate". Refer to `ServerMapPermission` for more permission details.

Throws:

[IllegalStateException](#) - if this method is invoked on a map other than a client map
[ObjectGridException](#) - if an error occurs during processing this request

See Also:

[BackingMap.CLIENT](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>	
PREV CLASS	NEXT CLASS	FRAMES		NO FRAMES	All			
		Classes						
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD								

[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class ClientReplicableMap.Mode

[java.lang.Object](#)



Enclosing interface:

[ClientReplicableMap](#)

```
public static final class ClientReplicableMap.Mode  
extends Object
```

Client Replication mode

Method Summary

S t r i n g	toString()
----------------------------	----------------------------

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Method Detail

toString

```
public String toString()
```

Overrides:

[toString](#) in class [Object](#)

com.ibm.websphere.objectgrid

Interface ClientClusterContext

public interface **ClientClusterContext**

This interface is a context to represent which cluster/domain the client connected to using one of the `ObjectGridManager.connect` methods. An instance of this interface is used to retrieve client `ObjectGrid` instances and for performing admin operations against an `ObjectGrid` cluster/domain or its servers.

Since:

WAS XD 6.0.1, XC10

See Also:

[ObjectGridManager](#)

Method Summary

C l i e n t P r o p e r t i e s	getClientProperties (String objectGridName) Retrieve the <code>ClientProperties</code> object for this <code>ClientClusterContext</code> for the specified <code>ObjectGrid</code> name.
S t r i n g	getClusterName () Gets the name of the domain to which the client is connected
v o i d	setClientProperties (String objectGridName, URL url) Sets the <code>ClientProperties</code> properties for the selected <code>ObjectGrid</code> using the specified client properties file.

Method Detail

getClusterName

[String](#) `getClusterName()`

Gets the name of the domain to which the client is connected

Returns:

the name of the domain this context is connected to

getClientProperties

[ClientProperties](#) `getClientProperties(String objectGridName)`

Retrieve the ClientProperties object for this ClientClusterContext for the specified ObjectGrid name. A ClientProperties is scoped to this ClientClusterContext and a single ObjectGrid.

Parameters:

objectGridName - the name of ObjectGrid

Returns:

the ClientProperties instance for this ObjectGrid.

Since:

WAS XD 6.1.0.3

setClientProperties

`void setClientProperties(String objectGridName,
URL url)`

Sets the ClientProperties properties for the selected ObjectGrid using the specified client properties file.

To further adjust the client properties, call the [getClientProperties\(String\)](#) method.

Parameters:

objectGridName - the name of ObjectGrid to apply the ClientProperties to.

url - the URL where the client properties file can be located.

Since:

WAS XD 6.1.0.3

Overvi	Packa	Cla	TreeSerializ	Depreca	IndexHelp	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
ew	ge	ss	ed	ted		
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All	
			Classes			

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid

Interface CatalogDomainManager

public interface **CatalogDomainManager**

Provides access to catalog domain configuration information for the current environment.

When running in a WebSphere Application Server profile augmented with WebSphere eXtreme Scale, the CatalogDomainManager returns the catalog service domain configuration information that is configured in the administration console.

Since:

8.5, XC10

See Also:

[ObjectGridManager.getCatalogDomainManager\(\)](#)

Method Summary

C
a
t
a
l
o
g
D
o
m
a
i
n
I
n
f
o

[getDefaultDomainInfo\(\)](#)

Retrieve the default, configured CatalogDomainInfo.

C
a
t
a
l
o
g
D
o
m
a
i
n
I
n
f
o

[getDomainInfo\(String domainId\)](#)

Retrieve the specified CatalogDomainInfo for the specified domainId.

C
o
l
l

e
c
t
i
o
n
<
C
e
t
e
r
i
n
a
r
y
>

[getDomainInfos\(\)](#)

Retrieve all configured CatalogDomainInfo objects.

Method Detail

getDefaultDomainInfo

[CatalogDomainInfo](#) `getDefaultDomainInfo()`

Retrieve the default, configured CatalogDomainInfo.

Returns:

the default CatalogDomainInfo, or null if not available.

getDomainInfo

[CatalogDomainInfo](#) `getDomainInfo(String domainId)`

Retrieve the specified CatalogDomainInfo for the specified domainId.

Parameters:

domainId - the domain identifier.

Returns:

the CatalogDomainInfo if found, or null.

getDomainInfos

[Collection](#)<[CatalogDomainInfo](#)> `getDomainInfos()`

Retrieve all configured CatalogDomainInfo objects.

Returns:

a collection of CatalogDomainInfo

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid
Interface CatalogDomainInfo

public interface **CatalogDomainInfo**

Identifies the configuration attributes of a catalog service domain.

Since:

8.5, XC10

See Also:

[CatalogDomainManager](#)

Method Summary

S t r i n g	<p>getClientCatalogServerEndpoints() Retrieve the catalog server endpoints used to connect a client to the remote catalog service domain.</p>
C l i e n t S e c u r i t V C o n f i g u r e t i o n	<p>getClientSecurityConfiguration() Retrieve the ClientSecurityConfiguration for the domain.</p>
S t r i n g	<p>getDomainId() Retrieve the identifier of the domain as specified by the catalog service domain.</p>

Method Detail

getDomainId

[String](#) `getDomainId()`

Retrieve the identifier of the domain as specified by the catalog service domain.

Note: This is different than the name of the domain which is specified when starting the catalog services.

Returns:
the identifier of the domain.

getClientCatalogServerEndpoints

[String](#) `getClientCatalogServerEndpoints()`

Retrieve the catalog server endpoints used to connect a client to the remote catalog service domain.

The catalog service endpoints are used with the [ObjectGridManager.connect\(String, com.ibm.websphere.objectgrid.security.config.ClientSecurityConfiguration, java.net.URL\)](#) method to connect to a catalog service domain.

Returns:
the catalog service endpoints.

getClientSecurityConfiguration

[ClientSecurityConfiguration](#) `getClientSecurityConfiguration()`

Retrieve the ClientSecurityConfiguration for the domain.

The ClientSecurityConfiguration are used with the [ObjectGridManager.connect\(String, com.ibm.websphere.objectgrid.security.config.ClientSecurityConfiguration, java.net.URL\)](#) method to connect to a catalog service domain.

Returns:
the ClientSecurityConfiguration or null if security is not configured.

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface BackingMap

All Superinterfaces:

[ClientReplicableMap](#)

```
public interface BackingMap  
extends ClientReplicableMap
```

This is the public interface to the BackingMap. It is returned when a new Map is defined on the ObjectGrid. It allows the Map to be customized with various plug-ins or by setting properties. The defaults are:

- No external Evictor, but an internal time-based evictor is provided by default
- No Loader
- No EventListeners
- No MapIndexPlugins
- An internal ObjectTransformer
- An internal OptimisticCallback
- Key is not copied
- A value CopyMode Of CopyMode.COPY_ON_READ_AND_COMMIT
- A LockStrategy of LockStrategy.OPTIMISTIC
- A default lock timeout
- null values are supported
- A default number of buckets
- A default number of lock buckets
- Synchronous preload
- Read/write map by default
- A TimeToLive of 0 (indicating unlimited time)
- A TtlEvictor type of TTLType.NONE
- Write-behind updates is disabled
- Time-based database updates are disabled
- Eviction triggers are not set

Since:

WAS XD 6.0, XC10

Nested Class Summary

Nested classes/interfaces inherited from interface
com.ibm.websphere.objectgrid.[ClientReplicableMap](#)

[ClientReplicableMap.Mode](#)

Field Summary

s t a t i	CLIENT
-----------------------	------------------------

c i n t	Constant used to indicate this map is a client to a server map
s t a t i c i n t	DEFAULT_LOCK_TIMEOUT Default lock timeout used if setLockTimeout(int) is not invoked.
s t a t i c i n t	DEFAULT_NUMBER_OF_BUCKETS Deprecated. <i>Deprecated in 8.6. Buckets are no longer required. Use the isNearCacheEnabled() flag to disable the near cache in the ObjectGrid configuration XML file.</i>
s t a t i c i n t	DEFAULT_NUMBER_OF_LOCK_BUCKETS Default number of lock buckets used if setNumberOfLockBuckets(int) is not invoked.
s t a t i c S t r i n g	EVICTIONTRIGGER_MEMORY_USAGE_THRESHOLD The eviction trigger string constant to enable memory based eviction using memory usage threshold provided by the java.lang.management.MemoryPoolMXBean.
s t a t i c i n t	LOCAL Constant used to indicate this map is not a distributed map.
s t a t i c i n t	SERVER Constant used to indicate this map is a server map.

Fields inherited from interface com.ibm.websphere.objectgrid.[ClientReplicableMap](#)
[CONTINUOUS_REPLICATION](#), [NONE](#), [SNAPSHOT_REPLICATION](#)



Method Summary

addMapEventListener(com.ibm.websphere.objectgrid.plugins.EventListener eventListener)
Adds an EventListener to this BackingMap.

addMapEventListener(MapEventListener eventListener)
Deprecated. *This method is deprecated in version 7.1.1, use the [addMapEventListener\(EventListener\)](#) method.*

addMapIndexPlugin(com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index)
Adds an MapIndexPlugin to this Map.

createDynamicIndex(com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index, com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)
Creates a dynamic index on the BackingMap.

createDynamicIndex(String name, boolean isRangeIndex, String attributeName, com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)
Creates a dynamic index on the BackingMap.

getCopyKey()
Gets whether keys are copied for this BackingMap.

getCopyMode()
Gets the CopyMode being used by this BackingMap.

getEntityMetadata()
Retreive the metadata for the entity associated with this backing map.

m
d
.
E
n
t
i
t
y
M
e
t
a
d
a
t
a

S
t
r
i
n
g

[getEvictionTriggers\(\)](#)

Returns the types of additional eviction triggers.

E
v
i
c
t
o
r

[getEvictor\(\)](#)

Gets the Evictor being used by this BackingMap.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

[getKeyOutputFormat\(\)](#)

Retrieves the data format for all data access APIs that return cache keys.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g
i
n
s
.
L
o
a
d
e
r

[getLoader\(\)](#)

Gets the Loader being used by this BackingMap.

L
o
c
k
S
t
r
a
t
e
g
y

[getLockStrategy\(\)](#)

Gets the LockStrategy object being used by this BackingMap.

i
n
t

[getLockTimeout\(\)](#)

Gets the lock timeout value used by the lock manager for this BackingMap.

L
i
s
t

[getMapEventListeners\(\)](#)

Gets the current list of EventListeners.

L
i
s
t

[getMapIndexPlugins\(\)](#)

Returns the current list of MapIndexPlugin objects for this BackingMap.

S
t
r
i
n
g

[getMapSetName\(\)](#)

Retrieves the name of the MapSet that this BackingMap is currently associated with.

n
g

i
n
t [getMapType\(\)](#)
Returns the type of BackingMap.

S
t
r
i
n
g

[getName\(\)](#)
Gets the name of the BackingMap.

b
o
o
l
e
a
n

[getNullValuesSupported\(\)](#)
Gets whether this BackingMap supports null values or not.

i
n
t

[getNumberOfBuckets\(\)](#)
Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

i
n
t

[getNumberOfLockBuckets\(\)](#)
Gets the number of lock buckets defined for the hash map used by lock manager for this backing map.

O
b
j
e
c
t
G
r
i
d

[getObjectGrid\(\)](#)
Gets the ObjectGrid that owns this BackingMap.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g

[getObjectTransformer\(\)](#)
Gets the ObjectTransformer object being used by this BackingMap and/or Loader.

i
n
s
.
O
b
j
e
c
t
T
r
a
n
s
f
o
r
m
e
r

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g
i
n
s
.
O
p
t
i
m
i
s
t
i
c
C
a
l
l
b
a
c
k

[**getOptimisticCallback\(\)**](#)

Gets the OptimisticCallback being used by this BackingMap and/or Loader or null if the LockStrategy is not optimistic.

i n t	<p>getPartitionId() Gets the partition identifier being used by this BackingMap.</p>
P a r t i t i o n M a n a g e r	<p>getPartitionManager() Allows access to the PartitionManager that is defined for this BackingMap.</p>
b o o l e a n	<p>getPreLoadMode() Returns whether this BackingMap will be asynchronously preloaded or not if a Loader is set.</p>
b o o l e a n	<p>getReadOnly() Retrieves the map type.</p>
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . p l u g i n s .	<p>getSerializerAccessor() Retrieve the SerializerAccessor for this map.</p>

i
o
.
S
e
r
i
a
l
i
z
e
r
A
c
c
e
s
s
o
r

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g
i
n
s
.
B
a
c
k
i
n
g
M
a
p
L
i
f
e
c
y
c
l

[getState\(\)](#)

Retrieve the current life cycle state of this map.

e
L
i
s
t
e
n
e
r
.
S
t
a
t
e

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
T
i
m
e
B
a
s
e
d
D
B
U
p
d
a
t
e
C
o
n
f
i
g

[getTimeBasedDBUpdateConfig\(\)](#)

Get the time-based database update configuration object.

i
n
t

[getTimeToLive\(\)](#)

Gets the number of seconds for an entry to live.

I
I
L

[getTtlEvictorType\(\)](#)

[Type](#)

Gets how expiration time of a BackingMap entry is computed.

com.ibm.websphere.obbj.ctgrid.OutputFormat

[getValueOutputFormat\(\)](#)

Retrieves the data output format for all data access APIs that return cache values.

[String](#)

[getWriteBehind\(\)](#)

Get the write-behind parameter.

boolean

[isNearCacheEnabled\(\)](#)

If true, the client near cache is enabled for supported configurations.

boolean

[isNearCacheInvalidationEnabled\(\)](#)

If true, clients with local caches are automatically invalidated when the data grid map is updated.

boolean

[isNearCacheLastAccessTTLSyncEnabled\(\)](#)

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

a n	
v o i d	removeDynamicIndex (String name) Removes a dynamic index on the BackingMap.
v o i d	removeMapEventListener (com.ibm.websphere.objectgrid.plugins.EventListener eventListener) Removes an EventListener from this BackingMap.
v o i d	removeMapEventListener (MapEventListener eventListener) Provided for compatibility with old releases, use the removeMapEventListener(EventListener) method.
v o i d	setCopyKey (boolean copy) Sets whether or not the key needs to be copied when a map entry is created.
v o i d	setCopyMode (CopyMode mode, Class valueInterface) Sets the CopyMode.
v o i d	setEvictionTriggers (String evictionTriggers) Sets the types of additional eviction triggers, all evictors for the backing map will use the provided set of triggers.
v o i d	setEvictor (Evictor e) Associates an Evictor with this BackingMap.
v o i d	setKeyOutputFormat (com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data output format for all data access APIs that return cache keys.
v o i d	setLoader (com.ibm.websphere.objectgrid.plugins.Loader loader) Associates a Loader with this BackingMap.
v o i d	setLockStrategy (LockStrategy lockStrategy) Sets the LockStrategy.
v o i d	setLockTimeout (int seconds) Sets the lock timeout used by the lock manager for this BackingMap.
v o i d	setMapEventListeners (List eventListenerList) Deprecated. <i>This method is deprecated in version 7.1.1. Use the addMapEventListener(EventListener) or removeMapEventListener(EventListener) methods. Plugins that implement the ObjectGridLifecycleListener interface are automatically registered with the grid. Using this method will remove those automatically added listeners.</i>
v o i d	setMapIndexPlugins (List indexList) Sets the list of MapIndexPlugin objects for this BackingMap.
v	

o i d	<p>setNullValuesSupported(boolean nullValuesSupported) Sets whether this BackingMap supports null values.</p>
v o i d	<p>setNumberOfBuckets(int numBuckets) Deprecated. <i>Deprecated in 8.6. Buckets are no longer required. Use the isNearCacheEnabled() flag to disable the near cache in the ObjectGrid configuration XML file.</i></p>
v o i d	<p>setNumberOfLockBuckets(int numBuckets) Sets the number of lock buckets used by the lock manager for this BackingMap.</p>
v o i d	<p>setObjectTransformer(com.ibm.websphere.objectgrid.plugins.ObjectTransformer t) Sets the ObjectTransformer object for use by this BackingMap and/or Loader.</p>
v o i d	<p>setOptimisticCallback(com.ibm.websphere.objectgrid.plugins.OptimisticCallback checker) Sets the OptimisticCallback.</p>
v o i d	<p>setPreloadMode(boolean async) Sets the preload mode if a Loader is set for this BackingMap.</p>
v o i d	<p>setReadOnly(boolean readOnlyEnabled) Sets the map type of this BackingMap.</p>
v o i d	<p>setTimeBasedDBUpdateConfig(com.ibm.websphere.objectgrid.TimeBasedDBUpdateConfig dbUpdateConfig) Set the time-based database update configuration object.</p>
v o i d	<p>setTimeToLive(int seconds) Sets "time to live" of each map entry in seconds.</p>
v o i d	<p>setTtlEvictorType(TTLType type) Sets how expiration time of a BackingMap entry is computed.</p>
v o i d	<p>setValueOutputFormat(com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data output format for all data access APIs that return cache values.</p>
v o i d	<p>setWriteBehind(String writeBehindParam) Enable write-behind updates for this map.</p>

Methods inherited from interface

com.ibm.websphere.objectgrid.[ClientReplicableMap](#)

[disableClientReplication](#), [enableClientReplication](#), [getReplicationMode](#)

Field Detail

DEFAULT_LOCK_TIMEOUT

static final int **DEFAULT_LOCK_TIMEOUT**

Default lock timeout used if `setLockTimeout(int)` is not invoked.

See Also:

[Constant Field Values](#)

DEFAULT_NUMBER_OF_BUCKETS

static final int **DEFAULT_NUMBER_OF_BUCKETS**

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Default number of lock buckets used if `setNumberOfBuckets(int)` is not invoked.

See Also:

[Constant Field Values](#)

DEFAULT_NUMBER_OF_LOCK_BUCKETS

static final int **DEFAULT_NUMBER_OF_LOCK_BUCKETS**

Default number of lock buckets used if `setNumberOfLockBuckets(int)` is not invoked.

See Also:

[Constant Field Values](#)

EVICTIONTRIGGER_MEMORY_USAGE_THRESHOLD

static final [String](#) **EVICTIONTRIGGER_MEMORY_USAGE_THRESHOLD**

The eviction trigger string constant to enable memory based eviction using memory usage threshold provided by the `java.lang.management.MemoryPoolMXBean`.

Since:

WAS XD 6.1.0.3

See Also:

[setEvictionTriggers\(String\)](#), [Constant Field Values](#)

LOCAL

static final int **LOCAL**

Constant used to indicate this map is not a distributed map.

Since:

WAS XD 6.1

See Also:

[getMapType\(\)](#), [Constant Field Values](#)

SERVER

static final int **SERVER**

Constant used to indicate this map is a server map.

Since:

WAS XD 6.1

See Also:

[getMapType\(\)](#), [Constant Field Values](#)

CLIENT

static final int **CLIENT**

Constant used to indicate this map is a client to a server map

Since:

WAS XD 6.1

See Also:

[getMapType\(\)](#), [Constant Field Values](#)

Method Detail

getName

[String](#) getName()

Gets the name of the BackingMap.

Returns:

value specified when BackingMap was created.

setEvictor

void setEvictor([Evictor](#) e)

Associates an Evictor with this BackingMap.

An Evictor aids with cleaning up the cache based on whatever algorithm is desired (LRU, LFU, etc). Passing null to this method removes a previously set Evictor object from an earlier invocation of this method.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

An Evictor that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous evictor which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

An Evictor may also implement the `BackingMapPlugin` interface in order to receive enhanced `BackingMap` plug-in lifecycle method calls. The plug-in is then also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

e - Evictor instance

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[Evictor](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getEvictor

[Evictor](#) `getEvictor()`

Gets the Evictor being used by this BackingMap.

Returns:

the argument that was passed to the `setEvictor(Evictor)` method of this interface or `null` if `setEvictor` was not previously called for this BackingMap object.

See Also:

[Evictor](#), [setEvictor\(Evictor\)](#)

setObjectTransformer

`void setObjectTransformer(com.ibm.websphere.objectgrid.plugins.ObjectTransformer t)`

Sets the ObjectTransformer object for use by this BackingMap and/or Loader.

An ObjectTransformer aids with the "serialization" of non-Serializable objects. It allows a custom copy function to be installed for more efficient object copy operations.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

An ObjectTransformer that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous transformer which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

t - ObjectTransformer instance

Throws:

[IllegalArgumentException](#) - if the passed in ObjectTransformer is null

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

ObjectTransformer, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getObjectTransformer

`com.ibm.websphere.objectgrid.plugins.ObjectTransformer getObjectTransformer()`

Gets the ObjectTransformer object being used by this BackingMap and/or Loader.

Returns:

the argument that was passed to the `setObjectTransformer(ObjectTransformer)` method of this interface or the default ObjectTransformer object if the `setObjectTransformer` method was not previously called for this object.

See Also:

ObjectTransformer, [setObjectTransformer\(ObjectTransformer\)](#)

setOptimisticCallback

```
void setOptimisticCallback(com.ibm.websphere.objectgrid.plugins.OptimisticCallback checker)
```

Sets the OptimisticCallback.

The OptimisticCallback will be used to check the versions of cache entries during the commit phase. If no OptimisticCallback was previously set, a default OptimisticCallback will be used. For Entities, the default OptimisticCallback will use a version field that was specified in the entity metadata. For POJO objects or Entities that do not have a version field specified, the default OptimisticCallback uses the entire object as the version value. In order for it to work for POJO objects, the application's value object needs to have a useful equals(Object) method. If your application does not require versioning, but is using Optimistic locking, the NoVersioningOptimistCallback should be used.

Note, to avoid an IllegalStateException, this method must be called prior to the ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

An OptimisticCallback that implements the BackingMapLifecycleListener is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous optimistic callback which implements BackingMapLifecycleListener is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

checker - OptimisticCallback instance

Throws:

[IllegalArgumentException](#) - if the passed in OptimisticCallback is null

[IllegalStateException](#) - if this method is called after the ObjectGrid.initialize() method is called.

See Also:

OptimisticCallback, NoVersioningOptimisticCallback, [LockStrategy.OPTIMISTIC](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getOptimisticCallback

```
com.ibm.websphere.objectgrid.plugins.OptimisticCallback getOptimisticCallback()
```

Gets the OptimisticCallback being used by this BackingMap and/or Loader or null if the LockStrategy is not optimistic.

If no OptimisticCallback was previously set, a default OptimisticCallback will be used. For Entities, the default OptimisticCallback will use a version field that was specified in the entity metadata. For POJO objects or Entities that do not have a version field specified, the default OptimisticCallback uses the entire object as the version value. In order for it to work for POJO objects, the application's value object needs to have a useful equals(Object) method. If your application does not require versioning, but is using Optimistic locking, the NoVersioningOptimistCallback should be used.

Returns:

the argument that was passed to the setOptimisticCallback(OptimisticCallback) method of this interface or the default OptimisticCallback object if the setOptimisticCallback method was not previously called for this object. If Optimistic locking is not being used, this method will return null after ObjectGrid.initialize() has been invoked.

See Also:

NoVersioningOptimisticCallback, OptimisticCallback, [LockStrategy.OPTIMISTIC](#), [setOptimisticCallback\(OptimisticCallback\)](#)

setLoader

```
void setLoader(com.ibm.websphere.objectgrid.plugins.Loader loader)
```

Associates a Loader with this BackingMap.

Only one Loader can be associated with a given BackingMap. Passing null to this method removes a previously set Loader object from an earlier invocation of this method and indicates that this BackingMap is not associated with a Loader.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

A loader that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous loader which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

A Loader may also implement the `BackingMapPlugin` interface in order to receive enhanced `BackingMap` plug-in lifecycle method calls. The plug-in is then also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

loader - Loader instance

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

Loader, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getLoader

`com.ibm.websphere.objectgrid.plugins.Loader getLoader()`

Gets the Loader being used by this BackingMap.

Returns:

the argument that was passed to the `setLoader(Loader)` method of this interface or null if `setLoader` was not previously called for this object.

See Also:

Loader, [setLoader\(Loader\)](#)

setPreloadMode

`void setPreloadMode(boolean async)`

Sets the preload mode if a Loader is set for this BackingMap.

If the parameter is true then the `Loader.preloadMap(Session, BackingMap)` is invoked asynchronously; otherwise it blocks the execution when loading data so the cache is unavailable until preload completes. Preloading occurs during `ObjectGrid` initialization.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

async - If this is true then the cache is loaded asynchronously otherwise it blocks and the cache is unavailable until preload completes.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()`

method is called.

See Also:

`Loader.preloadMap(Session, BackingMap)`

getPreLoadMode

boolean `getPreLoadMode()`

Returns whether this `BackingMap` will be asynchronously preloaded or not if a `Loader` is set.

If true is returned then the `Loader.preloadMap(Session, BackingMap)` method is invoked asynchronously; otherwise it blocks the execution when loading data so the cache is unavailable until preload completes. Preloading occurs during `ObjectGrid` initialization.

Returns:

the argument that was passed to the `setPreloadMode(boolean)` method of this interface or false if `setPreloadMode` was not previously called for this object.

See Also:

`Loader.preloadMap(Session, BackingMap)`, [setPreloadMode\(boolean\)](#)

addMapIndexPlugin

void `addMapIndexPlugin`(`com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index`)
throws `com.ibm.websphere.objectgrid.IndexAlreadyDefinedException`

Adds an `MapIndexPlugin` to this `Map`. This method assumes the index implementation was constructed with the name of the attribute to index. The name of the index is specified when the index is constructed.

Note, to avoid an `IllegalStateException`, this method must be called prior to `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

A `MapIndexPlugin` that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous index which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

`index` - The index implementation.

Throws:

`IndexAlreadyDefinedException` - if this index already exists.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

`MapIndexPlugin`, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getMapIndexPlugins

[List](#) `getMapIndexPlugins()`

Returns the current list of `MapIndexPlugin` objects for this `BackingMap`.

Returns:

The current list of `MapIndexPlugins` for this `BackingMap`. The list is empty if the `addMapIndexPlugin(MapIndexPlugin)` OR `setMapIndexPlugins(List)` method was not previously called for this `BackingMap`.

See Also:

[addMapIndexPlugin\(MapIndexPlugin\)](#), [setMapIndexPlugins\(List\)](#)

setMapIndexPlugins

void **setMapIndexPlugins**([List](#) indexList)

Sets the list of MapIndexPlugin objects for this BackingMap. If the BackingMap already has a List of MapIndexPlugin objects, that list is replaced by the List passed as an argument to the current invocation of this method.

Note, to avoid an `IllegalStateException`, this method must be called prior to `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

indexList - A non-null reference to a List of MapIndexPlugin objects.

Throws:

[IllegalArgumentException](#) - is thrown if indexList is null or the indexList contains either a null reference or an object that is not an instance of MapIndexPlugin.

See Also:

MapIndexPlugin, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

setCopyMode

void **setCopyMode**([CopyMode](#) mode,
[Class](#) valueInterface)

Sets the CopyMode.

The CopyMode determines whether a get operation of an entry in the BackingMap returns the actual value, a copy of the value, or a proxy for the value. In the case of a proxy, the copy of the value does not occur unless a set method of the application provided value interface is invoked. It also determines that when a transaction is committed, whether a copy of the value object of an entry that was marked as dirty by the transaction is put into the BackingMap at commit time. The CopyMode does not specify if the object is copied when being read or written to a Loader. It is the responsibility of the implementor of a Loader to make copies as appropriate. The default CopyMode is `CopyMode.COPY_ON_READ_AND_COMMIT`.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

mode - must be one of the final static variables defined in CopyMode. See CopyMode class for an explanation of each mode and how the valueInterface is used for `CopyMode.COPY_ON_WRITE`.

valueInterface - the value interface Class object. Specify null in version 7.1 and later.

Throws:

[IllegalArgumentException](#) - if mode is `CopyMode.COPY_ON_WRITE` and valueInterface parameter is null and CGLIB isn't in the classpath.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[CopyMode](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getCopyMode

[CopyMode](#) `getCopyMode()`

Gets the CopyMode being used by this BackingMap.

Returns:

the argument that was passed to the `setCopyMode(CopyMode, Class)` method of this interface or the default CopyMode object if `setCopyMode` was not previously called for this object.

See Also:

[CopyMode](#), [setCopyMode\(CopyMode, Class\)](#)

setLockStrategy

`void setLockStrategy(LockStrategy lockStrategy)`

Sets the LockStrategy.

The locking strategy represented by the LockStrategy object determines if the internal ObjectGrid lock manager is used whenever a map entry is accessed by a transaction. The default strategy is `LockStrategy.OPTIMISTIC`.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`lockStrategy` - must be one of the final static variables defined in `LockStrategy`. See `LockStrategy` class for an explanation of each locking strategy.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[LockStrategy](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getLockStrategy

[LockStrategy](#) `getLockStrategy()`

Gets the LockStrategy object being used by this BackingMap.

Returns:

the argument that was passed to the `setLockStrategy(LockStrategy)` method of this interface or the default LockStrategy object if `setLockStrategy` was not previously called for this object.

See Also:

[LockStrategy](#), [setLockStrategy\(LockStrategy\)](#)

setMapEventListeners

[@Deprecated](#)

`void setMapEventListeners(List eventListenerList)`

Deprecated. *This method is deprecated in version 7.1.1. Use the [addMapEventListener\(EventListener\)](#) or [removeMapEventListener\(EventListener\)](#) methods. Plugins that implement the `ObjectGridLifecycleListener` interface are automatically registered with the grid. Using this method will remove those automatically added listeners.*

Sets the list of EventListener objects.

If this BackingMap already has a List of EventListeners, that list is replaced by the List passed as an argument to the current invocation of this method. This method can be called before and after the ObjectGrid.initialize() method.

Parameters:

eventListenerList - A non-null reference to a List of EventListener objects that are instances of BackingMapLifecycleListener OR MapEventListener

Throws:

[IllegalArgumentException](#) - is thrown if eventListenerList is null, the eventListenerList contains either a null reference or an object that is not an instance of BackingMapLifecycleListener OR MapEventListener

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener, [addMapEventListener\(EventListener\)](#), [removeMapEventListener\(EventListener\)](#)

getMapEventListeners

[List](#) getMapEventListeners()

Gets the current list of EventListeners.

Returns:

the current list of EventListener objects for this BackingMap.

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener

addMapEventListener

void addMapEventListener(com.ibm.websphere.objectgrid.plugins.EventListener eventListener)

Adds an EventListener to this BackingMap.

Note, this method is allowed to be invoked before and after the ObjectGrid.initialize() method. Backing map plug-ins (Loader, Evictor, MapIndexPlugin, ObjectTransformer, OptimisticCallback) that implement the ObjectGridLifecycleListener are automatically added as listeners when added to the BackingMap.

Parameters:

eventListener - A non-null reference to a EventListener to add to the list. The listener must be an instance of BackingMapLifecycleListener OR MapEventListener

Throws:

[IllegalArgumentException](#) - if eventListener is null or not an instance of BackingMapLifecycleListener OR MapEventListener

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener

addMapEventListeners

void addMapEventListeners([MapEventListener](#) eventListener)

Deprecated. *This method is deprecated in version 7.1.1, use the [addMapEventListener\(EventListener\)](#) method.*

Provided for compatibility with old releases, use the [addMapEventListener\(EventListener\)](#) method.

Parameters:

eventListener - A non-null reference to a EventListener to add to the list. The listener must be an instance of BackingMapLifecycleListener OR MapEventListener

removeMapEventListener

void **removeMapEventListener**(com.ibm.websphere.objectgrid.plugins.EventListener eventListener)

Removes an EventListener from this BackingMap.

Note, this method is allowed to be invoked before and after the ObjectGrid.initialize() method. Backing map plug-ins (Loader, Evictor, MapIndexPlugin, ObjectTransformer, OptimisticCallback) that implement the ObjectGridLifecycleListener are automatically removed as listeners when removed from the ObjectGrid.

Parameters:

eventListener - A non-null reference to an event listener that was previously added by invoking either the addMapEventListener(EventListener) OR setMapEventListeners(List) method of this interface.

Throws:

[IllegalArgumentException](#) - if eventListener is null or not an instance of BackingMapLifecycleListener OR MapEventListener

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener, [addMapEventListener\(EventListener\)](#)

removeMapEventListener

void **removeMapEventListener**([MapEventListener](#) eventListener)

Provided for compatibility with old releases, use the [removeMapEventListener\(EventListener\)](#) method.

Parameters:

eventListener - A non-null reference to an event listener that was previously added by invoking either the addMapEventListener(EventListener) OR setMapEventListeners(List) method of this interface.

getPartitionId

int **getPartitionId**()

Gets the partition identifier being used by this BackingMap.

Returns:

The 0-based index for the partition represented by this BackingMap instance. If there is only a single partition defined for this BackingMap object, a 0 will be returned (default).

Since:

WAS XD 6.0.1

setReadOnly

void **setReadOnly**(boolean readOnlyEnabled)

Sets the map type of this BackingMap.

A map can be a read only map or a read/write map. Passing true as the parameter value will make this map a read only map; passing false as the parameter value will make this map a read/write map.

Note, to avoid an IllegalStateException, this method must be called prior to the

ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

Parameters:

readOnlyEnabled - If set to true, this BackingMap will be a read only map. If false, the map will be a read/write map.

Throws:

[IllegalStateException](#) - if this method is called after the ObjectGrid.initialize() method is called.

getReadOnly

boolean **getReadOnly()**

Retrieves the map type.

Returns:

the argument that was passed to setReadOnly(boolean) method of this interface. True is returned if this a read only map. A return value of false implies that this is a read/write map. If setReadOnly was never called, the default return value is false.

See Also:

[setReadOnly\(boolean\)](#)

getObjectGrid

[ObjectGrid](#) **getObjectGrid()**

Gets the ObjectGrid that owns this BackingMap.

Returns:

the ObjectGrid instance that owns this BackingMap.

See Also:

[ObjectGrid](#)

setNumberOfBuckets

void **setNumberOfBuckets**(int numBuckets)

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Sets the number of buckets used by this BackingMap.

The BackingMap implementation uses a hash map for its implementation. If there are a lot of entries in the BackingMap then more buckets means better performance because the risk of collisions is lower as the number of buckets grows. More buckets also means more concurrency. If number of buckets is 0, no entries will be stored in the map, but the appropriate ObjectGrid and BackingMap plug-ins will still be called.

Once the ObjectGrid is initialized this parameter cannot be changed. Therefore, to avoid an IllegalStateException, this method must be called prior to the ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

Parameters:

numBuckets - The number of buckets to use.

Throws:

[IllegalArgumentException](#) - if numBuckets is less than 0.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getNumberOfBuckets

```
int getNumberOfBuckets()
```

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Gets the number of buckets defined for this BackingMap.

Returns:

the same value passed to the `setNumberOfBuckets(int)` method or `DEFAULT_NUMBER_OF_BUCKETS` if `setNumberOfBuckets` was never called.

See Also:

[setNumberOfBuckets\(int\)](#), [DEFAULT_NUMBER_OF_BUCKETS](#)

setNumberOfLockBuckets

```
void setNumberOfLockBuckets(int numBuckets)
```

Sets the number of lock buckets used by the lock manager for this BackingMap.

When `LockStrategy.OPTIMISTIC` or `LockStrategy.PESSIMISTIC` is used for this BackingMap, a lock manager is created for the BackingMap. The lock manager uses a hash map to keep track of entries that are locked by 1 or more transactions. If there are a lot of entries in the hash map, then more lock buckets means better performance as the risk of collisions is lower as the number of buckets grows. More lock buckets also means more concurrency. When the lock strategy is `LockStrategy.NONE`, no lock manager is used by this BackingMap. In this case, a call to this method does nothing.

Once the `ObjectGrid` is initialized, the number of lock buckets cannot be changed. Therefore, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`numBuckets` - The number of lock buckets to use.

Throws:

[IllegalArgumentException](#) - if numBuckets is less than 1.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[LockStrategy](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getNumberOfLockBuckets

```
int getNumberOfLockBuckets()
```

Gets the number of lock buckets defined for the hash map used by lock manager for this backing map.

Returns:

the same value passed to the `setNumberOfLockBuckets(int)` method or `DEFAULT_NUMBER_OF_LOCK_BUCKETS` if `setNumberOfLockBuckets` was never called.

See Also:

[setNumberOfLockBuckets\(int\)](#), [DEFAULT_NUMBER_OF_LOCK_BUCKETS](#)

setLockTimeout

void **setLockTimeout**(int seconds)

Sets the lock timeout used by the lock manager for this `BackingMap`.

When `LockStrategy.OPTIMISTIC` or `LockStrategy.PESSIMISTIC` is used for this `BackingMap`, a lock manager is created for the `BackingMap`. To prevent deadlocks from occurring, the lock manager has a default timeout value for waiting for a lock to be granted. If this timeout limit is exceeded, a `LockTimeoutException` is thrown. The default value of `DEFAULT_LOCK_TIMEOUT` should be sufficient for most applications, but on a heavily loaded system, a timeout may occur when no deadlock exists. In that case, this method can be used to increase the lock timeout value from the default to whatever is desired to prevent false timeout exceptions from occurring. When the lock strategy is `LockStrategy.NONE`, no lock manager is used by this `BackingMap`. In this case, a call to this method does nothing. A lock timeout value of zero indicates to not wait for the lock if it is not immediately available.

Once the lock manager is initialized, the lock timeout value cannot be changed. Therefore, to avoid an `IllegalStateException`, this method must be called prior to `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application. When an entry is fetched the lock timeout can be changed for a given transaction using `ObjectMap.setLockTimeout(int)`

Parameters:

seconds - is the lock timeout value to use in seconds.

Throws:

[IllegalArgumentException](#) - if seconds is less than 0.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[DEFAULT_LOCK_TIMEOUT](#), [LockStrategy](#), [LockTimeoutException](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#), [ObjectMap.setLockTimeout\(int\)](#)

getLockTimeout

int **getLockTimeout**()

Gets the lock timeout value used by the lock manager for this `BackingMap`.

Returns:

the same value passed to the `setLockTimeout(int)` method or `DEFAULT_LOCK_TIMEOUT` if `setLockTimeout` was never called.

See Also:

[DEFAULT_LOCK_TIMEOUT](#), [setLockTimeout\(int\)](#)

setNullValuesSupported

void **setNullValuesSupported**(boolean nullValuesSupported)

Sets whether this `BackingMap` supports null values.

If null values are supported, users need to be careful when a get operation returns a null

reference. It could be due to the fact that the key is not found in the BackingMap, or that the value in the BackingMap is null. To determine if a key was not found, or the value is null, the `containsKey` method can be used.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`nullValuesSupported` - If set to true, null values are supported; otherwise null values are not supported.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#), [ObjectMap.containsKey\(Object\)](#)

getNullValuesSupported

boolean `getNullValuesSupported()`

Gets whether this BackingMap supports null values or not.

Returns:

the same value passed to the `setNullValuesSupported(boolean)` method or the default value of true if `setNullValuesSupported` was never called.

See Also:

[setNullValuesSupported\(boolean\)](#)

setCopyKey

void `setCopyKey(boolean copy)`

Sets whether or not the key needs to be copied when a map entry is created.

Copying the key object allows the application to use the same key object for each `ObjectMap` operation. The application changes the key object state prior to each `ObjectMap` operation so that it can work with different entries using the same key object. If a separate key object is used for each entry, then there is no reason to copy the key object. This attribute allows an application to make the tradeoff of copying key object versus using more memory as a result of separate key object used by the application for each entry. If this method is not called, then the default of false is used (e.g. the key is NOT copied).

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`copy` - If true is specified, then this BackingMap uses the `ObjectTransformer.copyKey(Object)` method to copy the key object when necessary.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#), `ObjectTransformer.copyKey(Object)`

getCopyKey

boolean `getCopyKey()`

Gets whether keys are copied for this BackingMap.

Returns:

the same value passed to the `setCopyKey(boolean)` method or the default value of false if `setCopyKey` was never called.

See Also:

[setCopyKey\(boolean\)](#)

setTimeToLive

void `setTimeToLive(int seconds)`

Sets "time to live" of each map entry in seconds.

If this method is not called, the lifetime of an entry is forever (or until the application explicitly removes or invalidates the entry, or a user defined Evictor evicts the entry). Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

seconds - the number of seconds a map entry is allowed to live in map before being evicted.

Throws:

[IllegalArgumentException](#) - if seconds is less than 0.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[setTtlEvictorType\(TTLType\)](#), [ObjectMap.setTimeToLive\(int\)](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getTimeToLive

int `getTimeToLive()`

Gets the number of seconds for an entry to live.

This value returned is in seconds and 0 indicates forever.

Returns:

the same value passed to the `setTimeToLive(int)` method or 0 if `setLockTimeout` was never called.

See Also:

[setTimeToLive\(int\)](#)

setTtlEvictorType

void `setTtlEvictorType(TTLType type)`

Sets how expiration time of a BackingMap entry is computed.

If this method is not called, `TTLType.NONE` is used to indicate the map entry has no expiration time (e.g. is allowed to live until explicitly removed or invalidated by the application, or evicted by a user defined Evictor).

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

type - must be one of the public constants declared in the `TTLType` class.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[TTLType](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getTtlEvictorType

[TTLType](#) getTtlEvictorType()

Gets how expiration time of a `BackingMap` entry is computed.

Returns:

the `TTLType` that was passed to the `setTtlEvictorType(TTLType)` or `TTLType.NONE` if `setTtlEvictorType` was never called.

See Also:

[setTtlEvictorType\(TTLType\)](#), [TTLType](#)

createDynamicIndex

```
void createDynamicIndex(String name,  
                        boolean isRangeIndex,  
                        String attributeName,  
                        com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)  
                        throws com.ibm.websphere.objectgrid.IndexAlreadyDefinedException,  
                               IllegalArgumentException
```

Creates a dynamic index on the `BackingMap`.

Required Client Permission: `ServerMapPermission.REPLICATE`

Parameters:

name - the name of the index. The name can not be null or a zero length string.

isRangeIndex - Indicate whether to create a `MapRangeIndex` or a `MapIndex`. If set to true, the index will be a type of `MapRangeIndex`.

attributeName - The name of the attribute to be indexed. The attributeName can not be null or a zero length string.

dynamicIndexCallback - The callback that will invoke upon dynamic index events. The `dynamicIndexCallback` is optional and can be null.

Throws:

[IllegalArgumentException](#) - if name or attributeName is null or a zero length string.

`IndexAlreadyDefinedException` - if a `MapIndexPlugin` with the specified name already exists.

Since:

WAS XD 6.0.1

See Also:

`MapIndex`, `MapIndexPlugin`, `MapRangeIndex`, [ObjectMap.getIndex\(String\)](#)

createDynamicIndex

```
void createDynamicIndex(com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index,
```

```
com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)
throws com.ibm.websphere.objectgrid.IndexAlreadyDefinedException,
       IllegalArgumentException
```

Creates a dynamic index on the BackingMap.

Required Client Permission: ServerMapPermission.DYNAMIC_INDEX

A MapIndexPlugin that implements the BackingMapLifecycleListener is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous index which implements BackingMapLifecycleListener is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

index - The index implementation. The index can not be null.

dynamicIndexCallback - The callback that will invoke upon dynamic index events. The dynamicIndexCallback is optional and can be null.

Throws:

[IllegalArgumentException](#) - if index is null or index.getName() returns null or a zero length string.

IndexAlreadyDefinedException - if a MapIndexPlugin with the specified name already exists.

Since:

WAS XD 6.0.1

See Also:

MapIndexPlugin, [ObjectMap.getIndex\(String\)](#)

removeDynamicIndex

```
void removeDynamicIndex(String name)
throws com.ibm.websphere.objectgrid.IndexUndefinedException,
       IllegalArgumentException
```

Removes a dynamic index on the BackingMap.

Required Client Permission: ServerMapPermission.DYNAMIC_INDEX

Parameters:

name - the name of the index. The name can not be null.

Throws:

[IllegalArgumentException](#) - if name is null.

IndexUndefinedException - if a MapIndexPlugin with the specified name does not exist.

Since:

WAS XD 6.0.1

See Also:

[createDynamicIndex\(MapIndexPlugin, DynamicIndexCallback\)](#), [createDynamicIndex\(String, boolean, String, DynamicIndexCallback\)](#)

getPartitionManager

```
PartitionManager getPartitionManager()
```

Allows access to the PartitionManager that is defined for this BackingMap. This access may be useful for Loaders during Loader.preloadMap(Session, BackingMap) processing (to properly partition the data to be loaded).

Returns:

PartitionManager associated with this BackingMap.

Since:

WAS XD 6.0.1

See Also:

getEntityMetadata

com.ibm.websphere.projector.md.EntityMetadata **getEntityMetadata()**

Retrieve the metadata for the entity associated with this backing map.

Returns:

the EntityMetadata if an entity is associated with this backing map or null if there is no entity associated with this backing map.

Since:

WAS XD 6.1

setWriteBehind

void **setWriteBehind**([String](#) writeBehindParam)

Enable write-behind updates for this map.

If a map is configured with write-behind loader update, the updates (could be insert type, remove type, or update type) to the backend are not instantly updated to the back end by calling the Loader.batchUpdate(TxID, LogSequence) method. Instead, they are queued in a write-behind queue map and updated to the back end periodically.

A write-behind update is pushed to the backend periodically within a different transaction from the one the update is made to ObjectGrid. When the write-behind update to the backend fails, for example, due to data integrity problem, it is too late to roll back the original ObjectGrid transaction. ObjectGrid will invalidate the entry and create an entry in a failed database update map. The name of this failed database update map is WriteBehindLoaderConstants.WRITE_BEHIND_FAILED_UPDATES_MAP_PREFIX+baseMapName. The key of the entry in this map is an auto-increment Integer, and the value is a [LogElement](#). The logElement can be used to compensate the failure.

Depending on your grid use case and your back end loader configuration, your back end loader or the back end database might benefit from having upsert operations instead of insert and update operations in the LogElements that it receives for a transaction. Use the ConvertToUpsert=true configuration option on the writeBehindParam to have the write behind loader convert insert and update LogElement operations to upsert LogElement operations when they are passed to the back end loader. Not all back end loaders may support the upsert operation, be certain that the back end loader supports upsert operations before using the ConvertToUpsert=true clause in the writeBehindParam. The default value is ConvertToUpsert=false.

Note, to avoid an IllegalStateException, this method must be called prior to the ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

Parameters:

writeBehindParam - a write-behind parameter consisting of a maximum update time and/or a maximum key update count. The format of the write-behind parameter is "T[time];C[count][;ConvertToUpsert=true]", for example, "T100;C2000". "T100;C2000" means the loader will write to the back end when there are 2000 pending keys to be updated or when 100 seconds have passed since the last update. The default update time is 300 seconds and the default update key count is 1000. You can configure the update time only, the update key count only, or an empty string. The default value(s) will then be used in either of the above three cases. The default value is null to disable write-behind updates.

Throws:

[IllegalArgumentException](#) - if the write behind parameters are unknown or improperly formatted.

[IllegalStateException](#) - if this method is called after the ObjectGrid.initialize() method is called.

Since:

WAS XD 6.1.0.3

See Also:

[WriteBehindLoaderConstants](#)

getWriteBehind

[String](#) `getWriteBehind()`

Get the write-behind parameter. A write-behind parameter consists of a maximum update time and/or a maximum key update count. The format of the write-behind parameter is "T[time];C[count][;ConvertToUpsert=true]".

Returns:

the write-behind parameter. If the write-behind parameter is not set, `null` will be returned.

Since:

WAS XD 6.1.0.3

See Also:

[setWriteBehind\(String\)](#)

setTimeBasedDBUpdateConfig

`void setTimeBasedDBUpdateConfig(com.ibm.websphere.objectgrid.TimeBasedDBUpdateConfig dbUpdateConfig)`

Set the time-based database update configuration object.

When a time-based database update configuration object is set, a thread will be started automatically to update or invalidate the ObjectGrid maps with the latest updates (inserts and updates) from the database.

For a local ObjectGrid map, the thread will be launched in the same JVM. For a distributed ObjectGrid map in an ObjectGrid container, the thread will be automatically launched in partition 0. No database update thread will be started in a client side near cache.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`dbUpdateConfig` - the time-based database update configuration object or `null`.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

Since:

WAS XD 6.1.0.3

getTimeBasedDBUpdateConfig

`com.ibm.websphere.objectgrid.TimeBasedDBUpdateConfig getTimeBasedDBUpdateConfig()`

Get the time-based database update configuration object.

Returns:

the time-based database update configuration object or `null` if not set.

Since:

WAS XD 6.1.0.3

See Also:

[setTimeBasedDBUpdateConfig\(TimeBasedDBUpdateConfig\)](#)

getMapType

int **getMapType**()

Returns the type of BackingMap.

The return value is equivalent to one of the constants declared on this interface, [LOCAL](#), [SERVER](#), or [CLIENT](#).

Returns:

the map type

Since:

WAS XD 6.1

getEvictionTriggers

[String](#) **getEvictionTriggers**()

Returns the types of additional eviction triggers.

The available eviction trigger strings are as described in the String constants in this interface that begin with the name: EVICTIONTRIGGER.

Returns:

a semicolon separated list of eviction triggers

Since:

WAS XD 6.1.0.3

setEvictionTriggers

void **setEvictionTriggers**([String](#) evictionTriggers)

Sets the types of additional eviction triggers, all evictors for the backing map will use the provided set of triggers.

The available eviction trigger strings are as described in the String constants in this interface that begin with the name: EVICTIONTRIGGER.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

evictionTriggers - a semicolon separated list of eviction triggers

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

[IllegalArgumentException](#) - if the eviction triggers are unknown or improperly formatted.

Since:

WAS XD 6.1.0.3

getMapSetName

[String](#) `getMapSetName()`

Retrieves the name of the MapSet that this BackingMap is currently associated with. A "null" return value indicates it is currently not associated with a MapSet. This method will only return a non null value for a client or server map.

Returns:

name of associated MapSet

Since:

7.1

See Also:

[getMapType\(\)](#)

getSerializerAccessor

`com.ibm.websphere.objectgrid.plugins.io.SerializerAccessor` **getSerializerAccessor()**

Retrieve the SerializerAccessor for this map.

Returns:

the SerializerAccessor

Since:

7.1.1

getState

`com.ibm.websphere.objectgrid.plugins.BackingMapLifecycleListener.State` **getState()**

Retrieve the current life cycle state of this map.

Returns:

the current state.

Since:

7.1.1

isNearCacheInvalidationEnabled

`boolean` **isNearCacheInvalidationEnabled()**

If true, clients with local caches are automatically invalidated when the data grid map is updated.

Returns:

true if client near cache invalidation is enabled.

Since:

8.6, XC10 2.5

isNearCacheLastAccessTTLSyncEnabled

`boolean` **isNearCacheLastAccessTTLSyncEnabled()**

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

Returns:

true if last-access time-to-live information is sent to the remote data grid.

Since:

8.6, XC10 2.5

isNearCacheEnabled

boolean **isNearCacheEnabled()**

If true, the client near cache is enabled for supported configurations. The client near cache can only be enabled when using optimistic locking or when locking is disabled.

Returns:

true if the client near cache is enabled.

getKeyOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getKeyOutputFormat()**

Retrieves the data format for all data access APIs that return cache keys.

This value does not reflect the data output format that plug-ins will see. See the `PluginOutputFormat` annotation for details on how to influence the data object format that plug-ins receive.

Returns:

the data output format.

Since:

8.6, XC10 2.5

setKeyOutputFormat

void **setKeyOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)

Sets the data output format for all data access APIs that return cache keys.

When set to `OutputFormat.UNDEFINED`, the key output format defaults to `OutputFormat.RAW` when using a custom `KeyDataSerializer` plug-in. The key output format is `OutputFormat.NATIVE` in all other cases.

Parameters:

outputFormat - the data output format to use or `OutputFormat.UNDEFINED` to use the default.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

Since:

8.6, XC10 2.5

getValueOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getValueOutputFormat()**

Retrieves the data output format for all data access APIs that return cache values.

This value does not reflect the data output format that plug-ins will see. See the `PluginOutputFormat` annotation for details on how to influence the data object format that plug-ins receive.

Returns:

the data output format or `OutputFormat.UNDEFINED` if the default should be used.

Since:

8.6, XC10 2.5

setValueOutputFormat

void **setValueOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)

Sets the data output format for all data access APIs that return cache values.

When set to `OutputFormat.UNDEFINED`, the value output format defaults to `OutputFormat.RAW` when using a custom `ValueDataSerializer` plug-in or when the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` set.

The value output format is `OutputFormat.NATIVE` in all other cases.

Parameters:

`outputFormat` - the data output format to use or `OutputFormat.UNDEFINED` to use the default.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

Since:

8.6, XC10 2.5

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All				
			Classes					
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD					

com.ibm.websphere.objectgrid

Class AvailabilityState

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public final class AvailabilityState
extends Object
implements Serializable
```

Each shard in a distributed ObjectGrid has an availability state associated with it. This state refers to the shard's ability to process incoming requests.

Since:

WAS XD 6.1.0.3, XC10

See Also:

[Serialized Form](#)

Field Summary	
s t a t i c A v a i l l e O F F L I N E	An AvailabilityState.OFFLINE indicates that a shard is offline and unable to process requests.
s t a t i c A v a i l l e O N L I N E	An AvailabilityState.ONLINE indicates that a shard is online and able to process requests.

b
i
l
l
i
t
y
S
t
a
t
e

An AvailabilityState.ONLINE indicates that a shard is online.

s
t
a
t
i
c
A
v
a
i
l
i
t
y
S
t
a
t
e

PRELOAD

An AvailabilityState.PRELOAD indicates that a shard is in the preload state.

s
t
a
t
i
c
A
v
a
i
l
i
t
y
S
t
a
t
e

QUIESCE

An AvailabilityState QUIESCE indicates that a shard is in quiesce.

s
t
a
t
i
c
A
v
a
i
l
i
t
y

UNKNOWN

An AvailabilityState.UNKNOWN indicates that the availability state of the shard could not be determined.

S
t
a
t
e

Method Summary

getId()
Returns the internal identifier for this state.

toString()

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

OFFLINE

```
public static final AvailabilityState OFFLINE
```

An `AvailabilityState.OFFLINE` indicates that a shard is offline and unable to process requests.

PRELOAD

```
public static final AvailabilityState PRELOAD
```

An `AvailabilityState.PRELOAD` indicates that a shard is in the preload state. When in the preload state, a shard will reject all requests that are not initiated from a client that is preloading data into the ObjectGrid.

ONLINE

```
public static final AvailabilityState ONLINE
```

An `AvailabilityState.ONLINE` indicates that a shard is online. A shard that is online is available for processing requests.

QUIESCE

```
public static final AvailabilityState QUIESCE
```

An `AvailabilityState QUIESCE` indicates that a shard is in quiesce. Quiesce is a transitional state. Shards that are in the quiesce state are on their way to being offline. A shard in the quiesce state will allow all pending transactions to complete before moving to the `AvailabilityState.OFFLINE`, assuming that all pending transactions complete within 30 seconds after entering the quiesce state.

UNKNOWN

public static final [AvailabilityState](#) UNKNOWN

An AvailabilityState.UNKNOWN indicates that the availability state of the shard could not be determined.

Method Detail

getId

public int **getId**()

Returns the internal identifier for this state.

Returns:

the internal id.

Since:

7.1.1

toString

public [String](#) **toString**()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

PREV CLASS [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | [FIELD](#) | CONSTR | [METH](#) DETAIL: [FIELD](#) | CONSTR | [METHOD](#) [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package **com.ibm.websphere.xs.ra**

These are the eXtreme Scale Resource Adapter APIs that allows integration with a Java EE application.

See:

[Description](#)

Interface Summary	
ObjectGridJ2CConnectionMBean	Allows administration clients control the lifecycle of the eXtreme Scale resource adapter's connection.

Class Summary	
XSConnection	An XSConnection represents an application level connection handle to an eXtreme Scale ObjectGrid.
XSConnectionFactory	The XSConnectionFactory creates connections to eXtreme Scale ObjectGrids.
XSConnectionSpec	The ConnectionSpec for retrieving XSConnections from an XSConnectionFactory
XSManagedConnectionFactory	The XSManagedConnectionFactory create managed connections to the eXtreme Scale ObjectGrids.
XSResourceAdapter	The eXtreme Scale resource adapter

Package **com.ibm.websphere.xs.ra** Description

These are the eXtreme Scale Resource Adapter APIs that allows integration with a Java EE application.

Introduction

The eXtreme Scale Resource Adapter provides client connection management and local transaction support, allowing Java EE applications to look-up eXtreme Scale client connections and demarcate transactions using the [LocalTransaction](#) interface or the [Session](#) interface.

When used with WebSphere Application Server with last participant support enabled, the eXtreme Scale transaction can be enlisted in a global transaction as the last, single-phase participant.

Programming Tutorial

The following sections show snippets on the usage of the ConnectionFactory. The resource

adapter is is JCA 1.5 compliant. To use the resource adapter the following steps must be followed:

1. Install the resource adapter
2. Configure a J2C ConnectionFactory
3. Configure a javax.resource.cci.ConnectionFactory resource reference in the application.
4. Look-up the [XSConnectionFactory](#)
5. Choose one of the following transaction options:

Use auto-commit, local transactions:

1. Retrieve a [XSConnection](#)
2. Retrieve and use the [Session](#) to interact with the data grid.
3. Close the connection.

Use an ObjectGrid Session to demarcate a local transaction:

1. Retrieve a [XSConnection](#)
2. Retrieve the [Session](#)
3. Use the Session.begin() method to start the transaction.
4. Use the Session to interact with the data grid.
5. Use the Session.commit() or rollback() methods to end the transaction.
6. Close the connection.

Use a javax.resource.cci.LocalTransaction to demarcate a local transaction:

1. Retrieve a [XSConnection](#)
2. Retrieve the javax.resource.cci.LocalTransaction using the XSConnection.getLocalTransaction() method.
3. Use the LocalTransaction.begin() method to start the transaction.
4. Retrieve and use the [Session](#) to interact with the data grid.
5. Use the LocalTransaction.commit() or rollback() methods to end the transaction.
6. Close the connection.

Enlist the connection in a global transaction:

1. Lookup the UserTransaction.
2. Begin the global transaction
3. Retrieve a [XSConnection](#)
4. Retrieve and use the [Session](#)
5. Close the connection.
6. Commit or rollback the global transaction.

Installing the resource adapter

The eXtreme Scale resource adapter is included in the wxsra.rar resource adapter archive with the eXtreme Scale product. See the [WebSphere eXtreme Scale version 8.5 information center](#) (or later) for details on how to install and configure the resource adapter.

Configuring the J2C ConnectionFactory

The eXtreme Scale resource adapter allows configuring one or J2C ManagedConnectionFactory instances. Each ManagedConnectionFactory is managed by the application server and represents a connection to a single catalog service domain. the ManagedConnectionFactory can include the name of the data grid, or the data grid can be provided when the connection is retrieved by the application.

The ManagedConnectionFactory is configured in the application server configuration, bound to a global JNDI name, and provides the following configuration properties:

ConnectionNa

me	(Optional) The name of the eXtreme Scale client connection.
CatalogServiceEndpoints	The catalog service domain end points in the form: <host>:<port>,<host><port>. Required if catalog service domain is not set.
CatalogServiceDomain	The catalog service domain name. Required if catalog service end points are not set.
ObjectGridName	(Optional) The data grid name. If not specified, the client must provide the data grid name when retrieving the connection resource.
ObjectGridURL	(Optional) The URL of the client data grid, override XML file.
ObjectGridResource	(Optional) The resource path of the client data grid, override XML file.
ClientPropertiesURL	(Optional) The URL of the client properties file.
ClientPropertiesResource	(Optional) The resource path of the client properties file.

Obtaining a ConnectionFactory instance.

The Java EE application can use resource injection to inject a ConnectionFactory resource into the application, or it can be looked-up using a resource reference. The Java EE application must first configure resource reference for a javax.resource.cci.ConnectionFactory.

For example:

```
InitialContext ctx = new InitialContext();
XSConnectionFactory cf = (XSConnectionFactory) ctx.lookup("java:comp/env/wxsconnection");
```

Retrieving a Connection

After the [XSConnectionFactory](#) has been looked-up or injected into the application, use one of the getConnection() methods to retrieve a client connection to the data grid. The connection will automatically be established when the first connection is retrieved and will be maintained until the resource adapter is stopped or the connection is reset using the ObjectGridJ2CConnection management bean.

For example:

```
XSConnection con = cf.getConnection("MyGrid");
```

Obtaining an ObjectGrid Session instance.

The XSConnection provides a getSession() method that gives the application direct access to the ObjectGrid Session. The Session is used to interact with the data grid and is valid for the life of the XSConnection. The XSConnection is a handle to a connection and becomes invalid after the application context completes per the Java EE specification.

An eXtreme Scale local transaction can be driven by the Session, javax.resource.cci.LocalTransaction or a global transaction. The transaction methods cannot be mixed.

Closing a connection

After the application has finished using the connection, the connection must be closed. The Java EE container typically will also close the connection automatically at the appropriate times. When the connection is closed the Session and any other objects retrieved directly or indirectly from the connection become invalid.

com.ibm.websphere.xs.ra

Class XSManagedConnectionFactory

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#), [ManagedConnectionFactory](#), [ResourceAdapterAssociation](#)

```

public final class XSManagedConnectionFactory
extends Object
implements ManagedConnectionFactory, ResourceAdapterAssociation
  
```

The XSManagedConnectionFactory create managed connections to the eXtreme Scale ObjectGrids.

Since:

8.5, XC10

See Also:

[Serialized Form](#)

Constructor Summary

XSManagedConnectionFactory()
--

Method Summary

Object	createConnectionFactory() Creates a non-managed XSConnectionFactory instance.
Object	createConnectionFactory(ConnectionManager conMgr) Creates a managed XSConnectionFactory instance.
ManagedConnection	createManagedConnection(Subject subject, ConnectionRequestInfo conReq) Creates a new physical connection to the underlying ObjectGrid

C
t
i
o
n

V
o
i
d

[destroy\(\)](#)

Destroy this MCF and release any other resources.

b
o
o
l
e
a
n

[equals\(Object obj\)](#)

S
t
r
i
n
g

[getCatalogServiceDomain\(\)](#)

Get the eXtreme Scale specific property: CatalogDomain

S
t
r
i
n
g

[getCatalogServiceEndpoints\(\)](#)

Get the eXtreme Scale specific property: CatalogServiceEndpoints

S
t
r
i
n
g

[getClientPropertiesResource\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesResource

S
t
r
i
n
g

[getClientPropertiesURL\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesURL

S
t
r
i
n
g

[getConnectionName\(\)](#)

Get the eXtreme Scale specific property: ConnectionName

P
r
i
n
t
W
r
i
t
e
r

[getLogWriter\(\)](#)

Get the log writer for this XManagedConnectionFactory instance.

S
t
r
i
n
g

[getObjectGridName\(\)](#)

Get the eXtreme Scale specific property: ObjectGridName

S t r i n g	<p>getObjectGridResource() Get the eXtreme Scale specific property: ObjectGridResource</p>
S t r i n g	<p>getObjectGridURL() Get the eXtreme Scale specific property: ObjectGridURL</p>
R e s o u r c e A d a p t e r	<p>getResourceAdapter() Retrieve the associated eXtreme Scale resource adapter instance.</p>
i n t	<p>hashCode()</p>
b o o l e a n	<p>isLocalGrid() Retrieve the eXtreme Scale specific property: LocalGrid</p>
M e n a g e d C o n n e c t i o n	<p>matchManagedConnections(Set connectionSet, Subject subject, ConnectionRequestInfo cxRequestInfo) Returns a matched connection from the candidate set of connections.</p>
V o i d	<p>setCatalogServiceDomain(String catalogServiceDomain) Set the eXtreme Scale specific property: CatalogDomain</p>
V o i d	<p>setCatalogServiceEndpoints(String catalogServiceEndpoints) Set the eXtreme Scale specific property: CatalogServiceEndpoints</p>
V o	<p>setClientPropertiesResource(String clientPropsResource) Get the eXtreme Scale specific property: ClientPropertiesResource</p>

i d	
v o i d	setClientPropertiesURL (String clientPropsURL) Set the eXtreme Scale specific property: ClientPropertiesURL
v o i d	setConnectionName (String connName) Set the eXtreme Scale specific property: ConnectionName
v o i d	setLocalGrid (boolean useLocal) Set the eXtreme Scale specific property: LocalGrid
v o i d	setLogWriter (PrintWriter out) Set the log writer for this ManagedConnectionFactory instance
v o i d	setObjectGridName (String objectGridName) Set the eXtreme Scale specific property: ObjectGrid Name
v o i d	setObjectGridResource (String objectGridResource) Set the eXtreme Scale specific property: ObjectGridResource
v o i d	setObjectGridURL (String objectGridURL) Set the eXtreme Scale specific property: ObjectGridURL
v o i d	setResourceAdapter (ResourceAdapter ra) Associate this object with an eXtreme Scale resource adapter.
S t r i n g	toString ()

Methods inherited from class [java.lang.Object](#)

[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XManagedConnectionFactory

```
public XManagedConnectionFactory()
```

Method Detail

getResourceAdapter

```
public ResourceAdapter getResourceAdapter()
```

Retrieve the associated eXtreme Scale resource adapter instance.

Specified by:

[getResourceAdapter](#) in interface [ResourceAdapterAssociation](#)

Returns:

The associated eXtreme Scale resource adapter

See Also:

[ResourceAdapterAssociation.getResourceAdapter\(\)](#)

setResourceAdapter

```
public void setResourceAdapter(ResourceAdapter ra)
    throws ResourceException
```

Associate this object with an eXtreme Scale resource adapter. Note, this method must be called exactly once. That is, the association must not change during the lifetime of this object.

Specified by:

[setResourceAdapter](#) in interface [ResourceAdapterAssociation](#)

Parameters:

ra - the eXtreme Scale resource adapter

Throws:

[ResourceException](#)

[IllegalStateException](#)

See Also:

[ResourceAdapterAssociation.setResourceAdapter\(ResourceAdapter\)](#)

createConnectionFactory

```
public Object createConnectionFactory()
    throws ResourceException
```

Creates a non-managed XSConnectionFactory instance. The XSConnectionFactory instance gets initialized with a default XSConnectionFactory instance provided by the eXtreme Scale resource adapter.

Specified by:

[createConnectionFactory](#) in interface [ManagedConnectionFactory](#)

Returns:

the new XSConnectionFactory instance

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.createConnectionFactory\(\)](#)

createConnectionFactory

```
public Object createConnectionFactory(ConnectionFactory conMgr)
    throws ResourceException
```

Creates a managed XSConnectionFactory instance. The XSConnectionFactory instance gets initialized with the passed ConnectionManager. In the managed scenario, ConnectionManager is provided by the application server.

Specified by:

[createConnectionFactory](#) in interface [ManagedConnectionFactory](#)

Parameters:

conMgr - - Connection Manager to be associated with the XSConnectionFactory.

Returns:

the new XSConnectionFactory instance

Throws:

[ResourceException](#) - ResourceAdapterInternalException

See Also:

[ManagedConnectionFactory.createConnectionFactory\(ConnectionManager\)](#)

createManagedConnection

```
public ManagedConnection createManagedConnection(Subject subject,  
                                                    ConnectionRequestInfo conReq)  
    throws ResourceException
```

Creates a new physical connection to the underlying ObjectGrid

Specified by:

[createManagedConnection](#) in interface [ManagedConnectionFactory](#)

Parameters:

subject - - Caller's security information. WebSphere eXtreme Scale client uses the credentials specified in a CredentialGenerator as part of the ConnectionSpec or the client properties.

conReq - - XS specific connection properties

Returns:

the new XSManagedConnection instance

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.createManagedConnection\(Subject, ConnectionRequestInfo\)](#)

getLogWriter

```
public PrintWriter getLogWriter()  
    throws ResourceException
```

Get the log writer for this XSManagedConnectionFactory instance.

Specified by:

[getLogWriter](#) in interface [ManagedConnectionFactory](#)

Returns:

PrintWriter

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.getLogWriter\(\)](#)

matchManagedConnections

```
public ManagedConnection matchManagedConnections(Set connectionSet,  
                                                    Subject subject,  
                                                    ConnectionRequestInfo cxRequestInfo)  
    throws ResourceException
```

Returns a matched connection from the candidate set of connections.

Specified by:

[matchManagedConnections](#) in interface [ManagedConnectionFactory](#)

Parameters:

connectionSet - - candidate connection set
subject - - caller's security information
cxRequestInfo - - XS specific connection properties

Returns:

XManagedConnection instance if acceptable match found otherwise null

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.matchManagedConnections\(Set, Subject, ConnectionRequestInfo\)](#)

setLogWriter

```
public void setLogWriter(PrintWriter out)
    throws ResourceException
```

Set the log writer for this ManagedConnectionFactory instance

Specified by:

[setLogWriter](#) in interface [ManagedConnectionFactory](#)

Parameters:

out - - PrintWriter - an out stream for error logging and tracing

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.setLogWriter\(PrintWriter\)](#)

destroy

```
public void destroy()
```

Destroy this MCF and release any other resources.

getConnectionName

```
public String getConnectionName()
```

Get the eXtreme Scale specific property: ConnectionName

Returns:

The name of the eXtreme Scale client connection

setConnectionName

```
public void setConnectionName(String connName)
```

Set the eXtreme Scale specific property: ConnectionName

Parameters:

connName - - The name of the eXtreme Scale client connection

getObjectGridName

```
public String getObjectGridName()
```

Get the eXtreme Scale specific property: ObjectGridName

Returns:

The data grid name

setObjectGridName

public void **setObjectGridName**([String](#) objectGridName)

Set the eXtreme Scale specific property: ObjectGrid Name

Parameters:

objectGridName - - The data grid name

getCatalogServiceEndpoints

public [String](#) **getCatalogServiceEndpoints**()

Get the eXtreme Scale specific property: CatalogServiceEndpoints

Returns:

The catalog service domain end points

setCatalogServiceEndpoints

public void **setCatalogServiceEndpoints**([String](#) catalogServiceEndpoints)

Set the eXtreme Scale specific property: CatalogServiceEndpoints

Parameters:

catalogServiceEndpoints - - The catalog service domain end points

getCatalogServiceDomain

public [String](#) **getCatalogServiceDomain**()

Get the eXtreme Scale specific property: CatalogDomain

Returns:

The catalog service domain name defined in WebSphere Application Server

setCatalogServiceDomain

public void **setCatalogServiceDomain**([String](#) catalogServiceDomain)

Set the eXtreme Scale specific property: CatalogDomain

Parameters:

catalogServiceDomain - - The catalog service domain name defined in WebSphere Application Server

getObjectGridURL

public [String](#) **getObjectGridURL**()

Get the eXtreme Scale specific property: ObjectGridURL

Returns:

The URL of the client data grid override XML file

setObjectGridURL

public void **setObjectGridURL**([String](#) objectGridURL)

Set the eXtreme Scale specific property: ObjectGridURL

Parameters:

objectGridURL - The URL of the client data grid override XML file

getObjectGridResource

public [String](#) **getObjectGridResource**()

Get the eXtreme Scale specific property: ObjectGridResource

Returns:

The resource path of the client data grid override XML file

setObjectGridResource

public void **setObjectGridResource**([String](#) objectGridResource)

Set the eXtreme Scale specific property: ObjectGridResource

Parameters:

objectGridResource - - The resource path of the client data grid override XML file

getClientPropertiesURL

public [String](#) **getClientPropertiesURL**()

Get the eXtreme Scale specific property: ClientPropertiesURL

Returns:

The URL of the client properties file

setClientPropertiesURL

public void **setClientPropertiesURL**([String](#) clientPropsURL)

Set the eXtreme Scale specific property: ClientPropertiesURL

Parameters:

clientPropsURL - - The URL of the client properties file

getClientPropertiesResource

public [String](#) **getClientPropertiesResource**()

Get the eXtreme Scale specific property: ClientPropertiesResource

Returns:

The resource path of the client properties file

setClientPropertiesResource

```
public void setClientPropertiesResource(String clientPropsResource)
```

Get the eXtreme Scale specific property: ClientPropertiesResource

Parameters:

clientPropsResource - - The resource path of the client properties file

setLocalGrid

```
public void setLocalGrid(boolean useLocal)
```

Set the eXtreme Scale specific property: LocalGrid

When set to true, the application uses the [XSConnectionSpec.setLocalObjectGrid\(com.ibm.websphere.objectgrid.ObjectGrid\)](#) to set the ObjectGrid instance to a local, in-memory grid instance or shard instance. If set to false (the default), the connection will be configured as a client connection to a remote data grid.

Parameters:

useLocal - set to true, to disable normal client ObjectGrid connection management.

Since:

8.6, XC10 2.5

isLocalGrid

```
public boolean isLocalGrid()
```

Retrieve the eXtreme Scale specific property: LocalGrid

Returns:

answers true if the connection is to be used with a local, im-memory ObjectGrid instance. Answers false if the connection represents a normal client ObjectGrid connection.

Since:

8.6, XC10 2.5

hashCode

```
public int hashCode()
```

Specified by:

[hashCode](#) in interface [ManagedConnectionFactory](#)

Overrides:

[hashCode](#) in class [Object](#)

See Also:

[ManagedConnectionFactory.hashCode\(\)](#)

equals

```
public boolean equals(Object obj)
```

Specified by:

[equals](#) in interface [ManagedConnectionFactory](#)

Overrides:

[equals](#) in class [Object](#)

See Also:

[ManagedConnectionFactory.equals\(Object\)](#)

toString

public [String](#) toString()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSConnectionSpec

[java.lang.Object](#)



All Implemented Interfaces:
[Cloneable](#), [ConnectionSpec](#)

```

public final class XSConnectionSpec
extends Object
implements ConnectionSpec, Cloneable
  
```

The ConnectionSpec for retrieving XSConnections from an XSConnectionFactory

Since:
 8.5, XC10

See Also:
[XSConnectionFactory](#)

Constructor Summary	
XSConnectionSpec()	Creates a default XSConnectionSpec instance.

Method Summary	
Object	clone()
boolean	equals(Object obj)
CredentialGenerator	getCredentialGenerator() Returns the CredentialGenerator object for this client security for the connection.

e
r
a
t
o
r

O
b
j
e
c
t
G
r
i
d

[getLocalObjectGrid\(\)](#)

Retrieve the local object grid instance used for the connection, if set.

S
t
r
i
n
g

[getObjectGridName\(\)](#)

Returns the object grid name for the connection.

i
n
t

[hashCode\(\)](#)

b
o
o
l
e
a
n

[isBeginNoWriteThrough\(\)](#)

Returns flag indicating whether the Session transaction should be started using using the [Session.beginNoWriteThrough\(\)](#) method.

b
o
o
l
e
a
n

[isMultiPartitionSupportEnabled\(\)](#)

Answers true if the [Session](#) will have the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

v
o
i
d

[setBeginNoWriteThrough\(boolean beginNoWriteThrough\)](#)

Set the boolean flag indicating whether the Session transaction should be started using using the [Session.beginNoWriteThrough\(\)](#) method.

v
o
i
d

[setCredentialGenerator\(CredentialGenerator credGen\)](#)

Sets the CredentialGenerator object for the client security for the connection.

v
o
i
d

[setLocalObjectGrid\(ObjectGrid instance\)](#)

Sets a local ObjectGrid instance to be used for connections.

v
o
i
d

[setMultiPartitionSupportEnabled\(boolean mptEnabled\)](#)

Set to true to retrieve a connection with a [Session](#) that has the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

v
o
i
d

[setObjectGridName\(String objectGridName\)](#)

Sets the object grid name to be used for the connection.

S
t
r

```
toString()
```

Methods inherited from class [java.lang.Object](#)

[finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSCConnectionSpec

```
public XSCConnectionSpec()
```

Creates a default XSCConnectionSpec instance.

Method Detail

isBeginNoWriteThrough

```
public boolean isBeginNoWriteThrough()
```

Returns flag indicating whether the Session transaction should be started using using the `Session.beginNoWriteThrough()` method.

Returns:

flag indicating whether the Session transaction should be started using using the `Session.beginNoWriteThrough()` method.

setBeginNoWriteThrough

```
public void setBeginNoWriteThrough(boolean beginNoWriteThrough)
```

Set the boolean flag indicating whether the Session transaction should be started using using the `Session.beginNoWriteThrough()` method.

Parameters:

`beginNoWriteThrough` -

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#)

getObjectGridName

```
public String getObjectGridName()
```

Returns the object grid name for the connection.

Returns:

object grid name

setObjectGridName

```
public void setObjectGridName(String objectGridName)
```

Sets the object grid name to be used for the connection.

Parameters:

objectGridName - object grid name to be used for the connection

getLocalObjectGrid

```
public ObjectGrid getLocalObjectGrid()
```

Retrieve the local object grid instance used for the connection, if set.

Returns:

null if the connection retrieves its ObjectGrid instance normally from the physical connection, or non-null if [setLocalObjectGrid\(ObjectGrid\)](#) was specified.

Since:

8.6, XC10 2.5

setLocalObjectGrid

```
public void setLocalObjectGrid(ObjectGrid instance)
```

Sets a local ObjectGrid instance to be used for connections. The local ObjectGrid instance overrides the standard behavior and allows a physical connection to be used with a specific local ObjectGrid instance, such as a shard.

Parameters:

instance - the local ObjectGrid instance.

Since:

8.6, XC10 2.5

getCredentialGenerator

```
public CredentialGenerator getCredentialGenerator()
```

Returns the CredentialGenerator object for this client security for the connection.

Returns:

the argument that was passed to the setCredGen(CredentialGenerator) method of this object or null if setCredGen was not previously called for this object.

setCredentialGenerator

```
public void setCredentialGenerator(CredentialGenerator credGen)
```

Sets the CredentialGenerator object for the client security for the connection.

Parameters:

credGen - a CredentialGenerator object

See Also:

[CredentialGenerator](#)

setMultiPartitionSupportEnabled

```
public void setMultiPartitionSupportEnabled(boolean mptEnabled)
```

Set to true to retrieve a connection with a [Session](#) that has the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

Parameters:

mptEnabled - set to true if the Session should be configured to write to multiple partitions.

Since:
8.6, XC10 2.5

isMultiPartitionSupportEnabled

public boolean **isMultiPartitionSupportEnabled**()

Answers true if the [Session](#) will have the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

Returns:
true if the Session is capable of writing to multiple partitions.

Since:
8.6, XC10 2.5

clone

public [Object](#) **clone**()

Overrides:
[clone](#) in class [Object](#)

hashCode

public int **hashCode**()

Overrides:
[hashCode](#) in class [Object](#)

equals

public boolean **equals**([Object](#) obj)

Overrides:
[equals](#) in class [Object](#)

toString

public [String](#) **toString**()

Overrides:
[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSResourceAdapter

[java.lang.Object](#)



All Implemented Interfaces:

[ResourceAdapter](#)

```

public final class XSResourceAdapter
extends Object
implements ResourceAdapter
    
```

The eXtreme Scale resource adapter

Since:

8.5, XC10

Constructor Summary

[XSResourceAdapter\(\)](#)

Method Summary

V O I D	<p>endpointActivation(MessageEndpointFactory msgEndpointFactory, ActivationSpec actSpec) This method does nothing as endpoint activation is not supported</p>
------------------	--

V O I D	<p>endpointDeactivation(MessageEndpointFactory msgEndpointFactory, ActivationSpec spec) This method does nothing as endpoint deactivation is not supported</p>
------------------	---

X A R E S O U R C E []	<p>getXAResources(ActivationSpec[] spec) This method returns null as the XA protocol is not supported</p>
--	--

V O I D	<p>start(BootstrapContext ctx) This method initializes the eXtreme Scale resource adapter instance</p>
------------------	---

V O	<p>stop()</p>
--------	-------------------------------

i d	This method performs the shutdown of the eXtreme Scale resource adapter instance
S t r i n g	toString()

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSResourceAdapter

```
public XSResourceAdapter()
```

Method Detail

endpointActivation

```
public void endpointActivation(MessageEndpointFactory msgEndpointFactory,  
                               ActivationSpec actSpec)  
    throws ResourceException
```

This method does nothing as endpoint activation is not supported

Specified by:

[endpointActivation](#) in interface [ResourceAdapter](#)

Throws:

[ResourceException](#)

See Also:

[ResourceAdapter.endpointActivation\(MessageEndpointFactory,ActivationSpec\)](#)

endpointDeactivation

```
public void endpointDeactivation(MessageEndpointFactory msgEndpointFactory,  
                                 ActivationSpec spec)
```

This method does nothing as endpoint deactivation is not supported

Specified by:

[endpointDeactivation](#) in interface [ResourceAdapter](#)

Throws:

[ResourceException](#)

See Also:

[ResourceAdapter.endpointDeactivation\(MessageEndpointFactory,ActivationSpec\)](#)

getXAResources

```
public XAResource[] getXAResources(ActivationSpec[] spec)  
    throws ResourceException
```

This method returns null as the XA protocol is not supported

Specified by:

[getXAResources](#) in interface [ResourceAdapter](#)

Throws:

[ResourceException](#)

See Also:

[ResourceAdapter.getXAResources\(ActivationSpec\[\]\)](#)

start

```
public void start(BootstrapContext ctx)
    throws ResourceAdapterInternalException
```

This method initializes the eXtreme Scale resource adapter instance

Specified by:

[start](#) in interface [ResourceAdapter](#)

Throws:

[ResourceAdapterInternalException](#)

See Also:

[ResourceAdapter.start\(BootstrapContext\)](#)

stop

```
public void stop()
```

This method performs the shutdown of the eXtreme Scale resource adapter instance

Specified by:

[stop](#) in interface [ResourceAdapter](#)

See Also:

[ResourceAdapter.stop\(\)](#)

toString

```
public String toString()
```

Overrides:

[toString](#) in class [Object](#)

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help
PREV CLASS	NEXT CLASS	FRAMES		NO FRAMES	All Classes	

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSConnection

[java.lang.Object](#)



All Implemented Interfaces:

[Connection](#)

```

public final class XSConnection
extends Object
implements Connection
  
```

An XSConnection represents an application level connection handle to an eXtreme Scale ObjectGrid.

Since:

8.5, XC10

Constructor Summary

[XSConnection\(\)](#)

Method Summary

V [close\(\)](#)
Initiates close of the connection handle at the application level.

I [createInteraction\(\)](#)
Creates an Interaction associated with this Connection.

L [getLocalTransaction\(\)](#)
Returns an LocalTransaction instance that enables a component to demarcate local transactions on the Connection.

t
i
o
n

C
o
n
n
e
c
t
i
o
n
M
e
t
a
D
a
t
a

[getMetaData\(\)](#)

Gets the information on the underlying eXtreme Scale instance represented through an active connection.

R
e
s
u
l
t
S
e
t
I
n
f
o

[getResultSetInfo\(\)](#)

Gets the information on the ResultSet functionality supported by a connected eXtreme Scale instance.

S
e
s
s
i
o
n

[getSession\(\)](#)

Retrieve the ObjectGrid Session associated with this connection.

S
t
r
i
n
g

[toString\(\)](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSConnection

```
public XSConnection()
```

Method Detail

getSession

```
public Session getSession()  
    throws ResourceException
```

Retrieve the ObjectGrid Session associated with this connection.

Returns:

The ObjectGrid Session

Throws:

[ResourceException](#)

close

```
public void close()  
    throws ResourceException
```

Initiates close of the connection handle at the application level.

Specified by:

[close](#) in interface [Connection](#)

Throws:

[ResourceException](#)

See Also:

[Connection.close\(\)](#)

createInteraction

```
public Interaction createInteraction()  
    throws ResourceException
```

Creates an Interaction associated with this Connection.

Specified by:

[createInteraction](#) in interface [Connection](#)

Returns:

Interaction instance

Throws:

[ResourceException](#)

[NotSupportedException](#)

See Also:

[Connection.createInteraction\(\)](#)

getLocalTransaction

```
public LocalTransaction getLocalTransaction()  
    throws ResourceException
```

Returns an LocalTransaction instance that enables a component to demarcate local transactions on the Connection.

Specified by:

[getLocalTransaction](#) in interface [Connection](#)

Returns:

LocalTransaction instance

Throws:

[ResourceException](#)

See Also:

[Connection.getLocalTransaction\(\)](#)

getMetaData

```
public ConnectionMetaData getMetaData()  
    throws ResourceException
```

Gets the information on the underlying eXtreme Scale instance represented through an active connection.

Specified by:

[getMetaData](#) in interface [Connection](#)

Returns:

WXS ConnectionMetaData

Throws:

[ResourceException](#)

See Also:

[Connection.getMetaData\(\)](#)

getResultSetInfo

```
public ResultSetInfo getResultSetInfo()  
    throws ResourceException
```

Gets the information on the ResultSet functionality supported by a connected eXtreme Scale instance.

Specified by:

[getResultSetInfo](#) in interface [Connection](#)

Returns:

ResultSetInfo instance

Throws:

[ResourceException](#)

See Also:

[Connection.getResultSetInfo\(\)](#)

toString

```
public String toString()
```

Overrides:

[toString](#) in class [Object](#)

[Overvi](#) [Packa](#) [Cla](#) [TreeSerializ](#) [Depreca](#) [IndexHelp](#)
[ew](#) [ge](#) [ss](#) [ed](#) [ted](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSConnectionFactory

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#), [Referenceable](#), [ConnectionFactory](#)

```
public final class XSConnectionFactory
extends Object
implements ConnectionFactory
```

The XSConnectionFactory creates connections to eXtreme Scale ObjectGrids.

Usage example:

```
InitialContext ctx = new InitialContext();
XSConnectionFactory cf = (XSConnectionFactory) ctx.lookup("java:comp/env/wxsconnection");
XSConnection con = cf.getConnection("MyGrid");
Session ogSession = con.getSession();
...

con.close();
```

For additional examples, see the [package documentation](#).

Since:

8.5, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[XSConnectionFactory\(\)](#)

Method Summary

[getCatalogServiceDomain\(\)](#)
Get the eXtreme Scale specific property: CatalogDomain

[getCatalogServiceEndpoints\(\)](#)
Get the eXtreme Scale specific property: CatalogServiceEndpoints

g
S
t
r
i
n
g
S
t
r
i
n
g
C
o
n
n
e
c
t
i
o
n
C
o
n
n
e
c
t
i
o
n
C
o
n
n
e
c
t
i
o
n
S
t
r
i
n
g
R
e
s
o
u
r
c
e
A
d
a
p
t
e
r
M
e
t
a

[getClientPropertiesResource\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesResource

[getClientPropertiesURL\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesURL

[getConnection\(\)](#)

Retrieve an XSConnection for the ObjectGrid configured on the ConnectionFactory.

[getConnection\(ConnectionSpec spec\)](#)

Retrieve an XSConnection for the ObjectGrid using the specified XSConnectionSpec properties.

[getConnection\(String gridName\)](#)

Retrieve an XSConnection for the specified ObjectGrid name.

[getConnectionName\(\)](#)

Get the eXtreme Scale specific property: ConnectionName

[getMetaData\(\)](#)

Retrieve the metadata information regarding the eXtreme Scale resource adapter

Data	
String	<p>getObjectGridName() Get the eXtreme Scale specific property: ObjectGridName</p>
String	<p>getObjectGridResource() Get the eXtreme Scale specific property: ObjectGridResource</p>
String	<p>getObjectGridURL() Get the eXtreme Scale specific property: ObjectGridURL</p>
RecordFactory	<p>getRecordFactory() Retrieve the RecordFactory instance</p>
Reference	<p>getReference() Gets the Reference instance</p>
boolean	<p>isLocalGrid() Get the eXtreme Scale specific property: LocalGrid</p>
void	<p>setReference(Reference ref) Sets the Reference instance</p>
String	<p>toString()</p>

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSConnectionFactory

```
public XSConnectionFactory()
```

Method Detail

setReference

```
public void setReference(Reference ref)
```

Sets the Reference instance

Parameters:

ref - - Reference instance

See Also:

[Referenceable.setReference\(Reference\)](#)

getReference

```
public Reference getReference()  
    throws NamingException
```

Gets the Reference instance

Specified by:

[getReference](#) in interface [Referenceable](#)

Returns:

Reference instance

Throws:

[NamingException](#)

See Also:

[Referenceable.getReference\(\)](#)

getConnection

```
public Connection getConnection()  
    throws ResourceException
```

Retrieve an XSConnection for the ObjectGrid configured on the ConnectionFactory.

Specified by:

[getConnection](#) in interface [ConnectionFactory](#)

Returns:

the connection to the eXtreme Scale ObjectGrid

Throws:

[ResourceException](#)

See Also:

[ConnectionFactory.getConnection\(\)](#)

getConnection

```
public Connection getConnection(ConnectionSpec spec)
    throws ResourceException
```

Retrieve an XSConnection for the ObjectGrid using the specified XSConnectionSpec properties.

Specified by:

[getConnection](#) in interface [ConnectionFactory](#)

Parameters:

spec - the [XSConnectionSpec](#) properties used to retrieve the correct connection.

Returns:

the connection to the eXtreme Scale ObjectGrid

Throws:

[ResourceException](#)

See Also:

[ConnectionFactory.getConnection\(ConnectionSpec\)](#), [XSConnectionSpec](#)

getConnection

```
public Connection getConnection(String gridName)
    throws ResourceException
```

Retrieve an XSConnection for the specified ObjectGrid name.

Parameters:

gridName - - The ObjectGrid name

Returns:

the connection to the eXtreme Scale ObjectGrid

Throws:

[ResourceException](#)

getMetaData

```
public ResourceAdapterMetaData getMetaData()
    throws ResourceException
```

Retrieve the metadata information regarding the eXtreme Scale resource adapter

Specified by:

[getMetaData](#) in interface [ConnectionFactory](#)

Returns:

The eXtreme Scale resource adapter metadata

Throws:

[ResourceException](#)

See Also:

[ConnectionFactory.getMetaData\(\)](#)

getRecordFactory

```
public RecordFactory getRecordFactory()
    throws ResourceException
```

Retrieve the RecordFactory instance

Specified by:

[getRecordFactory](#) in interface [ConnectionFactory](#)

Throws:

[NotSupportedException](#)

[ResourceException](#)

See Also:

[ConnectionFactory.getRecordFactory\(\)](#)

toString

public [String](#) toString()

Overrides:

[toString](#) in class [Object](#)

getCatalogServiceDomain

public [String](#) getCatalogServiceDomain()

Get the eXtreme Scale specific property: CatalogDomain

Returns:

The catalog service domain name defined in WebSphere Application Server

getCatalogServiceEndpoints

public [String](#) getCatalogServiceEndpoints()

Get the eXtreme Scale specific property: CatalogServiceEndpoints

Returns:

The catalog service domain end points

getClientPropertiesResource

public [String](#) getClientPropertiesResource()

Get the eXtreme Scale specific property: ClientPropertiesResource

Returns:

The resource path of the client properties file

getClientPropertiesURL

public [String](#) getClientPropertiesURL()

Get the eXtreme Scale specific property: ClientPropertiesURL

Returns:

The URL of the client properties file

getConnectionName

public [String](#) getConnectionName()

Get the eXtreme Scale specific property: ConnectionName

Returns:

The name of the eXtreme Scale client connection

getObjectGridName

public [String](#) getObjectGridName()

Get the eXtreme Scale specific property: ObjectGridName

Returns:

The data grid name

getObjectGridResource

public [String](#) getObjectGridResource()

Get the eXtreme Scale specific property: ObjectGridResource

Returns:

The resource path of the client data grid override XML file

getObjectGridURL

public [String](#) getObjectGridURL()

Get the eXtreme Scale specific property: ObjectGridURL

Returns:

The URL of the client data grid override XML file

isLocalGrid

public boolean isLocalGrid()

Get the eXtreme Scale specific property: LocalGrid

Returns:

Is this managed connection factory used only for access to a local ObjectGrid instance?

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Interface ObjectGridJ2CConnectionMBean

public interface **ObjectGridJ2CConnectionMBean**

Allows administration clients control the lifecycle of the eXtreme Scale resource adapter's connection.

The object name pattern for this MBean is:

com.ibm.websphere.objectgrid:type=ObjectGridJ2CConnection,objectGridName=<objectgrid>,domain=<domain name>,connectionName=<connection name>

Note: Additional properties may be included.

Since:

8.5, XC10

Field Summary	
s t a t i c S t r i n g	<p>CONNECTIONNAME_DEFAULT The default connection name prefix when not specified.</p>
s t a t i c S t r i n g	<p>CONNECTIONSTATUS_CONNECTED The connection status indicating that a connection has been established.</p>
s t a t i c S t r i n g	<p>CONNECTIONSTATUS_DISCONNECTED The connection status indicating that client is no longer connected.</p>
s	

t
a
t
i
c
s
t
r
i
n
g

MBEAN_TYPE

The MBean type

Method Summary

S
t
r
i
n
g

getCatalogServiceEndpoints()

The catalog service endpoints of the catalog service.

S
t
r
i
n
g

getConnectionName()

The name of the connection as specified on the J2C ConnectionFactory, or "DEFAULT" if not specified.

S
t
r
i
n
g

getConnectionStatus()

The status of the connection.

S
t
r
i
n
g

getDomainName()

The domain name of the catalog service domain as reported by the catalog service.

S
t
r
i
n
g

getObjectGridName()

The name of the ObjectGrid that is connected.

V
o
i
d

resetObjectGridConnection()

Reset the ObjectGrid connection.

Field Detail

MBEAN_TYPE

static final [String](#) MBEAN_TYPE

The MBean type

See Also:

[Constant Field Values](#)

CONNECTIONSTATUS_CONNECTED

static final [String](#) CONNECTIONSTATUS_CONNECTED

The connection status indicating that a connection has been established.

See Also:

[Constant Field Values](#)

CONNECTIONSTATUS_DISCONNECTED

static final [String](#) CONNECTIONSTATUS_DISCONNECTED

The connection status indicating that client is no longer connected.

See Also:

[Constant Field Values](#)

CONNECTIONNAME_DEFAULT

static final [String](#) CONNECTIONNAME_DEFAULT

The default connection name prefix when not specified.

See Also:

[Constant Field Values](#)

Method Detail

resetObjectGridConnection

void `resetObjectGridConnection()`

Reset the ObjectGrid connection.

This destroys the client connection to the ObjectGrid, including any local cache that may be created.

Subsequent uses of the ManagedConnectionFactory will result in a new ObjectGrid connection.

getObjectGridName

[String](#) `getObjectGridName()`

The name of the ObjectGrid that is connected.

Returns:

the ObjectGrid name

getConnectionName

[String](#) `getConnectionName()`

The name of the connection as specified on the J2C ConnectionFactory, or "DEFAULT" if not specified. If there are multiple connections with the same attributes in the same process, the "DEFAULT" name will have an integer appended.

Returns:
the connection name.

getDomainName

[String](#) getDomainName()

The domain name of the catalog service domain as reported by the catalog service.

Returns:
the catalog service domain name.

getCatalogServiceEndpoints

[String](#) getCatalogServiceEndpoints()

The catalog service endpoints of the catalog service.

Returns:
the catalog service endpoints

getConnectionStatus

[String](#) getConnectionStatus()

The status of the connection. Valid states include "CONNECTED" or "DISCONNECTED".
Other states may be returned in the future.

Returns:
one of the CONNECTIONSTATUS constants defined in this interface.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)
PREV CLASS [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package com.ibm.websphere.xsa

Class Summary

RestValue	Deprecated. Use the <i>RestValue</i> interface and the <i>RestValueFactory</i> .
---------------------------	---

com.ibm.websphere.xsa

Class RestValue

[java.lang.Object](#)



All Implemented Interfaces:

com.ibm.websphere.xs.rest.RestValue, [Externalizable](#), [Serializable](#)

Deprecated. Use the *RestValue* interface and the *RestValueFactory*.

```

public class RestValue
extends Object
implements com.ibm.websphere.xs.rest.RestValue, Externalizable
  
```

This object is used in the XC10 REST Gateway. It is a data wrapper for data to be accessed via XC10 REST Gateway. With its wrapped data, it will be stored in an object grid's map entry, with a String as the key of the map entry.

Since:

8.6, XC10 1.0.0.4

See Also:

[Serialized Form](#)

Field Summary

p r o t e c t e d	compressed Deprecated. Flag to indicate if the value is compressed or not
p r o t e c t e d	contentType Deprecated. Content type of the contained value

[S](#)
[t](#)
[r](#)
[i](#)
[n](#)

g	
p r o t e c t e d	value Deprecated. Contained value
b y t e []	

Constructor Summary

RestValue () Deprecated. Constructor
RestValue (byte[] value, String contentType) Deprecated. Constructor

Method Summary

b o o l e a n	equals (Object obj) Deprecated.
S t r i n g	getContentType () Deprecated. Return content type of the contained value
b y t e []	getValue () Deprecated. Return contained value
i n t	hashCode () Deprecated.
b o o l e a n	isCompressed () Deprecated. Indicate if the contained value is compressed
v o i d	readExternal (ObjectInput in) Deprecated.
v o	setCompressed (boolean compressed)

i d	Deprecated. Set indicator to indicate if the contained value is compressed
v o i d	setContentType(String contentType) Deprecated. Set content type of the contained value
v o i d	setValue(byte[] value) Deprecated. Set contained value
S t r i n g	toString() Deprecated.
v o i d	writeExternal(ObjectOutput out) Deprecated.

Methods inherited from class [java.lang.Object](#)
[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

value

protected byte[] **value**

Deprecated.
Contained value

contentType

protected [String](#) **contentType**

Deprecated.
Content type of the contained value

compressed

protected boolean **compressed**

Deprecated.
Flag to indicate if the value is compressed or not

Constructor Detail

RestValue

public **RestValue()**

Deprecated.

Constructor

Since:

8.6, XC10 1.0.0.4

RestValue

```
public RestValue(byte[] value,  
                 String contentType)
```

Deprecated.

Constructor

Parameters:

value - A value to be wrapped by RestValue class.

contentType - Content type of the value.

Since:

8.6, XC10 1.0.0.4

Method Detail

hashCode

```
public int hashCode()
```

Deprecated.

Overrides:

[hashCode](#) in class [Object](#)

equals

```
public boolean equals(Object obj)
```

Deprecated.

Overrides:

[equals](#) in class [Object](#)

toString

```
public String toString()
```

Deprecated.

Overrides:

[toString](#) in class [Object](#)

getValue

```
public byte[] getValue()
```

Deprecated.

Return contained value

Specified by:

getValue in interface [com.ibm.websphere.xs.rest.RestValue](#)

Returns:

An array of byte that represents the contained value

Since:

8.6, XC10 1.0.0.4

setValue

```
public void setValue(byte[] value)
```

Deprecated.

Set contained value

Parameters:

value - An array of byte that represents the contained value

Since:

8.6, XC10 1.0.0.4

getContentType

```
public String getContentType()
```

Deprecated.

Return content type of the contained value

Specified by:

getContentType in interface `com.ibm.websphere.xs.rest.RestValue`

Returns:

A content type

Since:

8.6, XC10 1.0.0.4

setContentType

```
public void setContentType(String contentType)
```

Deprecated.

Set content type of the contained value

Parameters:

contentType - A content type

Since:

8.6, XC10 1.0.0.4

readExternal

```
public void readExternal(ObjectInput in)
    throws IOException,
           ClassNotFoundException
```

Deprecated.**Specified by:**

[readExternal](#) in interface [Externalizable](#)

Throws:

[IOException](#)
[ClassNotFoundException](#)

writeExternal

public void writeExternal([ObjectOutput](#) out)
throws [IOException](#)

Deprecated.

Specified by:

[writeExternal](#) in interface [Externalizable](#)

Throws:

[IOException](#)

isCompressed

public boolean isCompressed()

Deprecated.

Indicate if the contained value is compressed

Returns:

true if the contained value is compressed

Since:

8.6, XC10 1.0.0.4

setCompressed

public void setCompressed(boolean compressed)

Deprecated.

Set indicator to indicate if the contained value is compressed

Parameters:

compressed - A boolean value that indicate if the contained value is compressed

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

IBM WebSphere eXtreme Scale Client for .NET API Specification

Contents

- [Namespaces](#)

Namespaces

Namespace

Description

The IBM.WebSphere.Caching namespace is the parent namespace for the the WebSphere eXtreme Scale C# client data access application programming interfaces.

The primary entry point for interacting with the data grid is the [GridManagerFactory](#), which provides access to the [IGridManager](#) and [IGrid](#) interfaces.

[IBM.WebSphere.Caching](#)

The [IGridManager](#) interface provides methods to connect and disconnect from a catalog service domain. It also provides access to IGrid instances from the catalog service domain connections.

The [IGrid](#) interface allows clients to interact with a named data grid using various map interfaces, such as the [IGridMapPessimisticTx TKey, TValue](#) and [IGridMapPessimisticAutoTx TKey, TValue](#).

For additional information about data access APIs, see the [IBM.WebSphere.Caching.Map](#) namespace documentation.

The IBM.WebSphere.Caching.Map namespace includes the data access application programming interfaces. See the [IBM.WebSphere.Caching](#) namespace documentation for a description on how to access a map.

[IBM.WebSphere.Caching.Map](#)

The eXtreme Scale client supports transactional data access to individual maps using automatic and manual transactions. The following maps are available:

- The [IGridMapPessimisticAutoTx TKey, TValue](#) interface provides automatic transactions.
- The [IGridMapPessimisticTx TKey, TValue](#) interface provides manual transaction demarcation.

See each respective interface for programming examples.

[IBM.WebSphere.Caching.Security](#)

The IBM.WebSphere.Caching.Security namespace includes the application programming interfaces specific to security.

The eXtreme Scale client supports transport security and authentication and authorization using the [ICredentialGenerator](#) interface. Security is configured using a client properties file.

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The IBM.WebSphere.Caching namespace is the parent namespace for the the WebSphere eXtreme Scale C# client data access application programming interfaces.

The primary entry point for interacting with the data grid is the [GridManagerFactory](#), which provides access to the [IGridManager](#) and [IGrid](#) interfaces.

The [IGridManager](#) interface provides methods to connect and disconnect from a catalog service domain. It also provides access to IGrid instances from the catalog service domain connections.

The [IGrid](#) interface allows clients to interact with a named data grid using various map interfaces, such as the [IGridMapPessimisticTx TKey, TValue](#) and [IGridMapPessimisticAutoTx TKey, TValue](#).

For additional information about data access APIs, see the [IBM.WebSphere.Caching.Map](#) namespace documentation.

Classes

Class	Description
AvailabilityException	An AvailabilityException exception occurs when a target is not in the correct state to handle a request that it receives.
ClientServerTransactionCallbackException	A ClientServerTransactionCallbackException exception occurs when a method call to the client/server TransactionCallback encounters a remote request problem.
GridConfigurationException	An GridConfigurationException exception occurs when a configuration problem is found. This exception might occur when the configuration specified in the deployment policy, ObjectGrid descriptor, or security descriptor is not correct.
GridException	An GridException exception is the base exception class for all checked exceptions that occur in the product.
GridManagerFactory	The GridManagerFactory is a factory for IGridManager instances, and is the entrypoint for all interactions with the data grid.
GridServerRuntimeException	A GridServerRuntimeException exception is a generic wrapper for exceptions that occur in the server runtime.
LifecycleFailedException	A LifecycleFailedException exception occurs on unexpected lifecycle states.
MixedTransportException	A MixedTransportException is thrown when server and client have mismatched transport. For example, a server is using ORB, but the client is eXtremeIO
NoActiveTransactionException	A NoActiveTransactionException exception indicates that no active transactions exist.
NotReentrantException	A NotReentrantException occurs when a thread tries to run a map operation, such as calling a method on ObjectMap interface, when another thread is already running a map operation for the Session. A Session object can be used by a single thread only to perform concurrent map operations.
OrderedDictionary TKey, TValue	A generic version of the non-generic OrderedDictionary class.
OrderedDictionary TKey	

[TValue Enumerator](#) The enumerator for iterating through the [OrderedDictionary TKey, TValue](#).

[ReplicationVotedToRollbackTransactionException](#)

A [ReplicationVotedToRollbackTransactionException](#) exception occurs when a transaction was rolled back because some or all of the synchronous replicas did not apply the transaction.

[TransactionAlreadyActiveException](#)

A [TransactionAlreadyActiveException](#) exception occurs to indicate that a transaction is already active for the current Session. This exception does not cause the current active transaction to be rolled back, so the [isTransactionActive](#) method returns true.

[TransactionCallbackException](#)

A [TransactionCallbackException](#) exception occurs when a [TransactionCallback](#) method call fails.

[TransactionException](#)

A [TransactionException](#) exception is a general locking exception that indicates something went wrong with a transaction. Use the [isTransactionActive\(\)](#) and [wasTransactionRolledBack\(\)](#) methods to determine whether transaction is still active or was rolled back as a result of this exception.

[TransactionTimeoutException](#)

A [TransactionTimeoutException](#) exception occurs when a transaction exceeds the transaction timeout value that was specified on the [ObjectGrid](#) or [Session](#).

Interfaces

Interface

Description

[ICatalogDomainInfo](#)

Identifies a catalog service domain to be used for connecting to an eXtreme Scale data grid.

[ICatalogDomainManager](#)

The [ICatalogDomainManager](#) is a factory for [ICatalogDomainInfo](#) objects used to connect to a catalog service domain. Use the [CatalogDomainManager](#) to retrieve an [ICatalogDomainManager](#) instance.

[IClientConnectionContext](#)

The handle to a connection to a catalog service domain. An [IClientConnectionContext](#) is returned from the [Connect\(ICatalogDomainInfo\)](#) method when connecting to a data grid. When finished with the data grid, use the [Disconnect\(IClientConnectionContext\)](#) method to disconnect from the data grid.

[IGrid](#)

An [IGrid](#) is the client's access to a named data grid and corresponds to a configured [ObjectGrid](#) in the catalog service domain. An [IGrid](#) is thread safe and should be cached and reused by the application and is valid until the connection is severed. Use an [IGrid](#) to get map instances. Maps are defined on the data grid and have unique names within each configured [ObjectGrid](#).

[IGridManager](#)

Provides methods for connecting to data grids. Use the [GridManagerFactory](#) to obtain an instance to an [IGridManager](#) instance.

[IGridTransaction](#)

Defines the interface for a transaction.

[IOrderedDictionary](#)

Specifies a generic version of the non-generic [IOrderedDictionary](#) interface.

[ITransactionable](#)

Implementors of [ITransactionable](#) provide transaction demarcation semantics.

Enumerations

Enum

ation**Description**

<input type="checkbox"/>	AvailabilityState	Each shard in a distributed data has an associated availability state. This state determines if the shard can process incoming requests.
<input type="checkbox"/>	TxnIsolationLevel	Specifies an enumeration that defines the valid transaction isolation level values.

Remarks

Note: The client API in this namespace and child namespaces will evolve in future releases. Therefore, to avoid problems with incompatible bindings in the future, interfaces should not be directly implemented nor extended unless explicitly noted in the interface documentation.

Examples

The following example illustrates how to connect to a data grid, retrieve an entry from a map, and disconnect from the grid:

[Copy to ClipboardPrint](#)

```
// Retrieve the IGridManager instance. IGridManager gm = GridManagerFactory.GetGridManager(); // Id
```

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An AvailabilityException exception occurs when a target is not in the correct state to handle a request that it receives.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching AvailabilityException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AvailabilityException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AvailabilityException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> AvailabilityException	Initializes a new instance of the AvailabilityException class.
<input type="checkbox"/> AvailabilityException(Exception)	Initializes a new instance of the AvailabilityException class with the specified exception cause.
<input type="checkbox"/> AvailabilityException(String)	Initializes a new instance of the AvailabilityException class with the specified error message.
<input type="checkbox"/> AvailabilityException(String, Exception)	Initializes a new instance of the AvailabilityException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)

<input type="checkbox"/>	NewAvailabilityState	Gets or sets the availability state. This state controls if the shard can process incoming requests.
<input type="checkbox"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> AvailabilityException	Initializes a new instance of the AvailabilityException class.
<input type="checkbox"/> AvailabilityException(Exception)	Initializes a new instance of the AvailabilityException class with the specified exception cause.
<input type="checkbox"/> AvailabilityException(String)	Initializes a new instance of the AvailabilityException class with the specified error message.
<input type="checkbox"/> AvailabilityException(String, Exception)	Initializes a new instance of the AvailabilityException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System Exception

Specifies the exception that caused of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that caused the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AvailabilityException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AvailabilityException](#) type exposes the following members.
Properties

Name	Description
<input type="text"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="text"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="text"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="text"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text"/> NewAvailabilityState	Gets or sets the availability state. This state controls if the shard can process incoming requests.
<input type="text"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException NewAvailabilityState Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the availability state. This state controls if the shard can process incoming requests.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Each shard in a distributed data has an associated availability state. This state determines if the shard can process incoming requests.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Members

Member Name	Description
OFFLINE 0	An AvailabilityState.OFFLINE state indicates that a shard is offline and cannot process requests.
PRELOAD 1	An AvailabilityState.PRELOAD state indicates that a shard is in the preload state. When a shard is in the preload state, the shard allows requests only from a client that is preloading data into the data grid. All other requests are rejected.
ONLINE 2	An AvailabilityState.ONLINE state indicates that a shard is online. A shard that is online is available for processing requests.
QUIESCE 3	An AvailabilityState. QUIESCE state indicates that a shard is in quiesce. Quiesce is a transitional state. Shards that are in the quiesce state are being taken offline.
UNKNOWN 4	An AvailabilityState.UNKNOWN state indicates that the availability state of the shard could not be determined.

See Also

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A ClientServerTransactionCallbackException exception occurs when a method call to the client/server TransactionCallback encounters a remote request problem.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

[IBM.WebSphere.Caching TransactionCallbackException](#)

IBM.WebSphere.Caching ClientServerTransactionCallbackException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ClientServerTransactionCallbackException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerTransactionCallbackException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> ClientServerTransactionCallbackException	Initializes a new instance of the ClientServerTransactionCallbackException class.
<input type="checkbox"/> ClientServerTransactionCallbackException(Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified exception cause.
<input type="checkbox"/> ClientServerTransactionCallbackException(String)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message.
<input type="checkbox"/> ClientServerTransactionCallbackException(String, Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> ClientServerTransactionCallbackException	Initializes a new instance of the ClientServerTransactionCallbackException class.
<input type="checkbox"/> ClientServerTransactionCallbackException(Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified exception cause.
<input type="checkbox"/> ClientServerTransactionCallbackException(String)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message.
<input type="checkbox"/> ClientServerTransactionCallbackException(String, Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException
Constructor (String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerTransactionCallbackException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBase Exception	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObject tData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> Member wiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional
information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerTransactionCallbackException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An GridConfigurationException exception occurs when a configuration problem is found. This exception might occur when the configuration specified in the deployment policy, ObjectGrid descriptor, or security descriptor is not correct.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching GridConfigurationException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridConfigurationException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification




The [GridConfigurationException](#) type exposes the following members.
Constructors





Name	Description
<input type="checkbox"/> GridConfigurationException	Initializes a new instance of the GridConfigurationException class.
<input type="checkbox"/> GridConfigurationException(Exception)	Initializes a new instance of the GridConfigurationException class with a specified exception cause.
<input type="checkbox"/> GridConfigurationException(String)	Initializes a new instance of the GridConfigurationException class with the specified error message.
<input type="checkbox"/> GridConfigurationException(String, Exception)	Initializes a new instance of the GridConfigurationException class with the specified error message and exception cause.

[Back to Top](#)
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)
Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)

	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridConfigurationException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GridConfigurationException	Initializes a new instance of the GridConfigurationException class.
<input type="checkbox"/> GridConfigurationException(Exception)	Initializes a new instance of the GridConfigurationException class with a specified exception cause.
<input type="checkbox"/> GridConfigurationException(String)	Initializes a new instance of the GridConfigurationException class with the specified error message.
<input type="checkbox"/> GridConfigurationException(String, Exception)	Initializes a new instance of the GridConfigurationException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[GridConfigurationException Class](#)
[GridConfigurationException Members](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException Constructor IBM WebSphere™ eXtreme Scale Client for .NET
(Exception) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridConfigurationException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)









See Also

[GridConfigurationException Class](#)[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridConfigurationException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridConfigurationException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An GridException exception is the base exception class for all checked exceptions that occur in the product.

Inheritance Hierarchy

System Object

System Exception

IBM.WebSphere.Caching GridException

[IBM.WebSphere.Caching AvailabilityException](#)

[IBM.WebSphere.Caching GridConfigurationException](#)

[IBM.WebSphere.Caching LifecycleFailedException](#)

[IBM.WebSphere.Caching.Map CacheKeyNotFoundException](#)

[IBM.WebSphere.Caching.Map DuplicateKeyException](#)

[IBM.WebSphere.Caching.Map LoaderException](#)

[IBM.WebSphere.Caching.Map LockException](#)

[IBM.WebSphere.Caching.Map LockStrategyNotSupportedException](#)

[IBM.WebSphere.Caching.Map MultiplePartitionWriteException](#)

[IBM.WebSphere.Caching.Map OptimisticCollisionException](#)

[IBM.WebSphere.Caching.Map ReadOnlyException](#)

[IBM.WebSphere.Caching.Map TargetNotAvailableException](#)

[IBM.WebSphere.Caching.Map UndefinedMapException](#)

[IBM.WebSphere.Caching MixedTransportException](#)

[IBM.WebSphere.Caching NotReentrantException](#)

[IBM.WebSphere.Caching TransactionCallbackException](#)

[IBM.WebSphere.Caching TransactionException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[GridException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [GridException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> GridException	Initializes a new instance of the GridException class.
<input type="checkbox"/> GridException(Exc eption)	Initializes a new instance of the GridException class with a specified exception cause.
<input type="checkbox"/> GridException(Stri ng)	Initializes a new instance of the GridException class with the specified error message.
<input type="checkbox"/> GridException(Stri ng, Exception)	Initializes a new instance of the GridException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBase Exception	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjec tData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> Member wiseClon e	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GridException	Initializes a new instance of the GridException class.
<input type="checkbox"/> GridException(Exc eption)	Initializes a new instance of the GridException class with a specified exception cause.
<input type="checkbox"/> GridException(Stri ng)	Initializes a new instance of the GridException class with the specified error message.
<input type="checkbox"/> GridException(Stri ng, Exception)	Initializes a new instance of the GridException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[GridException Class](#)

[GridException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional
information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [GridException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridException Class](#)









[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [GridException](#) type exposes the following members.

Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The GridManagerFactory is a factory for IGridManager instances, and is the entrypoint for all interactions with the data grid.

Inheritance Hierarchy

System Object

IBM.WebSphere.Caching GridManagerFactory

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridManagerFactory Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory
Members

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridManagerFactory](#) type exposes the following members.

Methods

	Name	Description
	GetGrid	Retrieves the IGridManager singleton instance that will be used by the client process to connect with the ObjectGrid.
	Manager	

[Back to Top](#)

See Also

[GridManagerFactory Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory Methods IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridManagerFactory](#) type exposes the following members.

Methods

	Name	Description
	GetGridManager	Retrieves the IGridManager singleton instance that will be used by the client process to connect with the ObjectGrid.

[Back to Top](#)

See Also

[GridManagerFactory Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory GetGridManager Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the IGridManager singleton instance that will be used by the client process to connect with the ObjectGrid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

The IGridManager singleton

See Also

[GridManagerFactory Class](#)

[GridManagerFactory Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A GridServerRuntimeException exception is a generic wrapper for exceptions that occur in the server runtime.

Inheritance Hierarchy

System Object

System Exception

IBM.WebSphere.Caching GridServerRuntimeException

[IBM.WebSphere.Caching.Security AccessControlException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridServerRuntimeException](#) type exposes the following members.

Constructors

Name	Description
GridServerRuntimeException(String)	Initializes a new instance of the GridServerRuntimeException class.
GridServerRuntimeException(String, String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message and the Java exception class name.
GridServerRuntimeException(String, Exception, String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message, a reference to the inner exception that is the cause of this exception, and the Java exception class name.

[Back to Top](#)

Methods

Name	Description
Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
ToString	Provides a user-readable representation of this GridServerRuntimeException exception. (Overrides Exception ToString .)

[Back to Top](#)

Properties

Name	Description
Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

<input type="text"/>	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text"/>	JavaException ClassName	Identifies the Java Exception class name that this exception identifies.
<input type="text"/>	Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
GridServerRuntime eException(String)	Initializes a new instance of the GridServerRuntimeException class.
GridServerRuntime eException(String , String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message and the Java exception class name.
GridServerRuntime eException(String , Exception, String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message, a reference to the inner exception that is the cause of this exception, and the Java exception class name.

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridServerRuntimeException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

javaExceptionClassName

Type: System String

Specifies the Java exception class name.

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[GridServerRuntimeException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException Constructor IBM WebSphere™ eXtreme Scale Client for .NET
(String, String) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridServerRuntimeException](#) class with the specified error message and the Java exception class name.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

javaExceptionClassName

Type: System String

Specifies the Java exception class name.

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[GridServerRuntimeException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException Constructor
(String, Exception, String)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridServerRuntimeException](#) class with the specified error message, a reference to the inner exception that is the cause of this exception, and the Java exception class name.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

innerException

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

javaExceptionClassName

Type: System String

Specifies the Java exception class name.

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[GridServerRuntimeException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridServerRuntimeException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Provides a user-readable representation of this GridServerRuntimeException exception. (Overrides Exception ToString .)

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException ToString Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Provides a user-readable representation of this [GridServerRuntimeException](#) exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

The user-readable representation of this exception.

Implements

[_Exception ToString](#)

[See Also](#)

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridServerRuntimeException](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/> JavaException ClassName	Identifies the Java Exception class name that this exception identifies.
<input type="checkbox"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException JavaException IBM WebSphere™ eXtreme Scale Client for .NET API Specification
ClassName Property

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Identifies the Java Exception class name that this exception identifies.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Identifies a catalog service domain to be used for connecting to an eXtreme Scale data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICatalogDomainInfo Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo
Members

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainInfo](#) type exposes the following members.

Properties

Name	Description
CatalogSe <input type="checkbox"/> rverAddre s ses	Retrieves the configured catalog service addresses. See the CreateCatalogDomainInfo(String) method for more details of the format of the catalog service address string.

[Back to Top](#)

See Also

[ICatalogDomainInfo Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainInfo](#) type exposes the following members.
Properties

Name	Description
CatalogSe <input type="checkbox"/> rverAddre s ses	Retrieves the configured catalog service addresses. See the CreateCatalogDomainInfo(String) method for more details of the format of the catalog service address string.

[Back to Top](#)

See Also

[ICatalogDomainInfo Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo CatalogServerAddresses Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the configured catalog service addresses. See the [CreateCatalogDomainInfo\(String\)](#) method for more details of the format of the catalog service address string.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICatalogDomainInfo Interface](#)

[ICatalogDomainInfo Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainManager
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The ICatalogDomainManager is a factory for ICatalogDomainInfo objects used to connect to a catalog service domain. Use the [CatalogDomainManager](#) to retrieve an ICatalogDomainMananager instance.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[ICatalogDomainManager Members](#)

[IBM.WebSphere.Caching Namespace](#)


IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainManager](#) type exposes the following members.
Methods

Name	Description
 CreateCatalogDomainInfo	<p>Creates an instance of the ICatalogDomainInfo object for the specified catalog service addresses.</p> <p>The catalog service addresses are expressed in a string of the form:</p> <pre>catalogServerAddresses ::= <catalogServiceEndpoint> [, <catalogServiceEndpoint>]</pre> <pre>catalogServiceEndpoint ::= <hostName> : <listenerPort></pre> <p>hostName ::= The IP address or host name of a catalog service.</p> <p>listenerPort ::= The listener port that the catalog service is configured to use.</p> <p>For example:</p> <pre>com.acme.server1:2809,com.acme.server2:2809</pre>

[Back to Top](#)

See Also

[ICatalogDomainManager Interface](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainManager](#) type exposes the following members.
Methods

Name	Description
CreateCatalogDomainInfo	<p>Creates an instance of the ICatalogDomainInfo object for the specified catalog service addresses.</p> <p>The catalog service addresses are expressed in a string of the form:</p> <pre>catalogServerAddresses ::= <catalogServiceEndpoint> [, <catalogServiceEndpoint>]</pre> <pre>catalogServiceEndpoint ::= <hostName> : <listenerPort></pre> <p>hostName ::= The IP address or host name of a catalog service.</p> <p>listenerPort ::= The listener port that the catalog service is configured to use.</p> <p>For example:</p> <pre>com.acme.server1:2809,com.acme.server2:2809</pre>

[Back to Top](#)

See Also

[ICatalogDomainManager Interface](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainManager CreateCatalogDo IBM WebSphere™ eXtreme Scale Client for
mainInfo Method .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Creates an instance of the ICatalogDomainInfo object for the specified catalog service addresses.

The catalog service addresses are expressed in a string of the form:

catalogServerAddresses ::= <catalogServiceEndpoint> [,<catalogServiceEndpoint>]

catalogServiceEndpoint ::= <hostName> : <listenerPort>

hostName ::= The IP address or host name of a catalog service.

listenerPort ::= The listener port that the catalog service is configured to use.

For example:

com.acme.server1:2809,com.acme.server2:2809

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

catalogServerAddresses

Type: System String

The catalog service addresses.

Return Value

The ICatalogDomainInfo representing the catalog service domain addresses.

See Also

[ICatalogDomainManager Interface](#)

[ICatalogDomainManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IClientConnectionContext
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The handle to a connection to a catalog service domain. An IClientConnectionContext is returned from the [Connect\(ICatalogDomainInfo\)](#) method when connecting to a data grid.

When finished with the data grid, use the [Disconnect\(IClientConnectionContext\)](#) method to disconnect from the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IClientConnectionContext Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IClientConnectionContext
Members


IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IClientConnectionContext](#) type exposes the following members.

Properties

Name	Description
 DomainName	Retrieves the name of the catalog service domain.

[Back to Top](#)

See Also

[IClientConnectionContext Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.


IClientConnectionContext
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IClientConnectionContext](#) type exposes the following members.
Properties

Name	Description
 DomainName	Retrieves the name of the catalog service domain.

[Back to Top](#)

See Also

[IClientConnectionContext Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IClientConnectionContext DomainName Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the name of the catalog service domain.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IClientConnectionContext Interface](#)

[IClientConnectionContext Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid Interface IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An IGrid is the client's access to a named data grid and corresponds to a configured ObjectGrid in the catalog service domain.

An IGrid is thread safe and should be cached and reused by the application and is valid until the connection is severed.

Use an IGrid to get map instances. Maps are defined on the data grid and have unique names within each configured ObjectGrid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [IGrid](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable .)
<input type="checkbox"/> GetGridMapNames	Returns a list of map names that are associated with this data grid.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String)	Returns a map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String, ICredentialGenerator)	Returns a map instance for the specified map using specified credential object (userid password).
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String)	Returns a grid map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String, ICredentialGenerator)	Returns a grid map instance for the specified map using specified credential object (userid password)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> IsSecurityEnabled	Retreives a value that indicates whether data grid security is enabled.
<input type="checkbox"/> Name	Retrieves the data grid name.
<input type="checkbox"/> TransactionTimeout	Gets or sets the default transaction timeout for the data grid.

[Back to Top](#)

See Also

[IGrid Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [IGrid](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable .)
<input type="checkbox"/> GetGridMapNames	Returns a list of map names that are associated with this data grid.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String)	Returns a map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String, ICredentialGenerator)	Returns a map instance for the specified map using specified credential object (userid password).
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String)	Returns a grid map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String, ICredentialGenerator)	Returns a grid map instance for the specified map using specified credential object (userid password)

[Back to Top](#)

See Also

[IGrid Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapNames
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a list of map names that are associated with this data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Returns a list of map names.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticAutoTx IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String)	Returns a map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String, ICredentialGenerator)	Returns a map instance for the specified map using specified credential object (userid password).

[Back to Top](#)

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticAutoTx TKey, TValue Method (String)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map instance for the specified map.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

mapName

Type: System String

Specifies the map name.

Type Parameters

TKey

Specifies a generic type key.

TValue

Specifies a generic type Value.

Return Value

An IGridMapPessimisticAutoTx<TKey,TValue> instance

Exceptions

Exception	Condition
-----------	-----------

[IBM.WebSphere.Caching.MapLockStrategyNotSupportedException](#)

Occurs when the locking strategy for mapName is not supported by this map interface.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticAutoTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticAutoTx TKey, TValue
Method (String, ICredentialGenerator)

IBM WebSphere™ eXtreme Scale
Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map instance for the specified map using specified credential object (userid password).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

mapName

Type: System String

Specifies the map name.

credentialGenerator

Type: [IBM.WebSphere.Caching.Security ICredentialGenerator](#)

Specifies the Credential Generator object.

Type Parameters

TKey

Specifies a generic type key.

TValue

Specifies a generic type Value.

Return Value

An IGridMapPessimisticAutoTx<TKey,TValue> instance

Exceptions

Exception	Condition
IBM.WebSphere.Caching.Map LockStrategyNotSupportedException	Occurs when the locking strategy for mapName is not supported by this map interface.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticAutoTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticTx
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String)	Returns a grid map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String, ICredentialGenerator)	Returns a grid map instance for the specified map using specified credential object (userid password)

[Back to Top](#)

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticTx TKey,
TValue Method (String)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a grid map instance for the specified map.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

mapName

Type: System String

mapName is the name of the specified map

Type Parameters

TKey

Generic type key.

TValue

Generic Type Value.

Return Value

An IGridMapPessimisticTx<TKey,TValue> instance

Exceptions

Exception	Condition
-----------	-----------

IBM.WebSphere.Caching.MapLockStrategyNotSupportedException	Occurs when the locking strategy for mapName is not configured for pessimistic locking.
--	---

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticTx TKey, TValue
Method (String, ICredentialGenerator)

IBM WebSphere™ eXtreme Scale Client
for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a grid map instance for the specified map using specified credential object (userid password)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

mapName

Type: System String

mapName is the name of the specified map

credentialGenerator

Type: [IBM.WebSphere.Caching.Security ICredentialGenerator](#)

Specifies the Credential Generator object.

Type Parameters

TKey

Generic type key.

TValue

Generic Type Value.

Return Value

An IGridMapPessimisticTx<TKey,TValue> instance

Exceptions

Exception	Condition
IBM.WebSphere.Caching.Map LockStrategyNotSupportedException	Occurs when the locking strategy for mapName is not configured for pessimistic locking.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)




IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [IGrid](#) type exposes the following members.

Properties

	Name	Description
	IsSecurityEnabled	Retrieves a value that indicates whether data grid security is enabled.
	Name	Retrieves the data grid name.
	TransactionTimeout	Gets or sets the default transaction timeout for the data grid.

[Back to Top](#)

See Also

[IGrid Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid IsSecurityEnabled
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retreives a value that indicates whether data grid security is enabled.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

true if data grid security is enabled; otherwise, false.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid Name Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the data grid name.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The data grid name.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid TransactionTimeout
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the default transaction timeout for the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The default transaction timeout value for the data grid.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Interface IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Provides methods for connecting to data grids. Use the GridManagerFactory to obtain an instance to an IGridManager instance.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Examples

The following example illustrates how to retrieve an IGridManager and disconnect:

[Copy to ClipboardPrint](#)

```
// Retrieve the singleton IGridManager instance.
IGridManager gm = GridManagerFactory.GetGridManager();

// Retrieve and cache the grid, and retrieve and update data....
...

// Disconnect from the data grid when all done and
// null out any references.
gm.Disconnect(ccc);
grid = null;
ccc = null;
```

See Also

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [IGridManager](#) type exposes the following members.

Methods

Name	Description
Connect(ICatalogDomainInfo)	<p>Connects a client process to the data grid.</p> <p>Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.</p>
Connect(ICatalogDomainInfo, String)	<p>Connects a client process to the data grid using connection properties that are specified in the the configFilename.</p> <p>Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.</p>
Disconnect	<p>Disconnects a client process from a catalog server. The IClientConnectionContext and associated IGrid instances are invalid once disconnected.</p>
GetGrid	<p>Returns a client data grid instance that corresponds to the data grid that is identified by the specified name. The IGrid is thread safe and should be cached for the life of the application.</p> <p>A grid name cooresponds to an ObjectGrid name in the data grid configuration.</p>

[Back to Top](#)

Properties

Name	Description
CatalogDomainManager	<p>Retrieves the CatalogDomainManager, which is used for creating ICatalogDomainInfo objects.</p>

[Back to Top](#)

See Also

[IGridManager Interface](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [IGridManager](#) type exposes the following members.

Methods

Name	Description
Connect(ICatalogDomainInfo)	Connects a client process to the data grid. Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.
Connect(ICatalogDomainInfo, String)	Connects a client process to the data grid using connection properties that are specified in the the configFilename. Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.
Disconnect	Disconnects a client process from a catalog server. The IClientConnectionContext and associated IGrid instances are invalid once disconnected.
GetGrid	Returns a client data grid instance that corresponds to the data grid that is identified by the specified name. The IGrid is thread safe and should be cached for the life of the application. A grid name cooresponds to an ObjectGrid name in the data grid configuration.

[Back to Top](#)

See Also

[IGridManager Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
Connect(ICatalogDomainInfo)	Connects a client process to the data grid. Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.
Connect(ICatalogDomainInfo, String)	Connects a client process to the data grid using connection properties that are specified in the the configFilename. Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.

[Back to Top](#)

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Connect Method
(ICatalogDomainInfo)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Connects a client process to the data grid.

Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the [Disconnect\(IClientConnectionContext\)](#) method should be invoked to disconnect from the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

catalogDomainInfo

Type: [IBM.WebSphere.Caching ICatalogDomainInfo](#)

Specifies the CatalogDomainInfo object that contains target catalog service domain information.

Return Value

Returns an IClientConnectionContext object that represents a handle to the data grid to which the client is connected.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a nullcatalogDomainInfo is specified.
IBM.WebSphere.Caching GridException	Occurs if there is an error connecting to the specified catalog domain.

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[Connect Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Connect Method
(ICatalogDomainInfo, String)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Connects a client process to the data grid using connection properties that are specified in the the configFilename.

Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the [Disconnect\(IClientConnectionContext\)](#) method should be invoked to disconnect from the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

catalogDomainInfo

Type: [IBM.WebSphere.Caching ICatalogDomainInfo](#)

Specifies the CatalogDomainInfo object that contains target catalog service domain information.

configFilename

Type: System String

Specifies the full path and file name of the client configuration file or null to use the default configuration.

Return Value

Returns an IClientConnectionContext object that represents a handle to the data grid to which the client is connected.

Exceptions

Exception	Condition
System ArgumentException	Occurs when a nullcatalogDomainInfo is specified.
IBM.WebSphere.Caching GridException	Occurs if there is an error connecting to the specified catalog domain.
System.IO FileNotFoundException	Occurs if configFilename does not exist.

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[Connect Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Disconnect
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Disconnects a client process from a catalog server. The IClientConnectionContext and associated IGrid instances are invalid once disconnected.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

context

Type: [IBM.WebSphere.Caching IClientConnectionContext](#)

Specifies the IClientConnectionContext object that was returned from a previous Connect method call.

Return Value

Returns true if the disconnect was successful, or false if the supplied context was not connected.

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager GetGrid
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a client data grid instance that corresponds to the data grid that is identified by the specified name. The IGrid is thread safe and should be cached for the life of the application.

A grid name corresponds to an ObjectGrid name in the data grid configuration.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

context

Type: [IBM.WebSphere.Caching IClientConnectionContext](#)

Specifies the IClientConnectionContext object returned from a Connect method call.

gridName

Type: System String

Specifies the name of the requested ObjectGrid.

Return Value

Returns client data grid instance that corresponds to the specified remote data grid.

Exceptions

Exception	Condition
-----------	-----------

IBM.WebSphere.Caching GridException	Occurs if the specified grid is not found or incompatible with the current configuration.
---	---

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [IGridManager](#) type exposes the following members.

Properties

Name	Description
<input type="text" value="CatalogDomainManager"/>	Retrieves the CatalogDomainManager, which is used for creating ICatalogDomainInfo objects.

[Back to Top](#)

See Also

[IGridManager Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager CatalogDomainManager Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the CatalogDomainManager, which is used for creating [ICatalogDomainInfo](#) objects.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Defines the interface for a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.





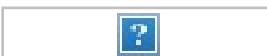
[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification





The [IGridTransaction](#) type exposes the following members.

Methods

Name	Description
 Begin	Begins a new transaction.
 Commit	Commits a transaction.
 Flush	Apply the current changes to the server without committing the transaction.
 MarkRollbackOnly	Marks the current transaction as being rollback only.
 Rollback	Rolls back a transaction.

[Back to Top](#)

Properties

Name	Description
 Active	Determines if a transaction is currently active.
 MarkedRollbackOnly	Determines if a transaction is roll back only.
 TransactionIsolationLevel	Gets or sets the transaction isolation level.
 TransactionTimeout	Gets or sets the amount of time in which a transaction must complete. To use an unlimited transaction timeout value, set the value to TimeSpan.Zero

[Back to Top](#)

See Also

[IGridTransaction Interface](#)

[IBM.WebSphere.Caching Namespace](#)






IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridTransaction](#) type exposes the following members.
Methods

	Name	Description
	Begin	Begins a new transaction.
	Commit	Commits a transaction.
	Flush	Apply the current changes to the server without committing the transaction.
	MarkRollbackOnly	Marks the current transaction as being rollback only.
	Rollback	Rolls back a transaction.

[Back to Top](#)

See Also

[IGridTransaction Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Begin
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Begins a new transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Commit
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Commits a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Flush
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Apply the current changes to the server without committing the transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction MarkRollbackOnly IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Marks the current transaction as being rollback only.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

error

Type: System Exception

Specifies the cause of the exception.

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Rollback
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Rolls back a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridTransaction](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Active	Determines if a transaction is currently active.
<input type="checkbox"/> MarkedRollbackOnlyTransaction	Determines if a transaction is roll back only.
<input type="checkbox"/> IsolationLevel	Gets or sets the transaction isolation level.
<input type="checkbox"/> TransactionTimeout	Gets or sets the amount of time in which a transaction must complete. To use an unlimited transaction timeout value, set the value to TimeSpan.Zero

[Back to Top](#)

See Also

[IGridTransaction Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Active
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines if a transaction is currently active.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction MarkedRollbackOnly IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines if a transaction is roll back only.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction TransactionIsolationLevel Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the transaction isolation level.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The transaction isolation level.

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction TransactionTimeout IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the amount of time in which a transaction must complete. To use an unlimited transaction timeout value, set the value to TimeSpan.Zero

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The transaction timeout value.

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IOrderedDictionary TKey, TValue IBM WebSphere™ eXtreme Scale Client for .NET API
Interface Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies a generic version of the non-generic IOrderedDictionary interface.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Type Parameters

TKey

Specifies the key type.

TValue

Specifies the value type.

See Also

[IOrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification






The [IOrderedDictionary TKey, TValue](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Add(T)	Adds an item to the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> Add(TKey, TValue)	Adds an element with the provided key and value to the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> Clear	Removes all items from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> Contains	Determines whether the ICollection T contains a specific value. (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> ContainsKey	Determines whether the IDictionary TKey, TValue contains an element with the specified key. (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> CopyTo	Copies the elements of the ICollection T to an Array, starting at a particular Array index. (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable KeyValuePair TKey , TValue .)
<input type="checkbox"/> GetEnumerator	Returns an enumerator that iterates through a collection. (Inherited from IEnumerable.)
<input type="checkbox"/> Insert	Inserts an item with the specified key and value into the map at the specified index.
<input type="checkbox"/> Remove(TKey)	Removes the element with the specified key from the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> Remove(T)	Removes the first occurrence of a specific object from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> RemoveAt	Removes the item at the specified index from the IOrderedDictionary object.
<input type="checkbox"/> TryGetValue	Gets the value associated with the specified key. (Inherited from IDictionary TKey , TValue .)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Count	Gets the number of elements contained in the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/>  IsReadOnly	Gets a value indicating whether the ICollection T is read-only. (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/>  Item TKey	Gets or sets the element with the specified key. (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/>  Item Int32	Gets or sets the value with the specified index.
<input type="checkbox"/>  Keys	Gets an ICollection T containing the keys of the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)



Values Gets an ICollection T containing the values in the IDictionary TKey, TValue .
(Inherited from IDictionary [TKey](#), [TValue](#) .)

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IOrderedDictionary TKey, TValue](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Add(T)	Adds an item to the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> Add(TKey, TValue)	Adds an element with the provided key and value to the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> Clear	Removes all items from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> Contains	Determines whether the ICollection T contains a specific value. (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> ContainsKey	Determines whether the IDictionary TKey, TValue contains an element with the specified key. (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> CopyTo	Copies the elements of the ICollection T to an Array, starting at a particular Array index. (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable KeyValuePair TKey , TValue .)
<input type="checkbox"/> GetEnumerator	Returns an enumerator that iterates through a collection. (Inherited from IEnumerable.)
<input type="checkbox"/> Insert	Inserts an item with the specified key and value into the map at the specified index.
<input type="checkbox"/> Remove(TKey)	Removes the element with the specified key from the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> Remove(T)	Removes the first occurrence of a specific object from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> RemoveAt	Removes the item at the specified index from the IOrderedDictionary object.
<input type="checkbox"/> TryGetValue	Gets the value associated with the specified key. (Inherited from IDictionary TKey , TValue .)

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)
[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Inserts an item with the specified key and value into the map at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

Specifies the zero-based index where the item should be inserted.

key

Type: [TKey](#)

Specifies the key of the item to insert.

value

Type: [TValue](#)

Specifies the value of the item to insert.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Occurs when index is out of range.
System.ArgumentNullException	Occurs when a null key is specified.
System.ArgumentException	Occurs when key already exists in the IOrderedDictionary object.

See Also

[IOrderedDictionary Interface](#)

[IOrderedDictionary Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the item at the specified index from the IOrderedDictionary object.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System Int32

The zero-based index of the item to remove.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Occurs when index is less than 0 or greater than Count value.

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IOrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)







IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IOrderedDictionary TKey, TValue](#) type exposes the following members.
Properties

Name	Description
 Count	Gets the number of elements contained in the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
 IsReadOnly	Gets a value indicating whether the ICollection T is read-only. (Inherited from ICollection KeyValuePair TKey , TValue .)
 Item TKey	Gets or sets the element with the specified key. (Inherited from IDictionary TKey , TValue .)
 Item Int32	Gets or sets the value with the specified index.
 Keys	Gets an ICollection T containing the keys of the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
 Values	Gets an ICollection T containing the values in the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)
[IBM.WebSphere.Caching Namespace](#)



IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
 Item TKey	Gets or sets the element with the specified key. (Inherited from IDictionary TKey , TValue .)
 Item Int32	Gets or sets the value with the specified index.

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IOrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IOrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET
Property (Int32) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value with the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

Specifies the zero-based index of the value to get or set.

Return Value

The value of the item at the specified index.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Occurs when index is less than 0 or greater than the Count value.

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IOrderedDictionary TKey, TValue Members](#)

[Item Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ITransactionable
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Implementors of ITransactionable provide transaction demarcation semantics.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ITransactionable Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ITransactional
Members


IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ITransactionable](#) type exposes the following members.

Properties

Name	Description
 Transaction	The Transaction instance used to configure and demarcate a transaction.

[Back to Top](#)

See Also

[ITransactionable Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.


ITransactional
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ITransactionable](#) type exposes the following members.
Properties

Name	Description
 Transaction	The Transaction instance used to configure and demarcate a transaction.

[Back to Top](#)

See Also

[ITransactionable Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ITransactionable Transaction Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The Transaction instance used to configure and demarcate a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ITransactionable Interface](#)

[ITransactionable Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LifecycleFailedException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LifecycleFailedException exception occurs on unexpected lifecycle states.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching LifecycleFailedException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LifecycleFailedException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LifecycleFailedException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LifecycleFailedException(String)	Initializes a new instance of the LifecycleFailedException class with the specified error message.
<input type="checkbox"/> LifecycleFailedException(String, Exception)	Initializes a new instance of the LifecycleFailedException class with the specified error message and exception cause.






[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)



StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)



TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[LifecycleFailedException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LifecycleFailedExcepti on(String)	Initializes a new instance of the LifecycleFailedException class with the specified error message.
<input type="checkbox"/> LifecycleFailedExcepti on(String, Exception)	Initializes a new instance of the LifecycleFailedException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LifecycleFailedException Class](#)

[LifecycleFailedException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LifecycleFailedException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LifecycleFailedException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[LifecycleFailedException Class](#)

[LifecycleFailedException Members](#)

[LifecycleFailedException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LifecycleFailedException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LifecycleFailedException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the exception message text.

cause

Type: System.Exception

Specifies the initial cause of the exception that caused this exception to occur.

See Also

[LifecycleFailedException Class](#)

[LifecycleFailedException Members](#)

[LifecycleFailedException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LifecycleFailedException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LifecycleFailedException Class](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LifecycleFailedException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LifecycleFailedException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A MixedTransportException is thrown when server and client have mismatched transport. For example, a server is using ORB, but the client is eXtremeIO

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching MixedTransportException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MixedTransportException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MixedTransportException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> MixedTransportException	Initializes a new instance of the MixedTransportException class.
<input type="checkbox"/> MixedTransportException(String)	Initializes a new instance of the MixedTransportException class with the specified error message.






[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)



StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)



TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[MixedTransportException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> MixedTransportException	Initializes a new instance of the MixedTransportException class.
<input type="checkbox"/> MixedTransportException(String)	Initializes a new instance of the MixedTransportException class with the specified error message.

[Back to Top](#)

See Also

[MixedTransportException Class](#)

[MixedTransportException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MixedTransportException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MixedTransportException Class](#)

[MixedTransportException Members](#)

[MixedTransportException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MixedTransportException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[MixedTransportException Class](#)

[MixedTransportException Members](#)

[MixedTransportException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MixedTransportException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MixedTransportException Class](#)
[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MixedTransportException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MixedTransportException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A NoActiveTransactionException exception indicates that no active transactions exist.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

[IBM.WebSphere.Caching TransactionException](#)

IBM.WebSphere.Caching NoActiveTransactionException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[NoActiveTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> NoActiveTransactionException	Initializes a new instance of the NoActiveTransactionException class.
<input type="checkbox"/> NoActiveTransactionException(Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified exception cause.
<input type="checkbox"/> NoActiveTransactionException(String)	Initializes a new instance of the NoActiveTransactionException class with the specified error message.
<input type="checkbox"/> NoActiveTransactionException(String, Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

Fields

Name	Description
ivTransactionRolledBack	Indicates whether the transaction was rolled back or

not.
(Inherited from [TransactionException.](#))

[Back to Top](#)

Properties

Name	Description
<input type="text" value="Data"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="text" value="HelpLink"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="text" value="HResult"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="text" value="InnerException"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text" value="Message"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text" value="Source"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text" value="StackTrace"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text" value="TargetSite"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> NoActiveTransactionException	Initializes a new instance of the NoActiveTransactionException class.
<input type="checkbox"/> NoActiveTransactionException(Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified exception cause.
<input type="checkbox"/> NoActiveTransactionException(String)	Initializes a new instance of the NoActiveTransactionException class with the specified error message.
<input type="checkbox"/> NoActiveTransactionException(String, Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification
(String, Exception)

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.


[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException Fields IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Fields

Name	Description
 ivTransactionRolledBack	Indicates whether the transaction was rolled back or not. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A NotReentrantException occurs when a thread tries to run a map operation, such as calling a method on ObjectMap interface, when another thread is already running a map operation for the Session. A Session object can be used by a single thread only to perform concurrent map operations.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching NotReentrantException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[NotReentrantException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NotReentrantException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> NotReentrantException	Initializes a new instance of the NotReentrantException class.
<input type="checkbox"/> NotReentrantException(Exception)	Initializes a new instance of the NotReentrantException class with a specified exception cause.
<input type="checkbox"/> NotReentrantException(String)	Initializes a new instance of the NotReentrantException class with the specified error message.
<input type="checkbox"/> NotReentrantException(String, Exception)	Initializes a new instance of the NotReentrantException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception.

		(Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error.
		(Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown.
		(Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception.
		(Inherited from Exception.)

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> NotReentrantException	Initializes a new instance of the NotReentrantException class.
<input type="checkbox"/> NotReentrantException(Exception)	Initializes a new instance of the NotReentrantException class with a specified exception cause.
<input type="checkbox"/> NotReentrantException(String)	Initializes a new instance of the NotReentrantException class with the specified error message.
<input type="checkbox"/> NotReentrantException(String, Exception)	Initializes a new instance of the NotReentrantException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NotReentrantException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NotReentrantException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A generic version of the non-generic OrderedDictionary class.

Inheritance Hierarchy

System Object

IBM.WebSphere.Caching OrderedDictionary TKey, TValue

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Type Parameters

TKey

The key type

TValue

The value type

See Also

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue](#) type exposes the following members.

Constructors

Name	Description
OrderedDictionary TKey, TValue	Initializes a new instance of the OrderedDictionary TKey, TValue class.
OrderedDictionary TKey, TValue (Int32)	Initializes a new instance of the OrderedDictionary TKey, TValue class with the specified capacity.
OrderedDictionary TKey, TValue (IOrderedDictionary TKey, TValue)	Initializes a new instance of the OrderedDictionary TKey, TValue class having its contents copied from the specified IOrderedDictionary TKey, TValue .

[Back to Top](#)

Methods

Name	Description
Add(KeyValuePair TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
Add(TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
Clear	Removes all items from the OrderedDictionary TKey, TValue .
Contains	Determines whether the OrderedDictionary TKey, TValue contains the specified item.
ContainsKey	Determines whether the OrderedDictionary TKey, TValue contains the specified key.
CopyTo(Array, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified single-dimension array at the specified index.
CopyTo(KeyValuePair TKey, TValue, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified array at the specified index.
Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
GetEnumerator	Returns a <code>Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey, TValue>></code> object that iterates through the OrderedDictionary TKey, TValue .
GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
GetType	Gets the Type of the current instance. (Inherited from Object.)
Insert(Int32, KeyValuePair TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
Insert(Int32, TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)

<input type="checkbox"/>	Remove(KeyValuePair TKey, TValue) TValue	Removes the specified item from the OrderedDictionary TKey, TValue .
<input type="checkbox"/>	Remove(TKey)	Removes the item with the specified key from the OrderedDictionary TKey, TValue .
<input type="checkbox"/>	RemoveAt	Removes the item at the specified index from the OrderedDictionary TKey, TValue .
<input type="checkbox"/>	ToString	Returns a String that represents the current Object. (Inherited from Object.)
<input type="checkbox"/>	TryGetValue	Get the value associated with the specified key.

[Back to Top](#)
Properties

Name	Description
<input type="checkbox"/> Count	Gets the number of items in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> IsFixedSize	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue has a fixed size.
<input type="checkbox"/> IsReadOnly	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue is read only.
<input type="checkbox"/> IsSynchronized	Gets a value indicating whether access to this Collection is synchronized (thread safe).
<input type="checkbox"/> Item Int32	Gets or sets the value associated with the specified index.
<input type="checkbox"/> Item TKey	Gets or sets the value associated with the specified key.
<input type="checkbox"/> Keys	Gets a ICollection T object containing all the keys in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> SyncRoot	Gets an object that can be used to synchronize this Collection.
<input type="checkbox"/> Values	Gets a ICollection T object containing all the values in the OrderedDictionary TKey, TValue .

[Back to Top](#)
See Also

[OrderedDictionary TKey, TValue Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> OrderedDictionary TKey, TValue	Initializes a new instance of the OrderedDictionary TKey, TValue class.
<input type="checkbox"/> OrderedDictionary TKey, TValue (Int32)	Initializes a new instance of the OrderedDictionary TKey, TValue class with the specified capacity.
<input type="checkbox"/> OrderedDictionary TKey, TValue (IOrderedDictionary TKey, TValue)	Initializes a new instance of the OrderedDictionary TKey, TValue class having its contents copied from the specified IOrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OrderedDictionary TKey, TValue](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[OrderedDictionary TKey, TValue Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue
Constructor (Int32)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OrderedDictionary TKey, TValue](#) class with the specified capacity.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

capacity

Type: System.Int32

The initial number of elements that [OrderedDictionary TKey, TValue](#) can contain.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[OrderedDictionary TKey, TValue Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Constructor
(IOrderedDictionary TKey, TValue)

IBM WebSphere™ eXtreme Scale Client
for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OrderedDictionary TKey, TValue](#) class having its contents copied from the specified [IOrderedDictionary TKey, TValue](#) .

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

orderedDictionary

Type: [IBM.WebSphere.Caching IOrderedDictionary TKey, TValue](#)

The [IOrderedDictionary TKey, TValue](#) whose contents are copied into the new [OrderedDictionary TKey, TValue](#) .

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[OrderedDictionary TKey, TValue Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Add(KeyValuePair TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Add(TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Clear	Removes all items from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Contains	Determines whether the OrderedDictionary TKey, TValue contains the specified item.
<input type="checkbox"/> ContainsKey	Determines whether the OrderedDictionary TKey, TValue contains the specified key.
<input type="checkbox"/> CopyTo(Array, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified single-dimension array at the specified index.
<input type="checkbox"/> CopyTo(KeyValuePair Pair TKey, TValue , Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified array at the specified index.
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetEnumerator	Returns a Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey,TValue>> object that iterates through the OrderedDictionary TKey, TValue .
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> Insert(Int32, KeyValuePair TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> Insert(Int32, TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> Remove(KeyValue Pair TKey, TValue)	Removes the specified item from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Remove(TKey)	Removes the item with the specified key from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> RemoveAt	Removes the item at the specified index from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)
<input type="checkbox"/> TryGetValue	Get the value associated with the specified key.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Add IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> Add(KeyValuePair TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Add(TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Add Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification
(KeyValuePair TKey, TValue)

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds an item with the specified key and value to the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

kvp

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The key-value pair to add.

Implements

ICollection T Add(T)

Exceptions

Exception	Condition
System ArgumentNullException	Thrown when the kvp key is null.
System.Collections.Generic KeyNotFoundException	Thrown when the kvp key is not found.

Remarks

Use this property to add value in the collection. Values are appended to the end of the collection. A key cannot be null; however a value can be. To update an existing item, use either the [Item Int32](#) or the [Item TKey](#) property.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Add Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Add
Method (TKey, TValue)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds an item with the specified key and value to the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the item to add.

value

Type: [TValue](#)

The value of the item to add.

Implements

IDictionary TKey, TValue Add(TKey, TValue)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentNullException	Thrown when key is null.
------------------------------	--------------------------

Remarks

Use this property to add a new item to the collection. Items are appended to the end of the collection. A key cannot be null; however a value can be. To update an existing item, use either the [Item Int32](#) or the [Item Int32](#) property.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Add Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Clear Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes all items from the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Implements

ICollection T Clear

IDictionary Clear

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Contains Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines whether the [OrderedDictionary TKey, TValue](#) contains the specified item.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

kvp

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The key-value pair to locate in the [OrderedDictionary TKey, TValue](#).

Return Value

true if the [OrderedDictionary TKey, TValue](#) contains the item associated with the specified kvp; otherwise, false.

Implements

ICollection T Contains(T)

Exceptions

Exception	Condition
-----------	-----------

System ArgumentNullException	Thrown when the kvp key is null.
------------------------------	----------------------------------

Remarks

The specified item's key must be contained in the [OrderedDictionary TKey, TValue](#) and it's associated value must be equal to the specified item's value for true to be returned.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue ContainsKey Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines whether the [OrderedDictionary TKey, TValue](#) contains the specified key.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key to locate in the [OrderedDictionary TKey, TValue](#).

Return Value

true if the [OrderedDictionary TKey, TValue](#) contains the item associated with the specified key; otherwise, false.

Implements

IDictionary TKey, TValue ContainsKey(TKey)

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when key is null.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> CopyTo(Array, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified single-dimension array at the specified index.
<input type="checkbox"/> CopyTo(KeyValuePair TKey, TValue , Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified array at the specified index.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue CopyTo
Method (Array, Int32)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Copies the items in the [OrderedDictionary TKey, TValue](#) into the specified single-dimension array at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

array

Type: System Array

The allocated single-dimension array into which the items from [OrderedDictionary TKey, TValue](#) will be copied.

index

Type: System Int32

The index into array where copying will begin.

Implements

ICollection CopyTo(Array, Int32)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentOutOfRangeException

Thrown when index is out of range.

System.ArgumentException

Thrown when array does not have sufficient size to hold the incoming data.

Remarks

The array must have sufficient storage to contain all items from the [OrderedDictionary TKey, TValue](#); otherwise a ArgumentException will be thrown. index must be equal or greater than 0.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[CopyTo Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue CopyTo Method IBM WebSphere™ eXtreme Scale Client
(KeyValuePair TKey, TValue , Int32) for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Copies the items in the [OrderedDictionary TKey, TValue](#) into the specified array at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

array

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The allocated array into which the items from [OrderedDictionary TKey, TValue](#) will be copied.

index

Type: System Int32

The zero-based index into array where copying will begin.

Implements

ICollection T CopyTo(T , Int32)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentOutOfRangeException	Thrown when index is out of range.
------------------------------------	------------------------------------

System.ArgumentException	Thrown when array does not have sufficient size to hold the incoming data.
--------------------------	--

Remarks

The array must have sufficient storage to contain all items from the [OrderedDictionary TKey, TValue](#); otherwise a ArgumentException will be thrown. index must be equal or greater than 0.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[CopyTo Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue GetEnumerator Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a `Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey,TValue>>` object that iterates through the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Return Value

A `Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey,TValue>>` object that iterates through the [OrderedDictionary TKey, TValue](#).

Implements

IEnumerable T GetEnumerator

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> Insert(Int32, KeyValuePair TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> Insert(Int32, TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)
[OrderedDictionary TKey, TValue Members](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Insert Method
(Int32, KeyValuePair TKey, TValue)

IBM WebSphere™ eXtreme Scale Client
for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Inserts an item with the specified key and value into the [OrderedDictionary TKey, TValue](#) at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

index

Type: System.Int32

The zero-based index where the item should be inserted.

kvp

Type: System.Collections.Generic.KeyValuePair [TKey](#), [TValue](#)

The item to insert.

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentNullException	Thrown when the kvp key is null.
System.ArgumentOutOfRangeException	Thrown when index is out of range.

Remarks

The index must be greater than or equal to 0 and the index must be less than the number of elements in the collection; otherwise a `ArgumentOutOfRangeException` is thrown.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Insert Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Insert
Method (Int32, TKey, TValue)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Inserts an item with the specified key and value into the [OrderedDictionary TKey, TValue](#) at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

index

Type: System.Int32

The zero-based index where the item should be inserted.

key

Type: [TKey](#)

The key of the item to insert.

value

Type: [TValue](#)

The value of the item to insert.

Implements

[IOrderedDictionary TKey, TValue Insert\(Int32, TKey, TValue\)](#)

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when the key is null.
System.ArgumentOutOfRangeException	Thrown when index is out of range.

Remarks

The index must be greater than or equal to 0 and the index must be less than the number of elements in the collection; otherwise a `ArgumentOutOfRangeException` is thrown.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Insert Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> Remove(KeyValuePair TKey, TValue)	Removes the specified item from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Remove(TKey)	Removes the item with the specified key from the OrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Remove Method IBM WebSphere™ eXtreme Scale Client
(KeyValuePair TKey, TValue) for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the specified item from the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

kvp

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The key-value pair to remove.

Return Value

true if the item is successfully removed; otherwise false is returned. false is also returned if the item is not found.

Implements

ICollection T Remove(T)

Exceptions

Exception	Condition
-----------	-----------

System ArgumentNullException	Thrown when the kvp key is null.
------------------------------	----------------------------------

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Remove Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Remove IBM WebSphere™ eXtreme Scale Client for .NET
Method (TKey) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the item with the specified key from the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the item to remove.

Return Value

true if the item is successfully removed; otherwise false. false is also returned if the key is not found.

Implements

IDictionary TKey, TValue Remove(TKey)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentNullException	Thrown when key is null.
------------------------------	--------------------------

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Remove Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the item at the specified index from the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

The zero-based index of the item to remove.

Implements

[IOrderedDictionary TKey, TValue RemoveAt\(Int32\)](#)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentOutOfRangeException	Thrown when index is out of range.
------------------------------------	------------------------------------

Remarks

The index must be greater than or equal to 0 and the index must be less than the number of elements in the collection; otherwise a `ArgumentOutOfRangeException` is thrown.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue TryGetValue Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Get the value associated with the specified key.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the value to get.

val

Type: [TValue](#)

The value associated with the specified key.

Return Value

true if the [OrderedDictionary TKey, TValue](#) contained the specified key; otherwise false.

Implements

IDictionary TKey, TValue TryGetValue(TKey, TValue)

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when key is null.

Remarks

If the key is not found, val is initialized to the default value associated with its data type.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Count	Gets the number of items in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> IsFixedSize	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue has a fixed size.
<input type="checkbox"/> IsReadOnly	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue is read only.
<input type="checkbox"/> IsSynchronized	Gets a value indicating whether access to this Collection is synchronized (thread safe).
<input type="checkbox"/> Item Int32	Gets or sets the value associated with the specified index.
<input type="checkbox"/> Item TKey	Gets or sets the value associated with the specified key.
<input type="checkbox"/> Keys	Gets a ICollection T object containing all the keys in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> SyncRoot	Gets an object that can be used to synchronize this Collection.
<input type="checkbox"/> Values	Gets a ICollection T object containing all the values in the OrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Count Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the number of items in the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The number of items in the [OrderedDictionary TKey, TValue](#).

Implements

ICollection T Count

ICollection Count

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue IsFixedSize Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a value that indicates whether or not the [OrderedDictionary TKey, TValue](#) has a fixed size.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Field Value

true if the [OrderedDictionary TKey, TValue](#) has a fixed size; otherwise false.

Implements

IDictionary IsFixedSize

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue IsReadOnly Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a value that indicates whether or not the [OrderedDictionary TKey, TValue](#) is read only.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

true if the [OrderedDictionary TKey, TValue](#) is read only; otherwise false.

Implements

ICollection T IsReadOnly

IDictionary IsReadOnly

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue IsSynchronized Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a value indicating whether access to this Collection is synchronized (thread safe).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

true if this Collection is synchronized.

Implements

ICollection IsSynchronized

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.



[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
 Item Int32	Gets or sets the value associated with the specified index.
 Item TKey	Gets or sets the value associated with the specified key.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET
Property (Int32) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value associated with the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

The zero-based index of the value to get or set.

Field Value

The value of the item at the specified index.

Implements

[IOOrderedDictionary Item Int32](#)

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Thrown when index is less than 0 or greater than the number of elements in the collection.
Remarks	

A set operation will add or replace a value in the collection, with new values being appended to the end of the collection. A key must be greater than or equal to 0. Values may be null.
See Also

[OrderedDictionary Class](#)

[OrderedDictionary Members](#)

[Item Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET
Property (TKey) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value associated with the specified key.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the value to get or set.

Field Value

The value associated with the specified key. If the specified key does not exist during a get operation, a KeyNotFoundException is thrown.

Implements

IDictionary TKey, TValue Item TKey

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when key is null.
System.Collections.Generic.KeyNotFoundException	Thrown when key is not found during a get operation.
Remarks	

A set operation will add or replace a value in the collection, with new values being appended to the end of the collection. A key cannot be null; however a value can be. The key cannot be an int; if it is an int, then the [Item Int32](#) property is used.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Item Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Keys IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a ICollection T object containing all the keys in the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

A ICollection T object containing all the keys in the [OrderedDictionary TKey, TValue](#).

Implements

IDictionary TKey, TValue Keys

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue SyncRoot Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets an object that can be used to synchronize this Collection.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

An object that can be used to synchronize this Collection.

Implements

ICollection SyncRoot

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Values Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a ICollection T object containing all the values in the [OrderedDictionary TKey, TValue](#) .

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

A ICollection T object containing all the values in the [OrderedDictionary TKey, TValue](#) .

Implements

IDictionary TKey, TValue Values

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator Class

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The enumerator for interating through the [OrderedDictionary TKey, TValue](#).

Inheritance Hierarchy

System Object

IBM.WebSphere.Caching OrderedDictionary TKey, TValue Enumerator

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A generic version of the non-generic OrderedDictionary class.

The [OrderedDictionary TKey, TValue Enumerator](#) generic type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> MoveNext	Moves the iterator to the next item in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Reset	Moves the iterator to the first item in OrderedDictionary TKey, TValue .
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Current	Gets the item at the iterator's current position.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue Enumerator](#) generic type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> MoveNext	Moves the iterator to the next item in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Reset	Moves the iterator to the first item in OrderedDictionary TKey, TValue .
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator MoveNext Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Moves the iterator to the next item in the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

true if the enumerator was successfully able to move the next item in the [OrderedDictionary TKey, TValue](#); otherwise, false is returned to indicate that the enumerator is at the end of the [OrderedDictionary TKey, TValue](#).

Implements

IEnumerator MoveNext

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator Reset Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Moves the iterator to the first item in [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Implements

IEnumerator Reset

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator Properties


IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue Enumerator](#) generic type exposes the following members.

Properties

Name	Description
 Current	Gets the item at the iterator's current position.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator Current Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the item at the iterator's current position.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Implements

IEnumerator T Current

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A ReplicationVotedToRollbackTransactionException exception occurs when a transaction was rolled back because some or all of the synchronous replicas did not apply the transaction.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.TransactionCallbackException](#)

IBM.WebSphere.Caching.ReplicationVotedToRollbackTransactionException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ReplicationVotedToRollbackTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [ReplicationVotedToRollbackTransactionException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with a specified exception cause.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String, Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

Overload List

Name	Description
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with a specified exception cause.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String, Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionException IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionExceptio IBM WebSphere™ eXtreme Scale Client for
n Constructor (Exception) .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System Exception

The exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionExcepti IBM WebSphere™ eXtreme Scale Client for
on Constructor (String) .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionException IBM WebSphere™ eXtreme Scale Client
Constructor (String, Exception) for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ReplicationVotedToRollbackTransactionException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBase Exception	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObject tData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> Member wiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional
information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ReplicationVotedToRollbackTransactionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionAlreadyActiveException exception occurs to indicate that a transaction is already active for the current Session. This exception does not cause the current active transaction to be rolled back, so the isTransactionActive method returns true.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

[IBM.WebSphere.Caching TransactionException](#)

IBM.WebSphere.Caching TransactionAlreadyActiveException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionAlreadyActiveException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [TransactionAlreadyActiveException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> TransactionAlreadyActiveException	Initializes a new instance of the TransactionAlreadyActiveException class.
<input type="checkbox"/> TransactionAlreadyActiveException(Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified exception cause.
<input type="checkbox"/> TransactionAlreadyActiveException(String)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message.
<input type="checkbox"/> TransactionAlreadyActiveException(String, Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

Fields

Name	Description
<input type="checkbox"/> ivTransactionRolledBack	Indicates whether the transaction was rolled back or

not.
(Inherited from [TransactionException.](#))

[Back to Top](#)

Properties

Name	Description
<input type="text" value="Data"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="text" value="HelpLink"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="text" value="HResult"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="text" value="InnerException"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text" value="Message"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text" value="Source"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text" value="StackTrace"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text" value="TargetSite"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionAlreadyActiveException	Initializes a new instance of the TransactionAlreadyActiveException class.
<input type="checkbox"/> TransactionAlreadyActiveException(Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified exception cause.
<input type="checkbox"/> TransactionAlreadyActiveException(String)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message.
<input type="checkbox"/> TransactionAlreadyActiveException(String, Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

The error message that explains the reason for the exception.

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException
Constructor (String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

The error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.


[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException IBM WebSphere™ eXtreme Scale Client for .NET API
Fields Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAlreadyActiveException](#) type exposes the following members.
Fields

Name	Description
 ivTransactionRolledBack	Indicates whether the transaction was rolled back or not. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAlreadyActiveException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)
[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAlreadyActiveException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionCallbackException exception occurs when a TransactionCallback method call fails.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching TransactionCallbackException

[IBM.WebSphere.Caching ClientServerTransactionCallbackException](#)

[IBM.WebSphere.Caching ReplicationVotedToRollbackTransactionException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionCallbackException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionCallbackException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> TransactionCallbackException	Initializes a new instance of the TransactionCallbackException class.
<input type="checkbox"/> TransactionCallbackException(Exception)	Initializes a new instance of the TransactionCallbackException class with a specified exception cause.
<input type="checkbox"/> TransactionCallbackException(String)	Initializes a new instance of the TransactionCallbackException class with the specified error message.
<input type="checkbox"/> TransactionCallbackException(String, Exception)	Initializes a new instance of the TransactionCallbackException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc ption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionCallbackException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionCallbackException	Initializes a new instance of the TransactionCallbackException class.
<input type="checkbox"/> TransactionCallbackException(Exception)	Initializes a new instance of the TransactionCallbackException class with a specified exception cause.
<input type="checkbox"/> TransactionCallbackException(String)	Initializes a new instance of the TransactionCallbackException class with the specified error message.
<input type="checkbox"/> TransactionCallbackException(String, Exception)	Initializes a new instance of the TransactionCallbackException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionCallbackException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)









See Also

[TransactionCallbackException Class](#)[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionCallbackException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionCallbackException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionException exception is a general locking exception that indicates something went wrong with a transaction. Use the isTransactionActive() and wasTransactionRolledBack() methods to determine whether transaction is still active or was rolled back as a result of this exception.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching TransactionException

[IBM.WebSphere.Caching NoActiveTransactionException](#)

[IBM.WebSphere.Caching TransactionAlreadyActiveException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> TransactionException on(Exception, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, Boolean)	Initializes a new instance of the TransactionException class with the specified error message and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(TransactionException, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, TransactionException, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, Exception, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed.
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back.

[Back to Top](#)

Fields

	Name	Description
<input type="checkbox"/>	ivTransactionRolledBack	Indicates whether the transaction was rolled back or not.

[Back to Top](#)
Properties

	Name	Description
<input type="checkbox"/>	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/>	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>	Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)
See Also

[TransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionException(Exception, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException(String, Boolean)	Initializes a new instance of the TransactionException class with the specified error message and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException(TransactionException, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException(String, TransactionException, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException(String, Exception, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

[Back to Top](#)

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor
(Exception, Boolean)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor
(String, Boolean)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with the specified error message and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor
(TransactionException, Boolean)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: [IBM.WebSphere.Caching TransactionException](#)

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

Remarks

The cause and a detailed message of (cause==null ? null : cause.toString()) is used, which typically contains the class and detailed message of cause. This constructor is useful for as a wrapper for other Throwable objects that occur.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor (String, TransactionException, Boolean)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: [IBM.WebSphere.Caching TransactionException](#)

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

Remarks

The detailed error message that is associated with the cause is not automatically incorporated in this the detailed message for this TransactionException exception.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor (String, Exception, Boolean) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.


[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Fields IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.
Fields

Name	Description
 ivTransactionRolledBack	Indicates whether the transaction was rolled back or not.

[Back to Top](#)

See Also

[TransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException ivTransactionRoll IBM WebSphere™ eXtreme Scale Client for .NET
edBack Field API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Indicates whether the transaction was rolled back or not.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed.
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back.

[Back to Top](#)

See Also

[TransactionException Class](#)[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException isTransactionActive Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

true if transaction is active, false if transaction never started or was complete.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException wasTransactionRoll IBM WebSphere™ eXtreme Scale Client for .NET
edBack Method API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns true if the transaction was rolled back.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Returns true if transaction was rolled back, false otherwise.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionTimeoutException exception occurs when a transaction exceeds the transaction timeout value that was specified on the ObjectGrid or Session.

Inheritance Hierarchy

System Object

System Exception

IBM.WebSphere.Caching TransactionTimeoutException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionTimeoutException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> TransactionTimeoutException(String, Exception)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and exception cause.
<input type="checkbox"/> TransactionTimeoutException(String, String)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and the transaction that timed out.




[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> getTxIDString	Gets the String representation of the TxID.toString() method for the transaction that timed out.
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> whenOccurred	Returns the time when this TransactionTimeoutException exception was created.

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerExc	Gets the Exception instance that caused the current exception.

ption (Inherited from Exception.)



Message Gets a message that describes the current exception.
(Inherited from Exception.)



Source Gets or sets the name of the application or the object that causes the error.
(Inherited from Exception.)



StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)



TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionTimeoutException(String, Exception)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and exception cause.
<input type="checkbox"/> TransactionTimeoutException(String, String)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and the transaction that timed out.

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionTimeoutException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[TransactionTimeoutException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException Constructor IBM WebSphere™ eXtreme Scale Client for .NET
(String, String) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionTimeoutException](#) class with the specified error message and the transaction that timed out.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

txIdString

Type: System.String

Specifies the result of the TxID.ToString() method for the transaction that timed out.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[TransactionTimeoutException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionTimeoutException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> getTxIDString	Gets the String representation of the TxID.toString() method for the transaction that timed out.
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> whenOccurred	Returns the time when this TransactionTimeoutException exception was created.

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)
[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException getTxIDSt IBM WebSphere™ eXtreme Scale Client for .NET
ring Method API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the String representation of the TxID.toString() method for the transaction that timed out.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Return Value

Specifies the result of TxID.toString() method for the transaction that timed out.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException whenOccu IBM WebSphere™ eXtreme Scale Client for .NET
rred Method API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns the time when this [TransactionTimeoutException](#) exception was created.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies a date object that represents the exact time when this exception object was created.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionTimeoutException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TxnIsolationLevel Enumeration IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies an enumeration that defines the valid transaction isolation level values.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Members

Member name	Value	Description
NotSet	-1	Specifies that the transaction isolation level value is not set.
ReadUncommitted	1	Specifies that dirty reads, non-repeatable, reads and phantom reads can occur.
ReadCommitted	2	Specifies that dirty reads are prevented, but non-repeatable reads and phantom reads can occur.
RepeatableRead	4	Specifies that dirty reads and non-repeatable reads are prevented, but phantom reads can occur.

See Also

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The IBM.WebSphere.Caching.Map namespace includes the data access application programming interfaces. See the [IBM.WebSphere.Caching](#) namespace documentation for a description on how to access a map.

The eXtreme Scale client supports transactional data access to individual maps using automatic and manual transactions. The following maps are available:

- The [IGridMapPessimisticAutoTx TKey, TValue](#) interface provides automatic transactions.
- The [IGridMapPessimisticTx TKey, TValue](#) interface provides manual transaction demarcation.

See each respective interface for programming examples.

Classes

Classes	Description
CacheKeyNotFoundException	A CacheKeyNotFoundException exception occurs if a key cannot be found in the cache.
ClassAlias	Specifies the class alias that you can use to correlate different class names that have the same class alias. The class alias and class fields or types are used to identify a unique class type ID during the object class serialization and de-serialization.
ClassAliasAttribute	The [ClassAlias] annotation can be specified for a user defined class.
ClientServerLoaderException	A ClientServerLoaderException exception is a base exception for any client/server operation exceptions.
DuplicateKeyException	A DuplicateKeyException exception occurs if a key cannot be inserted into the backing map because an object with the same key already exists.
FieldAlias	Specifies the field alias that you can use to correlate different class field names that have the same class field alias. The class alias and field alias or types are used to identify a unique class type ID during the object class serialization and de-serialization.
FieldAliasAttribute	The [FieldAlias] annotation can be specified for user defined class fields.
LoaderException	A LoaderException exception is the base exception that results for any exceptions that are encountered by a Loader.

[on](#)
[Loc](#)
[kDe](#)
[adlo](#)
[ckE](#)
[xce](#)
[ptio](#)
[n](#)

A LockStrategyNotSupportedException exception occurs when the lock manager detects a deadlock. This exception occurs to prevent the deadlock.

[Loc](#)
[kEx](#)
[cept](#)
[ion](#)

A LockException exception indicates errors with locking operations.

[Loc](#)
[kStr](#)
[ateg](#)
[yNo](#)

[tSu](#)
[ppo](#)
[rted](#)
[Exc](#)
[epti](#)
[on](#)

A LockStrategyNotSupportedException exception occurs if a map is configured with an unsupported lock strategy.

[Loc](#)
[kTi](#)
[meo](#)
[utE](#)
[xce](#)
[ptio](#)
[n](#)

A LockTimeoutException exception occurs when the lock manager detects that the lock wait time exceeded the maximum wait time. The timeout might be the result of a deadlock. If a deadlock is causing the timeout, the timeout is used to break the deadlock.

[Mul](#)
[tipl](#)
[ePa](#)
[rtiti](#)
[on](#)
[Writ](#)
[eEx](#)
[cept](#)
[ion](#)

A MultiplePartitionWriteException exception is a base exception for client/server operations when a user attempts to write to multiple remote partitions on remote servers in the same transaction.

An OptimisticCollisionException occurs when an optimistic locking strategy is used and more than one update transaction collides on the same map entry of an ObjectGrid instance. The first transaction to commit updates the version object for the map entry. Other transactions that read this same map entry before committing have the previous version object. When the other transactions try to commit, the version object that is read does not match the version that was last committed. Therefore, other transactions are prevented from updating a map entry with stale data.

[Opti](#)
[mist](#)
[icCo](#)
[llisi](#)
[onE](#)
[xce](#)
[ptio](#)
[n](#)

The default OptimisticCallback plug-in is used by the run time if an implementation is not provided by the application. If a well-constructed equals(Object) method is not on your value object, this exception occurs because the entire value object is used as the version object.

Because this exception indicates that the map entry contains stale data, stale map entries or entries as identified by the key parameter that is passed to the OptimisticCollisionException(String, String, String, Object) method are invalidated. If this exception is thrown by a Loader plug-in and a null reference is used as the key parameter by the loader, the run time assumes that the loader does not know which entry caused the exception. In this scenario, the LogSequence object is passed to the Loader.batchUpdate(TxID, LogSequence) method to determine which map entries to invalidate. Each LogElement entry in the LogSequence object that is type update or

delete is invalidated.

Specifies one or more attributes to use to calculate the partition hash code.

[PartitionKeyAttribute](#)

The PartitionKey attribute can be specified for the class using a path syntax to identify a single attribute. Multiple attributes can be specified using additional PartitionKey annotations on each field, using the [Order](#) attribute to specify the order in which the hash codes will be calculated.

The PartitionKey attribute is not inheritable.

[ReadOnlyException](#)

A ReadOnlyException exception occurs when a modify operation is attempted on a read-only map.

[TargetNotAvailableException](#)

A TargetNotAvailableException occurs when a remote target was not found or was not reachable.

[TransactionAffinityException](#)

A TransactionAffinityException exception occurs during server failover for inflight transactions. Applications can try the transaction again.

[UnavailableServiceException](#)

An UnavailableServiceException exception occurs when all servers are not running, or when all services are not available even though servers are running.

[UndefinedMapException](#)

An UndefinedMapException exception occurs to indicate that the map that an application tried to access is not defined in the ObjectGrid.

Interfaces

Interface	Description
IGridMap TKey , TValue	The top level interface for all maps. Use the interface to retrieve an appropriate IGridMap instance.
IGridMapPessimisticAutoTx TKey , TValue	Different IGridMap implementations are returned which allow additional operations for specific configurations and usage patterns.
IGridMapPessimisticAutoTx TKey , TValue	This is a handle to a map using automatic transaction demarcation.
IGridMapPessimisticAutoTx TKey , TValue	An instance of this IGridMapPessimisticAutoTx can only be used by the thread at a time. Use the Dispose method when finished with the map, to improve performance.
IGridMapPessimisticAutoTx TKey , TValue	This is a handle to a map using the pessimistic locking strategy and

manual transaction demarcation. All data access operations must occur within an active transaction.

[IGridMapPessimisticTx TKey, TValue](#)

Use the [Transaction](#) property to access the [IGridTransaction](#) instance that is associated with this map instance, and use the [Begin](#) method to begin a transaction. All keys in a single transaction must resolve to the same partition.

An instance of this IGridMapPessimisticTx is not thread-safe, and can only be used by the thread at a time. Use the [Dispose](#) method when finished with the map, to improve performance.

[IPartitionManager TKey, TValue](#)

An IPartitionManager provides properties and methods for determining how partitions are calculated. Retrieve an IPartitionManager from an [IGridMap TKey, TValue](#).

Enumerations

**E
n
u
m
e
r
a
t
i
o
n**

Description

[LockMode](#)

Specifies the strength of a lock to acquire.

[TTLType](#)

Every grid map has an optional, built in timed based evictor that is referred to as the "time to live" evictor or TTL evictor. Each grid map entry has an expiration time that determines how long the entry is allowed to live in the grid map. When the expiration time is reached, the TTL evictor causes the expired entry to be evicted from the grid map. This enum defines the TTLType value constants that determine how the the expiration time is computed for a map entry.

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A CacheKeyNotFoundException exception occurs if a key cannot be found in the cache.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map CacheKeyNotFoundException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[CacheKeyNotFoundException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CacheKeyNotFoundException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> CacheKeyNotFoundException	Initializes a new instance of the CacheKeyNotFoundException class.
<input type="checkbox"/> CacheKeyNotFoundException(Exception)	Initializes a new instance of the CacheKeyNotFoundException class with a specified exception cause.
<input type="checkbox"/> CacheKeyNotFoundException(String)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message.
<input type="checkbox"/> CacheKeyNotFoundException(String, Exception)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[CacheKeyNotFoundException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> CacheKeyNotFoundException	Initializes a new instance of the CacheKeyNotFoundException class.
<input type="checkbox"/> CacheKeyNotFoundException(Exception)	Initializes a new instance of the CacheKeyNotFoundException class with a specified exception cause.
<input type="checkbox"/> CacheKeyNotFoundException(String)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message.
<input type="checkbox"/> CacheKeyNotFoundException(String, Exception)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

The error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CacheKeyNotFoundException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)









See Also

[CacheKeyNotFoundException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CacheKeyNotFoundException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[CacheKeyNotFoundException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

Specifies the class alias that you can use to correlate different class names that have the same class alias. The class alias and class fields or types are used to identify a unique class type ID during the object class serialization and de-serialization.

The [ClassAlias] annotation can be specified for a user defined class.

Inheritance Hierarchy

System Object

System Attribute

IBM.WebSphere.Caching.Map ClassAliasAttribute

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

The syntax for the class could look like :

```
[ClassAlias("ACME_Employee")]
```

In the above example, ACME_Employee is the class alias for this user defined class.

If [ClassAlias] annotation is not defined, the name of this class is set as the ClassAlias.

[Copy to ClipboardPrint](#)

```
[ClassAlias("ACME_Employee")]
class Employee1 {
    [FieldAlias("Employee ID")]
    int empId = -1;

    [FieldAlias("Department No.")]
    int deptId = -1;

    [FieldAlias("Year Salary")]
    float salary = 0;

    [FieldAlias("SEX")]
    String sex = "M";

    int age = -1;
    String homeAddress = "";
}
```

When a ClassAlias and/or FieldAlias are specified in a user defined class, the ClassAlias and/or FieldAlias will be used to create or correlate with an object that are stored or will be stored in the grid. If two user defined classes (in a separate .NET application environment) have the different class name, but they were marked as the same ClassAlias, and all fields and field types are matched between these 2 classes, they will be correlated with the same class type ID even though they have the different class name. This way will allow the same class metadata to be reused between these 2 classes when running serialization and de-serialization in the different .NET application runtime, as well as to shared with Java when

the Alias for the class defined in Java and fields are also matched.

See Also

[ClassAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)









IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClassAliasAttribute](#) type exposes the following members.
Constructors

Name	Description
 ClassAliasAttribute	The class alias.



[Back to Top](#)

Methods

Name	Description
 Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
 Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
 GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
 GetType	Gets the Type of the current instance. (Inherited from Object.)
 IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
 Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
 MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
 ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
 Alias	The class alias.
 Typed	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[ClassAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClassAliasAttribute
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The class alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

value

Type: System String

The class alias value.

See Also

[ClassAliasAttribute Class](#)

[ClassAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClassAliasAttribute](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
<input type="checkbox"/> Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[ClassAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)



IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClassAliasAttribute](#) type exposes the following members.
Properties

	Name	Description
	Alias	The class alias.
	TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[ClassAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClassAliasAttribute Alias
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The class alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The class alias.

See Also

[ClassAliasAttribute Class](#)

[ClassAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A ClientServerLoaderException exception is a base exception for any client/server operation exceptions.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map LoaderException](#)

IBM.WebSphere.Caching.Map ClientServerLoaderException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[ClientServerLoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [ClientServerLoaderException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> ClientServerLoaderException	Initializes a new instance of the ClientServerLoaderException class.
<input type="checkbox"/> ClientServerLoaderException(Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified exception cause.
<input type="checkbox"/> ClientServerLoaderException(String)	Initializes a new instance of the ClientServerLoaderException class with the specified error message.
<input type="checkbox"/> ClientServerLoaderException(String, Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerLoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> ClientServerLoaderException	Initializes a new instance of the ClientServerLoaderException class.
<input type="checkbox"/> ClientServerLoaderException(Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified exception cause.
<input type="checkbox"/> ClientServerLoaderException(String)	Initializes a new instance of the ClientServerLoaderException class with the specified error message.
<input type="checkbox"/> ClientServerLoaderException(String, Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerLoaderException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)









See Also

[ClientServerLoaderException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerLoaderException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerLoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A DuplicateKeyException exception occurs if a key cannot be inserted into the backing map because an object with the same key already exists.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map DuplicateKeyException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[DuplicateKeyException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [DuplicateKeyException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> DuplicateKeyException	Initializes a new instance of the DuplicateKeyException class.
<input type="checkbox"/> DuplicateKeyException(Exception)	Initializes a new instance of the DuplicateKeyException class with a specified exception cause.
<input type="checkbox"/> DuplicateKeyException(String)	Initializes a new instance of the DuplicateKeyException class with the specified error message.
<input type="checkbox"/> DuplicateKeyException(String, Exception)	Initializes a new instance of the DuplicateKeyException class with the specified error message and exception cause.





[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception.

(Inherited from Exception.)

Gets or sets the name of the application or the object that causes the error.

 Source

(Inherited from Exception.)

Gets a string representation of the frames on the call stack at the time the current exception was thrown.

 StackTrace

(Inherited from Exception.)

Gets the method that throws the current exception.

 TargetSite

(Inherited from Exception.)

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> DuplicateKeyException	Initializes a new instance of the DuplicateKeyException class.
<input type="checkbox"/> DuplicateKeyException(Exception)	Initializes a new instance of the DuplicateKeyException class with a specified exception cause.
<input type="checkbox"/> DuplicateKeyException(String)	Initializes a new instance of the DuplicateKeyException class with the specified error message.
<input type="checkbox"/> DuplicateKeyException(String, Exception)	Initializes a new instance of the DuplicateKeyException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

The error message that explains the reason for the exception.

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies an error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [DuplicateKeyException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [DuplicateKeyException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

Specifies the field alias that you can use to correlate different class field names that have the same class field alias. The class alias and field alias or types are used to identify a unique class type ID during the object class serialization and de-serialization.

The [FieldAlias] annotation can be specified for user defined class fields.

Inheritance Hierarchy

System Object

System Attribute

IBM.WebSphere.Caching.Map FieldAliasAttribute

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

The syntax for the field alias could look as follows :

```
[FieldAlias("Employee ID")]
```

In the above example, "Employee ID" is a field alias for the empId field in this user defined class.

If [FieldAlias] annotation is not defined, the name of this field is set as the FieldAlias.

[Copy to ClipboardPrint](#)

```
[ClassAlias("ACME_Employee")]
class Employee1 {
    [FieldAlias("Employee ID")]
    int empId = -1;

    [FieldAlias("Department No.")]
    int deptId = -1;

    [FieldAlias("Year Salary")]
    float salary = 0;

    [FieldAlias("SEX")]
    String sex = "M";

    int age = -1;
    String homeAddress = "";
}
```

When a ClassAlias and/or FieldAlias are specified in a user defined class, the ClassAlias and/or FieldAlias will be used to create or correlate with an object that are stored or will be stored in the grid. If two user defined classes (in a separate .NET application environment) have the different class name, but they were marked as the same ClassAlias, and all fields and field types are matched between these 2 classes, they will be correlated with the same class type ID even though they have the different class name. This way will allow the same class metadata to be reused between these 2 classes when running serialization and de-

serialization in the different .NET application runtime, as well as to shared with Java when the Alias for the class defined in Java and fields are also matched.

See Also

[FieldAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)


IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)









The [FieldAliasAttribute](#) type exposes the following members.

Constructors

Name	Description
 FieldAliasAttribute	The field alias.



[Back to Top](#)

Methods

Name	Description
 Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
 Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
 GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
 GetType	Gets the Type of the current instance. (Inherited from Object.)
 IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
 Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
 MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
 ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
 Alias	The field alias.
 TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[FieldAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

FieldAliasAttribute
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The field alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

value

Type: System String

The field alias value.

See Also

[FieldAliasAttribute Class](#)

[FieldAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [FieldAliasAttribute](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
<input type="checkbox"/> Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[FieldAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)



IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [FieldAliasAttribute](#) type exposes the following members.
Properties

	Name	Description
	Alias	The field alias.
	TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[FieldAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

FieldAliasAttribute Alias
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The field alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The field alias.

See Also

[FieldAliasAttribute Class](#)

[FieldAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The top level interface for all maps. Use the interface to retrieve an appropriate IGridMap instance.

Different IGridMap implementations are returned which allow additional operations for specific configurations and usage patterns.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Type Parameters

TKey

Generic type key.

TValue

Generic type value.

See Also

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

[IBM.WebSphere.Caching IGrid](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue
Members




IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMap TKey, TValue](#) type exposes the following members.

Properties

	Name	Description
	Grid	Retrieves the IGrid instance associated with this map.
	Name	Retrieves the map name.
	PartitionManager	Retrieves the IPartitionManager associated with this map.

[Back to Top](#)

See Also

[IGridMap TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)




IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMap TKey, TValue](#) type exposes the following members.
Properties

	Name	Description
	Grid	Retrieves the IGrid instance associated with this map.
	Name	Retrieves the map name.
	PartitionManager	Retrieves the IPartitionManager associated with this map.

[Back to Top](#)

See Also

[IGridMap TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue Grid IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the IGrid instance associated with this map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The IGrid instance.

See Also

[IGridMap TKey, TValue Interface](#)

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue Name IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the map name.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The map name.

See Also

[IGridMap TKey, TValue Interface](#)

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey,
TValue PartitionManager Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the IPartitionManager associated with this map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The IPartitionManager associated with this map.

See Also

[IGridMap TKey, TValue Interface](#)

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Interface

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

This is a handle to a map using automatic transaction demarcation.

An instance of this IGridMapPessimisticAutoTx can only be used by the thread at a time. Use the Dispose method when finished with the map, to improve performance.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Type Parameters

TKey

Generic type key.

TValue

Generic type value.

Remarks

Each data access method includes a "Specification details" table that includes the following information:

Required permission: The permission required to use the API.

Pessimistic read lock acquired: The type of lock that is acquired when using pessimistic locking with repeatable read transaction isolation.

Cache tier: Identifies the map cache tiers that are included when fetching or updating cache entries in the call and under what circumstances. The following tiers are available for IGridMapPessimisticAutoTx maps:

- Server Cache (data grid)
- Loader (if enabled)

Examples

This sample demonstrates how to put a new cache entry into the data grid:

[Copy to ClipboardPrint](#)

```
// Assume we have already connected to the Grid...
```

```
IGrid grid = ...
```

```
// Retrieve a new map instance.
```

```
IGridMapPessimisticAutoTx<long, string> map = grid.GetGridMapPessimisticAutoTx<long, string>("MyPessimisticMap");
```

```
try
```

```
{
```

```
    // Put the entry in the cache.
```

```
    map.Put(123, "Value to cache");
```

```
}  
catch(GridException)  
{  
    // Handle any consequences of failed put.  
}  
  
// Dispose the map (optional, but it improves performance)  
map.Dispose();
```

See Also

[IGridMapPessimisticAutoTx TKey, TValue Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticAutoTx TKey, TValue](#) type exposes the following members.
Methods

Name	Description
Add	Adds the key-value pair to the data grid.
Add	The key must not exist before executing this method.
Add	A DuplicateKeyException is thrown when the duplicate key is discovered.
Add	Adds multiple key-value pairs to the data grid.
Add	The keys must not exist before executing this method.
Add	A DuplicateKeyException is thrown when the duplicate key is discovered, which may be during a flush or commit operation, in which the exception will be an inner exception.
ContainsKey	Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a nullkey cannot be specified.
ContainsKeyAll	Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
Get	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
Get	If the map supports null values, use the ContainsKey(TKey) to test for a key that may have a null value.
Get	Retrieves the values associated with the list of keys that are specified in the keyList. If the value is not found, a null is returned.
Get	If the map supports null values, use the ContainsKeyAll(IList TKey) to test for multiple keys that may have a null value.
Invalidate	Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).
Invalidate	If the key cannot be found in the map, it will be ignored.
InvalidateAll	Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).
InvalidateAll	If a key cannot be found in the map, it will be ignored.
Put	Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.
Put	Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.
Put	Puts multiple key-value pairs to the data grid, replacing or adding a new entries to each data grid tier as needed.
PutAll	Note: This method has the same specification as the ObjectMap.upsertAll method

in the eXtreme Scale Java client.

[Remove](#) Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

[RemoveAll](#) Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

[Replace](#) Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) is thrown during commit.

[ReplaceAll](#) Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[ResetDefaults](#) Resets the configurable settings for the map back to configured values.

[Touch](#) Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TouchAll](#) Updates the last access time for the data grid entries specified in the keyList without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TryAdd](#) A variant of [Add\(TKey, TValue\)](#) that does not throw exceptions.

[TryAddAll](#) Adds the key-value pairs to the data grid.

[TryInvalidate](#) Removes the entry associated with the specified key from the cache, leaving the data behind unchanged.

[TryInvalidateAll](#) Removes the entries associated with the keys that are specified in the keyList from the cache, leaving the data behind unchanged.

[TryPut](#) Adds the key-value pair to the grid. If an entry for the key exists in the data grid, the value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryPutAll](#) Adds each key-value pair to the data grid. If an entry for a key exists in the data grid, its value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryRemove](#) Removes the entry that is associated with the specified key from the data grid. If the key has no matching entry in the map, no action is taken.

[TryRemoveAll](#) Removes the list of entries associated with the keys that are specified in the keyList. If a key in the keyList has no matching entry in the map, no action is taken for that key.

[TryReplace](#) Replaces the value of a key-value pair in the data grid. If an entry for the key exists in the data grid, its value is replaced with the specified value.

[TryReplaceAll](#) Replaces each key-value pair that is specified in the dictionary object in the data grid. If an entry for the key exists in the data grid, its value is replaced with the corresponding value specified.

[TryTouch](#) Updates the last access time for the data grid entry that matches the key.

[TryTouchAll](#) Updates the last access time for the data grid entries that match each key in the keyList.

[Back to Top](#)

Properties

**N
a
m
e**

Description

[Grid](#) Retrieves the IGrid instance associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

[Item](#) Gets or sets the value in the map using the specified key.

[Lock](#)

[Timeout](#) Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

[Name](#)

[Map](#) Retrieves the map name.
(Inherited from [IGridMap TKey, TValue](#).)

[Partition](#)

[Manager](#) Retrieves the IPartitionManager associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

[TimeToLive](#)

Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".

[Li](#) This property can only be used when the [TtlEvictorType](#) property is set to [v](#) LastAccessTime or LastUpdateTime on the map configuration. If this method is called [e](#) on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException is thrown.

To disable the TTL timeout, use a value of TimeSpan.Zero.

To revert the TTL value to the configured default, use a value of TimeSpan.MinValue.

[T](#)
[tl](#)
[E](#)
[vi](#)
[ct](#)
=[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

Gets the [TtlType](#) of the map's TTL evictor.

[Back to Top](#)

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticAutoTx TKey, TValue](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Add	Adds the key-value pair to the data grid.
<input type="checkbox"/> Add	The key must not exist before executing this method.
<input type="checkbox"/> Add	A DuplicateKeyException is thrown when the duplicate key is discovered.
<input type="checkbox"/> Add	Adds multiple key-value pairs to the data grid.
<input type="checkbox"/> Add	The keys must not exist before executing this method.
<input type="checkbox"/> Add	A DuplicateKeyException is thrown when the duplicate key is discovered, which may be during a flush or commit operation, in which the exception will be an inner exception.
<input type="checkbox"/> ContainsKey	Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a nullkey cannot be specified.
<input type="checkbox"/> ContainsKeyAll	Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
<input type="checkbox"/> Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
<input type="checkbox"/> Get	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
<input type="checkbox"/> Get	If the map supports null values, use the ContainsKey(TKey) to test for a key that may have a null value.
<input type="checkbox"/> Get	Retrieves the values associated with the list of keys that are specified in the keyList. If the value is not found, a null is returned.
<input type="checkbox"/> GetAll	If the map supports null values, use the ContainsKeyAll(IList TKey) to test for multiple keys that may have a null value.
<input type="checkbox"/> Invalidate	Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).
<input type="checkbox"/> Invalidate	If the key cannot be found in the map, it will be ignored.
<input type="checkbox"/> InvalidateAll	Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).
<input type="checkbox"/> Put	If a key cannot be found in the map, it will be ignored.
<input type="checkbox"/> Put	Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.
<input type="checkbox"/> Put	Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.
<input type="checkbox"/> PutAll	Puts multiple key-value pairs to the data grid, replacing or adding a new entries to each data grid tier as needed.
<input type="checkbox"/> PutAll	Note: This method has the same specification as the ObjectMap.upsertAll method

in the eXtreme Scale Java client.

[Remove](#) Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

[RemoveAll](#) Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

[Replace](#) Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) is thrown during commit.

[ReplaceAll](#) Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[ResetDefaults](#) Resets the configurable settings for the map back to configured values.

[Touch](#) Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TouchAll](#) Updates the last access time for the data grid entries specified in the keyList without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TryAdd](#) A variant of [Add\(TKey, TValue\)](#) that does not throw exceptions.

[TryAddAll](#) Adds the key-value pairs to the data grid.

[TryInvalidate](#) Removes the entry associated with the specified key from the cache, leaving the data behind unchanged.

[TryInvalidateAll](#) Removes the entries associated with the keys that are specified in the keyList from the cache, leaving the data behind unchanged.

[TryPut](#) Adds the key-value pair to the grid. If an entry for the key exists in the data grid, the value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryPutAll](#) Adds each key-value pair to the data grid. If an entry for a key exists in the data grid, its value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryRemove](#) Removes the entry that is associated with the specified key from the data grid. If the key has no matching entry in the map, no action is taken.

[TryRemoveAll](#) Removes the list of entries associated with the keys that are specified in the keyList. If a key in the keyList has no matching entry in the map, no action is taken for that key.

[TryReplace](#) Replaces the value of a key-value pair in the data grid. If an entry for the key exists in the data grid, its value is replaced with the specified value.

- [TryReplaceAll](#) Replaces each key-value pair that is specified in the dictionary object in the data grid. If an entry for the key exists in the data grid, its value is replaced with the corresponding value specified.
- [TryTouch](#) Updates the last access time for the data grid entry that matches the key.
- [TryTouchAll](#) Updates the last access time for the data grid entries that match each key in the keyList.

[Back to Top](#)

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey, TValue Add Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pair to the data grid.

The key must not exist before executing this method.

A DuplicateKeyException is thrown when the duplicate key is discovered.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be inserted in the data grid.

value

Type: [TValue](#)

Specifies the value to be inserted in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when the key exists.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds multiple key-value pairs to the data grid.

The keys must not exist before executing this method.

A DuplicateKeyException is thrown when the duplicate key is discovered, which may be during a flush or commit operation, in which the exception will be an inner exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic.IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to be inserted into the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when a dictionary key in entries already exists in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ContainsKey Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a nullkey cannot be specified.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key to test in the map.

Return Value

Returns true if the key is found, false otherwise.

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ContainsKeyAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies a list of keys to test in the map.

Return Value

Specifies a list of boolean values. If the key is found in the keyList, true is listed. Otherwise, false is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when keyList is null, or when a null key is specified in keyList.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Get Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.

If the map supports null values, use the [ContainsKey\(TKey\)](#) to test for a key that may have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to query.

Return Value

The value that is associated with the specified key if it exists; otherwise null is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified, or when the data type of the value that was obtained from the data grid does not match the declared data type.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue GetAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the values associated with the list of keys that are specified in the keyList. If the value is not found, a null is returned.

If the map supports null values, use the [ContainsKeyAll\(IList TKey \)](#) to test for multiple keys that may have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to query.

Return Value

A list of values that are associated with the supplied keys. If the value associated with a particular key is not in the data grid, null is returned in the list at the position that is associated with the key.

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentException	Occurs when the keyList is null, or when a null key is specified in keyList.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing,
---	--

IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.
--	--

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Invalidate Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue InvalidateAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Put Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be put in the data grid.

value

Type: [TValue](#)

Specifies the value to be put in the data grid.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey, TValue PutAll Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts multiple key-value pairs to the data grid, replacing or adding a new entries to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsertAll method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic.IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to be put into the data grid.

Return Value

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
---	--

IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.
--	--

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Remove Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue RemoveAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Replace Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) is thrown during commit.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be updated.

value

Type: [TValue](#)

Specifies the value to be updated in the data grid and Loader.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ReplaceAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to replace in the data grid.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within entries.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Map.CacheKeyNotFoundException	Occurs when the key is not found in any of the cache tiers or Loader. This may be deferred until commit time.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ResetToDefaults Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Resets the configurable settings for the map back to configured values.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Remarks

This method only resets configuration parameters that can be overridden by the client.
See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Touch Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to have last access time updated.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.Map.CacheKeyNotFoundException	Occurs if the key does not exist in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TouchAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entries specified in the keyList without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the keys to have last access time updated.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList.
IBM.WebSphere.Caching.Map CacheKeyNotFoundException	Occurs if the key does not exist in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryAdd Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A variant of [Add\(TKey, TValue\)](#) that does not throw exceptions.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be inserted in the data grid.

value

Type: [TValue](#)

Specifies the value to be inserted in the data grid.

Return Value

Returns true if successful, false if unsuccessful.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryAddAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pairs to the data grid.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic.IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to be inserted into the data grid.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If entries is null or a null key is specified in a dictionary entry or a dictionary key in entries exists, false is returned. If all keys in the dictionary entries do not resolve to the same partition or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryInvalidate Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry associated with the specified key from the cache, leaving the data behind unchanged.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key for which you want to remove the value.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If key is null or the caller does not have authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryInvalidateAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries associated with the keys that are specified in the keyList from the cache, leaving the data behind unchanged.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

The list of keys whose values are to be removed.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If keyList is null or a null key is specified in a keyList entry, false is returned. If all keys in the keyList do not resolve to the same partition or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryPut Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pair to the grid. If an entry for the key exists in the data grid, the value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

specifies the key to be inserted into the data grid.

value

Type: [TValue](#)

specifies the value to be inserted into the data grid.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If the key is null, or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryPutAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds each key-value pair to the data grid. If an entry for a key exists in the data grid, its value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object that contains the key-value pairs to be added to the data grid.

Return Value

Returns true if successful, false if unsuccessful.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryRemove Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid. If the key has no matching entry in the map, no action is taken.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key for which you want to remove the entry.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If key is null or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryRemoveAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the list of entries associated with the keys that are specified in the keyList. If a key in the keyList has no matching entry in the map, no action is taken for that key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

The list of keys whose entries are to be removed.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If keyList is null or a null key is specified in a keyList entry, false is returned. If all keys in the keyList do not resolve to the same partition or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryReplace Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces the value of a key-value pair in the data grid. If an entry for the key exists in the data grid, its value is replaced with the specified value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key for the entry to be replaced.

value

Type: [TValue](#)

Specifies the new value.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If key is null or the key does not match an existing entry in the data grid, false is returned. If the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryReplaceAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces each key-value pair that is specified in the dictionary object in the data grid. If an entry for the key exists in the data grid, its value is replaced with the corresponding value specified.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object that contains key-value pairs to be replaced in the data grid.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If entries is null, a null key is specified in a dictionary entry, or the key does not match an existing entry in the data grid, false is returned. If all keys in the dictionary entries do not resolve to the same partition, or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryTouch Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entry that matches the key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to have last access time updated.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If the key is null or the key does not match an existing entry in the data grid, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryTouchAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entries that match each key in the keyList.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies a list of keys to have last access time updated.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If an entry is null or any key in the keyList is null, false is returned. If a key in the keyList does not match an existing entry in the data grid or all keys in the keyList do not resolve to the same partition, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticAutoTx TKey, TValue](#) type exposes the following members.
Properties

Name	Description
Grid	Retrieves the IGrid instance associated with this map. (Inherited from IGridMap TKey, TValue .)
Item	Gets or sets the value in the map using the specified key.
LockTimeOut	Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.
Name	Retrieves the map name. (Inherited from IGridMap TKey, TValue .)
PartitionManager	Retrieves the IPartitionManager associated with this map. (Inherited from IGridMap TKey, TValue .)
TimeToLive	Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".
UseTtlEvictor	This property can only be used when the TtlEvictorType property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException is thrown.

To disable the TTL timeout, use a value of `TimeSpan.Zero`.

To revert the TTL value to the configured default, use a value of `TimeSpan.MinValue`.

[T](#)
[tl](#)
[E](#)
[vi](#)
[ct](#)
[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

Gets the [TtlType](#) of the map's TTL evictor.

[Back to Top](#)

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Item Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value in the map using the specified key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the value to set or get.

Field Value

The value associated with key to get or set.

Remarks

This indexer retrieves or put's the key and value into the map using the [Get\(TKey\)](#) and [Put\(TKey, TValue\)](#) methods respectively.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

[IGridMapPessimisticAutoTx TKey, TValue Get\(TKey\)](#)

[IGridMapPessimisticAutoTx TKey, TValue Put\(TKey, TValue\)](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey, TValue LockTimeout Property

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The maximum time to wait when acquiring a lock on an item in the grid map.

Exceptions

Exception	Condition
System.ArgumentException	Occurs if the value is not \geq Zero.

Remarks

To prevent deadlocks from occurring, the grid map has a default timeout value of 15 seconds; however, on a heavily loaded system, lock timeouts can occur without an actual deadlock. Use this property to increase the value from the default to prevent false lock timeout exceptions from occurring.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

! " # \$ % & (+ , &&& % \$
 & (% " \$
 \$) *
 ! " # \$ % & (+ , &&& % \$
 (1 1 \$ & % \$ / \$ \$ \$ % 0 % %
 & \$ / \$ 0 % \$ \$ 0 %
 & \$ / \$ \$ \$ % \$ \$ & %
 \$ 2 2 \$ % \$ \$ / \$ \$ % \$ \$ & % \$ \$ & %
 \$ % \$ \$ / % \$ 0 \$ \$ / \$ & % \$ % \$ / %
 \$ % \$ \$ 1% 1&
 \$ \$ 0 \$ ()
 \$ \$ % \$ & % \$ \$
 3 \$ / \$ (\$ 0 \$ &
 3 \$ / \$ (\$ 0 \$ &
 % \$ & &
 % % % \$ & \$ &

Namespace: & # \$ / &
 Assembly: # \$ & 5 \$ # \$ & & 6 \$ 7, &&&
 \$

#8
 #99
) 8
 :
 # # \$
 (\$

Exception	Condition
\$ (/ ; % \$ \$ <= . \$ & \$ 3 % &	\$ (/ ; % \$ \$ <= . \$ & \$ 3 % &
+ *	

+ > \$7 \$&' ? (

\$ %
& # \$ / & ' &
 ! " # \$ % & (% \$ &) # \$ \$ A. BA&
 \$ % \$! " ! " & @ # / # \$ A. BA&
 \$) * \$? \$ \$ &

IGridMapPessimisticAutoTx TKey,
TValue TtlEvictorType Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the [TtlType](#) of the map's TTL evictor.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The [TtlType](#) of the map's TTL evictor.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

This is a handle to a map using the pessimistic locking strategy and manual transaction demarcation. All data access operations must occur within an active transaction.

Use the [Transaction](#) property to access the [IGridTransaction](#) instance that is associated with this map instance, and use the [Begin](#) method to begin a transaction. All keys in a single transaction must resolve to the same partition.

An instance of this IGridMapPessimisticTx is not thread-safe, and can only be used by the thread at a time. Use the Dispose method when finished with the map, to improve performance.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Type Parameters

TKey

Generic type key.

TValue

Generic type value.

Remarks

Each data access method includes a "Specification details" table that includes the following information:

Required permission: The permission required to use the API.

Pessimistic read lock acquired: The type of lock that is acquired when you are using pessimistic locking with repeatable read transaction isolation.

Pessimistic read lock held: The type of lock that is held for the duration of the transaction with repeatable read transaction isolation. Locks can be upgraded but not demoted during a transaction.

Identifies the map cache tiers that are included when fetching or updating cache entries in the call and under what circumstances. The following tiers are available for IGridMapPessimisticTx maps:

Cache tier:

- Transactional Cache
- Server Cache (data grid)
- Loader (if enabled)

Examples

This sample demonstrates how to put a new cache entry into the data grid:

[Copy to ClipboardPrint](#)

```

// Assume we have already connected to the Grid...
IGrid grid = ...

// Retrieve a new map instance.
IGridMapPessimisticTx<long, string> map = grid.GetGridMapPessimisticTx<long, string>("MyPessimisticMap");

// Start a transaction.
map.Transaction.Begin();

try
{
    // Lock the entry in the data grid with an Upgradable lock.
    map.Lock(123, LockMode.Upgradable);

    // Put the entry in the transactional cache.
    map.Put(123, "Value to cache");

    // Commit the transaction to the data grid.
    map.Transaction.Commit();
}
catch(GridException)
{
    // Clean-up the transaction if the lock could not be
    // acquired, or the commit failed.
    if(map.Transaction.Active)
    {
        try
        {
            map.Transaction.Rollback();
        }
        catch(Exception)
        {
            // Optionally log this exception, or ignore the exception.
        }
    }

    // Dispose the map (optional, but it improves performance)
    map.Dispose();

    // Rethrow the real exception.
    throw;
}

```

See Also

[IGridMapPessimisticTx TKey, TValue Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticTx TKey, TValue](#) type exposes the following members.
Methods

Name	Description
	Adds the key-value pair to the data grid.
Add	The key must not exist before you run this method.
Add	A DuplicateKeyException exception results when the duplicate key is discovered. Duplicate keys might be discovered with a flush or commit operation. In this scenario, the exception is an inner exception.
	Adds multiple key-value pairs to the data grid.
Add	The keys must not exist before executing this method.
All	A DuplicateKeyException exception results when the duplicate key is discovered. This discovery might happen during a flush or commit operation. In this scenario, the exception is an inner exception.
Contains	Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a null key cannot be specified.
Key	This API does not hold any locks. Use the Lock(TKey, LockMode) to test for a key and retain a lock.
Contains	Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
All	This API does not hold any locks. Use the LockAll(IList TKey, LockMode) to test for a key and retain a lock.
Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
Get	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
Get	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that may have a null value.
All	Retrieves the values that are associated with the list of keys that are specified in the keyList. If the value is not found, a null value is returned.
All	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for multiple keys that might have a null value.
Get	Locks the specified key and retrieves the associated value. If the value is not found, a null is returned.
And	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that might have a null value.
Lock	Locks the specified keys and retrieves the associated values. If a value is not found, a null is returned in the list.
All	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for a key that might have a null value.

invalidate	Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store). If the key cannot be found in the map, the operation is ignored.
invalidateAll	Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store). If a key cannot be found in the map, the operation is ignored.
lock	Locks the specified key and tests to see if the key was previously present in the data grid or Loader.
lockAll	Locks the specified keys and tests to see if each was previously present in the data grid or Loader. Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.
put	Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client. Puts multiple key-value pairs to the data grid, replacing or adding new entries to each data grid tier as needed.
putAll	Note: This method has the same specification as the ObjectMap.upsertAll method in the eXtreme Scale Java client.
remove	Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store). If the key cannot be found in the map, the operation is ignored.
removeAll	Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store). If a key cannot be found in the map, the operation is ignored.
replace	Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store). If the key cannot be found in the data grid a CacheKeyNotFoundException exception results during the commit operation.
replaceAll	Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store). If a key cannot be found in the map a CacheKeyNotFoundException exception results.
resetDefaults	Resets the configurable settings for the map back to configured values.
touch	Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value. If the key cannot be found in the map a CacheKeyNotFoundException exception results during the commit operation.
touchAll	Updates the last access time for the data grid entries that are specified in the keyList list without locking the entries or fetching the values. If a key cannot be found in the map a CacheKeyNotFoundException exception results during the commit operation.

[Back to Top](#)
Properties

N
a

Description

m
e
G
r
i
d
I
t
e
m
L
o
c
k
T
i
m
e
o
u
t
N
a
m
e
P
a
r
t
i
t
i
o
n
M
a
n
a
g
e
r
T
i
m
e
T
o
L
i
v
e
L
i
v
e
T
r
a
n
s
a
c
t

Retrieves the IGrid instance associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

An indexer that retrieves or puts the key and value into the map with the [Get\(TKey\)](#) and [Put\(TKey, TValue\)](#) methods.

Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

Retrieves the map name.
(Inherited from [IGridMap TKey, TValue](#).)

Retrieves the IPartitionManager associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".

This property can be used only when the [TtlEvictorType](#) property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException exception results.

To disable the TTL timeout, use a value of TimeSpan.Zero.

To revert the TTL value to the configured default, use a value of TimeSpan.MinValue.

The Transaction instance used to configure and demarcate a transaction.
(Inherited from [ITransactionable](#).)

[io](#)
[n](#)
[T](#)
[tl](#)
[E](#)
[vi](#)
[ct](#)
[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

Retrieves the time to live type for the evictor on the map.

[Back to Top](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticTx TKey, TValue](#) type exposes the following members.
Methods

Na me	Description
<input type="checkbox"/> Add	Adds the key-value pair to the data grid. The key must not exist before you run this method.
<input type="checkbox"/> Add All	A DuplicateKeyException exception results when the duplicate key is discovered. Duplicate keys might be discovered with a flush or commit operation. In this scenario, the exception is an inner exception. Adds multiple key-value pairs to the data grid. The keys must not exist before executing this method.
<input type="checkbox"/> Cont ains Key	A DuplicateKeyException exception results when the duplicate key is discovered. This discovery might happen during a flush or commit operation. In this scenario, the exception is an inner exception. Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a null key cannot be specified.
<input type="checkbox"/> Cont ains Key All	This API does not hold any locks. Use the Lock(TKey, LockMode) to test for a key and retain a lock. Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
<input type="checkbox"/> Disp ose	This API does not hold any locks. Use the LockAll(IList TKey, LockMode) to test for a key and retain a lock. Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
<input type="checkbox"/> Get	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
<input type="checkbox"/> Get All	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that may have a null value. Retrieves the values that are associated with the list of keys that are specified in the keyList. If the value is not found, a null value is returned.
<input type="checkbox"/> Get And Lock	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for multiple keys that might have a null value. Locks the specified key and retrieves the associated value. If the value is not found, a null is returned.
<input type="checkbox"/> Get And Lock All	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that might have a null value. Locks the specified keys and retrieves the associated values. If a value is not found, a null is returned in the list.
<input type="checkbox"/> Get And Lock All	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for a key that might have a null value.

[invalidate](#) Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

[invalidateAll](#) Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

[lock](#) Locks the specified key and tests to see if the key was previously present in the data grid or Loader.

[lockAll](#) Locks the specified keys and tests to see if each was previously present in the data grid or Loader.

Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.

[put](#) Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.

Puts multiple key-value pairs to the data grid, replacing or adding new entries to each data grid tier as needed.

[putAll](#) Note: This method has the same specification as the ObjectMap.upsertAll method in the eXtreme Scale Java client.

[remove](#) Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

[removeAll](#) Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

[replace](#) If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) exception results during the commit operation.

Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

[replaceAll](#) If a key cannot be found in the map a [CacheKeyNotFoundException](#) exception results.

[resetDefaults](#) Resets the configurable settings for the map back to configured values.

Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

[touch](#) If the key cannot be found in the map a [CacheKeyNotFoundException](#) exception results during the commit operation.

Updates the last access time for the data grid entries that are specified in the keyList list without locking the entries or fetching the values.

[touchAll](#) If a key cannot be found in the map a [CacheKeyNotFoundException](#) exception results during the commit operation.

[Back to Top](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Add Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pair to the data grid.

The key must not exist before you run this method.

A DuplicateKeyException exception results when the duplicate key is discovered. Duplicate keys might be discovered with a flush or commit operation. In this scenario, the exception is an inner exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be inserted in the data grid.

value

Type: [TValue](#)

Specifies the value to be inserted in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a nullkey is specified.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when the key exists.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds multiple key-value pairs to the data grid.

The keys must not exist before executing this method.

A DuplicateKeyException exception results when the duplicate key is discovered. This discovery might happen during a flush or commit operation. In this scenario, the exception is an inner exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: [IBM.WebSphere.Caching.IOrderedDictionary](#)

Specifies a [IOrderedDictionary](#) object of key-value pairs to be inserted into the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when a dictionary key in entries already exists in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx Interface](#)

[IGridMapPessimisticTx Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue ContainsKey Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a null key cannot be specified.

This API does not hold any locks. Use the [Lock\(TKey, LockMode\)](#) to test for a key and retain a lock.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key to test in the map.

Return Value

Returns true if the key is found, false otherwise.

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: No

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue ContainsKeyAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.

This API does not hold any locks. Use the [LockAll\(IList TKey , LockMode\)](#) to test for a key and retain a lock.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies a list of keys to test in the map.

Return Value

Specifies a list of boolean values. If the key is found in the keyList, true is listed. Otherwise, false is returned.

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentException	Occurs when keyList is null, or when a null key is specified in keyList.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
---	--

IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.
--	--

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: No

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Get Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.

If the map supports null values, use the [Lock\(TKey, LockMode\)](#) or the [ContainsKey\(TKey\)](#) to test for a key that may have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to query.

Return Value

The value that is associated with the specified key if it exists; otherwise null is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified, or when the data type of the value that was obtained from the data grid does not match the declared data type.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: Yes

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue GetAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the values that are associated with the list of keys that are specified in the keyList. If the value is not found, a null value is returned.

If the map supports null values, use the [LockAll\(IList TKey , LockMode\)](#) or the [ContainsKeyAll\(IList TKey \)](#) to test for multiple keys that might have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to query.

Return Value

A list of values that are associated with the supplied keys. If the value associated with a particular key is not in the data grid, null is returned in the list at the position that is associated with the key.

Exceptions

Exception	Condition
-----------	-----------

System ArgumentException	Occurs when the keyList is null, or when a null key is specified in keyList.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing,
---	--

IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.
---	--

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: Yes

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey, TValue GetAndLock Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified key and retrieves the associated value. If the value is not found, a null is returned.

If the map supports null values, use the [Lock\(TKey, LockMode\)](#) or the [ContainsKey\(TKey\)](#) to test for a key that might have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)

Specifies the type of lock to acquire.

Return Value

Specifies the value that is associated with the specified key if it exists. Otherwise, null is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:

MapPermission.READ

Pessimistic locks acquired:

LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive

Pessimistic locks held: Yes

Cache tier:

Progresses to all tiers until the key is found and the appropriate lock is acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified keys and retrieves the associated values. If a value is not found, a null is returned in the list.

If the map supports null values, use the [LockAll\(IList TKey , LockMode\)](#) or the [ContainsKeyAll\(IList TKey \)](#) to test for a key that might have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)
Specifies the type of lock to acquire.

Return Value

A list of values that are associated with the supplied keys. If the value associated with a particular key is not in the data grid, null is returned in the list at the position that is associated with the key.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:	MapPermission.READ
Pessimistic locks acquired:	LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive
Pessimistic locks held:	Yes
Cache tier:	Progresses to all tiers until the keys are found and the appropriate locks are acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)
[IGridMapPessimisticTx TKey, TValue Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Invalidate Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue InvalidateAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Lock Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified key and tests to see if the key was previously present in the data grid or Loader.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)

Specifies the type of lock to acquire.

Return Value

Returns true if the key is found in the data grid or Loader (back-end persistent store).

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:

MapPermission.READ

Pessimistic locks acquired:

LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive

Pessimistic locks held: Yes

Cache tier:

Progresses to all tiers until the key is found and the appropriate lock is acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue LockAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified keys and tests to see if each was previously present in the data grid or Loader.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)
Specifies the type of lock to acquire.

Return Value

A list of bool that are associated with the supplied keys, where true indicates that the key was found in the data grid or Loader (back-end persistent store).

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:	MapPermission.READ
Pessimistic locks acquired:	LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive
Pessimistic locks held:	Yes
Cache tier:	Progresses to all tiers until the key is found and the appropriate lock is acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional

information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Put Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be put in the data grid.

value

Type: [TValue](#)

Specifies the value to be put in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue PutAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts multiple key-value pairs to the data grid, replacing or adding new entries to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsertAll method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

entries

Type: [IBM.WebSphere.Caching.IOrderedDictionary](#) TKey, TValue

Specifies a [IOrderedDictionary](#) TKey, TValue object of key-value pairs to be put into the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Remove Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue RemoveAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Replace Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) exception results during the commit operation.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be updated.

value

Type: [TValue](#)

Specifies the value to be updated in the data grid and Loader.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue ReplaceAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) exception results.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

entries

Type: [IBM.WebSphere.Caching IOrderedDictionary TKey, TValue](#)

Specifies a [IOrderedDictionary TKey, TValue](#) object of key-value pairs to replace in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within entries.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Map.CacheKeyNotFoundException	Occurs when the key is not found in any of the cache tiers or Loader. This exception might be deferred until commit time.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue ResetToDefaults Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Resets the configurable settings for the map back to configured values.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Remarks

This method resets configuration parameters that can be overridden by the client only.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Touch Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) exception results during the commit operation.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to have last access time updated.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue TouchAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entries that are specified in the keyList list without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundExcep](#)tion results during the commit operation.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the keys to have last access time updated.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticTx TKey, TValue](#) type exposes the following members.
Properties

Name	Description
Grid	Retrieves the IGrid instance associated with this map. (Inherited from IGridMap TKey, TValue .)
Item	An indexer that retrieves or puts the key and value into the map with the Get(TKey) and Put(TKey, TValue) methods.
LockTimeOut	Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.
Name	Retrieves the map name. (Inherited from IGridMap TKey, TValue .)
PartitionManager	Retrieves the IPartitionManager associated with this map. (Inherited from IGridMap TKey, TValue .)
TimeToLive	Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".
TtlEvictorType	This property can be used only when the TtlEvictorType property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException exception results.

To disable the TTL timeout, use a value of `TimeSpan.Zero`.

To revert the TTL value to the configured default, use a value of `TimeSpan.MinValue`.

[T](#)
[r](#)
[a](#)
[n](#)
[s](#)
[a](#)
[c](#)
[t](#)
[i](#)
[o](#)
[n](#)
[T](#)
[t](#)
[l](#)
[E](#)
[v](#)
[i](#)
[c](#)
[t](#)
[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

The Transaction instance used to configure and demarcate a transaction.
(Inherited from [ITransactionable](#).)

Retrieves the time to live type for the evictor on the map.

[Back to Top](#)
See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Item Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An indexer that retrieves or puts the key and value into the map with the [Get\(TKey\)](#) and [Put\(TKey, TValue\)](#) methods.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue LockTimeout Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The maximum time to wait when acquiring a lock on an item in the grid map.

Exceptions

Exception	Condition
System.ArgumentException	Occurs if the value is not \geq Zero.

Remarks

To prevent deadlocks from occurring, the grid map has a default timeout value of 15 seconds; however, on a heavily loaded system, lock timeouts can occur without an actual deadlock. Use this property to increase the value from the default to prevent false lock timeout exceptions from occurring.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue TimeToLive Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".

This property can be used only when the [TtlEvictorType](#) property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException exception results.

To disable the TTL timeout, use a value of TimeSpan.Zero.

To revert the TTL value to the configured default, use a value of TimeSpan.MinValue.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Exceptions

Exception	Condition
System.ArgumentException	Occurs if the TimeSpan is not ≥ 0 or TimeSpan.MinValue or if the TtlEvictorType property is not LastAccessTime or LastUpdateTime.
Remarks	

Required Permission: MapPermission.INVALIDATE

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue TtlEvictorType Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the time to live type for the evictor on the map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey, TValue
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An IPartitionManager provides properties and methods for determining how partitions are calculated. Retrieve an IPartitionManager from an [IGridMap TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Type Parameters

TKey

Specifies a generic type key.

TValue

Specifies a generic type value.

Remarks

The partition id is calculated as follows:

- The key class is examined for the [PartitionKeyAttribute](#) attribute. If present, the GetHashCode of the referenced attributes is used to calculate the partition.
- Otherwise, the key's GetHashCode method is used to calculate the partition id.

See Also

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)




IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)


IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IPartitionManager TKey, TValue](#) type exposes the following members.
Methods

Name	Description
 GetPartition	Returns the partition number for the specified map element key.
 GetPartitionAll(IList TKey)	Returns a map of partition numbers to map keys from a collection of map element keys.
 GetPartitionAll(IOrderedDictionary TKey, TValue)	Returns a map of partition numbers to map keys from a collection of map element keys.

[Back to Top](#)

Properties

Name	Description
 NumPartitions	Retrieves the number of partitions that are defined for the map.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IPartitionManager TKey, TValue](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> GetPartition	Returns the partition number for the specified map element key.
<input type="checkbox"/> GetPartitionAll(IList TKey)	Returns a map of partition numbers to map keys from a collection of map element keys.
<input type="checkbox"/> GetPartitionAll(IOrderedDictionary TKey, TValue)	Returns a map of partition numbers to map keys from a collection of map element keys.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey,
TValue GetPartition Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns the partition number for the specified map element key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key for which to return a partition number.

Return Value

Returns the partition number for the specified map element key.

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GetPartitionAll(IList TKey)	Returns a map of partition numbers to map keys from a collection of map element keys.
<input type="checkbox"/> GetPartitionAll(IOrderedDictionary TKey, TValue)	Returns a map of partition numbers to map keys from a collection of map element keys.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey,
TValue GetPartitionAll Method (IList TKey)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map of partition numbers to map keys from a collection of map element keys.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies a list of keys for which you want to return partition numbers.

Return Value

Returns a collection of partition numbers and associated keys.

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[GetPartitionAll Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey, TValue GetPartitionAll
Method (IOrderedDictionary TKey, TValue)

IBM WebSphere™ eXtreme Scale
Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map of partition numbers to map keys from a collection of map element keys.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: [IBM.WebSphere.Caching.IOrderedDictionary TKey, TValue](#)

Specifies a dictionary object of key-value pairs for which you want to return partition numbers.

Return Value

Specifies a collection of partition numbers and associated key-value pairs.

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[GetPartitionAll Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.


IPartitionManager TKey, TValue
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IPartitionManager TKey, TValue](#) type exposes the following members.
Properties

Name	Description
 NumPartitions	Retrieves the number of partitions that are defined for the map.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey,
TValue NumPartitions Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the number of partitions that are defined for the map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LoaderException exception is the base exception that results for any exceptions that are encountered by a Loader.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map LoaderException

[IBM.WebSphere.Caching.Map ClientServerLoaderException](#)

[IBM.WebSphere.Caching.Map UnavailableServiceException](#)

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LoaderException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LoaderException	Initializes a new instance of the LoaderException class.
<input type="checkbox"/> LoaderException(Exception)	Initializes a new instance of the LoaderException class with a specified exception cause.
<input type="checkbox"/> LoaderException(String)	Initializes a new instance of the LoaderException class with the specified error message.
<input type="checkbox"/> LoaderException(String, Exception)	Initializes a new instance of the LoaderException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LoaderException	Initializes a new instance of the LoaderException class.
<input type="checkbox"/> LoaderException(Exception)	Initializes a new instance of the LoaderException class with a specified exception cause.
<input type="checkbox"/> LoaderException(String)	Initializes a new instance of the LoaderException class with the specified error message.
<input type="checkbox"/> LoaderException(String, Exception)	Initializes a new instance of the LoaderException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Constructor (String, IBM WebSphere™ eXtreme Scale Client for .NET API Exception) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

[VB](#)

[C#](#)

[C++](#)

[F#](#)

[JScript](#)

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LoaderException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LoaderException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockStrategyNotSupportedException exception occurs when the lock manager detects a deadlock. This exception occurs to prevent the deadlock.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map.LockException](#)

[IBM.WebSphere.Caching.Map.LockTimeoutException](#)

IBM.WebSphere.Caching.Map.LockDeadlockException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Remarks

Typically, a deadlock is the result of the following scenario: One transaction gets a map entry, resulting in a weaker lock than an existing lock on the same entry. At commit time, the transaction attempts to promote the weaker lock to a stronger lock to apply the changes to the data store. For example, two transactions try to promote from shared locks to exclusive locks, but each transaction already owns a shared lock.

See Also

[LockDeadlockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockDeadlockException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LockDeadlockException	Initializes a new instance of the LockDeadlockException class.
<input type="checkbox"/> LockDeadlockException(Exception)	Initializes a new instance of the LockDeadlockException class with the specified exception cause.
<input type="checkbox"/> LockDeadlockException(String)	Initializes a new instance of the LockDeadlockException class with the specified error message.
<input type="checkbox"/> LockDeadlockException(String, Exception)	Initializes a new instance of the LockDeadlockException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor. (Inherited from LockTimeoutException .)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.

(Inherited from Exception.)

InnerException Gets the Exception instance that caused the current exception.
(Inherited from Exception.)

[LockRequestQueueDetails](#) Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.
(Inherited from [LockTimeoutException](#).)

Message Gets a message that describes the current exception.
(Inherited from Exception.)

Source Gets or sets the name of the application or the object that causes the error.
(Inherited from Exception.)

StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)

TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LockDeadlockException	Initializes a new instance of the LockDeadlockException class.
<input type="checkbox"/> LockDeadlockException(Exception)	Initializes a new instance of the LockDeadlockException class with the specified exception cause.
<input type="checkbox"/> LockDeadlockException(String)	Initializes a new instance of the LockDeadlockException class with the specified error message.
<input type="checkbox"/> LockDeadlockException(String, Exception)	Initializes a new instance of the LockDeadlockException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockDeadlockException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor. (Inherited from LockTimeoutException .)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockDeadlockException](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/> LockRequestQueueDetails	Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred. (Inherited from LockTimeoutException .)
<input type="checkbox"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockException exception indicates errors with locking operations.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map LockException

[IBM.WebSphere.Caching.Map LockTimeoutException](#)

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

See Also

[LockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [LockException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LockException	Initializes a new instance of the LockException class.
<input type="checkbox"/> LockException(Exception)	Initializes a new instance of the LockException class with a specified exception cause.
<input type="checkbox"/> LockException(String)	Initializes a new instance of the LockException class with the specified error message.
<input type="checkbox"/> LockException(String, Exception)	Initializes a new instance of the LockException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LockException	Initializes a new instance of the LockException class.
<input type="checkbox"/> LockException(Exception)	Initializes a new instance of the LockException class with a specified exception cause.
<input type="checkbox"/> LockException(String)	Initializes a new instance of the LockException class with the specified error message.
<input type="checkbox"/> LockException(String, Exception)	Initializes a new instance of the LockException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LockException Class](#)

[LockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

The error message that explains the reason for the exception.

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [LockException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockException Class](#)









[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [LockException](#) type exposes the following members.

Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies the strength of a lock to acquire.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Members

Member name	Value	Description
Shared	0	Shared lock that can be obtained if no one is currently holding an exclusive lock.
Upgradable	1	Upgradable lock that can be obtained if no one is currently holding an upgradable or exclusive lock.
Exclusive	2	Exclusive lock that can only be obtained if no locks are held.

See Also

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedExceptio IBM WebSphere™ eXtreme Scale Client for .NET API
n Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockStrategyNotSupportedException exception occurs if a map is configured with an unsupported lock strategy.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map LockStrategyNotSupportedException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockStrategyNotSupportedException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [LockStrategyNotSupportedException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> LockStrategyNotSupportedException	Initializes a new instance of the LockStrategyNotSupportedException class.
<input type="checkbox"/> LockStrategyNotSupportedException(Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with a specified exception cause.
<input type="checkbox"/> LockStrategyNotSupportedException(String)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message.
<input type="checkbox"/> LockStrategyNotSupportedException(String, Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockStrategyNotSupportedException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LockStrategyNotSupportedException	Initializes a new instance of the LockStrategyNotSupportedException class.
<input type="checkbox"/> LockStrategyNotSupportedException(Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with a specified exception cause.
<input type="checkbox"/> LockStrategyNotSupportedException(String)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message.
<input type="checkbox"/> LockStrategyNotSupportedException(String, Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException
Constructor (String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockStrategyNotSupportedException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockStrategyNotSupportedException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockStrategyNotSupportedException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockStrategyNotSupportedException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockTimeoutException exception occurs when the lock manager detects that the lock wait time exceeded the maximum wait time. The timeout might be the result of a deadlock. If a deadlock is causing the timeout, the timeout is used to break the deadlock.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map.LockException](#)

IBM.WebSphere.Caching.Map.LockTimeoutException

[IBM.WebSphere.Caching.Map.LockDeadlockException](#)

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockTimeoutException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LockTimeoutException	Initializes a new instance of the LockTimeoutException class.
<input type="checkbox"/> LockTimeoutException(Exception)	Initializes a new instance of the LockTimeoutException class with a specified exception cause.
<input type="checkbox"/> LockTimeoutException(String)	Initializes a new instance of the LockTimeoutException class with the specified error message.
<input type="checkbox"/> LockTimeoutException(String, Exception)	Initializes a new instance of the LockTimeoutException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

<input type="checkbox"/>	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>	LockRequestQueueDetails	Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.
<input type="checkbox"/>	Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LockTimeoutException	Initializes a new instance of the LockTimeoutException class.
<input type="checkbox"/> LockTimeoutException -	
<input type="checkbox"/> LockTimeoutException n(Exception)	Initializes a new instance of the LockTimeoutException class with a specified exception cause.
<input type="checkbox"/> LockTimeoutException n(String)	Initializes a new instance of the LockTimeoutException class with the specified error message.
<input type="checkbox"/> LockTimeoutException n(String, Exception)	Initializes a new instance of the LockTimeoutException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockTimeoutException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException getMessage IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Return Value

Specifies the detailed message string of this [LockTimeoutException](#) instance.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockTimeoutException](#) type exposes the following members.
Properties

Name	Description
<input type="text"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="text"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="text"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="text"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text"/> LockRequestQueueDetails	Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.
<input type="text"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException LockRequestQueue IBM WebSphere™ eXtreme Scale Client for .NET
Details Property API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

Gets the value that was set in the LockRequestQueueDetails property. If the LockRequestQueueDetails property was not previously set for this exception, the return value is null.

Sets the details of the lock requests on the lock request queue at the time the lock timeout occurred.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A MultiplePartitionWriteException exception is a base exception for client/server operations when a user attempts to write to multiple remote partitions on remote servers in the same transaction.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map MultiplePartitionWriteException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MultiplePartitionWriteException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MultiplePartitionWriteException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> MultiplePartitionWriteException	Initializes a new instance of the MultiplePartitionWriteException class.
<input type="checkbox"/> MultiplePartitionWriteException(Exception)	Initializes a new instance of the MultiplePartitionWriteException class with a specified exception cause.
<input type="checkbox"/> MultiplePartitionWriteException(String)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message.
<input type="checkbox"/> MultiplePartitionWriteException(String, Exception)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> MultiplePartitionWriteException	Initializes a new instance of the MultiplePartitionWriteException class.
<input type="checkbox"/> MultiplePartitionWriteException(Exception)	Initializes a new instance of the MultiplePartitionWriteException class with a specified exception cause.
<input type="checkbox"/> MultiplePartitionWriteException(String)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message.
<input type="checkbox"/> MultiplePartitionWriteException(String, Exception)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MultiplePartitionWriteException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)








IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MultiplePartitionWriteException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An OptimisticCollisionException occurs when an optimistic locking strategy is used and more than one update transaction collides on the same map entry of an ObjectGrid instance. The first transaction to commit updates the version object for the map entry. Other transactions that read this same map entry before committing have the previous version object. When the other transactions try to commit, the version object that is read does not match the version that was last committed. Therefore, other transactions are prevented from updating a map entry with stale data.

The default OptimisticCallback plug-in is used by the run time if an implementation is not provided by the application. If a well-constructed equals(Object) method is not on your value object, this exception occurs because the entire value object is used as the version object.

Because this exception indicates that the map entry contains stale data, stale map entries or entries as identified by the key parameter that is passed to the OptimisticCollisionException(String, String, String, Object) method are invalidated. If this exception is thrown by a Loader plug-in and a null reference is used as the key parameter by the loader, the run time assumes that the loader does not know which entry caused the exception. In this scenario, the LogSequence object is passed to the Loader.batchUpdate(TxID, LogSequence) method to determine which map entries to invalidate. Each LogElement entry in the LogSequence object that is type update or delete is invalidated.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map OptimisticCollisionException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification



The [OptimisticCollisionException](#) type exposes the following members.
Constructors






Name	Description
<input type="checkbox"/> OptimisticCollisionException	Initializes a new instance of the OptimisticCollisionException class with the specified error message, data grid name, map name, and keys that caused the exception.

[Back to Top](#)
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> getGridName	Gets the name of the ObjectGrid instance in which the optimistic collision occurred.
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getKey	Gets key object or array of key objects that caused the OptimisticCollisionException to occur.
<input type="checkbox"/> getMapName	Gets the map name in which the OptimisticCollisionException exception occurred.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> setKey	Set the key that caused this exception to occur.
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)
Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc ption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[OptimisticCollisionException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OptimisticCollisionException](#) class with the specified error message, data grid name, map name, and keys that caused the exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

gridName

Type: System String

Specifies the name of the data grid to which the map belongs.

mapName

Type: System String

Specifies the name of the map.

key

Type: System Object

Specifies the key or array of keys that caused the optimistic collision exception to occur.

Remarks

If more than a single key caused the exception, use an array object for this parameter. Each array element identifies a single map entry that caused the exception to occur. Using array elements is useful when a Loader uses the batch update support of a Java Database Connectivity (JDBC) driver. Pass a null reference if you are unable to determine which key or set of keys caused this exception to occur.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OptimisticCollisionException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> getGridName	Gets the name of the ObjectGrid instance in which the optimistic collision occurred.
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getKey	Gets key object or array of key objects that caused the OptimisticCollisionException to occur.
<input type="checkbox"/> getMapName	Gets the map name in which the OptimisticCollisionException exception occurred.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> setKey	Set the key that caused this exception to occur.
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[OptimisticCollisionException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException GetGridName IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the name of the ObjectGrid instance in which the optimistic collision occurred.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies the name of the ObjectGrid instance.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException getKey IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets key object or array of key objects that caused the [OptimisticCollisionException](#) to occur.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies the key object or array of key objects that caused the exception to occur.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException getMapName Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the map name in which the [OptimisticCollisionException](#) exception occurred.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies the map name where the exception occurred.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException setKey IBM WebSphere™ eXtreme Scale Client for .NET API
Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Set the key that caused this exception to occur.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: System Object

the key or array of key objects that caused the exception

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OptimisticCollisionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[OptimisticCollisionException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies one or more attributes to use to calculate the partition hash code.

The PartitionKey attribute can be specified for the class using a path syntax to identify a single attribute. Multiple attributes can be specified using additional PartitionKey annotations on each field, using the [Order](#) attribute to specify the order in which the hash codes will be calculated.

The PartitionKey attribute is not inheritable.

Inheritance Hierarchy

System Object

System Attribute

IBM.WebSphere.Caching.Map PartitionKeyAttribute

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

The following examples illustrate how to identify a top-level, primitive field:

Simple class attribute example:

[Copy to ClipboardPrint](#)

```
[PartitionKey("deptId")] class Employee { int empId; int deptId; }
```

Simple field attribute example:

[Copy to ClipboardPrint](#)

```
class Employee { int empId; [PartitionKey] int deptId; }
```

Simple, multiple field attribute example:

[Copy to ClipboardPrint](#)

```
class Employee { int empId; [PartitionKey(order=0)] int deptId; [PartitionKey(order=1)] String coun
```

The following examples illustrate how to address an attribute inside an embedded class. In this case, the path separator is defined in the data grid as a "." character (the default for eXtreme Data Format):

Embedded class annotation example:

[Copy to ClipboardPrint](#)

```
[PartitionKey("deptKey.id")] class Employee { DepartmentKey deptKey; } class DepartmentKey { int id
```

Embedded field annotation example:

[Copy to Clipboard](#)[Print](#)

```
class Employee { [PartitionKey("id")] DepartmentKey deptKey; } class DepartmentKey { int id; }
```

See Also

[PartitionKeyAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [PartitionKeyAttribute](#) type exposes the following members.

Constructors

Name	Description
PartitionKeyAttribute	An attribute-less constructor for use when defined on a field, where the field name is the attribute.
PartitionKeyAttribute(String)	Specifies a single attribute to use to calculate the partition hash code.

[Back to Top](#)

Methods

Name	Description
Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
GetType	Gets the Type of the current instance. (Inherited from Object.)
IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
AttributeName	Identifies the path to the attribute that should be included as part of the partition key.
Order	Order of the fields that contribute to the partition calculation. The order values must be unique for all PartitionKey attributes defined on a key class.
TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> PartitionKeyAttribute	An attribute-less constructor for use when defined on a field, where the field name is the attribute.
<input type="checkbox"/> PartitionKeyAttribute(String)	Specifies a single attribute to use to calculate the partition hash code.

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An attribute-less constructor for use when defined on a field, where the field name is the attribute.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[PartitionKeyAttribute Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies a single attribute to use to calculate the partition hash code.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

attributeName

Type: System.String

The attribute name.

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[PartitionKeyAttribute Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [PartitionKeyAttribute](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
<input type="checkbox"/> Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [PartitionKeyAttribute](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> AttributeName	Identifies the path to the attribute that should be included as part of the partition key.
<input type="checkbox"/> Order	Order of the fields that contribute to the partition calculation. The order values must be unique for all PartitionKey attributes defined on a key class.
<input type="checkbox"/> TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IBM WebSphere DataPower XC10 アプライアンス バージョン 2.5 資料

IBM® WebSphere® DataPower® XC10 アプライアンス 資料へようこそ。ここでは、IBM WebSphere DataPower XC10 アプライアンス をインストール、保守、および使用する方法に関する情報を見つけることができます。

始めに

[IBM WebSphere DataPower XC10 アプライアンス の概要](#)

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

2.5+ [チュートリアル: 単純データ・グリッド・アプリケーションの入門](#)

[バージョン 2.5 の新機能](#)

[IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成](#)

[WebSphere DataPower XC10 アプライアンス の更新](#)

一般的なタスク

[アプライアンスの構成](#)

[集合およびゾーンの構成](#)

[データ・グリッド の構成](#)

[クライアントの構成](#)

[データ・グリッドの管理](#)

[単純データ・グリッドにアクセスするアプリケーションの開発](#)

[モニター](#)

[セキュリティ](#)

トラブルシューティングおよび その他の情報 サポート

[トラブルシューティング](#)

[サポート・ポータル](#)

[Fix central](#)

[フォーラム](#)

[リリース情報](#)

[IBM ソフトウェア・サポート・ホーム・ページ](#)

[IBM WebSphere DataPower XC10 アプライアンス 研修](#)

[記事](#)

[Redbooks](#)

[IBM Elastic Caching Community](#)

IBM WebSphere DataPower XC10 アプライアンス の概要

IBM® WebSphere® DataPower® XC10 アプライアンス ライセンス・プロダクトは、WebSphere アプリケーションのための、コスト効率がよく高速なドロップイン・キャッシングを実現するよう最適化された専用アプライアンスです。

利点

DataPower XC10 アプライアンス を使用すると、以下の利点が得られます。

容易なスケールアウト

DataPower XC10 アプライアンス には、ビジネスに不可欠なアプリケーション用のデータをホスティングするために使用できる、240 GB のサイズ調整が可能なデータ・グリッド (data grid)が含まれています。データ・グリッド (data grid)にさらにメモリーを追加するには、ご使用の構成にもう 1 つアプライアンスを追加して、データをホスティングするアプライアンスの集合を作成します。

コード変更なしのドロップインの使用

アプリケーションのコードを大幅に変更することなく DataPower XC10 アプライアンス を使用できます。アプライアンスは以下のドロップイン・シナリオで使用できます。

- HTTP 要求のセッション管理
- WebSphere Application Server動的キャッシュのサポート

フォールト・トレランス

データ・グリッド内のデータが自動的に複製され、データ損失のリスクを低減します。

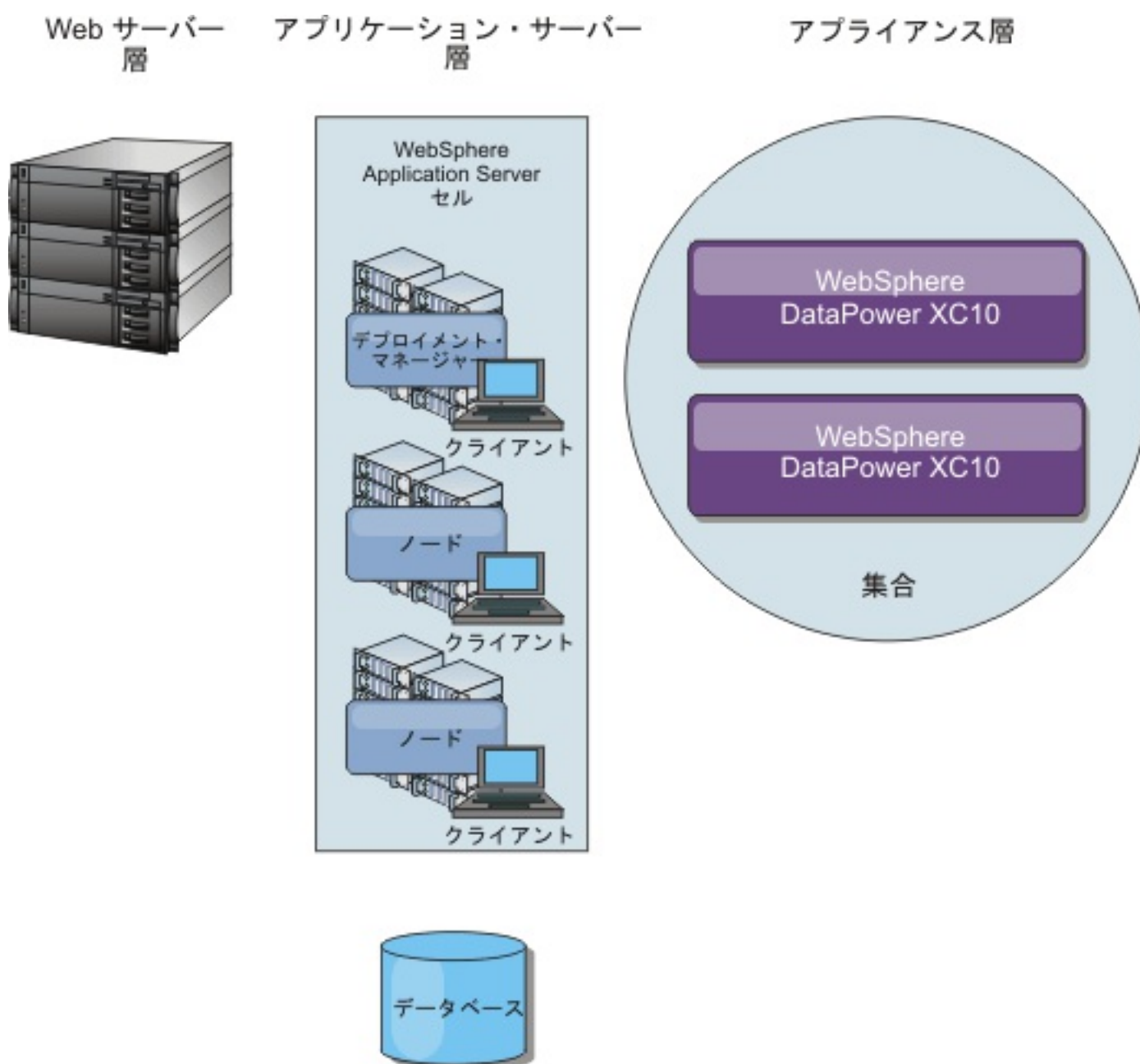
柔軟でシンプルなユーザー管理

ユーザー・インターフェースにより、ご使用のアプライアンスでユーザーおよびユーザー・グループの管理が容易に行えます。また、そのユーザー・インターフェースを使用して、ご使用のデータ・グリッドの作成、管理、およびモニターを行うことができます。

ハイレベル・トポロジー

DataPower XC10 アプライアンス がご使用のアプリケーション・サーバー層の後ろに控えています。アプリケーション・サーバー層では、デプロイメント・マネージャーを含めて、ノードごとに WebSphere eXtreme Scale クライアントをインストールします。このクライアントにより、アプリケーション・サーバー層とアプライアンス層との通信を可能にします。アプライアンス層では、アプライアンス同士を関連付けて集合を形成します。

図1. ハイレベル・トポロジー



バージョン 2.5 の新機能

バージョン 2.5 では、エンタープライズ・データ・グリッド、.NET アプリケーション、セキュリティー機能拡

張、連続照会、モニター向上、およびニア・キャッシュ無効化がサポートされるようになりました。

[リリース情報](#)

製品のサポート Web サイト、製品資料、および製品の最新の更新、制限、および既知の問題へのリンクが提供されています。

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

データ・グリッドは、特定のアプリケーションまたはアプリケーション群で使用するオブジェクトを保持するために作成されるストレージ・ユニットです。集合は、スケーラビリティおよび管理の目的でアプライアンスをグループ化します。ゾーンは、アプライアンスの物理的な場所を定義し、キャッシュ内でのデータの配置を決定するために使用されます。

2.5+ [エンタープライズ・データ・グリッドの概要](#)

エンタープライズ・データ・グリッドは eXtremeIO トランスポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトランスポートおよびシリアライゼーション形式により、Java™ クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。

[トランザクション処理の概要](#)

WebSphere eXtreme Scale クライアントは、データとの相互作用のメカニズムとしてトランザクションを使用します。

[特記事項](#)

[プライバシー・ポリシーに関する考慮事項](#)

バージョン 2.5 の新機能

バージョン 2.5 では、エンタープライズ・データ・グリッド、.NET アプリケーション、セキュリティー機能拡張、連続照会、モニター向上、およびニア・キャッシュ無効化がサポートされるようになりました。

WebSphere eXtreme Scale Client for .NET

WebSphere® eXtreme Scale Client for .NET をインストールすることによって、データ・グリッドにアクセスする .NET アプリケーションをデプロイすることができます。 [詳細...](#)

アプライアンス始動状況

アプライアンスの始動状況をユーザー・インターフェースで確認できるようになりました。 [詳細...](#)

ASP.NET セッション状態ストア・プロバイダー

データ・グリッドにセッション状態を保管するように ASP.NET アプリケーションを構成することができます。 [詳細...](#)

連続照会

データ・グリッドと対話するクライアント・アプリケーションを開発する際、新しいエントリーが挿入されたり更新されたりしたときに自動的なリアルタイムの結果を得る照会が必要になることがあります。連続照会を使用すれば、データ・グリッドでデータの挿入や更新が行われたとき、クライアント Java™ 仮想マシン (JVM) で通知されるようにすることができます。このフィーチャーは、開発者、管理者、またはその両者がグリッドおよびデータの管理をより容易に行えるようにします。 [詳細...](#)

タスクの削除

すべてのタスクをユーザー・インターフェースで削除することができます。 [詳細...](#)

.NET 用クライアント・プロパティの動的更新

クライアント・プロパティ・ファイル内のプロパティ値に対して行われた手動変更を動的に有効にするように WebSphere eXtreme Scale Client for .NET を構成することができます。データ・グリッドへの既存の接続を再始動しなくても変更が有効になります。 [詳細...](#)

FIPS を使用可能に設定

すべての暗号化ネットワーク通信で連邦情報処理標準 (FIPS) 140-2 を使用するようにアプライアンス集合を構成することができます。この標準により、通信で送信されるデータの優れた保護が確保されます。 [詳細...](#)

クライアント・プロパティ・ファイル内の credentialGeneratorProps プロパティを .NET 用にエンコード

FilePasswordEncoder ユーティリティを使用して Client.Net.properties ファイル内の credentialGeneratorProps プロパティの値をエンコードすることができます。 [詳細...](#)

エンタープライズ・データ・グリッド

エンタープライズ・データ・グリッドは eXtremeIO トランスポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトランスポートおよびシリアライゼーション形式により、Java クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。 [詳細...](#)

eXtreme Data Format (XDF)

XDF は MapSerializerPlugin プラグインに基づいており、現在は、IBM® eXtremeIO (XIO) の実行時に使用されるデフォルトのシリアライゼーション・テクノロジーです。

入門用サンプル

入門用サンプルを使用して、アプライアンス上のデータ・グリッドにアクセスする Java および .NET クライアント・アプリケーションを実行することができます。 [詳細...](#)

グリッド別名

REST ゲートウェイを使用して、グリッド別名を作成し、管理することができます。グリッド別名は、複数のグリッドを同時に取り込む必要がある場合に使用します。 [詳細...](#)

ホスト名管理

publishHost プロパティを使用して、公開されたホスト名を構成したりオーバーライドしたりすることができます。このプロパティを使用することで、クライアントとサーバー間での情報のやりとりをよりきめ細かく制御することが可能となり、そのため、データ・グリッド内のノード相互間の通信を最適化することができます。 [詳細...](#)

LDAP over SSL

Lightweight Directory Access Protocol (LDAP) ディレクトリーを確保して、アプライアンスにアクセスするユーザーの認証を行います。この LDAP ディレクトリーを確保するには、SSL 接続経由で LDAP の認証を行うようにアプライアンスを構成します。 [詳細...](#)

ニア・キャッシュの無効化

できるだけ速やかにニア・キャッシュから失効データが削除されるようにニア・キャッシュの無効化を構成することが

できます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化がトリガーされます。 [詳細...](#)

新しい xscmd ユーティリティー・コマンドとパラメーター

- **xscmd -c getNotificationFilter** コマンド: このコマンドを実行すると、メッセージ・センターからの新しい通知のための現行フィルターが表示されます。 [詳細...](#)
- **xscmd -c listenForNotifications** コマンド: このコマンドを実行すると、メッセージ・センターからの新しい通知が listen されます。 [詳細...](#)
- **xscmd -c setNotificationFilter** コマンド: このコマンドを実行すると、メッセージ・センターからの新しい通知のためのフィルターが作成されます。 [詳細...](#)
- **xscmd -c showLinkedDomains** コマンド: このコマンドを実行すると、ご使用のローカル・カタログ・サービス・ドメインにリンクされているカタログ・サービス・ドメインが表示されます。 [詳細...](#)
- **xscmd -c showNotificationHistory** コマンド: このコマンドを実行すると、イベント通知履歴の出力が表形式で表示されます。
- **xscmd -c showSessionSize** コマンド: このコマンドを実行すると、指定したセッションのサイズが表示されます。 [詳細...](#)
- **-to** または **--timeout** パラメーター: ネットワーク・ブラウン・アウトまたはシステム損失時にオペレーティング・システム・タイムアウトや他のネットワーク・タイムアウトが起こるまで待たないようにするためにタイムアウト値を減らすには、このパラメーターを指定します。 [詳細...](#)
- **-hc** または **--linkHealthCheck** パラメーター: プライマリー断片の持つ集合リンクの数が適切であるかどうかを確認するには、このパラメーターを **xscmd -c showLinkedPrimaries** コマンドで使用します。
- **xscmd -c listDisabledForPlacement** コマンド: 断片配置に対して使用不可にされた断片コンテナのリストを表示するには、このコマンドを実行します。
- **xscmd -c listIndoubts** コマンド: 未確定トランザクションのリストを表示するには、このコマンドを実行します。ある区画で起こった可能性があるロック・タイムアウト例外を解決したいときには、このコマンドを実行してください。 [詳細...](#)
- **xscmd -c showReplicationState** および **xscmd -c showDomainReplicationState** コマンド: ご使用のすべてのカタログ・サーバーまたはカタログ・サービス・ドメインについて改訂の状態を調べるには、これらのコマンドを実行します。 [詳細...](#)
- **xscmd -c showTransport** コマンド: カatalog・サービス・ドメインのトランスポート・タイプを表示するには、このコマンドを実行します。 [詳細...](#)

動的キャッシュ・データ・グリッドにおける **requestRetryTimeout** プロパティのオーバーライド
クライアント・プロパティ・ファイル内の **requestRetryTimeout** プロパティをオーバーライドしたり、タイムアウトまでの要求を実行できる時間 (ミリ秒) を指定したりするには、**com.ibm.websphere.xs.dynacache.request_retry_timeout_override** 動的キャッシュ・プロパティを使用します。 [詳細...](#)

アプライアンス・コマンド行インターフェース内からの xscmd ユーティリティーの実行

アプライアンス・コマンド行インターフェース内から **xscmd** ユーティリティーを実行することができます。アプライアンス・コマンド行インターフェース内からこのユーティリティーを呼び出す際に、セキュリティー・サーバー接続およびカタログ・サーバー接続についてはコマンド行引数を指定する必要はありません。 [詳細...](#)

秘密鍵生成

Web コンソールで固有の秘密鍵を指定するには、出荷時のデフォルト認証秘密鍵をオーバーライドします。 [詳細...](#)

更新された Simple Network Monitoring Protocol (SNMP) 統計

SNMP クライアントが使用できる SNMP データを指定するために WebSphere DataPower XC10 アプライアンス で提供される管理情報ベース (MIB) は、新しい統計で更新されるようになりました。新しい統計には、**applianceUsedBytes**、**applianceCapacity**、**gridUsedBytes**、および **gridCapacity** が含まれています。SNMP モニターを使用可能にし、この MIB ファイルをダウンロードする方法については、[アプライアンスでの SNMP モニターの使用可能化](#)を参照してください。

タイムアウト値および要求タイムアウト値の指定

eXtremeIO (XIO) トランスポートを使用するかオブジェクト・リクエスト・ブローカー・トランスポートを使用するかにかかわらず、接続あるいは要求が処理される時間を指定することができます。例えば、XIO の場合は、**xioTimeout** プロパティを使用してアウトバウンド・ソケット接続試行のタイムアウト値を指定でき、さらに、**xioRequestTimeout** プロパティを使用して要求がタイムアウトになるまでの要求の待ち時間 (秒数) を指定することができます。**requestRetryTimeout** プロパティは、XIO トランスポートまたは ORB トランスポートのいずれにも適用することができます。このプロパティを使用して、例外が発生した後いつまで要求の処理を続行するかを指定します。 [詳細...](#)

親トピック: [IBM WebSphere DataPower XC10 アプライアンスの概要](#)

リリース情報

製品のサポート Web サイト、製品資料、および製品の最新の更新、制限、および既知の問題へのリンクが提供されています。

- [最新の更新、制限、および既知の問題へのアクセス](#)
- [システムのアクセスおよびソフトウェア要件](#)
- [製品資料へのアクセス](#)
- [製品のサポート Web サイトへのアクセス](#)
- [IBM ソフトウェア・サポートへの連絡](#)

最新の更新、制限、および既知の問題へのアクセス

リリース情報は、製品のサポート・サイトの技術情報から入手できます。WebSphere® DataPower® XC10 アプライアンスのすべての技術情報のリストを参照するには、[サポート Web ページ](#)にアクセスしてください。ここで提供されているリンクをクリックすると、サポート Web ページで関連リリース・ノートが検索されます。関連リリース・ノートはリストとして返されます。

- リリース・ノートのリストを表示するには、[サポート Web ページ](#)にアクセスしてください。

システムのアクセスおよびソフトウェア要件

ハードウェアおよびソフトウェア要件は以下のページに記載されています。

- [システム要件](#)

製品資料へのアクセス

情報セット全体に関しては、[ライブラリー・ページ](#)にアクセスしてください。

製品のサポート Web サイトへのアクセス

最新の技術情報、ダウンロード、フィックス、およびその他のサポート関連情報を検索するには、[サポート・ページ](#)にアクセスしてください。

IBM ソフトウェア・サポートへの連絡

この製品で問題が発生した場合には、最初に以下のアクションを試行してください。

- 製品資料に記載されているステップを実行します。
- 関連資料をオンライン・ヘルプで検索します。
- エラー・メッセージをメッセージ解説書で検索します。

上記の方法で問題を解決できない場合、IBM® テクニカル・サポートに連絡してください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

親トピック: [トラブルシューティング](#)

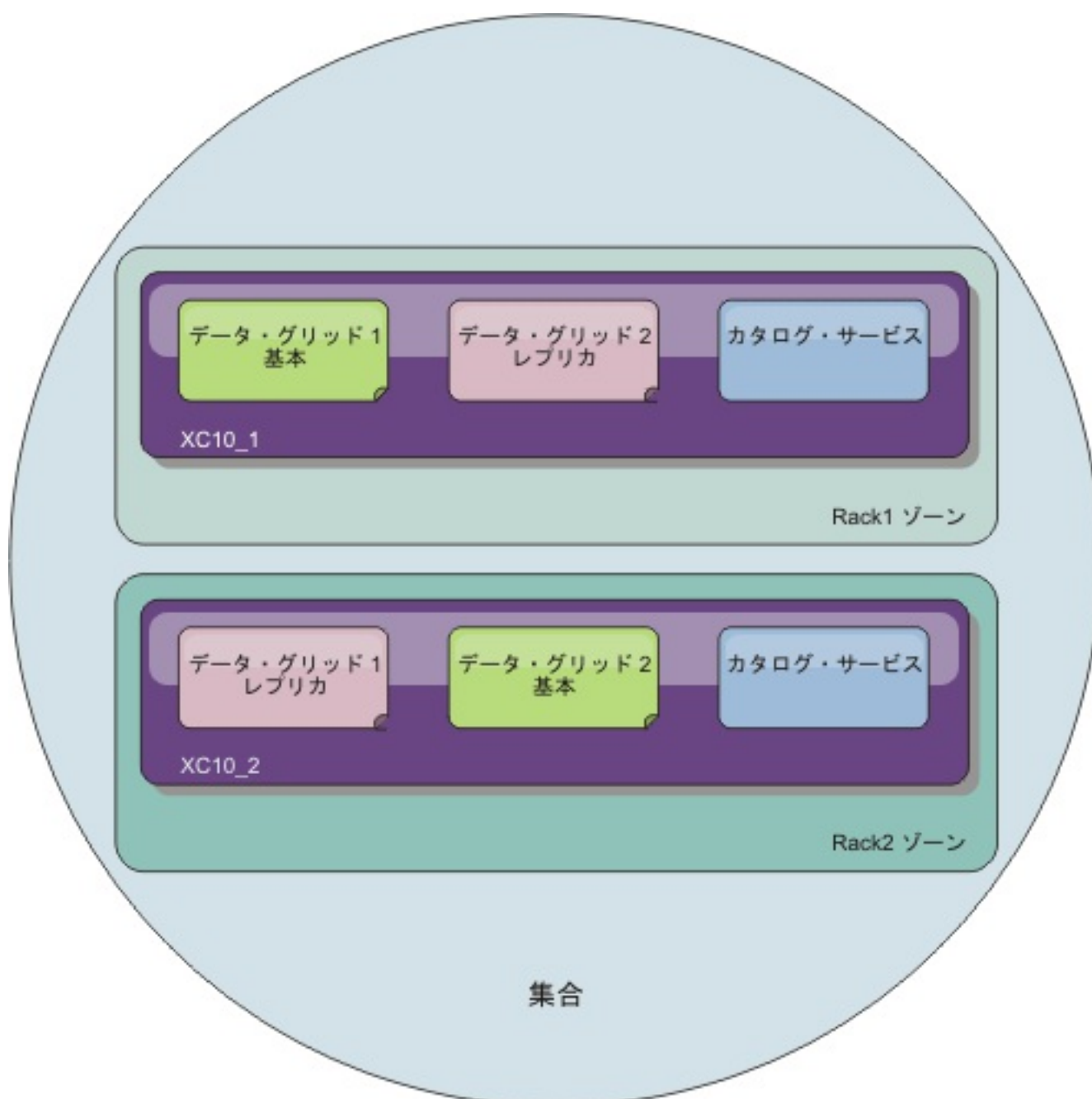
アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド

データ・グリッドは、特定のアプリケーションまたはアプリケーション群で使用するオブジェクトを保持するために作成されるストレージ・ユニットです。集合は、スケーラビリティおよび管理の目的でアプライアンスをグループ化します。ゾーンは、アプライアンスの物理的な場所を定義し、キャッシュ内でのデータの配置を決定するために使用されます。

アプライアンスのトポロジー

集合およびゾーンは、いずれも 1 つ以上の WebSphere® DataPower® XC10 アプライアンスに関連付けられています。各アプライアンスは 1 つの集合および 1 つのゾーンのメンバーとなり得ます。各アプライアンスは、キャッシュ・データを保持する複数のデータ・グリッドをホスティングします。

図 1. 集合およびゾーンのトポロジー



重要: データ・グリッドを高可用性にするために、2 つのアプライアンスが必要です。

集合とマルチマスター複製

マルチマスター複製は、複数のデプロイメント環境全体で連続可用性を確保するための技法です。マルチマスターのトポロジーは、複数の集合を作成して、それらをリンクで接続することで WebSphere DataPower XC10 アプライアンスに実装できます。集合を定義すると、次の情報が集合内のアプライアンス間で共有されます: データ・グリッド、モニター情報、集合およびゾーンのメンバー、およびユーザー。これらの情報のいずれかを更新すると、行われた変更が集合内の他のすべてのアプライアンスで保持されます。カタログ・サービスはアプライアンス間の通信を可能にします。カタログ・サービスはカタログ・サーバーのグループです。集合内の各アプライアンスは、集合ごとに 3 つのカタログ・サーバーを限度に、カタログ・サーバーを実行します。集合内に 4 つ以上のアプライアンスがある場合、カタログ・サービスは集合に追加された最初の 3 つのアプライアンス上で実行されます。カタログ・サーバーを持つアプライアンスを集合から除去する場合や、カタログ・サーバーを持つアプライアンスが使用できなくなった場合は、次に集合に追加されたアプライアンスがカタログ・サーバーを実行します。カタログ・サーバーは他のアプライアンスにフェイルオーバーしません。

アプライアンスを集合に追加するには、別のアプライアンスの集合構成パネルにそのアプライアンスのホスト名および共通鍵を追加します。この構成は集合内のどのアプライアンスでも作成できますが、それは集合のメンバーシップが集合メンバー間で保持されるためです。

アプライアンスは 1 つの集合内にのみ存在できます。既にある集合に存在するアプライアンスを別の集合に追加することはできません。2 つの集合を結合して 1 つの集合にすることもできません。別々の集合のアプライアンスを結合するには、各アプライアンスをそれぞれの集合から除去し、各アプライアンスをスタンドアロンにする必要があります。そうすれば、すべてのアプライアンスを含む新しい集合を作成できます。

集合を使用してほとんどの構成の変更が行える一方で、「アプライアンス」 > 「アプライアンスの設定」および「アプ

ライセンス」 > 「トラブルシューティング」の各パネルで設定を変更するためには、所定のアプライアンスにログインしなければなりません。

ゾーン

ゾーンは 例えば市区町村や研究所内のラックの場所など、アプライアンスの 物理的な場所と関連付けられます。ゾーンは、カタログ・サービスでご利用の データ・グリッド内のデータの保管場所を定義するのに 役立ちます。例えば、ある 1 つのゾーンにデータ・グリッド (data grid) の基本情報が保管されている場合に、レプリカのデータは別のゾーンにあるアプライアンスに保管されます。この構成では、データ・グリッド (data grid) の 基本を保持するアプライアンスに障害が起ると、基本からレプリカに フェイルオーバーが発生します。

データ・グリッド

データ・グリッドはアプリケーション用のオブジェクトを保持します。オブジェクトをキャッシングすることで、アプリケーションのパフォーマンスが向上します。 データ・グリッドには 3 つのタイプがあります。

単純データ・グリッド (data grid)

単純データ・グリッドは、データをキー値のペアで保持します。例えば、データベース照会の結果を単純 データ・グリッド (data grid) に保管できます。単純データ・グリッド (data grid) を実装するには、ObjectMap API を使用します。ObjectMap API は、Java™ Maps と同じように動作します。

セッションデータ・グリッド (data grid)

WebSphere Application Server セッションを使用する場合は、セッション管理データ用のアプライアンス上でセッションデータ・グリッド (data grid) を使用するようにアプリケーションを構成できます。新規のアプリケーションをインストールする際に、セッション データ・グリッド (data grid) を使用するようにアプリケーションを構成できます。既存のアプリケーションまたはサーバーの設定を更新して、アプライアンス上でセッションデータ・グリッド (data grid) を使用することもできます。

動的キャッシュデータ・グリッド (data grid)

WebSphere Application Server の動的キャッシュから取り出したデータを保管するために、アプライアンス上の動的キャッシュデータ・グリッド (data grid) を使用できます。動的キャッシュ API を使用して作成されたアプリケーションや、サーブレットのようなコンテナ・レベル・キャッシングを使用するアプリケーションを使用可能にして、アプライアンスをキャッシュ・プロバイダーとして使用することができます。その結果として、アプリケーション・サーバーによって使用されるメモリーが少なくなります。すべてのキャッシュ・データがアプライアンスにオフロードされ、アプリケーション・サーバーのメモリーにはもう保管されません。

データ・グリッド・レプリカ

特定のデータ・グリッド (data grid) のレプリカのターゲット数を定義できます。集合に少なくとも 2 つのアプライアンスがあるときに、レプリカは作成されます。アプライアンスが 1 つの場合、レプリカは作成されません。集合内のアプライアンス数が n の場合、1 つのアプライアンスが基本データ・グリッドをホスティングするため、レプリカの最大数は $n-1$ になります。レプリカのターゲット数が現行の $n-1$ よりも大きい場合は、アプライアンスを集合に追加するときに、追加のレプリカを作成できます。レプリカの数、今後必要となる可能性があるレプリカの最大数に設定しておくことを検討してください。レプリカ設定を編集すると、データ・グリッドの内容が消去されます。今後のレプリカの数、考慮してこの値を設定してください。新しいアプライアンスが集合に追加されると、追加のレプリカが作成されます。基本およびレプリカのデータ・グリッドは、集合内のすべてのアプライアンス間で均一に分散またはストライピングされます。新しいアプライアンスが集合に追加されると、基本およびレプリカのデータ・グリッドを分散するための再バランシングが行われます。

レプリカは、同期レプリカまたは 非同期レプリカのいずれかになります。同期レプリカは、基本データ・グリッド (data grid) 上の トランザクションの一部として更新を受け取ります。非同期レプリカは、基本データ・グリッド (data grid) 上の トランザクションがコミットされた後に 更新されます。同期レプリカはデータの一貫性を保証しますが、非同期レプリカに比べると要求への応答時間が長くなります。非同期レプリカは データの一貫性においては同様の保証ができませんが、トランザクションを より速く完了させることができます。データ・グリッド (data grid) にはデフォルトで 1 つの非同期レプリカがあります。配置アルゴリズムによって、レプリカの配置場所が制御されます。

マップ

マップは、キー値のペアにあるデータ・グリッドのデータを含むデータ構造です。1 つのデータ・グリッドは、データ・グリッドおよびデータ・グリッド・レプリカに存在する複数のマップを持つことができます。

ご利用のクライアント・アプリケーションを特定の名前の付いたマップに接続させることで、データ・グリッド内に追加のマップを作成できます。動的マップが自動的に作成されます。

集合リンク

障害の誤検出が起こる可能性があるため、単一の集合が信頼できないネットワーク上に広がってはなりません。しかし、信頼できないネットワークへの接続を持つアプライアンス間で、データ・グリッドのデータを複製する必要が生じる場合もあります。 このタイプのトポロジーを使用することになる一般的なシナリオには以下のようなものがあります。

- 1 つの集合がアクティブでもう 1 つの集合がバックアップとして使用されているデータ・センター間での災害復旧
- 地理的に近いクライアントのために、すべての集合がアクティブになっている地理的に分散されたデータ・センター

—。

2つの集合を接続すると、同じ名前を持つすべてのデータ・グリッドが集合間で非同期的に複製されます。これらのデータ・グリッドは、各集合内に同じ数のレプリカを持ち、同じ動的マップ構成を持っていない限りなりません。

[マルチマスター複製を実装する集合のリンクのトポロジー](#)

複数の集合を組み込んだデプロイメントのトポロジーを選択する際、いくつかの異なるオプションがあります。マルチマスター複製のトポロジーは、複数の集合を作成して、それらをリンクで接続することで DataPower XC10 アプライアンス に実装できます。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

関連タスク:

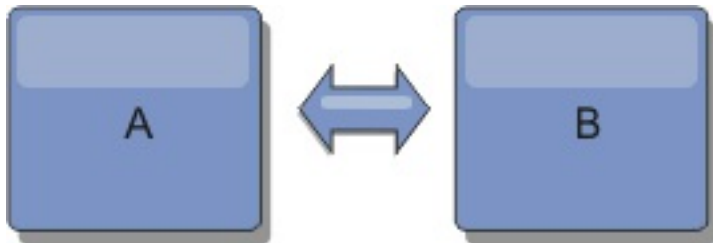
[集合間のマルチマスター複製の構成](#)

マルチマスター複製を実装する集合のリンクのトポロジー

複数の集合を組み込んだデプロイメントのトポロジーを選択する際、いくつかの異なるオプションがあります。マルチマスター複製のトポロジーは、複数の集合を作成して、それらをリンクで接続することで DataPower® XC10 アプライアンス に実装できます。

集合 を接続するリンク

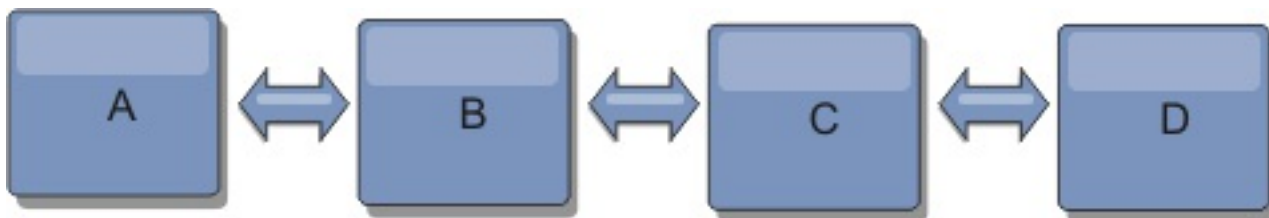
複製データ・グリッドのインフラストラクチャーは、集合間を双方向のリンクで接続したカタログ・サービス・ドメインのグラフです。リンクを使用して、2つの集合はデータ変更内容をやりとりできます。例えば、最も単純なトポロジーは、カタログ・サービス・ドメイン間に単一のリンクを持つ1対の集合です。集合は、左から A、B、C というようにアルファベット順で指定されています。リンクは、遠距離にわたる広域ネットワーク (WAN) を経由する場合があります。リンクが遮断されたとしても、いずれかの集合 (collective) でまだデータを変更できます。トポロジーは、リンクが集合と再接続したときに変更を調整します。ネットワーク接続が中断されると、リンクは自動的に再接続しようとします。



リンクをセットアップすると、この製品はまず、すべての集合 (collective) を同一にしようと試みます。次に、いずれかの集合 (collective) で変更が発生すると、eXtreme Scale は同一の状態を維持するよう試みます。目標は、各集合 (collective) がリンクで接続されたすべての他の集合 (collective) の正確なミラーになることです。集合間の複製リンクは、1つの集合 (collective) で行われたすべての変更を確実に他の集合にコピーするのに役立ちます。

ライン・トポロジー

ライン・トポロジーはこのような単純なデプロイメントですが、かなりのリンク品質を実証します。まず、変更を受け取るために、集合 (collective) は直接すべての他の集合 (collective) に接続する必要がありません。集合 (collective) B は集合 (collective) A から変更をプルします。集合 (collective) C は、集合 A と C を接続する集合 (collective) B を介して集合 (collective) A から変更を受信します。同様に、集合 (collective) D は集合 (collective) C を介して別の集合から変更を受信します。この機能により、変更配布の負荷が変更のソースから離れた場所に分散できます。



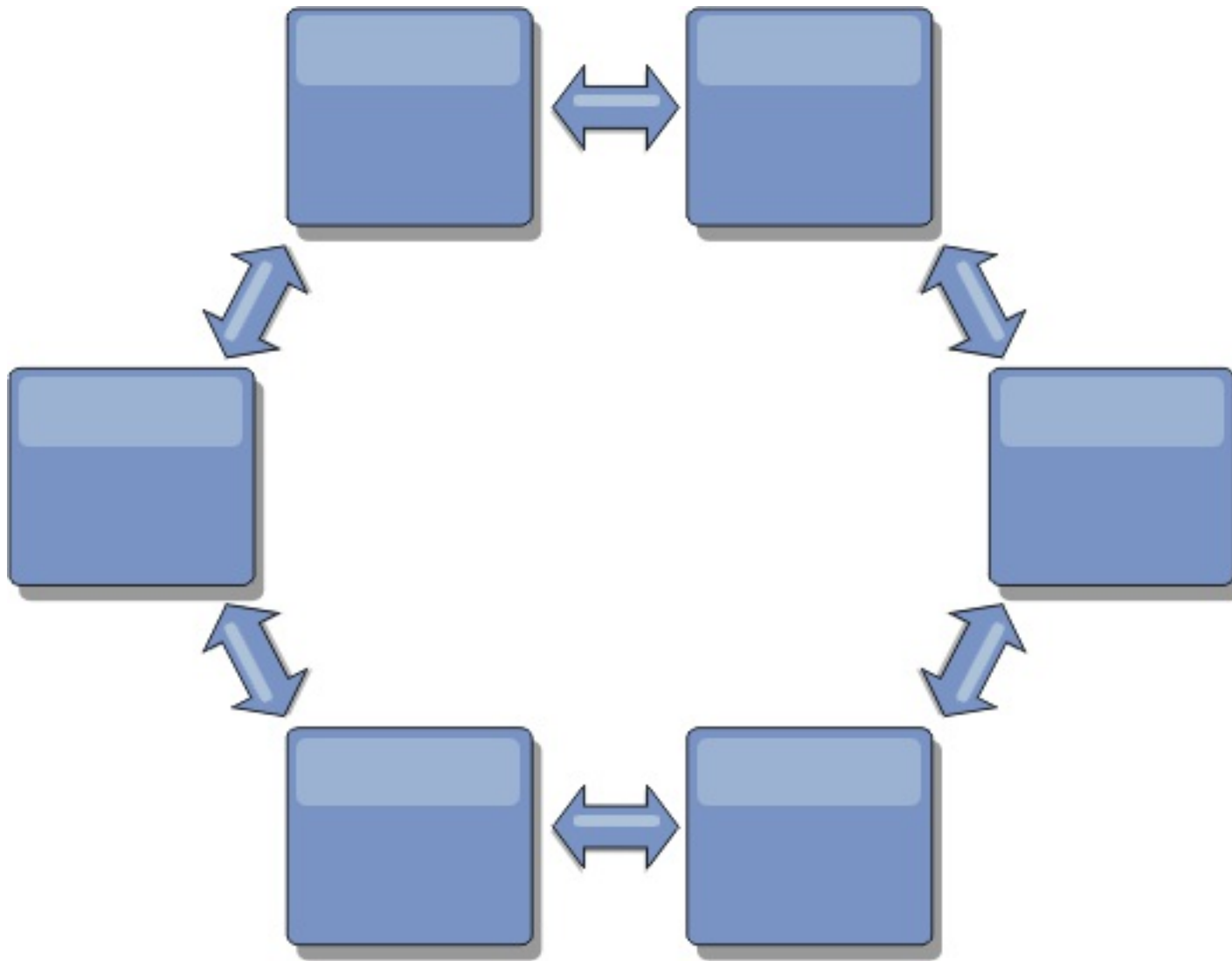
集合 (collective) C に障害が起こった場合、以下のアクションの発生が考えられることに注意してください。

1. 集合 (collective) D は、集合 (collective) C が再開されるまで孤立します。
2. 集合 (collective) C は、集合 (collective) A のコピーである集合 (collective) B と自分自身を同期させます。
3. 集合 (collective) D は、集合 (collective) C を使用して、集合 (collective) A と B で発生した変更と自分自身を同期させます。これらの変更は最初は、集合 (collective) D が孤立していた間 (集合 (collective) C がダウンしていた間) に発生しました。

最終的に、集合 A、B、C、および D はすべて、互いのドメインと再び同一になります。

リング・トポロジー

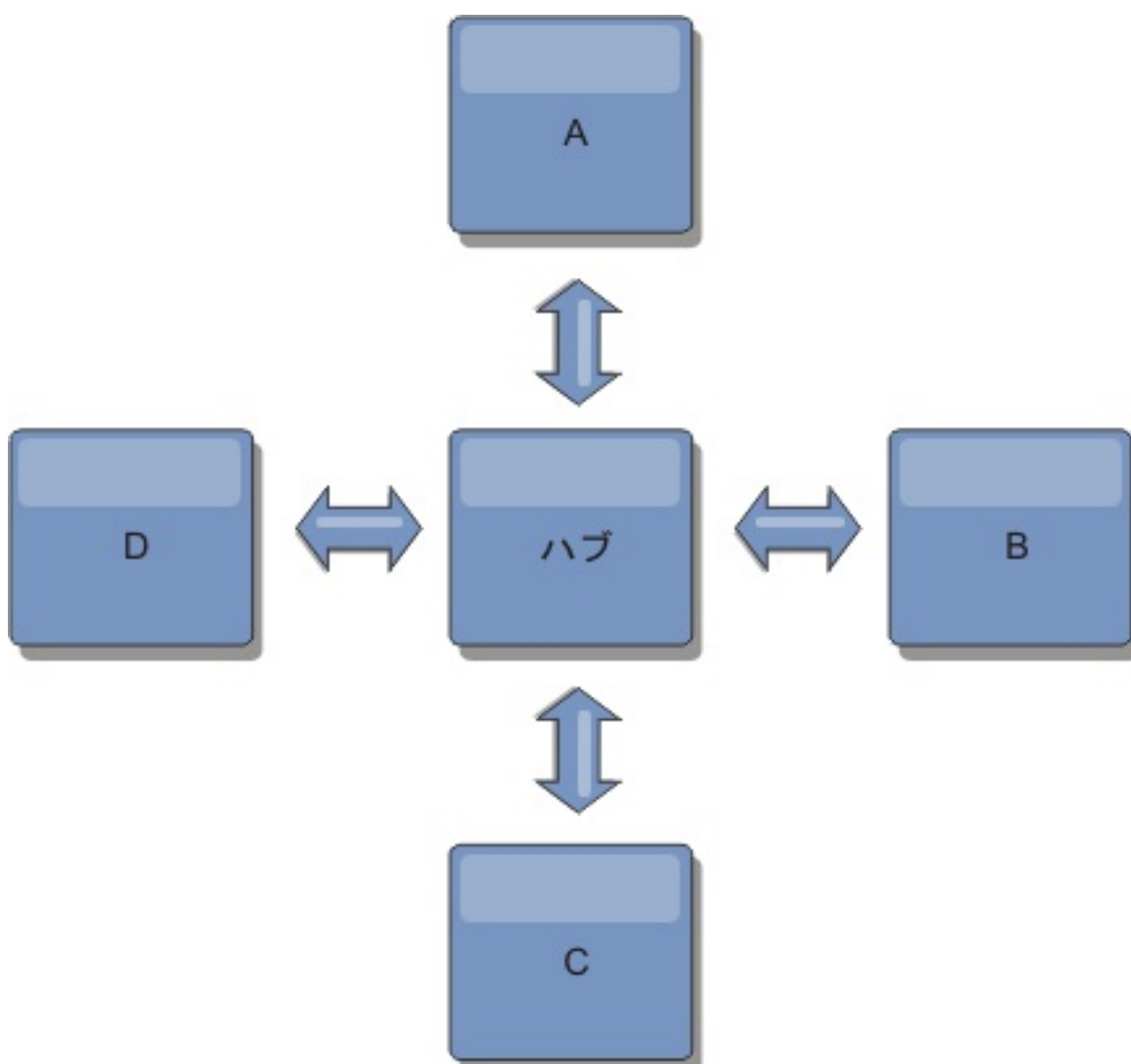
リング・トポロジーは、より回復力のあるトポロジーの例です。集合 (collective) または単一リンクに障害が起こった場合でも、残った集合がまだ変更を取得できます。その集合は、障害から離れて、リングの周りを回ります。リング・トポロジーの大きさには関係なく、各集合 (collective) は他の集合とのリンクを最大2つ持ちます。変更を伝搬するための待ち時間は長くなる場合があります。特定の集合 (collective) での変更は、すべての集合にその変更が反映されるまで、複数のリンクを経由して伝搬する必要がある場合があります。ライン・トポロジーにも同じ特性があります。



リングの中心に置いたルート・集合 (collective) を使用した、より洗練されたリング・トポロジーをデプロイすることも可能です。ルート・集合 (collective) は、調整の中心点として機能します。他の集合は、ルート・集合 (collective) で生じた変更に対する調整のリモート・ポイントとして働きます。ルート・集合 (collective) は、集合間の変更をアービトレーションすることができます。ルート・集合 (collective) を囲む複数のリングがリング・トポロジーに含まれている場合、集合 (collective) は最も内側にあるリング間の変更のみをアービトレーションすることができます。ただし、アービトレーションの結果は他のリングの集合にも広がります。

ハブ・アンド・スポーク・トポロジー

ハブ・アンド・スポーク・トポロジーでは、ハブ・集合 (collective) を経由して変更が伝搬します。ハブは指定される唯一の中間集合 (collective) であるため、ハブ・アンド・スポーク・トポロジーでは待ち時間が短縮されます。ハブ・集合 (collective) は、リンク経由ですべてのスポーク・集合 (collective) に接続されています。ハブは、集合間の変更を配布します。ハブは、衝突に対して調整のポイントとして機能します。更新頻度の高い環境では、同期を保つために、スポークよりも多くのハードウェア上でハブを稼働する必要がある場合があります。WebSphere® DataPower XC10 アプライアンスは、直線的に拡大するように設計されています。つまり、問題なく、必要に応じてハブをさらに大きくすることができます。ただし、ハブに障害が起こった場合は、変更はハブが再始動するまで配布されません。スポーク・集合上の変更は、ハブが再接続された後に配布されます。



また、完全に複製したクライアントを使用したストラテジー、すなわち、ハブとして稼働しているサーバーのペアを使用

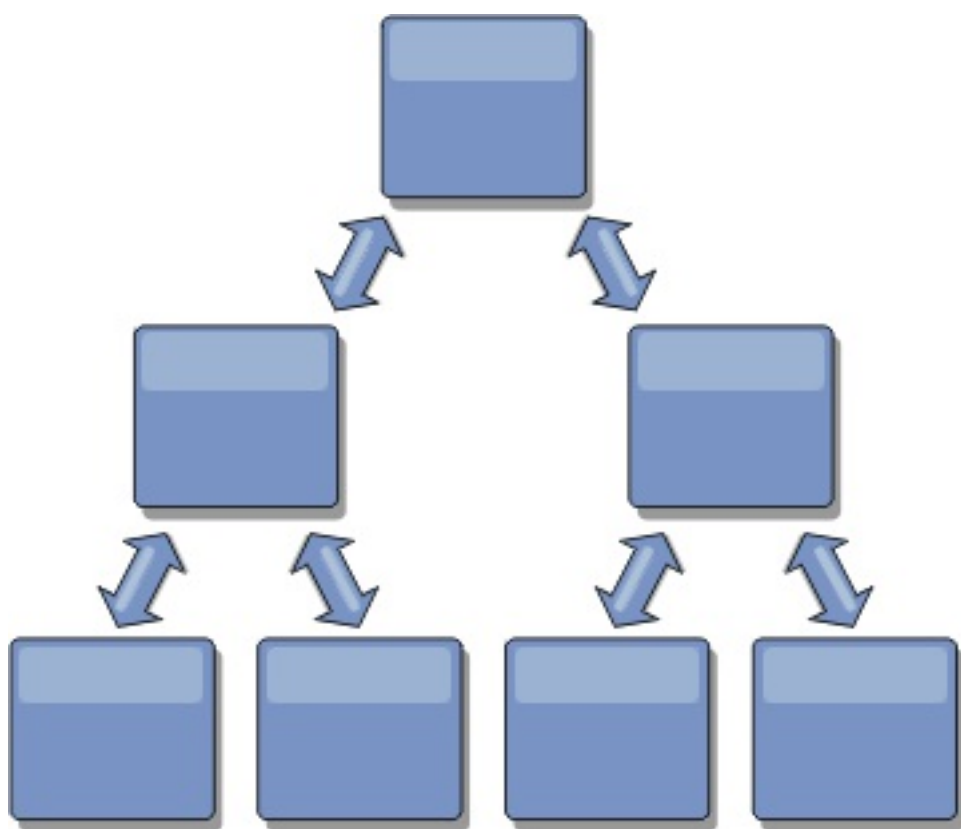
するトポロジーのバリエーションを使用することもできます。各クライアントは、クライアント JVM 内に、必要なものを完備した単一コンテナ・データ・グリッドとカタログを作成します。クライアントは、そのデータ・グリッドを使用してハブ・カタログに接続します。この接続により、クライアントはハブへの接続を取得すると、すぐにハブと同期するようになります。

クライアントによって行われた変更は、クライアントに対してローカルで、非同期でハブに複製されます。ハブはアービトレーション・集合 (collective) として機能し、すべての接続されたクライアントに変更を配布します。完全複製クライアントのトポロジーは、OpenJPA などのオブジェクト・リレーショナル・マッパーに信頼性の高い L2 キャッシュを提供します。変更はハブを介してクライアント JVM 間に迅速に配布されます。キャッシュ・サイズを使用可能なヒープ・スペース内に含むことができる場合、このトポロジーは L2 のこのスタイルにとって信頼できるアーキテクチャーです。

必要であれば、複数の区画を使用して、複数の JVM 上にハブ・集合 (collective) を拡張します。すべてのデータはまだ単一のクライアント JVM に収まらなければならないため、複数の区画を使用してハブの容量を増加させ、変更の配布とアービトレーションを行います。ただし、複数の区画を使用しても、単一集合 (collective) の容量は変更されません。

ツリー・トポロジー

非循環有向ツリーを使用することもできます。非循環ツリーには循環やループはなく、有向セットアップにより、リンクの存在は親と子の間にのみ制限されます。この構成は、多くの集合を含むトポロジーで役立ちます。これらのトポロジーでは、すべての接続可能なスポークに接続されている中央ハブを使用することは実用的ではありません。また、このタイプのトポロジーは、ルート・集合 (collective) を更新することなく子集合を追加する必要がある場合にも便利です。



ツリー・トポロジーでもまだ、ルート・集合 (collective) に調整の中心点を置くことができます。第 2 レベルはまだ、それらの下の集合 (collective) で生じた変更に対する調整のリモート・ポイントとして機能します。ルート・集合 (collective) は、第 2 レベルにある集合間の変更のみをアービトレーションすることができます。それぞれが各レベルで N 個の子を持つ、n 進ツリーを使用することもできます。それぞれの集合 (collective) は、n 個のリンクに接続します。

完全複製クライアント

このトポロジー変化には、ハブとして稼働する 1 対のサーバーが含まれます。各クライアントは、クライアント JVM 内に、必要なものを完備した単一コンテナ・データ・グリッドとカタログを作成します。クライアントは、そのデータ・グリッドを使用してハブ・カタログに接続します。これにより、クライアントはハブへの接続を取得すると、すぐにハブと同期するようになります。

クライアントによって行われた変更は、クライアントに対してローカルで、非同期でハブに複製されます。ハブはアービトレーション・集合 (collective) として機能し、すべての接続されたクライアントに変更を配布します。完全複製クライアントのトポロジーは、OpenJPA などのオブジェクト・リレーショナル・マッパーに適した L2 キャッシュを提供します。変更はハブを介してクライアント JVM 間に迅速に配布されます。キャッシュ・サイズをクライアントの使用可能なヒープ・スペース内に含むことができる限り、このトポロジーは L2 のこのスタイルに適したアーキテクチャーです。

必要であれば、複数の区画を使用して、複数の JVM 上にハブ・集合 (collective) を拡張します。すべてのデータはまだ単一のクライアント JVM に収まらなければならないため、複数の区画を使用してハブの容量を増加させ、変更の配布とアービトレーションを行います。ただし、単一集合 (collective) の容量は変更しません。

親トピック: [アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連タスク:

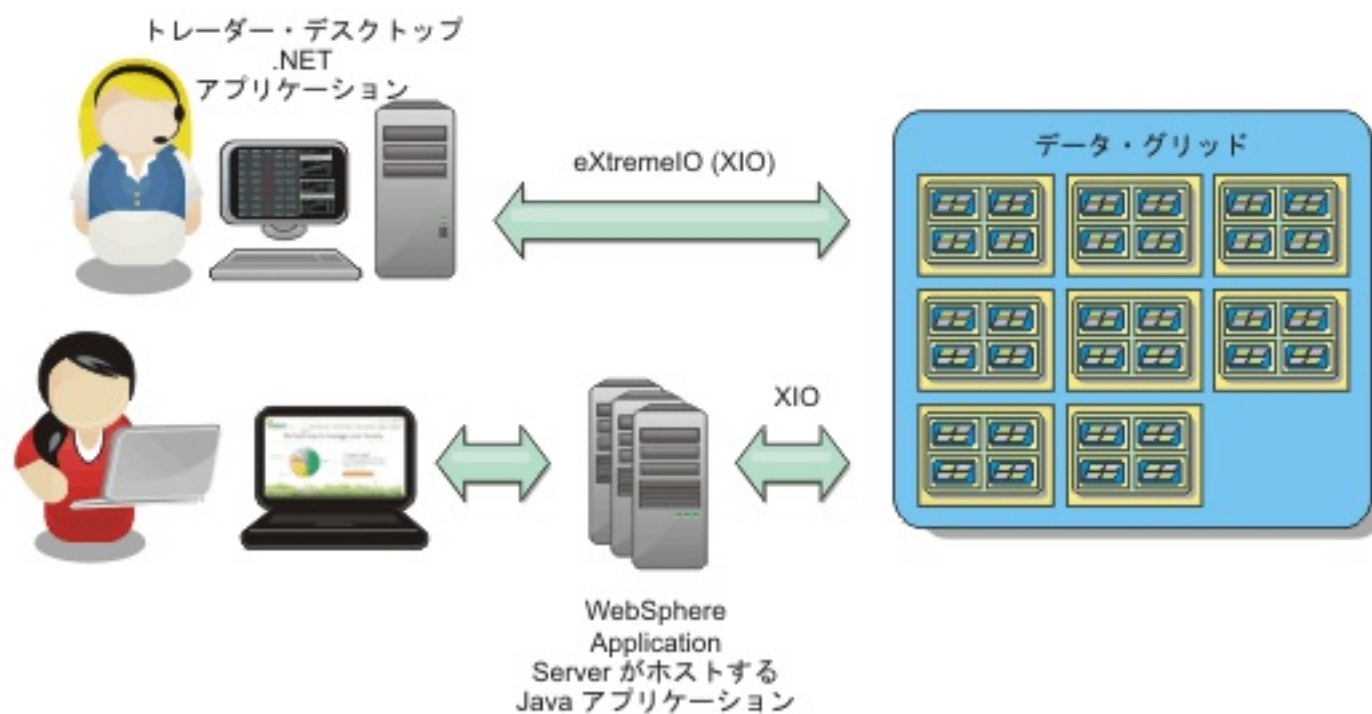
[集合間のマルチマスター複製の構成](#)

エンタープライズ・データ・グリッドの概要

エンタープライズ・データ・グリッドは eXtremeIO トラnsポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトラnsポートおよびシリアライゼーション形式により、Java™ クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。

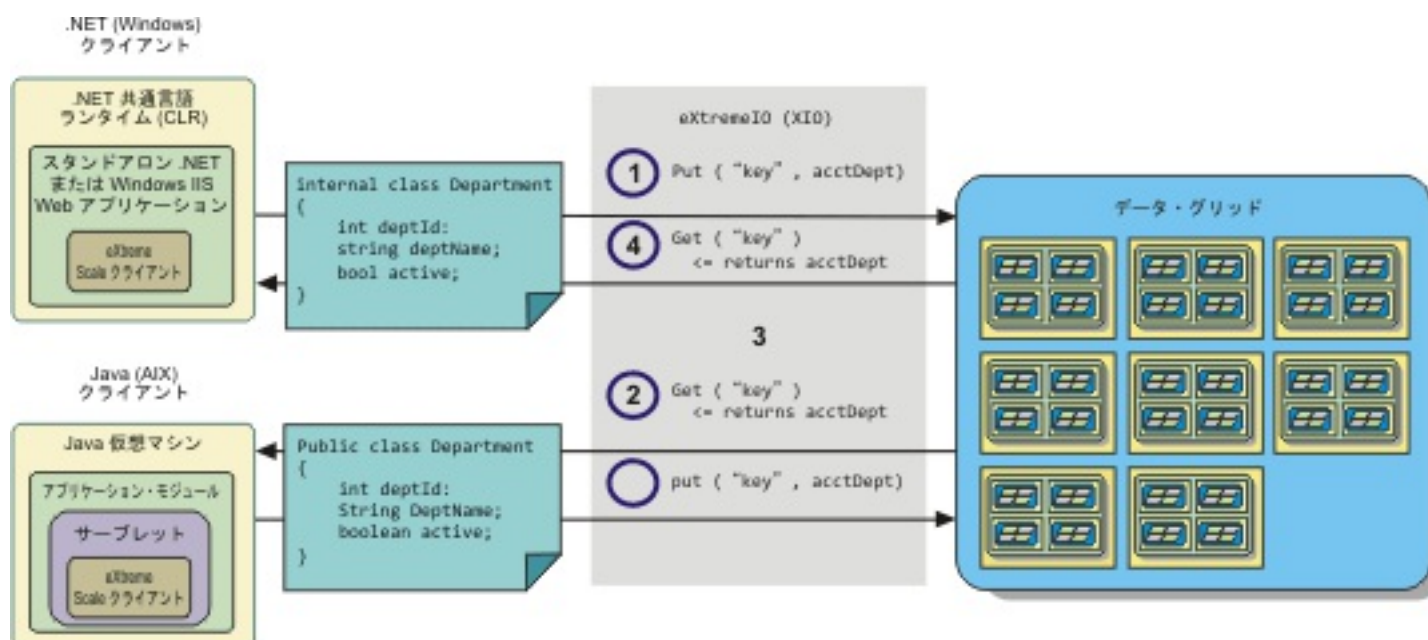
エンタープライズ・データ・グリッドにより、異なるプログラミング言語で書かれた、さまざまなタイプのアプリケーションを作成して、データ・グリッド内の同じオブジェクトにアクセスすることができます。これまでのリリースでは、データ・グリッド・アプリケーションは Java プログラミング言語でしか書くことができませんでした。エンタープライズ・データ・グリッド機能を使用すれば、Java アプリケーションと同じデータ・グリッドでオブジェクトの作成、検索、更新、および削除を行える .NET アプリケーションを書くことができます。

図1. エンタープライズ・データ・グリッドの概要



異なるアプリケーションにまたがるオブジェクト更新

図2. エンタープライズ・データ・グリッド・オブジェクト更新フロー



1. .NET クライアントが .NET 形式のデータをデータ・グリッドに保存します。
2. データは汎用形式で保管されるので、Java クライアントからこのデータが要求されたとき、このデータを Java 形式に変換することができます。
3. Java クライアントがデータを更新して再保存します。
4. .NET クライアントが更新されたデータにアクセスします。そのとき、データは .NET 形式に変換されます。

トラnsポート・メカニズム

eXtremeIO (XIO) はクロスプラットフォーム・トラnsポート・プロトコルです。XIO は Java の制約を受けるオブジェクト・リクエスト・ブローカー (ORB) に取って代わるものです。ORB の場合は、WebSphere® DataPower® XC10 アプライアンスは Java ネイティブ・クライアント・アプリケーションと密接に結びついています。XIO は、特にデータ・キャッシングを対象とし、異なるプログラミング言語で書かれたクライアント・アプリケーションがデータ・グリッドに接続できるようにする、カスタマイズされたトラnsポート・メカニズムです。

シリアライゼーション形式

eXtreme データ形式 (XDF) はクロスプラットフォームのシリアライゼーション形式です。XDF は、ObjectGrid 記述子 XML ファイルで copyMode 属性値が COPY_TO_BYTES であるマップでの Java シリアライゼーションに取って代わります。XDF を使用すれば、パフォーマンスが向上し、データがよりコンパクトになります。さらに、XDF の導入により、異なるプログラミング言語で書かれたクライアント・アプリケーションが同じデータ・グリッドに接続できるようになります。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

親トピック: [シナリオ: エンタープライズ・データ・グリッドの構成](#)

次のトピック: [IBM eXtremeIO \(XIO\) の構成](#)

トランザクション処理の概要

WebSphere® eXtreme Scale クライアント は、データとの相互作用のメカニズムとしてトランザクションを使用します。

Java

Java™ アプリケーションにおけるトランザクション処理

データとの相互作用のために、アプリケーション内のスレッドは、独自のセッションを必要とします。アプリケーションがスレッド上で ObjectGrid を使用する必要がある場合、ObjectGrid.getSession メソッドの 1 つを呼び出してセッションを取得します。このセッションを使用すると、アプリケーションは ObjectGrid マップに保管されているデータの処理を行うことができます。

アプリケーションが Session オブジェクトを使用する場合、そのセッションはトランザクションのコンテキスト内に必要があります。Session オブジェクトに対する begin メソッド、commit メソッド、および rollback メソッドにより、トランザクションは、開始してコミット、あるいは開始してロールバックを行います。また、アプリケーションは自動コミット・モードで稼働することもできます。このモードでは、マップに対して操作が実行されるたびに、Session は自動的にトランザクションを開始してコミットします。自動コミット・モードでは、複数の操作を単一のトランザクションにグループ化することができません。複数の操作のバッチを作成して単一のトランザクションにした場合、自動コミット・モードは低速のオプションとなります。ただし、単一の操作しか含まないトランザクションの場合は、自動コミット・モードの方が速いオプションとなります。

アプリケーションがセッションを終了したときは、オプションの Session.close() メソッドを使用してセッションを閉じます。セッションを閉じると、セッションがヒープから解放され、getSession() メソッドの後続の呼び出しの再利用が可能となるため、パフォーマンスが向上します。

.NET

.NET アプリケーションにおけるトランザクション処理

データとの相互作用のために、アプリケーション内の各スレッドは、独自のトランザクション・オブジェクトを必要とします。アプリケーションでスレッドに対して IGrid インターフェースを使用するには、次のいずれかのメソッドを呼び出します。

- IGrid.GetGridMapPessimisticAutoTx
- IGrid.GetGridMapPessimisticTx

これらのメソッドを呼び出すと、固有のトランザクション・オブジェクトを持つ IGridMap オブジェクトを取得します。この IGridMap オブジェクトを使用すると、アプリケーションは IGrid マップに保管されているデータの処理を行うことができます。アプリケーションが IGridMapPessimisticTx オブジェクトを使用する場合、そのデータ・グリッド操作はトランザクションのコンテキスト内に必要があります。アプリケーションは、IGridTransaction オブジェクトに対する begin、commit、および rollback メソッドを使用して、トランザクションを開始してコミットしたり、開始してロールバックしたりします。また、アプリケーションは自動コミット・モードで稼働することもできます。このモードでは、マップに対して操作が実行されるたびに、IGridMapPessimisticAutoTx は自動的にトランザクションを開始してコミットします。自動コミット・モードでは、複数の操作を単一のトランザクションにグループ化することができません。複数の操作のバッチを作成して単一のトランザクションにした場合、自動コミット・モードは低速のオプションとなります。ただし、単一の操作しか含まないトランザクションの場合は、自動コミット・モードの方が速いオプションとなります。

アプリケーションが IGridMap インスタンスを終了したときは、IGridMap オブジェクトを破棄してください。このオブジェクトを破棄すると、関連するトランザクション・オブジェクトが閉じられます。その結果、GetGridMapPessimisticAutoTx メソッドおよび GetGridMapPessimisticTx メソッドの後続の呼び出しで既存の空きトランザクション・オブジェクトを再利用でき、これによってパフォーマンスが向上します。

トランザクション

トランザクションには、データ保管および操作に関して多くの利点があります。トランザクションを使用すれば、同時変更からデータ・グリッドを保護したり、複数の変更を 1 つの並行ユニットとして適用したり、データを複製したり、変更に対するロックのライフサイクルを実装したりすることができます。

ロック・ストラテジー

ロック・ストラテジーには、ペシミスティック、オプティミスティック、およびロックなしがあります。

ロック・タイプ

ペシミスティック・ロックおよびオプティミスティック・ロックを使用する場合、整合性を保つために、共有 (S) ロック、アップグレード可能 (U) ロック、および排他的 (X) ロックが使用されます。ペシミスティック・ロックを有効にしている場合、ロックとその動作について理解することが重要です。オプティミスティック・ロックではロックを保持しません。ロックにはライフサイクルがあります。いろいろなタイプのロックがさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

デッドロック

2つのトランザクションが同じキャッシュ・エントリを更新しようとするするとデッドロックが起こることがあります。

データ・アクセスおよびトランザクション

WebSphere eXtreme Scale クライアント はトランザクションを使用します。アプリケーションがデータ・グリッドへの接続を獲得したら、ユーザーはデータ・グリッド内のデータにアクセスし、データを操作することができます。

トランザクション分離

3つのトランザクション分離レベル (反復可能読み取り、読み取りコミット済み、および読み取りアンコミット) のうち 1つを使用して、各キャッシュ・マップ内の整合性を保つロック・セマンティクスを調整することができます。

Java

2 フェーズ・コミットとエラー・リカバリー

2 フェーズ・コミット・プロトコルは、分散トランザクションに関与するすべての区画を、そのトランザクションをコミットするかロールバックするかに基づいて調整します。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

トランザクション

トランザクションには、データ保管および操作に関して多くの利点があります。トランザクションを使用すれば、同時変更からデータ・グリッドを保護したり、複数の変更を1つの並行ユニットとして適用したり、データを複製したり、変更に対するロックのライフサイクルを実装したりすることができます。

トランザクションが開始すると、WebSphere® eXtreme Scale クライアントは、そのトランザクションが使用するキーと値のペアの現在の変更またはコピーを保持する特別な差分マップを割り振ります。通常、キーと値のペアにアクセスすると、アプリケーションがその値を受け取る前に、値のコピーが作成されます。Java™ アプリケーションでは、この差分マップが挿入、更新、取得、除去などの操作によるすべての変更を追跡します。.NET アプリケーションでは、この差分マップが追加操作、置換操作、取得操作、および除去操作による変更を追跡します。キーは不変のものに見なされているため、コピーされません。トランザクションがロールバックされると、差分マップ情報が破棄され、エントリーに対するロックが解除されます。トランザクションをコミットすると、変更がマップに適用され、ロックが解除されません。

Java Java アプリケーションで ObjectTransformer オブジェクトが指定されると、このオブジェクトが値をコピーするために使用されます。トランザクションがオプティミスティック・ロックを使用している場合は、トランザクションのコミット時に、以前の値のイメージも比較のために追跡されます。

Java Java アプリケーションでオプティミスティック・ロックが使用されている場合は、WebSphere eXtreme Scale クライアントが値の変更前イメージ・バージョンとマップ内の値を比較します。トランザクションをコミットするには、これらの値が一致している必要があります。こうした比較によって複数バージョンのロック体系が可能になりますが、トランザクションがそのエントリーにアクセスすると、代わりに2つのコピーが作成されます。すべての値が再度コピーされ、新しいコピーがマップに保管されます。WebSphere eXtreme Scale クライアントは、コミット後に値へのアプリケーション参照を変更するアプリケーションから自身を保護するために、このコピーを実行します。

情報の複数のコピーを使用しないようにできます。アプリケーションは、並行性を制限する代償としてオプティミスティック・ロックの代わりにペシミスティック・ロックを使用することで、コピーを節約できます。コミット後に値を変更しないことにアプリケーションが同意すれば、コミット時の値のコピーも回避することができます。

.NET 注: .NET アプリケーションはペシミスティック・ロックのみをサポートします。

トランザクションの利点

トランザクションを使用するのは、以下の理由からです。

トランザクションを使用して、以下の操作を行うことができます。

- 例外が発生した場合や、ビジネス・ロジックにより状態変更を元に戻す必要がある場合に、変更をロールバックします。
- コミット時に複数の変更をアトミック単位で適用する
- データに対するロックの保持および解除を行い、コミット時に複数の変更をアトミック単位で適用します。
- 同時変更からスレッドを保護します。
- 変更に対するロックのライフサイクルを実装します。
- アトミック単位のレプリカ生成をします。

トランザクション・サイズ

トランザクションは、特にレプリカ生成の場合には、大きいほど効果的です。ただし、トランザクションが大きくなると、エントリーのロックの保持時間が長くなるため、並行性に悪影響を及ぼすことがあります。大きなトランザクションを使用すると、レプリカ生成のパフォーマンスが向上する場合があります。このパフォーマンスの向上は、マップを事前にロードする場合には重要です。さまざまなバッチ・サイズで実験を行い、使用するシナリオに最適なサイズを判別してください。

自動コミット・モード

Java アクティブに始動されたトランザクションがない場合は、アプリケーションが ObjectMap オブジェクトとの対話を行うと、アプリケーションの代わりに自動的に開始およびコミット操作が行われます。この自動的な開始およびコミット操作は役に立ちますが、ロールバックおよびロックが有効に機能する妨げとなります。トランザクションのサイズが小さすぎると、同期レプリカ生成スピードに影響します。エンティティ・マネージャー・アプリケーションを使用している場合は、自動コミット・モードは使用しないでください。その理由は、EntityManager.find メソッドで検索されたオブジェクトが、そのメソッドが戻されると同時に管理不能となり、使用不可となるためです。

.NET .NET アプリケーションでは、GridMapPessimisticAutoTx マップ・インターフェースが等価の自動開始およびコミット操作を提供します。制約は同じです。つまり、ロールバックとロックが正しく機能せず、同期レプリカ生成スピードが低下します。

Java

Java EE トランザクション統合

WebSphere eXtreme Scale クライアントは、リモート・データ・グリッドへのクライアント接続とローカル・トランザクション管理の両方をサポートする Java Connector Architecture (JCA) 1.5 準拠リソース・アダプターを含みます。サーブレットなどの Java Platform、Enterprise Edition (Java EE) アプリケーション、JavaServer Pages (JSP) フア

イル、および Enterprise JavaBeans (EJB) コンポーネントは、標準 javax.resource.cci.LocalTransaction インターフェイスまたは WebSphere eXtreme Scale クライアント セッション・インターフェイスを使用して、WebSphere eXtreme Scale クライアント トランザクションを区分的ことができます。

WebSphere Application Server で最終参加者サポートが有効になっている状態でアプリケーションを実行しているときには、WebSphere eXtreme Scale クライアント トランザクションを、他の 2 フェーズ・コミット・トランザクション・リソースと共にグローバル・トランザクションに登録することができます。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[.NET アプリケーションにおけるトランザクションのためのプログラミング](#)

ロック・ストラテジー

ロック・ストラテジーには、ペシミスティック、オプティミスティック、およびロックなしがあります。

ロックはトランザクションに 束縛されます。以下のロック設定を指定することができます。

Java | ロックなし

ロック設定を使用しないと、実行は最速になります。読み取り専用データを使用していれば、ロックは必要ない場合があります。

制約事項: ロックなしストラテジーを使用するように構成された BackingMap は、複数区画トランザクションに関与できません。

Java | .NET | ペシミスティック・ロック

エントリーに対するロックを取得し、コミット時までそのロックを保持します。このロック戦略は、スループットを低下させる代わりに、優れた一貫性を提供します。

Java | オプティミスティック・ロック

トランザクションがタッチするすべてのレコードの以前のイメージを取得して、トランザクションのコミット時に、そのイメージと現在のエントリーの値を比較します。エントリーの値が変更された場合、そのトランザクションはロールバックします。コミット時までロックは保持されません。このロック戦略は、ペシミスティック戦略よりも並行性において優れていますが、トランザクション・ロールバックのリスクがあり、エントリーのコピーを作成するためにメモリーを消費します。

重要: WebSphere® eXtreme Scale Client for .NET でクライアント・アプリケーションを使用している場合は、ペシミスティック・ロックのみがサポートされます。

ロック・マネージャー

ロック・ストラテジーに OPTIMISTIC または PESSIMISTIC が使用されている場合は、BackingMap に対してロック・マネージャーが作成されます。ロック・マネージャーは、ハッシュ・マップを使用して、1 つ以上のトランザクションによってロックされるエントリーを追跡します。ハッシュ・マップに多くのマップ・エントリーが存在する場合、ロック・バケットが多いほど、パフォーマンスが良好になる可能性が高くなります。バケット数が増えるにつれて、Java™ 同期の衝突のリスクは下がります。またロック・バケットを増やすことが、並行性の増大につながります。前の例では、特定の BackingMap インスタンスに使用するロック・バケットの数をアプリケーションでどのように設定できるかを示しています。

Java | .NET

ペシミスティック・ロック

ペシミスティック・ロック・ストラテジーは、キャッシュ・エントリーに対してロックを取得するため、データが頻繁に変更される場合に使用するようにしてください。キャッシュ・エントリーが読み取られる場合は、必ずロックが取得され、トランザクションが完了するまでロックが条件付きで保持されます。ロックによっては、セッションのトランザクション分離レベルを使用して、その期間を調整することができます。

ほかのロック・ストラテジーが可能でない場合は、マップの読み書きにペシミスティック・ロック・ストラテジーを使用します。ObjectGrid マップがペシミスティック・ロック・ストラテジーを使用するように構成されている場合、トランザクションが最初に BackingMap からのエントリーを取得すると、マップ・エントリーのペシミスティック・トランザクション・ロックが取得されます。ペシミスティック・ロックは、アプリケーションがトランザクションを完了するまでは保留されます。通常の場合、ペシミスティック・ロック・ストラテジーは、以下の状態で使用されます。

- BackingMap がで構成され、バージョン管理情報が使用可能でない場合。

制約事項: ローダー・プラグインで構成された BackingMap は、マルチ区画トランザクションのマップを読み取ることはできますが、書き込むことはできません。

- BackingMap が、並行処理制御について WebSphere eXtreme Scale クライアント からの支援を必要とするアプリケーションによって直接使用されている場合。
- バージョン管理情報は使用できるが、更新トランザクションがバックキング・エントリー上で頻繁に衝突し、その結果、オプティミスティック更新が失敗する場合。

ペシミスティック・ロック・ストラテジーは、パフォーマンスおよびスケーラビリティに最大の影響があります。したがって、このストラテジーは、他のロック・ストラテジーを使用できない場合に、マップの読み書きにのみ使用するようにしてください。例えば、こうした状態には、オプティミスティック更新の失敗が頻繁に発生する場合や、オプティミスティック障害からのリカバリーをアプリケーションが処理するには難しい場合が含まれます。

ペシミスティック・ロックを使用しているときには、ロック・メソッドを使用して、データ値を返すことなくデータすなわちキーをロックすることができます。メソッドのリストと、それらのメソッドが獲得するロックの種類については、[ロック・タイプ](#)を参照してください。

Java

オプティミスティック・ロック

デフォルトのロック・ストラテジーは、OPTIMISTIC です。データの変更が頻繁でない場合は、このオプティミスティック・ロックを使用します。データがキャッシュから読み取られ、トランザクションにコピーされる間、ロックは短期間だけ保持されます。トランザクション・キャッシュがメイン・キャッシュと同期されると、更新されたあらゆるキャッシュ・オブジェクトが元のバージョンに対してチェックされます。チェックが失敗すると、トランザクションはロールバックされ、OptimisticCollisionException 例外となります。

オプティミスティック・ロック・ストラテジーでは、並行して実行中である 2 つのトランザクションが同じマップ・エントリーの更新を試みることはないということが前提となっています。複数のトランザクションが同一のマップ・エントリーを同時に更新することはないと考えられるので、トランザクションのライフサイクルでロックは保持されません。オプティミスティック・ロック・ストラテジーは通常、以下の場合に使用されます。

- BackingMap が構成され、バージョン管理情報が使用可能である場合。

制約事項: ロダー・プラグインで構成された BackingMap は、マルチ区画トランザクションのマップを読み取ることにはできますが、書き込むことはできません。

- BackingMap のほとんどのトランザクションが読み取り操作である場合。BackingMap に対するエントリーの挿入、更新、または除去操作は、あまり行われません。
- BackingMap は、読み取りと比べてより頻繁に挿入、更新、または除去されるが、トランザクションは同じマップ・エントリー上でほとんど衝突しない場合。

ペシミスティック・ロック・ストラテジーと同様に、ObjectMap インターフェース上のメソッドは、WebSphere eXtreme Scale クライアント が、アクセス中のマップ・エントリーのロック・モードを自動的に取得する方法を決定します。ただし、ペシミスティック・ストラテジーとオプティミスティック・ストラテジーの間には、以下のような違いがあります。

- ペシミスティック・ロック・ストラテジーと同様に、メソッドの呼び出しの際、get メソッドおよび getAll メソッドによって S ロック・モードが取得されます。しかし、オプティミスティック・ロックを使用すると、S ロック・モードはトランザクションが完了するまで保留されません。代わりに、S ロック・モードはメソッドがアプリケーションに戻す前に保留解除されます。ロック・モードの取得の目的は、WebSphere eXtreme Scale クライアント が、その他のトランザクションからのコミット済みデータのみが現行トランザクションに可視となるように保証できるようにすることです。WebSphere eXtreme Scale クライアント がそのデータがコミット済みであることを確認した後で、S ロック・モードは保留解除されます。コミット時に、オプティミスティック・バージョン管理チェックが実行され、現行トランザクションがその S ロック・モードを保留解除した後で、マップ・エントリーを変更したトランザクションが他にないことが確認されます。更新、無効化、または削除される前にマップからエントリーがフェッチされない場合、WebSphere eXtreme Scale クライアント ランタイムによって、暗黙的にマップからエントリーがフェッチされます。この暗黙的な get 操作は、エントリーの変更が要求された時点における現行値を取得するために実行されます。
- ペシミスティック・ロック・ストラテジーとは異なり、getForUpdate メソッドと getAllForUpdate メソッドは、オプティミスティック・ロック・ストラテジーが使用された場合には、get メソッドと getAll メソッドと同様に処理されます。つまり、S ロック・モードはメソッドの開始時に取得され、S ロック・モードはアプリケーションに戻る前に保留解除されます。

他のすべての ObjectMap メソッドは、ペシミスティック・ロック・ストラテジーの場合と同様に扱われます。commit メソッドが呼び出されると、挿入、更新、除去、タッチ、または無効化の対象となるマップ・エントリーについては X ロック・モードが獲得されます。X ロック・モードは、トランザクションがコミット処理を完了するまで保持されません。

オプティミスティック・ロック・ストラテジーでは、並行して実行中のトランザクションが同じマップ・エントリーを更新することはないと想定します。この想定から、トランザクションの存続期間中、ロック・モードを保留する必要はありません。これは、複数のトランザクションがマップ・エントリーを並行して更新するとは考えられないためです。しかし、ロック・モードが保留されなかったため、現行トランザクションがその S ロック・モードを保留解除した後で、別の並行トランザクションがマップ・エントリーを更新する可能性があります。

この可能性に対処するため、WebSphere eXtreme Scale クライアント はコミット時に X ロックを取得し、オプティミスティック・バージョン管理チェックを行って、現行トランザクションが BackingMap からマップ・エントリーを読み取って以降、他にマップ・エントリーを変更したトランザクションがないことを確認します。別のトランザクションがマップ・エントリーを変更した場合、バージョン・チェックは失敗し、OptimisticCollisionException 例外が発生します。この例外により、現行トランザクションが強制的にロールバックされ、トランザクション全体がアプリケーションによって再試行されることとなります。オプティミスティック・ロック・ストラテジーは、マップがほとんど既読で、同じマップ・エントリーに対する更新が起こる可能性が低い場合に便利です。

Java

ロックなし

データがまったく更新されないか、静止期間のみに更新されるため、ロックが必要ない場合は、NONE ロック・ストラテジーを使用すれば、ロックを使用不可にすることができます。このストラテジーは、ロック・マネージャーを必要としないため、非常に高速です。NONE ロック・ストラテジーは、ルックアップ表または読み取り専用のマップの場合に理想的です。

BackingMap がロックなしストラテジーを使用するよう構成されている場合、マップ・エントリーのトランザクション・ロックは獲得されません。

制約事項: ロックなしストラテジーを使用するよう構成された BackingMap は、複数区画トランザクションに関与できません。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[ロック・ストラテジーの構成](#)

[Java](#) [Java アプリケーションにおけるロックの構成と実装](#)

[.NET](#) [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

[.NET](#) [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

例: [flush メソッドのロック順序](#)

ロック・タイプ

ペシミスティック・ロックおよびオプティミスティック・ロックを使用する場合、整合性を保つために、共有 (S) ロック、アップグレード可能 (U) ロック、および排他的 (X) ロックが使用されます。ペシミスティック・ロックを有効にしている場合、ロックとその動作について理解することが重要です。オプティミスティック・ロックではロックを保持しません。ロックにはライフサイクルがあります。いろいろなタイプのロックがさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

共有ロック、アップグレード可能ロック、および排他的ロック

アプリケーションがマップ・プログラミング・インターフェースのメソッドをどれか呼び出すと、WebSphere® eXtreme Scale クライアント は自動的に、アクセス対象のマップ・エンタリーに対するロックを獲得しようと試みます。

Java Java アプリケーションでは、アプリケーションが索引に対して検索メソッドを使用したときや照会を行ったときにも、ロックが獲得されます。

ペシミスティック・ロックを使用している場合、ロック・メソッドを使用して、データ値を返すことなくキーをロックすることができます。ロック・メソッドにより、データ・グリッド内のキーをロックするか、またはキーをロックして値がデータ・グリッド内に存在するかどうかを確認することができます。

LockMode は列挙型で以下に示す値を取ることができ、ここでロックしたいキーを指定できます。

- Java** SHARED、UPGRADABLE、および EXCLUSIVE
- .NET** Shared、Upgradable、Exclusive

WebSphere eXtreme Scale クライアント は、マップ・プログラミング・インターフェースでアプリケーションが呼び出すメソッドに基づいて以下のロック・モードを使用します。

S ロック

マップ・エンタリーのキーに対する共有ロック・モード。S ロックが保持されている期間は、使用されるトランザクション分離レベルによります。S ロック・モードでは、同一キーに対して S ロック・モードまたはアップグレード可能ロック (U ロック) モードを取得しようとする トランザクション間での並行処理が許されますが、同一キーに対して排他的ロック (X ロック) モードを取得しようとする他の トランザクションはブロックされます。

U ロック

マップ・エンタリーのキーに対するアップグレード可能ロック・モード。U ロックは、トランザクションが完了するまで保留になります。U ロック・モードでは、同一キーに対して S ロック・モードを取得する トランザクション間での並行処理が許されますが、同一キーに対して U ロック・モードまたは X ロック・モードを取得しようとする他の トランザクションはブロックされます。

X ロック

マップ・エンタリーのキーに対する排他的ロック・モード。X ロックは、トランザクションが完了するまで保留になります。X ロック・モードでは、1 つのトランザクションのみが所定のキー値のマップ・エンタリーを挿入、更新、または除去することになります。X ロックは、同一キーに対する S、U、または X ロック・モードを取得しようとする他のすべてのトランザクションをブロックします。

S ロック・モードは U ロック・モードより弱いロック・モードです。つまり、複数のトランザクションが同じマップ・エンタリーにアクセスするとき、S ロック・モードのほうがより多くのトランザクションを並行して実行できます。U ロック・モードは、S ロック・モードよりも少し強力です。それは、U ロック・モードまたは X ロック・モードのどちらかを要求している他のトランザクションをブロックするためです。S ロック・モードは、X ロック・モードを要求しているその他のトランザクションのみをブロックします。この小さな差が、一部のデッドロックの発生を防止するには重要です。X ロック・モードは、最強のロック・モードです。これは、同一のマップ・エンタリーに対して S、U、または X ロックのモードを取得しようとしているその他すべてのトランザクションをブロックするためにです。X ロック・モードは、1 つのトランザクションのみがマップ・エンタリーを挿入、更新、または除去できるようにするとともに、複数のトランザクションが同一のマップ・エンタリーの更新を試みている場合に更新が失われないようにします。

次の表を参照して、これらのロック・モード値と既存のメソッドの振る舞いとの関係を理解してください。

表 1. LockMode 値と既存の等価メソッド

ロック・モード (Java / .NET)	等価の Java メソッド	等価の .NET メソッド
SHARED / Shared	ObjectMap インターフェース上の get メソッドおよび getAll メソッド、索引メソッド、および照会	Get() メソッド、GetAndLock(Key, LockMode.Shared) メソッド、GetAndLockAll(KeyList, LockMode.Shared) メソッド、GetAll メソッド
UPGRADABLE /Upgradable	getForUpdate()、getAllForUpdate()	GetAndLock(Key, LockMode.Updgradable)、GetAndLockAll(KeyList, LockMode.Upgradable)

EXCLUSIVE / Exclusive	<p>getNextKey() メソッド、commit() メソッド、put メソッド、putAll メソッド、remove メソッド、removeAll メソッド、insert メソッド、update メソッド、および touch メソッド、グローバル invalidate メソッドおよびグローバル invalidateAll メソッド。(ローカルの invalidate メソッドおよび invalidateAll メソッドの場合、ロックは獲得されません。これは、どの BackingMap エントリーもローカル invalidate メソッド呼び出しによっては無効化されないからです。)</p> <p>Java  注: upsert メソッドおよび upsertAll メソッドが ObjectMap の put メソッドおよび putAll メソッドに取って代わります。データ・グリッド内のエントリーがキーと値をグリッドに配置しなければならないことを BackingMap とローダーに知らせるには、upsert メソッドを使用します。BackingMap とローダーは、insert または update のいずれかを行って、値をデータ・グリッドおよびローダーに配置します。アプリケーション内で upsert API を実行すると、ローダーは UPSERT LogElement タイプを取得します。これにより、ローダーは、insert や update を使用する代わりにデータベースの merge 呼び出しまたは upsert 呼び出しを行うことができます。</p>	<p>Commit()、Add、AddAll、Put、PutAll、Remove、RemoveAll、Replace、ReplaceAll、Touch、TouchAll、Invalidate、InvalidateAll</p> <p>注: Put メソッドおよび PutAll メソッドは Java の upsert および upsertAll メソッドと等価です。</p>
-----------------------	--	--

次表は、ロック・モードの互換性マトリックスです。前述のロック・モードをまとめたもので、互いに互換性のあるロック・モードはいずれかを調べる場合に使用してください。このマトリックスを読み取る場合、マトリックスの行は既に認可されているロック・モードを表します。列は、別のトランザクションによって要求されたロック・モードを表します。列に「あり」と表示されている場合は、別のトランザクションによって要求されたロック・モードは認可されています。これは、既に認可されているロック・モードと互換性があるためです。「なし」は、ロック・モードの互換性がないことを表します。その他のトランザクションは、最初のトランザクションが保持しているロックを解放するのを待たなければなりません。

表 2. ロック・モードの互換性マトリックス

ロック	ロック・タイプ S (共有)	ロック・タイプ U (アップグレード可能)	ロック・タイプ X (排他的)	強さ
S (共有)	はい	はい	いいえ	最弱
U (アップグレード可能)	はい	いいえ	いいえ	通常
X (排他的)	いいえ	いいえ	いいえ	最強

親トピック: [トランザクション処理の概要](#)

関連タスク:

[ロック・ストラテジーの構成](#)

Java [Java アプリケーションにおけるロックの構成と実装](#)

.NET [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

.NET [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

[例: flush メソッドのロック順序](#)

デッドロック

2つのトランザクションが同じキャッシュ・エントリを更新しようとするするとデッドロックが起こることがあります。

古典的なデッドロックの例

ロック・モード要求の以下のシーケンスについて検討します。

1. Xロックは、トランザクション 1 の key1 に対して認可されています。
2. Xロックは、トランザクション 2 の key2 に対して認可されています。
3. トランザクション 1 によって要求された、key2 に対する X ロック (トランザクション 1 はブロックされ、トランザクション 2 が所有するロックを待機しています。)
4. トランザクション 2 によって要求された、key1 に対する X ロック (トランザクション 2 はブロックされ、トランザクション 1 が所有するロックを待機しています。)

上記のシーケンスは、2つのトランザクションからなる古典的なデッドロックの例です。2つのトランザクションが複数のロックを取得しようとし、各トランザクションは異なる順序でロック取得します。このデッドロックを防止するには、各トランザクションが複数ロックを同じ順序で獲得しなければなりません。

Java

オプティミスティック・ロックによるデッドロック防止

オプティミスティック・ロック・ストラテジーが使用され、ObjectMap インターフェースの flush メソッドがアプリケーションによって絶対に使用されない場合は、ロック・モードがトランザクションによって要求されるのはコミット・サイクル中のみです。コミット・サイクル中、WebSphere® eXtreme Scale クライアントは、ロックする必要があるマップ・エントリのキーを決定し、キー・シーケンスのロック・モードを要求します (決定論的振る舞い)。WebSphere eXtreme Scale クライアントは、この方法を使用して多くの古典的デッドロックを防止します。

しかし、eXtreme Scale がすべてのデッドロック・シナリオを防止するわけでも、防止できるわけでもありません。アプリケーションが考慮する必要があるシナリオがいくつか存在します。以下は、アプリケーションが注意し、予防アクションを取らなければならないシナリオです。

1つのシナリオは、ロック待ちタイムアウトが発生するのを待たなくとも WebSphere eXtreme Scale クライアントがデッドロックを検出できる場合です。このシナリオが起こると、com.ibm.websphere.objectgrid.LockDeadlockExceptionException が発生します。次のコード例をご覧ください。

```
Session sess = ...;
ObjectMap person = sess.getMap("PERSON");
sess.begin();
Person p = (IPerson)person.get("Lynn");
// Lynn had a birthday; so make her 1 year older.
p.setAge( p.getAge() + 1 );
person.put( "Lynn", p );
sess.commit();
```

同じシナリオで、次のコード例のように upsert メソッドを使用することができます。

```
Session sess = ...;
ObjectMap person = sess.getMap("PERSON");
sess.begin();
Person p = (IPerson)person.get("Lynn");
// Lynn had a birthday; so make her 1 year older.
p.setAge( p.getAge() + 1 );
person.upsert( "Lynn", p );
sess.commit();
```

この場合、2つのトランザクションが Lynn person オブジェクトの年齢を更新しようとしています。この状態では、person.get("Lynn") メソッド呼び出しの結果として両方のトランザクションが PERSON マップの Lynn エントリに対して S ロック・モードを保持します。person.put ("Lynn", p) メソッド呼び出しの結果として、両方のトランザクションは S ロック・モードを X ロック・モードに格上げしようとしています。両方のトランザクションがブロックされ、そのどちらのトランザクションも、もう一方のトランザクションがその所有する S ロック・モードを解放するのを待っています。その結果、2つのトランザクションの間に循環待ち状態が存在することになってデッドロックが発生します。循環待ち状態は、複数のトランザクションが同一のマップ・エントリに対して持っているロックを弱いモードからより強いモードに昇格させようとしたときに発生します。このシナリオでは、LockTimeoutException 例外ではなく、LockDeadlockException 例外になります。

Java アプリケーションでは、ペシミスティック・ロック・ストラテジーではなくオプティミスティック・ロック・ストラテジーを使用することによって、上記の例の LockDeadlockException 例外を防止することができます。オプティミスティック・ロック・ストラテジーの使用は、マップが主として読み取りで、マップの更新がまれにしか行われない場合、推奨される解決策です。

ペシミスティック・ロックによるデッドロック防止

.NET 重要: .NET アプリケーションはペシミスティック・ロックのみをサポートします。次のセクションでは、Java メソッド名について説明します。ただし、.NET メソッド名にも当てはまります。これらのメソッドは、Get、GetAndLock、GetAndLockAll、Put、Add、Replace、および Remove です。

ペシミスティック・ロック・ストラテジーを使用する場合にデッドロックを防止するには、次のようにします。

- READ_COMMITTED トランザクション分離レベルを使用します。READ_COMMITTED トランザクション分離レベルは、get メソッドによって獲得された S ロックがトランザクション完了まで保持されることがないようにします。キーがトランザクション・キャッシュで無効化されない場合、反復可能読み取りは引き続き保証されます。
- get メソッドではなく代替の get メソッドを使用します。
 - **Java** getForUpdate メソッドを使用します。
 - **.NET** GetAndLock または GetAndLockAll メソッドを使用します。

最初のトランザクションは getForUpdate メソッドを呼び出して、S ロックではなく U ロック・モードを獲得します。このロック・モードにより、2 番目のトランザクションは、getForUpdate メソッドを呼び出したときにブロックされます。U ロック・モードが付与されるトランザクションは 1 つです。2 番目のトランザクションはブロックされるので、マップ・エントリーに対するロック・モードを何も所有しません。最初のトランザクションは、最初のトランザクションからの put メソッド呼び出しの結果として、U ロック・モードから X ロック・モードへの格上げをしようとしたときに、ブロックしません。この働きは、U ロック・モードがアップグレード可能ロック・モードと呼ばれる理由を説明しています。最初のトランザクションが完了すると、2 番目のトランザクションのブロックが解除され、U ロック・モードが付与されます。ペシミスティック・ロック・ストラテジーが使用される場合は、アプリケーションは、get メソッドの代替として getForUpdate メソッドを使用することによりロック昇格デッドロック・シナリオを回避することができます。

重要: この解決策は、読み取り専用トランザクションがマップ・エントリーを読み取るのを妨げません。読み取り専用トランザクションは get メソッドを呼び出します。読み取り専用トランザクションが put、insert、update、または remove メソッドを呼び出すことはありません。並行性は通常の get メソッドが使用されたときと同程度です。唯一、並行性が低減するのは、複数のトランザクションによって同一のマップ・エントリーに対して getForUpdate メソッドが呼び出されることです。

あるトランザクションが複数のマップ・エントリーに対して getForUpdate メソッドを呼び出す場合、各トランザクションによって確実に U ロックが同じ順序で取得されるように注意しなければなりません。例えば、最初のトランザクションがこのメソッドを 2 回 (キー 1、次いでキー 2 に対して) 呼び出し、別の並行トランザクションもこのメソッドを同じ 2 つのキーに対して (ただし逆順で) 呼び出すとします。このシーケンスにより、古典的なデッドロックが発生します。複数ロックが異なるトランザクションによって異なる順序で獲得されるためです。アプリケーションでは引き続き、複数のマップ・エントリーにアクセスするどのトランザクションもキー・シーケンスに従い、デッドロックが発生しないようにする必要があります。U ロックはコミット時ではなく、getForUpdate メソッドが呼び出される時に獲得されるので、WebSphere eXtreme Scale クライアントは、コミット・サイクル中に行われるようにロック要求を順序付けることはできません。アプリケーションは、このケースではロックの順序付けを制御する必要があります。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[ロック・ストラテジーの構成](#)

Java [Java アプリケーションにおけるロックの構成と実装](#)

.NET [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

.NET [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

例: [flush メソッドのロック順序](#)

データ・アクセスおよびトランザクション

WebSphere® eXtreme Scale クライアント はトランザクションを使用します。アプリケーションがデータ・グリッドへの接続を獲得したら、ユーザーはデータ・グリッド内のデータにアクセスし、データを操作することができます。

Java

Java アプリケーションにおけるトランザクション

Java アプリケーションを使用して、分散インスタンスへのクライアント接続を確立することができます。

アプリケーションがセッションとの対話を行う場合、そのセッションはトランザクションのコンテキストの中にある必要があります。トランザクションはセッション・オブジェクトの `Session.begin`、`Session.commit`、および `Session.rollback` メソッドを使用して、開始されたり、コミットまたはロールバックされます。アプリケーションは、自動コミット・モードで作業を行うこともできます。このモードの場合、アプリケーションがマップとの対話を行うたび、セッションが自動的に開始し、トランザクションをコミットします。ただし、自動コミット・モードは低速です。

Java アプリケーション内のスレッドには、独自のセッションが必要です。アプリケーションがスレッド上で `ObjectGrid` を使用するようにしたい場合は、`getSession` メソッドの 1 つを呼び出してセッションを取得します。アプリケーションがそのセッションを終了した後、`Session.close()` メソッドを呼び出します。このメソッドはセッションを閉じて、そのセッションをプールに返し、そのセッションのリソースを解放します。セッションを閉じるのはオプションですが、そうすると、`getSession()` メソッドに対する後続の呼び出しのパフォーマンスが向上します。アプリケーションが、Spring のような依存性注入フレームワークを使用する場合、必要なときにセッションをアプリケーション Bean に注入することができます。

セッションを取得した後、アプリケーションは `ObjectGrid` 内のマップに保管されたデータにアクセスできます。マップ・ベースの API は `Session.getMap` メソッドを使用して取得されます。

.NET

.NET アプリケーションにおけるトランザクション

.NET アプリケーションでは、スレッドごとに別々の `IGridMapPessimisticTx` オブジェクトまたは `IGridMapPessimisticAutoTx` オブジェクトを持つ必要があります。 `IGridMapPessimisticTx` オブジェクトでは、`Transaction` プロパティを使用してトランザクションの開始、コミット、またはロールバックが明示的に行われます。 `IGridMapPessimisticAutoTx` オブジェクトでは、トランザクションの開始、コミット、およびロールバック操作が自動的に行われます。これらのオブジェクトの 1 つを取得すると、アプリケーションはデータ・グリッド内の保管データにアクセスできるようになります。

トランザクション使用のロジック

トランザクションは遅く感じられるかもしれません。以下の理由からトランザクションを使用する必要があります。

1. 例外が発生した場合や、状態変更を元に戻すことをビジネス・ロジックが必要とする場合に、変更のロールバックが可能であること。
2. 1 つのトランザクションの存続時間中にデータに対するロックの保持と解除を行うことで、一連の変更がアトミックに行われる、つまり、データに対してすべての変更を行うか、何も変更しないかにできること。
3. 複製のアトミックな単位を生成できること。
4. [Java](#) 複数の区画を更新できること。

どの程度のトランザクション・サポートが必要であるかをカスタマイズすることができます。アプリケーションはロールバック・サポートおよびロックをオフにできますが、これはアプリケーションに犠牲を強いることになります。アプリケーションはこれらのフィーチャーの欠如に対処する必要があります。アプリケーションがトランザクション・サポートをどのように管理できるかの例を以下に示します。

- [Java](#) 動的マップ・ロック・ストラテジーを `NONE` に構成すると、アプリケーションはロックをオフにできます。詳しくは、[Java API を使用した動的マップの作成](#)を参照してください。このストラテジーは高速ですが、並行トランザクションが互いに保護されずに、同じデータを変更できるようになります。 `NONE` を使用する場合は、そのアプリケーションが、すべてのロックおよびデータの整合性に対する責任を持つことになります。 `PESSIMISTIC` ロック・ストラテジーしかサポートしていない `WebSphere eXtreme Scale Client for .NET` アプリケーションの場合、このオプションは無効です。

[Java](#) `NONE CopyMode` 値を使用して取得されたオブジェクトがアプリケーションによって変更されると、そのオブジェクトのコミット済みのコピーが直接変更されます。このモードでは、トランザクションのロールバックは意味がありません。 `ObjectGrids` での唯一のコピーが変更されます。 `NONE CopyMode` を使用すると処理は速くなりますが、その影響に注意する必要があります。 `NONE CopyMode` を使用するアプリケーションは、トランザクションを決してロールバックしてはなりません。もしアプリケーションがトランザクションをロールバックした場合、索引に変更を反映する更新は行われず、かつ、レプリカ生成がオンにされていても変更は複製されません。

デフォルト値を使用するほうが簡単で、誤りの可能性も低くなります。データ信頼性を犠牲にしてもパフォーマンスを上げたい場合は、意図しない問題を回避するために、アプリケーションは実行内容をよく認識する必要があります。

トランザクションと区画

Java アプリケーションにおけるトランザクションは単一の区画でも複数の区画でも更新できますが、単一区画の更新がデフォルトの動作です。.NET アプリケーションは単一区画の更新しか行えません。

WebSphere eXtreme Scale クライアントに対する複数区画トランザクション・サポートを使用可能にするには、TxCommitProtocol セッション API を使用します。次の 2 つのオプションを使用することができます。

- TxCommitProtocol.ONEPHASE (デフォルト): クライアントからのトランザクションは、複数の区画から読み取ることができますが、単一区画の更新しか行えません。複数区画の更新を試みると失敗します。
- TxCommitProtocol.TWOPHASE: クライアントからのトランザクションは複数の区画を読み取って更新することができます。このトランザクションは、2 フェーズ・コミット・プロトコルを使用して、これらの区画に書き込まれたデータが自動的にコミットまたはロールバックされるようにします。このトランザクションが単一の区画にのみ書き込む場合は、1 フェーズ・コミットメント・プロトコルが使用されます。

親トピック: [トランザクション処理の概要](#)

関連タスク:

[Java](#) [スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクションに書き込むアプリケーションの開発](#)

[Java](#) [Java アプリケーションのためのトランザクション内のデータとの対話](#)

[.NET](#) [.NET アプリケーションのためのトランザクション内のデータとの対話](#)

トランザクション分離

3つのトランザクション分離レベル(反復可能読み取り、読み取りコミット済み、および読み取りアンコミット)のうち1つを使用して、各キャッシュ・マップ内の整合性を保つロック・セマンティクスを調整することができます。

トランザクション分離の概説

トランザクション分離は、1つの操作で行われた変更がどのように他の並行操作に可視になるのかを定義します。

WebSphere® eXtreme Scale クライアントが各キャッシュ・マップ内の整合性を保つために使用するロック・セマンティクスを調整するためのトランザクション分離レベル(反復可能読み取り、読み取りコミット済み、および読み取りアンコミット)を定義することができます。

次のいずれかの方法でトランザクション分離レベルを設定することができます。

- **Java** Session インターフェイスで `setTransactionIsolation` メソッドを使用する。トランザクション分離は、現在進行中のトランザクションがなければ、セッションの存続期間中いつでも変更できます。
- **.NET** IGridTransaction インターフェイスで `TransactionIsolationLevel` プロパティを使用する。

この製品では、共有 (S) ロックが要求および保持される方法を調整することによって、さまざまなトランザクション分離セマンティクスが施行されます。トランザクション分離は、オプティミスティック・ロックを使用するように構成されているマップや、ロックを使用しないように構成されているマップに対しては何の影響もありません。また、アップグレード可能 (U) ロックが獲得されるときにも影響を及ぼしません。

ペシミスティック・ロックでの 反復可能読み取り

反復可能読み取りは、デフォルトのトランザクション分離レベルです。この分離レベルは、ダーティー読み取りおよび反復不能読み取りを防止しますが、幻償読み取りは防止しません。ダーティー読み取りとは、あるトランザクションによって変更されたが、コミットされていないという状態のデータに対して発生する読み取り操作のことです。反復不能読み取りは、読み取り操作実行時に読み取りロックが取得されていない場合に発生する可能性があります。幻償読み取りは、2つの同一の読み取り操作が実行されたが、操作と操作との間にデータに対する更新があったために2つの異なる結果セットが戻される場合に、発生する可能性があります。Java アプリケーションでは、照会または索引を使用しているときに幻償読み取りが発生する可能性があります。これは、ロックがデータの範囲に対して獲得されず、索引または照会基準を満たすキャッシュ・エントリに対してのみ獲得されるからです。この製品は、すべての S ロックを、ロックを所有するトランザクションが完了するまで保持し続けることによって、反復可能読み取りを実現します。X ロックは、すべての S ロックが解放されるまで認可されないため、S ロックを保持するすべてのトランザクションは、再読み取り時に同じ値を参照することが保証されます。

ペシミスティック・ロックでの読み取りコミット済み

読み取りコミット済みのトランザクション分離レベルを WebSphere eXtreme Scale クライアントで使用できます。この分離レベルは、ダーティー読み取りを防止しますが、反復不能読み取りまたは幻償読み取りを防止しないため、WebSphere eXtreme Scale クライアントは S ロックを引き続き使用してキャッシュ・マップからデータを読み取りますが、すぐにロックを解放します。

ペシミスティック・ロックでの読み取りアンコミット

読み取りアンコミットのトランザクション分離レベルを WebSphere eXtreme Scale クライアントで使用できます。この分離レベルは、ダーティー読み取り、反復不能読み取り、および幻償読み取りを許容します。

親トピック: [トランザクション処理の概要](#)

関連資料:

Java [Java のトランザクション分離の例](#)

2 フェーズ・コミットとエラー・リカバリー

2.5+ 2 フェーズ・コミット・プロトコルは、分散トランザクションに関与するすべての区画を、そのトランザクションをコミットするかロールバックするかに基づいて調整します。

分散データ・グリッドでは、区画は複数の Java™ 仮想マシン (JVM) に分散されます。これらの JVM は複数のシステムに存在することができます。複数の区画に書き込むトランザクションは、複数のシステムに影響を及ぼすトランザクション決定を伴うことがあります。このようなトランザクションが 2 フェーズ・コミット・プロトコルを使用してコミットされると、このコミット・プロセスは、トランザクション全体が保持されるか、またはトランザクションがまったく保持されないようにします。2 フェーズ・コミット・プロセスは、区画、システム、または通信に障害が起こった場合でも、この結果を保証します。第 2 フェーズで障害が起こると、そのエラーが手動で介入できる一定基準を満たしているとき以外は、WebSphere® eXtreme Scale クライアントが障害を自動的に解決しようと試みます。

複数の区画への書き込みが有効になっているトランザクションは 2 フェーズ・コミット・プロトコルを使用します。2 フェーズ・コミット・プロトコルは、すべての区画およびシステムでコミット・プロセスが整合性を保てるようにします。WebSphere eXtreme Scale は、2 フェーズ・コミット・プロセスを制御するコーディネーターとなります。このようなトランザクションに関係する区画を参加者あるいはリソース・マネージャー (RM) と呼びます。コーディネーターは、コミット・プロトコルの第 2 フェーズで、区画の 1 つがトランザクション・マネージャー (TM) として動作するように委任します。TM は、各トランザクションの決定の追跡と、障害が発生した場合のトランザクションのリカバリーを行います。

第 1 フェーズ:

アプリケーションがトランザクションをコミットすると、WebSphere eXtreme Scale クライアントはまず、RM として特定された各区画にコミット準備要求を送ることによって第 1 フェーズを開始します。各区画はトランザクション変更をバックアップ・マップに適用し、データ安全性を確保するためにすべてのロックを保持します。RM は WebSphere eXtreme Scale クライアントに通知します。RM として特定されたすべての区画が成功で応答すると、WebSphere eXtreme Scale クライアントはコミット・プロトコルの第 2 フェーズを開始します。

第 2 フェーズ:

第 1 フェーズで少なくとも 1 つの区画が障害を起こすと、コーディネーターは第 2 フェーズですべての区画をロールバックします。すべての RM 区画が成功で応答すると、WebSphere eXtreme Scale クライアントは区画の 1 つを TM 区画となるように任命します。WebSphere eXtreme Scale は、コーディネーターとして、トランザクションに関係するすべての区画にコミット要求またはロールバック要求を送ることによってコミット・プロトコルの第 2 フェーズを開始します。そうすると、RM として特定された各区画が変更をバックアップ・マップに適用するかまたはロールバックし、すべてのロックを解除します。その後、RM は WebSphere eXtreme Scale クライアントに通知します。第 2 フェーズで少なくとも 1 つの区画が障害を起こした場合は、委任された TM 区画が自動的にトランザクションをリカバリーします。自動リカバリーにより、トランザクションに関係するすべての区画の整合性が確保されます。

未確定フェーズ:

未確定フェーズとは、RM 区画が第 1 フェーズを正常に処理し終えてから第 2 フェーズを開始するまでの待機期間を指します。未確定期間中、RM 区画はトランザクションをコミットするかロールバックするかを知りません。RM 区画はロックを保持し続けます。ロックが保持されていると、他のトランザクションでロック競合が増加する場合があります。

2 フェーズ・コミット中のエラー・リカバリー

第 1 フェーズで障害が発生すると、WebSphere eXtreme Scale クライアントはトランザクションをロールバックします。区画の 1 つがトランザクションのコミットに失敗すると、TM は、トランザクションのコミットを周期的に試みることによってトランザクションがコミットされるようにします。このシナリオで生成されるログ・メッセージの例を次に示します。

```
00000099 TransactionLog I CW0BJ8705I: Automatic resolution of transaction WXS-40000139-DF01-216D-E002-1CB456931719 at RM:TestGrid:TestSet2:20 is still waiting for a decision. Another attempt to resolve the transaction will occur in 30 seconds.
```

WebSphere eXtreme Scale クライアントがトランザクションを解決できるようにしてください。手動介入を試みるのは、トランザクションが 1 分以内にリカバリーされない場合や、未確定トランザクションとなりアプリケーションで頻繁にロック競合が長発生する場合のみにしてください。トランザクションの手動リカバリーについて詳しくは、[マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)を参照してください。

親トピック: [トランザクション処理の概要](#)

親トピック: **2.5+** [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

関連概念:

[ロック・ストラテジー](#)

関連タスク:

[マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について 実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒103-8510

東京都中央区日本橋箱崎町19番21号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス渉外

以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムと その他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

IBM® Corporation

Mail Station P300

522 South Road

Poughkeepsie, NY 12601-5400

USA

Attention: Information Requests

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性がありますが、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

表示されている IBM の価格は IBM が小売り価格として提示しているもので、現行価格であり、通知なしに変更されるものです。卸価格は、異なる場合があります。

本書はプランニング目的としてのみ記述されています。記述内容は製品が使用可能になる前に変更になる場合があります。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。これらのサンプル・プログラムは特定物として現存するままの状態を提供されるものであり、いかなる保証も提供されません。IBM は、お客様の当該サンプル・プログラムの使用から生ずるいかなる損害に対しても一切の責任を負いません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。

© (お客様の会社名) (西暦年). このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。

© Copyright IBM Corp. _年を入れる_. All rights reserved.

プログラミング・インターフェース情報

本書の情報は、WebSphere® eXtreme Scale のプログラミング・インターフェースとして使用されることを意図して記述されたものではありません。本書には、プログラムを作成するユーザーが WebSphere eXtreme Scale のサービスを使用するためのプログラミング・インターフェースが記述されています。この情報は、章またはセクションの始まりの文によって、あるいは下記の表示によって、その出現箇所を識別できます: プログラミング・インターフェース情報。

商標

IBM、IBM ロゴ、および ibm.com は、世界の多くの国で登録された International Business Machines Corp. の商標です。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。現時点での IBM の商標リストについては、www.ibm.com/legal/copytrade.shtml の [Copyright and trademark information](#) をご覧ください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

プライバシー・ポリシーに関する考慮事項

サービス・ソリューションとしてのソフトウェアも含めた IBM® ソフトウェア製品（「ソフトウェア・オファリング」）では、製品の使用に関する情報の収集、エンド・ユーザーの使用感の向上、エンド・ユーザーとの対話またはその他の目的のために、Cookie はじめさまざまなテクノロジーを使用することがあります。多くの場合、ソフトウェア・オファリングにより個人情報が収集されることはありません。IBM の「ソフトウェア・オファリング」の一部には、個人情報を収集できる機能を持つものがあります。ご使用の「ソフトウェア・オファリング」が、これらの Cookie およびそれに類するテクノロジーを通じてお客様による個人情報の収集を可能にする場合、以下の具体的事項を確認ください。

この「ソフトウェア・オファリング」は、Cookie もしくはその他のテクノロジーを使用して個人情報を収集することはありません。

この「ソフトウェア・オファリング」が Cookie およびさまざまなテクノロジーを使用してエンド・ユーザーから個人情報を収集する機能を提供する場合、お客様は、個人情報を収集するにあたって適用される法律、ガイドライン等を遵守する必要があります。これには、エンド・ユーザーへの通知や同意取得の要求も含まれますがそれらには限りません。

このような目的での Cookie を含むさまざまなテクノロジーの使用について詳しくは、『IBM プライバシー・ポリシー』（<http://www.ibm.com/privacy>）および『IBM オンラインでのプライバシー・ステートメント』（<http://www.ibm.com/privacy/details/us/en>）の『クッキー、Web ビーコン、その他のテクノロジー』と『ソフトウェア製品と Software-as-a Service』のセクションを参照してください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス の概要](#)

シナリオ

全体像を描くために、シナリオには実在の情報が含まれています。シナリオは、新しい概念を理解したり、一般的なタスクを実行するのに役立ちます。

[シナリオ: エンタープライズ・データ・グリッドの構成](#)

Java™ アプリケーションと .NET アプリケーションの両方が同じデータ・グリッドに接続できるようにしたいときは、エンタープライズ・データ・グリッドを構成してください。

2.5+ [WebSphere DataPower XC10 アプライアンスの保護](#)

アプライアンス内のデータ・グリッドは、保護する必要がある機密情報を保管します。

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリ一貫複製セッションまたはデータベース・セッションのマイグレーション](#)

以前に設定したメモリー間複製セッションまたはデータベース・セッションをマイグレーションして、WebSphere® eXtreme Scale セッション管理を使用できます。

シナリオ: エンタープライズ・データ・グリッドの構成

Java™ アプリケーションと .NET アプリケーションの両方が同じデータ・グリッドに接続するようにしたいときは、エンタープライズ・データ・グリッドを構成してください。

始める前に

- 製品をインストールします。クライアントの場合は、Java クライアントと .NET クライアントの両方を使用することができます。詳しくは、[WebSphere DataPower XC10 アプライアンスのインストール](#)を参照してください。
- 前のリリースからアップグレードする場合は、すべてのコンテナ・サーバーおよびカタログ・サーバーが同じリリース・レベルでなければなりません。詳しくは、[WebSphere DataPower XC10 アプライアンスの更新](#)を参照してください。

1. [エンタープライズ・データ・グリッドの概要](#)

エンタープライズ・データ・グリッドは eXtremeIO トラnsポート・メカニズムと新しいシリアライゼーション形式を使用します。この新しいトラnsポートおよびシリアライゼーション形式により、Java クライアントと .NET クライアントの両方を同じデータ・グリッドに接続することができます。

2. [IBM eXtremeIO \(XIO\) の構成](#)

IBM® eXtremeIO (XIO) は、オブジェクト・リクエスト・ブローカー (ORB) を置き換えるトラnsポート・メカニズムです。

3. [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

IBM eXtremeIO を構成した後、エンタープライズ・データ・グリッドにアクセスするアプリケーションを作成することができます。

親トピック: [2.5+ シナリオ](#)

IBM eXtremeIO (XIO) の構成

2.5+ IBM® eXtremeIO (XIO) は、オブジェクト・リクエスト・ブローカー (ORB) を置き換えるトランスポート・メカニズムです。

始める前に

- **2.5** XIO を構成するためには、WebSphere® eXtreme Scale クライアント バージョン 8.6.0.2 以降が必要です。

カタログ・サーバーで XIO を使用可能にすることで、カタログ・サービス・ドメイン内のすべてのコンテナ・サーバーに対して XIO を構成できます。コンテナ・サーバーはカタログ・サーバーのトランスポート・タイプをディスカバーし、そのトランスポート・タイプを使用します。

2.5

このタスクについて

バージョン 2.5 以降を使用して新規構成を作成している場合は、XIO が、デフォルトのトランスポート・メカニズムです。前のファームウェア・バージョンからアップグレードする場合は、トランスポート設定は ORB を使用するように設定されます。

手順

集合で XIO トランスポート・メカニズムを使用可能に設定します。ユーザー・インターフェースで、「集合」 > 「設定」 > 「通信サービス」をクリックします。XIO トランスポート・メカニズムを選択します。トランスポートの設定は、集合全体での設定です。したがって、トランスポート設定を更新すると、集合内のすべてのアプライアンスを再始動する必要があります。

タスクの結果

構成したサーバーは、XIO トランスポートを使用します。構成が正しいことを確認するには、[カタログ・サービス・ドメインのトランスポート・タイプの表示](#)を参照してください。

集合が、XIO トランスポート・メカニズムを使用します。XIO を使用可能に設定すると、以下も使用可能になります。

eXtreme Data Format (XDF)

XDF は、キーおよび値を、言語に依存しない形式でシリアライズし、データ・グリッドに保管します。XDF を使用すると、Java アプリケーションと .NET アプリケーションの両方が同じデータ・グリッド・オブジェクトにアクセスできます。

IBM eXtremeMemory

eXtremeMemory を使用すると、ガーベッジ・コレクションの一時停止を回避することができ、パフォーマンスがより安定し、応答時間が予測可能になります。

2.5+ [カタログ・サービス・ドメインのトランスポート・タイプの表示](#)

カタログ・サービス・ドメインで現在使用されているトランスポート・タイプを表示できます。

親トピック: [シナリオ: エンタープライズ・データ・グリッドの構成](#)

前のトピック: [エンタープライズ・データ・グリッドの概要](#)

次のトピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

親トピック: [集合およびゾーンの構成](#)

前のトピック: [集合間のマルチマスター複製の構成](#)

エンタープライズ・データ・グリッド・アプリケーションの開発

IBM eXtremeIO を構成した後、エンタープライズ・データ・グリッドにアクセスするアプリケーションを作成することができます。

始める前に

- データ・グリッドにアクセスする既存の Java または .NET アプリケーションがなければなりません。アプリケーション作成入門について詳しくは、[入門チュートリアル・モジュール 2: クライアント・アプリケーションの作成](#)を参照してください。

[クラスの進化](#)

eXtreme data format (XDF) ではクラスの進化が可能です。クラスの進化を活用することで、旧バージョンのクラスを使用している古いアプリケーションに影響を及ぼすことなく、データ・グリッドで使用されるクラス定義を進化させることができます。これらの古いクラスは、新しいアプリケーションと同じマップ内のデータにアクセスします。

[Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

ClassAlias および FieldAlias 注釈を使用して、Java と .NET クラス間でのデータ・グリッド・データの共有を使用可能にします。

[PartitionKey 注釈を使用したキーから区画へのマップ](#)

PartitionKey 別名を使用して、データが保存される区画を判別するためにハッシュ・コード計算を実行するフィールドまたは属性を識別します。PartitionKey 注釈は、キー属性でのみ有効です。

[Java および C# の等価データ型](#)

エンタープライズ・データ・グリッド・アプリケーションを開発するときは、Java アプリケーションと C# アプリケーションの間でデータ型に互換性がなければなりません。

親トピック: [シナリオ: エンタープライズ・データ・グリッドの構成](#)

前のトピック: [IBM eXtremeIO \(XIO\) の構成](#)

クラスの進化

eXtreme data format (XDF) ではクラスの進化が可能です。クラスの進化を活用することで、旧バージョンのクラスを使用している古いアプリケーションに影響を及ぼすことなく、データ・グリッドで使用されるクラス定義を進化させることができます。これらの古いクラスは、新しいアプリケーションと同じマップ内のデータにアクセスします。

概要

クラスの進化とは、2つのタイプが共に機能できるだけの十分な互換性を備えているかどうかの決定にかかわる、クラスとフィールドの識別のさらなる拡張です。一方のクラスが他方のクラスよりフィールド数が少ないとき、これらのクラスは共に機能することができます。以下のユーザー・シナリオは XDF の実装を意図したものです。

同じオブジェクト・クラスに複数のバージョンがある場合

このシナリオには、販売アプリケーションで顧客の追跡に使用されるマップがあります。このマップには2つの異なるインターフェースがあります。1つ目は Web での購入用のインターフェースです。2つ目は電話での購入用のインターフェースです。この販売アプリケーションのバージョン2では、Web での買い物客に対して、それぞれの購買習慣に基づいて割引を行うことにします。この割引は「顧客」オブジェクトと一緒に保管されます。電話で受注をする販売員はまだこのアプリケーションのバージョン1を使用しています。このバージョン1では、Web バージョンにある新しい割引フィールドが認識されていません。アプリケーションのバージョン2にある「顧客」オブジェクトが、バージョン1アプリケーションで作成された「顧客」オブジェクトと連動するようにします(逆の場合も同じです)。

別のオブジェクト・クラスに複数のバージョンがある場合

このシナリオには、Java™ で作成され、「顧客」オブジェクトのマップを保持する販売アプリケーションがあります。また、C# で作成されているもう1つのアプリケーションがあり、このアプリケーションは倉庫内の在庫を管理し、商品を顧客に出荷するために使用されます。現在、これらのクラスは、クラス、フィールド、およびタイプの名前に基づき互換性があります。Java の販売アプリケーションでは、販売員を顧客アカウントと関連付けるオプションを「顧客」レコードに追加することにします。ただし、このフィールドは倉庫アプリケーションでは不要であるため、このフィールドを保管するように倉庫アプリケーションを更新することはしません。

同じクラスに複数の非互換バージョンがある場合

このシナリオでは、販売アプリケーションと在庫アプリケーションの両方に「顧客」オブジェクトが含まれています。在庫アプリケーションはstringのIDフィールドを使用し、販売アプリケーションは整数のIDフィールドを使用します。これらのタイプには互換性がありません。そのため、これらのオブジェクトはおそらく同じマップに保管されません。これらのオブジェクトは XDF シリアライゼーションによって処理され、かつ2つの異なるタイプとみなされるようにする必要があります。このシナリオは実際にはクラスの進化ではありませんが、アプリケーション設計全体の一部として考慮しなければならない事項です。

進化の判別

XDF では、クラス名が一致し、かつフィールド名のタイプが矛盾していないとき、クラスの進化が試みられます。C# アプリケーションと Java アプリケーションの間で、クラス名やフィールド名が若干異なるクラスを一致させようと試みる時は、ClassAlias および FieldAlias 注釈が役立ちます。これらの注釈は、Java アプリケーションまたは C# アプリケーションのいずれか、あるいはその両方に入れることができます。ただし、Java アプリケーションでクラスを検索することは、C# アプリケーションで ClassAlias を定義するよりも効率が悪いことがあります。ClassAlias および FieldAlias 注釈について詳しくは、[ClassAlias および FieldAlias 注釈](#)を参照してください。

シリアライズされたデータ内の欠落フィールドの影響

クラスのコンストラクターはデシリアライゼーション時に呼び出されないため、欠落フィールドは、言語に基づいてそれに割り当てられているデフォルト値を持つこととなります。新しいフィールドを追加するアプリケーションは、旧バージョンのクラスが取得されたとき、欠落フィールドを検出して対処できなければなりません。

古いアプリケーションが新しいフィールドを保持するようにするための唯一の方法であるデータの更新

アプリケーションがフェッチ操作を実行し、クライアントからのシリアライズされた値に含まれている一部のフィールドが欠落している旧バージョンのクラスでマップを更新することがあります。その後、サーバーはサーバー上で値をマージし、元のバージョンのいずれかのフィールドが新しいレコードにマージされるかどうかを判別します。アプリケーションがフェッチ操作を実行した後、エントリーを削除して挿入した場合は、元の値にあったフィールドが失われます。

マージ機能

配列やコレクション内のオブジェクトは XDF によってマージされません。配列やコレクションに対する更新がその配列のエレメントやタイプを変更するように意図されているかどうかは必ずしも明らかとは限りません。マージが位置付けに基づいて行われた場合は、配列内のエントリーが移動されると、XDF では、関連付けるように意図されていないフィールドがマージされることがあります。結果として、XDF では配列やコレクションの内容のマージが試行されないこととなります。ただし、新しいバージョンのクラス定義で配列を追加した場合は、その配列が旧バージョンのクラスにマージし直されます。

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義

ClassAlias および FieldAlias 注釈を使用して、Java™ と .NET クラス間でのデータ・グリッド・データの共有を使用可能にします。

始める前に

- IBM® eXtremeIO が構成されている必要があります。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

このタスクについて

既存の Java クラスがあり、対応する C# クラスを作成する場合は、ClassAlias 注釈と FieldAlias 注釈の使用を検討することができます。このシナリオでは、Java クラス名を含む注釈を C# クラスに追加できます。ClassAlias および FieldAlias 注釈について詳しくは、[ClassAlias および FieldAlias 注釈](#)を参照してください。

手順

ClassAlias および FieldAlias 注釈を使用して、Java クラスと C# クラス間でオブジェクトを相関付けます。 Java

図 1. ClassAlias および FieldAlias 注釈を使用した Java の例

```
@ClassAlias("Employee")
class com.company.department.Employee {

    @FieldAlias("id")
    int myId;

    String name;
}
```

.NET

図 2. ClassAlias および FieldAlias 属性を使用した .NET の例

```
[ ClassAlias( "Employee" ) ]
class Com.MyCompany.Employee {

    [ FieldAlias("id") ]
    int identifier;

    string name;
}
```

[ClassAlias および FieldAlias 注釈](#)

ClassAlias および FieldAlias 注釈を使用して、クラス間でのデータ・グリッドのデータの共有を可能にします。2 つの Java クラス間または Java クラスと .NET クラス間でデータを共有することができます。

.NET [ClassAlias および FieldAlias 注釈](#)

ClassAlias および FieldAlias 注釈を使用して、クラス間でのデータ・グリッドのデータの共有を可能にします。2 つの Java クラス間または Java クラスと .NET クラス間でデータを共有することができます。

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

親トピック: .NET **2.5+** [.NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[ClassAlias および FieldAlias 注釈](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

関連情報:

[レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成](#)

ClassAlias および FieldAlias 注釈

ClassAlias および FieldAlias 注釈を使用して、クラス間でのデータ・グリッドのデータの共有を可能にします。2つの Java™ クラス間または Java クラスと .NET クラス間でデータを共有することができます。

2つのクラスを同じ名前およびフィールドで定義した場合は、データ・グリッドのデータは自動的にクラス間で共有されます。例えば、Customer1 クラスが Java アプリケーションにあり、同じフィールドを持つ Customer1 クラスが .NET アプリケーションにある場合は、データはこれらのクラス間で共有されます。この例では、クラス名にクラス修飾子も含まれていて、そのクラス修飾子が Java のパッケージ名でもあり、C# の名前空間名でもあることが前提になっています。名前空間名とパッケージ名は一致するため、これらの名前は自動的に共有されます。次の例を参照してください。これらの名前は両方とも大/小文字を区別しません。

```
Java:
package com.mycompany.app
public class SampleClass {
    int field1;
    String field2;
}
```


```
C#
namespace Com.MyCompany.App
public class SampleClass {
    int field1;
    string field2;
}
```

ただし、異なる名前を持つクラス間でデータを相関することもできます。データ・グリッドに保管するデータを異なる名前の異なるクラス間で相関するには、ClassAlias 注釈または FieldAlias 注釈を使用します。

2つの Java アプリケーション間: 異なる名前を持つ2つの異なるクラスを別々の Java アプリケーション環境で定義することができます。同じ ClassAlias 注釈を持つクラスにマークを付けることによって、この2つのクラス間ですべてのフィールドおよびフィールド・タイプが突き合わされます。これらのクラスは、異なるクラス名を持っている場合でも、同じクラス・タイプ ID で相関されます。そのため、異なる Java アプリケーション・ランタイムでは、同じクラス・タイプ ID とメタデータがこれらのクラス間で再使用されます。

Java アプリケーションと .NET アプリケーション間: C# アプリケーション内で類似の注釈を使用して、C# クラスを Java クラスと相関させることができます。クラス C# に対して定義されている ClassAlias 属性およびフィールドが、同じ ClassAlias 注釈を持つ Java クラスに突き合わされます。

親トピック: [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

親トピック:  [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

関連タスク:

[Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

関連情報:

[レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成](#)

PartitionKey 注釈を使用したキーから区画へのマップ

PartitionKey 別名を使用して、データが保存される区画を判別するためにハッシュ・コード計算を実行するフィールドまたは属性を識別します。PartitionKey 注釈は、キー属性でのみ有効です。

始める前に

eXtreme Data Format (XDF) を使用している必要があります。IBM eXtremeIO を使用している場合、XDF は使用可能に設定されています。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

このタスクについて

PartitionKey 別名を設定して、複数のクラスでデータが同じ区画に保存されるようにすることができます。例えば、PartitionKey 値を departmentID キーに設定した場合は、従業員レコードは、同じ区画に配置されることになります。

PartitionableKey インターフェースは既存の Java インターフェースであり、C# の PartitionableKey 注釈よりも優先されます。

手順

- **Java** Java アプリケーションのフィールドに PartitionKey 注釈を定義します。

```
class Employee {
    int empId;

    @PartitionKey(order = 0)
    int deptId;
}
```

複数のキーで PartitionKey 注釈を設定できます。また、クラスで PartitionKey 別名を設定できます。Java アプリケーションで PartitionKey 注釈を設定する方法のさらなる例については、[Java API 資料: 注釈タイプ PartitionKeys](#)を参照してください。

- **.NET** .NET アプリケーションのフィールドで PartitionKey 属性を定義します。

```
class Employee {
    int empId;

    [PartitionKey]
    int deptId;
}
```

.NET クラスでも PartitionKey 属性を設定できます。詳しくは、[『.NET API documentation: PartitionKeyAttribute Class』](#)を参照してください。

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

親トピック: **.NET 2.5+** [.NET API を使用したデータ・グリッド・アプリケーションの開発](#)

Java および C# の等価データ型

エンタープライズ・データ・グリッド・アプリケーションを開発するときは、Java アプリケーションと C# アプリケーションの間でデータ型に互換性がなければなりません。

表 1. Java および C# 間での等価データ型

Java 型	C# 型
boolean	bool
java.lang.Boolean	bool?
byte	sbyte または byte
java.lang.Byte	sbyte?
short	short?、ushort
java.lang.Short	short?、ushort?
int	int、uint、ushort
java.lang.Integer	int?、uint?
long	long、ulong、uint
java.lang.Long	long?、ulong?、uint?
short または int	ushort
java.lang.Short または java.lang.Integer	ushort?
int または long	uint
java.lang.Integer または java.lang.Long	uint?
long または BigInteger	ulong
java.lang.Long または java.lang.BigInteger	ulong?
char, java.lang.Character	char
java.lang.Character	char?
float または java.lang.Float	float
java.lang.Float	float?
double	double
java.lang.Double	double?
java.math.BigDecimal	decimal または decimal?
java.math.BigInteger	decimal、long または ulong?
java.lang.String	ストリング (string)
java.util.Date、java.util.Calendar	System.DateTime
java.util.Date(rounding)、 java.util.Calendar(rounding)	System.DateTime
java.util.ArrayList	System.Collections.ArrayList, System.Collections.Generic.List
java.util.HashMap	System.Collections.Generic.Dictionary、 System.Collections.Hashtable
java.util.LinkedList	System.Collections.Generic.LinkedList
java.util.ArrayList、java.util.Vector	System.Collections.Generic.List
java.util.Stack	System.Collections.Generic.Stack
java.util.Vector	System.Collections.ArrayList、 System.Collections.Generic.List

親トピック: [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

WebSphere DataPower XC10 アプライアンスの保護

アプライアンス内のデータ・グリッドは、保護する必要がある機密情報を保管します。

始める前に

このシナリオのいくつかの要素、例えば、連邦情報処理標準 140-2 (FIPS) を使用可能な場合などは、集合内のすべてのメンバーが最新レベルでなければなりません。保護するアプライアンスが集合のメンバーである場合は、その集合のすべてのメンバーにアップグレード済みのファームウェアがなければなりません。そうでないと、このシナリオのタスクを完了できません。

このタスクについて

WebSphere® DataPower® XC10 アプライアンスには、総合的なセキュリティー管理が含まれています。デフォルト構成には、ユーザーが変更する必要があるデフォルトのパスワード、SSL 鍵、および認証の秘密が含まれています。以下のシナリオを完了して構成を変更し、その後アプライアンスのデプロイメントをセキュアにします。

セキュア・デプロイメントでは、セキュリティーを最適化するために、複数の層の保護を使用します。保護の最初のエレメントは、ネットワークをセグメント化するファイアウォールの使用です。Web アプリケーションの標準階層モデルは、Web クライアント、HTTP サーバーのプレゼンテーション層、アプリケーション・サーバーから成るアプリケーション層、データ層、およびストレージ層で構成されます。

WebSphere DataPower XC10 アプライアンスアプライアンスは、データ層の一部としてデプロイされます。標準プラクティスでは、1つのファイアウォールによって保護される非武装地帯 (DMZ) にプレゼンテーション層のサーバーを配置し、複数のファイアウォールによって保護されるネットワーク・セグメントにアプリケーション、データ、およびストレージの各層を配置します。アプライアンスを DMZ にデプロイしないでください。業界の標準プラクティスに従って、データ層の他のすべてのエレメントと同様にアプライアンスも保護する必要があります。

ただし、セキュリティー上の脅威に対する最適な保護のために、いくつかの追加手段によってアプライアンス操作やデータ・グリッドに保管されたデータを保護する徹底した防御メカニズムを使用してください。これらの追加手段は、外部からの脅威に対する保護に役立つだけでなく、アプライアンスが存在するネットワーク・セグメントにアクセスできる可能性がある従業員や請負業者による無許可データ・アクセスも防止します。

このシナリオのステップは、WebSphere DataPower XC10 アプライアンスの Web コンソールで実行されます。これらの各ステップは、プログラムから HTTP コマンド行インターフェースを呼び出すことによって自動化することもできます。HTTP コマンド行インターフェースの詳細については、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

1. **2.5+** [データ・グリッドへのセキュアなアクセスの構成](#)

管理アクセスを構成し、アプライアンス集合の設定を指定して、データ・グリッドへのセキュアな認証と許可を構成します。

2. **2.5+** [データ・グリッドのセキュリティーを使用可能に設定する](#)

データ・グリッドの作成後、データ・グリッド (data grid)のセキュリティーはデフォルトで使用不可になります。データ・グリッド (data grid)のセキュリティー設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

3. **2.5+** [クライアント・セキュリティーの構成](#)

アプライアンスのデータ・グリッドを保護したら、クライアントがその保護されているデータ・グリッドに接続できるようにクライアントを構成する必要があります。

親トピック: **2.5+** [シナリオ](#)

データ・グリッドへのセキュアなアクセスの構成

管理アクセスを構成し、アプライアンス集合の設定を指定して、データ・グリッドへのセキュアな認証と許可を構成します。

このタスクについて

セキュリティ上の脅威に対する最適な保護のために、以下のステップを実行してください。これらのステップは、外的脅威を防止したり、アプライアンスがクライアントとサーバーの間の通信を行うネットワーク・セグメントにアクセスできる可能性がある従業員や請負業者による無許可データ・アクセスを防止したりするのに役立ちます。

重要: 管理コンソールでサーバー間通信やネットワーク上のデータの保護のための集合設定を指定するときに、セキュリティを制御する設定を構成することができます。これらの設定を変更した場合、集合全体を再始動する必要があります。これらの設定を指定するには、「集合」 > 「設定」とクリックします。効率化のために、すべての変更を同時に行ってから、それらの変更を実行依頼するようにしてください。そうすれば、集合の再始動は一度だけで済みます。

手順

1. 管理アクセスを保護します。

管理ユーザー ID (xcadmin) とデフォルトの管理パスワード (xcadmin) を使用してアプライアンスを構成します。このユーザー ID とパスワードにより、デバイスと集合のすべての管理機能およびデータに対する全アクセス権限が付与されます。推測するのが難しいルート管理パスワードを構成する必要があります。このアクションを実行するには、Web コンソールで、「集合」 > 「ユーザー」とクリックします。「管理者」ユーザーを選択し、パスワードを編集してください。

2. SSL トランスポート・セキュリティを構成します。

アプライアンスは、デフォルトの鍵ストアとトラストストアが構成されて出荷されます。デフォルトのトラストストアには、デフォルトの鍵ストアの署名者証明書が含まれます。この証明書はどのアプライアンスにも同梱されているので、SSL 構成をセキュアにするためには、これを取り換える必要があります。証明書の取り換えには、いくつかのステップが必要です。

- a. 新規に証明書と秘密鍵を作成します。この証明書は、自己署名のもの、あるいは、ローカルの認証局が発行するもの (推奨) のいずれかを使用します。
- b. トラストストアを作成または変更します。1つの方法は、デフォルトのトラストストアをアプライアンスからダウンロードして、「**IBM WebSphere DataPower XC10**」というラベルの付いたトラストストアの署名者証明書を削除して、新しい鍵ストアでトラストを確立する証明書を追加することです。新しい鍵ストアに自己署名の証明書がある場合は、その証明書をトラストストアにインポートする必要があります。新しい鍵ストアにローカルの認証局により発行された証明書がある場合は、その認証局のルート証明書をトラストストアに追加する必要があります。
- c. 新しい鍵ストアおよびトラストストアをアップロードします。
- d. 鍵ストアの証明書の名前と一致するように証明書別名の設定を変更します。すべてのデータ・グリッド・クライアント用のトラストストア・ファイルも、アプライアンスの証明書とのトラストを確立するために更新する必要があります。

Java ランタイム環境 (JRE) に付属している keytool コマンドを使用して、鍵ストアとトラストストアを管理します。他の鍵管理ツールも同様に使用できます。鍵ストア・ファイルとトラストストア・ファイルを管理するために実行する以下の管理用タスクを参照してください。

- 自己署名証明書を生成します。

```
keytool -genkey -alias ogsample -keystore key.jks -storetype JKS -keyalg  
rsa -dname "CN=ogsample, OU=OGSample, O=acme, L=Your City, S=Your State,  
C=Your Country" -storepass ogpass -keypass ogpass -validity 3650
```

- アプライアンスからダウンロードするトラストストアの中から、アプライアンスのデフォルトの署名者証明書を取得します。

```
keytool -delete -alias "ibm websphere datapower xc10" -keystore trust.jks  
-storetype JKS -keypass xc10pass
```

- 鍵ストアから新規証明書をエクスポートします。

```
keytool -export -alias ogsample -keystore key.jks -file temp.key -  
storepass ogpass
```

- トラストストアにその証明書をインポートします。

```
keytool -import -noprompt -alias ogsamplepublic -keystore trust.jks -file temp.key -storepass xc10pass
```

このインポートのステップは、鍵ストアからエクスポートされた証明書をトラストするようトラストストアを構成するために実行する必要があります。このステップは、いくつかのコンテキストで使用されます。クライアントのトラストストアは、アプライアンスの鍵ストアの証明書をトラストするように更新される必要があります。構成によっては、アプライアンスのトラストストアが、クライアントの鍵ストアの証明書をトラストするように更新される必要があります。クライアントの鍵ストアとアプライアンスの鍵ストアの両方の証明書が、ローカルの認証局によって発行されるのが理想的です。この場合、認証局のルート署名者証明書をインポートすることでトラストが確立されます。

- 鍵ストアとトラストストア上のパスワードを変更します。

```
keytool -storepasswd -new newpass -keystore trust.jks -storepass xc10pass
```

- 新規鍵ストアとトラストストアをアプライアンス集合にアップロードします。

- e. 次のオプションの中から SSL モードを指定します: **TCP/IP** (この場合、SSL はデータ・グリッド通信には使用されません)、**TLS サポート**、および **TLS 必須**。

セキュリティのためには、「**TLS 必須**」を推奨します。TLS が使用されない場合、パスワードおよび機密グリッド・データが、グリッド・クライアントとアプライアンスを接続するネットワーク・リンク上を暗号化されないまま受け渡されます。

注: TLS 通信が必要な場合、アプライアンスのトラストストアは、クライアントの鍵ストアの証明書をトラストするように構成されている必要があります。クライアントのトラストストアは、サーバー鍵ストアの証明書をトラストするように構成されている必要があります。[クライアント・セキュリティの構成](#)を参照してください。

- f. 連邦情報処理標準 (FIPS) を使用可能に設定します。

すべての暗号化ネットワーク通信で FIPS 140-2 を使用するようにアプライアンス集合を構成することができます。この標準により、通信で送信されるデータの優れた保護が確保されます。「**FIPS 140-2 暗号方式を使用可能に設定**」を選択して FIPS を使用可能にします。

一部の Web ブラウザー・バージョンは、FIPS 対応サーバーで動作しません。Mozilla Firefox、Microsoft Internet Explorer、および Google Chrome などのほとんどのブラウザの現行バージョンは、FIPS 対応サーバーとの通信をサポートしています。FIPS モードでは SSLv3 がサポートされないため、ブラウザを構成して、TLS を使用できるようにする必要がある可能性があります。詳しくは、ご使用のブラウザの資料を参照してください。

- g. クライアント証明書認証を有効にします。

これを有効にする場合、アプライアンスと通信する各クライアントおよびブラウザの鍵ストアを、アプライアンス・トラストストアによってトラストされている証明書を持つように構成する必要があります。クライアント証明書認証は、ORB トランスポートでは使用されません。これは、HTTPS と XIO トランスポートの場合でのみ使用されます。クライアント証明書認証を有効にしなくてもセキュアな構成は保持されます。

3. サーバー間の認証を構成します。

集合内のアプライアンス間、およびリンクされたアプライアンス・ドメイン間のデータ・グリッド通信は、共有秘密鍵を使用して認証されます。アプライアンスは、各アプライアンスにハードコーディングされているデフォルトの秘密鍵で既に構成されます。セキュアな構成を保持するためには秘密鍵を変更する必要があります。「**出荷時のデフォルト認証秘密鍵をオーバーライド**」を選択して固有の秘密鍵を指定してください。

秘密鍵は、推測するのが難しい長いパスフレーズでなければなりません。秘密鍵を書き留めて、安全な場所に保管してください。リンク・ドメイン内で集合が結合される際、各集合は同じ秘密鍵を使用して構成される必要があります。

4. すべてのデータ・グリッド要求に認証を要求します。

データ・グリッドへのクライアント要求ごとに認証を構成することができます。デフォルトでは、この認証は必須ではありません。しかし、個々のグリッドを保護することによってデータ・グリッドへのアクセスを保護することができます。また、セキュア構成の場合は、すべてのグリッド要求について認証が必要になります。これが設定されると、アプライアンス集合によって認識されるユーザー ID とパスワードを使用して各クライアントを構成する必要があります。また、LDAP 認証の場合は、ユーザー ID とパスワードが LDAP に登録されていなければなりません。LDAP 認証なしにログインできるのは、ルート管理者 (xcadmin ユーザー ID) のみです。

ステップ 2 から 4 が完了したら、変更を実行依頼して、集合を再始動してください。これらの設定は、集合に同化されるアプライアンスに自動的に伝搬されます。FIPS を使用可能に設定した場合は、アプライアンスが集合に同化される前に各アプライアンスで FIPS が使用可能になります。

5. Telnet アクセスを使用不可に設定します。

アプライアンスのデフォルト構成には、アクティブな Telnet サーバーが含まれています。アプライアンスの Telnet 通信は、SSL をサポートしません。セキュアな構成では、Telnet サーバーを使用不可に設定してください。Telnet を使用不可にするには、管理者のユーザー ID とパスワードを使用してアプライアンスへの SSH セッションを確立し、コマンド **platform service telnet disable** を実行します。これは集合全体に対する設定ではありません。つまり、このコマンドをアプライアンスごとに実行する必要があります。コマンド **platform service telnet disable** は、Telnet が使用不可になっている場合に Telnet を開始します。この手順は手動であり、自動化することはできません。

6. LDAP 認証を構成します。

ブラウザー、REST、およびアプライアンスに対するデータ・グリッド・アクセスの認証は、次の 2 つの方法のいずれかで行われます。認証済み ID をアプライアンス集合または LDAP に保管することができます。セキュア構成ではどちらの方法も使用できます。

ヒント: ルート管理者の認証では、LDAP ではなく集合によって検証されたパスワードが常に使用されます。

LDAP 認証を使用する場合、パスワードが暗号化されないままネットワーク上を通過しないようにするために、SSL との LDAP 接続を保護してください。LDAP 接続に対して SSL を使用可能にするには、LDAPS URL (例えば、`ldaps://ldapserver.company.com:636` など) を指定します。ここで、LDAP サーバーとのトラストを確立して SSL 通信を行うために、証明書を使用してアプライアンス・トラストストアを構成する必要があります。LDAP 認証の構成の詳細については、[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)を参照してください。

親トピック: [2.5+ WebSphere DataPower XC10 アプライアンスの保護](#)

次のトピック: [2.5+ データ・グリッドのセキュリティーを使用可能に設定する](#)

関連情報:

↳ [連邦情報処理標準 Java Secure Socket Extension ファイルの構成](#)

データ・グリッドのセキュリティーを使用可能に設定する

データ・グリッドの作成後、データ・グリッド (data grid)のセキュリティーはデフォルトで使用不可になります。データ・グリッド (data grid)のセキュリティー設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

このタスクについて

データ・グリッドが集合の一部である場合、そのデータ・グリッドに対してセキュリティーおよびアクセス制御を構成することができます。セキュリティー設定は、そのデータ・グリッドにアクセスするクライアントが確実に認証されるようにします。すべてのデータ・グリッド・アクセスに対して認証が必要なように集合が構成されている場合、当該データ・グリッドへの認証が常に必要となります。認証を必要とするということは、グリッドへのアクセスが十分に制限されない可能性があるということです。例えば、LDAP 認証の場合は、どの LDAP ユーザーも特定のグリッドにアクセスできます。アクセスを制限するには、そのグリッドに対する許可を使用可能に設定します。

重要: データ・グリッドのセキュリティー設定を変更すると、データ・グリッドは自動的に再始動します。データ・グリッドが再始動すると、データ・グリッド内のすべてのデータが失われます。データ・グリッドのデータの保存を開始する前に、データ・グリッドのセキュリティーを構成します。

REST ゲートウェイを介した通信は、データ・グリッドでセキュリティーを有効にしていなくても、常に安全です。詳しくは、[REST ゲートウェイ: セキュリティー構成](#)を参照してください。

手順

1. ユーザー・インターフェースで、データ・グリッド (data grid)の設定にナビゲートします。「データ・グリッド」>「**data_grid_type**」をクリックします。編集する **data_grid_name** をクリックします。
2. データ・グリッド (data grid)に関するセキュリティーまたは許可を有効にします。「セキュリティーを有効にする」をクリックして、ユーザー・インターフェースにアクセス権限を持つすべてのユーザーがデータ・グリッド (data grid)にアクセスできるようにします。さらにアクセス権限を制限する場合は、「許可を使用可能に設定」をクリックします。

注: セッション・データ・グリッドの許可を使用可能にすることはできません。

許可を使用可能にすると、「アクセス認可」リストにユーザーまたはユーザー・グループのリストを指定できます。「許可を使用可能に設定」が選択された場合、このアクセス・リストに登録されたユーザーのみがデータ・グリッド (data grid)データにアクセスできます。ユーザー・インターフェースに表示されるデフォルトのアクセス・タイプの名前をクリックすることで、ユーザーまたはユーザー・グループに対して以下のアクセス権限を割り当てることができます。

- 「読み取り」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドのデータの読み取りまたは照会をすることができます。
- 「書き込み」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドに対するデータの読み取り、照会、および書き込みができます。
- 「作成」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドの読み取り、照会、書き込み、挿入、および動的マップの作成ができます。
- 「すべて」: この許可が割り当てられると、ユーザーまたはユーザー・グループは、データ・グリッドの読み取り、照会、書き込み、挿入、動的マップの作成、除去、およびデータの無効化を行うことができます。アプライアンス管理者には、デフォルトで「すべて」の許可があります。

セキュリティーおよび許可の設定を変更する場合、5 分のタイムアウト値があります。

- 認証タイムアウト: 既にデータ・グリッドに対して認証されたユーザーのユーザー・パスワードを変更した場合、元の資格情報は最大 5 分間有効なままです。
- 許可タイムアウト: ユーザーの許可を除去すると、そのユーザーは、最大 5 分間、その許可を所有し続けます。このタイムアウトは、除去された許可に対してのみ適用されます。ユーザーに許可を付与すると、そのユーザーは、直ちに許可を取得します。

親トピック: [2.5+ WebSphere DataPower XC10 アプライアンスの保護](#)

前のトピック: [2.5+ データ・グリッドへのセキュアなアクセスの構成](#)

次のトピック: [2.5+ クライアント・セキュリティーの構成](#)

親トピック: [データ・グリッドの構成](#)

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadmin/パスワード](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

Java

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

Java [クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)
[データ・グリッド・アプリケーション用の TLS の構成](#)
[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

クライアント・セキュリティーの構成

アプライアンスのデータ・グリッドを保護したら、クライアントがその保護されているデータ・グリッドに接続できるようにクライアントを構成する必要があります。

このタスクについて

アプライアンスが TLS を必要とするように構成されている場合、クライアントは SSL トランスポート用に構成されている必要があります。さらにクライアントに適切な鍵ストアとトラストストアが必要です。認証が必要な場合は、クライアントは、ユーザー ID とパスワードを使用して構成されている必要もあります。具体的な手順は、クライアントがスタンドアロンのインストール済み環境であるか、WebSphere® Application Server プロセスで稼働するインストール済み環境であるかによって異なります。

スタンドアロン環境の場合は、データ・グリッド・アプリケーションに渡される設定を含む `client.properties` ファイルを構成する必要があります。ご使用の環境に WebSphere Application Server がある場合は、WebSphere Application Server ツールを使用してクライアント・セキュリティーを構成してください。

手順

1. クライアントがスタンドアロンのインストール済み環境である場合は、クライアント・プロパティー・ファイルを使用して、クライアントとサーバーの間の通信およびクライアント・セキュリティーを構成します。このファイルのロケーションとフォーマットについては、[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)を参照してください。

Java™ クライアントまたは .NET クライアントのために構成できるセキュリティー・プロパティーについては、[クライアント・プロパティー・ファイル](#)を参照してください。

2. クライアントが WebSphere Application Server プロセスで稼働している場合は、WebSphere Application Server ツールを使用して、鍵ストアやトラストストアなどの SSL 構成値を構成します。

証明書をアプライアンス鍵ストアから WebSphere Application Server トラストストアにインポートするユーティリティーが提供されています。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

アプライアンス鍵ストアからの証明書を信頼するように、その WebSphere Application Server トラストストアを構成する必要があります。アプライアンスがクライアント証明書認証用に構成されている場合、またはオブジェクト・リクエスト・ブローカー (ORB) 通信が使用されている場合は、WebSphere Application Server 鍵ストアからの証明書も信頼するようにアプライアンス・トラストストアを構成する必要があります。

認証を必要とするようにアプライアンスが構成されている場合は、WebSphere Application Server のもとで実行されるアプライアンス・クライアントも、認証資格情報を使用して構成する必要があります。Java スタンドアロン・クライアントの場合と同様、プロパティー・ファイルを使用してクライアントを構成することができます。ただし、アプライアンス・クライアントが WebSphere Application Server にインストールされている場合、WebSphere Application Server 管理コンソールが拡張され、この管理コンソールで認証資格情報を指定できるようになります。この拡張により、カタログ・サービス・ドメインを構成したり、クライアント認証プロパティーを指定したりできます。

3. サーバーに連邦情報処理標準 (FIPS) が設定されている環境でクライアントが稼働している場合、クライアントは TLS ハンドシェイク・プロトコルを使用してアプライアンスと通信する必要があります。SSL ハンドシェイク・プロトコルは使用できません。TLS プロトコルは、クライアント・プロパティー・ファイル内の設定 `protocol=TLS` で構成されます。
 - XIO が使用され、クライアント・プロパティー・ファイルにプロトコル設定がない場合のデフォルト設定は SSL および TLS です。この設定は、アプライアンスで FIPS が使用可能になっている場合に機能します。
 - スタンドアロン・クライアントで ORB 通信が使用される場合、FIPS モードでアプライアンスと通信するためには、クライアント・プロパティー・ファイルでプロトコル設定を TLS にしなければなりません。
 - ORB 通信が使用され、クライアントが WebSphere Application Server と共に稼働している場合、ハンドシェイク・プロトコルを設定するために WebSphere セキュリティー設定が使用されます。デフォルトでは、この設定は SSL_TLS です。これは、アプライアンスで FIPS が使用可能になっている場合に機能します。

WebSphere Application Server 管理コンソールを使用して、この設定を TLS または SSL_TLS に構成することができます。「セキュリティー」 > 「SSL 証明書および鍵管理」 > 「エンドポイント・セキュリティー構成の管理」をクリックします。セル名を選択し、「SSL 構成」 > 「CellDefaultSSLSettings」 > 「保護品質 (QOP) 設定」をクリックします。プロトコル・フィールドが TLS または SSL_TLS に設定されていることを確認してください。

親トピック: [2.5+ WebSphere DataPower XC10 アプライアンスの保護](#)

前のトピック: [2.5+ データ・グリッドのセキュリティーを使用可能に設定する](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション

以前に設定したメモリー間複製セッションまたはデータベース・セッションをマイグレーションして、WebSphere® eXtreme Scale セッション管理を使用できます。

始める前に

- WebSphere Application Server において実行されているクライアント・アプリケーション (クラスター) の用のセッション・サポートでは、WebSphere eXtreme Scale を、WebSphere Application Server ノード・デプロイメント (デプロイメント・マネージャー・ノードを含む) 上にインストールする必要があります。[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。

このタスクについて

このシナリオの手順は、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

WebSphere DataPower XC10 アプライアンス セッションへのマイグレーションの一部として、WebSphere Application Server 管理コンソールの前の構成設定をメモする必要があります。WebSphere DataPower XC10 アプライアンス セッションにマイグレーションする際に、構成設定で、データベース・セッションおよびメモリー間セッション用に既に構成されている内容を反映する必要があります。

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

データ・グリッド内の HTTP セッションへのマイグレーションの一部として、WebSphere Application Server 管理コンソールでカタログ・サービス・ドメインを作成する必要があります。

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

WebSphere Application Server 管理コンソール内のメモした以前の構成設定を使用して、アプリケーションまたはアプリケーション・サーバーのいずれかを WebSphere eXtreme Scale セッション管理に関連付ける必要があります。

親トピック: [2.5+ シナリオ](#)

親トピック: [セッション・データ・グリッドの作成](#)

関連タスク:

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

WebSphere Application Server 管理コンソールの前の構成設定のメモ

WebSphere® DataPower® XC10 アプライアンス セッションへのマイグレーションの一部として、WebSphere Application Server 管理コンソールの前の構成設定をメモする必要があります。WebSphere DataPower XC10 アプライアンス セッションにマイグレーションする際に、構成設定で、データベース・セッションおよびメモリー間セッション用に既に構成されている内容を反映する必要があります。

このタスクについて

WebSphere Application Server 管理コンソール内にメモする必要がある特定の設定があります。これらの値は、splicer.properties ファイルの更新時に必要になります。この手順のステップは、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale クライアント バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

手順

- WebSphere Application Server 管理コンソールを開始します。
 - 以前にサーバー・レベルで設定を構成したことがある場合は、以下にアクセスします。
 - 「サーバー」>「サーバー・タイプ」>「**WebSphere Application Server**」
 - 「アプリケーション・サーバー」領域で、ご使用のサーバー名を選択します。
 - 「コンテナ設定」領域で、「セッション管理」をクリックします。
 - 以前にアプリケーション・レベルで設定を構成したことがある場合は、以下にアクセスします。
 - 「アプリケーション」>「すべてのアプリケーション」。
 - 「アプリケーション・サーバー」領域で、ご使用のアプリケーション名を選択します。
 - 「**Web** モジュール・プロパティ」領域で、「セッション管理」をクリックします。
- 「一般プロパティ」で、「オーバーフローの許可」チェック・ボックスを選択します。
- 「一般プロパティ」領域で、WebSphere Application Server 設定をメモします。これらの値は後から、splicer.properties ファイルのプロパティを更新する際に必要になります。

表 1. splicer.properties ファイルを更新するための構成設定

WebSphere Application Server 管理コンソールでの設定	splicer.properties ファイルで更新するプロパティ
Cookie 使用可能 (Enable cookies)	useCookies
URL 再書き込みを有効にする	useURLEncoding
メモリー内の最大セッション・カウント	sessionTableSize

- 「一般プロパティ」領域で、「**Cookie 使用可能 (Enable cookies)**」チェック・ボックスが選択されている場合は、それをクリックし、WebSphere Application Server 設定をメモします。これらの値は後から、splicer.properties ファイルのプロパティを更新する際に必要になります。

表 2. splicer.properties ファイルのプロパティの構成設定

WebSphere Application Server 管理コンソールでの設定	splicer.properties ファイルで更新するプロパティ
Cookie ドメイン	cookieDomain
Cookie パス	cookiePath

- 「セッション管理」をクリックし、「追加プロパティ」領域で、「分散環境設定」をクリックします。
- 「分散セッション」領域で、以前のデータベース構成またはメモリー間複製構成を「なし」に変更します。
- 「カスタム・チューニング・プロパティ (**Custom Tuning Properties**)」をクリックし、WebSphere Application Server 設定をメモします。これらの値は後から、splicer.properties ファイルのプロパティを更新する際に必要になります。

表 3. splicer.properties ファイルのプロパティの構成設定

WebSphere Application Server 管理コンソールでの設定	splicer.properties ファイルで更新するプロパティ
書き込み頻度	replicationInterval
書き込みの内容	fragmentedSession

次のタスク

次に、WebSphere DataPower XC10 アプライアンス セッション用のカタログ・サービス・ドメインを作成します。

親トピック: [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成

データ・グリッド内の HTTP セッションへのマイグレーションの一部として、WebSphere® Application Server 管理コンソールでカタログ・サービス・ドメインを作成する必要があります。

このタスクについて

この手順のステップは、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale クライアント バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

WebSphere Application Server 管理コンソールで、データ・グリッド用のカタログ・サービス・ドメインを作成します。詳しくは、[WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)を参照してください。

手順

1. WebSphere Application Server 管理コンソールを開始します。
2. 上部メニューで、「システム管理」>「**WebSphere eXtreme Scale**」>「カタログ・サービス・ドメイン」をクリックします。

注: WebSphere eXtreme Scale が表示されない場合は、WebSphere Application Server プロファイルがデータ・グリッド用に拡張されていません。を参照してください。
3. 「新規」をクリックします。
4. 「名前」ボックスにカタログ・サービスの名前を指定します。
5. 「カタログ・サーバー」領域で、「リモート・サーバー」を選択し、ボックス内にリモート・サーバーの場所または名前を指定します。
6. 「リスナー・ポート」ボックスにポート番号を指定します。
7. 「適用」または「**OK**」をクリックし、構成を保存します。

次のタスク

次に、WebSphere Application Server 管理コンソール内のメモした以前の構成設定を使用して、アプリケーションまたはアプリケーション・サーバーのいずれかを WebSphere eXtreme Scale セッション管理に関連付けます。

親トピック: [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

以前の構成設定を使用するための **WebSphere DataPower XC10** アプリケーションの構成

WebSphere® Application Server 管理コンソール内のメモした以前の構成設定を使用して、アプリケーションまたはアプリケーション・サーバーのいずれかを WebSphere eXtreme Scale セッション管理に関連付ける必要があります。

このタスクについて

この手順のステップは、WebSphere Application Server 管理コンソールのバージョン 8.5 用です。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注: WebSphere eXtreme Scale クライアント バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

手順

- WebSphere eXtreme Scale セッション管理に関連付けられるようにアプリケーションを構成する場合は、以下の手順に従います。
 1. WebSphere Application Server 管理コンソールを開始します。
 2. 上部メニューで、「アプリケーション」>「すべてのアプリケーション」をクリックします。
 3. 「**WebSphere** エンタープライズ・アプリケーション」領域で、「アプリケーション名」を選択します。
 4. 「**Web** モジュール・プロパティ」領域で、「セッション管理」をクリックします
 5. 「**eXtreme Scale** セッション管理設定」をクリックします。
 6. WebSphere eXtreme Scale が表示されない場合は、WebSphere Application Server プロファイルが WebSphere eXtreme Scale 用に拡張されていません。詳しくは、を参照してください。
 7. WebSphere eXtreme Scale クライアント 用にアプリケーションを構成するには、以下の手順に従います。
 - a. 「セッション・パーシスタンスの管理」リストで、「**IBM WebSphere DataPower XC10** アプリケーション」を選択します
 - b. リストから、作成したカタログ・サービス・ドメインの IP またはホスト名を指定します。
 - c. ユーザー名とパスワードを指定し、接続をテストします。
 - d. 新規データ・グリッドを指定するか、既存のものを選択します。
 8. 「適用」または「**OK**」をクリックし、構成を保存します。
 9. このアプリケーション用に新規 splicer.properties ファイルが作成されます。splicer.properties ファイルの場所は、新規プロパティ {application name}, com.ibm.websphere.xs.sessionFilterProps の値です。このカスタム・プロパティを見つけるには、「システム管理」>「セル」にアクセスし、「カスタム・プロパティ」をクリックします。
 10. [WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)で取得した値を使用して、splicer.properties ファイルを更新します。
 11. アプリケーション・サーバー・プロセスを再始動します。

注: デプロイメント・マネージャー・レベルで splicer.properties を変更して、プロパティがノード・エージェントに同期されるようにします。ノード・レベルで splicer.properties を更新した場合は、次回同期時に、デプロイメント・マネージャーによって splicer.properties ファイルが上書きされます。

注: データベース・セッション管理に戻ってから WebSphere eXtreme Scale セッション管理に戻ると、splicer.properties ファイルが再作成されるため、行った変更がすべてオーバーライドされます。デプロイメント・マネージャーからノードへのファイル同期プロセスおよび変更される内容については、[「System Management File Synchronization」](#)を参照してください。

- WebSphere eXtreme Scale セッション管理に関連付けられるようにアプリケーション・サーバーを構成する場合は、以下の手順に従います。
 1. WebSphere Application Server 管理コンソールを開始します。
 2. 上部メニューで、「サーバー」>「サーバー・タイプ」>「**WebSphere Application Server**」をクリックします。
 3. 「アプリケーション・サーバー」領域で、ご使用のサーバー名を選択します。
 4. 「コンテナ設定」領域で、「セッション管理」をクリックします。
 5. 「**eXtreme Scale** セッション管理設定」をクリックします。

注: WebSphere eXtreme Scale が表示されない場合は、WebSphere Application Server プロファイルが WebSphere eXtreme Scale 用に拡張されていません。詳しくは、を参照してください。

 6. WebSphere eXtreme Scale クライアント 用にアプリケーションを構成するには、以下の手順に従います。
 - a. 「セッション・パーシスタンスの管理」リストで、「**IBM WebSphere DataPower XC10** アプリケーション」を選択します
 - b. リストから、作成したカタログ・サービス・ドメインの IP またはホスト名を指定します。
 - c. ユーザー名とパスワードを指定し、接続をテストします。

- d. 新規データ・グリッドを指定するか、既存のものを選択します。
7. 「適用」または「OK」をクリックし、構成を保存します。
8. このアプリケーション用に新規 splicer.properties ファイルが作成されます。 splicer.properties ファイルの場所は、新規プロパティ com.ibm.websphere.xs.sessionFilterProps の値です。このカスタム・プロパティを見つけるには、「サーバー」>「サーバー・タイプ」>「**WebSphere Application Server**」にアクセスします。
9. 「アプリケーション・サーバー」領域で、ご使用のサーバー名を選択します。
10. 「サーバー・インフラストラクチャー」領域で、「カスタム・プロパティ」を選択します。
11. [WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)で取得した値を使用して、splicer.properties ファイルを更新します。
12. アプリケーション・サーバー・プロセスを再始動します。

注: デプロイメント・マネージャー・レベルで splicer.properties を変更して、プロパティがノード・エージェントに同期されるようにします。ノード・レベルで splicer.properties を更新した場合は、次回同期時に、デプロイメント・マネージャーによって splicer.properties ファイルが上書きされます。

注: データベース・セッション管理に戻ってから WebSphere eXtreme Scale セッション管理に戻ると、splicer.properties ファイルが再作成されるため、行った変更がすべてオーバーライドされます。デプロイメント・マネージャーからノードへのファイル同期プロセスおよび変更される内容については、[「System Management File Synchronization」](#)を参照してください。

タスクの結果

これで、WebSphere eXtreme Scale セッション管理で、メモリー間複製セッションまたはデータベース・セッション管理用に以前の構成設定が変更されました。

親トピック: [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[splicer.properties ファイル](#)

[| 次へ >](#)

チュートリアル: 単純データ・グリッド・アプリケーションの入門

入門用サンプル・アプリケーションを使用して、クライアントのインストール済み環境とアプライアンスの間の接続を検証できます。入門用サンプル・アプリケーションは、エンタープライズ・データ・グリッドの概要を紹介しています。

学習目標

- ユーザー・インターフェースでデータ・グリッドを作成する方法を学習する。
- クライアント・アプリケーションの開発について学習する (Java または .NET プログラミング言語での開発)。プログラミング言語間の相互運用方法およびエンタープライズ・データ・グリッドの作成について学習する。
- クライアント・アプリケーションを実行して、データをデータ・グリッドに挿入する。
- web コンソールでデータ・グリッドをモニターする。

所要時間

60 分

[| 次へ >](#)

[< 前へ | 次へ >](#)

入門チュートリアル・レッスン 1.1: データ・グリッドの定義

データ・グリッドは、ユーザー・インターフェースで定義できます。このチュートリアルでは、`my_simple_data_grid` という名前のデータ・グリッドを作成してください。

関連タスク:

[単純データ・グリッドの作成](#)

単純データ・グリッドの作成

単純データ・グリッド (data grid) を使用して、作成、取得、更新、および削除の各操作を実行することができます。

- アプライアンスを初期化して、構成する必要があります。詳しくは、[WebSphere DataPower XC10 アプライアンスのインストール](#) を参照してください。

単純データ・グリッド (data grid) を作成します。ユーザー・インターフェースで、「データ・グリッド」 > 「単純データ・グリッド」をクリックします。追加アイコン (+) をクリックし、作成する単純データ・グリッド (data grid) の名前を指定します。このチュートリアルでは、`my_simple_data_grid` という名前のデータ・グリッドを作成します。

レッスンのチェックポイント

このレッスンでは、以下を学習しました。

- ユーザー・インターフェースで単純データ・グリッドを作成する方法

[< 前へ | 次へ >](#)



[< 前へ | 次へ >](#)

入門チュートリアル・モジュール 2: クライアント・アプリケーションの作成

データ・グリッドでデータを挿入、更新、削除、および取得するクライアント・アプリケーションを作成します。サンプル・アプリケーションを使用して、ご使用の環境でアプリケーションを作成する方法が学習できます。

学習目標

このモジュールのレッスンを完了すると、以下のタスクの実行方法が理解できるようになります。

-  [Java クライアント・アプリケーションの開発](#)
-  [.NET クライアント・アプリケーションの開発](#)
- [エンタープライズ・データ・グリッド・アプリケーションの開発](#)

このモジュールのレッスン

[入門チュートリアル・レッスン 2.1: Java クライアント・アプリケーションの作成](#)

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる Java クライアント・アプリケーションが組み込まれています。

[入門チュートリアル・レッスン 2.2: .NET クライアント・アプリケーションの作成](#)

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる .NET クライアント・アプリケーションが組み込まれています。

[レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成](#)

Java クライアントと .NET クライアントの両方が同じデータ・グリッドを更新できるエンタープライズ・データ・グリッド・アプリケーションを作成するには、ご使用のクラスが互換性を持つようにする必要があります。入門用サンプル・アプリケーションでは、.NET サンプル・アプリケーションが Java デフォルトと一致する別名を持っています。

[< 前へ | 次へ >](#)

入門チュートリアル・レッスン 2.1: Java クライアント・アプリケーションの作成

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる Java クライアント・アプリケーションが組み込まれています。

[wxs_install_root/ObjectGrid/gettingstarted/client/src/](#) ディレクトリーにある `Client.java` ファイルは、カタログ・サーバーへの接続方法、ObjectGrid インスタンスの取得方法、および ObjectMap API の使用法を示したクライアント・プログラムです。ObjectMap API は、データをキー値ペアとして保管し、リレーションシップを持たないオブジェクトのキャッシングに適しています。以下のステップでは、`Client.java` ファイルの内容を検討します。

1. `ClientClusterContext` インスタンスを取得することで、カタログ・サービスに接続します。

カタログ・サーバーに接続するには、`ObjectGridManager` API の `connect` メソッドを使用します。以下のコード・スニペットは、カタログ・サーバーへの接続方法と `ClientClusterContext` インスタンスの取得方法を示します。

```
ClientClusterContext ccc =
ObjectGridManagerFactory.getObjectGridManager().connect(cep, null, null);
```

`connect` メソッドは、接続が成功するまでリスト内の各アプライアンスに接続しようとします。他のアプライアンスの 1 つが応答しなければ、自動フェイルオーバーが提供されます。

カタログ・サーバーへの接続が成功すれば、`connect` メソッドは `ClientClusterContext` インスタンスを戻します。この `ClientClusterContext` インスタンスは、`ObjectGridManager` API から `ObjectGrid` を取得するのに必要です。

2. `ObjectGrid` インスタンスを取得します。

`ObjectGrid` インスタンスを取得するには、`ObjectGridManager` API の `getObjectGrid` メソッドを使用します。`getObjectGrid` メソッドは、`ClientClusterContext` インスタンスと、データ・グリッド・インスタンスの名前との両方を必要とします。`ClientClusterContext` インスタンスは、カタログ・サーバーへの接続中に取得されます。データ・グリッド・インスタンスの名前は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。以下のコード・スニペットは、`ObjectGridManager` API の `getObjectGrid` メソッドを呼び出すことによってデータ・グリッドを取得する方法を示します。

```
ObjectGrid grid = ObjectGridManagerFactory.getObjectGridManager().getObjectGrid(ccc,
"my_simple_data_grid");
```

3. 必要なセキュリティー資格情報を設定します。

クライアント・セキュリティー構成を作成するほか、アプリケーションに提供するユーザー名とパスワードを使用して資格情報生成プログラムを作成します。使用するユーザー名とパスワードには、アプライアンス上のデータ・グリッドにアクセスするための権限が必要です。権限があるユーザーの作成については、[ユーザーおよびグループの管理](#)を参照してください。

```
// Creates a ClientSecurityConfiguration object using the specified
file
ClientSecurityConfiguration clientSC =
ClientSecurityConfigurationFactory.getClientSecurityConfiguration();
clientSC.setSecurityEnabled(true);
// Creates a CredentialGenerator using the passed-in user and
password.
CredentialGenerator credGen = new
UserPasswordCredentialGenerator(username,password);
clientSC.setCredentialGenerator(credGen);
return clientSC;
```

4. `Session` インスタンスを取得します。

取得した `ObjectGrid` インスタンスから、`Session` を取得することができます。`Session` インスタンスは、`ObjectMap` インスタンスの取得とトランザクション区分の実行のために必要です。以下のコード・スニペットは、`ObjectGrid` API の `getSession` メソッドを呼び出すことによって `Session` インスタンスを取得する方法を示します。

```
Session sess = grid.getSession();
```

5. ObjectMap インスタンスを取得します。

Session を取得した後、Session API の `getMap` メソッドを呼び出すことによって、Session インスタンスから ObjectMap インスタンスを取得することができます。`getMap` メソッドに渡すマップ・インスタンス名は、ユーザー・インターフェースで作成したデータ・グリッドと同じ名前です。以下のコード・スニペットは、Session API の `getMap` メソッドを呼び出すことによって ObjectMap を取得する方法を示します。

```
ObjectMap map1 = sess.getMap("my_simple_data_grid");
```

前の例は、データ・グリッドにちなんで名前が付けられるデフォルト・マップ・インスタンスを使用しています。次の例のように、新規マップ名を指定することもできます。

```
ObjectMap map2 = sess.getMap("my_simple_data_grid.CT.P");
ObjectMap map3 = sess.getMap("my_new_map.NONE");
```

`my_simple_data_grid.CT.P` マップは作成時除去とペシミスティック・ロックを使用するマップです。`my_new_map.NONE` マップには、除去もロック設定もありません。詳しくは、[動的マップの構成オプション](#)を参照してください。

6. ObjectMap メソッドを使用します。

ObjectMap インスタンスを取得した後、ObjectMap API を使用できます。ObjectMap インターフェースはトランザクション・マップであり、Session API の `begin` メソッドおよび `commit` メソッドを使用したトランザクション区分を必要とすることに注意してください。アプリケーションに明示的なトランザクション区分がない場合、ObjectMap 操作は自動コミット・トランザクションで実行します。

使用する鍵は、`java.lang.String` または `Integer` といった既存の Java 型にできます。値は、任意の直列化可能オブジェクト・タイプから構成できます。

- 以下のコード・スニペットは、自動コミット・トランザクションでの ObjectMap API の使用方法を示しています。

```
map1.insert(key1, value1);
```

- 一度に 1 つの区画でトランザクションを実行することもできれば、複数の区画でトランザクションを実行することもできます。単一の区画でトランザクションを実行するには、1 フェーズ・コミット・トランザクションを使用します。

```
sess.setTxCommitProtocol(TxCommitProtocol.ONEPHASE);
sess.begin();
map1.insert(k, v);
sess.commit();
```

複数の区画でトランザクションを実行するには、2 フェーズ・コミット・トランザクションを使用します。

```
sess.setTxCommitProtocol(TxCommitProtocol.TWOPHASE);
sess.begin();
map1.insert(k, v);
sess.commit();
```

7. オプション: セッションを閉じます。 Session 操作および ObjectMap 操作がすべて完了したら、`Session.close()` メソッドを使用してセッションを閉じます。このメソッドを実行すると、セッションで使用されていたリソースが返されます。

```
sess.close();
```

結果として、後続の `getSession()` メソッド呼び出しはより早く戻り、ヒープでの Session オブジェクトが少なくなります。

関連概念:

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連タスク:

[Java API 資料へのアクセス](#)

関連情報:

[API 資料](#)

レッスンのチェックポイント

このレッスンでは、データ・グリッドの操作を実行するシンプルなクライアント・アプリケーションを作成する方法について学習しました。

[< 前へ](#) | [次へ >](#)

入門チュートリアル・レッスン 2.2: .NET クライアント・アプリケーションの作成

データ・グリッドのデータを挿入、削除、更新、および取得するには、クライアント・アプリケーションを作成する必要があります。入門用サンプルには、独自のクライアント・アプリケーションの作成方法を学習できる .NET クライアント・アプリケーションが組み込まれています。

- WebSphere® eXtreme Scale クライアント for .NET がインストールされていなければなりません。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストール](#)を参照してください。
- このサンプルのプロジェクト・ファイルは Microsoft Visual Studio 2010 以降と連動します。これより前のバージョンの Microsoft Visual Studio を使用している場合は、独自のプロジェクト・ファイルを作成する必要があります。

以下の目的で .NET 入門用サンプル・アプリケーションを使用することができます。

- WebSphere eXtreme Scale クライアント for .NET が正しくインストールされていることを確認するため。
- データ・グリッドと通信する .NET クライアント用アプリケーションを作成する方法を学んでカスタム・アプリケーションを作成できるようになるため。このサンプルでは、リモート・カタログ・サーバー上のデータ・グリッドに接続する方法を示します。対話モードは、GridMapPessimisticTx マップを使用して手動トランザクションを実行する方法を示します。コマンド行モードは、GridMapPessimisticAutoTx マップを使用した自動コミット・トランザクションを示します。
- Java™ 入門用サンプルと相互運用する方法を学習するため。これらのサンプル・アプリケーションは両方とも、項目を TestKey/TestValue のペアでデータ・グリッドに保管します。 .NET サンプルには、シリアライゼーションおよびデシリアライゼーション用の固有の ID を作成する ClassAlias および FieldAlias 属性があります。Java クライアント・アプリケーションからキー挿入操作が実行されると、.NET クライアントは挿入されたキーに対して取得操作を実行することによって値を取得することができます。

.NET 入門用サンプル・アプリケーションには以下の制約があります。

- ペシミスティック・ロックのみがサポートされます。
- 2 フェーズ・コミット操作はサポートされていません。操作をコミットできる区画は 1 つのみです。複数の区画を伴うコミットを実行すると、MultiplePartitionWriteException 例外が発生します。
- サンプルでは、ヌル値はサポートされません。 .NET API はヌル値を許可しますが、ヌル可能タイプを使用する必要があります。

SimpleClient.csproj プロジェクト・ファイルが [net_client_home/sample/SimpleClient](#) ディレクトリーにあります。このプロジェクト・ファイルは、カタログ・サーバーに接続し、ObjectGrid インスタンスを取得し、ObjectMap API を使用する方法を示すクライアント・プログラムです。ObjectMap API は、データをキー値ペアとして保管し、リレーションシップを持たないオブジェクトのキャッシングに適しています。以下のステップには、SimpleClient.csproj ファイルの主な内容に関する情報が含まれています。Microsoft Visual Studio でこのプロジェクト・ファイルをさらに詳細に調べることができます。

このチュートリアルは、アプリケーションが対話モードで実行されるときに使用される手動トランザクション・マップである IGridMapPessimisticTx の使用を示します。アプリケーションをコマンド行モードで使用した場合は、IGridMapPessimisticAutoTx マップが使用されます。

1. IClientConnectionContext インスタンスを取得することで、カタログ・サービスに接続します。

カタログ・サーバーに接続するには、IGridManager API の connect メソッドを使用します。

```
IGridManager gm = GridManagerFactory.GetGridManager( );
ICatalogDomainInfo cdi = gm.CatalogDomainManager.CreateCatalogDomainInfo( endpoint
);
ccc = gm.Connect( cdi, "SimpleClient.properties" );
```

カタログ・サーバーへの接続が成功すれば、Connect メソッドは IClientConnectionContext インスタンスを戻します。IGridManager API からデータ・グリッドを取得するためには IClientConnectionContext インスタンスが必要です。

2. ObjectGrid インスタンスを取得します。

ObjectGrid インスタンスを取得するには、IGridManager API の GetGrid メソッドを使用します。GetGrid メソッドは、IClientConnectionContext インスタンスと、データ・グリッド・インスタンスの名前との両方を必要とします。IClientConnectionContext インスタンスは、カタログ・サーバーへの接続中に取得されます。データ・グリッド・インスタンスの名前は、objectgrid.xml ファイル内で指定されるグリッドです。

```
grid = gm.GetGrid( ccc, gridName );
```

3. map インスタンスを取得します。

IGrid API の GetGridMapPessimisticTx メソッドを呼び出すことによってマップ・インスタンスを取得することができます。マップの名前をパラメーターとして GetGridMapPessimisticTx メソッドに渡すことでマップ・インスタンスを取得します。

```
peessMap = grid.GetGridMapPessimisticTx<Object, Object>( mapName );
```

4. IGridMapPessimisticTx メソッドを使用します。

マップ・インスタンスが取得された後で、IGridMapPessimisticTx API を使用することができます。

次のコード・スニペットは、IGridMapPessimisticTx API の使用法を示しています。

- IGridMapPessimisticTx API でトランザクションを開始するには、map.Transaction.Begin() メソッドを呼び出す必要があります。このメソッドは、操作を実行できる新しいトランザクションを開始します。

```
case "begin":
    map.Transaction.Begin( );
    return 0;
```

- add メソッドは、新しいキー/値のペアを挿入します。キーが現在存在する場合は、例外がスローされません。

```
case "a":
    if( key == null ) throw new MissingParameterException( "key" );
    if( value == null ) throw new MissingParameterException( "value" );
    map.Add( key, value );
    Console.WriteLine( "SUCCESS: Added key '{0}' with value '{1}',
partitionId={2}", key, value, partitionId );
    return 0;
```

- put メソッドは、キー/値のペアを挿入または更新します。

```
case "p":
    if( key == null ) throw new MissingParameterException( "key" );
    if( value == null ) throw new MissingParameterException( "value" );
    map.Put( key, value );
    Console.WriteLine( "SUCCESS: Put key '{0}' with value '{1}',
partitionId={2}", key, value, partitionId );
    return 0;
```

- replace メソッドは、既存のキー/値のペアを置き換えます。項目が存在しない場合には、例外がスローされます。

```
case "r":
    if( key == null ) throw new MissingParameterException( "key" );
    if( value == null ) throw new MissingParameterException( "value" );
    map.Replace( key, value );
    Console.WriteLine( "SUCCESS: Replaced key '{0}' with value '{1}',
partitionId={2}", key, value, partitionId );
    return 0;
```

- remove メソッドは、キー/値のペアを削除します。

```
case "d":
    if( key == null ) throw new MissingParameterException( "key" );
    map.Remove( key );
    Console.WriteLine( "SUCCESS: Deleted value with key '{0}',
partitionId={1}", key, partitionId );
    return 0;
```

- get メソッドは、与えられたキーの値を取得します。

```
case "g":
    if( key == null ) throw new MissingParameterException( "key" );
    value = ( TestValue )map.Get( key );
    if( value != null )
    {
        Console.WriteLine( "SUCCESS: Value is '{0}',
```



```
partitionId={1}", value, partitionId );
}
else
{
    Console.WriteLine( "FAILED: Key not found" );
}
return 0;
```

- コミットする前の操作で実行した操作を取り消したい場合は、rollback メソッドを使用します。

```
case "rollback":
    map.Transaction.Rollback( );
    return 0;
```

- commit メソッドは、トランザクションで完了した操作をコミットします。

```
case "commit":
    map.Transaction.Commit( );
    return 0;
```

関連タスク:

[.NET](#) [.NET 開発環境の設定](#)

[.NET](#) [WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)

レッスンのチェックポイント

このレッスンでは、データ・グリッド操作を実行する簡単な .NET クライアント・アプリケーションを作成する方法について学習しました。

[< 前へ](#) | [次へ >](#)

[< 前へ](#) | [次へ >](#)

レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成

Java™ クライアントと .NET クライアントの両方が同じデータ・グリッドを更新できるエンタープライズ・データ・グリッド・アプリケーションを作成するには、ご使用のクラスが互換性を持つようにする必要があります。入門用サンプル・アプリケーションでは、.NET サンプル・アプリケーションが Java デフォルトと一致する別名を持っています。

クラス別名属性とフィールド別名属性を .NET アプリケーションに追加します。クラス別名を .NET アプリケーション、Java アプリケーション、またはその両方に追加することができます。 .NET サンプルには Java デフォルトと一致する別名があるため、Java アプリケーションは別名を必要としません。 TestKey.cs および TestValue.cs ファイルが [net_client_home/sample/SimpleClient](#) ディレクトリーにあります。

図 1. TestKey.cs ファイル内のクラス別名属性

```
[ClassAlias( "com.ibm.websphere.xs.sample.gettingstarted.model.TestKey" )]
```

図 2. TestValue.cs ファイル内のクラス別名属性

```
[ClassAlias( "com.ibm.websphere.xs.sample.gettingstarted.model.TestValue" )]
```

関連概念:

[ClassAlias および FieldAlias 注釈](#)

関連タスク:

[Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

レッスンのチェックポイント

クラス属性を .NET 入門用アプリケーションに追加しました。その結果、Java 入門用アプリケーションと相互運用してエンタープライズ・データ・グリッドを作成できるようになりました。

[< 前へ](#) | [次へ >](#)

[< 前へ | 次へ >](#)

モジュール 3: データ・グリッド内でのサンプル・アプリケーションの実行

アプライアンス上のデータ・グリッドに対してサンプル・クライアント・アプリケーションを実行することができます。

クライアント・アプリケーションをどのように実行するかは、Java サンプルを実行しているか、または .NET サンプルを実行しているかによって異なります。

学習目標

このモジュールのレッスンを完了すると、以下のタスクの実行方法が理解できるようになります。

- **Java** [Java 入門用サンプル・クライアント・アプリケーションの実行](#)
- **.NET** [.NET サンプル・クライアント・アプリケーションの実行](#)

Java サンプル・アプリケーションと .NET サンプル・アプリケーションを別々に実行するほかに、これらを同じデータ・グリッド上で同時に実行することもできます。例えば、.NET アプリケーションを使用して値をデータ・グリッドに挿入し、次いで Java アプリケーションを使用してその値を取得することができます。このシナリオでは、エンタープライズ・データ・グリッドを実行します。

このモジュールのレッスン

[入門チュートリアル・レッスン 3.1: Java 入門用サンプル・クライアント・アプリケーションの実行](#)

以下のステップを使用して、データ・グリッドと対話する Java クライアントを実行します。

[入門チュートリアル・レッスン 3.2: .NET サンプル・クライアント・アプリケーションの実行](#)

以下のステップを使用して、データ・グリッドと対話する WebSphere eXtreme Scale Client for .NET アプリケーションを実行します。この例では、カタログ・サーバー、コンテナ・サーバー、およびクライアントがすべて単一のサーバー上で実行されます。

[< 前へ | 次へ >](#)

[< 前へ](#) | [次へ >](#)



入門チュートリアル・レッスン 3.1: Java 入門用サンプル・クライアント・アプリケーションの実行

以下のステップを使用して、データ・グリッドと対話する Java クライアントを実行します。

[wxs_install_root/ObjectGrid/gettingstarted/env.bat|sh](#) ファイルを編集します。このファイルはクライアントによって自動的に呼び出されます。ファイルには、以下の情報が含まれています。

```
SET CATALOGSERVER_HOST=<xc10_hostname>
SET CATALOGSERVER_PORT=2809
SET GRID_NAME=my_simple_data_grid
SET MAP_NAME=my_map.P
```

アプライアンス上のデータ・グリッドに接続するには、**CATALOGSERVER_HOST** および **CATALOGSERVER_PORT** 変数を更新する必要があります。指定すべきカタログ・サーバーが、作成した単純データ・グリッドのユーザー・インターフェースのページに表示されます。「データ・グリッド」 > 「単純データ・グリッド」 > 「**my_simple_data_grid**」をクリックし、「カタログ・サービス」フィールドにある値を使用します。**GRID_NAME** プロパティの値は、作成したデータ・グリッドの名前と一致しなければなりません。**MAP_NAME** プロパティは、存続時間 (TTL) 設定およびペシミスティック・ロックのないマップを作成します。動的マップの命名については、[動的マップの構成オプション](#)を参照してください。

- クライアントを対話モードで実行します。コマンド行ウィンドウから、次のいずれかのコマンドを実行します。
 -  `./runclient.sh`
 -  `runclient.bat`
- 1. トランザクションを開始します。トランザクションに対して 1 フェーズ・コミット操作または 2 フェーズ・コミット操作を使用することができます。1 フェーズ・コミットの場合は、トランザクションが単一の区画に書き込まなければなりません。異なる区画に配置される複数のキーをトランザクション時に挿入すると、コミット時にトランザクションが失敗します。2 フェーズ・コミットを使用すれば、単一のトランザクションで複数の区画に書き込まれるようにすることができます。

- 1 フェーズ・コミット・トランザクションを開始します。

```
begin
```

- 2 フェーズ・コミット・トランザクションを開始します。

```
begin2pc
```

2. 値を挿入します。

```
> i key1 helloWorld
SUCCESS: Inserted TestValue [value=helloWorld] with key TestKey [key=key1], partitionId=6
```

3. 挿入した値を取得します。

```
> g key1
Value is TestValue [value=helloWorld], partitionId=6
```

4. 値を更新します。

```
> u key1 goodbyeWorld
SUCCESS: Updated key TestKey [key=key1] with value TestValue [value=goodbyeWorld], partitionId=6
```

5. トランザクションをロールバックします。トランザクションをロールバックすると、このトランザクションに関連するすべての操作がキャンセルされます。

```
> rollback
```

6. ロールバック操作をテストするには、キーの取得を再び試みます。トランザクションをロールバックしたので、キーが存在していません。

```
> g key1
```

7. 値を挿入します。

```
> i key1 helloWorld  
SUCCESS: Inserted TestValue [value=helloWorld] with key TestKey [key=key1], p  
art  
itionId=6
```

8. 値をコミットします。トランザクションをコミットした後から変更をロールバックすることはできません。

```
> commit
```

9. 挿入した値を削除します。

```
> d key1  
SUCCESS: Deleted value with key TestKey [key=key1], partitionId=6
```

10. テスト項目をいくつか挿入します。例えば、0 から 999 までの番号が付けられた 1000 個のキーおよび値を挿入するには、次のコマンドを使用します。

```
> n 1000
```

- クライアントをコマンド行モードで実行します。クライアント・アプリケーションを実行するスクリプトを作成したい場合は、コマンド行モードが役立ちます。対話モードで実行するコマンドと同じコマンドを実行することができます。コマンド行モードの構文の例を以下に示します。

- **UNIX | Linux**

```
./runclient.sh i "key1" "helloWorld"
```

- **Windows**

```
runclient.bat i "key1" "helloWorld"
```

レッスンのチェックポイント

学習した内容

このレッスンでは、以下を学習しました。

- データ・グリッドに対してデータの挿入、取得、更新、および削除を行う Java サンプル・クライアント・アプリケーションを実行する方法。

[< 前へ](#) | [次へ >](#)

入門チュートリアル・レッスン 3.2: .NET サンプル・クライアント・アプリケーションの実行

以下のステップを使用して、データ・グリッドと対話する WebSphere® eXtreme Scale Client for .NET アプリケーションを実行します。この例では、カタログ・サーバー、コンテナ・サーバー、およびクライアントがすべて単一のサーバー上で実行されます。

WebSphere eXtreme Scale Client for .NET は 1 フェーズ・コミットのみをサポートします。したがって、同じトランザクションで複数の値を挿入しようと試みた場合は、それらの値が異なる区画に送られるため例外が発生します。サンプルを実行したときこのような例外が発生しないようにするため、1つの区画を使用するようにデプロイメント・ポリシー記述子 XML ファイルを変更することができます。区画数の更新について詳しくは、[入門チュートリアル・レッスン 1.1: データ・グリッドの定義](#)を参照してください。

サンプル・アプリケーションは対話モードまたはコマンド行モードで実行することができます。対話モードでは、アプリケーションは IGridMapPessimisticTx API を使用して手動データ・グリッド・トランザクションを実行します。コマンド行モードでは、IGridMapPessimisticAutoTx API を使用して自動データ・グリッド・トランザクションを実行します。

サンプルは対話モードまたはコマンド行モードで実行することができます。

- サンプル・クライアント・アプリケーションを対話モードで実行します。
 1. シンプルなクライアント・アプリケーションを実行します。ファイルは [net_client_home](#) %gettingstarted%bin% ディレクトリーにあります。サンプルを対話モードで実行するには、次のコマンドを実行します。

```
SimpleClient.exe -i [-h <hostname:port>] [-g <gridname>] [-m <mapname>]
```

-h <hostname:port>

接続先のカタログ・サーバーのホスト名とポートを指定します。指定すべきカタログ・サーバーが、作成した単純データ・グリッドのユーザー・インターフェースのページに表示されます。「データ・グリッド」 > 「単純データ・グリッド」 > 「**my_simple_data_grid**」をクリックし、「カタログ・サービス」フィールドにある値を使用します。

-g <gridname>

使用するデータ・グリッドの名前を指定します。ペシミスティック・ロック・ストラテジーでマップを使用する必要があります。このチュートリアルでは、my_simple_data_grid、またはアプライアンス・ユーザー・インターフェースでデータ・グリッドを作成したときに指定した名前を使用してください。値を指定しなかった場合は、Grid データ・グリッドが使用されます。

-m <mapname>

使用するマップの名前を指定します。このチュートリアルでは、my_map.NONE.P を指定してください。値を指定しなかった場合は、エラーが発生します。マップ名について詳しくは、[動的マップの構成オプション](#)を参照してください。

パラメーターなしでアプリケーションを実行した場合は、アプリケーションのヘルプが表示されます。

2. 使用可能なコマンドのリストを表示します。

```
Enter a command: help
This program executes simple CRUD operations on a map.
  a - Adds a value with the specified key. If the key already exists,
      DuplicateKeyException is thrown
  p - Adds a value with the specified key, replacing the entry if it
      already exists
  r - Replaces the value of the specified key. If the key does not exist,
      a CacheKeyNotFound exception is thrown
  g - Retrieve and display the value of the specified key
  d - Deletes the key
  gp - Gets the partition id for the key
  ck - Checks if the map contains the key
  h - Display help
begin - Begin manual transaction
commit - Commit transactions
rollback - Rollback transactions
exit - Exit program
```


3. トランザクションを開始します。データ・グリッドでコマンドを実行するためにはトランザクションを開始する必要があります。トランザクションを開始しなかった場合は、NoActiveTransactionException 例外が発生します。

```
Enter a command: begin
```

4. データ・グリッドにデータを追加します。

```
Enter a command: a key1 value1
SUCCESS: Added 'TestKey [key=key1]' with value 'TestValue [value=value1]',
partitionId=6
```

5. 値を検索して表示します。

```
Enter a command: g key1
SUCCESS: Value is 'TestValue [value=value1]', partitionId=6
```

この例では、value1 が戻されます。

6. キーを更新します。put コマンドを使用します。このコマンドは、指定されたキーを持つ値を追加するもので、既存の値が存在する場合はそれを置き換えます。

```
Enter a command: p key1 value2
SUCCESS: Put key 'TestKey [key=key1]' with value 'TestValue [value=value2]',
partitionId=6
Enter a command: g key1
SUCCESS: Value is 'TestValue [value=value2]', partitionId=6
```

7. キーを置き換えます。replace コマンドは、指定されたキーの値を置き換えます。キーが存在しない場合は、CacheKeyException 例外が発生します。

```
Enter a command: r key1 value3
SUCCESS: Replaced key 'TestKey [key=key1]' with value 'TestValue [value=value
3]'
, partitionId=6
```

8. トランザクションをロールバックし、キーと値の再表示を試みます。トランザクションは、コミットする前であればいつでもロールバックすることができます。

```
Enter a command: rollback
Enter a command: begin
Enter a command: g key1
FAILED: Key not found
```

get コマンドを実行すると、キーが見つからなかったことを示すエラーが出されます。

9. キーと値をデータ・グリッドにコミットします。

```
Enter a command: begin
Enter a command: a key2 value2
SUCCESS: Added 'TestKey [key=key2]' with value 'TestValue [value=value2]',
partitionId=7
Enter a command: commit
```

10. キーの区画 ID を取得します。

```
Enter a command: begin
Enter a command: gp key2
SUCCESS: partitionId=7
```

11. キーのマップを確認します。

```
Enter a command: ck key2
SUCCESS: The map contains key 'TestKey [key=key2]'
Enter a command: ck key3
SUCCESS: The map does NOT contain key 'TestKey [key=key3]'
```


12. キーを削除し、終了します。

```
Enter a command: begin
Enter a command: d key2
SUCCESS: Deleted value with key 'TestKey [key=key2]', partitionId=7
Enter a command: commit
Enter a command: exit
```

- クライアントをコマンド行モードで実行します。コマンド行モードでは、IGridMapPessimisticAutoTx API を使用して自動データ・グリッド・トランザクションを実行します。このモードを使用するには、コマンド行でアクションを渡します。クライアント・アプリケーションを実行するスクリプトを作成したい場合は、コマンド行モードが役立ちます。対話モードで実行するコマンドと同じコマンドを実行することができます。コマンド行モードの構文の例を以下に示します。

```
SimpleClient [-h <host:port>] [-g <grid_name>] [-m <mapname>] <a | p | r | g | d>
<key> [<value>]
```

関連タスク:

[.NET .NET API を使用したデータ・グリッド・アプリケーションの開発](#)
[WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)

レッスンのチェックポイント

このレッスンでは、以下を学習しました。

- データ・グリッドに対してオブジェクトの挿入、取得、更新、および削除を行う .NET サンプル・クライアント・アプリケーションを実行する方法。

[< 前へ | 次へ >](#)

[< 前へ](#)

入門チュートリアル・レッスン 4: 環境のモニター

xscmd ユーティリティーおよび Web コンソールのツールを使用して、データ・グリッド環境をモニターできます。

関連タスク:

[ユーザー・インターフェースでのデータ・グリッドのモニター](#)

ユーザー・インターフェースでのデータ・グリッドのモニター

ユーザー・インターフェースで、環境内のデータ・グリッドの全体的なパフォーマンスを表示することができます。ユーザー・インターフェースのモニター・セクションには、アプライアンス上のすべてのデータ・グリッドの概要、個々のデータ・グリッドの概要、および個々のデータ・グリッドの詳細レポートがあります。

ユーザー・インターフェースでのデータ・グリッドのモニターについて詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

xscmd ユーティリティーによるモニター

1. オプション: クライアント認証が使用可能な場合: コマンド行ウィンドウをクライアントのインストール済み環境で開きます。コマンド行で、適切な環境変数を設定します。
2. [wxs_home](#)/bin ディレクトリーに移動します。

```
cd wxs\_home/bin
```

3. 各種コマンドを実行して、環境に関する情報を表示します。
 - Grid データ・グリッドと mapSet マップ・セットのすべてのオンライン・コンテナー・サーバーを表示します。

```
xscmd -c showPlacement -g Grid -ms mapSet
```

- データ・グリッドのルーティング情報を表示します。

```
xscmd -c routetable -g Grid
```

- データ・グリッド内のマップ・エントリーの数を表示します。

```
xscmd -c showMapSizes -g Grid -ms mapSet
```

レッスンのチェックポイント

このレッスンでは、以下を学習しました。

- データ・グリッドおよびサーバーの統計をモニターする方法。

[< 前へ](#)

DataPower XC10 アプライアンス環境の計画

DataPower® XC10 アプライアンスをトポロジーに組み込む前に、環境が次の前提条件およびソフトウェア要件を満たしている必要があります。

システム要件

ご使用の環境に WebSphere® DataPower XC10 アプライアンスを組み込むには、以下の要件がインストールされている必要があります。

ディレクトリー規則

`wxs_install_root` や `wxs_home` など、参照が必要な特別のディレクトリーに対して、資料全体で、次のディレクトリー規則が使用されます。インストール中、およびコマンド行ツールの使用時も含めて、さまざまなシナリオで、これらのディレクトリーにアクセスします。

ネットワーク・ポート

ファイアウォール内で WebSphere DataPower XC10 アプライアンスを使用している場合、次のポートを経由した通信を使用可能に設定する必要があります。

IBM WebSphere DataPower XC10 アプライアンス インストールの前提条件

IBM® WebSphere DataPower XC10 アプライアンスをインストールおよび構成するには、特定のハードウェア、ソフトウェア、ラック、およびツールの要件を満たす必要があります。この前提条件一式を使用して、IBM WebSphere DataPower XC10 アプライアンスのインストール、構成、および使用の計画を立てます。

アプライアンスの仕様およびフィーチャー

WebSphere DataPower XC10 アプライアンス バージョン 2.5 は Type 7199-92x ハードウェア上でサポートされます。ハードウェア Type 9235-92x にはもともとバージョン 1.0 が付属しており、このハードウェアではバージョン 2.1 までしかサポートされません。

製品のインターオペラビリティ

WebSphere DataPower XC10 アプライアンスは、他の IBM 製品とのインターオペラビリティについてテスト済みです。

.NET 2.5+ Microsoft .NET に関する考慮事項

WebSphere eXtreme Scale には 2 つの .NET 環境 (開発環境とランタイム環境) が存在します。これらの環境はそれぞれ固有の要件を持っています。

システム要件

ご使用の環境に WebSphere® DataPower® XC10 アプライアンス を組み込むには、以下の要件がインストールされている必要があります。

ハードウェア要件

初期構成を実行するには、シリアル接続を使用する必要があります。ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC から、 WebSphere DataPower XC10 アプライアンス アプライアンスのシリアル・ポートに対するシリアル接続が必要です。シリアル接続を行う場合は、付属の RJ45 から DB-9 のシリアル・ヌル・モデム・ケーブルを使用してください。

注: ターミナルにシリアル・ポートが装備されていない場合は、付属の USB シリアル変換ケーブルを使用してください。

無人インストールまたはリモート・インストールは、ネットワーク接続された端末サーバー経由で行うことができます。

ソフトウェア要件

WebSphere DataPower XC10 アプライアンスは、IBM 提供のソフトウェアに付属しています。アプライアンスに他のソフトウェアをインストールすることはできません。

Web ブラウザー要件

ユーザー・インターフェースでは、以下の Web ブラウザーがサポートされます。

- Mozilla Firefox、バージョン 3.5.x 以降
- Microsoft Internet Explorer バージョン 7 以降

サポートされるソフトウェア

次のソフトウェアは、WebSphere DataPower XC10 アプライアンスと連携して使用できます。

- WebSphere eXtreme Scale クライアント 8.5
- WebSphere eXtreme Scale クライアント バージョン 8.6
- WebSphere Application Server バージョン 6.1.0.41 以降
- WebSphere Application Server バージョン 7.0.0.21 以降
- WebSphere Application Server バージョン 8.0.0.2 以降
- WebSphere Application Server バージョン 8.5.0.0 以降
- WebSphere Application Server バージョン 8.5.5.0

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

ディレクトリー規則

`wxs_install_root` や `wxs_home` など、参照が必要な特別のディレクトリーに対して、資料全体で、次のディレクトリー規則が使用されます。インストール中、およびコマンド行ツールの使用時も含めて、さまざまなシナリオで、これらのディレクトリーにアクセスします。

`wxs_install_root`

`wxs_install_root` ディレクトリーは、WebSphere® eXtreme Scale 製品ファイルがインストールされているルート・ディレクトリーです。`wxs_install_root` ディレクトリーは、試用版のアーカイブが解凍されたディレクトリー、または WebSphere eXtreme Scale 製品がインストールされているディレクトリーの可能性があります。

- 試用版を解凍した場合の例:

例: `/opt/IBM/WebSphere/eXtremeScale`

- WebSphere eXtreme Scale がスタンドアロン・ディレクトリーにインストールされている場合の例:

UNIX 例: `/opt/IBM/eXtremeScale`

Windows 例: `C:\Program Files\IBM\WebSphere\eXtremeScale`

- WebSphere eXtreme Scale が WebSphere Application Server に統合されている場合の例:

例: `/opt/IBM/WebSphere/AppServer`

`wxs_home`

`wxs_home` ディレクトリーは、WebSphere eXtreme Scale 製品ライブラリー、サンプル、およびコンポーネントのルート・ディレクトリーです。このディレクトリーは、試用版を解凍した場合は、`wxs_install_root` ディレクトリーと同じです。スタンドアロンのインストール済み環境の場合、`wxs_home` ディレクトリーは、`wxs_install_root` ディレクトリー内の `ObjectGrid` サブディレクトリーです。WebSphere Application Server に統合されているインストール済み環境の場合、このディレクトリーは、`wxs_install_root` ディレクトリー内の `optionalLibraries/ObjectGrid` ディレクトリーです。

- 試用版を解凍した場合の例:

例: `/opt/IBM/WebSphere/eXtremeScale`

- WebSphere eXtreme Scale がスタンドアロン・ディレクトリーにインストールされている場合の例:

UNIX 例: `/opt/IBM/eXtremeScale/ObjectGrid`

Windows 例: `wxs_install_root\ObjectGrid`

- WebSphere eXtreme Scale が WebSphere Application Server に統合されている場合の例:

例: `/opt/IBM/WebSphere/AppServer/optionalLibraries/ObjectGrid`

`was_root`

`was_root` ディレクトリーは、WebSphere Application Server インストール済み環境のルート・ディレクトリーです。

例: `/opt/IBM/WebSphere/AppServer`

.NET `net_client_home`

`net_client_home` ディレクトリーは、.NET クライアント・インストール済み環境のルート・ディレクトリーです。

例: `C:\Program Files\IBM\WebSphere\eXtreme Scale .NET Client`

`java_home`

`java_home` は、Java™ Runtime Environment (JRE) インストール済み環境のルート・ディレクトリーです。

UNIX 例: `/opt/IBM/WebSphere/eXtremeScale/java`

Windows 例: `wxs_install_root\java`

`samples_home`

`samples_home` は、チュートリアルに使用するサンプル・ファイルを解凍したディレクトリーです。

UNIX 例: `wxs_home/samples`

Windows 例: `wxs_home\samples`

dvd_root

dvd_root ディレクトリーは、製品が含まれた DVD のルート・ディレクトリーです。

例: *dvd_root/docs/*

user_home

user_home ディレクトリーは、ユーザー・ファイル (セキュリティー・プロファイルなど) が保管されている場所です。

Windows c:¥Documents and Settings¥*user_name*

UNIX /home/*user_name*

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

ネットワーク・ポート

ファイアウォール内で WebSphere® DataPower® XC10 アプライアンスを使用している場合、次のポートを経由した通信を使用可能に設定する必要があります。

カタログ・サーバー

下記のポートはカタログ・サーバーによって使用されるものです。集合に追加される最初の 3 つのアプライアンスは、それぞれ 1 つのカタログ・サーバーを実行します。

ピア・ポート: カタログ・サーバー間の通信に使用されます。このポートは 6601 を使用するように設定されます。

クライアント・ポート: カタログ・サーバー間の通信に使用されます。このポートは 6602 を使用するように設定されます。

JMX サービス・ポート: 非 SSL JMX 接続のために使用されます (xscmd ユーティリティーなど)。このポートは 1099 を使用するように設定されます。

JMX コネクター・ポート: SSL JMX 接続のために使用されます (xscmd ユーティリティーなど)。このポートは 1100 を使用するように設定され、7100 から 7116 までの間のポートを選択します。

ORB リスナー・ポート: クライアントとデータ・グリッド・サーバー間の通信に使用されます。このポートは 2809 を使用するように設定されます。

CSiv2 リスナー・ポート: クライアントとデータ・グリッド・サーバー間のセキュア通信に使用されます。このポートは 7499 を使用するように設定されます。

SNMP ポート: SNMP モニターのために使用されます。このポートは 161 を使用するように設定されます。

注: WebSphere DataPower XC10 アプライアンスは、SNMP バージョン v2c をサポートします。

以下のポートは、コンテナ・サーバー間の通信に必要なものです。

DCS コア・グループ・ポート: 内部 WebSphere DataPower XC10 アプライアンス DCS 通信に使用されます。このポートは、6700 から 6716 の範囲内のポートを使用します。

ORB リスナー・ポート: クライアントとデータ・グリッド・サーバー間の通信に使用されます。このポートは、6800 から 6816 の範囲内のポートを使用します。

CSiv2 リスナー・ポート: クライアントとデータ・グリッド・サーバー間のセキュア通信に使用されます。このポートは、7500 から 7516 の範囲内のポートを使用します。

注: アプライアンスが集合内にあり、かつさまざまなファイアウォール側にある場合は、双方向通信のために、コンテナ・ポート DCS、ORB、および CSiv2 を、カタログ・ポート 6601 および 6602 と共に開く必要があります。

ブラウザー・ポート

アプライアンス・コンソールまたは REST サービスを使用する場合は、ポート 80 および 443 を開きます。集合内のすべてのアプライアンスでこれらのポートが開いていなければ、一部の構成オプションは失敗する可能性があります。

クライアント・サーバー

WebSphere eXtreme Scale クライアントは、リスナー・ポートを使用します。このポートは、`wxs_client_root` ¥properties ディレクトリー内のクライアント・プロパティー・ファイルに指定されています。

問題判別アプリケーション

問題判別アプリケーションにポート 9060 を開きます。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

IBM WebSphere DataPower XC10 アプライアンス インストールの前提条件

IBM® WebSphere® DataPower® XC10 アプライアンスをインストールおよび構成するには、特定のハードウェア、ソフトウェア、ラック、およびツールの要件を満たす必要があります。この前提条件一式を使用して、IBM WebSphere DataPower XC10 アプライアンス のインストール、構成、および使用の計画を立てます。

必要なスキル

アプライアンスをセットアップおよび管理するには、ネットワーク管理のスキルが必要です。

必要な情報

DataPower XC10 アプライアンス の基本構成を定義するためには、以下のデータを収集する必要があります。

- データ・グリッドに 1 ギガビット・イーサネット・ポートを使用しているか、10 ギガビット・イーサネット・ポートを使用しているか。使用するポートはすべて 1 ギガビット・ポート、またはすべて 10 ギガビット・ポートでなければなりません。MGMT0 の管理ポートに接続します。
- アプライアンス管理アクセス用のイーサネット・インターフェース (MGMT) の IP アドレスおよびサブネット・マスク。
- イーサネット・インターフェースのサブネットをサポートするデフォルト・ゲートウェイ (ルーター) の IP アドレス。
- ドメイン・ネーム・システム (DNS) サーバーの IP アドレス。フォワード・ルックアップとリバース・ルックアップの両方が可能となるように、DNS サーバーを構成してください。
- シリアル・インターフェースの通信パラメーター: 9600.8.n.1 (9600 ボー、8 データ・ビット、パリティなし、1 ストップ・ビット)。
- メール・サーバー情報 (E メール通知のセットアップ用)。
- (オプション) アプライアンス・サービス・アクセス用のイーサネット・インターフェース (ETH0、ETH1、および ETH2) の IP アドレスおよびサブネット・マスク。

ラック要件

DataPower XC10 アプライアンスをインストールするには、以下の基準を満たす、最小で 25 インチ (63.5 cm) の奥行きがある標準の 19 インチ (48.26 cm) ラックが必要です。

- 背面に取り付け用のコラムがある
- 正面取り付けと背面取り付けの両方がサポートされている
- 背面に取り付け用のコラムがある
- 正面取り付けと背面取り付けの両方がサポートされている

ラックの内部と周囲に以下の必要なスペースが確保されている状態にしてください。

- ラックの背後に 30 インチ (76.20 cm) 以上の空き領域がある
- アプライアンスの上下が 2 インチ (5.1 cm) 以上空いている
- イーサネット・ケーブルおよびシリアル・コンソール・ケーブル用に正面に十分な空きスペースがある

アプライアンスを取り付ける環境内の気温は、104°F (40°C) を超えてはいけません。

ツールおよびサプライ

DataPower XC10 アプライアンス をインストールするために、以下のサプライを揃えておいてください。

- 中型のプラス・ドライバー
- 2 本の標準ラックねじ (アプライアンスに同梱)
- 1 本から 4 本のネットワーク・ケーブル
- シリアル・ケーブルまたは PL-2303 USB シリアル・ケーブル

重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。

注: アプライアンスのインストール後も、ケーブルを 廃棄しないでください。今後、問題判別や保守に必要となる場合があります。

- オス DB9 コネクター付きシリアル・コンソール。このコンソールには、VT100 スタイルのコンソールなどの専用ハードウェア・デバイス、またはハイパーターミナルや Minicom などのエミュレーターが稼働する PC が考えられます。

ユーザー・インターフェースの要件

Web ユーザー・インターフェースを使用するには、以下のブラウザーのいずれかが使用できます。

- Mozilla Firefox、バージョン 3.5.x 以降
- Microsoft Internet Explorer バージョン 7 以降

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

アプライアンスの仕様およびフィーチャー

WebSphere® DataPower® XC10 アプライアンス バージョン 2.5 は Type 7199-92x ハードウェア上でサポートされます。ハードウェア Type 9235-92x にはもともとバージョン 1.0 が付属しており、このハードウェアではバージョン 2.1 までしかサポートされません。

アプライアンスのタイプを判別するには、「アプライアンス」 > 「設定」 > 「ファームウェア」とクリックします。このパネルには、アプライアンス・モデル・タイプとアプライアンスのシリアル番号が表示できます。

[アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

仕様とフィーチャーを参考に、アプライアンスを設置するために必要な物理的環境を 決定してください。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

アプライアンス Type 7199-92x の仕様およびフィーチャー

仕様とフィーチャーを参考に、アプライアンスを設置するために必要な物理的環境を決定してください。

Type 7199-92X アプライアンスについて

Type 7199-92X WebSphere® DataPower® XC10 アプライアンスには、WebSphere DataPower XC10 アプライアンスバージョン 2.0 ソフトウェアが付属しています。この新しいハードウェア・リリースでは、以前のリリースと比べて、より高速なプロセッサが搭載され、ネットワーク・ポート数とキャッシュ容量が増えました。

アプライアンスのタイプを判別するには、「アプライアンス」>「設定」>「ファームウェア」とクリックします。このパネルには、アプライアンス・モデル・タイプとアプライアンスのシリアル番号が表示できます。

仕様

表 1. Type 7199-92x アプライアンスの仕様. Type 7199 シャーシの仕様を要約します。

サイズ/重量:	
	7199
高さ	8.89 cm (3.5 インチ)
幅	42.8 cm (17.25 インチ)
奥行き	58.4 cm (23 インチ)
重量	最大: 21 kg (46.2 ポンド)
電気入力:	
正弦波	50 - 60 Hz (単相) が必要
110 ボルト AC	最小: 100 V _{RMS} 最大: 127 V _{RMS}
220 ボルト AC	最小: 200 V _{RMS} 最大: 240 V _{RMS}
電力使用	10 A (110 V AC) 5 A (220 V AC) Type 7199 アプライアンスには、720 ワット 電源モジュールが 2 つあります。両方の電源機構モジュールは同じ給電部に接続し、2 つの電源モジュール間で接地電圧に差が生じないようにしなければなりません。
環境	
気温	電源オン時: <ul style="list-style-type: none">高度 0 m から 914.4 m (3000 フィート) 50°F から 95°F (10°C から 35°C)高度: 914.4 m (3000 フィート) から 2133.6 m (7000 フィート) 50°F から 89.6°F (10°C から 32°C) 最大高度: 2133.6 m (7000 フィート) 電源オフ時: 50°F から 109.4°F (10°C から 43°C) 出荷時: -40°F から 140°F (-40°C から 60°C)
湿度	8% から 80%

フィーチャー

表 2. データ・ストレージ・オプション

フィーチャー	説明
ローカル側容量	ローカル・ファイル・システム上に 16 GB のストレージ
ハード・ディスク・アレイ	2 基の 1 TB シンプル・スワップ・シリアル接続 SCSI (SAS) ハード・ディスク 容量: 2 TB
データ・グリッド	320 GB Fusion DUO Drive (240 GB キャッシュ搭載)

[Type 7199-92x 正面図](#)

正面図では、Type 7199 アプライアンスのコントロール、LED、およびコネクタを表示します。イーサネット・モジュールおよびハード・ディスク・モジュールは、Type 7199-92x アプライアンスのフロント・パネルから取り付けることができます。

[Type 7199-92x 背面図](#)

背面図では、アプライアンスの背面にある構成装置と LED を表示します。ファン・モジュールおよび電源モジュ

ールは、アプライアンスの背面に 取り付けられます。

イーサネット・ネットワーク構成

イーサネット・モジュールは、ネットワーク接続オプションを拡張します。アプライアンスには、それぞれ 2 つのイーサネット・モジュールがあります。イーサネット・モジュールは左から右に 番号が振られます。ただし、モジュールのポート数が 8 個より少ない場合、そのモジュールは 範囲内で一番小さいポート番号を使用します。

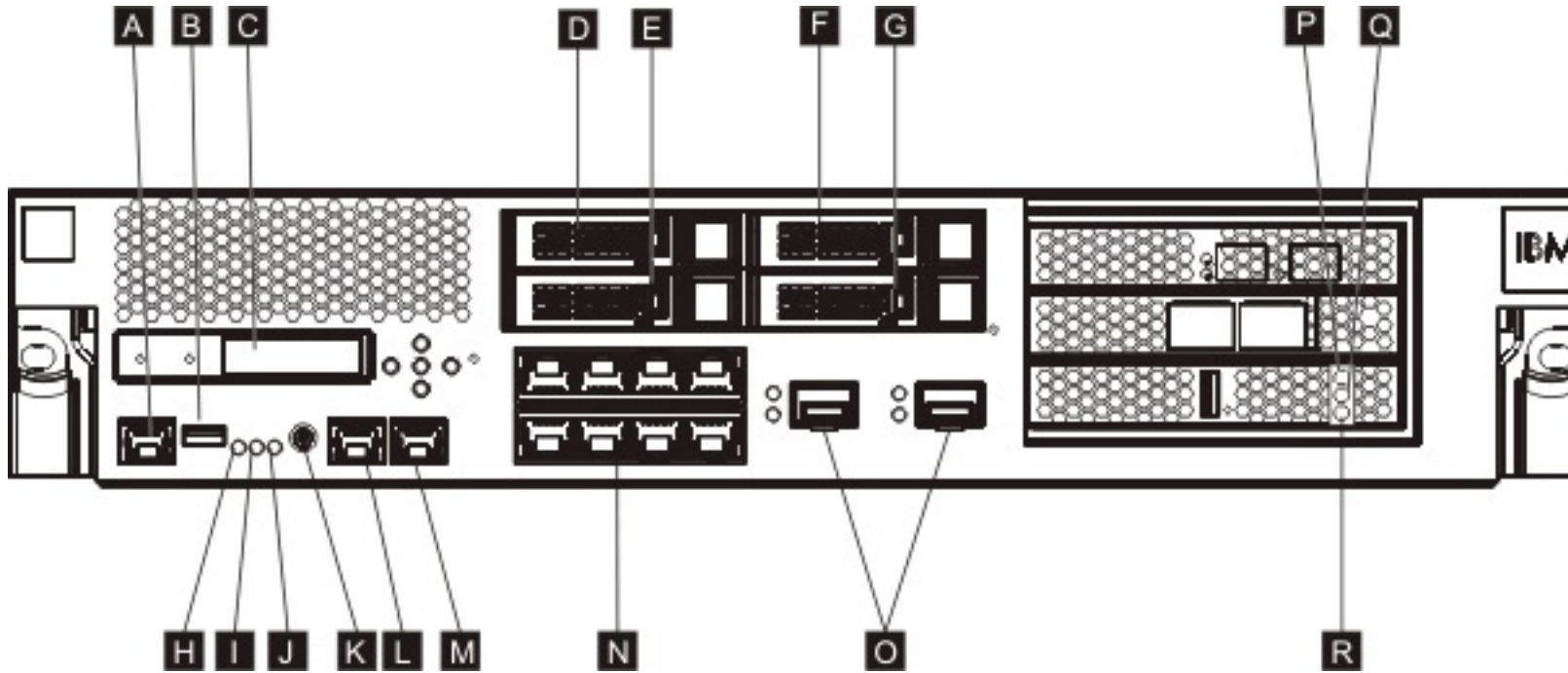
親トピック: [アプライアンスの仕様およびフィーチャー](#)

Type 7199-92x 正面図

正面図では、Type 7199 アプライアンスのコントロール、LED、およびコネクタを表示します。イーサネット・モジュールおよびハード・ディスク・モジュールは、Type 7199-92x アプライアンスのフロント・パネルから取り付けることができます。

正面図

図 1. Type 7199-92x 正面図



上図のラベルは、Type 7199-92x アプライアンスのフロント・パネルにある以下の構成装置に対応しています。

- A** コンソール・コネクタ
- B** USB ポート
- C** LCM モジュール
- D** ハード・ディスク・モジュール 2
- E** ハード・ディスク・モジュール 0
- F** ハード・ディスク・モジュール 3
- G** ハード・ディスク・モジュール 1
- H** 障害 LED
- I** 場所の特定 LED
- J** 電源 LED
- K** 電源ボタン
- L** MGT0 イーサネット・コネクタ
- M** MGT1 イーサネット・コネクタ
- N** 左方イーサネット・モジュール:
 - eth0
 - eth1
 - eth2

- eth3
- eth4
- eth5
- eth6
- eth7

O

右方イーサネット・モジュール:

- eth8
- eth9

P

琥珀色。キャッシュのエラーを示す LED 表示ライト。

Q

黄色。キャッシュの書き込みを示す LED 表示ライト。

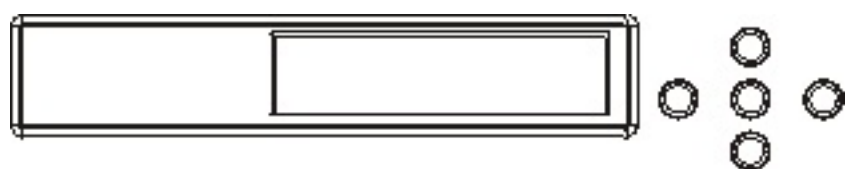
R

緑色。キャッシュの読み取りを示す LED 表示ライト。

LCD モジュール

アプライアンスのフロント・パネルには、液晶ディスプレイ (LCD) と 5 つのメニュー・ボタンからなる LCD モジュールがあります。LCD はアプライアンスのモデル・タイプとビルド番号に関する情報を示しますが、メニュー・ボタンは機能しません。

図 2. LCD モジュール



コンソール・コネクタ

すべての Type 7199 アプライアンスのフロント・パネルにコンソール・コネクタがあります。初期構成では、アプライアンスに付属の RJ45 (ISO 8877) - DB-9 (DE-9 または EIA/TIA-562 と呼ばれます) シリアル・ヌル・モデム・ケーブルを使用して、ASCII 端末 ¹ から、または 端末エミュレーション・ソフトウェアを実行している PC からアプライアンスに接続します。ケーブルの一方の端には RJ45 接続があり、ケーブルのもう一方の端には DB-9 シリアル・ヌル・モデム接続があります。RJ45 側の終端はアプライアンスに接続し、DB-9 シリアル・ヌル・モデム・ケーブル側の終端は ASCII 端末またはパーソナル・コンピュータに接続します。ケーブルをパーソナル・コンピュータに接続する場合は、USB シリアル・コンバーター・ケーブルを使用してください。

注: 初期構成の場合、アプライアンスに付属の RJ45 シリアル接続ケーブルを使用するか、下の表にリストするケーブル・ピン配列仕様に基づいてケーブルを作成することができます。シリアル・コンソール・ポートをイーサネット・ネットワークに接続するとき、イーサネット・ケーブルは使用しないでください。

表 1. シリアル・ポートのピン配列. コンソール・コネクタのシリアル・ポートのピン配列を記載します。

RJ45		DB9	
ピン番号	シグナル	ピン番号	シグナル
1	RTS	8	CTS
2	DTR	6	DSR
3	TXD	2	RXD
4	GND	5	GND
5	GND	5	GND
6	RXD	3	TXD
7	DSR	4	DTR
8	CTS	7	RTS

USB ポート

アプライアンスのフロント・パネルには、USB 2.0 デバイスに規格適合した USB インターフェースがあります。この USB コネクタは使用可能に設定されていないため、どのような接続も提供しません。

LED

すべての Type 7199 アプライアンスのフロント・パネルに、3 つのスタンドアロン LED があります。

障害 LED

クリティカル・イベントが検出されると、オレンジ色の障害 LED が点灯します。

場所の特定 LED

場所の特定 LED は青色で、ファームウェアによってアクティブにされると点灯します。この LED を点灯するかどうかは、コマンド行から制御できます。この LED は非アクティブにされるまで点灯し続けます。コマンド行インターフェースで次の **locate-led** コマンドを使用します。

- アクティブにする場合は、次のコマンドを入力します。

```
locate-led on
```

- 非アクティブにする場合は、次のコマンドを入力します。

```
locate-led off
```

電源 LED

緑色の電源 LED は、アプライアンスを給電部に接続し、アプライアンスの電源を入れると点灯します。

- 緑色の電源 LED は、アプライアンスの電源が入っていて、完全に機能しているときに点灯します。
- この LED が点灯していない場合、アプライアンスは電源が切られています。

電源ボタン

電源ボタンは、アプライアンスのフロント・パネルにあります。次の場合、電源ボタンを押します。

- アプライアンスの電源をオンにします。
- 安全なシャットダウンを開始する (既にアプライアンスの電源が入っている場合)。

電源ボタンを 5 秒間押し続けるとハードウェアの即時シャットダウンが実行されます。

注: 電源ボタンを押してアプライアンスの電源を切っても、デバイスには引き続き電流が供給されます。アプライアンスへの全電力を完全に停止するには、すべての電源コードを抜いてください。

ネットワーク・コネクタ

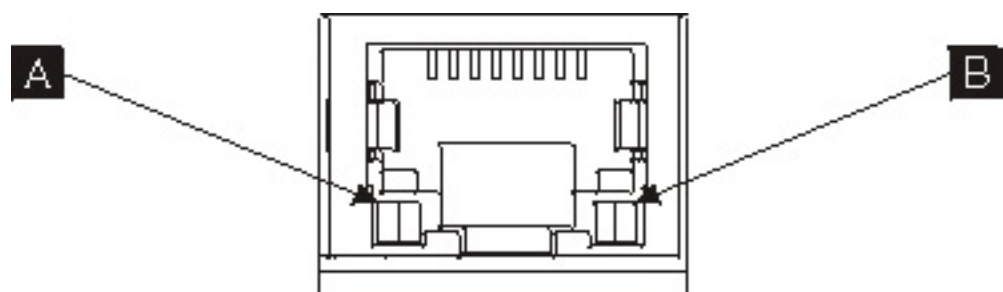
各アプライアンスのフロント・パネルには、2 つの LAN 管理イーサネット・ポートと 2 つのイーサネット・モジュールがあります。イーサネットの命名規則に関する説明については、[イーサネット・ネットワーク構成](#)を参照してください。

LAN 管理イーサネット・ポート

2 つのシステム管理イーサネット・ポートが LAN への接続を提供します。これらのポートは、デバイスへのリモート管理アクセスを可能にするものであり、データ・ポートとしては使用できません。残りのイーサネット・インターフェースは、各種 DataPower® サービスとの間のデータ・トラフィックやロギング機能进行处理できます。

ベスト・プラクティス: 着信 SNMP、SSH のネットワーク・トラフィックを処理するシステム全体の管理機能、およびイントラネット上でのユーザー・インターフェース機能には、MGT0 または MGT1 イーサネット・インターフェースを使用してください。

図 3. イーサネット・ポート LED



MGT0 イーサネット・コネクタ

このイーサネット・インターフェースで、アプライアンス上のすべてのトランザクション・データを管理できます。MGT0 イーサネット・コネクタは、IPMI over LAN (Serial over LAN を含む) もサポートします。MGT0 には関連の速度 LED とアクティビティ LED があります。

速度 LED (A)

- 緑色の LED は 1 Gbps 接続を示します。
- オレンジ色の LED は 10 Mbps または 100 Mbps 接続を示します。

アクティビティ LED (B)

- 緑色の LED は、ポートがリンクされていることを示します。
- 明滅する緑色の LED は、ポートがアクティブなことを示します。

MGT1 イーサネット・コネクタ

このイーサネット・インターフェースで、アプライアンス上のすべてのトランザクション・データを管理できま

す。MGT1 には関連のアクティビティ LED ライトと 速度 LED ライトがあります。

速度 LED (A)

- 緑色の LED は 1 Gbps 接続を示します。
- オレンジ色の LED は 10 Mbps または 100 Mbps 接続を示します。

アクティビティ LED (B)

- 緑色の LED は、ポートがリンクされていることを示します。
- 明滅する緑色の LED は、ポートがアクティブなことを示します。

イーサネット・モジュール

DataPower アプライアンスには、イーサネット接続用の 2 つのイーサネット・モジュールがあります。左方イーサネット・モジュールには 8 つの RJ45 ポートがあり、右方イーサネット・モジュールには 2 つの 10 ギガバイト Small Form Factor Pluggable (SFP+) ポートがあります。イーサネット・インターフェースの名前は、モジュール構成に依存します。

1 ギガビット・モジュールは、シールドなしの対より線を使用するイーサネットをサポートし、インターフェース規格には以下が含まれます。

- 10BASE-T
- 100BASE-TX
- 1000BASE-T

10 ギガビット・モジュールは、インターフェース・モジュールとパッチ・ケーブルを使用する Small Form Factor Pluggable (SFP+) ポートをサポートします。オートネゴシエーションは常に使用可能です。

10GBASE-SR
10GBASE-LR

左方イーサネット・モジュール

シールドなしの対より線 (RJ45) イーサネット・ポートが 8 つあります。イーサネット番号は ETH0 から ETH7 までで、それぞれ使用可能なポートの番号に関連付けられます。

右方イーサネット・モジュール

10 ギガビットの SFP ポートが 2 つあります。イーサネット番号は ETH8 から ETH9 までで、それぞれ使用可能なポートの番号に関連付けられます。

イーサネットの番号付けの説明については、[イーサネット・ネットワーク構成](#)を参照してください。

注: イーサネット・モジュールはホット・スワップ可能ではありません。このモジュールをホット・スワップすると、システムが破壊され、アプライアンスにも損傷を与える恐れがあります。

ハード・ディスク・モジュール

アプライアンスのフロント・パネルには、2.5 インチ・ハード・ディスク・モジュールが 4 つあります。アプライアンスは SAS ハード・ディスクをサポートし、各ハード・ディスク・モジュールに 2 つの LED があります。左側の LED はハード・ディスク・アクティビティをモニターし、右側の LED は潜在的な問題を示します。

- 明滅する緑色の LED は、ハード・ディスクがアクセス中であることを示します。
- 明滅するオレンジ色の LED は、ハード・ディスクに障害があったことを示します。
- LED が点灯していない場合は、ハード・ディスクがアクティブでないことを示します。

注: ハード・ディスク・モジュールはホット・スワップ可能ではありません。このモジュールをホット・スワップすると、システムが破壊される恐れがあります。

親トピック: [アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

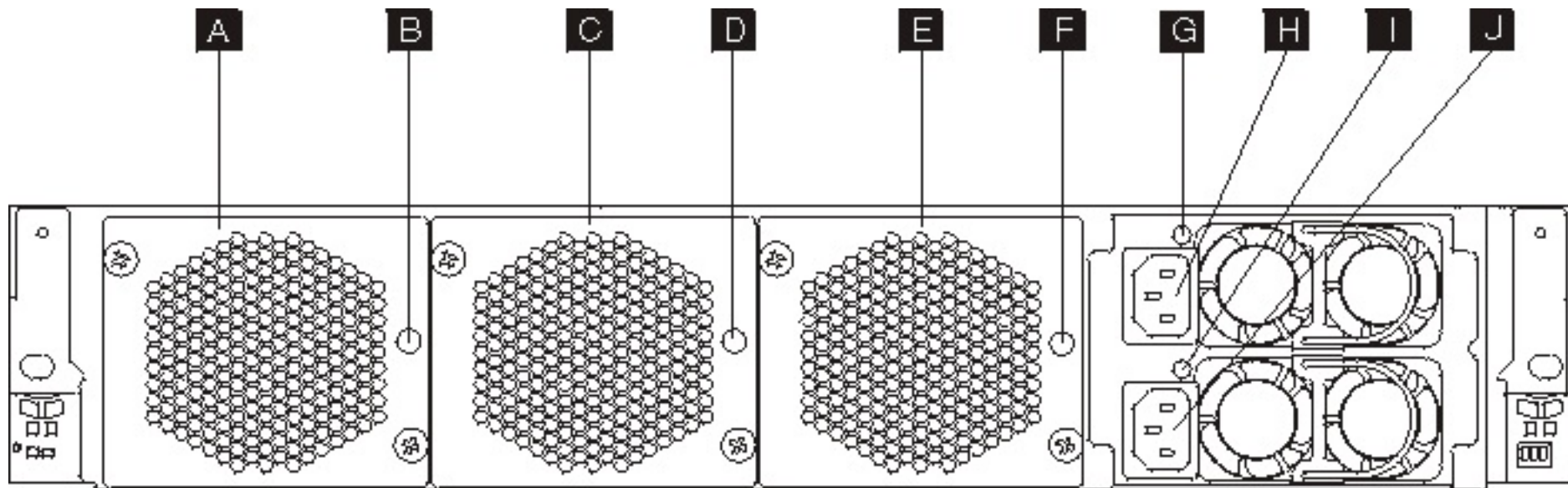
¹ ASCII データを送信 (入力) および受信 (出力) するシンプルなデバイス。

Type 7199-92x 背面図

背面図では、アプライアンスの背面にある構成装置と LED を表示します。ファン・モジュールおよび電源モジュールは、アプライアンスの背面に取り付けられます。

背面図

図 1. Type 7199-92x 背面図



上図のラベルは、Type 7199-92x アプライアンスの背面パネルにある以下の構成装置に対応しています。

- A**
ファン・モジュール 1
- B**
ファン・モジュール 1 LED
- C**
ファン・モジュール 2
- D**
ファン・モジュール 2 LED
- E**
ファン・モジュール 3
- F**
ファン・モジュール 3 LED
- G**
電源機構モジュール 1 LED
- H**
電源機構モジュール 1
- I**
電源機構モジュール 2 LED
- J**
電源機構モジュール 2

ファン・モジュール

アプライアンスには 3 つのファン・モジュールがあります。各ファン・モジュールには個別の冷却ファンがあり、ファン・モジュールごとに LED も付いています。

- オレンジ色の LED が点灯した場合は、ファン・モジュールに問題があります。
- オレンジ色の LED が点灯していない場合、ファンは正常に作動しています。

ファンの速さは、アプライアンスの温度によって変わります。温度が上がると、ファンの速さを増して、アプライアンスの温度バランスを保つようにします。

電源機構モジュール

アプライアンスには、予備電源を含む 2 つの電源機構モジュールにより電力が供給されます。1 つの電源機構モジュール

ルで、アプライアンスの動作をサポートするために十分な電力を供給可能です。電源機構モジュールはホット・スワップ可能であるため、アプライアンスを電源遮断せずに、一方の電源機構モジュールを交換できます。電源機構モジュールにはそれぞれ LED が付いています。

- オレンジ色の電源機構 LED が点灯した場合は、電源機構にエラーがあります。
- LED が点灯していない場合、電源機構は想定どおり作動しています。

注: 電源ボタンを押してアプライアンスの電源を切っても、デバイスには引き続き電流が供給されます。アプライアンスへの全電力を完全に停止するには、すべての電源コードを抜いてください。

親トピック: [アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

イーサネット・ネットワーク構成

イーサネット・モジュールは、ネットワーク接続オプションを拡張します。アプライアンスには、それぞれ 2 つのイーサネット・モジュールがあります。イーサネット・モジュールは左から右に番号が振られます。ただし、モジュールのポート数が 8 個より少ない場合、そのモジュールは範囲内で一番小さいポート番号を使用します。

番号付け規則

イーサネット・インターフェースを構成し、ネットワーク・ケーブルを取り付ける場合の番号付け規則は、次のとおりです。

- 左方モジュールの範囲は ETH0 から ETH7。
- 右方モジュールの範囲は ETH8 から ETH9。

Type 7199 接続

各イーサネット・モジュールは、以下の構成のいずれかを備えています。

- 左方イーサネット・モジュールには、1 ギガビット・イーサネット・ポートが 8 つあり、それらは RJ45 コネクターです。
- 右方イーサネット・モジュールには、10 ギガビット・イーサネット・ポートが 2 つあり、それらは Small Form Factor Pluggable (SFP+) トランシーバーです。

アプライアンスには 10 のイーサネット接続が装備されています。イーサネット・インターフェースの名前は、ETH0 から ETH7、ETH8、および ETH9 です。

図 1. 8x2 イーサネット接続



親トピック: [アプライアンス Type 7199-92x の仕様およびフィーチャー](#)

製品のインターオペラビリティ

WebSphere® DataPower® XC10 アプライアンスは、他の IBM® 製品との インターオペラビリティについてテスト済みです。

WebSphere Application Server

WebSphere Application Server は、HTTP セッションおよび 動的キャッシュ・データ・グリッド・シナリオを実装するために必須です。具体的に必要な WebSphere Application Server の リリースの詳細については、[システム要件](#)を参照してください。

WebSphere DataPower Integration アプライアンス XI50

WebSphere DataPower XC10 アプライアンスは、WebSphere DataPower Integration アプライアンス XI50 の サイド・キャッシュとして使用できます。REST ゲートウェイにより、非 Java ベースのクライアントが、HTTP ベースの一連の操作を使用して単純データ・グリッドにアクセスできるようになり、この統合が可能になります。REST ゲートウェイの詳細については、[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。WebSphere DataPower XC10 アプライアンスと WebSphere DataPower Integration アプライアンス XI50 間の統合を示すサンプル・アプリケーションの情報については、「[Using the WebSphere DataPower XC10 Appliance as a side cache for the WebSphere DataPower XI50 Integration Appliance \(WebSphere DataPower XI50 統合アプライアンスのサイド・キャッシュとしての、WebSphere DataPower XC10 の使用\)](#)」を参照してください。

WebSphere Portal

WebSphere Portal の HTTP セッションをアプライアンスのデータ・グリッドに保持できます。この構成の作成について詳しくは、[WebSphere Portal での HTTP セッション・マネージャーの構成](#)を参照してください。さらに、IBM WebSphere Portal の IBM Web Content Manager は、動的キャッシュ・インスタンスを使用して、拡張キャッシングが使用可能のときに Web Content Manager から取得されるレンダリング内容を保管することができます。WebSphere DataPower XC10 アプライアンス キャッシュ・アプライアンスは、デフォルトの動的キャッシュ実装を使用する代わりに、キャッシュ内容を柔軟性のあるデータ・グリッドに保管する、動的キャッシュの実装を提供します。

WebSphere Commerce

WebSphere Commerce バージョン 7.0.0.1 は、WebSphere eXtreme Scale クライアント バージョン 7.1 の使用をサポートするようになりました。WebSphere DataPower XC10 アプライアンス を使用して、WebSphere Commerce の動的キャッシュ・データをキャッシュに入れることができます。詳しくは、[動的キャッシュの作成データ・グリッド](#)を参照してください。

Tivoli monitoring

WebSphere DataPower XC10 アプライアンス の SNMP モニターを有効にすると、IBM Tivoli® Monitoring バージョン 6.2.2 フィックスパック 2 以降にそのデータをインポートすることができます。Tivoli Monitoring を使用して、複数のオペレーティング・システム上のシステムとネットワークのアプリケーションをモニターおよび管理し、企業内のあらゆる部分の可用性とパフォーマンスを追跡し、傾向を追跡するレポートを作成し、問題のトラブルシューティングをすることができます。詳しくは、[WebSphere DataPower XC10 アプライアンスから Tivoli Monitoring への SNMP データのインポート](#)を参照してください。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

関連情報:

☞ [WebSphere eXtreme Scale を使用した WebSphere Portal および IBM Web Content Manager パフォーマンスの向上](#)

Microsoft .NET に関する考慮事項

WebSphere® eXtreme Scale には 2 つの .NET 環境 (開発環境とランタイム環境) が存在します。これらの環境はそれぞれ固有の要件を持っています。

開発環境の要件

Microsoft .NET バージョン

.NET 3.5 以降のバージョンがサポートされます。

Microsoft Visual Studio

次のいずれかの Visual Studio バージョンを使用することができます。

- Visual Studio 2008 SP1
- Visual Studio 2010 SP1
- Visual Studio 2012

Windows

ご使用の Visual Studio のリリースでサポートされる任意の Windows バージョンがサポートされます。Visual Studio の Windows 要件については、次のリンクを参照してください。

- [Visual Studio 2008 システム要件](#)
- [Visual Studio 2010 Professional システム要件](#)
- [Visual Studio 2012 Professional システム要件](#)

メモリー

- 1 GB (32 ビットおよび 64 ビットのいずれのインストール済み環境にも適用されます)

ディスク・スペース

WebSphere eXtreme Scale は、Visual Studio 要件のほかに、さらに 50 MB の使用可能ディスク・スペースを必要とします。

ランタイム環境

Microsoft .NET バージョン

.NET 3.5 以降のバージョン (.NET 4.0 専用環境での実行を含む) がサポートされます。

Windows

上記の Microsoft .NET バージョン要件を満たす任意の Windows 環境。

メモリー

WebSphere eXtreme Scale サーバーに保管されているデータにアクセスするプロセスあたり 65 MB。

ディスク・スペース

WebSphere eXtreme Scale は 35 MB の使用可能ディスク・スペースを必要とします。トレースを使用可能に設定しているときは、最大 2.5 GB の追加のディスク・スペースが必要です。

WebSphere eXtreme Scaleruntime

.NET クライアント・アプリケーションを使用しているときには、eXtremeIO トランспорт・メカニズムを使用していなければなりません。eXtremeIO については、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

2.5+ ASP.NET セッション状態ストア・プロバイダーの要件

- ASP.NET セッション・ストア・プロバイダーでは、以下の IIS サーバー・バージョンのいずれかが必要です。
 - IIS 6.0 (Windows Server 2003 に同梱)
 - IIS 7.0 (Windows Server 2008 に同梱)
 - IIS 7.5 (Windows Server 2008 R2 に同梱)
 - IIS 8.0 (Windows Server 2012 に同梱)
- メモリー: プロセスごとに 120 MB の追加 (合計メモリー 185 MB)。
- セキュリティー: ASP.NET アプリケーション・プール ID が管理者特権を有する必要があります。
- セキュリティー: ASP.NET アプリケーションのトラスト・レベルを full に設定する必要があります。

親トピック: [DataPower XC10 アプライアンス環境の計画](#)

WebSphere DataPower XC10 アプライアンス のインストール

ご使用の既存の環境に WebSphere® DataPower® XC10 アプライアンス をインストールするには、最初にアプライアンス・ハードウェアをインストールする必要があります。次に、アプリケーション環境に WebSphere eXtreme Scale クライアントをインストールします。

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

アプライアンスの使用を開始する前に、アプライアンスのラックへのインストール、アクセスのためのアプライアンスの構成、ユーザー・インターフェースの開始、およびアプライアンスの操作状態の確認を行う必要があります。

[IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成](#)

DataPower XC10 アプライアンス ハードウェアを取り付け、接続すると、アプライアンスを初期設定および構成する準備は完了します。

[WebSphere eXtreme Scale クライアント のインストール](#)

IBM® WebSphere DataPower XC10 アプライアンス を WebSphere Application Server または、スタンドアロン環境のクライアントと連携させるには、WebSphere eXtreme Scale クライアント をインストールするか、クライアントの Java™ アーカイブ (JAR) ファイルをアプリケーションとともにパッケージ化する必要があります。

[Liberty プロファイルのインストール](#)

Installation Manager を使用するか、Java アーカイブ (JAR) ファイルを実行することで、Liberty プロファイル・アプリケーション・サービス環境をインストールします。

[製品インストールのトラブルシューティング](#)

IBM Installation Manager は多くの IBM ソフトウェア製品のための共通インストーラーです。このバージョンの WebSphere eXtreme Scale をインストールするには、このインストーラーを使用します。

クイック・スタート: アプライアンス・ハードウェアのインストール

アプライアンスの使用を開始する前に、アプライアンスのラックへのインストール、アクセスのためのアプライアンスの構成、ユーザー・インターフェースの開始、およびアプライアンスの操作状態の確認を行う必要があります。

始める前に

- MGMT0 の管理ポート用に 1 ギガビット・ネットワーク接続が必要です。
- データ・グリッドに 1 ギガビット・イーサネット・ポートを使用するか、10 ギガビット・イーサネット・ポートを使用するかを決定します。左のイーサネット・モジュール (1 ギガビット・ポート) または右のイーサネット・モジュール (10 ギガビット・ポート) のいずれかを使用する必要があります。
- 管理アクセス用およびデータ・グリッド用のイーサネット・インターフェースの IP アドレス。
- イーサネット・インターフェースのサブネットをサポートするデフォルト・ゲートウェイ (ルーター) の IP アドレス。
- ネットワーク・サービス (SSH、Telnet、など) の IP アドレス。

手順

1. アプライアンスを物理的にラックに取り付けます。

重要: アプライアンスのケースは開けないでください。ケースを開けると、セキュリティ障害が発生したとみなされ、アプライアンスは作動しなくなります。その場合、再設定のためにアプライアンスを IBM® に返却しなければなりません。

- a. アプライアンスを箱から丁寧に取り出します。付属の電源コード、シリアル・ケーブル、およびレールをすべて確認します。
- b. アプライアンスを取り付けるラックの位置を決め、アプライアンス保守のため、アプライアンスの上側および下側に、空気の流れを確保する十分なスペースがあることを確認します。
- c. マウント・レールを取り付けます。
- d. アプライアンスをレールに取り付け、アプライアンスを所定の位置までスライドします。
- e. アプライアンスの正面から、ネットワーク・ケーブルを使用してアプライアンスをネットワークに接続します。イーサネット・ケーブルは付属していません。
 - MGMT0 管理ポートを 1 ギガビット・ネットワークに接続します。
 - データ・グリッドの非管理トラフィックでは、左のイーサネット・モジュール (1 ギガビット・ポート) または右のイーサネット・モジュール (10 ギガビット・ポート) のいずれかを使用します。
- f. 付属の電源コードを使用して、電源機構を電源コンセントに接続します。
- g. シリアル・コンソールを、デバイス正面の CONSOLE コネクタに接続します。接続構成は、9600 ボー、8N1 (キヤラクター長 8 ビット、パリティなし、1 ストップ・ビット)、およびフロー制御なしです。提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して接続します。PL-2303 USB シリアル・ケーブルを使用する場合は、ケーブル用のドライバーをダウンロードしてインストールしてください。シリアル・コンソールに推奨される端末エミュレーションは、VT100 です。
- h. 電源ボタンを押します。緑色の電源 LED ライトが点灯し、シリアル・コンソールが表示されます。
- i. ログイン・プロンプトが表示されます。初期アプライアンス構成では、ユーザー ID とパスワード `xcadmin/xcadmin` を使用してログインします。

重要: `xcadmin` ユーザー ID とパスワードを失くさないよう注意してください。これらの情報を失くすと、アプライアンスに再度ログインすることができなくなり、再設定のためにアプライアンスを IBM に返送していただく必要があります。この場合、アプライアンスにあるデータはすべて消去されます。`xcadmin` ユーザー ID とパスワードに確実にアクセスできるようにするため、SMTP サーバーと E メール・アドレスを構成し、`xcadmin` パスワードを再設定できるようにすることをお勧めします。

2. シリアル・コンソールでのアクセス用にアプライアンスを構成します。ご使用条件に同意し、イーサネット・ポートを構成する一連のプロセスをウィザードが支援します。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。
3. ご使用のアプライアンス・ファームウェアを最新のバージョンに更新してください。詳しくは、[ファームウェアの更新](#)を参照してください。
4. ファームウェアのアップグレード後にアプライアンスの始動状況を追跡します。ユーザー・インターフェースでアプライアンスの始動状況を追跡することができます。Web ブラウザーのアドレス・バーから `https://<appliance_hostname>:9443/` にアクセスしてください。アプライアンスの初期化に使用している端末から `start-progress` コマンドを実行して進行状況をモニターすることもできます。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。
5. ユーザー・インターフェースを開始します。
 - a. Web ブラウザーのアドレス・バーに、デバイスの初期設定時に定義した URL およびポート情報を入力します。定義した IP アドレスまたはその IP アドレスに対応するホスト名 (例え

ば、<https://myXC10.ibm.com>) を使用できます。セキュア HTTP プロトコル (HTTPS) を使用してください。

- b. 「ユーザー」フィールドに `xcadmin` を入力します。
- c. 「パスワード」フィールドに対応するパスワードを入力します。このパスワードの初期設定は `xcadmin` です。
- d. 「ログイン」をクリックします。ログアウトするには、「ログアウト」をクリックします。

6. アプライアンスの動作状態を確認します。

- アプライアンス正面の緑色の電源 LED が点灯している。
- アプライアンス正面のオレンジ色の障害 LED が点灯していない。
- アプライアンス正面の緑色のキャッシング LED が点灯している。
- アプライアンス正面の琥珀色のキャッシング・エラー LED が点灯していない。
- アプライアンス正面の液晶ディスプレイ (LCD) に製品のリリースおよびビルド・バージョンが表示されている。

これらの各 LED のアプライアンス上での場所を示した図については、[アプライアンスの仕様およびフィーチャー](#) を参照してください。

ご不明な点がある場合は、IBM サポートにお問い合わせください。http://www-947.ibm.com/support/entry/portal/overview/software/websphere/websphere_datapower_xc10_appliance からアクセスできます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

関連タスク:

[ファームウェアの更新](#)

[IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成](#)

IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成

DataPower® XC10 アプライアンス ハードウェアを取り付け、接続すると、アプライアンスを初期設定および構成する準備は完了します。

始める前に

初期構成では、シリアル接続を使用する必要があります。ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC を、アプライアンスのシリアル・ポートにシリアル接続する必要があります。シリアル・コンソールとして PC を使用している場合は、Windows 用または Linux 用の PC ベースのシリアル通信プログラムを使用する必要があります。VT100 スタイルのコンソールなどの専用ハードウェア・デバイス、またはハイパーターミナルや Minicom などのエミュレーターが稼働する PC が使用できます。提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して、アプライアンスに接続します。

重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。

基本構成を定義する前に、以下の情報を確認してください。

- MGMT0 の管理ポート用に 1 ギガビット・ネットワーク接続が必要です。
- データ・グリッドに 1 ギガビット・イーサネット・ポートを使用するか、10 ギガビット・イーサネット・ポートを使用するかを決定します。左のイーサネット・モジュール (1 ギガビット・ポート) または右のイーサネット・モジュール (10 ギガビット・ポート) のいずれかを使用する必要があります。
- アプライアンス管理アクセス用のイーサネット・インターフェース (MGMT) の IP アドレスおよびサブネット・マスク。
- イーサネット・インターフェースのサブネットをサポートするデフォルト・ゲートウェイ (ルーター) の IP アドレス。
- ドメイン・ネーム・システム (DNS) サーバーの IP アドレス。
- メール・サーバー情報 (E メール通知のセットアップ用)。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)を参照してください。
- 集合内のアプライアンス間の通信を可能にするための、Network Time Protocol (NTP) サーバー情報。

このタスクについて

初めて DataPower XC10 アプライアンス の初期設定を行うには、このタスクを使用します。アプライアンスに初めてサインオンする手順は、2 回目以降のサインオン手順と若干異なります。

注意:

1. [xcadminパスワード](#)の **xcadmin** ユーザー ID および パスワードの保存に関する重要な情報を参照してください。
2. アプライアンスのケースは開けないでください。ケースを開けると、セキュリティー障害が発生したとみなされ、アプライアンスは作動しなくなります。アプライアンスを復元するには、**device clear-intrusion** コマンドを実行する必要があります。
3. アプライアンスに付属されていたシリアル・ケーブルを紛失しないでください。アプライアンスの初期セットアップにこのケーブルは必要ですが、問題判別のために後から必要になることもあります。このケーブルは、このアプライアンスで機能することを目的として特別に設計されました。他のシリアル・ケーブルでは、このアプライアンスで機能しない場合があります。

初期の基本ファームウェア構成を実行してください。この構成は、WebSphere® DataPower XC10 アプライアンス をユーザー環境に追加する最小構成です。

手順

1. アプライアンスを初期設定します。以下の手順に従ってください。
 - a. アプライアンスに同梱されているシリアル・ケーブルまたは USB シリアル・ケーブルを使用して、シリアル・コンソールをアプライアンスに接続します。デバイスの前面にある CONSOLE コネクタにケーブルを接続し、ASCII 端末または PC で端末エミュレーション・ソフトウェアを実行させておく必要があります。アプライアンスが始動すると、この接続を使用して、アプライアンスからのメッセージを表示することができます。9600 ボー 8N1 (キャラクター長 8 ビット、パリティなし、1 ストップ・ビット)、およびフロー制御なしで、エミュレーション・ソフトウェアを構成します。シリアル・コンソールに推奨される端末エミュレーションは、VT100 です。
 - b. アプライアンスの電源がオンになっていることを確認してください。アプライアンスの電源がオンになっていない場合は、電源ボタンを切り替えてください。電源ボタンは、アプライアンスの正面にあります。アプライアンスが始動されるまで数秒間待ちます。アプライアンスの電源をオンにすると、以下の状態になります。
 - アプライアンス正面の緑色の電源 LED ライトが点灯し、ファンが起動します。
 - シリアル・コンソールにログイン・プロンプトが表示されます。初期アプライアンス構成では、ユーザー ID とデフォルト・パスワード **xcadmin/xcadmin** を使用してログインします。

- c. WebSphere DataPower XC10 アプライアンス 使用許諾契約書に同意します。先に進む前に、初回の場合はライセンスに同意する必要があります。各ライセンス・プロンプトで、「Accept」、「Reject」、または「StartOver」と入力します。
 - d. MGMT イーサネット・ポートを構成します。クラスレス・ドメイン間ルーティング (CIDR) 形式で IP アドレスを指定します。
 - e. MGMT イーサネット・ポートのデフォルト・ゲートウェイを構成します。
 - f. データ・グリッドのイーサネット・ポートを構成します。使用するポートがすべて 1 ギガビット・ポートであるか、またはすべて 10 ギガビット・ポートであるかを指定します。適用可能なポートの CIDR アドレスを構成します。
 - g. ドメイン・ネーム・システム (DNS) サーバーを構成します。DNS サーバーの有効な IP アドレスを指定します。
2. アプライアンス・ファームウェアを更新します。WebSphere DataPower XC10 アプライアンスのファームウェアをダウンロードまたは更新するには、アプライアンス管理許可が必要です。ファームウェア更新を取得するために、アプライアンスを外部インターネットに接続する必要はありません。ファームウェアのダウンロードまたは更新について詳しくは、『[ファームウェアの更新](#)』を参照してください。
 3. 新規アプライアンスにファームウェアをはじめてインストールする場合には、アプライアンスで **clear-all** コマンドを実行する必要があります。ファームウェア更新からの再始動が終了した後に、以下の手順を実行します。
 - a. xcadmin ユーザーを使用して、アプライアンスとの接続を確立します。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
 - b. **clear-all** コマンドを実行します。

```
Console> clear-all
Force Stopped all XC-10 processes
Deleting configuration data and logs
Deleting grid data
```

注意:

clear-all コマンドを実行する前に、他のコマンドを実行しないでください。他のコマンドを実行すると、アプライアンス構成で問題が生じる可能性があります。

注意:

clear-all コマンドの実行中に、他のコマンドを実行しないでください。 **start-progress** コマンドまたはアプライアンス始動状況ユーザー・インターフェース・ページを使用して進行状況をモニターするだけにしてください。

- c. **clear-all** コマンドの実行後に、アプライアンスの始動の進行状況を追跡することができます。以下のいずれかのオプションを使用します。
 - **2.5+** アプライアンスの始動状況をユーザー・インターフェースで追跡することができます。Web ブラウザーのアドレス・バーで、次の URL およびポート情報を入力してください。 https://<appliance_hostname>:9443/。詳しくは、[アプライアンスの始動状況のモニタ](#)を参照してください。
 - **start-progress** コマンドを実行して、進行をモニターします。このコマンドが STARTED を返す場合、アプライアンスは使用可能です。
4. 構成を保護するために、xcadmin ユーザーのパスワードを変更します。デフォルト・パスワードは xcadmin です。以下のコマンドでパスワードを変更できます。

```
user password <old_password> <new_password>
```

5. 構成を検証します。Web ブラウザーのユーザー・インターフェースを使用して、構成を検証します。

重要: 検証手順では、イーサネット・インターフェースのホスト名を myXC10.ibm.com と想定しています。

ブラウザーからユーザー・インターフェースにアクセスするには、以下の手順を使用します。

- a. Web ブラウザーを開きます。ネットワークに接続されたコンピューターで、Web ブラウザーを開きます。
- b. URL を入力します。アドレス・バーに、デバイスの初期設定時に定義した URL を入力します。例えば、<https://myXC10.ibm.com> と入力します。

注: http ではなく https プロトコルを使用してください。

- c. アプライアンスにログオンします。ローカルの XCADMIN のアカウントおよびパスワードを使用して、アプライアンスにログオンします。入力したパスワードは平文のため、表示されません。

d. 「ログイン」をクリックします。

「ようこそ」ページが表示されたら、ローカル XCADMIN アカウントの認証は成功です。

6. 構成を完了します。

タスクの結果

以上で、WebSphere DataPower XC10 アプライアンス の初期設定は完了です。

次のタスク

ユーザー・インターフェースでアプライアンス設定を構成します。これらの設定には、ユーザーとユーザー・グループ、セキュリティー、Eメール送達、ドメイン・ネーム・システム (DNS) サーバー、および日時の各設定が含まれます。詳しくは、[アプライアンスの構成](#)を参照してください。

[xcadmin/パスワード](#)

管理 (xcadmin) ユーザー ID とパスワードを設定した後は、その情報を信頼のおける場所に保管してください。Eメール送達を構成して、ユーザー・インターフェースでのパスワードの再設定機能を使用可能にすることができます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

関連タスク:

[ファームウェアの更新](#)

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

xcadminパスワード

管理 (xcadmin) ユーザー ID とパスワードを設定した後は、その情報を信頼のおける場所に保管してください。Eメール送達を構成して、ユーザー・インターフェースでのパスワードの再設定機能を使用可能にすることができます。

ユーザー ID およびパスワードの保存

xcadmin ユーザー ID とパスワードはアプライアンスにログオンするために必要です。そのため、ユーザー ID とパスワードを変更した後は、それらを安全な場所に保管してください。それらを忘れてしまい、復旧する手段を用意していない場合には、再設定のためにアプライアンスを IBM® に返却する必要があります。

Eメールでのパスワードの再設定

Eメール送達を構成すると、すべてのユーザーが、ユーザー・インターフェースログイン画面の「パスワードを忘れた場合」リンクをクリックすることで、パスワードを復元できます。新たに生成されたパスワードが含まれたEメールがユーザーに送達されます。

注意:

ユーザー・インターフェースのログイン画面の「パスワードを忘れた場合」リンクが、**xcadmin** パスワードを再設定するための唯一の方法です。Eメール送達を構成していない場合にパスワードを忘れた場合は、**device RESET** コマンドを使用してアプライアンスを再設定する必要があります。このコマンドでは、アプライアンスのすべての設定が再設定されます。**device RESET** コマンドについて詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。

xcadmin パスワードの変更

ユーザー・インターフェースまたはアプライアンスのコマンド行インターフェースで、xcadmin のパスワードを編集できます。

ユーザー・インターフェースでパスワードを変更する場合は、ユーザーを編集します。詳しくは、[ユーザーの管理](#)を参照してください。

アプライアンスのコマンド行インターフェースでパスワードを変更する場合は、**user password** コマンドを使用できます。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。

親トピック: [IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)
[ユーザー許可](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)
[ユーザーおよびグループの管理](#)
[データ・グリッドのセキュリティーを使用可能に設定する](#)
[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)
[IBM WebSphere DataPower XC10 アプライアンスのEメール送達の管理](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

WebSphere eXtreme Scale クライアント のインストール

IBM® WebSphere® DataPower® XC10 アプライアンス を WebSphere Application Server または、スタンドアロン環境のクライアントと連携させるには、 WebSphere eXtreme Scale クライアント をインストールするか、クライアントの Java™ アーカイブ (JAR) ファイルをアプリケーションとともにパッケージ化する必要があります。

このタスクについて

クライアント・アプリケーションが IBM WebSphere DataPower XC10 アプライアンスと通信するには、 WebSphere eXtreme Scale クライアントをインストールする必要があります。 クライアントをインストールする前に、スタンドアロン環境でクライアントをインストールするのか、あるいは、 WebSphere Application Server 環境にインストールするのかを認識している必要があります。

アプリケーションで WebSphere Application Server 動的キャッシュまたは HTTP セッションを使用する場合は、 WebSphere eXtreme Scale クライアントを WebSphere Application Server 環境にインストールする必要があります。

単純なデータ・グリッドは、組み込み WebSphere Application Server 環境でも、スタンドアロン環境でも使用できます。 クライアントをスタンドアロン環境にインストールすると、インストール済み環境は WebSphere Application Server を使用しません。

データ・グリッドのタイプの詳細については、 [アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#) を参照してください。

手順

1. IBM Installation Manager をダウンロードおよびインストールして、必要な製品リポジリーが確実にインストールされていることを確認してください。 [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#) を参照してください。
2. サポート・サイトから WebSphere eXtreme Scale クライアントをダウンロードします。 クライアントをダウンロードする場所の詳細については、 [サポート・ポータル](#) を参照してください。
3. Installation Manager を使用して、インストールを実行します。 正しい製品オフリングを選択してください。 製品オフリングは、Installation Manager 内のインストールの設定に追加したリポジリーに応じて使用可能になります。 WebSphere eXtreme Scale クライアント に使用可能な製品オフリングには、以下が含まれます。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 6
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

以下の中から、計画した構成に適切なインストールを使用します。

- **組み込みクライアント・インストール:** 動的キャッシュまたは HTTP セッションを使用する場合は、組み込みインストールを使用する必要があります。 また、単純なデータ・グリッドで組み込みインストールを使用することもできます。 詳しくは、 [WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- **スタンドアロン・クライアント・インストール:** スタンドアロン環境のクライアント・インストールは、単純なデータ・グリッドでのみ使用できます。 詳しくは、 [GUI の使用による WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- **サイレント・インストール:** スタンドアロン環境、あるいは、組み込み WebSphere Application Server 環境で、応答ファイルを使用したクライアントのインストールもできます。 詳しくは、 [応答ファイルの使用による WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。

次のタスク

アプライアンスを構成します。 詳しくは、 [アプライアンスの構成](#) を参照してください。

[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングは、製品リポジリーで使用可能です。 これらのリポジリーにアクセスするには、まず IBM Installation Manager をインストールする必要があります。

[IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

IBM Installation Manager を使用して WebSphere eXtreme Scale クライアント 製品オフリングをアンインストールします。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

WebSphere eXtreme Scale Client for .NET は、ランタイム環境でインストールすることもできれば、ランタイム環境と開発環境の両方でインストールすることもできます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール

WebSphere® eXtreme Scale クライアント 製品オフリングは、製品リポジトリで使用可能です。これらのリポジトリにアクセスするには、まず IBM® Installation Manager をインストールする必要があります。

Installation Manager は、製品メディア内のファイル、パスポート・アドバンテージのサイトから入手したファイル、もしくは [IBM Installation Manager ダウンロード Web サイト](#) から入手したファイルのいずれかを使用してインストールできます。ファイルはインストール・イメージが収められた圧縮ファイルです。

注:

Installation Manager は、32 ビット版または 64 ビット版でダウンロードすることができます。どちらの版の Installation Manager を使用しても、WebSphere eXtreme Scale をインストールできます。

Installation Manager を使用すると、必要な製品リポジトリにアクセスできます。WebSphere eXtreme Scale 製品オフリングをインストールするには、これらのリポジトリにアクセスする必要があります。

製品リポジトリにアクセスするには、2 つのオプションがあります。

オプション 1: 物理メディア上の製品リポジトリにアクセスして、ローカル・インストールを行う

1. システムに Installation Manager をインストールします。
2. Installation Manager を使用して、メディア上の製品リポジトリから製品オフリングをインストールします。

オプション 2: パスポート・アドバンテージから製品リポジトリをダウンロードして、ローカル・インストールを行う

1. パスポート・アドバンテージのサイトからリポジトリをダウンロードします。

注: IBM パスポート・アドバンテージ・オンライン Web サイトからダウンロード可能な IBM WebSphere eXtreme Scale のインストール・イメージのリストとその他の情報については、[サポートされるソフトウェア](#)を参照してください。

2. システムに Installation Manager をインストールします。
3. Installation Manager を使用して、ダウンロードした製品リポジトリから製品をインストールします。

[WebSphere eXtreme Scale クライアント 製品のオフリング ID](#)

製品の更新をインストールする場合、あるいはフィックスをロールバックする場合、コマンド行でオフリング ID を指定する必要があります。製品オフリングは、下の表を使用して特定してください。

[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)

WebSphere eXtreme Scale クライアントを、WebSphere Application Server または WebSphere Application Server Network Deployment がインストールされている環境にインストールできます。WebSphere Application Server または WebSphere Application Server Network Deployment の既存のフィーチャーを使用して、WebSphere DataPower XC10 アプライアンス アプリケーションを拡張できます。

[GUI の使用による IBM Installation Manager のインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM Installation Manager をインストールする必要があります。Installation Manager はウィザード・コンソールを使用してインストールできます。

[コマンド行の使用による IBM Installation Manager のインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM Installation Manager をインストールする必要があります。Installation Manager はコマンド行からインストールできます。

[応答ファイルの使用による IBM Installation Manager のインストール](#)

WebSphere eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM Installation Manager をインストールする必要があります。Installation Manager は応答ファイルを使用してインストールできます。

親トピック: [WebSphere eXtreme Scale クライアント のインストール](#)

WebSphere eXtreme Scale クライアント 製品のオフリング ID

製品の更新をインストールする場合、あるいはフィックスをロールバックする場合、コマンド行でオフリング ID を指定する必要があります。製品オフリングは、下の表を使用して特定してください。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール

WebSphere® eXtreme Scale クライアントを、WebSphere Application Server または WebSphere Application Server Network Deployment がインストールされている環境にインストールできます。WebSphere Application Server または WebSphere Application Server Network Deployment の既存のフィーチャーを使用して、WebSphere DataPower® XC10 アプライアンス アプリケーションを拡張できます。

始める前に

- ターゲット・インストール・ディレクトリーに WebSphere eXtreme Scale クライアントの既存のインストール済み環境が含まれていないことを確認します。
- WebSphere Application Server または WebSphere Application Server Network Deployment 環境で実行中のすべてのプロセスを停止します。**stopManager**、**stopNode**、および **stopServer** コマンドについて詳しくは、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。

注意:

すべての実行中のプロセスが停止していることを確認します。実行中のプロセスが停止していない場合でもインストールは続行しますが、予測不能な結果が生じます。一部のプラットフォームでは、インストールが不確定な状態のままになることがあります。

重要: WebSphere eXtreme Scale クライアント をインストールする際には、WebSphere Application Server をインストールしたのと同じディレクトリーにインストールする必要があります。例えば、WebSphere Application Server を C:¥[was_root](#) にインストールした場合は、C:¥[was_root](#) を、WebSphere eXtreme Scale クライアント のインストールのターゲット・ディレクトリーとしても選択する必要があります。

このタスクについて

eXtreme Scale を WebSphere Application Server または WebSphere Application Server Network Deployment に統合して、eXtreme Scale の機能をご使用の Java™ Platform, Enterprise Edition アプリケーションに適用します。Java EE アプリケーションは、クライアント接続を使用してそのデータ・グリッドにアクセスします。

手順

- WebSphere eXtreme Scale クライアント を WebSphere Application Server バージョン 8 環境内にインストールする場合、以下のステップを実行します。
 1. IBM Installation Manager をインストールします。詳しくは、[GUI の使用による IBM Installation Manager のインストール](#)を参照してください。
 2. Installation Manager を使用して、適切な eXtreme Scale 製品オフリングをインストールします。
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。
 3. パスポート・アドバンテージのサイトから必要な WebSphere Application Server バージョン 8 のリポジトリをダウンロードします。詳しくは、「[How to download WebSphere Application Server - Express V8.5 from Passport Advantage](#)」を参照してください。
 4. WebSphere Application Server バージョン 8 をインストールします。詳しくは、[GUI を使用した、分散オペレーティング・システムでの製品のインストール](#)を参照してください。
- WebSphere eXtreme Scale クライアント を WebSphere Application Server バージョン 7 環境内にインストールする必要がある場合、以下のステップを実行します。
 1. IBM Installation Manager をインストールします。詳しくは、[GUI の使用による IBM Installation Manager のインストール](#)を参照してください。
 2. InstallShield MultiPlatform (ISMP) インストーラーを使用して、WebSphere Application Server バージョン 7 をインストールします。詳しくは、[アプリケーション・サービス提供環境のインストール](#)を参照してください。
 3. インストール後、インストールを完了するために WebSphere Application Server バージョン 7 を Installation Manager にインポートする必要があります。WebSphere Application Server バージョン 7 を Installation Manager にインポートすると、製品のフィックスパックの管理とインストールが 1 つのロケーションから行えます。フィックスパックおよび更新にアクセスできるよう、必要なリポジリーが Installation Manager 内にセットアップされていることを確認してください。WebSphere Application Server 7 の既存のインストール済み環境を Installation Manager にインポートする方法については、[IBM Installation Manager レジストリーへの IBM WebSphere Application Server 製品情報のインポート \(Importing IBM WebSphere Application Server product information into the IBM Installation Manager registry\)](#) を参照してください。
 4. Installation Manager を使用して、適切な eXtreme Scale 製品オフリングをインストールします。

- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフラインのインストール](#)

GUI の使用による IBM Installation Manager のインストール

WebSphere® eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM® Installation Manager をインストールする必要があります。Installation Manager はウィザード・コンソールを使用してインストールできます。

始める前に

IBM Installation Manager をインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

手順

1. Installation Manager のインストール・ファイルが含まれているロケーションから、次のいずれかのコマンドを実行します。

管理インストール:

- **Windows** install.exe
- **UNIX Linux** ./install

非管理インストール:

- **Windows** userinst.exe
- **UNIX Linux** ./userinst

管理インストールおよび非管理インストールについて詳しくは、[管理者、非管理者、またはグループとしてインストール](#)を参照してください。

グループ・モードでのインストール:

- **UNIX Linux** ./groupinst

グループ・モードに関する注記:

- グループ・モードでは、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理することができます。

グループ・モードでは、IBM Installation Manager の単一インスタンスを 2 人のユーザーが同時に使用することはできません。
- **Windows** グループ・モードは、Windows オペレーティング・システムでは使用できません。
- グループ・モードを使用せずに Installation Manager をインストールする場合、後からこの Installation Manager インスタンスを使用してインストールする製品を管理するためにグループ・モードを使用することはできません。
- 現行ユーザーのインストール・ロケーションをデフォルト・ロケーションから、グループ内のすべてのユーザーがアクセスできるロケーションに変更します。
- グループ・モードでインストールする前に、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップの説明のとおり、グループ、権限、および環境変数を設定してください。
- グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。

2. Installation Manager のパッケージが選択されていることを確認して、「次へ」をクリックします。
3. 使用条件の条項に同意し、「次へ」をクリックします。
4. 「次へ」をクリックします。
5. 要約情報を検討して、「インストール」をクリックします。正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。
6. 製品リポジトリを Installation Manager 設定に追加します。
 - a. Installation Manager を開始します。
 - b. トップ・メニューで、「ファイル」>「設定」をクリックします。
 - c. 「リポジトリ」を選択します。
 - d. 「リポジトリの追加」をクリックします。
 - e. リポジトリ・ファイルを含むロケーション内の repository.config ファイルのパスを入力します。以下に例を示します。

- **Windows** C:¥repositories¥product_name¥local-repositories

- **UNIX** | **Linux** /var/repositories/product_name/local-repositories

- f. 「**OK**」をクリックします。
7. 「リポジトリ」ウィンドウにリストされているロケーションのうち使用していないものをクリアします。
8. 「適用」をクリックします。
9. 「**OK**」をクリックします。
10. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

次のタスク

Installation Manager のインストールとリポジトリのセットアップが正常に完了したら、引き続き製品オフライン用の WebSphere eXtreme Scale クライアント、または for をインストールすることができます。詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

ウィザード・コンソールから Installation Manager を使用し、WebSphere eXtreme Scale クライアント 製品 オフラインをインストールします。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフラインのインストール](#)

GUI の使用による WebSphere eXtreme Scale クライアント のインストール

ウィザード・コンソールから Installation Manager を使用し、WebSphere® eXtreme Scale クライアント 製品オフアリングをインストールします。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフアリングのインストール](#)を参照してください。

手順

1. Installation Manager を開始します。

UNIX | **Linux** ヒント: `./IBMIM` コマンドを使用して、Installation Manager をグループ・モードで開始することができます。

- グループ・モードを使用すると、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理できるようになります。
- グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。

2. 「インストール」をクリックします。

注: 認証のプロンプトが出されたら、プログラムの Web サイトで登録した IBM ID とパスワードを使用してください。

Installation Manager は、定義済みリポジトリ内にある使用可能なパッケージを検索します。

3. 以下のいずれかの製品オフアリングと適切なバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

ご使用のシステムに既に当該製品がインストールされている場合は、その製品が既にインストールされていることを示すメッセージが表示されます。別のロケーションに製品のインストール済み環境を別途作成するには、「続行」をクリックします。

ヒント: 「インストール・マネージャー・リポジトリ設定」ページで「インストール中および更新中にサービス・リポジトリの検索」オプションが選択されていて、インターネットに接続している場合、「他のバージョンと拡張機能の確認」をクリックできます。これにより、選択したパッケージのデフォルト更新リポジトリで更新を検索することができます。この場合、「インストール・マネージャー・リポジトリ設定」ページに特定のサービス・リポジトリ URL を追加する必要はありません。

- a. インストールするフィックスを選択します。

デフォルトでは、任意の推奨フィックスが選択されます。

推奨フィックスがある場合、推奨フィックスのみを表示し、非推奨フィックスを非表示にするオプションを選択できます。

- b. 「次へ」をクリックします。

注: Installation Manager は、リポジトリに接続する際に、最新レベルの Installation Manager への更新を要求するプロンプトを出す場合があります。プロンプトが出されたら、新しいバージョンに更新してから続行します。自動更新については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

4. 使用条件の条項に同意し、「次へ」をクリックします。
5. 製品のインストール・ルート・ディレクトリを指定します。

パネルには共有リソース・ディレクトリおよびディスク・スペース情報も指定されます。

注: 初めて Installation Manager を使用してパッケージをインストールする場合、共有リソース・ディレクトリを指定します。共有リソース・ディレクトリは、1 つ以上のパッケージ・グループで使用できるインストール成果物が置かれるディレクトリです。このインストールには、最も大きいドライブを使用してください。すべてのパッケージをアンインストールし終わるまで、このディレクトリ・ロケーションは変更できません。

制約事項:

- デフォルト・ターゲット・ロケーションを削除し、インストール・ディレクトリー・フィールドを空のままにすると、続行できなくなります。
- 宛先ディレクトリーとしてシンボリック・リンクを使用しないでください。
シンボリック・リンクはサポートされていません。
- ディレクトリー名にセミコロンを使用しないでください。
ターゲット・ディレクトリーにセミコロンが含まれている場合、WebSphere eXtreme Scale は想定どおりにインストールされません。

Windows セミコロンは、Windows システムでは、クラスパスを構成するために使用する文字です。

- **Windows** Windows Server 2008、Windows Vista、および Windows 7 オペレーティング・システムでの最大パス長は 60 文字です。

6. 「次へ」をクリックします。
7. インストールするコンテンツの翻訳言語を選択します。

常に英語が選択されています。

8. 「次へ」をクリックします。
9. インストールするフィーチャーを選択します。

選択した製品オフリングに応じて、次のフィーチャーから選択できます。

- コンソール

すべての WebSphere eXtreme Scale 製品オフリングで使用可能です。 モニター・コンソールのインストールを選択することができます。 Web コンソールでは、現在と過去の統計をグラフにできます。このコンソールには、概要を表示するようにグラフがいくつか用意されているほか、使用可能な統計からグラフを作成できるカスタム・レポート・ページもあります。 WebSphere eXtreme Scale のモニター・コンソールのグラフ機能を使用して、環境内のデータ・グリッドの全体的なパフォーマンスを表示できます。

- サンプル

すべての WebSphere eXtreme Scale 製品オフリングで使用可能です。

10. 「次へ」をクリックします。
11. 要約情報を検討して、「インストール」をクリックします。

- 正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。

注: さらに、プログラムによって、重要なポストインストール指示も指定されることがあります。

- インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。

12. このインストールの完了後に開始するツールを選択します。

- 実稼働環境に適した設定のアプリケーション・サーバー・プロファイルを新規に作成する場合は、「**プロファイルを作成するためのプロファイル管理ツール**」を選択します。
- 開発環境に適した設定のアプリケーション・サーバー・プロファイルを作成する場合は、「**開発環境のアプリケーション・サーバー・プロファイルを作成するためのプロファイル管理ツール (Profile Management Tool to create an application server profile for a development environment)**」を選択します。

注: development 設定は、アプリケーションの更新が頻繁に行われ、システム・リソースが最小限である開発環境に適しています。実動サーバーには development 設定を使用しないでください。

- このインストールの完了後に、新しいプロファイルを作成しない場合は、「なし」を選択します。

制約事項: プロファイル管理ツールを起動するオプションは、プロファイル管理ツールを含むバージョンの WebSphere Application Server がインストールされている場合にのみ使用可能です。

13. 「終了」をクリックします。
14. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

親トピック: [GUI の使用による IBM Installation Manager のインストール](#)

関連タスク:

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

コマンド行の使用による IBM Installation Manager のインストール

WebSphere® eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM® Installation Manager をインストールする必要があります。Installation Manager はコマンド行からインストールできます。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

手順

1. Installation Manager のインストール・ファイルが含まれているロケーションに移動して、以下のコマンドのいずれかを実行します。

管理インストール:

- **Windows** `installc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./installc -acceptLicense -log log_file_path_and_name`

非管理インストール:

- **Windows** `userinstc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./userinstc -acceptLicense -log log_file_path_and_name`

グループ・モードでのインストール:

```
UNIX Linux ./groupinstc -acceptLicense -dataLocation application_data_location  
-log log_file_path_and_name -installationDirectory Installation_Manager_home
```

グループ・モードに関する注記:

- グループ・モードを使用すると、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理できるようになります。
 - **Windows** グループ・モードは、Windows オペレーティング・システムでは使用できません。
 - グループ・モードを使用せずに Installation Manager をインストールする場合、この Installation Manager を使用して今後インストールする製品はグループ・モードを使用して管理できません。
 - インストール・ロケーションは、必ず現在のユーザーのホーム・ディレクトリー内のデフォルトのロケーションから、グループ内のすべてのユーザーがアクセス可能なロケーションに変更してください。
 - グループ・モードでインストールする前に、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップの説明のとおり、グループ、権限、および環境変数を設定してください。
 - グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。
2. オプション: リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、`imutilsc` コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

次のタスク

Installation Manager のインストールとリポジトリのセットアップが正常に完了したら、引き続き WebSphere eXtreme Scale WebSphere eXtreme Scale クライアント、または WebSphere eXtreme Scale for WebSphere Application Server 製品オフリングをインストールできます。詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

[コマンド行の使用による WebSphere eXtreme Scale クライアント のインストール](#)

コマンド行から Installation Manager を使用し、WebSphere eXtreme Scale クライアント 製品オフリングをインストールします。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

コマンド行の使用による **WebSphere eXtreme Scale** クライアント のインストール

コマンド行から Installation Manager を使用し、WebSphere® eXtreme Scale クライアント 製品オフリングをインストールします。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

手順

1. システムにログオンします。
2. Installation Manager をインストールしたディレクトリーの eclipse/tools サブディレクトリーに移動します。
3. 製品リポジトリが使用可能であることを確認してください。

Windows

```
imcl.exe listAvailablePackages -repositories source_repository
```

UNIX Linux

```
./imcl listAvailablePackages -repositories source_repository
```

オフリングの 1 つ以上のレベルが表示されるはずですが。

4. **imcl** コマンドを使用して、この製品をインストールします。

Windows

```
imcl.exe install com.ibm.websphere.v85_offering_version,optional_feature_ID  
-repositories source_repository  
-installationDirectory installation_directory  
-sharedResourcesDirectory shared_directory  
-accessRights access_mode  
-preferences preference_key=value  
-properties property_key=value  
-keyring keyring_file -password password  
-acceptLicense
```

UNIX Linux

```
./imcl install com.ibm.websphere.offering_version,optional_feature_ID  
-repositories source_repository  
-installationDirectory installation_directory  
-sharedResourcesDirectory shared_directory  
-accessRights access_mode  
-preferences preference_key=value  
-properties property_key=value  
-keyring keyring_file -password password  
-acceptLicense
```

ヒント:

- *offering_ID* は、[WebSphere eXtreme Scale クライアント 製品のオフリング ID](#) にリストされるオフリング ID です。
- *offering_version* は、アンダースコアを使用してオフリング ID にオプションで付加することができます。これはインストール対象のオフリングの特定バージョンです (例: 8.6.0.20110503_0200)。
 - *offering_version* が指定されていない場合、オフリングの最新バージョンと、そのバージョンのすべてのインテリム・フィックスがインストールされます。
 - *offering_version* が指定されている場合、オフリングの指定バージョンがインストールされ、そのバージョンのインテリム・フィックスはインストールされません。

オフリングのバージョンは、リポジトリに対して次のコマンドを実行すると、オフリング ID の最後にアンダースコアを使用して付加されている形で見つかります。

```
imcl listAvailablePackages -repositories source_repository
```

- また、`-installFixes` 引数に `none`、`recommended`、または `all` を指定することで、どのインテリム・フィックスをオフアリングと一緒にインストールするのかが示すこともできます。
 - オフアリングのバージョンが指定されていない場合、`-installFixes` オプションはデフォルトで `all` になります。
 - オフアリングのバージョンが指定されている場合、`-installFixes` オプションはデフォルトで `none` になります。
- コンマで区切られたフィーチャーのリストを追加できます。以下に例を示します。

```
imcl -acceptLicense install  
com.ibm.websphere.WXS.v85,xs.console.feature,xs.samples.feature
```

```
imcl -acceptLicense install  
com.ibm.websphere.WXS.v86,xs.console.feature,xs.samples.feature
```

- すべての製品オフアリングで使用可能です。xs.console.feature モニター・コンソールのインストールを選択することができます。Web コンソールでは、現在と過去の統計をグラフにできます。このコンソールには、概要を表示するように事前構成されたグラフがいくつか用意されているほか、使用可能な統計からグラフを作成できるカスタム・レポート・ページもあります。WebSphere eXtreme Scale のモニター・コンソールのグラフ機能を使用して、環境内のデータ・グリッドの全体的なパフォーマンスを表示できます。
- xs.samples.feature すべての製品オフアリングで使用可能です。サンプルのインストールを選択できます。

注:

- 以前に Installation Manager をインストールするモードを指定した場合、`-accessRights` パラメーターは必要ありません
- 後で問題が生じた場合のために、Installation Manager はパッケージの以前のバージョンをロールバックの対象として保存することができます。Installation Manager が以前のバージョンにパッケージをロールバックする際は、現行バージョンのファイルはアンインストールされて、以前のバージョンが再インストールされます。ロールバック用にファイルを保存しないことを選択する場合は、次の設定を指定することによりファイルが保存されないようにできます。

```
-preference  
com.ibm.cic.common.core.preferences.preserveDownloadedArtifacts=False
```

Installation Manager 設定の設定方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: ロールバック用にファイルを保存しないよう選択した場合でも、リポジトリからロールバック用の製品ファイルにアクセスすることは可能です。

- プログラムが重要なポストインストール指示を標準出力に書き込むことがあります。

imcl コマンドを使用して製品をインストールする方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [コマンド行の使用による IBM Installation Manager のインストール](#)

関連タスク:

[コマンド行の使用による WebSphere eXtreme Scale クライアントのアンインストール](#)

応答ファイルの使用による IBM Installation Manager のインストール

WebSphere® eXtreme Scale クライアント 製品オフリングをインストールできるように、必要な製品リポジトリにアクセスするには、IBM® Installation Manager をインストールする必要があります。Installation Manager は応答ファイルを使用してインストールできます。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

手順

Installation Manager のインストール・ファイルが含まれているロケーションに移動して、以下のコマンドのいずれかを実行して Installation Manager をインストールします。

管理インストール:

- **Windows** `installc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./installc -acceptLicense -log log_file_path_and_name`

非管理インストール:

- **Windows** `userinstc.exe -acceptLicense -log log_file_path_and_name`
- **UNIX Linux** `./userinstc -acceptLicense -log log_file_path_and_name`

グループ・モードでのインストール:

```
UNIX Linux ./groupinstc -acceptLicense -dataLocation application_data_location -log log_file_path_and_name -installationDirectory Installation_Manager_home
```

グループ・モードに関する注記:

- グループ・モードを使用すると、複数のユーザーが IBM Installation Manager の 1 つのインスタンスを使用してソフトウェア・パッケージを管理できるようになります。

グループ・モードは、IBM Installation Manager の単一インスタンスを 2 人のユーザーが同時に使用できるという意味ではありません。

- **Windows** グループ・モードは、Windows オペレーティング・システムでは使用できません。
- グループ・モードを使用せずに Installation Manager をインストールする場合、この Installation Manager を使用して今後インストールする製品はグループ・モードを使用して管理できません。
- インストール・ロケーションは、必ず現在のユーザーのホーム・ディレクトリ内のデフォルトのロケーションから、グループ内のすべてのユーザーがアクセス可能なロケーションに変更してください。
- グループ・モードでインストールする前に、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップの説明のとおり、グループ、権限、および環境変数を設定してください。
- グループ・モードの使用方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)のグループ・モードのロードマップを参照してください。

次のタスク

Installation Manager のインストールとリポジトリのセットアップが正常に完了したら、引き続き製品オフリング用の WebSphere eXtreme Scale WebSphere eXtreme Scale クライアント、または WebSphere eXtreme Scale for WebSphere Application Server をインストールできます。詳しくは、[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)を参照してください。

[応答ファイルの使用による WebSphere eXtreme Scale クライアント のインストール](#)

応答ファイルで Installation Manager を使用し、WebSphere eXtreme Scale クライアント 製品オフリングをインストールします。

親トピック: [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)

応答ファイルの使用によるWebSphere eXtreme Scale クライアント のインストール

応答ファイルで Installation Manager を使用し、WebSphere® eXtreme Scale クライアント 製品オフリングをインストールします。

始める前に

Installation Manager に必要な製品ファイルをインストールし、必要なリポジトリにアクセス可能でなければなりません。詳しくは、[IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オフリングのインストール](#)を参照してください。

このタスクについて

Installation Manager を使用して、応答ファイルを処理してさまざまな方法で製品のインストールを行うことができます。GUI を使用して応答ファイルを記録できます。

手順

1. コマンド行で、Installation Manager をインストールしたディレクトリーの Eclipse サブディレクトリーに移動します。
2. -record オプションを使用して、コマンド行から Installation Manager を開始します。

以下に例を示します。

- **Windows** 管理者または非管理者:

```
IBMIM.exe -skipInstall "C:¥temp¥imRegistry"
-record C:¥temp¥install_response_file.xml
```

- **UNIX** | **Linux** 管理者:

```
./IBMIM -skipInstall /var/temp/imRegistry
-record /var/temp/install_response_file.xml
```

- **UNIX** | **Linux** 非管理者:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry
-record user_home/var/temp/install_response_file.xml
```

ヒント: 新しい応答ファイルを記録する際に、-skipInstall パラメーターを指定することができます。このパラメーターには、以下の利点があります。

- ファイルをインストールしないので、記録の速度が速くなります。
- -skipInstall パラメーターを指定して一時データ・ロケーションを使用すると、Installation Manager は、記録する際に、インストール・レジストリーを指定されたデータ・ロケーションに書き込みます。-skipInstall パラメーターを指定せずに Installation Manager を再度開始すると、応答ファイルを使用して実際のインストール・レジストリーにインストールすることができます。

-skipInstall の操作は、Installation Manager が使用する実際のエージェント・データ・ロケーションで使用することはできません。この操作はサポートされていません。クリーンで書き込み可能なロケーションを使用し、今後の記録セッションの際にはこのロケーションを再利用してください。

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

3. 適切なリポジトリを Installation Manager 設定に追加します。
 - a. トップ・メニューで、「ファイル」>「設定」をクリックします。
 - b. 「リポジトリ」を選択します。
 - c. 各リポジトリに対して以下のアクションを実行します。
 - i. 「リポジトリの追加」をクリックします。
 - ii.
 - iii. Web ベースのリモート・リポジトリ内、またはリポジトリ・ファイルを解凍したローカル・ディレクトリー内の、repository.config ファイルのパスを入力します。

例:

- リモート・リポジトリ:

https://downloads.mycorp.com:8080/WXS_85_repository

■ ローカル・リポジトリ:

- **Windows** C:\%repositories%\wxs85\local-repositories
- **UNIX | Linux** /var/repositories/wxs85/local-repositories

- iv. 「**OK**」をクリックします。
- v. 「**適用**」をクリックします。
- vi. 「**OK**」をクリックします。

d. 「インストール」をクリックします。

注: 認証のプロンプトが出されたら、プログラムの Web サイトで登録した IBM ID とパスワードを使用してください。

Installation Manager は、定義済みリポジトリ内にある使用可能なパッケージを検索します。

4. 以下のいずれかの製品オフラインと適切なバージョンを選択します。
- WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

ご使用のシステムに既に当該製品がインストールされている場合は、その製品が既にインストールされていることを示すメッセージが表示されます。別のロケーションに製品のインストール済み環境を別途作成するには、「**続行**」をクリックします。

ヒント: 「インストール・マネージャー・リポジトリ設定」ページで「インストール中および更新中にサービス・リポジトリの検索」オプションが選択されていて、インターネットに接続している場合、「**他のバージョンと拡張機能の確認**」をクリックできます。これにより、選択したパッケージのデフォルト更新リポジトリで更新を検索することができます。この場合、「インストール・マネージャー・リポジトリ設定」ページに特定のサービス・リポジトリ URL を追加する必要はありません。

5. インストールするフィックスを選択します。

デフォルトでは、任意の推奨フィックスが選択されます。

推奨フィックスがある場合、推奨フィックスのみを表示し、非推奨フィックスを非表示にするオプションを選択できます。

6. 「**次へ**」をクリックします。
7. 使用条件の条項に同意し、「**次へ**」をクリックします。
8. 製品のインストール・ルート・ディレクトリを指定します。

パネルには共有リソース・ディレクトリおよびディスク・スペース情報も指定されます。

注: 初めて Installation Manager を使用してパッケージをインストールする場合、共有リソース・ディレクトリを指定します。共有リソース・ディレクトリは、1 つ以上のパッケージ・グループで使用できるインストール成果物が置かれるディレクトリです。このインストールには、最も大きいドライブを使用してください。すべてのパッケージをアンインストールし終わるまで、このディレクトリ・ロケーションは変更できません。

制約事項:

- デフォルト・ターゲット・ロケーションを削除し、インストール・ディレクトリ・フィールドを空のままにすると、続行できなくなります。
- 宛先ディレクトリとしてシンボリック・リンクを使用しないでください。
シンボリック・リンクはサポートされていません。
- **Windows** Windows Server 2008、Windows Vista、および Windows 7 オペレーティング・システムでの最大パス長は 60 文字です。

9. 「**次へ**」をクリックします。
10. インストールするコンテンツの翻訳言語を選択します。
常に英語が選択されています。
11. 「**次へ**」をクリックします。
12. インストールするフィーチャーを選択します。
- コンソール

すべての WebSphere eXtreme Scale 製品オファリングで使用可能です。モニター・コンソールのインストールを選択することができます。Web コンソールでは、現在と過去の統計をグラフにできます。このコンソールには、概要を表示するようにグラフがいくつか用意されているほか、使用可能な統計からグラフを作成できるカスタム・レポート・ページもあります。WebSphere eXtreme Scale のモニター・コンソールのグラフ機能を使用して、環境内のデータ・グリッドの全体的なパフォーマンスを表示できます。

- サンプル

すべての WebSphere eXtreme Scale 製品オファリングで使用可能です。

13. 「次へ」をクリックします。

14. 要約情報を検討して、「インストール」をクリックします。

- 正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。

注: さらに、プログラムによって、重要なポストインストール指示も指定されることがあります。

- インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。

15. 「終了」をクリックします。

16. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

鍵リングの作成

WebSphere eXtreme Scale WebSphere eXtreme Scale クライアント 製品オファリングをインストールするための応答ファイルを、Installation Manager を使用して記録した後、鍵リング・ファイルの作成を選択することができます。認証を必要とするリモート・リポジトリを使用している場合、インストールのための鍵リングを作成できます。

親トピック: [応答ファイルの使用による IBM Installation Manager のインストール](#)

関連タスク:

[応答ファイルを使用した WebSphere eXtreme Scale クライアントのアンインストール](#)

鍵リングの作成

WebSphere® eXtreme Scale WebSphere eXtreme Scale クライアント 製品オフリングをインストールするための応答ファイルを、Installation Manager を使用して記録した後、鍵リング・ファイルの作成を選択することができます。認証を必要とするリモート・リポジトリを使用している場合、インストールのための鍵リングを作成できます。

始める前に

応答ファイルを記録する必要があります。詳しくは、[応答ファイルの使用によるWebSphere eXtreme Scale クライアントのインストール](#)を参照してください。

手順

1. コマンド行で、Installation Manager をインストールしたディレクトリー の Eclipse サブディレクトリーに移動します。
2. -record オプションを使用して、コマンド行から Installation Manager を開始します。

例:

- **Windows** 管理者または非管理者:

```
IBMIM.exe -skipInstall "C:¥temp¥imRegistry"  
-keyring C:¥IM¥im.keyring  
-record C:¥temp¥keyring_response_file.xml
```

- **UNIX** | **Linux** 管理者:

```
./IBMIM -skipInstall /var/temp/imRegistry  
-keyring /var/IM/im.keyring  
-record /var/temp/keyring_response_file.xml
```

- **UNIX** | **Linux** 非管理者:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry  
-keyring user_home/var/IM/im.keyring  
-record user_home/var/temp/keyring_response_file.xml
```

3. 認証済みリモート・リポジトリを使用するための資格情報を要求するウィンドウが開いたら、正しい資格情報を入力して、その情報を保存します。
4. 「ファイル」 > 「終了」をクリックして、Installation Manager を閉じます。

詳しくは、[IBM® Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [応答ファイルの使用によるWebSphere eXtreme Scale クライアントのインストール](#)

IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール

IBM® Installation Manager を使用して WebSphere® eXtreme Scale クライアント 製品オフリングをアンインストールします。

[GUI の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

IBM Installation Manager のウィザード・コンソールを使用して WebSphere eXtreme Scale クライアント をアンインストールできます。

[コマンド行の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

コマンド行から IBM Installation Manager を使用して WebSphere eXtreme Scale クライアント をアンインストールできます。

[応答ファイルを使用した WebSphere eXtreme Scale クライアントのアンインストール](#)

IBM Installation Manager で応答ファイルを使用して WebSphere eXtreme Scale クライアント をアンインストールできます。

親トピック: [WebSphere eXtreme Scale クライアント のインストール](#)

GUI の使用による WebSphere eXtreme Scale クライアント のアンインストール

IBM® Installation Manager のウィザード・コンソールを使用して WebSphere® eXtreme Scale クライアント をアンインストールできます。

始める前に

WebSphere eXtreme Scale をアンインストールする前に、すべての WebSphere Application Server プロファイルから WebSphere eXtreme Scale 拡張を削除する必要があります。WebSphere eXtreme Scale をアンインストールすると、拡張解除を実行できなくなります。manageprofiles コマンドを使用して、WebSphere eXtreme Scale 環境内にある既存のプロファイルを拡張解除します。

手順

1. 製品をアンインストールします。
 - a. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
 - b. Installation Manager を開始します。
 - c. 「アンインストール」をクリックします。
 - d. 「パッケージのアンインストール」ウィンドウで、以下のアクションを実行します。
 - i. 以下のいずれかと適切なバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 6
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
 - ii. 「次へ」をクリックします。
 - e. アンインストール・ウィザードで拡張された WebSphere Application Server プロファイルのリストが表示された場合は、アンインストールを続行するために、それらのプロファイルを拡張解除する必要があります。
 - f. サマリー情報を検討します。
 - g. 「アンインストール」をクリックします。
 - アンインストールが正常に行われた場合には、正常に終了したことを示すメッセージがプログラムによって表示されます。
 - アンインストールが正常に行われなかった場合には、「ログの表示」をクリックして、問題をトラブルシューティングします。
 - h. 「終了」をクリックします。
 - i. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。
2. オプション: IBM Installation Manager をアンインストールします。

重要: IBM Installation Manager をアンインストールする前に、Installation Manager によってインストールされたすべてのパッケージをアンインストールする必要があります。

この手順の実行については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

関連タスク:

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

コマンド行の使用による **WebSphere eXtreme Scale** クライアント のアンインストール

コマンド行から IBM® Installation Manager を使用して WebSphere® eXtreme Scale クライアント をアンインストールできます。

始める前に

WebSphere eXtreme Scale をアンインストールする前に、すべての WebSphere Application Server プロファイルから WebSphere eXtreme Scale 拡張を削除する必要があります。WebSphere eXtreme Scale をアンインストールすると、拡張解除を実行できなくなります。manageprofiles コマンドを使用して、WebSphere eXtreme Scale 環境内にある既存のプロファイルを拡張解除します。

手順

1. システムにログオンします。
2. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
3. Installation Manager をインストールしたディレクトリーの eclipse/tools サブディレクトリーに移動します。
4. **imcl** コマンドを使用して、この製品をアンインストールします。

Windows

```
imcl.exe uninstall com.ibm.websphere.v85,optional_feature_ID  
-installationDirectory installation_directory
```

UNIX Linux

```
./imcl uninstall com.ibm.websphere.v85,optional_feature_ID  
-installationDirectory installation_directory
```

ヒント:

- *offering_ID* は、[WebSphere eXtreme Scale クライアント 製品のオファリング ID](#) にリストされるオファリング ID です。
- コンマとフィーチャー ID で区切られたフィーチャーのリストを削除できます。以下に例を示します。

```
imcl uninstall com.ibm.websphere.WXS.v85,xs.console.feature,xs.samples.feature
```

- client スタンドアロン・クライアントのフィーチャーを示します
- server スタンドアロン・サーバーのフィーチャーを示します
- console Web ベースのモニター・コンソールを示します
- samples はサンプルを示します。

- フィーチャーのリストが指定されていない場合は、製品全体がアンインストールされます。

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

5. アンインストール・プロセスで拡張された WebSphere Application Server プロファイルのリストが表示された場合は、アンインストールを続行するために、それらのプロファイルを拡張解除する必要があります。
6. オプション: IBM Installation Manager をアンインストールします。

重要: IBM Installation Manager をアンインストールする前に、Installation Manager によってインストールされたすべてのパッケージをアンインストールする必要があります。

アンインストール・スクリプトを使用してこの手順を実行する方法について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

関連タスク:

[コマンド行の使用による WebSphere eXtreme Scale クライアント のインストール](#)

応答ファイルを使用した **WebSphere eXtreme Scale** クライアントのアンインストール

IBM® Installation Manager で応答ファイルを使用して WebSphere® eXtreme Scale クライアント をアンインストールできます。

始める前に

WebSphere eXtreme Scale をアンインストールする前に、すべての WebSphere Application Server プロファイルから WebSphere eXtreme Scale 拡張を削除する必要があります。WebSphere eXtreme Scale をアンインストールすると、拡張解除を実行できなくなります。manageprofiles コマンドを使用して、WebSphere eXtreme Scale 環境内にある既存のプロファイルを拡張解除します。

オプション: Installation Manager のインストールおよび製品のインストールを実行して (あるいは記録のみを行って)、いずれかのシステムの一時インストール・レジストリーに記録します。こうすることにより、Installation Manager がインストールされている標準レジストリーを使用せずに、この一時レジストリーを使用して、アンインストールを記録することができます。

このタスクについて

Installation Manager を使用して、応答ファイルを処理してさまざまな方法で製品のアンインストールを行うことができます。次の手順で説明するように GUI を使用して応答ファイルを記録することができます。あるいは、手動で、またはサンプルを入手して変更することによって、新しい応答ファイルを生成することもできます。

手順

- ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
- オプション: 製品をアンインストールする応答ファイルの記録: いずれかのシステムで、以下のアクションを実行して、製品をアンインストールする応答ファイルを記録します。
 - コマンド行で、Installation Manager をインストールしたディレクトリーの Eclipse サブディレクトリーに移動します。
 - record オプションを使用して、コマンド行から Installation Manager を開始します。

例:

- **Windows** 管理者または非管理者:

```
IBMIM.exe -skipInstall "C:¥temp¥imRegistry"  
-record C:¥temp¥uninstall_response_file.xml
```

- **UNIX** | **Linux** 管理者:

```
./IBMIM -skipInstall /var/temp/imRegistry  
-record /var/temp/uninstall_response_file.xml
```

- **UNIX** | **Linux** 非管理者:

```
./IBMIM -skipInstall user_home/var/temp/imRegistry  
-record user_home/var/temp/uninstall_response_file.xml
```

ヒント: 「始める前に」の説明に従って作成した一時インストール・レジストリーで -skipInstall パラメーターの使用を選択すると、Installation Manager は応答ファイルを記録する際に一時インストール・レジストリーを使用します。-skipInstall パラメーターが指定されていると、いずれの製品パッケージもインストールまたはアンインストールされないことにご注意ください。Installation Manager で行うすべてのアクションは、指定の一時レジストリーに保管されているインストール・データを更新するのみです。応答ファイルの生成後に、この応答ファイルを使用して、製品ファイルを削除して標準インストール・レジストリーを更新し、製品をアンインストールできます。

-skipInstall の操作は、Installation Manager が使用する実際のエージェント・データ・ロケーションで使用することはできません。これはサポートされていません。空の書き込み可能なロケーションを使用し、今後の記録セッションにはそのロケーションを再使用してください。

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

- c. 「アンインストール」をクリックします。
- d. 「パッケージのアンインストール」ウィンドウで、以下のアクションを実行します。
 - i. 以下のいずれかと適切なバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 6
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
 - ii. 「次へ」をクリックします。
 - iii. 「次へ」をクリックします。
- e. サマリー情報を検討します。
- f. 「アンインストール」をクリックします。
 - アンインストールが正常に行われた場合には、正常に終了したことを示すメッセージがプログラムによって表示されます。
 - アンインストールが正常に行われなかった場合には、「ログの表示」をクリックして、問題をトラブルシューティングします。
- g. 「終了」をクリックします。
- h. 「ファイル」 > 「終了」をクリックして、Installation Manager を閉じます。

3. 応答ファイルを使用した製品のアンインストール: 製品をアンインストールする各システムのコマンド行で、Installation Manager をインストールしたディレクトリーの eclipse/tools サブディレクトリーに移動し、作成した応答ファイルを使用して製品をアンインストールします。

例:

- o **Windows** 管理者または非管理者:

```
imcl.exe
input C:\temp\uninstall_response_file.xml
-log C:\temp\uninstall_log.xml
```

- o **UNIX Linux** 管理者:

```
./imcl
input /var/temp/uninstall_response_file.xml
-log /var/temp/uninstall_log.xml
```

- o **UNIX Linux** 非管理者:

```
./imcl
input user_home/var/temp/uninstall_response_file.xml
-log user_home/var/temp/uninstall_log.xml
```

詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

4. オプション: すべてのインストール済みパッケージをリストして、アンインストールを検証します。

UNIX Linux

```
./imcl listInstalledPackages
```

Windows

```
imcl listInstalledPackages
```

- 5. アンインストール・プロセスで拡張された WebSphere Application Server プロファイルのリストが表示された場合は、アンインストールを続行するために、それらのプロファイルを拡張解除する必要があります。
- 6. オプション: IBM Installation Manager をアンインストールします。

重要: IBM Installation Manager をアンインストールする前に、Installation Manager によってインストールされたすべてのパッケージをアンインストールする必要があります。

アンインストール・スクリプトを使用してこの手順を実行する方法については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

親トピック: [IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)

関連タスク:

[応答ファイルの使用による WebSphere eXtreme Scale クライアント のインストール](#)

WebSphere eXtreme Scale Client for .NET のインストールの概要

WebSphere® eXtreme Scale Client for .NET は、ランタイム環境でインストールすることもできれば、ランタイム環境と開発環境の両方でインストールすることもできます。

.NET アプリケーションをビルドしてテストするには、WebSphere eXtreme Scale Client for .NET を開発環境でインストールします。開発環境のインストールには、必ずランタイム環境のインストールが含まれます。ランタイム・アセンブリーは、ディスクおよびグローバル・アセンブリー・キャッシュ (GAC) にインストールされます。開発環境のインストールでは、さらに、サンプル・コード、Visual Studio IntelliSense 機能の統合 (吹き出しクラスとメソッドの説明用)、および API の資料がインストールされます。サンプルの WebSphere eXtreme Scale Client for .NET ソース・コードおよび Visual Studio プロジェクトは [net_client_home¥sample](#) ディレクトリーにインストールされます。API 資料は [net_client_home¥doc](#) ディレクトリーにインストールされます。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET のインストール](#)

このフレームワークで実行されるアプリケーションがある場合は、WebSphere eXtreme Scale クライアントを .NET 環境でインストールすることができます。

.NET 2.5+ [サイレント・モードでの WebSphere eXtreme Scale クライアント for .NET のインストール](#)

WebSphere eXtreme Scale クライアントを .NET 環境でサイレント・モードでインストールすることができます。この方法は、インストールを自動で行いたい場合や、製品を複数のマシンにインストールする必要がある場合に使用されます。サイレント・モードのインストールでは、まず最初に、応答ファイルを記録し、このファイルにパラメーターを渡す必要があります。

.NET 2.5+ [WebSphere eXtreme Scale クライアント for .NET のアンインストール](#)

WebSphere eXtreme Scale クライアント for .NET を環境から削除するには、Windows コントロールパネルからこれをアンインストールすることができます。また、サイレント・モードでアンインストールする場合は応答ファイルを記録することができます。複数の WebSphere eXtreme Scale クライアントをインストールしており、それらのインストール済み環境を速やかに削除する必要がある場合は、応答ファイルを記録しておくことが最適な方法です。

.NET 2.5+ [インストール・プログラムによらない WebSphere eXtreme Scale Client for .NET のインストール](#)

setup.exe ファイルを実行する許可がない場合は、既存のインストール済み環境から別の Windows システムにファイルをコピーすることができます。

親トピック: [WebSphere eXtreme Scale クライアントのインストール](#)

WebSphere eXtreme Scale Client for .NET のインストール

このフレームワークで実行されるアプリケーションがある場合は、WebSphere® eXtreme Scale クライアントを .NET 環境でインストールすることができます。

始める前に

- DVD から WebSphere eXtreme Scale Client for .NET を取得します。ルート・ディレクトリー内で setup.exe ファイル、すなわち、 /ClientForDotNet/setup.exe を見付けるか、または、 [サポート・サイト](#) からダウンロードすることができます。
- WebSphere eXtreme Scale Client for .NET を開発環境にインストールする場合は、ハードウェア要件およびソフトウェア要件を満たす Windows システムを使用している必要があります。詳しくは、 [Microsoft .NET に関する考慮事項](#) を参照してください。

このタスクについて

WebSphere eXtreme Scale Client for .NET は、ランタイム環境でインストールするか、またはランタイム環境と開発環境の両方でインストールすることができます。

手順

1. ウィザードを使用して、DVD からクライアントをインストールします。ルート・ディレクトリー内で setup.exe、すなわち、 /ClientForDotNet/setup.exe を見付けるか、または、 [サポート・サイト](#) からダウンロードすることができます。
2. setup.exe ファイルを実行します。
3. ウィザードのプロンプトに従って進み、「次へ」をクリックします。「セットアップ・タイプ」ページが表示されます。
4. WebSphere eXtreme Scale Client for .NET をランタイム環境にインストールすることに決めていた場合は、「ランタイム」をクリックします。次のステップを実行します。

「インストール」をクリックしてインストーラーを実行し、「終了」をクリックします。デフォルトのインストール・ディレクトリーは C:\Program Files (x86)\IBM\WebSphere\eXtreme Scale .NET Client です。
5. WebSphere eXtreme Scale クライアント をランタイム環境と開発環境の両方にインストールすることに決めていた場合は、「カスタム」を選択し、以下のステップを実行します。
 - a. WebSphere eXtreme Scale クライアント をデフォルトのインストール・ディレクトリーにインストールするか、またはインストール・ディレクトリーを選択し、「次へ」をクリックします。
 - b. デフォルトでは、ランタイム環境と開発環境の両方がフィーチャーとして選択されます。両方をインストールする場合は、十分なディスク・スペースがあることを確認してください。「次へ」をクリックします。
 - c. ログ・ファイルの場所を選択し、「次へ」をクリックします。
 - d. 「インストール」をクリックしてインストーラーを実行し、「終了」をクリックします。

次のタスク

- 入門アプリケーションを実行して、ご使用の WebSphere eXtreme Scale Client for .NET をテストします。詳しくは、 [チュートリアル: 単純データ・グリッド・アプリケーションの入門](#) を参照してください。
- WebSphere eXtreme Scale Client for .NET を構成します。詳しくは、 [WebSphere eXtreme Scale Client for .NET の構成](#) を参照してください。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

[2.5+ WebSphere eXtreme Scale クライアント for .NET のアンインストール](#)

[.NET 2.5+ インストール・プログラムによらない WebSphere eXtreme Scale Client for .NET のインストール](#)

サイレント・モードでの WebSphere eXtreme Scale クライアント for .NET のインストール

WebSphere® eXtreme Scale クライアント を .NET 環境でサイレント・モードでインストールすることができます。この方法は、インストールを自動で行いたい場合や、製品を複数のマシンにインストールする必要がある場合に使用されます。サイレント・モードのインストールでは、まず最初に、応答ファイルを記録し、このファイルにパラメーターを渡す必要があります。

始める前に

- DVD から WebSphere eXtreme Scale クライアント を取得します。ルート・ディレクトリー内で `setup.exe`、すなわち、`/net_client_home/setup.exe` を見付けるか、または、[サポート・サイト](#) からダウンロードすることができます。
- WebSphere eXtreme Scale クライアント for .NET を開発環境にインストールする場合は、Microsoft .NET の考慮事項に記載されているハードウェア要件およびソフトウェア要件を満たす Windows システムを使用している必要があります。詳しくは、[Microsoft .NET に関する考慮事項](#) を参照してください。

このタスクについて

WebSphere eXtreme Scale クライアント for .NET は、ランタイム環境でインストールするか、またはランタイム環境と開発環境の両方でインストールすることができます。

手順

1. コマンド・プロンプトを開き、次のスクリプトを実行します: `setup.exe /r /f1"<Response_Files_Directory>%Setup.iss"` <Response_Files_Directory> は応答ファイルを作成する場所です。
2. ウィザードのプロンプトに従って進み、「次へ」をクリックします。「セットアップ・タイプ」ページが表示されます。
3. 選択するオプションに応じて、Setup.iss 応答ファイルを作成するために次の値を渡すことができます。
 - WebSphere eXtreme Scale クライアント をランタイム環境でインストールすることを選択するか、またはカスタム・インストールを選択します。カスタム・インストールでは、製品を両方の環境でインストールすることができます。
 - WebSphere eXtreme Scale クライアント をランタイム環境にインストールすることに決めていた場合は、「ランタイム」をクリックし、以下のことを行います。
 - a. 「インストール」をクリックして、「終了」をクリックします。デフォルトのインストール・ディレクトリーは `C:%Program Files (x86)%IBM%WebSphere%eXtreme Scale .NET Client` です。
 - WebSphere eXtreme Scale クライアント をランタイム環境と開発環境の両方にインストールすることに決めていた場合は、「カスタム」を選択し、以下のことを行います。
 - a. WebSphere eXtreme Scale クライアント をデフォルトのインストール・ディレクトリーにインストールするか、またはインストール・ディレクトリーを選択します。「次へ」をクリックします。
 - b. ランタイム環境と開発環境の両方を選択します。両環境でインストールしたい場合は、十分なディスク・スペースがあることを確認してください。「次へ」をクリックします。
 - c. ログ・ファイルの場所を選択し、「次へ」をクリックします。
4. 「インストール」をクリックして、「終了」をクリックします。
5. コマンド・プロンプトを開き、スクリプト `setup.exe /s /f1"<Response_Files_Directory>%Setup.iss"` を実行して、サイレント・モードで WebSphere eXtreme Scale クライアント をインストールします。ここで、<Response_Files_Directory> は、応答ファイルが置かれる場所です。

次のタスク

有効な eXtreme Scale Client for .NET API を試みるように SimpleClient を更新または変更することができます。インストール・ディレクトリーにある `<installation_directory>%sample%SimpleClient` で、SimpleClient を見付けて、このファイルを Visual Studio にロードすると、単純な作成、検索、更新、および削除操作を使用するサンプル・アプリケーションを表示することができます。SimpleClient をデータ・グリッドへのアクセス・ガイドとして使用します。このアプリケーションを変更することもできれば、eXtreme Scale for .NET クライアントがサポートする API セットを使用する新しいアプリケーションを作成することもできます。詳しくは、[チュートリアル: 単純データ・グリッド・アプリケーションの入門](#) を参照してください。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

[2.5+ WebSphere eXtreme Scale クライアント for .NET のアンインストール](#)

WebSphere eXtreme Scale クライアント for .NET のアンインストール

WebSphere® eXtreme Scale クライアント for .NET を環境から削除するには、Windows コントロール パネルからこれをアンインストールすることができます。また、サイレント・モードでアンインストールする場合は応答ファイルを記録することができます。複数の WebSphere eXtreme Scale クライアント をインストールしており、それらのインストール済み環境を速やかに削除する必要がある場合は、応答ファイルを記録しておくことが最適な方法です。

始める前に

開発環境から製品をアンインストールする場合は、必ず Visual Studio をシャットダウンしてください。

重要: アンインストーラーは、すべてのバイナリー・ファイルと、フィックスパックやインテリム・フィックスなどのすべての保守を同時に削除します。

手順

1. すべての .NET eXtreme Scale プロセスを停止します。
2. 以下のいずれかの方法を使用して、WebSphere eXtreme Scale クライアント for .NET をアンインストールすることができます。
 - Windows の「コントロール パネル」から「プログラムの追加と削除」をクリックして、「IBM WebSphere eXtreme Scale クライアント for .NET」を選択してアンインストールします。
 - 応答ファイルを記録する必要がある場合は、コマンド・プロンプトを開き、次のスクリプトを実行します。

```
setup.exe /uninst /r /f1"<Response_Files_Directory>%Setup.iss"
```

- a. アンインストール・ウィザードが開き、確認ウィンドウが表示されます。WebSphere eXtreme Scale クライアント for .NET とそのすべての機能を削除することを確認してください。「OK」をクリックします。
 - b. アンインストール・プロセスが完了したら、「終了」をクリックします。
3. オプション: 応答ファイルを使用して WebSphere eXtreme Scale クライアント の既存のインストール済み環境をアンインストールする場合は、次のようにして、記録しておいた応答ファイルを実行します。
 - a. コマンド・プロンプトを開き、次のスクリプトを実行して WebSphere eXtreme Scale クライアント for .NET をサイレント・モードでアンインストールします。

```
setup.exe /uninst /s /f1"<Response_Files_Directory>%Setup.iss"
```

次のタスク

Windows Explorer で、すべてのフォルダーがインストール・ディレクトリーから削除されたことを確認します。また、Windows コントロール パネルでも、製品がリストされていないことを確認してください。インストーラーは、インストール後に生成されたファイル (例えば、ログ・ファイル、カスタム構成ファイル、SimpleClient サンプルのビルドで作成された任意の成果物) を含むフォルダーを削除しません。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

[.NET 2.5+ WebSphere eXtreme Scale Client for .NET のインストール](#)

[.NET 2.5+ サイレント・モードでの WebSphere eXtreme Scale クライアント for .NET のインストール](#)

インストール・プログラムによらない **WebSphere eXtreme Scale Client for .NET** のインストール

2.5+ setup.exe ファイルを実行する許可がない場合は、既存のインストール済み環境から別の Windows システムにファイルをコピーすることができます。

始める前に

ファイルのコピー元となる WebSphere® eXtreme Scale Client for .NET の完全なインストール済み環境がなければなりません。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストールの概要](#)を参照してください。

このタスクについて

ご使用のクライアント・サーバー上で setup.exe ファイルを実行する許可がない場合は、既存の WebSphere eXtreme Scale Client for .NET からご使用の .NET アプリケーションのランタイム・ディレクトリーにファイルをコピーすることができます。

警告: 自動アップグレードおよび自動マイグレーションは、WebSphere eXtreme Scale Client for .NET インストール・プログラムを使用した場合にのみ使用可能です。以下の手順を使用して製品をインストールする場合は、手動アップグレードおよび手動マイグレーションが必要です。

手順

- Windows** ご使用の WebSphere eXtreme Scale Client for .NET インストール済み環境から、以下のファイルをターゲットの Windows システムにコピーします。ターゲットの Windows システム・ディレクトリーは、ご使用のデータ・グリッドの .NET アプリケーションを実行するプロセスのランタイム・ディレクトリーでなければなりません。このプロセスは、ターゲット・ディレクトリーに対する読み取りおよび書き込み権限を持っていないければなりません。ログ・ファイルは、このディレクトリーの下にある logs フォルダーに生成されます。
 - [net_client_home](#)\bin\IBM.WebSphere.Caching.dll
 - [net_client_home](#)\bin\IBM.WebSphere.Caching.CredentialGenerator.dll (提供された資格情報生成プログラムを使用するように WebSphere eXtreme Scale Client for .NET アプリケーションが構成された場合にのみ必要です。)
 - [net_client_home](#)\config\Client.Net.properties
 - [net_client_home](#)\config\Client.Net.Log.config
- オプション: 手動で IBM.WebSphere.Caching.dll アセンブリーをグローバル・アセンブリー・キャッシュ (GAC) にインストールします。GAC をインストールする正確な手順は、Windows 環境によって異なります。

次のタスク

WebSphere eXtreme Scale Client for .NET を構成します。詳しくは、[WebSphere eXtreme Scale Client for .NET の構成](#)を参照してください。

親トピック: **.NET** **2.5+** [WebSphere eXtreme Scale Client for .NET のインストールの概要](#)

関連タスク:

.NET **2.5+** [WebSphere eXtreme Scale Client for .NET のインストール](#)

Liberty プロファイル のインストール

Installation Manager を使用するか、Java™ アーカイブ (JAR) ファイルを実行することで、Liberty プロファイル・アプリケーション・サービス環境をインストールします。

このタスクについて

Liberty プロファイルをインストールするには、WebSphere® eXtreme Scale クライアント for WebSphere Application Server バージョン 8.5 以上をインストールすることができます。または、提供された JAR ファイルを実行することによって Liberty プロファイルをインストールすることができます。

z/OS® オペレーティング・システムでは、Liberty プロファイル は運用環境を提供します。この環境に関するネイティブでの作業は、MVS™ コンソールを使用して行うことができます。アプリケーション開発の場合は、別の分散システム、Mac OS、または z/OS 上の Linux シェルで Eclipse ベースの開発者ツールを使用することを検討してください。

手順

- 以下のいずれかのオプションを使用して、WebSphere Application Server Liberty プロファイル をインストールします。
 - WebSphere DataPower® XC10 アプライアンス では、WebSphere eXtreme Scale クライアント をインストールする必要があります。このトポロジーで Liberty プロファイル を実行するためには、[Java アーカイブ \(JAR\) ファイルを実行することによって Liberty プロファイル・アプリケーション・サービス環境をインストール](#)する必要があります。
- WebSphere eXtreme Scale クライアント バージョン 8.5 以上をインストールします。このオプションを使用して製品をインストールすると、自動的に Liberty プロファイルがインストールされます。
- ご使用の Liberty プロファイル環境をマイグレーションします。Liberty プロファイルをあるメジャー・リリースからより高いメジャー・リリースにマイグレーションするときは、Liberty プロファイルの server.xml ファイルでフィーチャー・バージョン番号を変更する必要があります。

例えば、製品がサポートしていた Liberty プロファイル の初期バージョンでは、フィーチャー・バージョン番号は 1.0 レベルでした。WebSphere eXtreme Scale クライアント バージョン 8.6 以上では、フィーチャーバージョン番号は 1.1 レベルです。

注: WebSphere Application Server V8.5.5 にアップグレードすると、Liberty プロファイルは削除されます。WebSphere eXtreme Scale はこのフィーチャーの削除をサポートします。ただし、WebSphere Application Server V8.5 にロールバックすると、再び Liberty プロファイルが追加されます。これは、WebSphere eXtreme Scale で問題を引き起こします。

[データ・キャッシングと Liberty プロファイル](#)

データ・キャッシング製品では、WebSphere Application Server Liberty プロファイルを使用して、HTTP セッション (REST ゲートウェイを介したクライアント/サーバー接続) を開発したり、その他のキャッシュ統合シナリオを管理したりすることができます。

[JAR ファイルの実行による Liberty プロファイルのアプリケーションのサービス環境のインストール](#)

配布イメージが入った Java アーカイブ (JAR) ファイルを実行することで、アプリケーション・サービス環境をインストールし、Liberty サーバーを作成する準備ができます。

親トピック: [WebSphere DataPower XC10 アプライアンス のインストール](#)

関連タスク:

[GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)

[コマンド行の使用による WebSphere eXtreme Scale クライアント のインストール](#)

[応答ファイルの使用による WebSphere eXtreme Scale クライアント のインストール](#)

データ・キャッシングと Liberty プロファイル

Java データ・キャッシング製品では、WebSphere® Application Server Liberty プロファイルを使用して、HTTP セッション (REST ゲートウェイを介したクライアント/サーバー接続) を開発したり、その他のキャッシュ統合シナリオを管理したりすることができます。

WebSphere DataPower® XC10 アプライアンスでは、Liberty プロファイルを使用してアプライアンス内のデータ・グリッドに接続することができます。例えば、WebSphere eXtreme Scale クライアントを Liberty プロファイルと共にインストールすると、HTTP セッション・アプリケーション、Java クライアント・アプリケーション、および REST クライアント・アプリケーション (これらのアプリケーションは Liberty プロファイルにインストールされます) を管理するために使用できるフィーチャーへのアクセスが可能になります。

以下のフィーチャーには、主要な使用可能フィーチャーに関する情報が記載されています。構成にフィーチャーを組み込むと、さらに 1 つ以上のフィーチャーが自動的にロードされることがあります。各フィーチャーには、要旨とそのフィーチャーの宣言例が含まれています。


クライアント・フィーチャー

クライアント・フィーチャーには、eXtreme Scale のプログラミング・モデルの大部分が含まれています。Liberty プロファイルで実行され、eXtreme Scale API を使用するアプリケーションがあるときは、クライアント・フィーチャーを追加してください。

`wlp_install_root/usr/server/wlp_install_root/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale Client">
  <featureManager>
    <feature>eXtremeScale.client-1.1</feature>
  </featureManager>
</server>
```

Web フィーチャー

 Web フィーチャーは推奨されません。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを使用してください。

Web フィーチャーには、Liberty プロファイル Web アプリケーションを拡張する機能が含まれています。フォールト・トレランスのために HTTP セッション・データを複製するときは、Web フィーチャーを追加してください。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">
  <featureManager>
    <feature>eXtremeScale.web-1.1</feature>
  </featureManager>
  <xsWebAppV85/>
</server>
```

WebApp フィーチャー

webApp フィーチャーには、Liberty プロファイル Web アプリケーションを拡張する機能が含まれています。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを追加してください。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">
  <featureManager>
    <feature>eXtremeScale.webApp-1.1</feature>
  </featureManager>
  <xsWebApp/>
</server>
```

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

<featureManager>
<feature>eXtremeScale.webGrid-1.1</feature>
</featureManager>

<xsWebGrid/>
</server>
```

REST フィーチャー

Representational State Transfer (REST) ゲートウェイを使用して、Liberty プロファイル内の集合がホストする単純データ・グリッドにアクセスします。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

<featureManager>
<feature>eXtremeScale.rest-1.1</feature>
</featureManager>

<xsRest/>
</server>
```

動的キャッシュ・フィーチャー

Liberty プロファイル・サーバーは、動的キャッシュが使用可能になっているアプリケーションのデータをキャッシュするデータ・グリッドをホストすることができます。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

  <featureManager>
    <feature>eXtremeScale.dynacacheGrid-1.1</feature>
  </featureManager>

  <xsDynacacheGrid/>
</server>
```

2.5+

動的キャッシュ・アプリケーション・フィーチャー

Liberty プロファイル・サーバーは、このフィーチャーを使用して動的キャッシュ・プロバイダーとして構成できる WebSphere eXtreme Scale をホストすることができます。

Liberty プロファイルにおけるデフォルト・キャッシュ・プロバイダーの使用については次の例を参照してください。

`wlp_install_root/usr/server/server_name/server.xml` ファイル

```
<server description="WebSphere eXtreme Scale enabled Web Server">

<featureManager>
<feature>eXtremeScale.dynacacheapp-1.1</feature>
</featureManager>

<xsClientDomain default="production">
<endpointConfig> production;localhost:2809 </endpointConfig>
</xsClientDomain>

<distributedMap id="baseCache" libraryRef="idgenerator" cacheProviderName="WebSphere
eXtreme Scale">
<xsDynacacheApp remoteDomain="production" />
</distributedMap>

<distributedMap id="cache01" jndiName="cache01" memorySizeInEntries="2000"
createCacheAtServerStartup="true" cacheProviderName="WebSphere eXtreme Scale">
```

```
<xsDynacacheApp remoteDomain="production" />  
</distributedMap>  
</server>
```

親トピック: [Liberty プロファイルのインストール](#)

JAR ファイルの実行による Liberty プロファイルのアプリケーションのサービス環境のインストール

配布イメージが入った Java アーカイブ (JAR) ファイルを実行することで、アプリケーション・サービス環境をインストールし、Liberty サーバーを作成する準備ができます。

このタスクについて

このトピックで説明しているように JAR ファイルを実行するか、Installation Manager を使用することで、Liberty プロファイル・アプリケーション・サービス環境をインストールできます。

WebSphere® Application Server 用の JAR ファイルを実行して Liberty プロファイルをインストールする場合は、まずその JAR ファイルを解凍する必要があります。次に、WebSphere eXtreme Scale 用の Liberty プロファイル JAR ファイルを解凍します。IBM Installation Manager を使用して WebSphere Application Server バージョン 8.5 をインストールし、Liberty プロファイルを取得する場合は、Installation Manager を使用して WebSphere eXtreme Scale をインストールする必要もあります。

このタスクは以下のエディションをサポートします。

- WebSphere Application Server Liberty Core
- WebSphere Application Server, Base および Developer エディション
- WebSphere Application Server, Network Deployment
- WebSphere Application Server for z/OS

Liberty プロファイル・アプリケーション・サービス環境およびデータ・キャッシング環境のダウンロードについては、[WASdev コミュニティーのダウンロード・ページ](#)を参照してください。

手順

1. WebSphere Application Server Liberty プロファイル 配布イメージを任意のディレクトリーに解凍します。

このイメージは JAR ファイルとしてパッケージされています。例: `wlp-edition-8.6.0.0.jar`。次のいずれかのアクションを使用してこの JAR ファイルを解凍します。

- ウィザードを使用して配布イメージを解凍するには、`java -jar wlp-edition-8.6.0.0.jar` を実行します。
- ライセンス条件をサイレントに受け入れて配布イメージを解凍するには、`java -jar wlp-edition-8.6.0.0.jar -acceptLicense` を実行します。
- 使用可能なすべてのオプションを表示するには、`java -jar wlp-edition-8.6.0.0.jar -help` を実行します。

アプリケーション・サーバー・ファイルはすべて、`wlp` ディレクトリーのサブディレクトリーに格納されています。

2. オプション: ご使用の環境に `JAVA_HOME` プロパティを設定します。

Liberty プロファイルを使用するには、その実行場所となる JRE が必要です。WebSphere Application Server の完全プロファイルが使用する JDK または JRE を共有しません。『[Liberty プロファイル環境のカスタマイズ](#)』で説明されているように、`server.env` ファイル内の `JAVA_HOME` プロパティを使用して JDK または JRE の場所を指定できます。Linux システムまたは UNIX システムでは、代わりに、ユーザーの `.bashrc` ファイル内の `JAVA_HOME` を設定するか、`PATH` 環境変数に JDK または JRE のパスを追加することができます。Windows システムでは、代わりに、`JAVA_HOME` をシステム環境変数として設定するか、`PATH` システム変数に JDK または JRE のパスを追加することができます。

Windows 例えば、Windows システムでは、以下のコマンドを使用して、`JAVA_HOME` プロパティを設定し、パスに `Java™ /bin` ディレクトリーを追加することができます。

```
set JAVA_HOME=C:¥Progra~1¥Java¥JDK16
set PATH=%JAVA_HOME%¥bin;%PATH%
```

注:

- Liberty プロファイルのランタイム環境は、`JAVA_HOME` プロパティ、`JRE_HOME` プロパティ、システム `PATH` プロパティの順に `java` コマンドを検索します。
- サポートされる Java 環境およびその取得場所について詳しくは、『[Liberty プロファイル: ランタイム環境での既知の制約事項](#)』の『[Java の最小サポート・レベル](#)』を参照してください。

3. WebSphere eXtreme Scale 配布イメージを、`wlp-edition-8.6.0.0.jar` ファイルを解凍したディレクトリーに解凍します。

このイメージは、V8.6 で `wxs-wlp_8.6.0.0.jar` と呼ばれ、V8.6 Fix Pack 2 で `wxs-wlp_8.6.0.2.jar` と呼ばれる JAR ファイルとしてパッケージされています。配布イメージを解凍するには、JAR ファイルを実行します。例えば、eXtreme Scale のバージョンに応じて次のコマンドを実行します。

```
java -jar wxs-wlp_8.6.0.0.jar -acceptLicense
```

```
java -jar wxs-wlp_8.6.0.2.jar -acceptLicense
```

タスクの結果

ファイル `wxs-wlp_8.6.0.0.jar` ファイル (V8.6) および `wxs-wlp_8.6.0.2.jar` ファイル (V8.6 Fix Pack 2) を解凍する場合、両方の JAR ファイルを同じディレクトリーに解凍すると、eXtreme Scale は WebSphere Application Server Liberty プロファイル上にインストールされます。

親トピック: [Liberty プロファイルのインストール](#)

製品インストールのトラブルシューティング

IBM® Installation Manager は多くの IBM ソフトウェア製品のための共通インストーラーです。このバージョンの WebSphere® eXtreme Scale をインストールするには、このインストーラーを使用します。

タスクの結果

ロギングおよびトレースに関する注記:

- Installation Manager を開いて「ファイル」>「ログの表示」をクリックすると、ログを簡単に表示できます。表内のログ・ファイルを個々に選択して、「ログ・ファイルを開く」アイコンをクリックすることにより、個々のログ・ファイルを開くことができます。
- ログは、Installation Manager のアプリケーション・データ・ロケーションの logs ディレクトリーにあります。次に例を示します。

- **Windows** 管理インストール:

```
C:\Documents and Settings\All Users\Application Data\IBM\Installation Manager
```

- **Windows** 非管理インストール:

```
C:\Documents and Settings\user_name\Application Data\IBM\Installation Manager
```

- **UNIX** | **Linux** 管理インストール:

```
/var/IBM/InstallationManager
```

- **UNIX** | **Linux** 非管理インストール:

```
user_home/var/ibm/InstallationManager
```

- メイン・ログ・ファイルは、タイム・スタンプが記されている XML ファイルで、logs ディレクトリーにあります。これらのログ・ファイルは、任意の標準 Web ブラウザーを使用して表示できます。
- logs ディレクトリー内の log.properties ファイルは、Installation Manager が使用するロギングまたはトレースのレベルを指定します。WebSphere eXtreme Scale プラグインのトレースをオンにするには、例えば、以下のような内容の log.properties ファイルを作成します。

```
com.ibm.ws=DEBUG
com.ibm.cic.agent.core.Engine=DEBUG
global=DEBUG
```

必要に応じて Installation Manager を再始動します。Installation Manager によって、WebSphere eXtreme Scale プラグインのトレースが出力されます。

トラブルシューティングに関する注記:

- **UNIX** | **Linux** デフォルトで、一部の HP-UX システムは、ホスト名の解決に DNS を使用しないように構成されます。その場合、Installation Manager は外部のリポジトリーに接続できない場合があります。

リポジトリーを ping することはできますが、nslookup は何も返しません。

システム管理者と相談して、DNS を使用するようにご使用のマシンを構成するか、またはリポジトリーの IP アドレスを使用してください。

- 場合によっては、Installation Manager の既存のチェック・メカニズムを迂回する必要がある場合もあります。
 - 一部のネットワーク・ファイル・システムでは、ディスク・スペースが正しく報告されない場合があります。この場合、ディスク・スペース・チェックを迂回して、インストールを続行する必要があります。

ディスク・スペース・チェックを使用不可にするには、`IM_install_root/eclipse/configuration` の `config.ini` ファイルで以下のシステム・プロパティーを指定して、Installation Manager を再始動します。

```
cic.override.disk.space=sizeunit
```

ここで `size` は正整数です。`unit` は、バイトの場合は空白、キロの場合は `k`、メガバイトの場合は `m`、ギガバイトの場合は `g` です。例:

```
cic.override.disk.space=120 (120 バイト)
cic.override.disk.space=130k (130 キロバイト)
cic.override.disk.space=140m (140 メガバイト)
cic.override.disk.space=150g (150 ギガバイト)
cic.override.disk.space=true
```

Installation Manager は Long.MAX_VALUE のディスク・スペース・サイズを報告します。使用可能なディスク・スペースが非常に大容量の場合は、表示されずに N/A が表示されます。

- オペレーティング・システムの前条件チェックを迂回するには、`disableOSPrereqChecking=true` を `IM_install_root/eclipse/configuration` の `config.ini` ファイルに追加して、Installation Manager を再始動します。

これらの迂回メソッドのいずれかを使用する必要がある場合は、Installation Manager チェック・メカニズムを迂回しないソリューションの開発における支援を受けるため、IBM サポートにお問い合わせください。

- Installation Manager の使用について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

Installation Manager の最新バージョンの詳細については、リリース情報を参照してください。リリース情報にアクセスするには、以下のタスクを実行します。

- **Windows** 「スタート」 > 「プログラム」 > 「IBM Installation Manager」 > 「リリース情報」をクリックします。
- **UNIX | Linux** Installation Manager がインストールされているディレクトリーにある文書サブディレクトリーに移動し、`readme.html` ファイルを開きます。
- 製品をインストールしようとしている時に致命的エラーが生じた場合は、以下の手順に従ってください。
 - IBM サポートが後で確認する必要がある場合があるので、現在の製品インストール・ディレクトリーのバックアップ・コピーを取ってください。
 - Installation Manager を使用して、製品のインストール・ロケーション (パッケージ・グループ) にインストールしたものをすべてアンインストールします。このときエラーが生じる可能性もありますが、無視しても支障ありません。
 - 製品のインストール・ディレクトリーに残っているものをすべて削除します。
 - Installation Manager を使用し、同じロケーションか新しいロケーションに製品を再インストールします。

バージョンおよび履歴情報に関する注記: **versionInfo** コマンドおよび **historyInfo** コマンドは、システム上で実行されるインストール、アンインストール、更新、および、ロールバックのすべてのアクティビティーに基づいて、バージョンと履歴情報を戻します。

親トピック: [WebSphere DataPower XC10 アプライアンスのインストール](#)

WebSphere DataPower XC10 アプライアンス の更新

ご使用の WebSphere® DataPower® XC10 アプライアンス 環境に保守を適用するには、まずファームウェア更新を適用してください。その後、ご使用の WebSphere eXtreme Scale クライアント インストール済み環境に更新を適用することができます。

1. [ファームウェアの更新](#)

IBM® WebSphere DataPower XC10 アプライアンス ソフトウェアのアップグレードは、ファームウェア更新を使用して行うことができます。Fix Central Web サイトから新しいファームウェア・バージョンをダウンロードして、アプライアンス・ソフトウェアを更新します。

2. [WebSphere Application Server 上の WebSphere eXtreme Scale クライアント の更新](#)

WebSphere Application Server を新しいバージョンにマイグレーションするとき、WebSphere eXtreme Scale クライアント の構成を新しい WebSphere Application Server のインストール済み環境にマイグレーションすることもできます。

3. [IBM Installation Manager の使用によるフィックスパックのインストール](#)

IBM Installation Manager を使用して、WebSphere eXtreme Scale クライアント 製品オフリングに使用可能なフィックスパックでこの製品を更新することができます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してインストールできます。

4. [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

IBM Installation Manager を使用して WebSphere eXtreme Scale クライアント 製品オフリングを前のバージョンにロールバックできます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してアンインストールできます。

5. **2.5+** [WebSphere eXtreme Scale Client for .NET のアップグレード](#)

インストール・プログラムを実行して、既存の WebSphere eXtreme Scale Client for .NET インストール済み環境をアップグレードすることができます。プログラムは既存のインストール済み環境を検出し、該当するファイルを置き換えます。

ファームウェアの更新

IBM® WebSphere® DataPower® XC10 アプライアンス ソフトウェアのアップグレードは、ファームウェア更新を使用して行うことができます。Fix Central Web サイトから新しいファームウェア・バージョンをダウンロードして、アプライアンス・ソフトウェアを更新します。

始める前に

アプライアンス管理権限が割り当てられている必要があります。

ファームウェア更新プロセスを開始する前に、すべてのアクティブ・ジョブが完了していることを確認してください。ファームウェア更新プロセスによって、実行中のアクティブ・ジョブが中断されます。これにより、データの不整合が発生する可能性があります。更新プロセスを開始する前に、必ずこの動作を想定した対応策を立てておいてください。

重要: 新しいアプライアンスに初めてファームウェアをインストールする場合は、ファームウェア更新が完了してから **clear-all** コマンドをアプライアンスで実行してください。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。

このタスクについて

IBM から新しいファームウェアの更新をダウンロードし、アプライアンスを新しいファームウェアで更新することによって、アプライアンスのファームウェアの更新を行うことができます。ユーザー・インターフェース、またはコマンド行でファームウェア更新を実行することができます。

注意:

2.0 以前のバージョンから、バージョン **2.5** にアップグレードするときは、現在ワークロードにサービスを提供している集合でファームウェア・アップグレードを実行することはできません。集合をアップグレードすると、データ・グリッドにロードされたデータがすべて失われます。集合内のアプライアンスでファームウェア・アップグレードを実行している場合、集合内のすべてのアプライアンスがアップグレードされるまで、アップグレードしたアプライアンスは、完全には再始動しません。まだアップグレードされていない残りのアプライアンスが要求を処理します。集合内では **1** つのアプライアンスでファームウェア・アップグレード・プロセスを完了してから、別のアプライアンスでプロセスを開始する必要があります。すべてのアプライアンスが同じファームウェア・レベルになるまで構成変更は行わないでください。

手順

- ファームウェア・アップグレードをダウンロードします。

[IBM Fix Central](#) にアクセスし、ご使用のローカル・ファイル・システムにファームウェア更新をダウンロードします。「Fix Central」ページの「製品グループ」で「**WebSphere**」を選択し、製品リストから「**WebSphere DataPower XC10 アプライアンス**」を選択します。ご使用のアプライアンス・ハードウェアのタイプに適したファームウェア・アップグレードをダウンロードします。

- アプライアンス・タイプ 7199-92X のアプライアンス・ファームウェアの更新は、.scrypt3 という拡張子の付いた単一のファイルになります。

この scrypt ファイルには、実行する更新の安全性を保証するために署名が付けられています。ファームウェア・アップグレードの実行にユーザー・インターフェースを使用している場合は、ユーザー・インターフェースへのアクセスに使用するコンピューターに、このファイルを保存します。ファームウェア・アップグレードの実行にコマンド行インターフェースを使用している場合、アプライアンスからアクセス可能なサーバーにこのファイルを保存します。

- ユーザー・インターフェースでファームウェア更新を実行するには、以下を行います。

1. ユーザー・インターフェースにログオンします。
2. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースで、「アプライアンス」 > 「設定」をクリックします。
 - 「ホーム」パネルで、「ステップ **1**: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
3. 「ファームウェア」を展開します。アプライアンスにインストールされているファームウェアのレベルが表示されます。
4. 新しいファームウェア更新をインストールします。
 - a. 「参照...」をクリックし、新しいファームウェア更新ファイルを選択します。
 - b. 新しいファームウェア更新ファイルを選択して、「**OK**」をクリックします。
 - c. 「アップグレード」をクリックします。ファームウェア更新のアップロードに要する時間は、接続速度によって異なります。アップロードが完了し、ファームウェア更新が開始されると、メッセージが表示されます。更新が開始されると、アプライアンスが再始動します。ユーザー・インターフェースでは、画面に進行状況は示されず、変更も表示されません。変更が表示されないのは、アプライアンスの再始動時にはセッションが終了し、アップグレード・プロセス中にユーザー・インターフェースが使用できないためです。

重要: アプライアンスは、ファームウェア・アップグレードのプロセス中に数回再始動します。ファームウェア・アップグレード中に、プロセスを中断したり手動でアプライアンスを再始動したりしないでください。

ファームウェア更新が完了したことを確認するには、更新が完了したと予想される頃に、もう一度ログインする必要があります。実際のファームウェア更新にかかる時間は、開始後、平均して約 10 分から 15 分ですが、これより長くかかる場合もあります。ファームウェア更新の進行状況をモニターしたい場合は、アプライアンス始動状況ユーザー・インターフェース・パネルを使用するか、またはコマンド行インターフェースで **start-progress** コマンドを使用してください。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。

- コマンド行インターフェースでファームウェア更新を実行するには、以下を行います。

1. コマンド行を使用して、またはシリアル・コンソールからアプライアンスに接続します。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
2. .script ファイルをアプライアンスに置きます。コマンド行インターフェースから、次のコマンドを実行します。

```
file get <url_to_script_file> <firmware_file>
```

ここで、*url_to_script_file* は Fix Central から .script ファイルを保存したローカル・サーバーで、*firmware_file* はアプライアンスで使用する .script ファイルの名前です。

3. ファームウェア更新をインストールします。以下のコマンドを実行します。

```
firmware upgrade <firmware_file>
```

重要: ファームウェア・アップグレードのプロセス中、アプライアンスは数回再始動します。ファームウェア・アップグレード中に、プロセスを中断したり手動でアプライアンスを再始動したりしないでください。

4. アプライアンス始動の進行状況をモニターします。

2.5+ 以下のいずれかのオプションを使用します。

- **2.5+** アプライアンスの始動状況をユーザー・インターフェースで追跡することができます。Web ブラウザーのアドレス・バーで、次の URL およびポート情報を入力してください。https://<appliance_hostname>:9443/。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。
- **start-progress** コマンドを実行して、進行をモニターします。このコマンドが STARTED を返す場合、アプライアンスは使用可能です。

タスクの結果

アプライアンスには更新されたファームウェアのバージョンがあります。ファームウェアのアップグレードが正常に行われたことを確認するために、「アプライアンス」 > 「設定」パネルの「ファームウェア」セクションで現行ファームウェア・レベルを確認できます。このパネルには、アプライアンス・モデル・タイプとアプライアンスのシリアル番号も表示されます。集合内の他のアプライアンスに対して、この手順を繰り返します。

次のタスク

新しいアプライアンスに初めてファームウェアをインストールする場合は、ファームウェア更新が完了してから **clear-all** コマンドをアプライアンスで実行してください。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)を参照してください。

親トピック: [WebSphere DataPower XC10 アプライアンスの更新](#)

次のトピック: [WebSphere Application Server 上の WebSphere eXtreme Scale クライアントの更新](#)

関連タスク:

[クイック・スタート: アプライアンス・ハードウェアのインストール](#)

[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)

WebSphere Application Server 上の WebSphere eXtreme Scale クライアントの更新

WebSphere® Application Server を新しいバージョンにマイグレーションするとき、WebSphere eXtreme Scale クライアントの構成を新しい WebSphere Application Server のインストール済み環境にマイグレーションすることもできます。

始める前に

- WebSphere eXtreme Scale バージョン 7 と WebSphere eXtreme Scale バージョン 8 をいずれも同じサーバー上にインストールすることを前提とします。
- WebSphere Application Server バージョン 7 を WebSphere Application Server バージョン 8 にマイグレーションします。詳しくは、[製品構成のマイグレーション](#)を参照してください。
- WebSphere eXtreme Scale クライアント バージョン 8 をお使いの WebSphere Application Server バージョン 8 のインストール済み環境にインストールします。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。WebSphere eXtreme Scale クライアントのマイグレーション・スクリプトはすべて、WebSphere eXtreme Scale クライアント バージョン 8.5 以降から実行する必要があります。例えば、バージョン 7.x からバージョン 8 にマイグレーションする場合、マイグレーション用のスクリプトを <WXS_v8_install_root>/bin ディレクトリーから実行します。

このタスクについて

WebSphere eXtreme Scale を統合している WebSphere Application Server の新しいバージョンをインストールするときは、まず、通常のプロセスで WebSphere Application Server をアップグレードします。次に、その新しいインストール済み環境に WebSphere eXtreme Scale クライアントの新しいバージョンをインストールします。その後、**xsmigration** スクリプトを使用して WebSphere eXtreme Scale 構成情報を新しい WebSphere Application Server インストール済み環境に移動できます。

手順

1. デプロイメント・マネージャー関連の構成をバージョン 7 からバージョン 8 にマイグレーションします。
 - a. WebSphere Application Server バックアップ・スクリプトを実行します。詳しくは、[WASPreUpgrade コマンド](#)を参照してください。
 - b. デプロイメント・マネージャーを停止します。
 - c. WebSphere eXtreme Scale クライアント 構成内のデプロイメント・マネージャー・サーバーにアクセスして、マイグレーション・スクリプトを実行します。
 - i. ディレクトリーを <WXS_v8_install_root>/bin に移動します。
 - ii. 以下のコマンドを実行します。

```
xsmigration.bat|sh -targetwashome <WAS8x_HOME>  
-sourcwashome <WAS7x_HOME> -targetprofilepath <WAS8x_DmgrProfile>  
-sourceprofilepath <WAS7x_DmgrProfile>
```

各部の意味は、次のとおりです。

- <WAS8x_HOME> は、WebSphere Application Server バージョン 8.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere8
- <WAS7x_HOME> は、WebSphere Application Server バージョン 7.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere7
- <WAS8x_DmgrProfile> は、WebSphere Application Server バージョン 8.x のデプロイメント・マネージャーのプロファイルが置かれているロケーションです。例: /opt/IBM/WebSphere8/profiles/DMgr01
- <WAS7x_DmgrProfile> は、WebSphere Application Server バージョン 7.x のデプロイメント・マネージャーのプロファイルが置かれているロケーションです。例: /opt/IBM/WebSphere7/profiles/DMgr01

2. アプリケーション・サーバー関連の構成をバージョン 7 からバージョン 8 にマイグレーションします。
 - a. ディレクトリーを <WXS_v8_install_root>/bin に移動します。
 - b. 以下のコマンドを実行します。

```
xsmigration.bat|sh -targetwashome <WAS8x_HOME>  
-sourcwashome <WAS7x_HOME> -targetprofilepath <WAS8x_AppServerProfile>  
-sourceprofilepath <WAS7x_AppServerProfile>
```

各部の意味は、次のとおりです。

- <WAS8x_HOME> は、WebSphere Application Server バージョン 8.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere8

- <WAS7x_HOME> は、WebSphere Application Server バージョン 7.x のインストール済み環境のルート・ロケーションです。例: /opt/IBM/WebSphere7
- <WAS8x_AppServerProfile> は、WebSphere Application Server バージョン 8.x のアプリケーション・サーバー・プロファイルが置かれているロケーションです。例:
/opt/IBM/WebSphere8/profiles/AppServer01
- <WAS7x_AppServerProfile> は、WebSphere Application Server バージョン 7.x のアプリケーション・サーバー・プロファイルが置かれているロケーションです。例:
/opt/IBM/WebSphere7/profiles/AppServer01

3. WebSphere Application Server バージョン 8 のデプロイメント・マネージャーを再始動して、すべての管理対象ノードを同期化します。

親トピック: [WebSphere DataPower XC10 アプライアンスの更新](#)

前のトピック: [ファームウェアの更新](#)

次のトピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

IBM Installation Manager の使用によるフィックスパックのインストール

IBM® Installation Manager を使用して、WebSphere® eXtreme Scale クライアント 製品オフリングに使用可能なフィックスパックでこの製品を更新することができます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してインストールできます。

[GUI の使用によるフィックスパックのインストール](#)

IBM Installation Manager ウィザードを使用して、WebSphere eXtreme Scale クライアント を新しいバージョンに更新できます。

[コマンド行の使用によるフィックスパックのインストール](#)

IBM Installation Manager をコマンド行から使用して、WebSphere eXtreme Scale 製品オフリングに使用可能なフィックスパックで製品を更新することができます。

[応答ファイルの使用によるフィックスパックのインストール](#)

IBM Installation Manager で応答ファイルを使用して、WebSphere eXtreme Scale クライアント を新しいバージョンに更新できます。

親トピック: [WebSphere DataPower XC10 アプライアンス の更新](#)

前のトピック: [WebSphere Application Server 上の WebSphere eXtreme Scale クライアント の更新](#)

次のトピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

GUI の使用によるフィックスパックのインストール

IBM® Installation Manager ウィザードを使用して、WebSphere® eXtreme Scale クライアント を新しいバージョンに更新できます。

始める前に

WebSphere eXtreme Scale クライアントを更新する前に、アプライアンス・ファームウェアをアップグレードします。詳しくは、[ファームウェアの更新](#)を参照してください。WebSphere eXtreme Scale スタンドアロンまたは WebSphere eXtreme Scale for WebSphere Application Server 製品オフリング用のアップグレードに関する情報については、IBM ソフトウェア・サポート・センターにお問い合わせください。最新の情報は、IBM ソフトウェア・サポート・センターおよび [Fix Central](#) から入手できます。

IBM Installation Manager を使用して、以下の製品オフリングに製品保守を適用します。

- WebSphere eXtreme Scale クライアント (スタンドアロン環境)
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

Web ベースのサービス・リポジトリまたはローカル・サービス・リポジトリのロケーションがリストされ、チェックされているかどうか、あるいは Installation Manager 設定の「リポジトリ」パネルで「インストール中および更新中にサービス・リポジトリの検索」オプションが選択されているかどうかを確認してください。Installation Manager でのサービス・リポジトリの使用について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

このタスクについて

制約事項: Installation Manager を使用して、インストール済み環境をアップグレードしたり、完全な WebSphere Application Server プロファイルの機能を追加または削除したりすることはできません。

手順

1. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
2. システムにログオンします。
3. 更新される WebSphere Application Server インストール済み環境で、すべてのサーバーおよびアプリケーションを停止します。
4. Installation Manager を開始します。
5. 「更新」をクリックします。

注: 認証のプロンプトが出されたら、保護されている IBM ソフトウェア Web サイトへのアクセスに使用する IBM ID とパスワードを使用してください。

6. 更新するパッケージ・グループを選択します。

ヒント: 「すべて更新」を選択すると、Installation Manager は過去にインストールしたすべてのパッケージ・グループに対する更新の追加リポジトリおよび定義済みリポジトリをすべて検索します。この機能は、ターゲット・リポジトリ内に含まれるフィックスについて完全に制御できる場合のみ使用してください。インストールしたいフィックスのみを含んだ一連のカスタム・リポジトリを作成して指定しているのであれば、この機能を使用しても問題はありません。サービス・リポジトリの検索を使用可能にする場合や、他の Web ベースのライブ・リポジトリから直接フィックスをインストールする場合は、このオプションの選択は推奨されません。未選択であれば、後続のパネルで各オフリングに対してインストールするフィックスのみを選択できます。

7. 「次へ」をクリックします。
8. 更新後のバージョンを選択します。
 - WebSphere eXtreme Scale クライアント (スタンドアロン環境)
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
 - WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8
9. インストールするフィックスを選択します。

デフォルトでは、任意の推奨フィックスが選択されます。

推奨フィックスがある場合、推奨フィックスのみを表示し、非推奨フィックスを非表示にするオプションを選択できます。

10. 「次へ」をクリックします。

11. 使用条件の条項に同意し、「次へ」をクリックします。
12. 更新後のインストール済み環境に必要なオプション・フィーチャーを選択します。
13. 要約情報を検討して、「更新」をクリックします。
 - 正常にインストールされた場合は、プログラムにより、インストールが正常に行われたことを示すメッセージが表示されます。
 - インストールが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。
14. 「終了」をクリックします。
15. 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

親トピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

関連タスク:

[GUI の使用によるフィックスパックのアンインストール](#)

コマンド行の使用によるフィックスパックのインストール

IBM® Installation Manager をコマンド行から使用して、WebSphere® eXtreme Scale 製品オフリングに使用可能なフィックスパックで製品を更新することができます。

始める前に

WebSphere eXtreme Scale クライアントを更新する前に、アプライアンス・ファームウェアをアップグレードします。詳しくは、[ファームウェアの更新](#)を参照してください。WebSphere eXtreme Scale スタンドアロンまたは WebSphere eXtreme Scale for WebSphere Application Server 製品オフリングのアップグレードに関する情報については、IBM ソフトウェア・サポート・センターにお問い合わせください。最新の情報は、IBM ソフトウェア・サポート・センターおよび [Fix Central](#) から入手できます。

IBM Installation Manager を使用して、以下の製品オフリングに製品保守を適用します。

- WebSphere eXtreme Scale クライアント (スタンドアロン環境)
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 7
- WebSphere eXtreme Scale クライアント for WebSphere Application Server バージョン 8

このタスクについて

制約事項: Installation Manager を使用して、インストール済み環境をアップグレードしたり、完全な WebSphere Application Server プロファイルの機能を追加または削除したりすることはできません。

手順

1. WebSphere eXtreme Scale 8.5 に有効なインテリム・フィックスとフィックスパックのリスト、および各フィックスに関する具体的な情報を入手するには、次のアクションを実行します。
 - a. [Fix Central](#) に移動します。
 - b. **WebSphere** を製品グループとして選択します。
 - c. WebSphere eXtreme Scale を製品として選択します。
 - d. インストール・バージョンとして **8.5** を選択します。
 - e. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - f. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - g. 各フィックスの下の「**詳細**」をクリックすると、そのフィックスの情報が表示されます。
 - h. **推奨:** インストールしたいフィックスパックの名前をメモしておいてください。
2. 次の手順を使用して、フィックスパックで WebSphere eXtreme Scale バージョン 8.5 を更新してください。
 - フィックスパックを含むファイルを Fix Central からダウンロードし、ローカル更新を行います。

フィックスパックを含む圧縮ファイルは Fix Central からダウンロードできます。フィックスパックの各圧縮ファイルには、そのフィックスパックの Installation Manager リポジトリが含まれていて、通常は .zip という拡張子が付いています。フィックスパック・ファイルをダウンロードして解凍した後、Installation Manager を使用してフィックスパックで WebSphere Application Server バージョン 8.x を更新します。

- a. フィックスパックをダウンロードするには、以下の手順を行います。
 - i. [Fix Central](#) に移動します。
 - ii. **WebSphere** を製品グループとして選択します。
 - iii. **WebSphere eXtreme Scale** を製品として選択します。
 - iv. インストール・バージョンとして **8.5** を選択します。
 - v. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - vi. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - vii. ダウンロードするフィックスパックを選択して、「**続行**」をクリックします。
 - viii. ダウンロード・オプションを選択して、「**続行**」をクリックします。
 - ix. ご使用条件に同意する場合は「**同意します**」をクリックします。
 - x. 「**今すぐダウンロード**」をクリックしてフィックスパックをダウンロードします。
 - xi. 圧縮ファイルをバイナリー形式でインストール先のシステムに転送します。
 - xii. 圧縮されたリポジトリ・ファイルをシステム上のディレクトリに解凍します。
- b. ダウンロードしたファイルからフィックスパックをインストールするには、次のアクションを実行します。

- i. システムにログオンします。
- ii. ご使用の環境で実行中のすべてのプロセスを停止します。WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
- iii. `Installation_Manager_binaries/eclipse/tools` ディレクトリーに移動します。ここでは、`Installation_Manager_binaries` が Installation Manager のインストール・ルート・ディレクトリーです。
- iv. フィックスパックをインストールします。

UNIX | Linux

```
./imcl install offering_ID offering_version,optional_feature_ID  
-installationDirectory product_installation_location  
-repositories location_of_expanded_files  
-acceptLicense
```

Windows

```
imcl.exe install offering_ID offering_version,optional_feature_ID  
-installationDirectory product_installation_location  
-repositories location_of_expanded_files  
-acceptLicense
```

ヒント:

- `offering_version` は、アンダースコアを使用してオファリング ID にオプションで付加することができます。これはインストール対象のオファリングの特定バージョンです (例: 8.5.0.20110503_0200)。
 - `offering_version` が指定されていない場合、オファリングの最新バージョンと、そのバージョンのすべてのインテリム・フィックスがインストールされます。
 - `offering_version` が指定されている場合、オファリングの指定バージョンがインストールされ、そのバージョンのインテリム・フィックスはインストールされません。

オファリングのバージョンは、リポジトリーに対して次のコマンドを実行すると、オファリング ID の最後にアンダースコアを使用して付加されている形で見つかります。

```
imcl listAvailablePackages -repositories source_repository
```

- また、`-installFixes` 引数に `none`、`recommended`、または `all` を指定することで、どのインテリム・フィックスをオファリングと一緒にインストールするのかを示すこともできます。
 - オファリングのバージョンが指定されていない場合、`-installFixes` オプションはデフォルトで `all` になります。
 - オファリングのバージョンが指定されている場合、`-installFixes` オプションはデフォルトで `none` になります。
 - コンマで区切られたフィーチャーのリストを追加できます。フィーチャーのリストが指定されていない場合は、デフォルトのフィーチャーがインストールされます。
- v. オプション: すべてのインストール済みパッケージをリストして、インストールを検証します。

UNIX | Linux

```
./imcl listInstalledPackages -long
```

Windows

```
imcl.exe listInstalledPackages -long
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

関連タスク:

[コマンド行の使用によるフィックスパックのアンインストール](#)

応答ファイルの使用によるフィックスパックのインストール

IBM® Installation Manager で応答ファイルを使用して、WebSphere® eXtreme Scale クライアント を新しいバージョンに更新できます。

始める前に

ヒント: 本書で説明した手順に代わる手段として、Installation Manager で、応答ファイルまたはコマンド行で **updateAll** コマンドを使用してすべてのインストール済みパッケージの検索と更新を行うことができます。このコマンドは、ターゲット・リポジトリ内にどのフィックスを含めるかについて、完全に制御できる場合のみ使用してください。インストールしたいフィックスのみを含んだ一連のカスタム・リポジトリを作成して指定しているのであれば、このコマンドを使用しても問題はありません。サービス・リポジトリの検索を使用可能にする場合や、他の Web ベースのライブ・リポジトリから直接フィックスをインストールする場合は、このオプションの選択は推奨されません。未選択であれば、コマンド行から **install** コマンドと共に **-installFixes** オプションを使用するか、応答ファイルで **installFixes** 属性を使用して、インストールするフィックスのみを選択できます。

手順

1. WebSphere eXtreme Scale クライアント に有効なインテリム・フィックスとフィックスパックのリスト、および各フィックスに関する具体的な情報を入手するには、次のアクションを実行します。
 - a. [Fix Central](#) に移動します。
 - b. **WebSphere** を製品グループとして選択します。
 - c. WebSphere eXtreme Scale クライアントを製品として選択します。
 - d. インストール・バージョンとして **8.x** を選択します。
 - e. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - f. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - g. 各フィックスの下の「**詳細**」をクリックすると、そのフィックスの情報が表示されます。
 - h. **推奨:** インストールしたいフィックスパックの名前をメモしておいてください。
2. 次の手順を使用して、フィックスパックで WebSphere eXtreme Scale クライアント を更新してください。
 - フィックスパックを含むファイルを Fix Central からダウンロードし、ローカル更新を行います。

フィックスパックを含む圧縮ファイルは Fix Central からダウンロードできます。フィックスパックの各圧縮ファイルには、そのフィックスパックの Installation Manager リポジトリが含まれていて、通常は .zip という拡張子が付いています。フィックスパック・ファイルをダウンロードして解凍した後、Installation Manager を使用してフィックスパックで WebSphere eXtreme Scale クライアント を更新します。
 - a. フィックスパックをダウンロードするには、以下の手順を行います。
 - i. [Fix Central](#) に移動します。
 - ii. **WebSphere** を製品グループとして選択します。
 - iii. **WebSphere eXtreme Scale** クライアントを製品として選択します。
 - iv. インストール・バージョンとして **8.6** を選択します。
 - v. プラットフォームとしてご使用のオペレーティング・システムを選択し、「**続行**」をクリックします。
 - vi. 「**フィックスの参照**」を選択し、「**続行**」をクリックします。
 - vii. ダウンロードするフィックスパックを選択して、「**続行**」をクリックします。
 - viii. ダウンロード・オプションを選択して、「**続行**」をクリックします。
 - ix. ご使用条件に同意する場合は「**同意します**」をクリックします。
 - x. 「**今すぐダウンロード**」をクリックしてフィックスパックをダウンロードします。
 - xi. 圧縮ファイルをバイナリ形式でインストール先のシステムに転送します。
 - xii. 圧縮されたリポジトリ・ファイルをシステム上のディレクトリに解凍します。

b. 次のアクションを実行します。

- i. システムにログオンします。
- ii. リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、**imutilsc** コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

- iii. `Installation_Manager_binaries/eclipse/tools` ディレクトリーに移動します。ここでは、`Installation_Manager_binaries` が Installation Manager のインストール・ルート・ディレクトリーです。
- iv. 応答ファイルを使用してフィックスパックをインストールします。

例:

- **Windows** 管理者または非管理者:

```
imcl.exe -acceptLicense
input C:¥temp¥update_response_file.xml
-log C:¥temp¥update_log.xml
-keyring C:¥IM¥im.keyring
```

- **UNIX** | **Linux** 管理者:

```
./imcl -acceptLicense
input /var/temp/update_response_file.xml
-log /var/temp/update_log.xml
-keyring /var/IM/im.keyring
```

- **UNIX** | **Linux** 非管理者:

```
./imcl -acceptLicense
input user_home/var/temp/update_response_file.xml
-log user_home/var/temp/update_log.xml
-keyring user_home/var/IM/im.keyring
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

関連タスク:

[応答ファイルを使用したフィックスパックのアンインストール](#)

IBM Installation Manager の使用によるフィックスパックのアンインストール

IBM® Installation Manager を使用して WebSphere® eXtreme Scale クライアント 製品オフリングを前のバージョンにロールバックできます。フィックスパックは GUI、コマンド行、または応答ファイルを使用してアンインストールできます。

[GUI の使用によるフィックスパックのアンインストール](#)

IBM Installation Manager GUI を使用して、WebSphere eXtreme Scale クライアント を前のバージョンにロールバックできます。

[コマンド行の使用によるフィックスパックのアンインストール](#)

コマンド行から IBM Installation Manager を使用して、WebSphere eXtreme Scale クライアント を前のバージョンにロールバックできます。

[応答ファイルを使用したフィックスパックのアンインストール](#)

IBM Installation Manager で応答ファイルを使用して、WebSphere eXtreme Scale クライアント を前のバージョンにロールバックできます。

親トピック: [WebSphere DataPower XC10 アプライアンス の更新](#)

前のトピック: [IBM Installation Manager の使用によるフィックスパックのインストール](#)

次のトピック: **2.5+** [WebSphere eXtreme Scale Client for .NET のアップグレード](#)

GUI の使用によるフィックスパックのアンインストール

IBM® Installation Manager GUI を使用して、WebSphere® eXtreme Scale クライアント を前のバージョンにロールバックできます。

始める前に

ロールバック・プロセス中は、Installation Manager から前バージョンのパッケージのファイルにアクセスする必要があります。デフォルトでは、これらのファイルはパッケージのインストール時に、ご使用のコンピューターに保管されます。デフォルト設定を変更した場合や保存されたファイルを削除した場合は、前バージョンのインストールに使用したリポジトリに Installation Manager からアクセスする必要があります。

このタスクについて

制約事項: Installation Manager を使用してインストール済み環境をロールバックしたり、フィーチャーを追加または削除したりすることはできません。

手順

- ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
- Installation Manager を開始します。
- 「ロールバック」をクリックします。
- ロールバックするパッケージ・グループを選択します。
- 「次へ」をクリックします。
- ロールバック後のバージョンを選択します。
- 「次へ」をクリックします。
- 要約情報を検討して、「ロールバック」をクリックします。
 - 正常にロールバックされた場合は、プログラムにより、ロールバックが正常に行われたことを示すメッセージが表示されます。
 - ロールバックが正常に行われなかった場合は、「ログ・ファイルの表示」をクリックして、問題のトラブルシューティングを行います。
- 「終了」をクリックします。
- 「ファイル」>「終了」をクリックして、Installation Manager を閉じます。

親トピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

関連タスク:

[GUI の使用によるフィックスパックのインストール](#)

コマンド行の使用によるフィックスパックのアンインストール

コマンド行から IBM® Installation Manager を使用して、WebSphere® eXtreme Scale クライアント を前のバージョンにロールバックできます。

始める前に

制約事項: この手順を実行するには、ご使用のシステムに Installation Manager バージョン 1.5 以降がインストールされている必要があります。

ロールバック・プロセス中は、Installation Manager から前バージョンのパッケージのファイルにアクセスする必要があります。デフォルトでは、これらのファイルはパッケージのインストール時に、ご使用のコンピューターに保管されます。デフォルト設定を変更した場合や保存されたファイルを削除した場合は、前バージョンのインストールに使用したリポジトリに Installation Manager からアクセスする必要があります。

このタスクについて

制約事項: Installation Manager を使用してインストール済み環境をロールバックしたり、完全な WebSphere Application Server プロファイルのフィーチャーを追加または削除したりすることはできません。

手順

1. オプション: リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、**imutilsc** コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

2. システムにログオンします。
3. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
4. Installation Manager をインストールしたディレクトリーの `eclipse/tools` サブディレクトリーに移動します。
5. **imcl** コマンドを使用して、この製品をロールバックします。

UNIX

Linux

```
./imcl rollback offering_ID_offering_version
-repositories source_repository
-installationDirectory installation_directory
-preferences preference_key=value
-properties property_key=value
-keyring keyring_file -password password
-acceptLicense
```

Windows

```
imcl.exe rollback offering_ID_offering_version
-repositories source_repository
-installationDirectory installation_directory
-preferences preference_key=value
-properties property_key=value
-keyring keyring_file -password password
-acceptLicense
```

ヒント:

- `offering_version` は、アンダースコアを使用してオフリング ID にオプションで付加することができますが、これはロールバック対象のオフリングの特定バージョンです (例: `8.5.0.20110503_0200`)。
 - `offering_version` が指定されていない場合、インストール済み環境はそのオフリングの以前インストールされたバージョンにロールバックし、そのバージョンのすべてのインテリム・フィックスがインストールされます。
 - `offering_version` が指定されている場合、インストール済み環境はそのオフリングの指定され

た前のバージョンにロールバックし、そのバージョンのインテリム・フィックスはインストールされません。

オフリングのバージョンは、**historyInfo** コマンドまたは **genHistoryReport** コマンドを `app_server_root/bin` ディレクトリーから実行する際に生成されるレポートの「パッケージ」セクションで、アンダースコアを使用してオフリング ID の最後に付加されている場合があります。

Installation Manager の使用について詳しくは、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

6. オプション: すべてのインストール済みパッケージをリストして、ロールバックを検証します。

UNIX | Linux

```
./imcl listInstalledPackages -long
```

Windows

```
imcl.exe listInstalledPackages -long
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

関連タスク:

[コマンド行の使用によるフィックスパックのインストール](#)

応答ファイルを使用したフィックスパックのアンインストール

IBM® Installation Manager で応答ファイルを使用して、WebSphere® eXtreme Scale クライアント を前のバージョンにロールバックできます。

始める前に

ロールバック・プロセス中は、Installation Manager から前バージョンのパッケージのファイルにアクセスする必要があります。デフォルトでは、これらのファイルはパッケージのインストール時に、ご使用のコンピューターに保管されます。デフォルト設定を変更した場合や保存されたファイルを削除した場合は、前バージョンのインストールに使用したリポジトリに Installation Manager からアクセスする必要があります。

このタスクについて

制約事項: Installation Manager を使用してインストール済み環境をロールバックしたり、完全な WebSphere Application Server プロファイルの機能を追加または削除したりすることはできません。

手順

1. オプション: リポジトリでユーザー名とパスワードが必要な場合、このリポジトリにアクセスするために鍵リング・ファイルを作成します。

Installation Manager の鍵リング・ファイルの作成については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

ヒント: 鍵リング・ファイルを作成する際、**imutilsc** コマンドで、指定したリポジトリの URL を見つけられない場合は、その URL のロケーションの最後に `/repository.config` を追加してください。

2. システムにログオンします。
3. ご使用の環境で実行中のすべてのプロセスを停止します。
 - WebSphere Application Server 環境で実行中のすべてのプロセスを停止する場合は、[コマンド行ユーティリティー \(Command-line utilities\)](#) を参照してください。
4. 応答ファイルを使用して、この製品をロールバックします。

Installation Manager をインストールしたディレクトリーの `eclipse/tools` サブディレクトリーに移動して、この製品をロールバックします。

以下に例を示します。

- **Windows** 管理者または非管理者:

```
imcl.exe
input C:%temp%rollback_response_file.xml
-log C:%temp%rollback_log.xml
-keyring C:%IM%im.keyring
```

- **UNIX** | **Linux** 管理者:

```
./imcl
input /var/temp/rollback_response_file.xml
-log /var/temp/rollback_log.xml
-keyring /var/IM/im.keyring
```

- **UNIX** | **Linux** 非管理者:

```
./imcl
input user_home/var/temp/rollback_response_file.xml
-log user_home/var/temp/rollback_log.xml
-keyring user_home/var/IM/im.keyring
```

注: プログラムが重要なポストインストール指示を標準出力に書き込むことがあります。

Installation Manager の使用については、[IBM Installation Manager バージョン 1.5 インフォメーション・センター](#)を参照してください。

5. オプション: すべてのインストール済みパッケージをリストして、ロールバックを検証します。

```
./imcl listInstalledPackages -long
```

Windows

```
imcl.exe listInstalledPackages -long
```

親トピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

関連タスク:

[応答ファイルの使用によるフィックスパックのインストール](#)

WebSphere eXtreme Scale Client for .NET のアップグレード

インストール・プログラムを実行して、既存の WebSphere® eXtreme Scale Client for .NET インストール済み環境をアップグレードすることができます。プログラムは既存のインストール済み環境を検出し、該当するファイルを置き換えます。

始める前に

- WebSphere eXtreme Scale Client for .NET をアップグレードする前にアプライアンス・ファームウェアを更新してください。詳しくは、[ファームウェアの更新](#)を参照してください。
- WebSphere eXtreme Scale Client for .NET のアップグレードをダウンロードしてください。最新の情報は、IBM ソフトウェア・サポート・センターおよび [Fix Central](#) から入手できます。使用可能なダウンロードが新しいインストール済み環境とアップグレードの両方に適用されます。

このタスクについて

このインストール手順によって、既存のインストール済み環境は即座に置き換えられます。

手順

1. ご使用の環境で実行中のすべてのプロセスを停止します。
2. ウィザードを実行して、WebSphere eXtreme Scale Client for .NET アップグレードをインストールします。インストール・ウィザードを実行して、古いインストール済み環境が検出されたら、その既存のインストール済み環境をアップグレードすることを確認する必要があります。ウィザードの進行状況パネルに、古いバージョンとアップグレードされたバージョンが示されます。

タスクの結果

既存のすべての WebSphere eXtreme Scale Client for .NET コードが置き換えられます。この処理は、ディスク上とグローバル・アセンブリー・キャッシュ (GAC) 内の両方で行われます。公開されたポリシー・ファイルがディスクと GAC にインストールされ、既存のファイルに取って代わります。

次のタスク

- WebSphere eXtreme Scale Client for .NET を構成します。詳しくは、[WebSphere eXtreme Scale Client for .NET の構成](#)を参照してください。
- .NET アプリケーションを開発します。詳しくは、[.NET API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。

[WebSphere eXtreme Scale Client for .NET のフィックスパックのサイドバイサイド・インストールの作成](#)

サイドバイサイド・インストールを作成すると、WebSphere eXtreme Scale Client for .NET のフィックスパックの複数バージョンを同じサーバー上で実行することができます。旧バージョン向けに作成された既存の .NET アプリケーションは、旧クライアント向けに引き続き実行することができます。

親トピック: [WebSphere DataPower XC10 アプライアンス の更新](#)

前のトピック: [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)

WebSphere eXtreme Scale Client for .NET のフィックスパックのサイドバイサイド・インストールの作成

サイドバイサイド・インストールを作成すると、WebSphere® eXtreme Scale Client for .NET のフィックスパックの複数バージョンを同じサーバー上で実行することができます。旧バージョン向けに作成された既存の .NET アプリケーションは、旧クライアント向けに引き続き実行することができます。

始める前に

- WebSphere eXtreme Scale Client for .NET の旧バージョンがインストールされているシステムと、最新のフィックスパックをインストールする別のシステムが必要です。
- WebSphere eXtreme Scale Client for .NET の最新バージョンを、実動システムとは別のシステムにインストールします。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストール](#)を参照してください。

このタスクについて

WebSphere eXtreme Scale Client for .NET のフィックスパック・インストールを実行すると、既存のインストール済み環境が新しいバージョンで置き換えられます。環境に応じて、新しいバージョンにアップグレードする前にテストをしたいことがあります。また、アプリケーションによって、旧バージョンを使用させたいものと、最新バージョンを使用させたいものがあります。WebSphere eXtreme Scale Client for .NET の手動インストールによって、複数のバージョンをサイドバイサイドで使用することができます。

WebSphere eXtreme Scale Client for .NET の新しいメジャー・リリースをインストールすると、サイドバイサイド・インストールが自動的に行われます。

手順

1. 最新のインストール済み環境を持つシステムから、[net_client_home](#) インストール・ディレクトリー全体とそのすべてのサブフォルダーをコピーします。これらのファイルは、旧バージョンがインストールされているターゲット・システム上の既存のインストール済み環境にある別のディレクトリーに配置します。このディレクトリーは *sidebyside_home* と呼ばれます。
2. 手動で、最新の WebSphere eXtreme Scale Client for .NET アセンブリーを *sidebyside_home/bin* ディレクトリーから GAC にインストールします。このインストールが完了すると、WebSphere eXtreme Scale Client for .NET アセンブリーの旧バージョンと新バージョンの両方が GAC にインストールされたこととなります。手動でアセンブリーを GAC にインストールする方法は、インストールされている Windows と .NET フレームワークのバージョンによって異なります。例えば、Windows エクスプローラーを使用して、アセンブリー・ファイルを %systemroot%\assembly ディレクトリーにコピーすることがあります。また、Microsoft から *gacutil.exe* ユーティリティーをダウンロードして使用するということがあります。
3. 新しいバージョンを使用するように WebSphere eXtreme Scale Client for .NET アプリケーションを更新します。以下のいずれかのオプションを使用します。
 - 更新したい各アプリケーションで、アセンブリー・リダイレクト・エレメントを構成ファイルに追加します。このエレメントは、古い WebSphere eXtreme Scale Client for .NET アセンブリーの参照を新しいアセンブリーにリダイレクトします。

```
<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <dependentAssembly>
      <assemblyIdentity name="IBM.WebSphere.Caching"
publicKeyToken="b439a24ee43b0816" />
      <bindingRedirect oldVersion="8.6.0.0-8.6.0.1" newVersion="8.6.0.2" />
    </dependentAssembly>
  </assemblyBinding>
</runtime>
```

- アプリケーション・プロジェクトのアセンブリー参照のリストにある新しいアセンブリーを使用して WebSphere eXtreme Scale Client for .NET をビルドし直します。

タスクの結果

- 更新したアプリケーションは WebSphere eXtreme Scale Client for .NET の新しいほうのバージョンを使用し、その他のアプリケーションは旧バージョンを使用します。
- 新しいバージョンを使用しているアプリケーションのログ・ファイルは、既存のアプリケーションが使用するのと同じログ・ディレクトリーにあります。
- デフォルトのクライアント構成は、従来どおり、[net_client_home/config](#) ディレクトリーにある古い WebSphere eXtreme Scale Client for .NET インストール済み環境から取得されます。別のプロパティー・ファイルを使用するには、新しいプロパティー・ファイルのパスを Connect() API に明示的に渡すようにしてくだ

さい。

警告: 行った変更は手動でアンインストールしなければなりません。手動インストール済み環境は、WebSphere eXtreme Scale Client for .NET インストール・プログラムによるアップグレードの対象外です。

親トピック: **2.5+** [WebSphere eXtreme Scale Client for .NET のアップグレード](#)

アプライアンスの構成

シリアル・コンソールを使用してアプライアンスを初期設定した後、アプライアンスの機能を有効にするために、ユーザー・インターフェースを使用してアプライアンスを構成する必要があります。

始める前に

- アプライアンス・ハードウェアのインストールおよび初期化を完了します。詳しくは、[WebSphere DataPower XC10 アプライアンスのインストール](#)を参照してください。
- 以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

シリアル・コンソールを使用する初期設定プロセスによって、アプライアンスはユーザー・インターフェースを使用して管理されるように準備されます。アプライアンスを使用できるように準備するには、追加設定を構成する必要があります。

注: これらのすべての設定が重要ですが、ドメイン・ネーム・システム (DNS) サーバーの追加、Network Time Protocol (NTP) サーバーの追加、およびメール送達の構成を適切に行うことが特に重要です。これらのタスクは、他の何らかの使用が発生する前に実行する必要があります。

[セキュリティ](#)

ユーザー・インターフェース・セキュリティおよびトランスポート・レベル・セキュリティを含めて、アプライアンス上のセキュリティのいくつかの側面を構成できます。

[ユーザーおよびグループの管理](#)

ユーザーおよびユーザー・グループが用意されているため、WebSphere DataPower XC10 アプライアンスへのアクセス・レベルを個人別に管理できます。ユーザー・グループを使用して、ユーザーのグループに許可を適用できます。

[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)

初回のシリアル接続時に、アプライアンスをネットワークに接続するために *mgmt* イーサネット・インターフェースを構成しました。追加のプライベート・イーサネット・ポートをユーザー・インターフェースで定義することができます。

[構成のインポートとエクスポート](#)

新規アプライアンスを構成する際、そのアプライアンスの構成設定をエクスポートして保管することができます。後に、アプライアンスの除去や再インストールを必要とする変更が発生した場合に、構成設定を損失することなく、保管しておいた構成データをインポートすることができます。

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

通信にドメイン・ネーム・システム (DNS) ルックアップ・サービスが使用されるため、IBM® WebSphere DataPower XC10 アプライアンスには DNS サーバーが必須です。アプライアンスの初期化時に、DNS サーバーを指定する必要があります。

[ホスト名に対する IP アドレスのマッピング](#)

アドレス情報を使用して TCP/IP ネットワーク・プロトコルに接続を作成するには、IP アドレスをホスト名に関連付ける必要があります。アプライアンス上で *etc/hosts* ファイルを編集することで、ホスト名に対して IP アドレスを解決できます。*etc/hosts* ファイルはユーザー・インターフェースから編集可能です。

[日時設定の管理](#)

集合内のご使用のアプライアンス全体で同期化された日時を維持するために、Network Time Protocol (NTP) サーバーを使用します。

[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)

アプライアンスのメール送達機能は、ユーザー・パスワードのリセットに使用します。ユーザーが新規パスワードを要求すると、アプライアンスは新規パスワードを記述した E メールをユーザーに送信します。

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

WebSphere DataPower XC10 アプライアンスは、ユーザー・インターフェースから再始動またはシャットダウンできます。

セキュリティ

ユーザー・インターフェース・セキュリティおよびトランスポート・レベル・セキュリティを含めて、アプライアンス上のセキュリティのいくつかの側面を構成できます。

[IBM WebSphere DataPower XC10 アプライアンス セキュリティ概要](#)

IBM WebSphere DataPower XC10 アプライアンスにより、アプライアンス本体と、アプライアンス上に保持されるデータ・グリッド (data grid) データの両方に対するアクセス権限をコントロールできます。

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティの構成](#)

WebSphere DataPower XC10 アプライアンスで提供されるセキュリティ機能の多くは、アプライアンスの構造内に構築されます。追加のセキュリティ・オプションをご使用の環境に提供するために、追加のセキュリティ設定が組み込まれています。

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

IBM WebSphere DataPower XC10 アプライアンスでユーザーを認証するために、任意で Lightweight Directory Access Protocol (LDAP) ディレクトリーを使用できます。

[ユーザーおよびグループの管理](#)

ユーザーおよびユーザー・グループが用意されているため、WebSphere DataPower XC10 アプライアンスへのアクセス・レベルを個人別に管理できます。ユーザー・グループを使用して、ユーザーのグループに許可を適用できます。

[データ・グリッドのセキュリティを使用可能に設定する](#)

データ・グリッドの作成後、データ・グリッド (data grid) のセキュリティはデフォルトで使用不可になります。データ・グリッド (data grid) のセキュリティ設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

Java

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

Java

[クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

アプリケーション用に構成しているデータ・グリッドでセキュリティを使用する場合、client.properties ファイルを構成して、そのデータ・グリッド・アプリケーションに受け渡し設定を含むようにします。

[データ・グリッド・アプリケーション用の TLS の構成](#)

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

[REST ゲートウェイ: セキュリティ構成](#)

REST ゲートウェイを介してデータ・グリッドにアクセスするには、データ・グリッドのセキュリティが使用可能に設定されているかどうかに関わらず、ユーザーは WebSphere DataPower XC10 アプライアンスに認証される必要があります。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを常に提供しなければなりません。REST ゲートウェイを介してデータ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。

親トピック: [アプライアンスの構成](#)

関連タスク:

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要

IBM® WebSphere® DataPower® XC10 アプライアンスにより、アプライアンス本体と、アプライアンス上に保持されるデータ・グリッド (data grid) データの両方に対するアクセス権限をコントロールできます。

アプライアンス・セキュリティー

アプライアンスをセキュア・ファウンデーションとするための主なフィーチャーを以下に示します。

不正開封防止ケースに入れられたアプライアンス

シャーシに侵入検知スイッチがあり、絶えずモニターされています。スイッチが起動した場合、アプライアンスは始動しません。アプライアンスを再び始動するには、アプライアンスを IBM に返却する必要があります。ケースの不正開封防止ねじなど、ケースが開かれないようにするための追加部品も使用されています。アプライアンスは、ケースを開かずにアプライアンスの背面からお客様が交換可能なアイテムにアクセスできるように設計されています。

シェル経由でのオペレーティング・システムへのアクセス不可

アプライアンスのオペレーティング・システムにはコマンド・シェルがありません。設計上、セキュリティーの脆弱性を削減するために、アプライアンスにはコマンド・インタープリターが組み込まれていません。アプライアンスには、オペレーティング・システムのユーザー ID が 1 つだけ存在します。使用可能なシェルがないため、ユーザー ID を使用して外部からアプライアンスにログオンすることはできません。

ユーザー提供のロジックを、アプライアンスで実行することはできません。

アプライアンスでは、ユーザーが実行可能スクリプトまたはコードをアップロードすることはできません。上記に関する唯一の例外は、システム・ファームウェア更新です。システム・ファームウェア更新では、スクリプトを実行して、更新されたファームウェアをアプライアンスにインストールできます。このようなシステム更新は、予防措置としてファームウェア製造元によって署名されています。ユーザー提供の非トラステッド・ソフトウェアをアプライアンスで実行することはできません。

データ・グリッドセキュリティー

ご使用のデータ・グリッドに含まれる情報へのアクセスをコントロールできます。ご使用のデータ・グリッド (data grid) でセキュリティーを有効にしないと、どのようなアプリケーションからでもデータ・グリッド (data grid) 内の情報にアクセス可能になります。データ・グリッド (data grid) において一般的なセキュリティーを有効にすることができますが、この場合はアプライアンス上にユーザー・アカウントとパスワードを持つユーザーなら誰もがデータ・グリッド (data grid) にアクセスできます。データ・グリッド (data grid) で許可を使用可能に設定することにより、特定のユーザーまたはユーザー・グループにアクセスを制限することもできます。

Transport Layer Security (TLS)

鍵ストア、トラストストア、および証明書別名を構成することにより、TLS を使用してデータ・グリッドおよびユーザー・インターフェースを保護できます。TLS の設定は、集合内のすべてのアプライアンスに適用されます。

ユーザーおよびユーザー・グループ

アプライアンスの管理およびデータ・グリッド (data grid) のセキュリティーの両方に対して、ユーザーおよびユーザー・グループの許可を定義できます。

親トピック: [セキュリティー](#)

関連概念:

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[クライアント・セキュリティーの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

[クライアント・プロパティー・ファイル](#)

WebSphere Application Server 用の Transport Layer Security (TLS) の構成

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

始める前に

- TLS は、バージョン 1.0.0.4 以降で構成できます。
 - WebSphere® eXtreme Scale クライアント、バージョン 7.1 フィックス 1 以降を使用している必要があります。
 - アプライアンス管理権限が割り当てられている必要があります。
 - アプライアンス構成に追加する、鍵ストアまたはトラストストアと、関連付けられたパスワードが必要です。
 - 既存のトラストストアを変更したければ、アプライアンスからトラストストアをダウンロードすることができます。
 - クライアントの公開証明書を使用してトラストストアを更新する必要があります。アプライアンスは、接続するクライアントを信頼する必要があります。
 - 指定されたトラストストアには、鍵ストア内のエントリーに対応する公開証明書が含まれている必要があります。アプライアンスの証明書別名の可能な構成オプションとして鍵ストアの証明書別名を指定できるようにするためには、その鍵ストアの証明書別名をトラストストア内で信頼する必要があります。
 - WebSphere Application Server のグローバル・セキュリティー設定によって、サーバーが WebSphere DataPower® XC10 アプライアンス への接続を 試行する方法が決まります。
 - グローバル・セキュリティー設定が使用不可の場合は、TCP/IP を使用して接続が試行されます。
 - グローバル・セキュリティー設定が使用可能な場合は、アプライアンスの公開証明書を WebSphere Application Server トラストストアに追加する必要があります。
- WebSphere DataPower XC10 アプライアンスの構成が「**TLS 必須**」になっている場合は、グローバル・セキュリティーを使用可能にする必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#)を参照してください。

このタスクについて

TLS 設定は、ユーザー・インターフェースおよびデータ・グリッドに適用されます。設定は、集合内のすべてのアプライアンスに適用されます。

手順

1. **WebSphere Application Server では必須:** アプライアンスの公開証明書を WebSphere Application Server のデフォルト・トラストストアに追加します。

- デフォルトのアプライアンス・トラストストアを使用する場合:

デプロイメント・マネージャーの `was_root/bin` ディレクトリーから `addXC10PublicCert.py` スクリプトを実行します。以下のコマンドを使用してください。

```
wsadmin -lang jython -f addXC10PublicCert.py
```

- アプライアンスのカスタム・キーを使用する場合:

`-certPath` コマンド行オプションを使用して、デプロイメント・マネージャーの `was_root/bin` ディレクトリーから `addXC10PublicCert.py` スクリプトを実行します。`-certPath` コマンド行オプションの値は、アプライアンスの鍵ストア用に構成された別名に対応する公開証明書のディスクの場所です。

```
wsadmin -lang jython -f addXC10PublicCert.py -certPath ./trustStore.jks
```

2. **WebSphere Application Server に必要:** アプライアンスのトラストストアと WebSphere Application Server の公開証明書をダウンロードし、`keytool` ユーティリティーを実行して、証明書をトラストストアに追加します。このツールは、WebSphere Application Server の証明書を組み込むようにアプライアンス・トラストストアを更新します。

- a. デフォルトのアプライアンス・トラストストアを使用する場合は、アクティブなトラストストアをダウンロードします。「集合」 > 「設定」 > 「**Transport Layer Security (TLS)**」をクリックします。「アクティブなトラストストアのダウンロード」をクリックし、ディスク上のファイルの保存場所 (例えば `/downloads/trustStore.jks` ディレクトリーなど) を覚えておきます。
- b. WebSphere Application Server の公開証明書を抽出します。
 - i. WebSphere Application Server 管理コンソールで、「セキュリティー」 > 「**SSL 証明書および鍵管理**」 > 「鍵ストアと証明書」をクリックします。
 - ii. 「鍵ストアの使用法」から「ルート証明書鍵ストア」を選択します。
 - iii. 「**DmgrDefaultRootStore**」を選択します。
 - iv. 「個人証明書」を選択します。

- v. ルート鍵ストア内の証明書の横にあるチェック・ボックスをクリックします。抽出する証明書の完全修飾ファイル名 (例: /certificates/public.cer) を指定します。
- vi. コマンド行ウィンドウで、次のコマンドを実行します: cd /java_home/bin
- vii. keytool ユーティリティを実行します。

```
keytool -import -noprompt -alias "example alias" -keystore
/downloads/trustStore.jks -file /certificates/public.cer -storepass
xc10pass -storetype jks
```

- viii. 追加でインポートする証明書があれば、この手順を繰り返し、証明書を抽出して keytool ユーティリティを再度実行します。
3. 鍵ストアとトラストストアの情報をアプライアンスにアップロードします。アプライアンス・ユーザー・インターフェイスで、「集合」 > 「設定」 > 「**Transport Layer Security (TLS)**」をクリックします。WebSphere Application Server の手順を完了したら、更新済みの /downloads/trustStore.jks ファイルをアップロードします。鍵ストアまたはトラストストアをアップロードした後、関連付けられたパスワードを更新する必要があります。デフォルトのトラストストアを使用する場合、パスワードは xc10pass です。
4. 集合の証明書別名を選択します。
5. トランスポート・タイプを指定します。以下のトランスポート・タイプ設定のいずれかを選択してください。
 - **TLS サポート**: データ・グリッドは TCP/IP、SSL、または TLS を使用して通信を行います。ユーザー・インターフェイスは HTTP および HTTPS を使用してアクセスできます。
 - **TLS 必須**: データ・グリッドは SSL または TLS のみを使用して通信を行います。ユーザー・インターフェイスは HTTPS のみを使用してアクセスできます。
 - **データ・グリッド TLS 使用不可**: データ・グリッドは非セキュア接続を使用して通信を行います。ユーザー・インターフェイスは HTTP および HTTPS を使用してアクセスできます。
6. 通信を可能にするための信頼できる証明書の送付をクライアントに要求するために、「**クライアント証明書認証を使用可能にする**」を選択します。
7. 「**TLS 設定の送信**」をクリックして、変更を構成に保存します。

タスクの結果

TLS 構成の変更を完了するために、集合を再始動する必要があります。

集合が再始動しているときには、ユーザー・インターフェイスの限られた一部分がアクセス可能です。ユーザー・インターフェイスの一部分にアクセスできない場合は、適当な時間を置いて、再度要求を実行依頼してください。「タスク」パネルに成功の状況が表示されて、いくつかの TLS 変更が完了したことが自動的に示されます。

アプライアンスが使用する証明書別名を変更した場合、ブラウザを再始動したり、ユーザー・インターフェイスをログアウトして再度ログインしたり、ブラウザ・プロンプトで新しい証明書を信頼したりする必要が生じる場合もあります。

クライアント認証が使用可能なときにユーザー・インターフェイスを使用できないようであれば、信頼できるクライアント証明書をブラウザにインポートしたかどうか確認してください。信頼できるクライアント証明書がブラウザにインポートされていない場合は、ユーザー・インターフェイスにアクセスできません。ユーザー・インターフェイスに正常にログオンした後、タスクによって、TLS 構成の成功が示されます。

次のタスク

ベスト・プラクティス

- 別のアプライアンスからユーザー・インターフェイスにアクセスする際、ブラウザからの警告を回避するために、鍵ストアの証明書の共通名 (CN) に、ワイルドカードを含めることを考慮してください。各アプライアンスは、TLS 構成に、証明書別名で指定されたものと同じ証明書を使用します。例えば、myhost.mycompany.com ではなく、*.mycompany.com を使用して、証明書を mycompany ドメイン内のすべてのホストに有効にすることができます。
- プライベート認証局 (CA) を使用して、TLS 構成で選択した証明書別名に関連付けられた証明書に署名することもできます。そうすると、CA 証明書をブラウザにインポートして、プライベート CA で署名された証明書によって、プロンプトの出力なしにそれぞれの集合を信頼することができます。一般に、プライベート CA の使用は、専用イントラネットでのアクセスにのみ適しています。

親トピック: [セキュリティ](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティ概要](#)
[ユーザー許可](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェイス・セキュリティの構成](#)
[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)
[ユーザーおよびグループの管理](#)
[データ・グリッドのセキュリティを使用可能に設定する](#)

Java [クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)
[データ・グリッド・アプリケーション用の TLS の構成](#)
[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)
[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)
[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)
[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)
[xscmd ユーティリティによるモニター](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)
[splicer.properties ファイル](#)

関連情報:

[IKEYMAN グラフィカル・インターフェースを使用した鍵の管理](#)

IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成

WebSphere® DataPower® XC10 アプライアンス で提供されるセキュリティー機能の多くは、アプライアンスの構造内に構築されます。追加のセキュリティー・オプションをご使用の環境に提供するために、追加のセキュリティー設定が組み込まれています。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

アプライアンスに組み込まれるセキュリティー機能について理解するには、『[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)』を参照してください。

このタスクについて

アプライアンスのセキュリティー適用範囲を広げるために、ユーザーの振る舞いを制御するための オプションをいくつか構成できます。

手順

- 「設定」パネルにナビゲートします。セキュリティー・オプションを管理するには、以下のいずれかの方法で「設定」パネルまでナビゲートしてください。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ようこそ」ページで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
- 「セキュリティー」を展開します。
- 使用するセキュリティー権限を設定します。
 - 「新規ユーザーに自分のアカウントの作成を許可する」フィールドを設定します。このフィールドのデフォルト値は「使用不可」です。このフィールドには、ユーザーが自分自身のアカウントを作成できるかどうかを指定します。このフィールドが「使用可能」の場合、「登録」ボタンがログイン画面に表示されます。ユーザー自己登録について詳しくは、『[新規ユーザー・アカウントの登録](#)』を参照してください。
 - 「シリアル・コンソールからのパスワードのリセットを許可する」フィールドを設定します。このフィールドのデフォルト値は「使用不可」です。

「使用不可:」SMTP サーバーおよびxcadmin ユーザーの E メール・アドレスを必ず構成してください。これらの構成により、xcadmin パスワードを忘れた場合でも、パスワードをリセットする方法が用意されます。このフィールドが使用不可で、かつこれらの構成が行われていない場合、忘れた xcadmin パスワードをリセットすることができず、再設定のためにアプライアンスを IBM に返却する必要があります。

「使用可能:」その他の資格情報を必要とせず、また SMTP メッセージが表示されることもなく、シリアル接続を使用して xcadmin ユーザーのパスワードをリセットできます。このオプションを選択している場合は、WebSphere DataPower XC10 アプライアンス への物理的アクセスが通常よりもさらに重要です。マシンへの物理的アクセスが可能であれば、すべてのユーザーがアプライアンスに対する管理者権限を取得できます。
- アプライアンスを Lightweight Directory Access Protocol (LDAP) ディレクトリーを使用してユーザーを認証するように構成します。LDAP ディレクトリーを使用した認証のためのアプライアンスの構成について詳しくは、『[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)』を参照してください。

タスクの結果

上記の手順を正常に終わると、アプライアンスでの特定のセキュリティー関連シナリオの扱い方、およびユーザー・インターフェースへのアクセスに外部認証を使用するかどうかの指定が完了します。

次のタスク

ユーザーとグループを構成してユーザー・インターフェースへのアクセスを提供します。また、ユーザーとグループを使用してデータ・グリッドへのアクセスを提供します。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成

IBM® WebSphere® DataPower® XC10 アプライアンス でユーザーを認証するために、任意で Lightweight Directory Access Protocol (LDAP) ディレクトリーを使用できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーの認証に LDAP サーバーを使用するのはオプションです。IBM WebSphere DataPower XC10 アプライアンスのユーザーを LDAP ディレクトリーで認証するために、「ユーザー名」属性が使用されます。主アプライアンス管理者は別にして、LDAP ディレクトリーに登録されていないユーザーを認証することはできません。

2.5+ SSL 接続経由で LDAP の認証を行うように WebSphere DataPower XC10 アプライアンス を構成することができます。これは、LDAP サーバーへの接続のための Lightweight Directory Access Protocol over SSL (LDAPS) URL を指定して行います。LDAPS を使用する場合は、LDAP サーバーの SSL 証明書が信頼されるようにアプライアンスのトラストストアを変更する必要があります。この証明書の発行元が認証局である場合は、その認証局のルート署名者証明書をアプライアンスのトラストストアに追加する必要があります。LDAP サーバーの SSL 証明書が自己署名証明書である場合は、その証明書自体をアプライアンスのトラストストアに追加する必要があります。

2.5+ LDAP 認証を構成している場合、構成した基本ドメイン・ネームとユーザー用の構成済み検索フィルターが一致している LDAP ユーザー ID であれば、アプライアンスへの認証ができます。各ユーザーは、その特定ユーザーに付与される許可とデータ・グリッド権限、さらにそのユーザーが所属する LDAP グループに付与される許可とデータ・グリッド権限を保有します。個々のユーザーを集合に追加する必要はありません。すべての LDAP ユーザーは、Everyone グループに付与されている許可とアクセス権限も保有します。個々のユーザーにデータ・グリッド権限と許可が構成されるようにするために、ユーザーを集合に追加することが必要な場合があります。

LDAP 認証を構成している場合、LDAP グループのみを集合に追加できます。アクセス権限と許可は、指定されたグループに対して付与できます。

LDAP 認証を構成している場合、アプライアンスの管理コンソールやアプライアンス・プログラミング・インターフェースを使用して、グループのメンバーを追加/削除することはできません。グループ・メンバーシップは、LDAP ディレクトリーの管理ツールを使用して管理します。

V2.5 より前のリリースの IBM WebSphere DataPower XC10 アプライアンス では、集合に明確に追加されたユーザーのみが、許可やアクセス権限を付与されます。汎用的な LDAP アクセスのサポートは V2.5 で追加されています。V2.5 より前のリリースでは、グループが集合に追加されるときに、アプライアンスが、LDAP からグループ・メンバーシップをインポートします。その後は、アプライアンス上でグループが保守されます。このため、メンバーシップは、LDAP に保管されていたものとは異なるものとなる可能性があります。V2.5 からは、LDAP 認証が構成されている場合のグループ・メンバーシップは、常に、LDAP サーバーに照会することで解決されます。

マイグレーションの考慮事項: 以前のファームウェア・バージョンを実行するアプライアンスと V2.5 が一緒に含まれる集合では、LDAP 内に存在するがアプライアンス集合に保管されていないユーザーは、以前のファームウェアを使用するデバイス上の制限付きリソースにアクセスできません。したがって、すべてのデバイスがアップグレードされるまで、クライアントは、アプライアンス集合に追加されているユーザー ID しか使用できません。

集合に V2.5 より古いファームウェア・バージョンを実行するメンバーが含まれている場合は、以前のアプライアンス上に保管されているグループ・メンバーシップが、LDAP に保管されているものとは異なる可能性があります。こうした矛盾が原因で問題が発生する可能性があります。例えば、ユーザー ID が以前のアプライアンスに保管されているグループに含まれ、許可やアクセス権限がそのグループに関連付けられているものの、そのグループが LDAP 内に存在しない場合、またはそのアプライアンスに保管されたグループのメンバーシップが、LDAP 内にあるものとは異なる場合、そのユーザー ID は、新しいアプライアンス上の制限付きリソースにアクセスできない可能性があります。こうした動作が起きるのは、V2.5 のアプライアンスが LDAP を直接検査し、グループ・メンバーシップのローカル・バージョンを検査していないためです。マイグレーションを行う場合は、アプライアンス・データのアクセスに使用するユーザー ID が、個々のユーザー ID に関連付けられた必要なアクセス権限や許可を持っていることを必ず確認してください。また、それらのユーザー ID が、必要な許可を持つ LDAP グループのメンバーになっていることも確認してください。

手順

1. 「設定」パネルにナビゲートします。次のいずれかの方式を使用します。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ようこそ」ページで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「セキュリティ」を展開します。
3. アプライアンスを LDAP ディレクトリーを使用してユーザーを認証するように構成します。
 - a. LDAP 認証を使用可能にするには、「LDAP 認証を使用可能に設定」の横にあるチェック・ボックスを選択

します。「**LDAP 認証を使用可能に設定**」チェック・ボックスは、デフォルトでは選択されていません。このチェック・ボックスを選択すると、WebSphere DataPower XC10 アプライアンス でログイン時にユーザーを認証するために指定された LDAP サーバーを使用できるようになります。

- b. JNDI プロバイダー URL を入力します。非 SSL LDAP の例:

```
ldap://mycompany.com:389/
```

または

```
ldap://mycompany.com/
```

ポートが明示的に指定されていない場合は、デフォルトのポート番号が 389 になります。SSL LDAP の例:

```
ldaps://mycompany.com:636/
```

または

```
ldaps://mycompany.com/
```

ポートが明示的に指定されていない場合は、デフォルトのポート番号が 636 になります。

- c. JNDI ベース DN (ユーザー) を入力します。例:

```
CN=users,DC=mycompany,DC=com
```

- d. JNDI ベース DN (グループ) を入力します。例:

```
DC=mycompany,DC=com
```

- e. 検索フィルター (ユーザー) を入力します。例:

```
(&(sAMAccountName={0})(objectcategory=user)) または uid={0}
```

注: ユーザー ID はプレースホルダー {0} に組み込まれます。{0} は、ログイン画面で入力したログイン・ユーザー ID で置き換えられます。

- f. JNDI セキュリティー認証を入力します。このフィールドは、LDAP サーバーで匿名 LDAP 照会の実行が許可されていない場合以外はオプションです。例:

```
CN=Administrator,CN=users,DC=mycompany,DC=com
```

- g. パスワードを入力します。このフィールドは JNDI セキュリティー資格情報であり、LDAP サーバーで匿名 LDAP 照会の実行が許可されていない場合以外はオプションです。

4. 構成した LDAP 認証設定をテストします。LDAP サーバーで認証を構成するために使用した設定をテストできます。このセクションでは、LDAP 照会を実行し、指定されたユーザーおよびグループを探すことができます。

- a. 「**LDAP 認証設定のテスト**」をクリックし、このセクションを展開します。
b. ユーザー名をテストするには、「LDAP ユーザー名」フィールドにユーザー名を入力し、関連する「**LDAP 照会のテスト**」ボタンをクリックします。例:

```
test_user@us.ibm.com
```

照会が正常に行われると、「*Found LDAP User DN: <ユーザー情報>*」というメッセージが表示されます。照会が正常に行われなかった場合は、エラー・メッセージが表示されます。

- c. グループ名をテストするには、「LDAP グループ名」フィールドにグループ名を入力し、関連する「**LDAP 照会のテスト**」ボタンをクリックします。例:

```
Test Group
```

照会が正常に行われると、「*Found LDAP Group DN: <ユーザー情報>*」というメッセージが表示されます。照会が正常に行われなかった場合は、エラー・メッセージが表示されます。

タスクの結果

ユーザー・インターフェースへアクセスする際の外部認証に LDAP ディレクトリーが指定されます。

次のタスク

ご使用の環境のさまざまな領域に対するユーザー・アクセスを制御する方法を理解することは、セキュリティー・ソリューションの重要な部分です。ユーザーおよびグループとその許可の管理方法について詳しくは、『[ユーザーおよびグループの管理](#)』を参照してください。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)
[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

ユーザーおよびグループの管理

ユーザーおよびユーザー・グループが用意されているため、WebSphere® DataPower® XC10 アプライアンス へのアクセス・レベルを 個人別に管理できます。ユーザー・グループを使用して、ユーザーのグループに許可を適用できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーおよびユーザー・グループは、WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用して管理できます。

[ユーザーの作成](#)

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ユーザーが WebSphere DataPower XC10 アプライアンス にアクセスし、これを管理できるようにするには、以下の手順で新規アカウントを作成します。

[ユーザーの管理](#)

ユーザーの作成後に追加の権限が必要になった場合、ユーザー設定を手動で変更する必要があります。また、この手順は、情報が変わった場合に、ユーザーを変更するためにも使用できます。

[ユーザーの除去](#)

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ただし、特定のユーザーが不要になった場合は、WebSphere DataPower XC10 アプライアンス からそのユーザーを除去できます。

[新規ユーザー・アカウントの登録](#)

管理者が「[新規ユーザーに自分のアカウントの作成を許可する](#)」オプションを使用可能にしている場合、自分自身のユーザー・アカウントを作成できます。このタスクを使用して、ユーザー・アカウントをログイン画面から作成します。

[ユーザー・グループの作成](#)

特定の WebSphere DataPower XC10 アプライアンス リソースに対するユーザーの許可の管理を容易にするために、ユーザー・グループを作成できます。

[ユーザー・グループの管理](#)

ユーザー・グループを作成した当初は、このユーザー・グループにはメンバーとして指定されたユーザーは存在しません。LDAP 認証が使用可能でない場合、手動でユーザーをグループに追加する必要があります。

[ユーザー・グループの除去](#)

不要になったユーザー・グループは WebSphere DataPower XC10 アプライアンス から除去できます。

[ユーザー許可](#)

ユーザー許可は、各ユーザーに向けて表示できるパネルを決定したり、特定オブジェクトに対するユーザー・アクセスを決定したりするために定義します。

親トピック: [アプライアンスの構成](#)

親トピック: [セキュリティ](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[セキュリティ](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティの構成](#)

[データ・グリッド のセキュリティを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

ユーザーの作成

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ユーザーが WebSphere® DataPower® XC10 アプライアンス にアクセスし、これを管理できるようにするには、以下の手順で新規アカウントを作成します。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。 Lightweight Directory Access Protocol (LDAP) を使用してユーザーを認証する場合、まず、登録しようとしているユーザーが LDAP リポジトリに存在していなければなりません。LDAP 構成の作成について詳しくは、『[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)』を参照してください。

このタスクについて

WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用してユーザーを作成するには、以下の手順に従ってください。

手順

1. 「ユーザー」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「集合」 > 「ユーザー」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「ユーザーの作成」リンクをクリックします。
2. 追加アイコン (+) をクリックして、新規ユーザーの追加を開始します。
3. 「ユーザー名」フィールドに ID を入力します。このフィールドの値は 64 文字まで指定可能です。このフィールドをブランクにすることはできません。すべての英数字と大部分の特殊文字を使用することができます。スペースと特殊文字 < # は使用できません。このフィールドは、ユーザーの作成後には変更できません。LDAP 認証を使用する場合、登録しようとしているユーザーは LDAP リポジトリに存在していなければなりません。
4. オプション: 「フルネーム」フィールドにユーザーの名前を入力します。このフィールドは、ユーザー・インターフェースでの表示用に使用されます。このフィールドに値を入力しなければ、ユーザー名が表示されます。ユーザーを作成した後は、そのユーザーのみがこのフィールドを編集できます。ユーザーが作成された後、管理者はこのフィールドの値を変更できません。
5. 「パスワード」フィールドにユーザーのパスワードを入力します。パスワードは、「ユーザー名」フィールドに使用できる文字と同じ文字を使用できます。Simple Mail Transfer Protocol (SMTP) が使用可能であれば、ユーザーの作成時に「パスワード」フィールドは空のままにすることができ、パスワードは自動的に生成されます。LDAP 認証が使用可能になっている場合は、LDAP レジストリーのパスワードが認証に使用されるため、「パスワード」フィールドは表示されません。「パスワードの確認」フィールドに、上と同じユーザーのパスワードを再入力します。
6. 「E メール・アドレス」フィールドにユーザーの有効な E メール・アドレスを入力します。このフィールドには、ユーザーがパスワードを忘れた場合の新規パスワードの供給や、その他の通知の配信に使用される E メール・アドレスを指定します。Simple Mail Transfer Protocol (SMTP) サーバーを使用するときに E メール・アドレスが必要です。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)を参照してください。
7. 「OK」をクリックします。

タスクの結果

これで、ユーザー・インターフェースへのログオンに使用できる新規ユーザー・アカウントができました。

次のタスク

ユーザーの作成当初は、デフォルトの権限のみが割り当てられています。新規ユーザーに追加の許可を与える場合は、ユーザーへの許可の追加について『[ユーザーの管理](#)』を参照してください。新規ユーザーをユーザー・グループに追加する方法については、『[ユーザー・グループの管理](#)』を参照してください。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザーの管理

ユーザーの作成後に追加の権限が必要になった場合、ユーザー設定を手動で変更する必要があります。また、この手順は、情報が変わった場合に、ユーザーを変更するためにも使用できます。




始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーを作成した時点では、デフォルト権限のみが割り当てられています。ユーザー・アカウントに追加の権限が必要である場合は、最初のユーザー作成後に、これらの権限を手動で追加する必要があります。ユーザー・アカウントが自己登録機能を使用して作成された場合は、ユーザー情報のサブセットのみが使用可能になります。残りの情報は、アプライアンス管理権限が割り当てられているユーザーが追加する必要があります。WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェースを使用してユーザーを変更するには、以下の手順に従ってください。

手順

1. 「ユーザー」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「集合」 > 「ユーザー」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「ユーザーの作成」リンクをクリックします。
2. 変更するユーザーの「ユーザー名」をクリックします。ユーザーの作成後は、表示名およびユーザー名を変更できません。
3. ユーザーのパスワードおよび E メール・アドレスを編集することができます。パスワードを変更するには、フィールドの【編集】をクリックします。新規パスワードを入力し、パスワードを変更します。
4. ユーザー・アクティビティを表示します。以下のユーザーの活動状況をユーザー・アカウント画面に表示できます。
 - **現在の状況:** このフィールドには、ユーザーの状況が示されます。以下のリストには、起こり得るユーザー状況が含まれています。
 - : 最近 5 分間はアクティブ
 - : 5 分を超えて非アクティブ
 - : ユーザーがまだログインしていません
 - **ユーザー・グループ:** このフィールドには、ユーザーがメンバーになっているすべてのユーザー・グループがリストされます。

グループ名を入力し、ユーザーをグループに追加します。ユーザー・グループ名を入力すると、入力内容に一致するユーザー・グループのリストが表示されます。ユーザーをグループに追加するには、ユーザー・グループ名をクリックしてください。ユーザー・グループ名を入力するだけでは、ユーザーはグループに追加されません。ユーザーをユーザー・グループに追加すると、ユーザー・グループの権限がユーザーに割り当てられます。ユーザーに割り当てられていた以前の権限レベルは保持されません。ユーザーをユーザー・グループから除去する場合は、ユーザーを除去するグループの隣にある【除去】リンクをクリックします。ユーザーは、(Everyone グループ以外の) すべてのグループから除去されると、最後に除去されたグループに割り当てられていた権限を保持します。

5. このユーザーの権限を変更します。ユーザーに割り当てられるアクセス・レベルを制御するために、権限を選択またはクリアできます。ユーザーがグループ (Everyone グループは含めません) のメンバーの場合は、ユーザー権限は変更できません。ユーザーがグループのメンバーの場合、そのグループで定義された権限をユーザーは持ちます。ユーザーが複数グループのメンバーの場合、それらのグループで定義された権限の合計をユーザーは持ちます。グループに定義されている権限を変更すると、その変更は、そのグループのすべてのメンバーに伝搬します。ユーザーごとに選択可能な権限は、以下のとおりです。
 - アプライアンス管理
 - アプライアンス・モニター
 - データ・グリッドの作成これらの権限設定について詳しくは、『[ユーザー許可](#)』を参照してください。

タスクの結果

ユーザー・アカウントは正常に変更されています。

次のタスク

ユーザーの変更後、このユーザーをユーザー・グループに追加できます。ユーザー・グループの作成方法、およびユーザーをユーザー・グループに追加する方法については、『[ユーザー・グループの作成](#)』および『[ユーザー・グループの管理](#)』を参照してください。Lightweight Directory Access Protocol (LDAP) サーバーを認証用を使用して、セキュリティーのレイヤーをアプライアンスに追加できます。LDAP サーバーによるアプライアンスの保護について詳しくは、

『[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)』を参照してください。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザーの除去

ユーザー・インターフェースにログインするには、ユーザー名とパスワードが必要です。ただし、特定のユーザーが不要になった場合は、WebSphere® DataPower® XC10 アプライアンス からそのユーザーを除去できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザーを削除すると、そのユーザーが所有していたすべてのリソースが、自動的に実行者に転送されます。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用してユーザー・アカウントをアプライアンスから除去するには、以下の手順に従ってください。

手順

1. 「ユーザー」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「集合」 > 「ユーザー」にナビゲートします。
 - 「ホーム」ページで、「ステップ 1: アプライアンスのセットアップ」セクションにある「ユーザーの作成」リンクをクリックします。
2. ユーザーの **<user_name>** をクリックして、削除するユーザーを選択します。

注: 「管理者」ユーザー・アカウント xadmin は削除できません。
3. 「削除」アイコン (✖) をクリックして、ユーザーの削除を開始します。メッセージ・ボックスが表示され、このユーザーが完全に削除されることに対する確認が要求されます。
4. 「OK」をクリックします。

タスクの結果

これで、アプライアンスからユーザー・アカウントが除去されました。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

新規ユーザー・アカウントの登録

管理者が「新規ユーザーに自分のアカウントの作成を許可する」オプションを使用可能にしている場合、自分自身のユーザー・アカウントを作成できます。このタスクを使用して、ユーザー・アカウントをログイン画面から作成します。

始める前に

新規ユーザー・アカウントを登録するには、「新規ユーザーに自分のアカウントの作成を許可する」フィールドが「使用可能」に設定されている必要があります。ユーザー登録を使用可能にする方法については、『[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)』を参照してください。

手順

1. WebSphere DataPower XC10 アプライアンス のログイン画面にアクセスします。
2. 「登録...」 ボタンをクリックして、ユーザー・アカウントの作成を開始します。
3. 「ユーザー名」フィールドにログイン ID を入力します。このフィールドに入力された値は、ユーザー名およびこのユーザーの表示名の両方に使用されます。このフィールドの値は 64 文字まで指定可能です。すべての英数字を使用できます。特殊文字 !@#%^*-&+= も使用可能です。

重要: Lightweight Directory Access Protocol (LDAP) を使用してユーザーを認証する場合は、登録されるユーザーは LDAP リポジトリに存在していなければなりません。LDAP 構成の作成については、『[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)』を参照してください。

4. 「パスワード」フィールドにユーザーのパスワードを入力します。パスワードは、「ユーザー名」フィールドに使用できる文字と同じ文字を使用できます。Simple Mail Transfer Protocol (SMTP) または LDAP サーバーが定義されていない場合は、「パスワード」フィールドは必須です。SMTP が使用可能であれば、パスワードを入力するか、あるいは、そのフィールドを空白のままにして生成されたパスワードを E メール・アドレスに送信するようにするかを選択できます。LDAP を使用してユーザーを認証する場合、既存の LDAP パスワードが使用され、パスワードを入力する必要はありません。
5. 「パスワードの確認」フィールドに、上と同じユーザーのパスワードを再入力します。このフィールドに入力する値は、「パスワード」フィールドに入力した値と同じでなければなりません。これらのフィールドの値が一致しない場合、「登録」をクリックするとエラーが表示されます。ユーザーを作成するためには、このエラーを解決する必要があります。
6. 「E メール・アドレス」フィールドに E メール・アドレスを入力します。Simple Mail Transfer Protocol (SMTP) サーバーを使用するときに E メール・アドレスが必要です。詳しくは、[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)を参照してください。
7. 「登録」ボタンをクリックして、登録プロセスを完了します。

タスクの結果

上記の手順が正常に完了すると、ユーザー・インターフェースへのログインに使用できるユーザー・アカウントが登録されます。デフォルトでは、アプライアンス・モニター権限のみが割り当てられます。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザー・グループの作成

特定の WebSphere® DataPower® XC10 アプライアンス リソースに対するユーザーの許可の管理を容易にするために、ユーザー・グループを作成できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ユーザー・グループを作成して、ユーザーのコレクション (集合) の許可をリソースまたはリソースのグループに迅速に割り当てます。ユーザー・グループは、最初に作成した時点では空です。そのため、ユーザーを新規ユーザー・グループにそれぞれ手動で追加する必要があります。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用してユーザーを作成するには、以下の手順に従ってください。

手順

1. 「集合」 > 「ユーザー・グループ」にナビゲートします。
2. 追加アイコン (+) をクリックして、グループを作成します。
3. 「グループ名」フィールドに名前を入力します。このフィールドの値は 64 文字まで指定可能です。このフィールドを空白にすることはできません。すべての英数字と、特殊文字 !@#%^*-&-+= を使用できます。LDAP 認証が構成されている場合、LDAP 内に存在するグループのみを集合に追加することができます。
4. 「説明」フィールドにその他の追加情報を入力します。このフィールドは、ユーザー・グループに関する詳細を付加するために使用できます。
5. 「OK」をクリックします。

タスクの結果

上記の手順が正常に完了すると、WebSphere DataPower XC10 アプライアンス ユーザーの許可の管理に役立つ 新規ユーザー・グループが作成されます。

次のタスク

作成したグループにユーザーを追加できます。ユーザーをユーザー・グループに追加する手順については、『[ユーザー・グループの管理](#)』を参照してください。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

ユーザー・グループの管理

ユーザー・グループを作成した当初は、このユーザー・グループにはメンバーとして指定されたユーザーは存在しません。LDAP 認証が使用可能でない場合、手動でユーザーをグループに追加する必要があります。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェースを使用して、ユーザーをユーザー・グループに追加するか、またはユーザー・グループから除去するには、以下の手順に従ってください。

手順

1. 「集合」 > 「ユーザー・グループ」にナビゲートします。
2. 「<group_name>」をクリックして、変更するグループを選択します。
3. ユーザー・グループの説明を変更したい場合は、既存の説明をクリックして、変更内容を入力します。
4. ユーザーをグループに追加する場合は、追加するユーザーを入力して、その「<user_name>」をクリックします。

ユーザー名を入力すると、入力内容に一致するユーザーのリストが表示されます。ユーザーをグループに追加するには、ユーザー名をクリックしてください。ユーザー名を入力するだけでは、ユーザーはグループに追加されません。ユーザーをユーザー・グループに追加すると、ユーザー・グループの権限がユーザーに割り当てられます。ユーザーに割り当てられていた以前の権限レベルは保持されません。

注: LDAP 認証が使用可能な場合、WebSphere DataPower XC10 アプライアンス 内のグループのメンバーシップを変更することはできません。

5. グループに割り当てられた権限を変更します。

ユーザー・グループに適用可能な権限は以下のとおりです。

- アプライアンス管理
- アプライアンス・モニター
- データ・キャッシュの作成

これらの権限設定について詳しくは、『[ユーザー許可](#)』を参照してください。

6. ユーザーをグループから除去する場合は、除去するユーザーの隣にある【除去】リンクをクリックします。ユーザーを除去するときに確認は要求されません。したがって、ユーザー・グループを管理するときは、十分な注意が必要です。ユーザーは、Everyone グループ以外のすべてのグループから除去されると、最後に除去されたグループに割り当てられていた権限を保持します。

タスクの結果

これで、ユーザー・グループの変更が完了しました。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの除去](#)

ユーザー・グループの除去

不要になったユーザー・グループは WebSphere® DataPower® XC10 アプライアンス から除去できます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

この手順を実行すると、WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースを使用して、ユーザー・グループをアプライアンスから除去できます。

手順

1. 「集合」 > 「ユーザー・グループ」をクリックします。
2. **<user_group_name>**をクリックして、除去するユーザー・グループを選択します。
注: 「全員」ユーザー・グループは除去できません。
3. 「除去」アイコン (✕) をクリックして、グループの除去を開始します。
4. 選択されたユーザー・グループを除去することを確認して、「**OK**」をクリックします。

タスクの結果

各ユーザー・グループ・メンバーがグループから除去され、ユーザー・グループが削除されます。

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[ユーザー許可](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

ユーザー許可

ユーザー許可は、各ユーザーに向けて表示できるパネルを決定したり、特定オブジェクトに対するユーザー・アクセスを決定したりするために定義します。

ユーザーに割り当てた許可によって、そのユーザーが実行可能な WebSphere® DataPower® XC10 アプライアンス管理用タスクが定義されます。表示される管理ページが決定され、さらに「ようこそ」ページの内容が、ユーザーに割り当てたアクセス・レベルに応じて表示されるように、動的に生成されます。ユーザーが最初に登録する時に、アプライアンスのモニター権限が与えられます。すべてのアプライアンス管理者は、データ・グリッド (data grid) の作成権限またはアプライアンス管理権限を割り当てる必要があります。

表 1. 各許可レベルの表示可能パネル

	アプライアンス・モニター権限	データ・グリッド作成権限	アプライアンス管理権限
ウェルカム・ページ	はい	はい	はい
データ・グリッドの作成	いいえ	はい	はい
データ・グリッドの削除	いいえ	はい	はい
モニター・メニューの表示	はい	いいえ	はい
タスクの表示	はい	はい	はい
2.5+ 完了したタスクの削除	いいえ	いいえ	はい
集合およびゾーンの表示と作成	いいえ	いいえ	はい
アプライアンスの設定の変更	いいえ	いいえ	はい
ユーザーおよびグループの変更	いいえ	いいえ	はい
2.5+ データ管理の表示	はい	いいえ	はい
2.5+ メッセージ・センターの表示	はい	いいえ	はい

親トピック: [ユーザーおよびグループの管理](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)
[xcadminパスワード](#)

関連タスク:

[ユーザーの作成](#)

[ユーザーの管理](#)

[ユーザーの除去](#)

[新規ユーザー・アカウントの登録](#)

[ユーザー・グループの作成](#)

[ユーザー・グループの管理](#)

[ユーザー・グループの除去](#)

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理

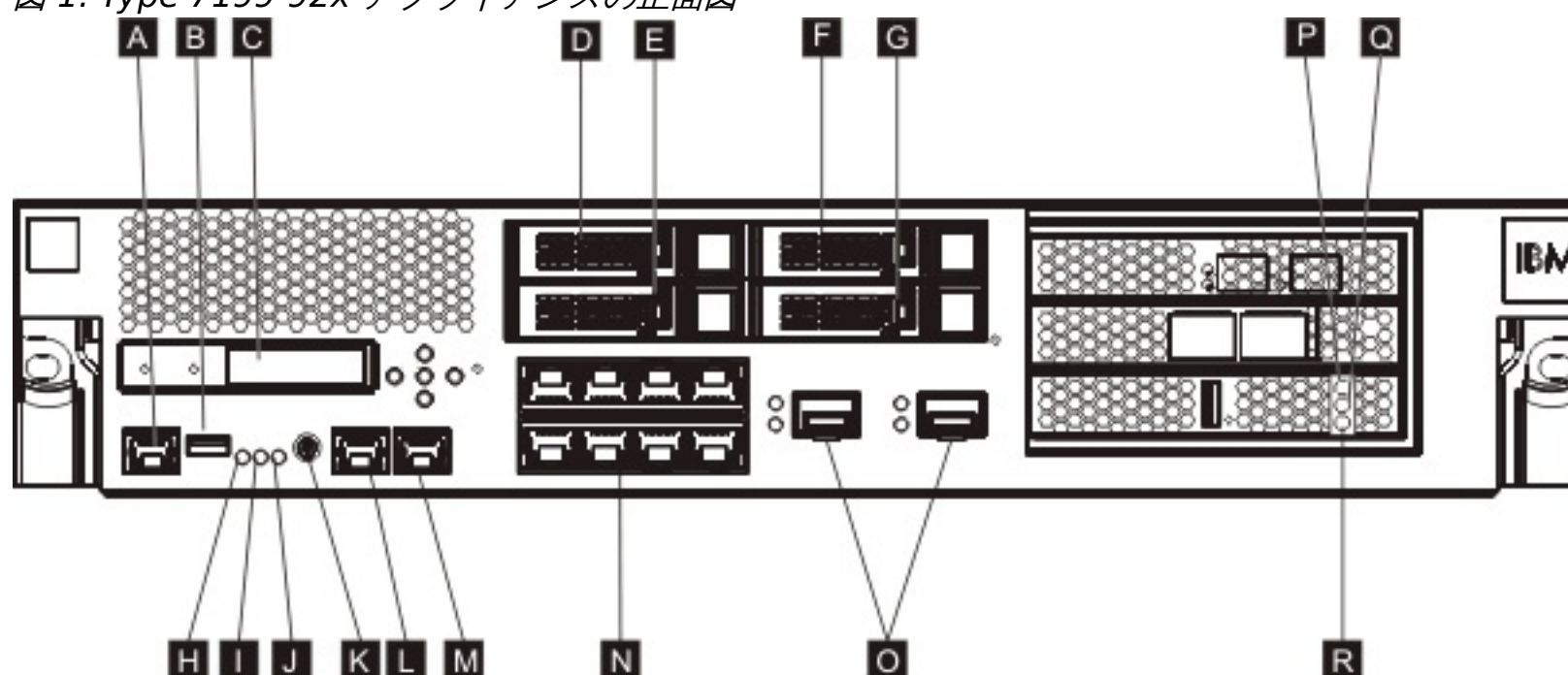
初回のシリアル接続時に、アプライアンスをネットワークに接続するために *mgmt* イーサネット・インターフェースを構成しました。追加のプライベート・イーサネット・ポートをユーザー・インターフェースで定義することができます。

始める前に

アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

図 1. Type 7199-92x アプライアンスの正面図



- mgmt0 (L)
- mgmt1 (M)
- 左方 1 ギガビット・イーサネット・モジュール eth0-eth7 (N)
- 右方 10 ギガビット・イーサネット・モジュール eth8-eth9 (O)

Type 7199-92x アプライアンスでは、データ・グリッドに 1 ギガビット・イーサネット・ポートまたは 10 ギガビット・イーサネット・ポートのいずれかを使用します。使用するポートは、すべて 1 ギガビット・ポートであるか、すべて 10 ギガビット・ポートでなければなりません。初期構成完了後は、ポート・タイプを変更できません。MGMT0 (L) の管理ポートに接続します。

スタンドアロンのアプライアンスのイーサネット・インターフェースを編集する場合は、設定の変更後、構成をクリアし、アプライアンスを再始動する必要があります。アプライアンスが集合内にある場合、イーサネット・インターフェースは更新できません。

手順

1. スタンドアロンのアプライアンスのイーサネット・インターフェースを編集します。「設定」パネルにナビゲートします。
 - WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「イーサネット・インターフェース」を展開します。
3. 「使用可能」チェック・ボックスを選択またはクリアして、イーサネット・インターフェースを使用可能または使用不可にします。 *mgmt* インターフェースは使用不可にできません。
4. 「IP アドレス/マスク」を変更します。 <ip_address>/<subnet_mask> という形式で IP アドレスと サブネット・マスクを入力します。サブネット・マスクは、クラスレス・ドメイン間ルーティング (CIDR) 表記を使用して入力する必要があります。例: 長い表記での 255.255.255.0 は CIDR 表記では 24 です。
5. 「デフォルト・ゲートウェイ」を変更します。アプライアンスは、宛先ベースのルーティングではなく、送信元ベースのルーティングを使用します。パケットは、パケットが受信されたものと同じインターフェース上の宛先へ送信されます。各インターフェースには、他のインターフェースから分離された、専用のルーティング・テーブルがあります。ローカル・サブネット経由で宛先に到達する必要があるインターフェースごとに、そのインターフェースから直接到達可能なデフォルトの経路を指定してください。
6. 指定した IP アドレスがプライベート IP アドレスであるかどうかを指定します。
7. 「最大伝送単位 (MTU)」を変更します。このフィールドには、イーサネット・インターフェースを使用した通信時に、プロトコル・データ単位で可能な最大サイズ (バイト単位) を指定します。デフォルト値は 1500 バイトで、これは、このフィールドの最大許容値でもあります。

8. 「モード」を変更します。イーサネット・インターフェースでは以下のイーサネット・モードを使用できます。
 - 自動
 - 10baseT-HD
 - 10baseT-FD
 - 100baseTx-HD
 - 100baseTx-FD
 - 1000baseTx-FD
9. チェック・ボックス「**使用可能**」を選択することで、構成をクリアし、アプライアンスを再始動します。このアクションによって構成がクリアされます。WebSphere DataPower XC10 アプライアンス プロセスがイーサネット・インターフェースにバインドするためには、再始動が必要です。

次のタスク

イーサネット・インターフェースの状況をモニターするには、[シリアル接続を使用したイーサネット・インターフェースの状況詳細のモニター](#)を参照してください。

[集約インターフェースの追加](#)

WebSphere DataPower XC10 アプライアンス で集約インターフェースを構成することができます。

[集約インターフェースの編集](#)

集約インターフェースのメンバーの追加と削除、および IP アドレス、デフォルト・ゲートウェイ、最大伝送単位 (MTU) などのプロパティの変更を行えます。集約ポリシーを変更することもできます。

[集約インターフェースの削除](#)

集約インターフェースを削除することができます。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集約インターフェースの追加

WebSphere® DataPower® XC10 アプライアンス で集約インターフェースを構成することができます。

始める前に

以下のステップを実行するには、WebSphere DataPower XC10 アプライアンス 管理権限が割り当てられている必要があります。

このタスクについて

複数のネットワーク・インターフェースを 1 つの集約インターフェースにリンクできます。1 つの集約インターフェースが 1 つ以上のイーサネット・インターフェースで構成され、単一の論理装置として機能します。この機能を使用して、ネットワーク上でトラフィックを分散することができます。集約インターフェースの作成は、コマンド行インターフェースまたは Web コンソールを使用して行うことができます。集約インターフェースの推奨される作成方法は、コマンド行インターフェースを使用する方法です。いずれの場合も、アプライアンスにログインしてから行う必要があります。操作が完了したら、**clear-all** コマンドを実行し、アプライアンスを再始動します。

スイッチ間のリンク集約により、接続性、冗長性が増加し、処理能力が拡大します。

- 処理能力の増大

複数のイーサネット・インターフェースが、集約インターフェースとして、1 つの論理リンクに結合されて、これにより、アプライアンスが使用できる処理能力が増大します。

要確認: アプライアンスでは、1 Gbps ポートまたは 10 Gbps ポートのいずれかを使用可能に設定できます。すなわち、10 Gbps イーサネット・ポートを使用可能に設定する場合、10 Gbps ポートを集約することもできます。

- 自動フェイルオーバー

集合内で障害が起こったイーサネット・ポートのトラフィックは、作動しているイーサネット・ポートに転送されます。単一のスイッチを使用すると、これによりポートの障害を防ぐことができます。また、予備のスイッチを使用することで、ネットワーク・スイッチの障害に対して防御することも可能です。

- ロード・バランシング

ロード・バランシング・ポリシーを選択して、インバウンド・トラフィックとアウトバウンド・トラフィックの両方を分散することができます。

- 冗長性のサポート

2 つのシステムを、さまざまな集約構成の複数のリンクによって接続することができます。

- 管理の改善

管理者は、複数のイーサネット・インターフェースを単一の集約インターフェース・ユニットとして管理できます。

手順

- コマンド行インターフェースを使用して集約インターフェースを追加する場合。次の例を使用して集約インターフェースを作成します。

```
Console> create aggregate-interface myAgg
Console aggregate-interface:myAgg> member eth3 eth4
Console aggregate-interface:myAgg> ip
CWZBR02205I: Entering "ip" mode
Console aggregate-interface:myAgg ip> address 10.5.5.5/24
Console aggregate-interface:myAgg ip> exit
Console aggregate-interface:myAgg> primary-member eth3
Console aggregate-interface:myAgg> aggregation-policy balance-tlb
Console aggregate-interface:myAgg> show
aggregate-interface myAgg:
name "myAgg"
AdminState "Enabled"
use-arp "true"
mtu "1500"
ip
  use-dhcp "false"
  address "10.5.5.5/24"
  use-slaac "false"
```

```
dad-transmits "1"
dad-retransmit-timer "1000"
end ip
aggregation-policy "balance-tlb"
lacp-selection-logic "stable"
transmit-hash-policy "layer2"
member "eth3" "eth4"
primary-member "eth3"

Console aggregate-interface:myAgg> exit
```

- Web コンソールを使用して集約インターフェースを追加する場合。次のいずれかの方法を使用して「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
1. 「集約インターフェース」を展開します。
 2. 「新規集約インターフェースの追加」をクリックします。
 3. 追加する集約インターフェースを記述するためのフォームに入力します。

名前

インターフェースの名前を指定します。

IP アドレス/マスク

集約インターフェースに割り当てる IP アドレスを指定します。 <ip_address>/<subnet_mask> という形式で IP アドレスと サブネット・マスクを入力します。

ヒント: IPv4 アドレスと IPv6 アドレスは、いずれもこのプロパティを使用して割り当てられます。アドレスの重複は許可されません。

集約ポリシー

アクティブ・バックアップ

高可用性を求める場合は、アクティブ・バックアップ・ポリシーを指定します。このポリシーを使用すると、集約インターフェースの 1 メンバーとして同時に使用可能になるのは、1 つのイーサネット・インターフェースのみです。そのイーサネット・インターフェースに障害が発生した場合、別のメンバーが処理を再開します。

LACP

高可用性と処理能力の両方を求める場合は、Link Aggregation Control Protocol (LACP) ポリシーを指定します。LACP ポリシーは、そのモードが「OFF」に設定されていない場合にのみ使用できます。デフォルトで、選択ロジックは「安定」で、送信ハッシュ・モードは「layer2」です。

balance-tlb

ロード・バランシングと高可用性を求める場合は、balance-tlb ポリシーを指定します。このポリシーは、各メンバーの現在の負荷に応じてアウトバウンド・トラフィックを分散します。インバウンド・トラフィックは、基本メンバーとして選択されているイーサネット・インターフェースに送信されます。受信中のイーサネット・インターフェースに障害が発生した場合、別のメンバーが処理を再開します。

メンバー

集約インターフェースに追加するイーサネット・インターフェースを指定します。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

制約事項: イーサネット・インターフェースは、複数の集約インターフェースのメンバーになることはできません。

重要: イーサネット・インターフェースを集約インターフェースに追加するには、そのインターフェースが必ず使用不可の状態になっていなければなりません。イーサネット・インターフェースを使用不可にするには、「イーサネット・インターフェース」パネル上の「使用可能」チェック・ボックスのチェックを外す必要があります。このパネルへのナビゲートの方法については、[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)を参照してください。

4. 「OK」をクリックします。

重要: 集約インターフェースは、デフォルトで使用可能になっています。新規集約インターフェースを使用せずに、既存の集約インターフェースの使用を継続する場合、まず新規のインターフェースを無効にする必要があります。集約インターフェースを無効にするには、集約インターフェースを展開して、追加したばかりの集約インターフェースを見つけます。「使用可能」チェック・ボックスのチェックを外します。新規集約インターフェースを使用可能のままにすると、次に WebSphere DataPower XC10 アプライアンスを再始動する際に、その新規集約インターフェースがデータ・グリッドのバインドに使用されるため、予

期しない結果につながるおそれがあります。

注: 集約インターフェースを変更した場合は常に、WebSphere DataPower XC10 アプライアンス が変更内容を使用する前に再始動が必要です。しかし、変更を有効にするために clear-all コマンドを発行し、次に device-restart コマンドを発行する必要があります。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集約インターフェースの編集

集約インターフェースのメンバーの追加と削除、および IP アドレス、デフォルト・ゲートウェイ、最大伝送単位 (MTU) などのプロパティの変更を行えます。集約ポリシーを変更することもできます。

このタスクについて

集約インターフェースの編集は、コマンド行インターフェースまたは Web コンソールを使用して行うことができます。集約インターフェースの推奨される編集方法は、コマンド行インターフェースを使用する方法です。いずれの場合も、アプライアンスにログインしてから行う必要があります。操作が完了したら、**clear-all** コマンドを実行し、アプライアンスを再始動します。

手順

- コマンド行インターフェースを使用して集約インターフェースを編集する場合。次の例を使用して集約インターフェースを編集します。

```
Console aggregate-interface:myAgg> aggregation-policy lacp
Console aggregate-interface:myAgg> transmit-hash-policy layer3+4
Console aggregate-interface:myAgg> exit
Console> show aggregate-interface myAgg
aggregate-interface myAgg: [Up]

name "myAgg"
AdminState "Enabled"
use-arp "true"
mtu "1500"
ip
  use-dhcp "false"
  address "10.5.5.5/24"
  use-slaac "false"
  dad-transmits "1"
  dad-retransmit-timer "1000"
end ip
aggregation-policy "lacp"
lacp-selection-logic "stable"
transmit-hash-policy "layer3+4"
member "eth3" "eth4"
primary-member "eth3"
```

- Web コンソールを使用して集約インターフェースを編集する場合。次のいずれかの方法を使用して「設定」パネルにナビゲートします。
 - WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」>「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。

1. 「集約インターフェース」を展開します。
2. 既存の集約インターフェースのメンバーを追加したり、削除したりできます。この集約のメンバーであるイーサネット・インターフェースのリストが表示されます。

ヒント: メンバーが 1 つのみの場合は削除できません。また、基本メンバーも削除することはできません。

重要: イーサネット・インターフェースを集約インターフェースに追加するには、そのインターフェースが必ず使用不可の状態になっていなければなりません。イーサネット・インターフェースを使用可能もしくは使用不可に設定する方法について詳しくは、[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#) を参照してください。

- メンバーを追加するには、「さらに追加...」をクリックしてください。
 - イーサネット・インターフェースを選択して、「OK」をクリックします。
3. `<ip_address>/<subnet_mask>` という形式で IP アドレスとサブネット・マスクを変更します。サブネット・マスクは、クラスレス・ドメイン間ルーティング (CIDR) 表記を使用して入力する必要があります。例えば、長い表記での `255.255.255.0` は、CIDR 表記では `24` です。
 4. 「デフォルト・ゲートウェイ」を変更します。アプライアンスは、宛先ベースのルーティングではなく、送信元ベースのルーティングを使用します。パケットは、パケットが受信されたものと同じインターフェース上の宛先へ送信されます。各インターフェースには、他のインターフェースから分離された、専用のルーティング・テーブルがあります。ローカル・サブネット経由で宛先に到達する必要があるインターフェースごとに、そのインターフェースから直接到達可能なデフォルトの経路を指定してください。

5. 「**最大伝送単位 (MTU)**」を変更します。このフィールドには、イーサネット・インターフェースを使用した通信時に、プロトコル・データ単位で可能な最大サイズ (バイト単位) を指定します。デフォルト値は 1500 バイトで、これは、このフィールドの最大許容値でもあります。
6. 「**編集**」をクリックして集約ポリシーを変更します。変更する集約ポリシーに応じて、以下のプロパティが表示されます。

集約ポリシー

アクティブ・バックアップ

高可用性を求める場合は、アクティブ・バックアップ・ポリシーを指定します。このポリシーを使用すると、一度に使用可能になるのは、(集約インターフェースの 1 メンバーである) 1 つのイーサネット・インターフェースのみです。そのイーサネット・インターフェースに障害が発生した場合、別のメンバーが引き継ぎます。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

基本メンバー

基本メンバーとして指定するイーサネット・インターフェースを特定します。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

集約ポリシー

LACP

高可用性と処理能力の両方を求める場合は、Link Aggregation Control Protocol (LACP) ポリシーを指定します。LACP ポリシーは、そのモードが「OFF」に設定されていない場合にのみ使用できます。デフォルトでは、選択ロジックは「安定」で、送信ハッシュ・モードは「layer2」です。

選択ロジック

- 安定

最大の処理能力を持つイーサネット・インターフェースを指定します。安定設定を選択すると、有効な集約インターフェースに使用可能なメンバーがないときに、このイーサネット・インターフェースが再度選択されます。安定設定はデフォルトです。

- 処理能力

最大の処理能力を持つイーサネット・インターフェースを指定します。別のメンバーが追加/削除されるか、あるいは、集約インターフェースが使用可能/使用不可に設定されると、このイーサネット・インターフェースが再度選択されます。

- Count

メンバーとしてのイーサネット・インターフェース数が最も多い集約インターフェースを指定します。

ハッシュ・ポリシー

- layer2

ハッシュを生成する場合は、MAC アドレスの排他 OR (XOR) 結果を指定します。

- layer2+3

ハッシュを生成する場合は、MAC アドレスと IP アドレスの XOR 結果を指定します。

- layer3+4

ハッシュを生成する場合は、IP アドレスとポート番号の XOR 結果を指定します。

集約ポリシー

balance-tlb

ロード・バランシングと高可用性を求める場合は、balance-tlb ポリシーを指定します。このポリシーは、各メンバーの現在の負荷に応じてアウトバウンド・トラフィックを分散します。インバウンド・トラフィックは、基本メンバーとして選択されているイーサネット・インターフェースに常に送信されます。受信中のイーサネット・インターフェースに障害が発生した場合、別のメンバーが引き継ぎます。

基本メンバー

基本メンバーとして指定するイーサネット・インターフェースを特定します。デフォルトでは、リストに追加された最初のイーサネット・インターフェースが基本メンバーとして指定されます。

7. 「**OK**」をクリックします。

重要: 集約インターフェースを編集する際、アプライアンスを再始動するまで変更は有効になりません。WebSphere DataPower XC10 アプライアンス プロセスがイーサネット・インターフェースにバインドするためには、再始動が必要です。変更した集約インターフェースを使用不可に設定してから、WebSphere DataPower XC10 アプライアンス を再始動する必要があります。インターフェースを使用不可に設定す

るには、「集約インターフェース」を展開して、変更したばかりの集約インターフェースを見つけます。
「使用可能」チェック・ボックスのチェックを外します。集約インターフェースを使用不可にした後、コマンド行インターフェースで clear-all コマンドと device-restart コマンドを実行する必要があります。これらのコマンドを実行しないと、変更が有効になりません。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

関連タスク:

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集約インターフェースの削除

集約インターフェースを削除することができます。

このタスクについて

集約インターフェースを削除は、コマンド行インターフェースまたは Web コンソールを使用して行うことができます。集約インターフェースの推奨される削除方法は、コマンド行インターフェースを使用する方法です。いずれの場合も、アプライアンスにログインしてから行う必要があります。操作が完了したら、**clear-all** コマンドを実行し、アプライアンスを再始動します。

手順

- コマンド行インターフェースを使用して集約インターフェースを削除する場合。次の例を使用して集約インターフェースを削除します。

```
Console> delete aggregate-interface myAgg
Console> list aggregate-interface
```

- Web コンソールを使用して集約インターフェースを削除する場合。次のいずれかの方法を使用して「設定」パネルにナビゲートします。
 - WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。

1. 集約インターフェースを除去するには、「除去」アイコン (✖) をクリックします。

重要: 集約インターフェースを削除する場合、アプライアンスを再始動するまで変更は有効になりません。集約インターフェースを削除した後、コマンド行インターフェースで **clear-all** コマンドと **device-restart** コマンドを実行する必要があります。これらのコマンドを実行しないと、WebSphere DataPower XC10 アプライアンス は、このインターフェースの使用を続行します。

親トピック: [IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

構成のインポートとエクスポート

新規アプライアンスを構成する際、そのアプライアンスの構成設定をエクスポートして保管することができます。後に、アプライアンスの除去や再インストールを必要とする変更が発生した場合に、構成設定を損失することなく、保管しておいた構成データをインポートすることができます。

始める前に

構成をインポートあるいはエクスポートするには、まず、データ・グリッド、ユーザー/グループ、Lightweight Directory Access Protocol (LDAP) 情報を含めたアプライアンス環境を構成する必要があります。詳しくは、[アプライアンスの構成](#)を参照してください。

このタスクについて

管理者としてアプライアンスをセットアップするとき、データ・グリッドのユーザーおよびグループと、LDAP 構成設定の作成が要求されます。このセットアップを完了したら、これらの構成をファイルにエクスポートすることで、この情報を保存することができます。そうすると、そのエクスポート・ファイルを後で使用して、エクスポート済み構成情報をインポートしてアプライアンスに戻すことができます。以下の構成情報をインポートおよびエクスポートすることができます。

- データ・グリッド構成
- ユーザー構成
- グループ構成
- LDAP 構成

集合からのアプライアンス削除か、あるいは、スタンドアロン環境でのアプライアンスの再インストールが必要とされるエラーが発生した場合に、構成データのインポートおよびエクスポートが必要になることがあります。例えば、集合に追加するアプライアンスから構成情報をエクスポートすると、インポート機能を使用して、集合を結合する前のそのアプライアンスの構成にリカバリーすることができます。スタンドアロン環境もしくは集合環境のいずれでも、インポート・アクションおよびエクスポート・アクションを使用すると、削除したアプライアンスの構成情報を手動で再作成する時間を省けます。

スタンドアロン環境の場合: スタンドアロン・アプライアンスを使用している場合、ステップ 1 と 4 を実行してください。これには、構成のインポートおよびエクスポート用の次の **config** コマンド構文が含まれます。

```
config <import|export> -file <filename> [-silent]
```

config export コマンドは、値が 0 のパラメーターも受け入れます。このコマンドは、-file フラグを使用して指定できないアーカイブされたエクスポートを実行するからです。

コマンドの使用法について参照するには、**config usage** と入力してください。

手順

1. 構成情報を、後で使用できるようにファイルにエクスポートします。例えば、ファイル `foo.json` にエクスポートします。コマンド行インターフェースを使用し、指定可能なパラメーターのリストを後ろに指定して **config** コマンドを実行します。

```
config <export> -file foo.json -silent
```

前の例での **-silent** パラメーターはオプションです。**-silent** フラグを指定すると構成状況メッセージが画面に表示されなくなります。これらのメッセージは、デフォルトで表示されることになっています。

2. オプション: 新規アプライアンスを集合に追加します。
3. 構成データを破損するようなエラーが発生したら、アプライアンスを集合から削除します。ログ分析ツールを使用して、ランタイム環境のパフォーマンスを分析したり、お使いのアプライアンス環境内で発生した問題を解決したりできます。
4. 最初のステップで作成した `foo.json` ファイルをインポートします。コマンド行インターフェースを使用し、指定可能なパラメーターのリストを後ろに指定して **config** コマンドを実行します。

```
config <import> -file foo.json -silent
```

前の例での **-silent** パラメーターはオプションです。**-silent** フラグを指定すると構成状況メッセージが画面に表示されなくなります。これらのメッセージは、デフォルトで表示されることになっています。

5. 原型としてそのアプライアンスを含めた新規集合を作成します。
6. その新規集合に追加する形で他のアプライアンスを一度に一つずつ、戻します。

2.5+ [インポートされた構成の管理と構成のエクスポート](#)

いつ WebSphere DataPower XC10 アプライアンス から構成をエクスポートするかを指定することができます

す。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

インポートされた構成の管理と構成のエクスポート

いつ WebSphere® DataPower® XC10 アプライアンス から構成をエクスポートするかを指定することができます。

このタスクについて

構成のエクスポートは、月に 1 回、週に 1 回、または日に 1 回の割合で行えます。また、エクスポートを実行する時刻を指定することもできます。例えば、データ・グリッド構成のエクスポートを月の最初の日に行い、エクスポートした構成をヒストリーに 1 年間保存するように指定することができます。

エクスポートされた構成は最大 10 個保管できます。これには、手動で作成したエクスポート、および Web コンソールを使って作成したエクスポートも含まれます。

手順

1. Web コンソールで、「アプライアンス」 > 「構成のインポートおよびエクスポート」をクリックします。
2. 「エクスポート」セクションを展開します。
 - a. 「エクスポートのスケジュール」を選択します。
 - b. エクスポートを開始する日時を指定します。
 - c. このスケジュールをどのくらいの頻度で続けるかを指定します。
3. 「スケジュールリングの更新」をクリックします。
4. オプション: スケジュールに入れられたエクスポートとエクスポートの間で構成をエクスポートしたい場合は、手動で構成をエクスポートします。手動で構成をエクスポートするには、「今すぐエクスポート」をクリックします。

親トピック: [構成のインポートとエクスポート](#)

ドメイン・ネーム・システム (DNS) サーバーの管理

通信にドメイン・ネーム・システム (DNS) ルックアップ・サービスが使用されるため、IBM® WebSphere® DataPower® XC10 アプライアンスには DNS サーバーが必須です。アプライアンスの初期化時に、DNS サーバーを指定する必要があります。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

DNS サーバーは、アプライアンスが正しく機能するために必要です。ご使用の DNS サーバーには、WebSphere DataPower XC10 アプライアンスで管理されている IP アドレス範囲の、順方向および逆方向の DNS 項目がある必要があります。WebSphere DataPower XC10 アプライアンスは、仮想システムのデプロイ中に、リバース・ルックアップによって取り出されたホスト名を使用します。ホスト名が定義されていないためにリバース・ルックアップが失敗した場合、デプロイメントは、IP アドレスだけでなくホスト名も必要とするため、失敗します。

手順

1. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「ドメイン・ネーム・サーバー」を展開します。
3. オプション: 既存の DNS サーバーを変更するには、「<IP アドレス>」をクリックします。
4. 新しい DNS サーバーを追加するには、「追加するにはここをクリック」を選択します。DNS サーバーは、アプライアンスの初期化中に構成されます。
5. オプション: DNS サーバーを除去するには、「除去」アイコン (✖) をクリックします。

タスクの結果

上記の手順を正常に終わると、通信時のルックアップに使用する DNS サーバーの定義は完了します。

次のタスク

コマンド行インターフェースを使用して、DNS サーバーを設定することもできます。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。

親トピック: [アプライアンスの構成](#)

関連概念:

[xcadminパスワード](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

ホスト名に対する IP アドレスのマッピング

アドレス情報を使用して TCP/IP ネットワーク・プロトコルに接続を作成するには、IP アドレスをホスト名に関連付ける必要があります。アプライアンス上で etc/hosts ファイルを編集することで、ホスト名に対して IP アドレスを解決できます。etc/hosts ファイルはユーザー・インターフェースから編集可能です。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェースから etc/hosts ファイルを編集します。

手順

1. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「ホスト名に対する IP アドレス」を展開します。
3. 「マッピングの作成」をクリックして、IP アドレスとホスト名を指定します。
4. 「OK」をクリックして、マッピングを作成します。このマッピングは、アプライアンス上の etc/hosts ファイルを編集します。
5. オプション: マッピングを除去したい場合は、除去アイコン (✕) をクリックします。

タスクの結果

上記手順を正常に終わると、IP アドレスをホスト名と関連付ける etc/hosts ファイルの編集が完了します。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

日時設定の管理

集合内のご使用の appliance 全体で同期化された日時を維持するために、Network Time Protocol (NTP) サーバーを使用します。

始める前に

以下の手順を実行するには、appliance 管理権限が割り当てられている必要があります。

このタスクについて

集合を使用する場合、NTP サーバーを構成することは重要です。NTP サーバーにより、集合内の異なる appliance 間のログ・ファイルが関連付けられ、ログ・エントリーの日時の一貫性が確実に保たれるようになります。

手順

1. 「設定」パネルにナビゲートします。日時設定を管理するには、以下のいずれかの方法で「設定」パネルまでナビゲートしてください。
 - WebSphere® DataPower® XC10 appliance ユーザー・インターフェース上部のメニュー・バーで、「appliance」>「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: appliance のセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「日時」を展開します。
3. 「現在の時間帯」ドロップダウン・メニューから正しい時間帯を選択します。
4. 新しい NTP サーバーを追加するには、「追加するにはここをクリック」をクリックします。デフォルトでは、NTP サーバーは構成されません。
5. サーバー名をクリックおよびドラッグして、各 NTP サーバーを再配列します。リスト内の最初の使用可能 NTP サーバーが、同期を維持するために使用されます。
6. NTP サーバーを削除するには、「除去」アイコン (✕) をクリックします。
7. appliance を再始動して、変更を有効にします。詳しくは、[ユーザー・インターフェースからの appliance のシャットダウンまたは再始動](#)を参照してください。

タスクの結果

ご使用の集合内の appliance 全体でクロック同期を維持するための NTP サーバーの定義は完了します。

親トピック: [appliance の構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 appliance のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[IBM WebSphere DataPower XC10 appliance の E メール送達の管理](#)

[ユーザー・インターフェースからの appliance のシャットダウンまたは再始動](#)

IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理

アプライアンスのメール送達機能は、ユーザー・パスワードのリセットに使用します。ユーザーが新規パスワードを要求すると、アプライアンスは新規パスワードを記述した E メールをユーザーに送信します。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

Simple Mail Transfer Protocol (SMTP) サーバーを、WebSphere® DataPower® XC10 アプライアンス で使用できるように構成する必要があります。メール送達機能は、ユーザーが自分のパスワードを忘れた場合に、新規パスワードを送信するために使用します。ユーザーがパスワードを忘れた場合に、SMTP サーバーが構成されていないと、そのユーザーは 新規パスワードを受け取ることができません。

手順

1. 「設定」パネルにナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部の メニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「メール送達」を展開します。
3. SMTP サーバーを追加します。WebSphere DataPower XC10 アプライアンス で使用する SMTP サーバーの IP アドレスまたはホスト名を指定します。このフィールドでホスト名が使用される場合、ホスト名は、アプライアンスに定義されている ドメイン・ネーム・システム (DNS) サーバーで解決可能でなければなりません。DNS サーバーの追加について詳しくは、『[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)』を参照してください。
4. 返信先アドレスを追加します。このフィールドには、管理者の E メール・アドレスを使用してください。

タスクの結果

パスワード・リセットに使用する SMTP サーバーと返信先アドレスの指定は完了します。「パスワードをお忘れですか?」リンクがアプライアンスのログイン画面に表示されるので、ユーザーはパスワードを再設定することができます。

親トピック: [アプライアンスの構成](#)

関連概念:

[xcadminパスワード](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動

WebSphere® DataPower® XC10 アプライアンスは、ユーザー・インターフェースから再始動またはシャットダウンできます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。アプライアンスのシャットダウンまたはリスタートを行う場合は、稼働しているプロセスがすべて完了していることをまず確認する必要があります。「タスク」をクリックすると、稼働中のプロセスの進行をチェックできます。

このタスクについて

WebSphere DataPower XC10 アプライアンスは、ユーザー・インターフェースからシャットダウンまたは再始動できます。

手順

1. 「設定」パネルにナビゲートします。アプライアンスへの電源を管理するには、以下のいずれかの方法で「設定」パネルまでナビゲートしてください。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「設定」にナビゲートします。
 - WebSphere DataPower XC10 アプライアンス ユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」 > 「トラブルシューティング」にナビゲートします。
 - 「ホーム」パネルで、「ステップ 1: アプライアンスのセットアップ」セクションにある「設定のカスタマイズ」リンクをクリックします。
2. 「電源」を展開します。
3. 「アプライアンスの再始動」をクリックして、アプライアンスをリサイクルします。再始動中に、アプライアンス上のすべてのソフトウェアが停止され、その後アプライアンスは再始動されます。アプライアンスを即座に再始動するか、またはすべてのアクティブ・タスクが完了するまで待ってから再始動するかを選択できます。
4. 「アプライアンスのシャットダウン」をクリックして、アプライアンスの電源をオフにします。シャットダウン中に、アプライアンス上のすべてのソフトウェアが停止され、アプライアンスは停止されます。アプライアンスを即座にシャットダウンするか、またはすべてのアクティブ・タスクが完了するまで待ってからシャットダウンするかを選択できます。アプライアンスへの電源を停止するには、アプライアンスの背面にある物理的な電源スイッチを使用してアプライアンスの電源を切る必要があります。

タスクの結果

2.5+ アプライアンスの始動状況をユーザー・インターフェースで追跡することができます。詳しくは、[アプライアンスの始動状況のモニター](#)を参照してください。

上記の手順が正常に完了すると、選択内容に基づいてアプライアンスがシャットダウンまたは再始動されます。

親トピック: [アプライアンスの構成](#)

関連タスク:

[セキュリティ](#)

[ユーザーおよびグループの管理](#)

[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)

[構成のインポートとエクスポート](#)

[ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)

[ホスト名に対する IP アドレスのマッピング](#)

[日時設定の管理](#)

[IBM WebSphere DataPower XC10 アプライアンスの E メール送達の管理](#)

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

[タスクによるアクティビティのモニター](#)

集合およびゾーンの構成

集合およびゾーンは、WebSphere® DataPower® XC10 アプライアンス の物理的なトポロジーまたはロケーションの定義をサポートします。

1. [集合の作成](#)
複数の WebSphere DataPower XC10 アプライアンスを可用性、スケーラビリティ、および管理の目的でグループ化するには、集合を作成する必要があります。
2. [ゾーンの作成および変更](#)
ゾーンを使用して、アプライアンスの物理的な場所を指定できます。この物理的な場所は同じデータ・センター内の電源グリッドの 2 つの異なる エリアの場合や、おそらくは国内の別々の場所である、まったく異なるデータ・センターにあるアプライアンスの場合が考えられます。
3. [集合間のマルチマスター複製の構成](#)
マルチマスター複製は、複数のデプロイメント環境全体で連続可用性を確保するための技法です。集合内のカタログ・サーバー間に集合リンクを確立することで、2 つの異なる集合内のデータ・グリッド間で非同期的にデータを複製することができます。
4. **2.5+** [IBM eXtremeIO \(XIO\) の構成](#)
IBM® eXtremeIO (XIO) は、オブジェクト・リクエスト・ブローカー (ORB) を置き換えるトランスポート・メカニズムです。

集合の作成

複数の WebSphere® DataPower® XC10 アプライアンスを可用性、スケーラビリティ、および管理の目的でグループ化するには、集合を作成する必要があります。

始める前に

集合をセットアップする前に、集合に追加するアプライアンスのコンソールに対するアプライアンス管理権限を備える必要があります。

このタスクについて

集合を作成するには、集合にグループ化するすべてのアプライアンスのすべてのホスト名と共通鍵を、アプライアンスの 1 つの構成に追加しなければなりません。アプライアンスは、すべての構成情報と対話できます。この情報にはデータ・グリッド、モニター情報、ユーザーおよびグループ、および集合のメンバーシップが含まれます。

アプライアンスはすべて、同じファームウェア・レベルにしてからでないと集合に追加することはできません。ファームウェア・アップグレードの実行について詳しくは、[ファームウェアの更新](#)を参照してください。

手順






1. 集合の他のアプライアンスのホスト名および共通鍵を追加するアプライアンスを 1 つ選択します。他のアプライアンスのそれぞれにおいて、新しいブラウザー・ウィンドウを開いて以下のステップを完了します。
 - a. ユーザー・インターフェースで、「集合」 > 「メンバー」をクリックします。
 - b. 集合内のアプライアンスのリストが表示されます。ログインしているアプライアンスには、コンソール・アイコン () のマークが付いています。このアプライアンスの「ホスト名」と「共通鍵」をメモします。

表 1. アプライアンスの状態

アイコン	定義
	ログインしているアプライアンスを示し、そのアプライアンスが稼働中であることを示します。
	アプライアンスが開始済みで、構成および使用が可能であることを示します。
	アプライアンスが停止済みで、使用不可であることを示します。

2. 各アプライアンスのホスト名と共通鍵を、グループ内のアプライアンスの 1 つに追加します。
 - a. ユーザー・インターフェースで、「集合」 > 「メンバー」をクリックします。
 - b. 追加アイコンをクリックします ()。
 - c. 集合に追加したいアプライアンスの「ホスト名」および「共通鍵」を入力します。
3. タスク・ビューで、アプライアンスの集合への追加の進行をモニターできます。ユーザー・インターフェースで、「タスク」をクリックします。

タスクの結果

アプライアンスは、スケーラビリティおよび管理の目的で集合にグループ化されます。

親トピック: [集合およびゾーンの構成](#)

次のトピック: [ゾーンの作成および変更](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

ゾーンの作成および変更

ゾーンを使用して、アプライアンスの物理的な場所を指定できます。この物理的な場所は同じデータ・センター内の電源グリッドの2つの異なるエリアの場合や、おそらくは国内の別々の場所である、まったく異なるデータ・センターにあるアプライアンスの場合が考えられます。

始める前に

ゾーンを作成および構成する前に、ご使用のアプライアンスを集合にインストールし、構成、およびアセンブルする必要があります。その後、集合内のアプライアンスを特定のゾーンに存在するように選択します。ゾーンの作成後に追加のアプライアンスを集合に追加する場合は、アプライアンスの追加の際にゾーンを指定できます。

このタスクについて

アプライアンスのロケーションの物理的な境界を定めるためにゾーンを指定することにより、データ・グリッド内に保管されているデータを適切なロケーションに配置できます。DataPower® XC10 アプライアンスは、ゾーンを使用してキャッシュのために基本およびレプリカの情報どこに配置するかを決定します。例えば、基本データ・グリッド (data grid) がゾーン 1 にある場合、そのデータ・グリッド (data grid) のレプリカはゾーン 2 に配置されます。1つのゾーンしかない場合には、レプリカは作成されません。

初めてアプライアンスが初期化されるときに、**DefaultZone** と呼ばれるデフォルト・ゾーンに追加されます。作成したゾーンにアプライアンスを追加するときに、アプライアンスは **DefaultZone** ゾーンから除去されます。アプライアンスが **DefaultZone** ゾーンから除去された後は、そのアプライアンスを再び **DefaultZone** ゾーンに戻すことはできません。

手順

1. ユーザー・インターフェースで、「**集合**」 > 「**ゾーン**」をクリックします。
2. ゾーンを追加します。「**ゾーンの追加**」をクリックします。追加するゾーンの名前を指定します。ゾーン名がコンソール・ページに表示されます。複数のゾーンを作成する場合は、これらの手順を繰り返します。
3. ゾーンにアプライアンスを追加します。ゾーン名を展開して、ゾーンに追加するアプライアンスを選択します。集合内のすべてのアプライアンスがゾーンに追加されるまで、これらの手順を繰り返します。すべてのアプライアンスをゾーンに配置する必要があります。単一のゾーンには、異なる集合のアプライアンスを含めることはできません。
4. ゾーンの変更を適用します。ゾーンの変更内容は、「**ゾーンの変更をアクティブにする**」をクリックするまで有効になりません。ゾーンの情報を変更する場合は、削除アイコン (✖) をクリックしてゾーンを除去するか、「**ゾーンの変更をリセットする**」をクリックしてゾーンをアクティブにした最後の構成にリセットします。

タスクの結果

DataPower XC10 アプライアンスは、定義した物理ロケーション、即ちゾーンに基づいて、基本およびレプリカのデータ・キャッシュを作成できます。ユーザー・インターフェースで、「**集合**」 > 「**ゾーン**」をクリックすれば、いつでもゾーンのトポロジーの表示か変更を行うことができます。

次のタスク

ご使用のアプリケーションおよびデータ・グリッドを作成し、構成します。

親トピック: [集合およびゾーンの構成](#)

前のトピック: [集合の作成](#)

次のトピック: [集合間のマルチマスター複製の構成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

集合間のマルチマスター複製の構成

マルチマスター複製は、複数のデプロイメント環境全体で連続可用性を確保するための技法です。集合内のカタログ・サーバー間に集合リンクを確立することで、2つの異なる集合内のデータ・グリッド間で非同期的にデータを複製することができます。

始める前に

- 少なくとも2つの集合を構成する必要があります。集合間でデータを複製するには、各集合内に同一のデータ・グリッド構成を作成する必要があります。
- 定義する複数の集合トポロジーのタイプを決定します。マルチマスター・トポロジーについて詳しくは、[マルチマスター複製を実装する集合のリンクのトポロジー](#)を参照してください。
- 集合間のリンクは、ユーザー・インターフェースまたは **xscmd** ユーティリティで構成できます。

このタスクについて

障害の誤検出が起こる可能性があるため、単一の集合の範囲は信頼できないネットワーク上には及びません。しかし、信頼できないネットワークへの接続を持つアプライアンス間で、データ・グリッドのデータを複製する必要が生じる場合もあります。このタイプのトポロジーを使用することになる一般的なシナリオには以下のようなものがあります。

- 1つの集合がアクティブでもう1つの集合がバックアップとして使用されているデータ・センター間での災害復旧。
- 地理的に近いクライアントのために、すべての集合がアクティブになっている地理的に分散されたデータ・センター。

2つの集合を接続すると、同じ名前を持つすべてのデータ・グリッドが集合間で非同期的に複製されます。これらのデータ・グリッドは、各集合内に同じ数のレプリカを持ち、同じ動的マップ構成を持っていない限りなりません。

手順

1. 集合間にリンクを確立します。
 - **ユーザー・インターフェースで集合リンクを確立する:**
 - a. ユーザー・インターフェースで、「集合」 > 「集合リンク」をクリックします。
 - b. 追加アイコン (+) をクリックし、リモート・アプライアンスのホスト名または IP アドレス、ユーザー名、およびパスワードを入力します。
 - **xscmd ユーティリティで集合リンクを確立する:**
 - a. 各集合内のカタログ・サーバーのカタログ・サーバー IP アドレスおよびポートを取得します。ユーザー・インターフェースで、「集合」 > 「メンバー」 > 「**collective_member_name**」をクリックします。カタログ・サーバーおよびポート番号のリストが表示されます。接続する各集合についてこのステップを繰り返します。
 - b. **xscmd** ユーティリティを集合の1つに接続します。**xscmd** ユーティリティの開始について詳しくは、[xscmd ユーティリティによる管理](#)を参照してください。

```
xscmd.sh -ts xsatruststore.jks -tst jks -ssl -tsp xc10pass
-user xcadm -tsp xcadm -cep myxc10.mycompany.com:2809
[additional_xscmd_parameters]
```

- c. 現在接続している集合から、次のコマンドを実行します。

```
xscmd -c establishLink -cep myxc10.mycompany.com:2809 -fd dname
-fe myxc102.mycompany.com:2809,myxc103.mycompany.com:2809
```

-fd dname は、リンクを確立しようとしているリモート集合の名前を指定します。この例では、**dname** がリモート集合の名前です。メンバー詳細パネルでカタログ・サーバー・リストを調べれば、集合の名前がわかります。集合名はリスト内の最初の IP アドレスです。

リンクが確立されると、集合内のカタログ・サーバーが相互に複製を開始します。双方向にリンクを確立する必要はありません。

2. 集合間のリンクを除去することも可能です。
 - **ユーザー・インターフェースで集合リンクを除去する:** ユーザー・インターフェースで、除去アイコン (X) をクリックし、選択したリンクを除去します。
 - **xscmd ユーティリティで集合リンクを除去する:** **xscmd** ユーティリティで、次のコマンドを実行します。

```
xscmd -c dismissLink -cep myxc10.mycompany.com:2809 -fd dname
```

親トピック: [集合およびゾーンの構成](#)

前のトピック: [ゾーンの作成および変更](#)

次のトピック: **2.5+** [IBM eXtremeIO \(XIO\) の構成](#)

関連概念:

[マルチマスター複製を実装する集合のリンクのトポロジー](#)

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

カタログ・サービス・ドメインのトランスポート・タイプの表示

カタログ・サービス・ドメインで現在使用されているトランスポート・タイプを表示できます。

始める前に

スタンドアロン・カタログ・サービス・ドメイン、または WebSphere® Application Server で実行されているカタログ・サービス・ドメインで使用されているトランスポート・タイプを表示できます。

- スタンドアロン・カタログ・サービス・ドメインを使用している場合は、**xscmd** ユーティリティーを使用して、カタログ・サービス・ドメインに関するトランスポート情報を表示します。**xscmd** ユーティリティーのセットアップについて詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。
- WebSphere Application Server で実行されているカタログ・サービス・ドメインを使用している場合は、**wsadmin** ユーティリティーを使用して、トランスポート・タイプを表示できます。**wsadmin** ユーティリティーについて詳しくは、[『wsadmin スクリプトによる wsadmin スクリプト・クライアントの開始』](#)を参照してください。

手順

- スタンドアロン・カタログ・サービス・ドメインのトランスポート・タイプを表示します。**xscmd** ユーティリティーで、以下のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showTransport`
 - **Windows** `xscmd.bat -c showTransport`このコマンドにより、トランスポート・タイプが表示されます。表示される値は、「eXtremeIO」または「Object Request Broker」です。
- WebSphere Application Server で実行されているカタログ・サービス・ドメインのトランスポート・タイプを表示します。
 - 管理コンソールで、「システム管理」 > 「WebSphere eXtreme Scale」 > 「カタログ・サービス・ドメイン」 > 「**catalog_service_domain_name**」をクリックします。「**IBM eXtremeIO (XIO) 通信を有効にする**」設定が選択されているかどうかを確認します。
 - **wsadmin** ユーティリティーで、以下のコマンドを実行します。
 - Jacl を使用:

```
$AdminTask getTransport {-domainName TestDomain }
```

- Jython スtring を使用:

```
AdminTask.getTransport ('[-domainName testDomain]')
```

このコマンドにより、トランスポート・タイプが表示されます。表示される値は、「eXtremeIO」または「Object Request Broker」です。リモート・サーバーを含むカタログ・サービス・ドメインでこのコマンドを実行した場合、または `catalogServerName` がリモート・サーバーである場合は、エラーとなります。リモート・サーバーの場合は、**xscmd -c showTransport** コマンドを使用する必要があります。**wsadmin** ユーティリティーの **getTransport** コマンドについて詳しくは、[カタログ・サービス・ドメイン管理用タスク](#)を参照してください。

親トピック: [2.5+ IBM eXtremeIO \(XIO\) の構成](#)

データ・グリッド の構成

3つのタイプのデータ・グリッドが作成できます。単純データ・グリッド、セッションデータ・グリッド、およびダイナミック・キャッシュデータ・グリッドです。

[単純データ・グリッドの作成](#)

単純データ・グリッド (data grid)を使用して、オペレーションの作成、取得、更新、および削除を行うことができます。特に、単純データ・グリッド (data grid)を使用すると、データベース内に保管されたデータにより速くアクセスできるようになります。

[セッション・データ・グリッドの作成](#)

Java または .NET アプリケーションからの HTTP セッションを保管するデータ・グリッドを作成することができます。

Java [動的キャッシュの作成データ・グリッド](#)

IBM® WebSphere® DataPower® XC10 アプライアンス を使用して、WebSphere Application Server 動的キャッシュ・インスタンスのデータを保管することができます。この機能をセットアップすることにより、WebSphere Application Server 動的キャッシュ・インスタンスを使用するよう作成されたアプリケーションでアプライアンスのフィーチャーおよびパフォーマンス機能を活用できるようにします。

[データ・グリッドの最大容量の構成](#)

集合内の各データ・グリッドの最大容量を定義できます。容量限度を構成すると、特定のデータ・グリッドで使用可能なデータ・ストレージの容量が制限されます。この容量限度により、確実に、集合の使用可能なストレージ容量が予測可能な形で使用されるようになります。

[データ・グリッドのセキュリティーを使用可能に設定する](#)

データ・グリッドの作成後、データ・グリッド (data grid)のセキュリティーはデフォルトで使用不可になります。データ・グリッド (data grid)のセキュリティー設定を変更して、特定のユーザーまたはユーザー・グループにアクセスを制限することができます。

[データ・グリッドのクリア](#)

データ・グリッド内のすべての項目を恒久的に削除することができます。失効した情報またはテスト項目を除去するために、データ・グリッドをクリアできます。

[データ・グリッドの除去](#)

データ・グリッド (data grid)のデータを初期化する場合は、データ・グリッド (data grid)を除去してからデータ・グリッド (data grid)を再作成します。

[Spring キャッシュ・プロバイダーの構成](#)

Spring Framework バージョン 3.1 では、新しいキャッシュの抽象化が導入されています。この新しい抽象化を使用すると、既存の Spring アプリケーションに対してキャッシュを容易に追加できます。WebSphere DataPower XC10 アプライアンス をキャッシュ抽象化のキャッシュ・プロバイダーとして使用できます。

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

単純データ・グリッドの作成

単純データ・グリッド (data grid)を使用して、オペレーションの作成、取得、更新、および削除を行うことができます。特に、単純データ・グリッド (data grid)を使用すると、データベース内に保管されたデータにより速くアクセスできるようになります。

始める前に

単純データ・グリッドは、WebSphere® Application Server またはスタンドアロンの Java™ アプリケーションで使用できます。どちらかのシナリオに WebSphere eXtreme Scale クライアント がインストールされている必要があります。

このタスクについて

単純データ・グリッドを使用すると、データベースの負荷を軽減することで、動的 Web アプリケーションを高速化できます。メモリー内の任意のデータのキー値のペアを保管して、高価な データベース照会を削減できます。鍵は、java.lang.String または Integer といった 既存の Java 型にできます。値は、任意の直列化可能オブジェクト・タイプにできます。データが必要なたびに、アプライアンス上の単純 データ・グリッド (data grid)が最初にチェックされます。アプライアンスにそのデータがない場合、データベースからデータが取り出され、単純データ・グリッド (data grid)に挿入されます。

手順

1. 単純データ・グリッド (data grid)を作成します。ユーザー・インターフェースで、「データ・グリッド」 > 「単純データ・グリッド」をクリックします。追加アイコン (+) をクリックし、作成する単純データ・グリッド (data grid)の名前を指定します。データ・グリッドの名前では、文字 ^ . ¥¥ / , # \$ @ : ; ¥ * ? < > | = + & % [] " " は使用できません。
2. 単純データ・グリッド (data grid)用の objectgrid.xml ファイルをダウンロードします。作成した単純データ・グリッド (data grid)の構成で、ダウンロード・アイコン (📄) をクリックして、ファイルをご使用のローカル・ファイル・システムに保存します。
3. データ・グリッド (data grid)にアクセスするアプリケーションを作成します。詳しくは、[単純データ・グリッドにアクセスするアプリケーションの開発](#)を参照してください。

次のタスク

- データ・グリッドへのデータの送信を開始する前に、セキュリティーを構成します。詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。
- レプリカを構成します。レプリカにより、基本コピーで障害が発生した場合にもデータ・グリッドのデータが使用可能になります。レプリカを構成するには、「データ・グリッド」 > 「単純データ・グリッド」 > 「拡張属性の表示」をクリックします。アプライアンスが集合内にある場合にのみ、レプリカが作成されます。集合内のアプライアンスの数が n の場合、レプリカの最大数は $n-1$ になります。したがって、3つのレプリカを構成するが、集合内にあるアプライアンスが2つのみである場合、作成されるレプリカは1つのみです。集合にアプライアンスを追加すると、追加のレプリカが作成されます。アプライアンスが集合に追加されたときに新規レプリカを作成できるようにするため、レプリカの数適切な数に設定してください。レプリカ数を編集すると、データ・グリッドの内容が消去されます。
- データ・グリッドの容量限度を構成します。データ・グリッドの容量限度の構成により、集合のストレージ容量が予測可能な形で使用されることを確実にできます。詳しくは、[データ・グリッドの最大容量の構成](#)を参照してください。
- 単純グリッドの存続時間 Evictor を構成します。詳しくは、[存続時間 \(TTL\) Evictor の構成](#)を参照してください。
- DataPower® XC10 アプライアンス ユーザー・インターフェース でデータ・グリッド (data grid)をモニターできます。詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

[動的マップの構成](#)

マップ・テンプレートの事前定義セットに基づいて、動的にマップを作成することができます。

[ロック・ストラテジーの構成](#)

WebSphere eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。WebSphere eXtreme Scale Client for .NET からアクセスするバックキング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。

[存続時間 \(TTL\) Evictor の構成](#)

単純グリッドを作成する際、1つのデフォルト (静的) マップと一連の動的マップが作成されます。デフォルトでは、デフォルト・マップ用に構成された存続時間 Evictor はありません。動的マップがある場合は、作成時間 (*CT)、最終更新時間 (*LUT)、または最終アクセス時間 (*LAT) の TTL 値を設定できます。TTL Evictor がデフォルト・マップでも使用可能になるようにこのデフォルトの動作を変更することも可能です。

Java

[ニア・キャッシュの構成](#)

ローカルのインライン・キャッシュを持つようにクライアントを構成することができます。オプションのこのキ

キャッシュはニア・キャッシュと呼ばれます。これは、各クライアントにある独立したデータ・グリッドであり、リモート用のキャッシュ(サーバー・サイド・キャッシュ)として機能します。ニア・キャッシュは、ロックが使用不可になっているかオプティミスティックとして構成されている場合、デフォルトで使用可能であり、ロックがペシミスティックに構成されている場合は使用することができません。

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

接続済みのクライアントとサーバーに対してネットワーク・トランスポートが公開するホスト名は、**publishHost** プロパティを使用して構成します。

親トピック: [データ・グリッドの構成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

動的マップの構成

マップ・テンプレートの事前定義セットに基づいて、動的にマップを作成することができます。

始める前に

- 単純データ・グリッドを作成します。詳しくは、[単純データ・グリッドの作成](#)を参照してください。
- 動的マップで、どの構成オプションを使用する必要があるかを決定します。詳しくは、[動的マップの構成オプション](#)を参照してください。

次のタスク

定義済みテンプレートを使用して、以下のように動的マップを作成します。

- **Java** **Java API** を使用: 動的マップを定義するために `Session.getMap(String)` メソッドを呼び出す場合の例は、[Java API を使用した動的マップの作成](#)を参照してください。
- **.NET** **2.5+** **.NET API** を使用: 詳しくは、[.NET API を使用した動的マップの作成](#)を参照してください。
- **REST** ゲートウェイを使用: 詳しくは、[REST ゲートウェイの例: 動的マップの作成](#)を参照してください。

親トピック: [単純データ・グリッドの作成](#)

関連タスク:

[ロック・ストラテジーの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

Java [ニア・キャッシュの構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

Java **2.5+** [ニア・キャッシュ無効化の構成](#)

関連資料:

[動的マップの構成オプション](#)

ロック・ストラテジーの構成

WebSphere® eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。 WebSphere eXtreme Scale Client for .NET からアクセスするバックイング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。

始める前に

使用したいロック・ストラテジーを決めます。詳しくは、[ロック・ストラテジー](#)を参照してください。

このタスクについて

次のいずれかのロック・ストラテジーを使用するように各 BackingMap インスタンスを構成することができます。

- オプティミスティック・ロック・モード (デフォルト)
- ペシミスティック・ロック・モード (.NET アプリケーションの場合は必要な)
- なし

手順

オプティミスティック・ロック・モード、ペシミスティック・ロック・モード、またはロックなしモードについては、[動的マップの構成オプション](#)を参照してください。

親トピック: [単純データ・グリッドの作成](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[動的マップの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

 [ニア・キャッシュの構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

 [Java アプリケーションにおけるロックの構成と実装](#)

[ロック・ストラテジーの構成](#)

 [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

 [.NET アプリケーションにおけるロックの構成と実装](#)

関連資料:

例: [flush メソッドのロック順序](#)

存続時間 (TTL) Evictor の構成

単純グリッドを作成する際、1つのデフォルト (静的) マップと一連の動的マップが作成されます。デフォルトでは、デフォルト・マップ用に構成された存続時間 Evictor はありません。動的マップがある場合は、作成時間 (*CT)、最終更新時間 (*LUT)、または最終アクセス時間 (*LAT) の TTL 値を設定できます。TTL Evictor がデフォルト・マップでも使用可能になるようにこのデフォルトの動作を変更することも可能です。

始める前に

- ご使用の構成用の単純データ・グリッドを作成します。
- どの時間ベースの Evictor を使用するか決定します。特定の Evictor タイプの使用については、[Evictor](#) を参照してください。

手順

1. ユーザー・インターフェースで、「データ・グリッド」 > 「単純データ・グリッド」 > 「*data_grid_name*」 > 「拡張属性の表示」をクリックします。
2. 「次の名前前のマップ用 **Evictor**」リスト・ボックスで、Evictor タイプを選択します。このオプションを「なし」のままにする場合は、存続時間 Evictor はデフォルト・マップに対してのみ使用不可になります。または、Evictor タイプに「なし」以外を選択すると、TTL がデフォルト・マップに対して使用可能になります。
3. 「次の時間が経過したら存続時間マップからデータを除去」ボックスに値を指定します。デフォルトの TTL 値は 3600 秒です。Evictor タイプが「なし」以外に設定されている場合、ここで指定する値はデフォルト・マップにも適用されます。値が「0」の場合は、グリッド全体に対して TTL 除去がオフになります。
4. 「変更の適用」をクリックして保存します。グリッドの再始動後のデータ損失について警告が表示されます。

例

デフォルトのシナリオ:

このシナリオでは、「次の名前前のマップ用 **Evictor**」リスト・ボックスで選択された Evictor タイプは「なし」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「3600」です。この結果、*CT、*LUT、および *LAT で終わる動的マップについてのみ、1 時間後にエントリーが除去されます。

デフォルトの **Evictor** タイプを「なし」以外にカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 **Evictor**」リスト・ボックスで選択された Evictor タイプは「作成時刻」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「3600」です。この結果、*CT、*LUT、および *LAT で終わる動的マップについて、1 時間後にエントリーが除去されます。デフォルト・マップは、作成時間から 1 時間後にエントリーを除去します。

デフォルトの **Evictor** タイプと TTL 値をカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 **Evictor**」リスト・ボックスで選択された Evictor タイプは「最終更新時刻 (**Last updated time**)」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「4800」です。この結果、*CT、*LUT、および *LAT で終わる動的マップについて、80 分後にエントリーが除去されます。デフォルト・マップは、エントリーが最後に更新された時間から 80 分後にエントリーを除去します。

TTL 値のみをカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 **Evictor**」リスト・ボックスで選択された Evictor タイプは「なし」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「0」です。この結果、このグリッドのすべての (デフォルトおよび動的) マップについて、TTL の除去はオフになります。

デフォルトの **Evictor** タイプと TTL 値をカスタマイズするシナリオ:

このシナリオでは、「次の名前前のマップ用 **Evictor**」リスト・ボックスで選択された Evictor タイプは「最終アクセス時間」で、「次の時間が経過したら存続時間マップからデータを除去」フィールドに指定された TTL 値は「0」です。この結果、このグリッドのすべての (デフォルトおよび動的) マップについて、TTL の除去はオフになります。

Evictor

Evictor は、データ・グリッドからデータを削除します。単純グリッド上の動的マップおよびデフォルト・マップ用の Evictor を構成できます。

親トピック: [単純データ・グリッドの作成](#)

関連概念:

[Evictor](#)

関連タスク:

[動的マップの構成](#)

[ロック・ストラテジーの構成](#)

[Java](#) [ニア・キャッシュの構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

[Java](#) **2.5+** [ニア・キャッシュ無効化の構成](#)

関連資料:

[動的マップの構成オプション](#)

Evictor

Evictor は、データ・グリッドからデータを削除します。単純グリッド上の動的マップおよびデフォルト・マップ用の Evictor を構成できます。

Evictor タイプ

Evictor は、存続時間の概念に基づいてエントリーを除去します。作成時間、最終アクセス時間、または最終更新時間に基づく Evictor を選択できます。デフォルトでは、Evictor は動的マップで作成されます。単純グリッドのデフォルト・マップで Evictor を使用可能にするには、[存続時間 \(TTL\) Evictor の構成](#) を参照してください。

なし

エントリーの期限切れがないように、それによってマップからエントリーが除去されることがないように指定します。

作成時間

作成された時に応じてエントリーが除去されるように指定します。

作成時間 Evictor を使用している場合、Evictor は、作成からの時間がその TTL 値に達するとエントリーを除去します。TTL 値は、アプリケーション構成でミリ秒単位で設定されます。TTL 値を 10 秒に設定すると、エントリーは挿入の 10 秒後に自動的に除去されます。

この値を作成時間 Evictor タイプに設定する場合は注意が必要です。この Evictor は、一定時間にのみ使用される、キャッシュへの妥当な追加量がある場合に、最も有効に使用されます。このストラテジーによって、作成されたものはすべて、一定時間後に除去されます。

作成時間 Evictor タイプは、20 分以下の間隔で株価情報を最新表示するようなシナリオで役立ちます。例えば、ある Web アプリケーションは株価情報の取得をしますが、最新情報を取得することは重要でないとします。この場合、株価情報は 20 分間、データ・グリッドにキャッシュされます。20 分後、マップの有効期限が切れ、除去されます。ほぼ 20 分ごとに、グリッドはデータをデータベースのデータでリフレッシュします。データベースは 20 分ごとに最新の株価情報によって更新されます。

最終アクセス時間

(読み取りか更新かに関わらず) 最後にアクセスされた時に応じてエントリーが除去されるように指定します。

最終更新時間

最後に更新された時に応じてエントリーが除去されるように指定します。

最終アクセス時間 または最終更新時間の Evictor タイプを使用している場合は、作成時間 Evictor を使用している場合よりも TTL 値を低い数値に設定します。エントリーは、アクセスされるたびにリセットされるからです。言い換えれば、値が 15 で、エントリーが 14 秒間存在し、それからアクセスされた場合、このエントリーはあと 15 秒間有効期限が切れることはありません。TTL 値を比較的高い数値に設定した場合は、多くのエントリーがまったく除去されなくなる可能性があります。ただし、この値を 15 秒程度に設定すると、エントリーは頻繁にアクセスされない場合に除去されることとなります。

最終アクセス時間 または最終更新時間 Evictor タイプは、データ・グリッド・マップを使用してクライアントからのセッション・データを保持するシナリオなどで役立ちます。セッション・データは、クライアントがそのセッション・データを一定時間使用しない場合は破棄する必要があります。例えば、セッション・データは、クライアントによるアクティビティが 30 分間なかった後にタイムアウトになるとします。この場合、最終アクセス時間 または最終更新時間の Evictor タイプを使用し、TTL 値を 30 分に設定するのが、このアプリケーションにおいて適切です。

親トピック: [存続時間 \(TTL\) Evictor の構成](#)

関連タスク:

[存続時間 \(TTL\) Evictor の構成](#)

関連資料:

[動的マップの構成オプション](#)

ニア・キャッシュの構成

ローカルのインライン・キャッシュを持つようにクライアントを構成することができます。オプションのこのキャッシュはニア・キャッシュと呼ばれます。これは、各クライアントにある独立したデータ・グリッドであり、リモート用のキャッシュ(サーバー・サイド・キャッシュ)として機能します。ニア・キャッシュは、ロックが使用不可になっているかオプティミスティックとして構成されている場合、デフォルトで使用可能であり、ロックがペシミスティックに構成されている場合は使用することができません。

始める前に

ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。

このタスクについて

ニア・キャッシュは、リモート側で保管されているキャッシュ・データ・セット全体のサブセットへのローカル・メモリー内アクセスを可能にするため、高速です。デフォルトで、クライアント・サイドのニア・キャッシュには最大サイズがないため、クライアントのメモリー不足エラーが発生する可能性があります。ニア・キャッシュのサイズを制御するには、クライアント上で存続時間(TTL)または最長未使用時間(LRU) Evictor を使用可能に設定するクライアント・サイドのオーバーライドを構成してください。クライアント設定のオーバーライドについては、[XML 構成を使用した Java クライアントの構成](#)および[Java クライアントのプログラマチック構成](#)を参照してください。

手順

動的マップ名を設定することで、クライアント上にニア・キャッシュが作成されるように設定できます。動的マップは、オプティミスティック・ロックまたは非ロック式になるように設定する必要があります。ニア・キャッシュを使用可能に設定しているマップ名の例を以下に示します。

```
my_grid.NONE  
my_grid.NONE.0
```

詳しくは、[動的マップの構成](#)を参照してください。動的マップ構成オプションのリストについては、[動的マップの構成オプション](#)を参照してください。

タスクの結果

ニア・キャッシュが使用可能になっているかどうかを確認するには、クライアントで `BackingMap.isNearCacheEnabled()` メソッドを実行します。また、ログ・ファイルで `CWOBJ1128I` メッセージを検索して、ニア・キャッシュが使用可能になっているかどうかを確認することもできます。

Java 2.5+ [ニア・キャッシュ無効化の構成](#)

ニア・キャッシュ無効化を構成して、失効データをニア・キャッシュから可能な限り迅速に除去できます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化操作がトリガーされます。このメカニズムは、もう 1 つのオプション(ニア・キャッシュでの存続時間(TTL)除去の使用)よりも迅速に機能します。

親トピック: [単純データ・グリッドの作成](#)

関連タスク:

[動的マップの構成](#)

[ロック・ストラテジーの構成](#)

[存続時間\(TTL\) Evictor の構成](#)

2.5+ [クライアントとサーバー間の通信用ホスト名の管理](#)

ニア・キャッシュ無効化の構成

2.5+ ニア・キャッシュ無効化を構成して、失効データをニア・キャッシュから可能な限り迅速に除去できます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化操作がトリガーされます。このメカニズムは、もう 1 つのオプション (ニア・キャッシュでの存続時間 (TTL) 除去の使用) よりも迅速に機能します。

始める前に

- IBM eXtremeIO を使用している必要があります。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。
- ニア・キャッシュを使用している必要があります。ニア・キャッシュが使用可能になっているかどうかを確認するには、クライアントで `BackingMap.isNearCacheEnabled()` メソッドを実行します。ニア・キャッシュの構成について詳しくは、[ニア・キャッシュの構成](#)を参照してください。
- ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。

このタスクについて

ニア・キャッシュ無効化を使用可能にすると、リモート・データが変更されるとニア・キャッシュが更新されるため、リモート・データ・グリッドからのデータ・セットがより正確なものになります。

手順

2.5+ ニア・キャッシュ無効化オプションを含む動的マップを作成します。詳しくは、[動的マップの構成](#)を参照してください。ニア・キャッシュ無効化が使用可能になっているマップの以下の例を参照してください。

```
my_grid.NCI
my_grid.CT.NCI
my_grid.LAT.NCI
my_grid.LUT.NCI
my_grid.NONE.NCI
my_grid.CT.0.NCI
my_grid.LAT.0.NCI
my_grid.LUT.0.NCI
my_grid.NONE.0.NCI
```

動的マップ構成オプションのリストについては、[動的マップの構成オプション](#)を参照してください。

親トピック: [Java](#) [ニア・キャッシュの構成](#)

関連タスク:

[動的マップの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

関連資料:

[動的マップの構成オプション](#)

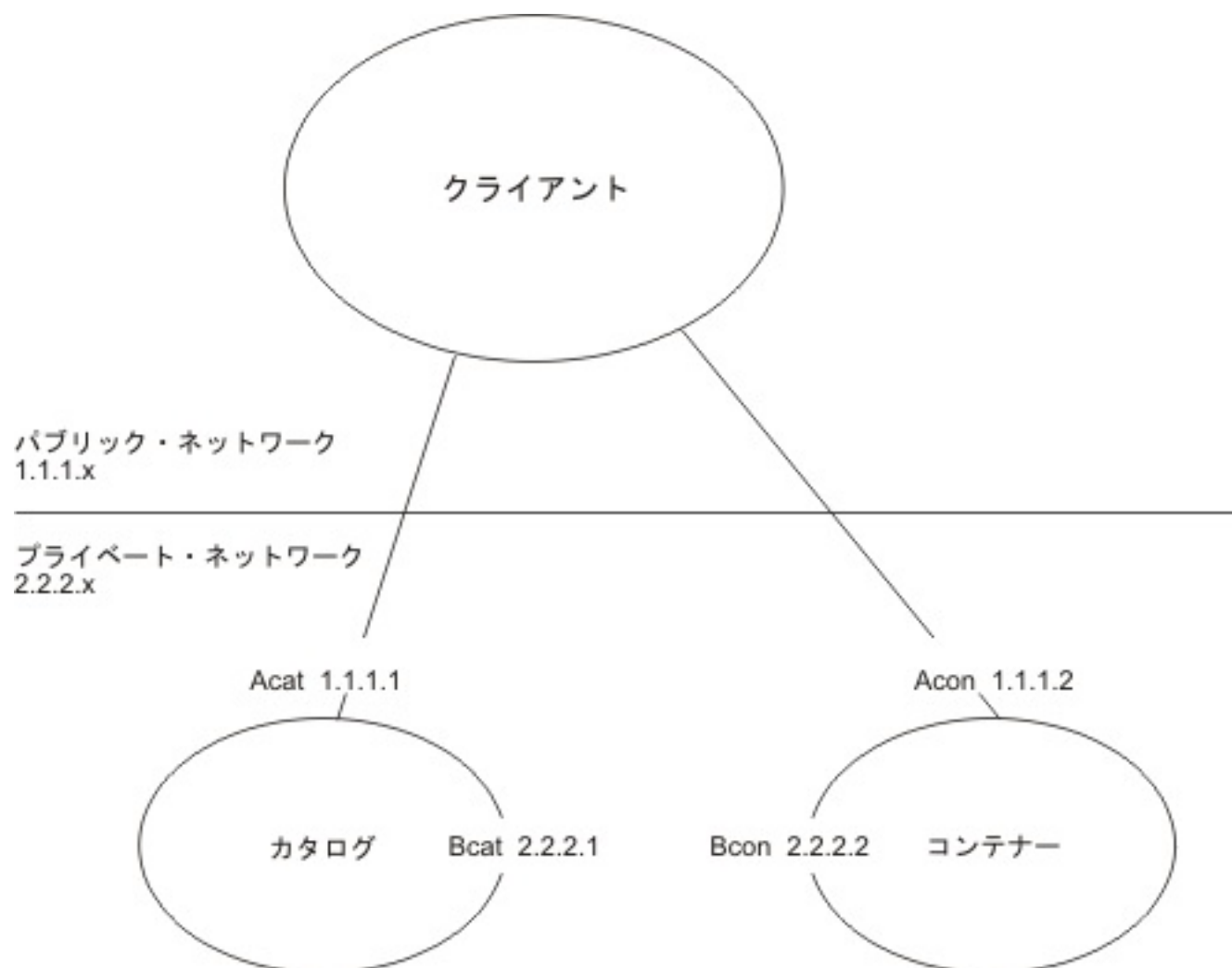
クライアントとサーバー間の通信用ホスト名の管理

接続済みのクライアントとサーバーに対してネットワーク・トランスポートが公開するホスト名は、**publishHost** プロパティを使用して構成します。

このタスクについて

ドメイン・ネーム・システム (DNS) サーバーまたはホスト・ファイルを併用して、**publishHost** プロパティで公開されるホストを構成したり、オーバーライドしたりすることができます。これで、クライアントとサーバーの間に行き来する通信について、一層きめ細かい制御ができるようになり、また、データ・グリッド内のノード間の通信を最適化する機能も提供されます。例えば、次の図は、どのようにクライアントをパブリックのインターネット・プロトコル (IP) (1.1.1.x) に接続し、サーバーをプライベートの IP (2.2.2.x) に接続できるかを示しています。この例では、カタログ・サーバーとコンテナ・サーバーには、それぞれのネットワーク・カードに対する 2 つのホスト名、すなわち、パブリックで使用可能な IP アドレス用の A* とプライベート IP アドレス用の B* があります。

図 1. **publishHost** プロパティのトポロジー



C* ホスト名は、公開する総称ホスト名として使用される、**publishHost** プロパティの新しい値を表します。これは A* または B* の代わりとしても使用されます。A* 名はプライベート・ネットワークで通信したいサーバーには不適切であり、B* 名はパブリック・ネットワークで通信したいクライアントには不適切であるからです。これにより、代替パブリック・アドレスおよび代替プライベート・アドレスの中から選択できるようになります。

手順

1. **publishHost** プロパティをカタログ・サーバーの Ccat およびコンテナ・サーバーの Ccon に設定します。このプロパティは、クライアントとサーバーの通信時に公開され、DNS にも公開されるホスト名を設定します。
2. オプション: **listenerHost** プロパティを設定します。このプロパティは、各サーバーがどの IP アドレスを listen するかを制御します。
3. パブリック・ネットワークを使用するクライアントの DNS (または /etc/hosts ファイル) が Ccat および Ccon ホストを指していることを確認します。これらのホストのパブリック IP アドレスはそれぞれ Ccat 1.1.1.1 および Ccon 1.1.1.2 です。プライベート・ネットワーク上のサーバーの DNS (または /etc/hosts ファイル) が Ccat および Ccon ホストを指していることを確認します。これらのホストのプライベート IP アドレスはそれぞれ Ccat 2.2.2.1 および Ccon 2.2.2.2 です。
4. カatalog・サーバー (Ccat) の **publishHost** 値を、クライアントおよびサーバー上のオブジェクト・グリッド構成で使用されるカタログ・サーバー・エンドポイントのホスト名として使用します。

タスクの結果

これで、サーバー間を流れるトラフィックはパブリック・ネットワーク上にあり、クライアントからコンテナに流れるトラフィックはプライベート・ネットワーク上にあるようになりました。

親トピック: [単純データ・グリッドの作成](#)

関連タスク:

[動的マップの構成](#)

[ロック・ストラテジーの構成](#)

[存続時間 \(TTL\) Evictor の構成](#)

[Java ニア・キャッシュの構成](#)

セッション・データ・グリッドの作成

Java または .NET アプリケーションからの HTTP セッションを保管するデータ・グリッドを作成することができます。

[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パーシスタンスの構成](#)

アプライアンスをセッション管理で使用するよう WebSphere® Application Server アプリケーションを構成します。新規アプリケーションのインストール時にアプライアンスを選択することも、既存のアプリケーションまたはサーバーの設定をアプライアンスを使用するよう更新することもできます。

[WebSphere Application Server での HTTP セッション・マネージャーの構成](#)

WebSphere Application Server はセッション管理機能を備えていますが、要求の数が増えるとパフォーマンスが低下します。WebSphere eXtreme Scale には、セッション・レプリカ生成、高可用性、優れたスケーラビリティ、および堅固な構成オプションを備えたセッション管理実装がバンドルされています。

[WebSphere Portal での HTTP セッション・マネージャーの構成](#)

WebSphere Portal の HTTP セッションをデータ・グリッドに保持できます。

[各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成](#)

WebSphere eXtreme Scale クライアント には、Web コンテナの デフォルト・セッション・マネージャーをオーバーライドするセッション管理実装がバンドルされています。この実装は、セッション・レプリカ生成、高可用性、より優れたスケーラビリティと 構成オプションを提供します。Tomcat などさまざまなアプリケーション・サーバーに対して WebSphere eXtreme Scale クライアント セッション複製マネージャーおよび汎用組み込み ObjectGrid コンテナの開始を有効にします。

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

以前に設定したメモリー間複製セッションまたはデータベース・セッションをマイグレーションして、WebSphere eXtreme Scale セッション管理を使用できます。

.NET [2.5+ ASP.NET セッション状態ストア・プロバイダーの構成](#)

データ・グリッド内の ASP.NET アプリケーションからセッション状態を保管できます。

[2.5+ HTTP セッション・サイズを表示](#)

`xscmd` コーティリティーを使用して、Java アプリケーションでのセッションのサイズを表示することができます。セッション・サイズを表示することで、データ・グリッドの問題のトラブルシューティングが容易になります。

[サーブレット・コンテキスト初期化パラメーター](#)

以下に示すサーブレット・コンテキスト初期化パラメーターのリストは、選択した接続メソッドに必要なスプライサー・プロパティ・ファイルに指定できるものです。

[splicer.properties ファイル](#)

`splicer.properties` ファイルには、サーブレット・フィルター・ベースのセッション・マネージャーを構成するための、すべての構成オプションが含まれます。

親トピック: [データ・グリッドの構成](#)

データ・グリッド (data grid)への WebSphere Application Server HTTP セッション・パーシスタンスの構成

アプライアンスをセッション管理で使用するよう WebSphere® Application Server アプリケーションを構成します。新規アプリケーションのインストール時にアプライアンスを選択することも、既存のアプリケーションまたはサーバーの設定をアプライアンスを使用するよう更新することもできます。

始める前に

WebSphere Application Server で構成を変更する前に、以下の情報が必要です。

- 構成する WebSphere Application Server セルに対するアクセス権限。
- アプライアンスの IP アドレスまたは完全修飾ホスト名。
- アプライアンスのユーザー・インターフェースにログインするために使用するユーザー ID とパスワード。データ・グリッドを作成するには、データ・キャッシュ作成許可が必要です。
- WebSphere Application Server 構成に WebSphere eXtreme Scale クライアントがインストールされていること。詳しくは、[WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- WebSphere Application Server 管理コンソールで使用可能に設定されるグローバル・セキュリティー。アプライアンスのトランスポート層セキュリティーが使用可能に設定されている場合、またはクライアントでトランスポート層セキュリティーを使用するようになりたい場合は、グローバル・セキュリティーを使用可能に設定してください。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

手順

- アプリケーションのインストール時にセッション管理を構成するには、以下の手順を実行します。
 1. WebSphere Application Server の管理コンソールで、「アプリケーション」 > 「新規アプリケーション」 > 「新規エンタープライズ・アプリケーション」をクリックします。「詳細」パスを選択してアプリケーションを作成し、初期のウィザード・ステップを完了します。
 2. ウィザードの「**eXtreme Scale** セッション管理設定」ステップで、使用するデータ・グリッドを構成します。「セッション・パーシスタンスの管理」フィールドで、**WebSphere DataPower XC10** アプライアンスを選択します。アプライアンス、および使用するアプライアンス上のデータ・グリッド (data grid)の情報を入力します。データ・グリッド (data grid)を作成することも、既にアプライアンスで構成されている既存のデータ・グリッド (data grid)を使用することもできます。

アプライアンス上の既存のデータ・グリッド (data grid)内でセッションを保存する場合は、使用するデータ・グリッド (data grid)の名前を知っていなければなりません。ただし、アプリケーションの構成時にアプライアンスにデータ・グリッド (data grid)を作成することができます。アプライアンス・ユーザー・インターフェースでアプリケーションを構成する前にセッション・データ・グリッド (data grid)を作成したい場合は、「データ・グリッド」 > 「セッション」をクリックします。「追加」アイコン (+) をクリックして、作成するセッション・データ・グリッドの名前を指定します。データ・グリッドの名前では、文字 ^ . ¥ ¥ / , # \$ @ : ; ¥ * ? < > | = + & % [] " " は使用できません。

3. ウィザードのステップを完了して、アプリケーションのインストールを終了します。

wsadmin スクリプトを使用して、アプリケーションをインストールすることもできます。次の例では、**-SessionManagement** パラメーターにより、管理コンソールで作成可能なものと同じ構成を作成します。

```
AdminApp.install('C:/A.ear', '[ -nopreCompileJSPs -distributeApp
-nouseMetaDataFromBinary -nodeployejb -appname A -edition 8.0
-createMBeansForResources -noreloadEnabled -nodeployws -validateinstall
off -noprocessEmbeddedConfig -filepermission .*¥.dll=755#.*¥.so=755#.*¥.a=755#.*
¥.sl=755
-buildVersion Unknown -noallowDispatchRemoteInclude -noallowServiceRemoteInclude
-asyncRequestDispatchType DISABLED -nouseAutoLink -SessionManagement [[true
XC10SessionManagement myXC10.ibm.com::username::password::AGrid80]]
-MapWebModToVH [[MicroWebApp microwebapp.war,WEB-INF/web.xml default_host]
[MicroSipApp
microsipapp.war,WEB-INF/web.xml default_host] [MicroDG1App microdg1app.war,WEB-
INF/web.xml
default_host] [MicroDG2App microdg2app.war,WEB-INF/web.xml default_host]
[MicroSip2App
microsip2app.war,WEB-INF/web.xml default_host]]]')
```

- **WebSphere Application Server** の管理コンソールで既存のアプリケーション上に セッション管理を構成する場合は以下を行います。

注: WebSphere DataPower XC10 アプライアンス を使用するようアプリケーションが設定されると、「セッ

セッション管理のオーバーライド」ボックスにチェック・マークが付けられます。つまり、WebSphere Application Server 構成に対して行われたサーバー・レベルのセッション設定はアプリケーション・レベルのセッション設定によって上書きされることとなります。これが望ましくない場合は、WebSphere DataPower XC10 アプライアンスをサーバー・レベルで使用可能にすることができます。

1. WebSphere Application Server の管理コンソールで、「アプリケーション」 > 「アプリケーション・タイプ」 > 「WebSphere エンタープライズ・アプリケーション」 > 「application_name」 > 「Web モジュール・プロパティ」 > 「セッション管理」 > 「eXtreme Scale セッション管理設定」をクリックします。
2. フィールドを更新して、データ・グリッドへのセッション・パーシスタンスを使用可能にします。

wsadmin スクリプトを使用して、アプリケーションを更新することもできます。次の例では、**-SessionManagement** パラメーターにより管理コンソールで作成するものと同じ構成を作成できます。

```
AdminApp.edit('A-edition9.0', '[ -SessionManagement [[true  
XC10SessionManagement myXC10.ibm.com:!:username:!:password:!:AGrid80]]]')
```

渡される :!: 文字は、区切り文字として使用されます。渡される値は次のとおりです。

```
applicationIdentifier:!:username:!:password:!:  
gridName
```

変更を保存する場合、アプリケーションはアプライアンス上のセッション・パーシスタンスに構成済みのデータ・グリッド (data grid) を使用します。

- 既存のサーバー上でセッション管理を構成するには、以下を行います。

1. WebSphere Application Server の管理コンソールで、「サーバー」 > 「サーバー・タイプ」 > 「WebSphere アプリケーション・サーバー」 > 「server_name」 > 「セッション管理」 > 「eXtreme Scale セッション管理設定」をクリックします。
2. フィールドを更新して、セッション・パーシスタンスを使用可能にします。

また、以下の wsadmin ツール・コマンドで、既存のサーバーでのセッション管理を構成できます。

```
AdminTask.configureServerSessionManagement('[-nodeName my_node  
-serverName server1 -enableSessionManagement true -sessionManagementType  
XC10SessionManagement -XC10SessionManagement [-applianceIdentifier myserver.ibm.com  
-userName -password ***** -gridName myTestGrid]]')
```

変更を保存すると、今後サーバーはサーバー上で稼働するすべてのアプリケーションでセッション・パーシスタンスの構成済み データ・グリッド (data grid) を使用します。

- HTTP セッション構成の他の局面を編集する場合は、splicer.properties ファイルを編集します。**sessionFilterProps** カスタム・プロパティを見つけ、splicer.properties ファイルのパスの場所を取得できます。サーバー・レベルでセッション・パーシスタンスを構成した場合、カスタム・プロパティの名前は com.ibm.websphere.xs.sessionFilterProps です。アプリケーション・レベルでセッション・パーシスタンスを構成した場合、カスタム・プロパティの名前は `<application_name>,com.ibm.websphere.xs.sessionFilterProps` です。これらのカスタム・プロパティは、おそらく、次のいずれかの場所にあります。

- WebSphere Application Server Network Deployment 環境の場合: splicer.properties ファイルはデプロイメント・マネージャー・プロファイル・パス上にあります。
- スタンドアロン WebSphere Application Server 環境の場合: アプリケーション・サーバー上のカスタム・プロパティ

示されたファイルを開き、変更し、ノードを同期できます。それによって、更新されたプロパティ・ファイルが、構成内の他のノードに伝搬されます。セッションを適切に存続させるために、すべてのアプリケーション・サーバー・ノードは、splicer.properties ファイルが指定されたパスにあることを必要とします。

重要: URL 再書き込みを使用するセッションのパーシスタンスを有効にするには、splicer.properties ファイル内の **useURLEncoding** プロパティを true に設定します。

splicer.properties ファイルのプロパティに関して詳しくは、[splicer.properties ファイル](#)を参照してください。

タスクの結果

データ・グリッド (data grid) へのセッションをパーシストするように HTTP セッション・マネージャーが構成されました。セッションがタイムアウトになると、項目はデータ・グリッドから除去されます。WebSphere Application Server 管理コンソールでのセッション・タイムアウト値の更新について詳しくは、[セッション管理設定](#)を参照してください。

アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアント

トから到達できない場合、クライアントは、代わりに WebSphere Application Server の基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとアプライアンス間にネットワーク上の問題がある。
- アプライアンス上のサーバー・プロセスが停止されている。

最長未使用時間セッションは、Web コンテナのセッション・キャッシュから無効にされます。アプライアンス上のデータ・グリッドが使用可能になった場合、Web コンテナ・キャッシュから無効化されたセッションは、リモート・データ・グリッドからデータを取得して、データを新規セッションにロードすることができます。アプライアンス上のデータ・グリッド全体が使用不可の状態、セッションがセッション・キャッシュから無効にされている場合、ユーザー・セッション・データは失われます。この問題があるため、負荷をかけられた状態でシステムを実行している場合は、実動データ・グリッド全体をシャットダウンしないでください。

注意:

このシナリオを構成すると、**IBM WebSphere DataPower XC10** アプライアンスのセキュリティー資格情報が **WebSphere Application Server** 構成に自動的に保管されます。初期構成後にデータ・グリッドの資格情報を変更した場合には、**WebSphere Application Server** には適切な資格情報が含まれなくなります。**eXtreme Scale** セッション管理設定を再度適用することで、資格情報を再設定できます。

次のタスク

- データ・グリッドへのデータの送信を開始する前に、セキュリティーを構成します。詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。
- レプリカを構成します。レプリカにより、基本コピーで障害が発生した場合にもデータ・グリッドのデータが使用可能になります。レプリカを構成するには、「データ・グリッド」>「セッション」>「拡張属性の表示」をクリックします。アプライアンスが集合内にある場合にのみ、レプリカが作成されます。集合内のアプライアンスの数が n の場合、レプリカの最大数は $n-1$ になります。したがって、3つのレプリカを構成するが、集合内にあるアプライアンスが2つのみである場合、作成されるレプリカは1つのみです。集合にアプライアンスを追加すると、追加のレプリカが作成されます。アプライアンスが集合に追加されたときに新規レプリカを作成できるようにするため、レプリカの数適切な数に設定してください。レプリカ数を編集すると、データ・グリッドの内容が消去されます。
- データ・グリッドの容量限度を構成します。データ・グリッドの容量限度の構成により、集合のストレージ容量が予測可能な形で使用されることを確実にできます。詳しくは、[データ・グリッドの最大容量の構成](#)を参照してください。
- DataPower XC10 アプライアンス ユーザー・インターフェース でセッション・データ・グリッド (data grid) をモニターできます。詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

親トピック: [セッション・データ・グリッドの作成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連資料:

[splicer.properties](#) ファイル

関連情報:

☞ [コンソールを使用したエンタープライズ・アプリケーション・ファイルのインストール](#)

☞ [wsadmin スクリプト](#)

WebSphere Application Server での HTTP セッション・マネージャーの構成

WebSphere® Application Server はセッション管理機能を備えています。要求の数が増えるとパフォーマンスが低下します。WebSphere eXtreme Scale には、セッション・レプリカ生成、高可用性、優れたスケーラビリティ、および堅固な構成オプションを備えたセッション管理実装がバンドルされています。

始める前に

- デフォルトでは、XC10 スプライサー・プロパティ・ファイルはシステムが生成するユーザー ID とパスワードを使用します。システム生成ユーザー ID は XC10_USER_GridName です。このユーザー ID のパスワードは変更できません。ただし、ユーザー ID を別のユーザーに変更する場合は、credentialGeneratorProps プロパティを変更することによって、splicer.properties ファイル内のユーザー ID とパスワードの組み合わせを手動で更新する必要があります。詳しくは、[splicer.properties ファイルを編集する方法](#)を参照してください。システム生成のユーザー ID とパスワードは、始動時に splicer.properties ファイルを取得するためにだけ使用されます。これらの資格情報は WebSphere Application Server クライアントでは使用されなくなりました。
- eXtreme Scale セッション・マネージャーを使用するには、WebSphere eXtreme Scale は、WebSphere Application Server または WebSphere Application Server Network Deployment セルにインストールする必要があります。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- HTTP セッション複製用の WebSphere eXtreme Scale が WebSphere Application Server で使用される場合は、該当するすべての Web アプリケーションおよびその Web アプリケーションをホストするアプリケーション・サーバーについて「オーバーフローの許可」セッション管理設定にチェック・マークを付ける必要があります。詳しくは、「[セッション管理設定](#)」を参照してください。
- アプライアンスで Secure Sockets Layer (SSL) を使用可能に設定した場合は、WebSphere Application Server でグローバル・セキュリティーを使用可能に設定する必要があります。SSL がサポートされている集合に対して SSL を使用したい場合も、これを使用可能に設定する必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#)を参照してください。

このタスクについて

WebSphere eXtreme Scale HTTP セッション・マネージャーは、キャッシング用に、組み込みサーバーとリモート・サーバーの両方をサポートします。

• 組み込みシナリオ

組み込みシナリオでは、サーブレットが実行される同じプロセス内でデータ・グリッド・サーバーが相互に連結されています。セッション・マネージャーはローカルの ObjectGrid インスタンスと直接通信できるため、コストのかかるネットワーク遅延を回避することができます。

WebSphere Application Server を使用している場合は、提供された [wxs_home/session/samples/objectGrid.xml](#) および [wxs_home/session/samples/objectGridDeployment.xml](#) ファイルを、ご使用の Web アーカイブ (WAR) ファイルの META-INF ディレクトリーに配置してください。アプリケーションが始動して、セッション・マネージャーと同じプロセス内の eXtreme Scale コンテナを自動的に始動すると、eXtreme Scale がこれらのファイルを自動的に検出します。

objectGridDeployment.xml ファイルは変更することができます。複製を同期にするか非同期にするか、またはレプリカをいくつ構成するかによってこのファイルを変更します。

• リモート・サーバー・シナリオ

リモート・サーバー・シナリオでは、実行されるコンテナ・サーバーはサーブレットとは異なるプロセスにあります。セッション・マネージャーはリモートのコンテナ・サーバーと通信します。リモートのネットワーク接続のコンテナ・サーバーを使用するためには、カタログ・サービス・ドメインのホスト名およびポート番号によってセッション・マネージャーを構成する必要があります。そうすると、セッション・マネージャーは、eXtreme Scale クライアント接続を使用して、カタログ・サーバーおよびコンテナ・サーバーと通信します。

コンテナ・サーバーを独立したスタンドアロン・プロセス内で開始する場合は、セッション・マネージャーのサンプル・ディレクトリーで提供される objectGridStandAlone.xml ファイルと objectGridDeploymentStandAlone.xml ファイルを使用して、データ・グリッド・コンテナを開始します。

手順

- アプリケーションを接合することで、アプリケーションがセッション・マネージャーを使用できるようにします。セッション・マネージャーを使用するためには、適切なフィルター宣言をアプリケーションの Web デプロイメント記述子に追加する必要があります。さらに、セッション・マネージャー構成パラメーターが、デプロイメ

ント記述子内のサブレット・コンテキスト初期化パラメーターという形式でセッション・マネージャーに渡されます。この情報は、以下に示す複数の方法でアプリケーションに導入することができます。

- **WebSphere Application Server との自動接合**

アプリケーションのインストール時にデータ・グリッドに対して HTTP セッション・マネージャーを使用するようにアプリケーションを構成することができます。また、アプリケーションまたはサーバーの構成を編集して、WebSphere eXtreme Scale HTTP セッション・マネージャーを使用することもできます。詳しくは、[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パースタンスの構成](#)を参照してください。

- **カスタム・プロパティを使用したアプリケーションの自動接合**

WebSphere Application Server または WebSphere Application Server Network Deployment でアプリケーションを実行している場合には、アプリケーションを手動で接合する必要はありません。

カスタム・プロパティをセルまたはサーバーに追加して、そのスコープにあるすべての Web アプリケーションに `splicer.properties` ファイルを設定します。次のステップを実行して、カスタム・プロパティを構成します。

- a. WebSphere Application Server 管理コンソールで、カスタム・プロパティを設定する正しいパスにナビゲートし、`splicer.properties` ファイルの場所を指示します。
 - カスタム・プロパティをすべてのアプリケーションまたは特定のアプリケーションに設定するには、「システム管理」 > 「セル」 > 「カスタム・プロパティ」をクリックします。
 - 特定のアプリケーション・サーバー上のすべてのアプリケーションに適用するカスタム・プロパティを設定するには、「アプリケーション・サーバー」 > 「<server_name>」 > 「管理」 > 「カスタム・プロパティ」をクリックします。プロパティ名は `com.ibm.websphere.xs.sessionFilterProps` で、その値はアプリケーションが必要とする `splicer.properties` ファイルの場所です。ファイルの場所のパスは、例えば `/opt/splicer.properties` です。
- b. `com.ibm.websphere.xs.sessionFilterProps` カスタム・プロパティを追加します。このカスタム・プロパティの値には、編集する `splicer.properties` ファイルのロケーションが指定されています。このファイルは、デプロイメント・マネージャーに存在します。セル・レベルのカスタム・プロパティを使用して特定のアプリケーション用の `splicer.properties` ファイルを指示する必要がある場合は、カスタム・プロパティの名前を `<application_name>,com.ibm.websphere.xs.sessionFilterProps` のように入力します。ここで、`application_name` は、カスタム・プロパティを適用するアプリケーションの名前を示します。

重要: 更新済み `splicer.properties` ファイルが、セッション・レプリカ生成のために接合されるアプリケーションをホスティングしているアプリケーション・サーバーを含んでいるすべてのノードで、同じパス上に存在することを確認してください。

使用可能なスコープはセル、サーバー、およびアプリケーションであり、デプロイメント・マネージャーで実行している場合にのみ使用可能です。別のスコープが必要な場合は、Web アプリケーションを手動で接合してください。

要確認: 自動接合オプションは、アプリケーションを実行しているすべてのノードの同じパスに `splicer.properties` ファイルが存在する場合にのみ機能する点にも注意してください。Windows ノードと UNIX ノードがともに存在する混合環境では、このオプションは使用できないため、アプリケーションを手動で接合する必要があります。

- **addObjectGridFilter スクリプトによるアプリケーションの接合**

eXtreme Scale とともに提供されるコマンド行スクリプトを使用して、フィルター宣言と構成によってアプリケーションをサブレット・コンテキスト初期化パラメーターの形式で接合します。WebSphere Application Server デプロイメントの場合、このスクリプトは `<was_home>/optionalLibraries/ObjectGrid/session/bin/addObjectGridFilter.bat/sh` にあります。スタンドアロン・デプロイメントの場合、スクリプトは `WXS_HOME/ObjectGrid/session/bin/addObjectGridFilter.sh/bat` にあります。

addObjectGridFilter スクリプトは 2 つのパラメーターを使用します。

- アプリケーション - 接合するエンタープライズ・アーカイブ・ファイルへの絶対パス
- 各種構成プロパティが入ったスプライサー・プロパティ・ファイルへの絶対パス

このスクリプトの使用形式は次のとおりです。

Windows

```
addObjectGridFilter.bat [ear_file] [splicer_properties_file]
```

UNIX

```
addObjectGridFilter.sh [ear_file] [splicer_properties_file]
```

UNIX UNIX 上の WebSphere Application Server にインストールされている eXtreme Scale の使用例:

- a. cd `wxs_home/optionalLibraries/ObjectGrid/session/bin`
- b. `addObjectGridFilter.sh /tmp/mySessionTest.ear`
`was_root/optionalLibraries/ObjectGrid/session/samples/splicer.properties`

UNIX UNIX 上のスタンドアロン・ディレクトリーにインストールされている eXtreme Scale の使用例:

- a. cd `was_root/session/bin`
- b. `addObjectGridFilter.sh /tmp/mySessionTest.ear`
`was_root/session/samples/splicer.properties`

接合されるサーブレット・フィルターは構成値のデフォルトを保持します。これらのデフォルト値は、2 番目の引数にあるプロパティ・ファイルで指定する構成オプションでオーバーライドできます。使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

eXtreme Scale インストールとともに提供されるサンプルの `splicer.properties` ファイルを変更して使用することができます。また、各サーブレットを拡張することによってセッション・マネージャーを挿入する、`addObjectGridServlets` スクリプトも使用できます。ただし、推奨スクリプトは `addObjectGridFilter` スクリプトです。

◦ **Ant** ビルド・スクリプトによるアプリケーションの手動接合

WebSphere eXtreme Scale には Apache Ant で使用できる `build.xml` ファイルが同梱されています。このファイルは WebSphere Application Server インストールの `was_root/bin` フォルダーに含まれています。`build.xml` ファイルを変更して、セッション・マネージャー構成プロパティを変更できます。構成プロパティは `splicer.properties` ファイル内のプロパティ名と同一です。`build.xml` を変更し、次のコマンドを実行して Ant プロセスを開始します。

- **UNIX** `ant.sh`、`ws_ant.sh`
- **Windows** `ant.bat`、`ws_ant.bat`

(UNIX) または (Windows)

◦ **Web** 記述子の手動更新

Web アプリケーションに同梱されている `web.xml` ファイルを編集して、フィルター宣言、そのサーブレット・マッピング、およびサーブレット・コンテキスト初期化パラメーターが組み込まれるようにします。この方法はエラーを起こしやすいため、使用しないようにしてください。

使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

2. アプリケーションをデプロイします。サーバーやクラスターに対して通常使用する手順に従ってアプリケーションをデプロイしてください。アプリケーションをデプロイした後、アプリケーションを始動することができます。
3. アプリケーションにアクセスします。これで、セッション・マネージャーおよび WebSphere eXtreme Scale と対話するアプリケーションにアクセスすることができます。

次のタスク

アプリケーションの装備時にセッション・マネージャーの構成属性の大部分を変更して、セッション・マネージャーを使用するようにすることができます。これらの属性には、同期または非同期のレプリカ生成、メモリー内セッション・テーブル・サイズなどがあります。アプリケーションの装備時に変更できる属性を別にすれば、アプリケーションのデプロイメント後に変更できるその他の構成属性は、WebSphere eXtreme Scale サーバー・クラスター・トポロジーと、それらのクラスターのクライアント (セッション・マネージャー) がそれらのクラスターに接続する方法に関する属性のみです。

リモート・シナリオの動作: アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアントから到達できない場合、クライアントは、代わりに WebSphere Application Server の基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとリモート・コンテナ・サーバー間のネットワークの問題
- リモート・コンテナ・サーバーのプロセスが停止した場合

sessionTableSize パラメーターによって指定される、メモリー内に保持されるセッション参照の数は、セッションが基本 Web コンテナ内に保管されている場合、そのまま維持されます。セッション数が **sessionTableSize** の値を超えると、最長未使用時間を基にセッションが Web コンテナ・セッション・キャッシュで無効化されます。リモート・データ・グリッドが使用可能になると、Web コンテナ・キャッシュで無効化されたセッションは、リモート・データ・グリッドからデータを取得し、データを新規セッションにロードできます。リモート・データ・グリッド全体が使用不可なまま、セッションがセッション・キャッシュで無効化されると、ユーザー・セッション・データは失われま

す。このような問題があるため、負荷の下でシステムを実行する場合、実動リモート・データ・グリッド全体をシャットダウンすることはしないでください。

親トピック: [セッション・データ・グリッドの作成](#)

WebSphere Portal での HTTP セッション・マネージャーの構成

WebSphere® Portal の HTTP セッションをデータ・グリッドに保持できます。

始める前に

WebSphere eXtreme Scale クライアント と WebSphere Portal 環境が、次の要件を満たしている必要があります。

- WebSphere Application Server および WebSphere Portal ノードに WebSphere eXtreme Scale クライアント をインストールしてください。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- WebSphere Portal バージョン 7 以降。
- カスタム・ポートレットは、WebSphere Portal 内で構成しなければなりません。現在、WebSphere Portal に 付属の管理ポートレットは、データ・グリッドと統合できません。

このタスクについて

WebSphere DataPower XC10 アプライアンス を WebSphere Portal 環境に導入することは、以下のシナリオでメリットがあります。

重要: 以下のシナリオでは、メリットについて紹介しますが、WebSphere DataPower XC10 アプライアンス を環境に導入することにより、WebSphere Portal 層での プロセッサ使用量が増える場合もあります。

- セッション・パーシスタンスが必要な場合

例えば、WebSphere Portal Server の 障害時でもカスタム・ポートレットのセッション・データを使用可能な状態で 維持する必要がある場合は、HTTP セッションを WebSphere DataPower XC10 アプライアンス データ・グリッドに 保持できます。複数のサーバーにデータを複製しておくことで、データの可用性が高まります。

- 複数データ・センター・トポロジー

ロケーションが物理的に異なる複数のデータ・センターに またがるトポロジーの場合、WebSphere Portal HTTP セッションを WebSphere DataPower XC10 アプライアンス データ・グリッドに 保持できます。セッションは、複数あるデータ・センター内のデータ・グリッドに 複製されます。あるデータ・センターで障害が起こると、セッションは、データ・グリッドのデータのコピーを 保持する別のデータ・センターにロールオーバーされます。

- WebSphere Portal Server 層のメモリー所要量を低下させる場合

セッション・データを コンテナ・サーバーのリモート層にオフロードすることで、セッションのサブセットが WebSphere Portal Server 上に 存在することになります。このデータのオフロードにより、WebSphere Portal Server 層のメモリー所要量が低下します。

手順

1. セッションをデータ・グリッドに格納できるように、wps WebSphere Portal アプリケーションと任意のカスタム・ポートレットを構成します。

詳しくは、[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パーシスタンスの構成](#)を参照してください。この操作により、カスタム・ポートレットが接合され、データ・グリッドでセッション・パーシスタンスが有効になります。

2. WebSphere Portal サーバーとアプライアンスに対して Transport Layer Security/Secure Sockets Layer (TLS/SSL) が構成されている場合は、TLS/SSL トラストストアを構成する必要があります。
 - この結果、WebSphere Portal サーバーからアプライアンスへのアウトバウンド通信で TL/SSL 通信が使用される場合は、アプライアンス証明書を WebSphere Application Server 構成に追加する必要があります。以下のように **addXC10PublicCert.py** スクリプトを使用します。このスクリプトは `was_root/bin` ディレクトリにあります。

```
wsadmin.bat -conntype SOAP -port <PORTAL_SERVER_SOAP_PORT> -lang jython  
-user wpsadmin -password wpsadmin -f addXC10PublicCert.py
```

- この結果、アプライアンスから WebSphere Portal サーバーへのインバウンド通信で TLS/SSL が使用される場合、WebSphere Portal サーバーの公開証明書を組み込むようにアプライアンス・トラストストアを更新してください。トラストストアを更新することで、アプライアンスと WebSphere Portal の間の通信が可能になります。
 - a. Portal Server 個人証明書の公開鍵を抽出します。 **IKEYMAN** ユーティリティーを使用します。このユーティリティーにより `.arm` ファイルが作成されます。詳しくは、『[トラストストア・ファイル用の公開証明書の抽出](#)』を参照してください。
 - b. アプライアンスのパブリック・トラストストアをダウンロードします。詳しくは、[WebSphere](#)

[Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

- c. **iKeyman** ユーティリティを使用して、アプライアンスから抽出した `truststore.jks` ファイルを、`.arm` ファイル内の Portal Server 公開証明書で更新します。詳しくは、『[署名者証明書のインポート](#)』を参照してください。
 - d. 更新したトラストストア・ファイルをアプライアンスにアップロードします。トラストストアをアップロードした後に、「**TLS 設定の送信**」をクリックします。TLS 設定を送信し、新規トラストストアが集合内の他のアプライアンスに追加されると、集合が自動的に再始動します。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。
3. WebSphere Portal サーバーの一部のバージョンでは、Cookie が HTTP 応答に追加されるとランタイム・エラーが起こることがあります。WebSphere DataPower XC10 アプライアンスはフェイルオーバーその他の目的で Cookie を追加するので、これらの Cookie は WebSphere Portal サーバーの Cookie 無視リストに追加する必要があります。詳しくは、IBM WebSphere Portal wiki に掲載されている、複数のユーザーが共有するページのキャッシングに関するページの `cookie.ignore.regex` パラメーターのセクションを参照してください。リストに追加する必要がある 2 つの Cookie は `IBMID.*` と `IBMSessionHandle.*` です。更新されたリストは、例えば次のようになります。`"digest\\.ignore.*|LtpaToken|LtpaToken2|JSESSIONID|IBMID.*|IBMSessionHandle.*"`。詳しくは、IBM WebSphere Portal wiki の[複数のユーザーが共有するページのキャッシング \(Caching pages shared by multiple users\)](#)を参照してください。
 4. WebSphere Portal Server を再始動します。詳しくは、『[WebSphere Portal Version 7: Starting and stopping servers, deployment managers, and node agents \(WebSphere Portal バージョン 7: サーバー、デプロイメント・マネージャー、およびノード・エージェントの開始と停止\)](#)』を参照してください。

タスクの結果

WebSphere Portal Server にアクセスでき、構成されているカスタム・ポートレットの HTTP セッション・データはデータ・グリッドに保持されます。

アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアントから到達できない場合、クライアントは、代わりに WebSphere Application Server の基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとリモート・コンテナ・サーバー間のネットワークの問題
- リモート・コンテナ・サーバーのプロセスが停止した場合

sessionTableSize パラメーターによって指定される、メモリー内に保持されるセッション参照の数は、セッションが基本 Web コンテナ内に保管されている場合、そのまま維持されます。セッション数が **sessionTableSize** の値を超えると、最長未使用時間を基にセッションが Web コンテナ・セッション・キャッシュで無効化されます。リモート・データ・グリッドが使用可能になると、Web コンテナ・キャッシュで無効化されたセッションは、リモート・データ・グリッドからデータを取得し、データを新規セッションにロードできます。リモート・データ・グリッド全体が使用不可なまま、セッションがセッション・キャッシュで無効化されると、ユーザーのセッション・データは失われます。このような問題があるため、負荷の下でシステムを実行する場合、実動リモート・データ・グリッド全体をシャットダウンすることはしないでください。

親トピック: [セッション・データ・グリッドの作成](#)

関連タスク:

[各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成](#)
[HTTP コマンド・インターフェースでの管理](#)

関連情報:

☞ [IKEYMAN グラフィカル・インターフェースを使用した鍵の管理](#)

各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成

WebSphere® eXtreme Scale クライアントには、Web コンテナのデフォルト・セッション・マネージャーをオーバーライドするセッション管理実装がバンドルされています。この実装は、セッション・レプリカ生成、高可用性、より優れたスケーラビリティと構成オプションを提供します。Tomcat などさまざまなアプリケーション・サーバーに対して WebSphere eXtreme Scale クライアント セッション複製マネージャーおよび汎用組み込み ObjectGrid コンテナの開始を有効にします。

始める前に

WebSphere DataPower® XC10 アプライアンスで SSL と一緒に Transport Layer Security (TLS) を使用可能に設定することができます。詳しくは、[データ・グリッド・アプリケーション用の TLS の構成](#)を参照してください。

このタスクについて

WebSphere Application Server Community Edition などの WebSphere Application Server を実行していない他のアプリケーション・サーバーで HTTP セッション・マネージャーを使用できます。データ・グリッドを使用するように他のアプリケーション・サーバーを構成するには、アプリケーションを接合して、WebSphere eXtreme Scale クライアント Java アーカイブ (JAR) ファイルをアプリケーションに取り込む必要があります。

手順

1. アプリケーションを接合することで、アプリケーションがセッション・マネージャーを使用できるようにします。セッション・マネージャーを使用するためには、適切なフィルター宣言をアプリケーションの Web デプロイメント記述子に追加する必要があります。さらに、セッション・マネージャー構成パラメーターが、デプロイメント記述子内のサーブレット・コンテキスト初期化パラメーターという形式でセッション・マネージャーに渡されます。この情報は、以下に示す 3 とおりの方法でアプリケーションに導入することができます。

- **addObjectGridFilter** スクリプト:

WebSphere eXtreme Scale クライアントとともに提供されるコマンド行スクリプトを使用して、フィルター宣言と構成によってアプリケーションをサーブレット・コンテキスト初期化パラメーターの形式で接合します。[wxs_home/session/bin/addObjectGridFilter.sh|bat](#) スクリプトは、2 つのパラメーターを使用します。1 つは接合するエンタープライズ・アーカイブ (EAR) ファイルまたは Web アーカイブ (WAR) ファイルへの絶対パスで、もう 1 つは各種構成プロパティが含まれたスプライサー・プロパティ・ファイルへの絶対パスです。このスクリプトの使用形式は以下の通りです。

Windows

```
addObjectGridFilter.bat <ear_or_war_file> <splicer_properties_file>
```

UNIX

```
addObjectGridFilter.sh <ear_or_war_file> <splicer_properties_file>
```

UNIX

UNIX 上のスタンドアロン・ディレクトリーにインストールされている WebSphere eXtreme Scale クライアントの使用例:

- a. `cd wxs_home/session/bin`
- b. `addObjectGridFilter.sh /tmp/mySessionTest.ear
wxs_home/session/samples/splicer.properties`

接合されるサーブレット・フィルターは構成値のデフォルトを保持します。これらのデフォルト値は、2 番目の引数にあるプロパティ・ファイルで指定する構成オプションでオーバーライドできます。使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

WebSphere eXtreme Scale クライアント インストールとともに提供されるサンプルの `splicer.properties` ファイルを変更して使用することができます。ただし、ファイルを調べ、使用したいプロパティ (`catalogHostPort` プロパティなど) のコメントが外されていることを確認してください。また、各サーブレットを拡張することによってセッション・マネージャーを挿入する、**addObjectGridServlets** スクリプトも使用できます。ただし、推奨スクリプトは **addObjectGridFilter** スクリプトです。

- Ant ビルド・スクリプト:

WebSphere eXtreme Scale クライアントには Apache Ant で使用できる `build.xml` ファイルが同梱されています。このファイルは WebSphere Application Server インストールの [was_root](#)/bin フォルダに含まれています。`build.xml` ファイルを変更して、セッション・マネージャー構成プロパティを変更できます。構成プロパティは `splicer.properties` ファイル内のプロパティ名と同一です。`build.xml` ファイルの変更後に、`ant.sh`、`ws_ant.sh` (UNIX) または `ant.bat`、`ws_ant.bat` (Windows) を実行することで、Ant プロセスを呼び出します。

- 手動による Web 記述子の更新:

Web アプリケーションに同梱されている web.xml ファイルを編集して、フィルター宣言、そのサーブレット・マッピング、およびサーブレット・コンテキスト初期化パラメーターが組み込まれるようにします。この方法はエラーを起こしやすいため、使用しないようにしてください。

使用できるパラメーターのリストについては、[サーブレット・コンテキスト初期化パラメーター](#)を参照してください。

2. WebSphere eXtreme Scale クライアント セッション・レプリカ生成マネージャーの JAR ファイルをアプリケーションに取り込みます。ファイルは、アプリケーション・モジュールの WEB-INF/lib ディレクトリーまたはアプリケーション・サーバーのクラスパスに組み込むことができます。必要な JAR ファイルは、以下のように、使用しているコンテナのタイプによって異なります。
 - リモート・コンテナ・サーバー: ogclient.jar と sessionobjectgrid.jar
 - 組み込みコンテナ・サーバー: objectgrid.jar と sessionobjectgrid.jar
3. アプリケーションをデプロイします。サーバーやクラスターに対して通常使用する手順に従ってアプリケーションをデプロイしてください。アプリケーションをデプロイした後、アプリケーションを始動することができます。
4. アプリケーションにアクセスします。これで、セッション・マネージャーおよび WebSphere eXtreme Scale クライアント と対話するアプリケーションにアクセスすることができます。

次のタスク

アプリケーションの装備時にセッション・マネージャーの構成属性の大多数を変更して、セッション・マネージャーを使用できるようにすることができます。これらの属性には、レプリカ生成タイプ (同期または非同期) のバリエーション、メモリー内セッション・テーブルのサイズなどがあります。アプリケーションの装備時に変更できる属性を別にすれば、アプリケーションのデプロイメント後に変更できるその他の構成属性は、WebSphere eXtreme Scale サーバー・クラスター・トポロジーと、それらのクラスターのクライアント (セッション・マネージャー) がそれらのクラスターに接続する方法に関する属性のみです。

リモート・シナリオの動作: アプリケーション・セッション・データをホスティングしている全データ・グリッドに Web コンテナ・クライアントから到達できない場合、クライアントは、代わりにアプリケーション・サーバーの基本 Web コンテナをセッション管理に使用します。次のようなシナリオでは、データ・グリッドに到達できないことがあります。

- Web コンテナとリモート・コンテナ・サーバー間のネットワークの問題
- リモート・コンテナ・サーバーのプロセスが停止した場合

sessionTableSize パラメーターによって指定される、メモリー内に保持されるセッション参照の数は、セッションが基本 Web コンテナ内に保管されている場合、そのまま維持されます。セッション数が **sessionTableSize** の値を超えると、最長未使用時間を基にセッションが Web コンテナ・セッション・キャッシュで無効化されます。リモート・データ・グリッドが使用可能になると、Web コンテナ・キャッシュで無効化されたセッションは、リモート・データ・グリッドからデータを取得し、データを新規セッションにロードできます。リモート・データ・グリッド全体が使用不可なまま、セッションがセッション・キャッシュで無効化されると、ユーザー・セッション・データは失われます。このような問題があるため、負荷の下でシステムを実行する場合、実動リモート・データ・グリッド全体をシャットダウンすることはしないでください。

親トピック: [セッション・データ・グリッドの作成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連タスク:

[WebSphere Portal での HTTP セッション・マネージャーの構成](#)

[HTTP コマンド・インターフェースでの管理](#)

関連資料:

[splicer.properties ファイル](#)

関連情報:

☞ [コンソールを使用したエンタープライズ・アプリケーション・ファイルのインストール](#)

☞ [wsadmin スクリプト](#)

☞ [IKEYMAN グラフィカル・インターフェースを使用した鍵の管理](#)

ASP.NET セッション状態ストア・プロバイダーの構成

2.5+ データ・グリッド内の ASP.NET アプリケーションからセッション状態を保管できます。

始める前に

- WebSphere eXtreme Scale Client for .NET をインストールします。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストールの概要](#)を参照してください。
- ご使用の .NET 環境が、システム要件を満たしている必要があります。詳しくは、[Microsoft .NET に関する考慮事項](#)を参照してください。
- ご使用の .NET アプリケーションがセッション状態を維持するように構成されている必要があります。

2.5+ [ASP.NET セッション状態ストア・プロバイダーで使用するデータ・グリッドの作成](#)

ASP.NET アプリケーションのセッション状態を保存するためのデータ・グリッドの作成

2.5+ [ASP.NET セッション状態ストア・プロバイダーを使用する .NET アプリケーションの構成](#)

ASP.NET セッション状態ストア・プロバイダーを構成するには、web.config ファイルを更新して、ASP.NET アプリケーションにセッション状態ストア・プロバイダーとその構成を組み込む必要があります。

親トピック: [セッション・データ・グリッドの作成](#)

ASP.NET セッション状態ストア・プロバイダーで使用するデータ・グリッドの作成

2.5+ ASP.NET アプリケーションのセッション状態を保存するためのデータ・グリッドの作成

手順

ASP.NET セッション状態ストア・プロバイダーにデータ・グリッドを作成します。ユーザー・インターフェースで、「データ・グリッド」 > 「セッション」をクリックします。「追加」アイコン (+) をクリックして、作成するセッション・データ・グリッドの名前を指定します。デフォルトでは、ASP.NET セッション状態ストア・プロバイダーは、session というデータ・グリッドと連動するように構成されます。session 以外の名前を持つデータ・グリッドを作成した場合は、ASP.NET セッション状態ストア・プロバイダーを、作成したデータ・グリッドの名前で構成する必要があります。

次のタスク

データ・グリッドを使用するために、Web アプリケーションの構成を更新してください。詳しくは、[ASP.NET セッション状態ストア・プロバイダーを使用する .NET アプリケーションの構成](#)を参照してください。

親トピック: **.NET** **2.5+** [ASP.NET セッション状態ストア・プロバイダーの構成](#)

ASP.NET セッション状態ストア・プロバイダーを使用する .NET アプリケーションの構成

2.5+ ASP.NET セッション状態ストア・プロバイダーを構成するには、web.config ファイルを更新して、ASP.NET アプリケーションにセッション状態ストア・プロバイダーとその構成を組み込む必要があります。

始める前に

- ASP.NET HTTP セッション状態を保管するデータ・グリッドを構成します。詳しくは、[ASP.NET セッション状態ストア・プロバイダーで使用するデータ・グリッドの作成](#)を参照してください。
- カタログ・サーバーのホストおよびポートが必要です。アプライアンス・ユーザー・インターフェースでカタログ・サーバーのホストとポートを取得するには、「集合」 > 「メンバー」をクリックします。集合メンバーを選択します。カタログ・サーバーの IP とポート番号が表示されます。

手順

1. ASP.NET アプリケーションの web.config ファイルを、ASP.NET セッション状態ストア・プロバイダーの設定で更新します。web.config ファイルに、次の例の太字体のテキストを更新または追加する必要があります。

```
<system.web>
  ...
  <sessionState
    mode="Custom"
    customProvider="WxsSessionStateStoreProvider">
    <providers>
      <add
        name="WxsSessionStateStoreProvider"
        type="IBM.WebSphere.Caching.SessionStateStore.WxsSessionStateStore,
          IBM.WebSphere.Caching,
          Version=8.6.0.2, Culture=neutral,
          PublicKeyToken=b439a24ee43b0816"
        wxsPropertyFile="¥optional¥path¥to¥NET-
client.properties"
        wxsHostAndPort="optionalHostAndPort"
        wxsGridName="session"
        wxsMapName="ASPNET.SessionState"
      />
    </providers>
  </sessionState>
  ...
</system.web>
```

wxsPropertyFile (オプション)

プロバイダーが Connect API によるデータ・グリッドへの接続時に使用する完全修飾プロパティ・ファイルを指定します。この属性が指定されなかったか、空文字列である場合、プロバイダーは Web アプリケーション・プロセスの現行ランタイム・ディレクトリーで Client.Net.properties ファイルを探します。このファイルがランタイム・ディレクトリーで見つからない場合、プロバイダーは [net_client_home](#)\config ディレクトリーで Client.Net.properties ファイルを探します。

wxsHostAndPort

セッション状態ストア・プロバイダーがデータ・グリッドにアクセスするときに接続する、カタログ・サーバーのホストとポートのペアのコンマ区切りリストを指定します。形式は次のとおりです。

```
<host name or host ip address>:<tcp port>[,<host name or host ip address>:<tcp port>]
```

wxsGridName (オプション)

ASP.NET セッション・ストア・プロバイダーの接続先データ・グリッドの名前を指定します。ASP.NET セッション状態のデータ・グリッドを作成した場合は、作成したデータ・グリッド名を指定してください。値を指定しない場合、プロバイダーは session データ・グリッドに接続します。

wxsMapName (オプション)

プロバイダーの接続先マップを指定します。値を指定しない場合、プロバイダーは ASPNET.SessionState マップに接続します。

2. ターゲット Web アプリケーションを再始動します。IIS がプロバイダーをロードできるように、Web アプリケ

セッションは、再始動する必要があります。多くの場合、web.config ファイルが変更されて、現行 HTTP 要求の処理が完了すると、自動的に再始動が行われます。

タスクの結果

ASP.NET アプリケーションからの ASP.NET セッション状態がデータ・グリッドに保管されます。

親トピック: [.NET 2.5+ ASP.NET セッション状態ストア・プロバイダーの構成](#)

HTTP セッション・サイズの表示

2.5+ **xscmd** ユーティリティを使用して、Java アプリケーションでのセッションのサイズを表示することができます。セッション・サイズを表示することで、データ・グリッドの問題のトラブルシューティングが容易になります。

始める前に

- カタログ・サーバーに接続するためには、**xscmd** ユーティリティがセットアップされていなければなりません。詳しくは、[xscmd ユーティリティによる管理](#)を参照してください。
- セッション・サイズを表示するためには、そのセッションのセッション ID とセッション・ハンドルが必要です。

手順

xscmd -c showSessionSize コマンドを実行します。コマンドの構文は以下のとおりです。

```
xscmd.bat|sh -c showSessionSize -cep hostname:port(,hostname:port)
-sid session_ID -sh session_handle -g grid_name
[-wacr web_application_context_root]
```

例えば、以下のようなコマンドを実行できます。

```
xscmd.bat -c showSessionSize -g session -cep 9.42.139.213:2809 -sh c -sid
LFMzQnDX5k87xztMF3ri6jU -wacr /A -user xcadmin -pwd xcadmin
```

コマンドが完了すると、以下のセッション情報が表示されます。

```
*** Session Id: LFMzQnDX5k87xztMF3ri6jU
Number of session attributes: 2
Total size of session metadata: 488 bytes
Total size of session attributes: 243 bytes
Total size of session: 731 bytes

Metadata Key and Value:
Key: LFMzQnDX5k87xztMF3ri6jU/A
Size of key: 88 bytes
Size of value: 400 bytes

Individual Attribute Key and Value:
Key: LFMzQnDX5k87xztMF3ri6jU/A_session.reqCount
Size of key: 128 bytes
Size of value: 2 bytes
Key: LFMzQnDX5k87xztMF3ri6jU/A_user
Size of key: 104 bytes
Size of value: 9 bytes
```

親トピック: [セッション・データ・グリッドの作成](#)

サブレット・コンテキスト初期化パラメーター

以下に示すサブレット・コンテキスト初期化パラメーターのリストは、選択した接続メソッドに必要なスプライサー・プロパティ・ファイルに指定できるものです。

パラメーター

applicationQualifiedCookies

true または false のいずれかのストリング値。ご使用の環境に、固有の Cookie 名を使用するアプリケーションが複数ある場合は、true に設定します。デフォルトは false で、すべてのアプリケーションが同じ Cookie 名を使用していることが前提となっています。

authenticationRetryCount

資格情報の有効期限が切れている場合の認証の再試行カウントを指定します。値が 0 に設定されていると、認証の再試行は行われません。

catalogHostPort

カタログ・サーバーに接続して、クライアント・サイドの ObjectGrid インスタンスを取得できます。値の形式は、host:port<,host:port> でなければなりません。host は、カタログ・サーバーが実行されているリスナー・ホストです。port は、そのカタログ・サーバー・プロセスのリスナー・ポートです。このリストは任意の長さにすることができ、ブートストラッピングにのみ使用されます。最初の実行可能なアドレスが使用されます。このパラメーターは、**catalog.services.cluster** プロパティが構成されている場合は、WebSphere® Application Server 内でオプションです。

credentialAuthentication

クライアントの資格情報認証のサポートを指定します。指定できる値は以下のとおりです。

- Never- クライアントは資格情報認証をサポートしません。
- Supported - サーバーが資格情報認証をサポートする場合のみ、クライアントも資格情報認証をサポートします。
- Required - クライアントは資格情報認証を必要とします。デフォルト値は Supported です。

cookieDomain

ホストをまたいだセッションへのアクセスが必要かどうかを指定します。値を、ホスト間に共通のドメインの名前に設定してください。

cookiePath

com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator インターフェースを実装するクラスの名前を指定します。このクラスを使用して、クライアントの資格情報が取得されます。

credentialGeneratorClass

com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator インターフェースを実装するクラスの名前。このクラスを使用して、クライアントの資格情報が取得されます。

credentialGeneratorProps

CredentialGenerator 実装クラスのプロパティ。このプロパティが、setProperty(String) メソッドを使用してオブジェクトに設定されます。credentialGeneratorProps 値は、**credentialGeneratorClass** プロパティの値が非ヌルの場合にのみ使用されます。

enableSessionStats

true または false のいずれかのストリング値。eXtreme Scale クライアント HTTP セッションの統計追跡を使用可能にします。

fragmentedSession

true または false のいずれかのストリング値。デフォルト値は true です。この設定を使用して、製品がセッション・データをエントリ全体として保管するか、それぞれの属性を個別に保管するかを制御します。

Web アプリケーションのセッションが持っている属性の数が多い場合や属性のサイズが大きい場合は、fragmentedSession パラメーターを true に設定してください。すべての属性はデータ・グリッド内の同じキーに保管されるため、セッションが持っている属性の数が少ない場合は fragmentedSession を false に設定してください。

以前のフィルター・ベースの実装環境では、このプロパティは「persistenceMechanism」と呼ばれており、設定可能な値は ObjectGridStore (フラグメント化されている場合) および ObjectGridAtomicSessionStore (フラグメント化されていない場合) でした。

objectGridType

REMOTE または EMBEDDED のいずれかのストリング値。デフォルトは REMOTE です。

このパラメーターが REMOTE に設定されている場合、セッション・データは Web アプリケーションが実行されているサーバーの外に保管されます。

このパラメーターが EMBEDDED に設定されている場合は、Web アプリケーションが実行されているアプリケーション・サーバー・プロセス内で組み込みの eXtreme Scale コンテナが開始されます。

objectGridName

特定の Web アプリケーションで使用する ObjectGrid インスタンスの名前を定義するストリング値。デフォルトの名前は「session」です。

eXtreme Scale コンテナ・サーバーの始動に使用する ObjectGrid XML ファイルとデプロイメント XML ファイルの両方で、このプロパティは objectGridName を反映する必要があります。

objectGridXML

objectgrid.xml ファイルが置かれているロケーション。objectGridType=EMBEDDED および **objectGridXML** プロパティが指定されていない場合、eXtreme Scale ライブラリーにパッケージされた組み込み XML ファイルが自動的にロードされます。

objectGridDeploymentXML

objectGrid デプロイメント・ポリシーの XML ファイルが置かれているロケーションを指定します。objectGridType=EMBEDDED および **objectGridDeploymentXML** プロパティが指定されていない場合、eXtreme Scale ライブラリーにパッケージされた組み込み XML ファイルが自動的にロードされます。

replicationInterval

更新されたセッションを ObjectGrid に書き込む時間間隔を秒数で定義する整数値。デフォルトは 10 秒です。0 から 60 までの値を設定できます。0 の場合は、更新されたセッションを ObjectGrid に書き込むのは、各要求のサーブレット・サービス・メソッド呼び出しの最後ということになります。**replicationInterval** 値が高いほどパフォーマンスは向上します。これは、データ・グリッドに書き込まれるアップデートの数が少ないためです。ただし、値が高いとそれだけ構成のフォールト・トレラント性が低くなります。

この設定は、objectGridType が REMOTE に設定されている場合のみ適用されます。

reuseSessionID

true または false のいずれかのストリング値。デフォルトは false です。基礎になっている Web コンテナが異なるホストへの要求でセッション ID を再利用する場合は、true に設定します。このプロパティの値は、Web コンテナの中での値と同じでなければなりません。WebSphere Application Server を使用していて、管理コンソールまたは **wsadmin** ツール・スクリプトを使用して eXtreme Scale HTTP セッション・パーシスタンスを構成している場合、デフォルトで、Web コンテナ・カスタム・プロパティ `HttpSessionIdReuse=true` が追加されます。**reuseSessionID** も true に設定されます。セッション ID の再利用を望まない場合は、eXtreme Scale セッション・パーシスタンスを構成する前に、Web コンテナ・カスタム・プロパティに `HttpSessionIdReuse=false` カスタム・プロパティを設定します。

sessionIdOverrideClass

`com.ibm.websphere.xs.sessionmanager.SessionIDOverride` インターフェースを実装するクラスの名前。このクラスは、`HttpSession.getId()` メソッドで取得した固有のセッション ID をオーバーライドして、すべてのアプリケーションが同じ ID を持つようにするために使用されます。デフォルトでは、`HttpSession.getId()` から得られたユーザー ID が使用されます。

sessionStatsSpec = session.all = enabled

eXtreme Scale クライアント HTTP 統計仕様のストリング。

shareSessionsAcrossWebApps

true または false のいずれかのストリング値。デフォルトは false です。セッションが Web アプリケーション間で共有されるかどうかを指定します。true または false のいずれかのストリング値として指定されます。サーブレット仕様では、HTTP セッションを Web アプリケーション間で共有できないとされています。この共有を可能にするため、サーブレット仕様の拡張が提供されます。

sessionTableSize

メモリー内に保持するセッション参照数を定義する整数値。デフォルトは 1000 です。

EMBEDDED トポロジーは既に Web コンテナと同じ層にセッション・データを持っているため、この設定は REMOTE トポロジーのみに関するものです。

セッションは、最長未使用時間 (LRU) ロジックに基づいて、メモリー内のテーブルから除去されます。メモリー内のテーブルから除去されたセッションは Web コンテナから無効化されます。しかし、データ自体はグリッドから削除されないため、そのセッションに対する後続の要求は引き続きデータを検索することができます。この値は、Web コンテナの最大スレッド・プール値よりも高く設定されなければなりません。そうすると、セッション・キャッシュでの競合を削減できます。

securityEnabled

true または false のいずれかのストリング値。デフォルト値は false です。この設定により、eXtreme Scale クライアント・セキュリティーを使用可能にすることができます。この設定は、eXtreme Scale サーバー・プロパティ・ファイルの **securityEnabled** 設定と一致していなければなりません。これらの設定が一致しないと、例外が発生します。

sessionIdOverrideClass

取得されたアプリケーションのセッション ID をオーバーライドします。デフォルトでは、HttpSession.getId() メソッドから得られた ID が使用されます。eXtreme Scale クライアント HTTP セッションがアプリケーションの固有セッション ID をオーバーライドできるようにします。これにより、すべてのアプリケーションが同じ ID で検索されるようになります。com.ibm.websphere.xs.sessionmanager.SessionIDOverride インターフェースを実装するために設定します。このインターフェースは、HttpServletRequest オブジェクトに基づいて HttpSession ID を決定します。

traceSpec

ストリング値として設定される、IBM® WebSphere のトレース仕様を指定します。この設定は、WebSphere Application Server以外のアプリケーション・サーバーに使用してください。

traceFile

トレース・ファイルの場所をストリング値として指定します。この設定は、WebSphere Application Server以外のアプリケーション・サーバーに使用してください。

useURLEncoding

true または false のいずれかのストリング値。デフォルトは false です。URL 再書き込みを使用可能にする場合は、true に設定します。デフォルト値は false です。これは、セッション・データの保管に Cookie が使用されることを示します。このパラメーターの値は、セッション管理の Web コンテナ設定と同じでなければなりません。

useCookies

true または false のいずれかのストリング値。基礎になっている Web コンテナがさまざまなホストへの要求でセッション ID を再利用する場合は、true に設定します。デフォルトは false です。このプロパティの値は、Web コンテナで設定される値と同じでなければなりません。

親トピック: [セッション・データ・グリッドの作成](#)

splicer.properties ファイル

splicer.properties ファイルには、サブレット・フィルター・ベースのセッション・マネージャーを構成するための、すべての構成オプションが含まれます。

サンプル・スプライサー・プロパティ

このファイルに説明されている追加プロパティを使用する場合は、有効にするプロパティの行のコメントを外してください。

```
# サブレット・フィルター・ベース ObjectGrid
# セッション・マネージャーが使用するように構成できる、
# すべての構成オプションが含まれたプロパティ・ファイル。
#
# このプロパティ・ファイルで、これらの構成設定に割り当てるすべての
# デフォルト値を保持できます。また、このプロパティ・ファイルを
# filtersplicer ANT タスクと組み合わせて使用するには、ANT タスク・
# プロパティを使用して、個々の設定をオーバーライドできます。

# スtring値「REMOTE」または「EMBEDDED」。デフォルトは REMOTE です。
# If it is set to "REMOTE", the session data will be stored outside of
# the server on which the web application is running. If it is set to
# "EMBEDDED", an embedded WebSphere eXtreme Scale container will start
# in the application server process on which the web application is running.

objectGridType = REMOTE

# A string value that defines the name of the ObjectGrid
# instance used for a particular web application. The default name
# is session. This property must reflect the objectGridName in both
# the objectgrid xml and deployment xml files used to start the eXtreme
# Scale containers.

objectGridName = session

# カタログ・サーバーに接続して、クライアント・サイドの ObjectGrid
# インスタンスを取得できます。The value needs to be of the
# form "host:port<,host:port>", where the host is the listener host
# on which the catalog server is running, and the port is the listener
# port for that catalog server process.
# このリストは任意の長さにするのができ、ブートストラッピングにのみ使用
# されます。最初の実行可能なアドレスが使用されます。It is optional inside WebSphere
# if the catalog.services.cluster property is configured.

# catalogHostPort = host:port<,host:port>

# An integer value (in seconds) that defines the time in seconds between
# writing of updated sessions to ObjectGrid. The default is 10. This property
# is only used when objectGridType is set to REMOTE. Possible values are
# from 0 to 60. 0 means updated sessions are written to the ObjectGrid
# at the end of servlet service method call for each request.

replicationInterval = 10

# メモリー内に保持するセッション参照数を定義する整数値。
# デフォルトは 1000 です。This property is only used when
# objectGridType is set to REMOTE. When the number of sessions stored
# in memory in the web container exceeds this value, the least recently
# accessed session is invalidated from the web container. If a request
# comes in for that session after it's been invalidated, a new session
# will be created (with a new session ID if reuseSessionId=false),
# populated with the invalidated session's attributes. This value should
# always be set to be higher than the maximum size of the web container
# thread pool to avoid contention on this session cache.

sessionTableSize = 1000

# A string value of either "true" or "false", default is "true".
# It is to control whether we store session data as a whole entry
```



```
# or store each attribute separately.
# This property was referred to as persistenceMechanism in the
# previous filter-based implementation, with the possible values
# of ObjectGridStore (fragmented) and ObjectGridAtomicSessionStore
# (not fragmented).

fragmentedSession = true

# A string value of either "true" or "false", default is "false".
# Enables eXtreme Scale client security. This setting needs to match
# the securityEnabled setting in the eXtreme Scale server properties
# file. これらの設定が一致しないと、例外が発生します。

securityEnabled = false

# Specifies the client credential authentication support.
# The possible values are:
# Never - The client does not support credential authentication.
# Supported* - The client supports the credential authentication if and only if the
server
# supports too.
# Required - The client requires the credential authentication.
# The default value is Supported.

# credentialAuthentication =

# Specifies the retry count for authentication if the credential
# is expired. If the value is set to 0, there will not be
# any authentication retry.

# authenticationRetryCount =

# Specifies the name of the class that implements the
# com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator
# interface. This class is used to get credentials for clients.

# credentialGeneratorClass =

# Specifies the properties for the CredentialGenerator implementation
# class. The properties are set to the object with the setProperties(String)
# method. The credentialGeneratorProps value is used only if the value of the
# credentialGeneratorClass property is not null.

# credentialGeneratorProps =

# The file location of the objectgrid xml file.
# The built-in xml file packaged in the eXtreme Scale library
# will automatically be loaded if this property
# is not specified and if objectGridType=EMBEDDED

# objectGridXML =

# The file location of the objectGrid deployment policy xml file.
# The built-in xml file packaged in the eXtreme Scale library
# will automatically be loaded if this property
# is not specified and if objectGridType=EMBEDDED

# objectGridDeploymentXML =

# IBM WebSphere トレース仕様を示すストリング。
# useful for all other application servers besides WebSphere.

# traceSpec =

# トレース・ファイルの場所を示すストリング。
# useful for all other application servers besides WebSphere.

# traceFile=
```

```
# This property should be set if you require sessions to be
# accessible across hosts. The value will be the name of the
# common domain between the hosts.

# cookieDomain=

# This property should be set to the same path you have configured
# for your application server cookie settings. The default path
# is /.

# cookiePath

# Set to true if the underlying web container will reuse
# session ID's across requests to different hosts. デフォルト
# is false. The value of this should be the same as what is
# set in the web container.

# reuseSessionId=

# A string value of either "true" or "false", the default is
# "false". Per the servlet specification, HTTP Sessions cannot
# be shared across web applications. An extension to the servlet
# specification is provided to allow this sharing.

# shareSessionsAcrossWebApps = false

# A string value of either "true" or "false", default is "false".
# Set to true if you want to enable urlRewriting. Default is
# false. The value of this should reflect what is set in the
# web container settings for session management.

# useURLEncoding = false

# Set to false if you want to disable cookies as the session tracking
# mechanism. デフォルトは true です。The value of this should reflect what
# is set in the web container settings for session management.

# useCookies = true

# A string value of either "true" or "false", the default is "false".
# Enables eXtreme Scale client HTTP Sessions statistics tracking.

# enableSessionStats = false

# Overrides the retrieved session ID of an application.
# The default is to use the ID derived from the HttpSession.getId() method.
# Enables eXtreme Scale client HTTP Sessions to override
# the unique session ID of an application so that all applications are retrieved
# with the same ID.
# Set to the implementation of the
# com.ibm.websphere.xs.sessionmanager.SessionIDOverride interface.
# This interface determines the HttpSession ID based on
# the HttpServletRequest object.

# sessionIdOverrideClass = # A string of eXtreme Scale client HTTP statistics
specification.

# sessionStatsSpec = session.all = enabled

# Set to true if your environment contains multiple applications that
# use unique cookie names. Default is false, which assumes all applications
# are using the same cookie name.

# applicationQualifiedCookies=false
```

親トピック: [セッション・データ・グリッドの作成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連タスク:

[WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)

[WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)

[データ・グリッド内における セッション管理用のカタログ・サービス・ドメインの作成](#)

[以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンス の構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パーシスタンスの構成](#)

[HTTP コマンド・インターフェースでの管理](#)

[各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成](#)

関連情報:

☞ [コンソールを使用したエンタープライズ・アプリケーション・ファイルのインストール](#)

☞ [wsadmin スクリプト](#)

動的キャッシュの作成データ・グリッド

IBM® WebSphere® DataPower® XC10 アプライアンス を使用して、WebSphere Application Server 動的キャッシュ・インスタンスのデータを保管することができます。この機能をセットアップすることにより、WebSphere Application Server 動的キャッシュ・インスタンスを使用するよう作成されたアプリケーションでアプライアンスのフィーチャーおよびパフォーマンス機能を活用できるようにします。

始める前に

- WebSphere eXtreme Scale クライアントを WebSphere Application Server 構成にインストールする必要があります。詳しくは、[WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- アプライアンスの Transport Layer Security (TLS) が有効であるか、クライアントが必ず Transport Layer Security (TLS) を使用するようにしたい場合、WebSphere Application Server 管理コンソールでグローバル・セキュリティーを有効にする必要もあります。詳しくは、[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。

このタスクについて

WebSphere Application Server 内の動的キャッシュ・プロバイダーを構成して DataPower XC10 アプライアンス を使用する場合、キャッシュ・データは WebSphere Application Server のトポロジーの外に保管されます。キャッシュ・データはすべてアプライアンス内に保管されます。ご使用のアプリケーション・サーバー内でキャッシングに使用される メモリーを他の目的のためにも使用できます。

IBM WebSphere Commerce での WebSphere eXtreme Scale クライアント および DataPower XC10 アプライアンス 動的キャッシュ・データ・グリッドの使用については、IBM WebSphere Commerce 資料の以下のトピックを参照してください。

- [Enabling the dynamic cache service and servlet caching](#)
- [Enabling WebSphere Commerce data cache](#)

手順

1. DataPower XC10 アプライアンスで構成するキャッシュ・インスタンスを作成します。詳しくは、[動的キャッシュ・インスタンスの構成](#)を参照してください。
2. カタログ・サービスを構成します。カタログ・サービスにより、WebSphere Application Server 動的キャッシュ構成で DataPower XC10 アプライアンスと通信できます。カタログ・サービス・ドメインを作成することで、WebSphere Application Server 管理コンソールでカタログ・サービスを構成できます。詳しくは、[WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)を参照してください。
3. DataPower XC10 アプライアンス上にデータ・グリッドを作成して、必要なセキュリティー設定を構成します。dynaCfgToAppliance スクリプトを実行することも、構成を手動で作成することもできます。

dynaCfgToAppliance スクリプトを使用したデータ・グリッド構成の作成:

このスクリプトは、WebSphere eXtreme Scale クライアントのインストール時にデプロイメント・マネージャー・プロファイルの bin ディレクトリーにインストールされます。このスクリプトを実行する前に、アプライアンスとデプロイメント・マネージャーの両方が稼働中であることを確認してください。

```
dynaCfgToAppliance <IP_address> <cache_jndi_name> <appliance_admin> <admin_pw>  
<SOAP_port> <soap.client.props>
```

IP_address

動的キャッシュ・データを保管する DataPower XC10 アプライアンスの IP アドレスを指定します。

cache_jndi_name

動的キャッシュの JNDI 名を指定します。動的キャッシュの JNDI 名の中にスラッシュ (/) があると、DataPower XC10 アプライアンスのデータ・グリッドの名前では、そのスラッシュはダッシュに変換されます。例えば、動的キャッシュ名が services/cache1 であれば、アプライアンスで作成されたデータ・グリッドは、services-cache1 という名前になります。DataPower XC10 アプライアンスでは、データ・グリッドの名前の中に ^ . ¥¥ / , # \$ @ : ; ¥ * ? < > | = + & % [] " " 文字は使用できません。

appliance_admin

DataPower XC10 アプライアンス のユーザー・インターフェース にログインするための管理者 ID を指定します。

admin_pw

DataPower XC10 アプライアンス のユーザー・インターフェースにログインするための管理者パスワードを指定します。

SOAP_port

(オプション) デフォルトの 8879 以外のポート番号を使用する場合は、デプロイメント・マネージャーの SOAP ポートを指定します。

soap.client.props

(オプション) soap.client.props ファイルのパスを指定します。 WebSphere Application Server でセキュリティを有効にした場合は、このファイルを指定する必要があります。このファイルは SOAP セキュリティを有効にし、WebSphere Application Server デプロイメント・マネージャーを管理するためのユーザー名とパスワードを指定します。

```
com.ibm.SOAP.securityEnabled=true
com.ibm.SOAP.loginUserId=
com.ibm.SOAP.loginPassword=
```

soap.client.props ファイルについて詳しくは、[SOAP コネクタおよびプロセス間通信コネクタのプロパティ・ファイル \(SOAP connector and Inter-Process Communications connector properties files\)](#) を参照してください。

このスクリプトは、アプライアンス上にデータ・グリッドを作成します。また、スクリプトは、**appliance_admin** パラメーターおよび **admin_pw** パラメーターで指定した DataPower XC10 アプライアンス固有の ID およびパスワードの資格情報を以下のカスタム・プロパティを使用して設定します。

- xc10.<data_grid_name>.userid
- xc10.<data_grid_name>.password

これらのプロパティ名はケース・センシティブではありません。パスワードの値はエンコードされます。初期構成後にスクリプトを再実行すると、カスタム・プロパティが更新されます。

データ・グリッド構成の手動作成:

- DataPower XC10 アプライアンスのユーザー・インターフェースで動的キャッシュ・データ・グリッドを作成します。「データ・グリッド」 > 「動的キャッシュ」をクリックします。キャッシュの名前は、WebSphere Application Server 構成の動的キャッシュの JNDI 名と一致している必要があります。JNDI 名を入力する際、DataPower XC10 アプライアンスのデータ・グリッドの名前で、スラッシュはすべて (/) ダッシュに置換します。例えば、動的キャッシュ名が services/cache1 であれば、名前を services-cache1 として入力します。
 - WebSphere Application Server セルで、xc10.<data_grid_name>.userid および xc10.<data_grid_name>.password の各カスタム・プロパティを作成します。各カスタム・プロパティの <data_grid_name> の値は、スラッシュ (/) がダッシュと置き換えられた、データ・グリッドの JNDI 名です。例えば、前の例で、カスタム・プロパティ名は xc10.services-cache1.userid と xc10.services-cache1.password です。値は、DataPower XC10 アプライアンス構成内のデータ・グリッドにアクセスできるユーザー ID とパスワードに設定する必要があります。encodePassword スクリプト (デプロイメント・マネージャーの bin ディレクトリーにあります) を使用して、パスワードをエンコードできます。
4. **2.5+** オプション: DataPower XC10 アプライアンス 動的キャッシュ・プロバイダーでマルチマスター複製を使用可能に設定する選択ができます。詳しくは、[集合間のマルチマスター複製の構成](#) を参照してください。

注: WebSphere Portal Server または WebSphere Commerce Server の動的キャッシュ・グリッド・ユーザーは、複数のキャッシュ・インスタンスをそれぞれの WebSphere Application Server 構成内に定義している可能性があります。DataPower XC10 アプライアンスでマルチマスター複製を使用可能に設定すると、この構成は、動的キャッシュ・プロバイダーとして DataPower XC10 アプライアンスを使用するように定義されているキャッシュ・インスタンスのみに影響し、デフォルトの WebSphere Application Server 動的キャッシュ・プロバイダーを使用するように定義されているキャッシュ・インスタンスに影響しません。

タスクの結果

動的キャッシュ・サービスでアプライアンスをキャッシュ・プロバイダーとして使用するように構成することで、アプリケーション・サーバーが使用するメモリーが減少します。すべてのキャッシュ・データがアプライアンスにオフロードされ、アプリケーション・サーバーのメモリーにはもう保管されません。

次のタスク

- データ・グリッドへのデータの送信を開始する前に、セキュリティを構成します。詳しくは、[データ・グリッドのセキュリティを使用可能に設定する](#) を参照してください。
- レプリカを構成します。レプリカにより、基本コピーで障害が発生した場合にもデータ・グリッドのデータが使用可能になります。レプリカを構成するには、「データ・グリッド」 > 「動的キャッシュ」 > 「拡張属性の表示」をクリックします。アプライアンスが集合内にある場合のみ、レプリカが作成されます。集合内のアプライアンスの数が n の場合、レプリカの最大数は $n-1$ になります。したがって、3つのレプリカを構成するが、集合内にあるアプライアンスが2つのみである場合、作成されるレプリカは1つのみです。集合にアプライアンスを追加すると、追加のレプリカが作成されます。アプライアンスが集合に追加されたときに新規レプリカを作成できるようにするため、レプリカの数を適切な数に設定してください。レプリカの数を編集すると、データ・グリッドの内容が消去されます。
- 動的キャッシュ・インスタンスのグリッド容量限度は、WebSphere Application Server 動的キャッシュ・インスタンスの作成時に構成されます。このサイズを超えた場合は、項目数を構成された限度内に保つために最長未使用時間 (LRU) 除去ストラテジーが使用されます。
- DataPower XC10 アプライアンス ユーザー・インターフェース 内の動的キャッシュデータ・グリッド (data

grid)はモニターできます。詳しくは、[ユーザー・インターフェースでのデータ・グリッドのモニター](#)を参照してください。

Java 2.5+ [動的キャッシュ・インスタンスの構成](#)

WebSphere Application Server の動的キャッシュ・サービスは、デフォルト・キャッシュ・インスタンス (baseCache) と追加のサブレット・キャッシュ・インスタンスおよびオブジェクト・キャッシュ・インスタンスの両方の作成をサポートします。

Java 2.5+ [動的キャッシュ・プロバイダーの概要](#)

WebSphere Application Server は、デプロイされた Java EE アプリケーションが使用できる動的キャッシュ・サービスを提供します。このサービスは、サブレット、JSP、またはコマンドからの出力などのデータ、および DistributedMap API を使用してエンタープライズ・アプリケーション内にプログラマチックに指定されているオブジェクト・データをキャッシュするために使用されます。この機能をセットアップすることにより、動的キャッシュ・サービスを使用するアプリケーションが WebSphere DataPower XC10 アプライアンス のフィーチャーおよびパフォーマンス機能を使用できるようになります。

Java [WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)

WebSphere DataPower XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

Java 2.5+ [動的キャッシュのニア・キャッシュの構成](#)

アプライアンスまたは集合で、動的キャッシュ・データ・グリッドと一緒に動作するローカルのニア・キャッシュを構成することができます。ニア・キャッシュは、ローカル Java 仮想マシン (JVM) リソースを使用します。一般に、ニア・キャッシュには、アプライアンスの動的キャッシュ・データ・グリッドにあるデータのサブセットが保持されます。

親トピック: [データ・グリッドの構成](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

動的キャッシュ・インスタンスの構成

WebSphere® Application Server の動的キャッシュ・サービスは、デフォルト・キャッシュ・インスタンス (baseCache) と追加のサーブレット・キャッシュ・インスタンスおよびオブジェクト・キャッシュ・インスタンスの両方の作成をサポートします。

このタスクについて

デフォルト・キャッシュ・インスタンス (baseCache) は、当初は WebSphere Application Server によってサポートされる唯一の動的キャッシュ・インスタンスでしたが、現在は、WebSphere Commerce Suite によって使用される、すぐに使用可能な動的キャッシュ・インスタンスです。追加のサーブレット・キャッシュ・インスタンスおよびオブジェクト・キャッシュ・インスタンスは、WebSphere Application Server の後継リリースで追加されたもので、WebSphere Application Server 管理コンソールの独立した「キャッシュ・インスタンス」セクションで構成されます。

Java 2.5+ [デフォルト動的キャッシュ・インスタンス \(baseCache\) の構成](#)

デフォルト動的キャッシュ・インスタンス (baseCache と呼ばれる) は、WebSphere Application Server 動的キャッシュ・サービスによって作成された動的キャッシュ・インスタンスのデフォルトです。このサーブレット動的キャッシュ・インスタンスは、IBM WebSphere Commerce などの製品によって使用されます。baseCache は、WebSphere Application Server で定義される他のキャッシュ・インスタンスとは異なり、ただ 1 つのサーバー・インスタンスまたはクラスター・インスタンスに固有のものです。下記の手順を使用して、WebSphere eXtreme Scale と一緒に動的キャッシュ・プロバイダーとして使用するよう WebSphere Application Server で baseCache インスタンスを構成します。

Java 2.5+ [オブジェクトまたはサーブレットの動的キャッシュ・インスタンスの構成](#)

WebSphere Application Server では、デフォルト・インスタンスのほかに、複数のオブジェクト動的キャッシュ・インスタンスやサーブレット動的キャッシュ・インスタンスを構成することができます。下記の手順を使用して、追加のオブジェクト・キャッシュ・インスタンスやサーブレット・キャッシュ・インスタンスを構成します。

Java 2.5+ [カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)

WebSphere Application Server により、動的キャッシュ・インスタンスに対してカスタム・プロパティを設定することができます。

親トピック: **Java** [動的キャッシュの作成データ・グリッド](#)

デフォルト動的キャッシュ・インスタンス (baseCache) の構成

デフォルト動的キャッシュ・インスタンス (baseCache と呼ばれる) は、WebSphere® Application Server 動的キャッシュ・サービスによって作成された動的キャッシュ・インスタンスのデフォルトです。このサーブレット動的キャッシュ・インスタンスは、IBM WebSphere Commerce などの製品によって使用されます。baseCache は、WebSphere Application Server で定義される他のキャッシュ・インスタンスとは異なり、ただ 1 つのサーバー・インスタンスまたはクラスター・インスタンスに固有のものです。下記の手順を使用して、WebSphere eXtreme Scale と一緒に動的キャッシュ・プロバイダーとして使用するよう WebSphere Application Server で baseCache インスタンスを構成します。

始める前に

- 動的キャッシュ・プロバイダーを使用するには、WebSphere Application Server ノードの各デプロイメント (デプロイメント・マネージャー・ノードを含む) 上に WebSphere eXtreme Scale クライアント をインストールしておく必要があります。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- WebSphere eXtreme Scale カタログ・サービス・ドメインが構成されていなければなりません。詳しくは、[WebSphere Application Server でのカタログ・サービス・ドメインの作成](#) を参照してください。
- カタログ・サービス・ドメイン内のカタログ・サーバーで Secure Sockets Layer (SSL) が使用可能になっている場合、または SSL がサポートされたカタログ・サービス・ドメインで SSL を使用する場合は、WebSphere Application Server 管理コンソールでグローバル・セキュリティーを有効にする必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#) を参照してください。

このタスクについて

この手順に含まれるステップは、WebSphere Application Server 管理コンソールのバージョン 8.0 に対応します。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注:

- WebSphere eXtreme Scale バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。
- 下記の手順は、リモート WebSphere eXtreme Scale 動的キャッシュ・トポロジーに固有のものです。現在、WebSphere eXtreme Scale バージョン 8.6 では、その他のトポロジー (組み込み、組み込み区画化、ローカルなど) はすべて非推奨です。
- 次の手順では、WebSphere Application Server V7.0 Fix Pack 27、V8.0 Fix Pack 6、V8.5 Fix Pack 2 以上を使用していることが前提になっています。これらのバージョンには WebSphere Application Server APAR PM71992 が含まれています。

手順

- WebSphere Application Server 管理コンソールを開始します。
- トップ・メニューで、「サーバー」>「サーバー・タイプ」>「**WebSphere** アプリケーション・サーバー」をクリックします。
- 「アプリケーション・サーバー」領域で、「**your server name**」を選択します。
- 「構成」パネルで、「コンテナー・サービス」をクリックし、「動的キャッシュ・サービス」を選択します。
- 「**Cache provider**」ドロップダウン・リストで「WebSphere eXtreme Scale」を選択します。
- キャッシュ・サイズを変更したい場合は、そのキャッシュ・サイズを「キャッシュ・サイズ」ボックスで指定します。キャッシュ・サイズ値は、この動的キャッシュ・インスタンス用の WebSphere eXtreme Scale グリッド内の各区画で許容されるエントリーの最大数を指定します。デフォルトは 1 区画内 2000 エントリーです。
- 「キャッシュ複製を使用可能にする (**Enable cache replication**)」を選択します。このチェック・ボックスを使用可能にすると、キャッシュ・データはローカルではなくリモート側でデータ・グリッドに保管されます。WebSphere eXtreme Scale をキャッシュ・プロバイダーとして使用するときは、このオプションを選択する必要があります。
- 「適用」または「**OK**」をクリックし、構成を保存します。
- トップ・メニューで、「サーバー」>「サーバー・タイプ」>「**WebSphere** アプリケーション・サーバー」をクリックします。
- 「アプリケーション・サーバー」領域で、「**your server name**」を選択します。
- 「構成」パネルで、「**Web** コンテナー設定」をクリックし、「**Web** コンテナー」を選択します。
- 「サーブレット・キャッシングを有効にする」チェック・ボックスを選択します。
- 「適用」または「**OK**」をクリックし、構成を保存します。

次のタスク

デフォルトでは、WebSphere Application Server で構成されている各動的キャッシュ・インスタンスは、キャッシュ・インスタンスの JNDI 名と同じ名前を持つ動的キャッシュ・データ・グリッドに対応します。またデフォルトでは、そのキャッシュ・インスタンスのデータはその動的キャッシュ・データ・グリッド内の動的マップに保管され、その動的マップ名の接尾部もキャッシュ・インスタンスの JNDI 名に対応します。例えば、cache1 という JNDI 名を持つキャッシュ・インスタンスを WebSphere Application Server で構成すると、cache1 という名前を持つ動的キャッシュ・データ・グリッドがアプライアンスに作成されます。データ・グリッド cache1 の内部では、データを保管するために、IBM_DC_PARTITIONED_cache1 という動的マップが作成されます。

ほとんどの場合、この構成を変更する必要はありません。ただし、状況によっては、異なる JNDI 名を持つ複数のキャッシュ・インスタンスが同じデータ・グリッド・インスタンス内の異なる動的マップにマップされるようにしたいことがあります。また状況が違えば、同じ JNDI 名を持つ複数のキャッシュ・インスタンスが同じ動的キャッシュ・データ・グリッド内の異なる動的キャッシュ・データ・グリッド・インスタンスまたは異なる動的マップ・インスタンスにマップされるようにしたいことがあります。例えば、デフォルトの動的キャッシュ・インスタンス (baseCache) を使用するアプリケーションがある場合、キャッシュ・データを別々のデータ・グリッドに保存するか、同じデータ・グリッド内の別々の動的マップに保存して、テスト・レベル環境と実動レベル環境の両方に対して同じアプライアンスを使用する必要がある場合があります。

この構成を制御するために、キャッシュ・インスタンスで以下のカスタム・プロパティを設定することができます。

ヒント: これらのプロパティは、デフォルトの動的キャッシュ・インスタンス (baseCache) を使用するとき特に役立ちます。これは、キャッシュの JNDI 名は自動的に baseCache に設定され、変更できないからです。

com.ibm.websphere.xs.dynacache.grid_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッド・インスタンス (これに動的キャッシュ・インスタンスが対応します) の名前を指定します。

com.ibm.websphere.xs.dynacache.cache_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッドとそのデータ・グリッド内の動的マップの両方に対して (JNDI 名の代わりに) 使用する名前を指定します。

com.ibm.websphere.xs.dynacache.grid_name カスタム・プロパティも設定された場合は、このプロパティの値は動的マップ名にのみ適用されます。

これらのプロパティの設定方法については、[カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)を参照してください。

親トピック:  **2.5+** [動的キャッシュ・インスタンスの構成](#)

オブジェクトまたはサーブレットの動的キャッシュ・インスタンスの構成

WebSphere® Application Server では、デフォルト・インスタンスのほかに、複数のオブジェクト動的キャッシュ・インスタンスやサーブレット動的キャッシュ・インスタンスを構成することができます。下記の手順を使用して、追加のオブジェクト・キャッシュ・インスタンスやサーブレット・キャッシュ・インスタンスを構成します。

始める前に

- 動的キャッシュ・プロバイダーを使用するには、WebSphere Application Server ノードの各デプロイメント (デプロイメント・マネージャー・ノードを含む) 上に WebSphere eXtreme Scale をインストールしておく必要があります。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#) を参照してください。
- WebSphere eXtreme Scale カタログ・サービス・ドメインが構成されていなければなりません。 [WebSphere Application Server でのカタログ・サービス・ドメインの作成](#) を参照してください。
- カタログ・サービス・ドメイン内のカタログ・サーバーで Secure Sockets Layer (SSL) が使用可能になっている場合、または SSL がサポートされたカタログ・サービス・ドメインで SSL を使用する場合は、WebSphere Application Server 管理コンソールでグローバル・セキュリティーを有効にする必要があります。グローバル・セキュリティーの構成に関して詳しくは、[グローバル・セキュリティーの設定](#) を参照してください。

このタスクについて

この手順で作成できるキャッシュ・インスタンスには、オブジェクト・キャッシュ・インスタンスとサーブレット・キャッシュ・インスタンスという 2 つのタイプがあります。オブジェクト・キャッシュ・インスタンスは、デフォルトの共有動的キャッシュであることに加えて、Java 2 Platform, Enterprise Edition (J2EE) アプリケーションがオブジェクトを保管、配布、および共有できるロケーションでもあります。オブジェクト・キャッシュ・インスタンスを構成した後、com.ibm.websphere.cache パッケージ内の DistributedMap または DistributedObjectCache インターフェースを使用して、プログラマチックにキャッシュ・インスタンスにアクセスすることができます。DistributedMap または DistributedObjectCache インターフェースについて詳しくは、[追加のアプリケーション・プログラミング・インターフェース \(API\)](#) を参照してください。サーブレット・キャッシュ・インスタンスは、デフォルト動的キャッシュのほかに、動的キャッシュが呼び出されたサーブレットの出力および副次作用を保管、配布、および共有できるロケーションでもあります。サーブレット・キャッシュ・インスタンスを構成することにより、アプリケーションは柔軟性が増し、キャッシュ・リソースを調整しやすくなります。管理コンソールでキャッシュ・インスタンスに対して指定される Java Naming and Directory Interface (JNDI) 名が、cachespec.xml 構成ファイル内のキャッシュ・インスタンス・エレメントにマップされます。 <cache-instance> エレメント内で指定される <cache-entry> エレメントは、そのキャッシュ・インスタンスに作成されます。 <cache-instance> エレメント外で指定される <cache-entry> エレメントは、デフォルト動的キャッシュ・インスタンスに格納されます。オブジェクト型およびサーブレット型のキャッシュ・インスタンスについて詳しくは、[キャッシュ・インスタンス](#) を参照してください。

この手順に含まれるステップは、WebSphere Application Server 管理コンソールのバージョン 8.0 に対応します。この情報は、ご使用の WebSphere Application Server バージョンによって、少し違う可能性があります。

注:

- WebSphere eXtreme Scale バージョン 8.6 は、バージョン 7.0 より前の WebSphere Application Server のバージョンではサポートされていません。

手順

- WebSphere Application Server 管理コンソールからオブジェクト・キャッシュまたはサーブレット・キャッシュを構成するには、以下のステップを使用します。
 - WebSphere Application Server 管理コンソールを開始します。
 - トップ・メニューで、「リソース」>「キャッシュ・インスタンス」>「オブジェクト・キャッシュ・インスタンス」をクリックします。
 - 「オブジェクト・キャッシュ・インスタンス」領域で、作成したいキャッシュ・インスタンスのタイプを選択します。オブジェクト・キャッシュ・インスタンスまたはサーブレット・キャッシュ・インスタンスのいずれかを選択することができます。
 - キャッシュ・インスタンスの有効範囲を指定します。セルの有効範囲を指定して、セル内のすべてのサーバーがこのキャッシュ・インスタンスを使用できるようにします。ノードの有効範囲を指定すると、キャッシュ・インスタンスはノードのすべてのサーバーで使用可能になります。サーバーの有効範囲を指定すると、キャッシュ・インスタンスは選択したサーバーでのみ使用可能になります。必要に応じて、有効範囲を混用することができます。
 - 「適用」をクリックして範囲を保管します。
 - 「新規」をクリックし、オブジェクト・キャッシュ・インスタンスを定義します。
 - 「Cache provider」ドロップダウン・リストで「WebSphere eXtreme Scale」を選択します。

注: WebSphere eXtreme Scale が動的キャッシュ・プロバイダーとして表示されなかった場合は、WebSphere Application Server プロファイルが WebSphere eXtreme Scale 用に拡張されていま

せん。

8. 動的キャッシュ・インスタンスの JNDI 名を指定します。オブジェクト・キャッシュの場合、この名前はキャッシュの検索時に使用されます。サーブレット・キャッシュの場合、これは `cachespec.xml` ファイル内の `<cache-instance>` エレメントで指定される `name` 属性です。
 9. 動的キャッシュ・オブジェクトの JNDI 名を指定します。
 10. キャッシュ・サイズを変更したい場合は、そのキャッシュ・サイズを「キャッシュ・サイズ」ボックスで指定します。キャッシュ・サイズ値は、この動的キャッシュ・インスタンス用の WebSphere eXtreme Scale グリッド内の各区画で許容されるエントリーの最大数を指定します。デフォルトは 1 区画内 2000 エントリーです。
 11. 「**キャッシュ複製を使用可能にする (Enable cache replication)**」ボックスを選択します。このチェック・ボックスを使用可能にすると、キャッシュ・データはローカルではなくリモート側でグリッドに保管されます。WebSphere eXtreme Scale をキャッシュ・プロバイダーとして使用しているときは、このボックスを選択する必要があります。
 12. 「**適用**」または「**OK**」をクリックし、構成を保存します。
- `cacheinstances.properties` ファイルを使用してオブジェクト・キャッシュまたはサーブレット・キャッシュを構成するには、以下のステップを使用します。
 1. `cacheinstances.properties` ファイルを作成します。必要とされるコンテンツについては、[キャッシュ・インスタンス・プロパティ・ファイル](#) を参照してください。
 2. `cacheinstances.properties` ファイルを、アプリケーション・サーバーかアプリケーション・クラスパスのいずれかに配置します。例えば、アプリケーション WAR (Web アプリケーション・アーカイブ) ファイル (`WEB-INF\classes` ディレクトリー) を使用することもできれば、`server_root\classes` ディレクトリーを作成して、そこにファイルを置くこともできます。

次のタスク

デフォルトでは、WebSphere Application Server で構成されている各動的キャッシュ・インスタンスは、キャッシュ・インスタンスの JNDI 名と同じ名前を持つ動的キャッシュ・データ・グリッドに対応します。またデフォルトでは、そのキャッシュ・インスタンスのデータはその動的キャッシュ・データ・グリッド内の動的マップに保管され、その動的マップ名の接尾部もキャッシュ・インスタンスの JNDI 名に対応します。例えば、`cache1` という JNDI 名を持つキャッシュ・インスタンスを WebSphere Application Server で構成すると、`cache1` という名前を持つ動的キャッシュ・データ・グリッドがアプライアンスに作成されます。データ・グリッド `cache1` の内部では、データを保管するために、`IBM_DC_PARTITIONED_cache1` という動的マップが作成されます。

ほとんどの場合、この構成を変更する必要はありません。ただし、状況によっては、異なる JNDI 名を持つ複数のキャッシュ・インスタンスが同じデータ・グリッド・インスタンス内の異なる動的マップにマップされるようにしたいことがあります。また状況が違えば、同じ JNDI 名を持つ複数のキャッシュ・インスタンスが同じ動的キャッシュ・データ・グリッド内の異なる動的キャッシュ・データ・グリッド・インスタンスまたは異なる動的マップ・インスタンスにマップされるようにしたいことがあります。例えば、デフォルトの動的キャッシュ・インスタンス (`baseCache`) を使用するアプリケーションがある場合、キャッシュ・データを別々のデータ・グリッドに保存するか、同じデータ・グリッド内の別々の動的マップに保存して、テスト・レベル環境と実動レベル環境の両方に対して同じアプライアンスを使用する必要がある場合があります。

この構成を制御するために、キャッシュ・インスタンスで以下のカスタム・プロパティを設定することができます。

ヒント: これらのプロパティは、デフォルトの動的キャッシュ・インスタンス (`baseCache`) を使用するとき特に役立ちます。これは、キャッシュの JNDI 名は自動的に `baseCache` に設定され、変更できないからです。

com.ibm.websphere.xs.dynacache.grid_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッド・インスタンス (これに動的キャッシュ・インスタンスが対応します) の名前を指定します。

com.ibm.websphere.xs.dynacache.cache_name

このカスタム・プロパティを使用して、動的キャッシュ・データ・グリッドとそのデータ・グリッド内の動的マップの両方に対して (JNDI 名の代わりに) 使用する名前を指定します。

com.ibm.websphere.xs.dynacache.grid_name カスタム・プロパティも設定された場合は、このプロパティの値は動的マップ名にのみ適用されます。

これらのプロパティの設定方法については、[カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#) を参照してください。

親トピック:  [動的キャッシュ・インスタンスの構成](#)

カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ

WebSphere® Application Server により、動的キャッシュ・インスタンスに対してカスタム・プロパティを設定することができます。

始める前に

デフォルト・キャッシュ・インスタンスまたは追加のオブジェクト型あるいはサーブレット型のキャッシュ・インスタンスのいずれかが既に構成されていなければなりません。 [デフォルト動的キャッシュ・インスタンス \(baseCache\) の構成](#) または [オブジェクトまたはサーブレットの動的キャッシュ・インスタンスの構成](#) を参照してください。

このタスクについて

次のいずれかの方法で、特定の動的キャッシュ・インスタンスに固有のカスタム・プロパティを設定することができます。

- 特定の動的キャッシュ・インスタンスに固有のカスタム・プロパティを設定したい場合は、APAR PM71992 が適用された WebSphere Application Server 管理コンソールを使用します。このフィックスをお持ちでない場合は、WebSphere Application Server サポート・ページ (<http://www.ibm.com/software/webservers/appserv/was/support>) にお問い合わせください。
- cacheinstances.properties ファイルが作成されている場合は、そのファイル内のカスタム・プロパティを設定することができます。この方法は、デフォルト動的キャッシュ (baseCache) インスタンスに対するカスタム・プロパティの設定には使用できません。
- WebSphere Application Server 管理コンソールを使用して、Java 仮想マシン (JVM) のカスタム・プロパティの値を変更することができます。

注: 設定した JVM プロパティは特定の JVM 内で使用されたすべてのキャッシュ・インスタンスに影響を及ぼすことがあります。

注: JVM プロパティの有効範囲は WebSphere Application Server JVM 内のすべてのキャッシュ・インスタンスまで及ぶことがあります。したがって、ほとんどの状況では、WebSphere Application Server 管理コンソール (デフォルト・キャッシュ・インスタンスの場合は APAR PM71992 を適用) でのキャッシュ固有カスタム・プロパティの使用または cacheinstances.properties ファイルの使用をお勧めします。

手順

- WebSphere Application Server 管理コンソールでキャッシュ・インスタンスに対してカスタム・プロパティを設定するには、以下のステップを使用します。

- WebSphere Application Server 管理コンソールを開始します。

注: WebSphere Application Server APAR PM71992 を適用するまでは、デフォルト (baseCache) インスタンスに対してこれらのステップを使用することはできません。このフィックスは WebSphere Application Server バージョン 7.0.0.27、8.0.0.6、および 8.5.0.2 以上で使用可能です。このフィックスをお持ちでない場合は、WebSphere Application Server サポート・ページ (<http://www.ibm.com/software/webservers/appserv/was/support>) にお問い合わせください。

- 既に構成済みである目的のキャッシュ・インスタンスに移動します。
- そのキャッシュ・インスタンス・パネルから、「追加プロパティ」>「カスタム・プロパティ」をクリックします。
- 「新規」を選択し、カスタム・プロパティの名前と値を指定します。
- 「適用」または「OK」をクリックし、構成を保存します。
- Deployment Manager とすべてのアプリケーション・サーバー・プロセスを再始動します。

- cacheinstances.properties ファイルを使用してキャッシュ・インスタンスのカスタム・プロパティを設定するには、以下のステップを使用します。

注: デフォルト (baseCache) インスタンスに対してこれらのステップを使用することはできません。

- カスタム・プロパティを cacheinstances.properties ファイルに追加します。このファイルを作成する必要がある場合は、[キャッシュ・インスタンス・プロパティ・ファイル](#) を参照して必要な内容を確認してください。
- cacheinstances.properties ファイルを、アプリケーション・サーバーかアプリケーション・クラスパスのいずれかに配置します。例えば、アプリケーション WAR (Web アプリケーション・アーカイブ) ファイル (WEB-INF\classes ディレクトリ) を使用することもできれば、server_root\classes ディレクトリを作成して、そこにファイルを置くこともできます。

- WebSphere Application Server 管理コンソールを使用して、Java 仮想マシン (JVM) のカスタム・プロパティの値を変更することができます。詳しくは、[Java 仮想マシンのカスタム・プロパティ](#) を参照してください。

Java **2.5+** [キャッシュ・インスタンス・プロパティ・ファイル](#)

cacheinstances.properties ファイルを使用して、オブジェクトまたはサーブレット・キャッシュを構成できます。

Java **2.5+** [動的キャッシュ・カスタム・プロパティ](#)

デフォルト動的キャッシュ・インスタンスやサーブレット・キャッシュ・インスタンスまたはオブジェクト・キャッシュ・インスタンスのカスタム・プロパティを設定する際、この表が参照として役立ちます。

親トピック: **Java** **2.5+** [動的キャッシュ・インスタンスの構成](#)

キャッシュ・インスタンス・プロパティ・ファイル

cacheinstances.properties ファイルを使用して、オブジェクトまたはサブレット・キャッシュを構成できます。

表 1. キャッシュ・インスタンス・プロパティ

プロパティ名 - x は インスタンス番号	必須	有効範囲	可能値	説明
cache.instance.x	はい	キャッシュ・ インスタンス 当たり	任意のストリング (デフォ ルト設定なし)	キャッシュ・インス タンス名または JNDI 名を指定します。
cache.instance.x.cac heSize	いいえ	キャッシュ・ インスタンス 当たり	> 0 (デフォルトは 2000)	WebSphere® eXtreme Scale グ リッド内の単一区画 で許容されるエント リーの最大数を指定 します。これに区画 数を掛ける と、WebSphere eXtreme Scale グ リッド内のキャッシ ュの容量が分かりま す。
cache.instance.x.cre ateCacheAtServerSt artup	いいえ	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	構成されたキャッシ ュ・インスタンスが サーバーの始動時に 作成されるかどうか を指定します。
cache.instance.x.ena bleServletSupport	いいえ	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	キャッシュ・インス タンスがサブレッ ト・キャッシュかオ ブジェクト・キャッシ ュかを指定しま す。
cache.instance.x.ena bleCacheReplication	はい (APAR までの み)	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	キャッシュがアプリ ケーション・サーバ ーからリモートであ ることを示していま す。WebSphere eXtreme Scale リ モート・トポロジー では、このプロパテ ィーを True に設定 する必要があります。
cache.instance.x.cac heProviderName	はい	キャッシュ・ インスタンス 当たり	com.ibm.ws.objectgrid. dynacache.CacheProvid erImpl	動的キャッシュ・プ ロバイダーを示しま す。WebSphere eXtreme Scale が 指定されなかった場 合は、WebSphere Application Server プロバイダーがデフ ォルトとして使用さ れます。
cache.instance.x.ign oreValueInInvalidatio nEvent	いいえ	キャッシュ・ インスタンス 当たり	true または false (デフォ ルトは false)	無効化イベントのキ ャッシュ値を無視す るかどうかを指定し ます。true の場合 は、コードが呼び出 し元に返されるとき に、無効化イベント のキャッシュの値が NULL に設定されま

cache.instance.x. <custom property>	追加した プロパ ティに よって 変わ りま す。	キャッ シュ・ イン スタ ンス 当 た り	追加したいプロパティ によって変わります。	す。 このファイルには、 すべてのカスタム・ プロパティを追加 できます。
--	---	---	--------------------------	---

親トピック: [Java 2.5+](#) [カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)

動的キャッシュ・カスタム・プロパティ

デフォルト動的キャッシュ・インスタンスやサブレット・キャッシュ・インスタンスまたはオブジェクト・キャッシュ・インスタンスのカスタム・プロパティを設定する際、この表が参照として役立ちます。

表 1. 動的キャッシュ・カスタム・プロパティ

プロパティ名	必須	有効範囲	指定可能な値	説明
com.ibm.websphere.xs.dynacache.topology	はい (WebSphere® Application Server APAR PM71992 が適用されると、以後このプロパティは不要になります。)	キャッシュ・インスタンス当たり	remote	キャッシュ・インスタンスのトポロジーを示します。remote トポロジーは、WebSphere DataPower® XC10 アプリケーションで使用できる唯一の有効なトポロジーです。。
com.ibm.ws.cache.CacheConfig.ignoreValueInInvalidationEvent	いいえ	キャッシュ・インスタンス当たり	true または false デフォルト: true	無効化イベントのキャッシュ値を無視するかどうかを指定します。true の場合は、コードが呼び出し元に返されるときに、無効化イベントのキャッシュの値が NULL に設定されます。
com.ibm.websphere.xs.dynacache.ignore_value_in_change_event	いいえ	キャッシュ・インスタンス当たり	true または false デフォルト: true	変更イベントのキャッシュ値を無視するかどうかを指定します。true の場合は、コードが呼び出し元に返されるときに、変更イベントのキャッシュの値が NULL に設定されます。
com.ibm.websphere.xs.dynacache.cs_override	いいえ	キャッシュ・インスタンス当たり	カタログ・サービス・エンドポイント 例: 9.5.12.345:2819	このキャッシュ・インスタンスと関連付けるデータ・グリッドのカタログ・サービス・エンドポイントを指定します。このフィールドは、WebSphere Application Server 管理コンソールで指定されなかった場合は必須です。
com.ibm.websphere.xs.dynacache.grid_name	いいえ	キャッシュ・インスタンス当たり	任意のストリング デフォルト: JNDI name of the cache instance	作成したデータ・グリッドの名前を指定してください。

com.ibm.websphere.xs.dynacache.map_template_name	いいえ	キャッシュ・インスタンスあたり	以下のマップ・テンプレートのいずれかを使用できます。 <ul style="list-style-type: none"> • IBM_DC_PARTITIONED_.* (デフォルト) • IBM_DC_NCI_PARTITIONED_.* (このキャッシュがニア・キャッシュを使用することを示します。) 	テンプレート・マップ接頭部の名前を指定します。
com.ibm.websphere.xs.dynacache.cache_name	いいえ	キャッシュ・インスタンスあたり	任意のストリング デフォルト: cache.instance.x の値	テンプレート・マップの名前として使用される固有の接尾部の名前を指定します。例えば、IBM_DC_PARTITIONED.<cache_name>
com.ibm.websphere.xs.dynacache.near_cache_size	いいえ	キャッシュ・インスタンスあたり	ゼロより大きい値。 デフォルト: cache.instance.x.cacheSize に指定された値	ニア・キャッシュ・インスタンスで許容されるエントリーの最大数を指定します。デフォルトでは、この値は、このキャッシュ・インスタンスのリモート ObjectGrid 内の単一区画で許容されるエントリーの最大数と同じです。
com.ibm.websphere.xs.dynacache.request_retry_timeout_override	いいえ	キャッシュ・インスタンスあたり	整数値 (ミリ秒): <ul style="list-style-type: none"> • -1: 無期限。要求はいつまでもタイムアウトになりません。 • 0: 要求はただちにタイムアウトになります。 • >0: 要求がタイムアウトにならずに実行される時間 (ミリ秒)。 • 2000: このカスタム・プロパティまたはクライアント・プロパティ 	要求がタイムアウトにならないで実行できる時間 (ミリ秒) を指定します。このプロパティは、クライアント・プロパティ・ファイルで requestRetryTimeout プロパティが設定されている場合、その値をオーバーライドします。詳しくは、 クライアント・プロパティ・ファイル を参照してください。 動的キャッシュ・インスタンスの要求再試行タイムアウト値が、このカスタム・プロパティでも、クライアント・プロパティ・ファイルでも設定されていない場合、デフォルトで 2000 ミリ秒になります。

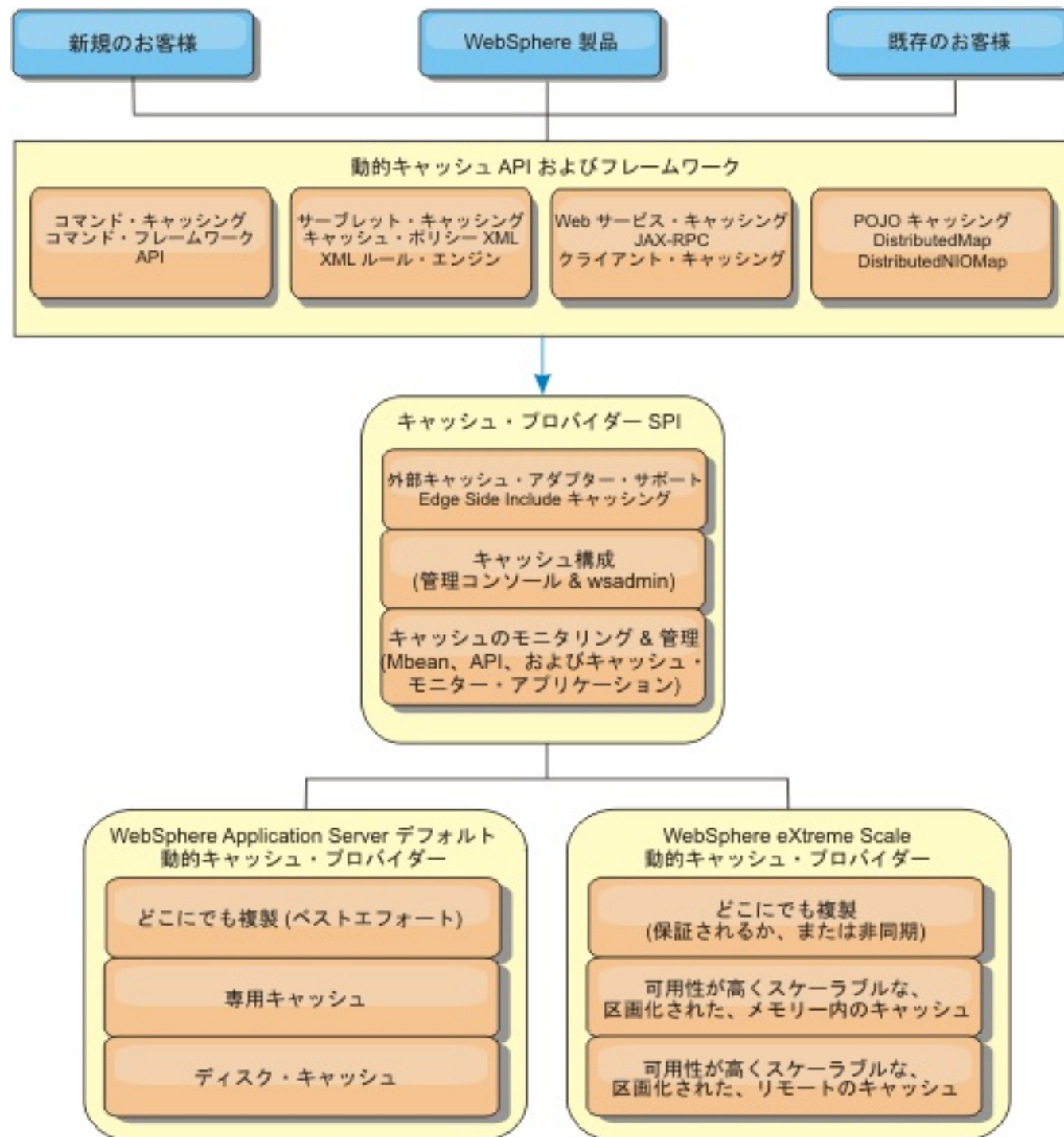
			ー・ファイルを使用してこの値を指定しなかった場合のデフォルト値。	
--	--	--	----------------------------------	--

親トピック: Java **2.5+** [カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)

動的キャッシュ・プロバイダーの概要

Java WebSphere® Application Server は、デプロイされた Java™ EE アプリケーションが使用できる動的キャッシュ・サービスを提供します。このサービスは、サーブレット、JSP、またはコマンドからの出力などのデータ、および DistributedMap API を使用してエンタープライズ・アプリケーション内にプログラマチックに指定されているオブジェクト・データをキャッシュするために使用されます。この機能をセットアップすることにより、動的キャッシュ・サービスを使用するアプリケーションが WebSphere DataPower® XC10 アプライアンス のフィーチャーおよびパフォーマンス機能を使用できるようになります。

当初は、動的キャッシュ・サービスのサービス・プロバイダーのみが、WebSphere Application Server に組み込まれているデフォルト動的キャッシュ・エンジンでした。また、WebSphere DataPower XC10 アプライアンス を任意のキャッシュ・インスタンス用のキャッシュ・プロバイダーに指定することもできます。この機能をセットアップすることにより、動的キャッシュ・サービスを使用するアプリケーションが のフィーチャーおよびパフォーマンス機能を使用できるようになります。



動的キャッシュ・プロバイダーのインストールと構成の手順については、[デフォルト動的キャッシュ・インスタンス \(baseCache\) の構成](#)を参照してください。

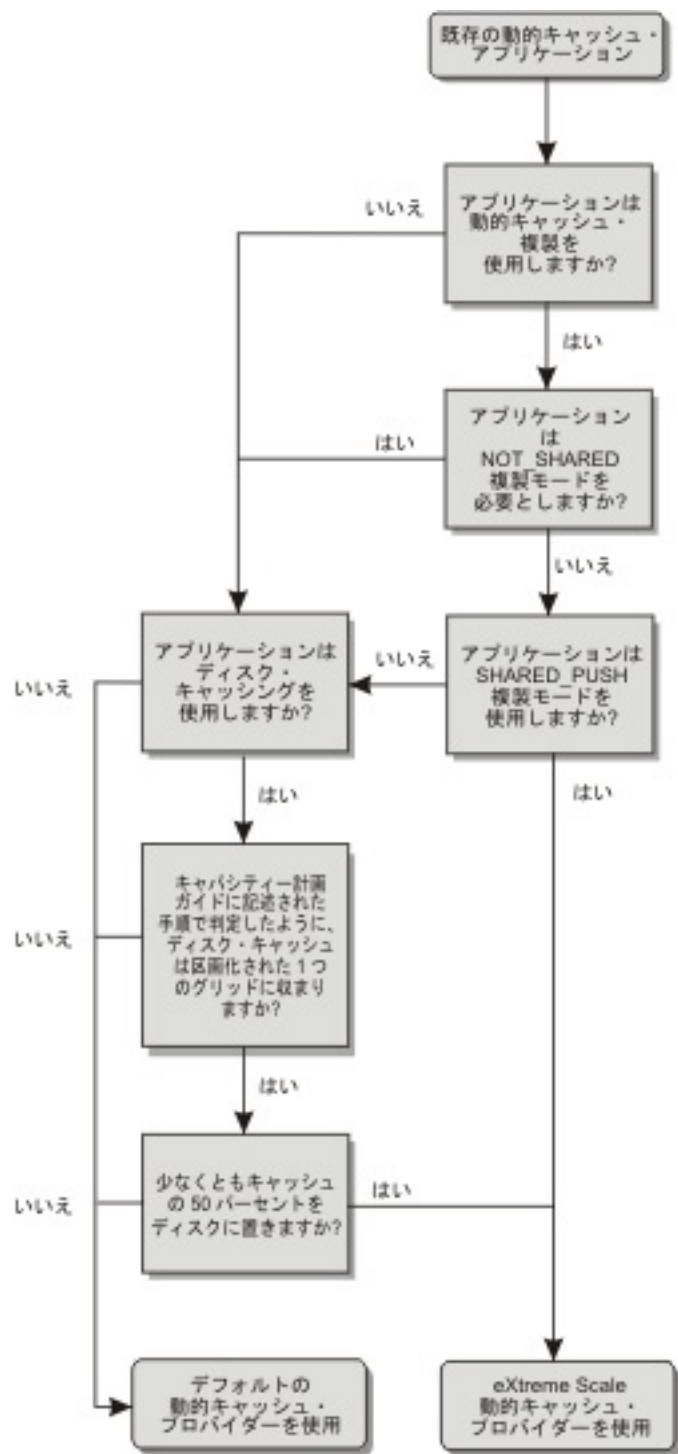
WebSphere DataPower XC10 アプライアンス

の使用方法の決定

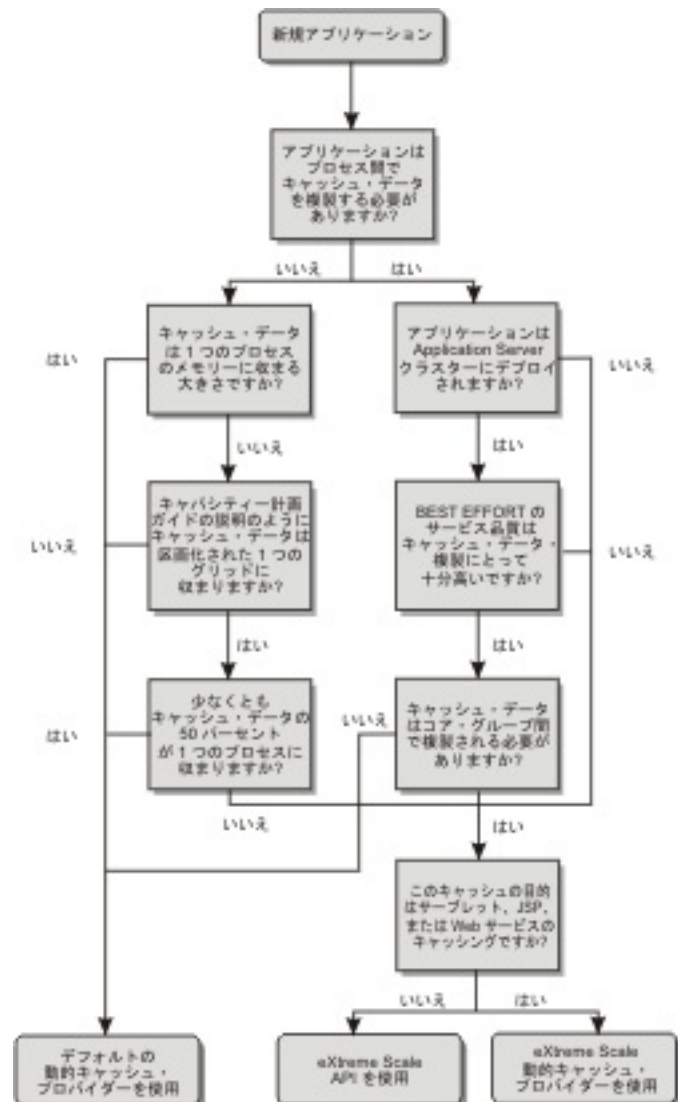
WebSphere DataPower XC10 アプライアンス の使用可能なフィーチャーにより、動的キャッシュ・サービスの分散機能は、デフォルト動的キャッシュ・プロバイダーおよびデータ複製サービスで提供される機能を超えて大幅に向上します。eXtreme Scale を使用することにより、複数のサーバー間で単に複製し、同期化するだけでなく、サーバー間で本当に分散したキャッシュを作成できます。さらに、eXtreme Scale キャッシュは、トランザクション・ベースであり、可用性がとて高く、動的キャッシュ・サービスに関して各サーバーが同じ内容を参照することを保証します。WebSphere DataPower XC10 アプライアンス は、キャッシュ複製に関して、DRS よりも高いサービス品質を提供します。

ただし、これらの利点は、どのようなアプリケーションでも eXtreme Scale 動的キャッシュ・プロバイダー が正しい選択であることを意味するわけではありません。以下のデシジョン・ツリーおよび機能比較マトリックスを使用して、ご使用のアプリケーションに最適のテクノロジーを決定してください。

既存の動的キャッシュ・アプリケーションをマイグレーションする際のデシジョン・ツリー



新規アプリケーションのキャッシュ・プロバイダーを選択する際のデシジョン・ツリー



機能比較

表 1. 機能比較

キャッシュ機能	デフォルト・プロバイダー	eXtreme Scale プロバイダー	eXtreme Scale API
ローカルのメモリー内のキャッシング	はい	ニア・キャッシュ機能を介して	ニア・キャッシュ機能を介して

分散キャッシング	DRS を介して	はい	はい
直線的にスケーラブル	いいえ	はい	はい
信頼できる複製 (同期)	いいえ	はい	はい
ディスク・オーバーフロー	はい	N/A	N/A
除去	LRU/TTL/ヒープ・ベース	LRU/TTL (区画ごと)	LRU/TTL (区画ごと)
Invalidation	はい	はい	はい
関係	依存関係 / テンプレート ID の関係	はい	いいえ (他の関係は可能)
非キー検索	いいえ	いいえ	照会と索引を介して
バックエンド統合	いいえ	いいえ	ローダーを介して
トランザクションの	いいえ	はい	はい
キー・ベースの保管	はい	はい	はい
イベントおよびリスナー	はい	いいえ	はい
WebSphere Application Server 統合	単一セルのみ	複数セル	セルに無関係
Java Standard Edition サポート	いいえ	はい	はい
モニタリングおよび統計	はい	はい	はい
セキュリティー	はい	はい	はい

注: eXtreme Scale 分散 キャッシュは、キーと値が両方とも java.io.Serializable インターフェースを実装するエンタリのみを保管できます。

動的キャッシュ・エンジンおよび eXtreme Scale の機能の相違点

WebSphere DataPower XC10 アプライアンス がバックアップしたキャッシュは、メモリー内キャッシュのサイズに関する統計および操作、またはディスク・オフロードをサポートしない点は除いて、ユーザーはこれら 2 つのキャッシュの機能的な違いに気付かないはずで

デフォルトの動的キャッシュ・プロバイダーを使用しているか、eXtreme Scale キャッシュ・プロバイダーを使用しているかに関わらず、ほとんどの場合、動的キャッシュ API 呼び出しが返す結果にそれほどの相違はありません。一部の操作については、eXtreme Scale を使用して動的キャッシュ・エンジンの動作をエミュレートできません。

動的キャッシュ統計

MBean 呼び出し

WebSphere eXtreme Scale 動的キャッシュ・プロバイダー はディスク・キャッシングをサポートしません。ディスク・キャッシングに関係する MBean 呼び出しはすべて 機能しません。

動的キャッシュ複製ポリシーのマッピング

eXtreme Scale 動的キャッシュ・プロバイダーのリモート・トポロジーは、(デフォルトの WebSphere Application Server 動的キャッシュ・プロバイダーによって使用される用語を使用して) SHARED_PULL および SHARED_PUSH_PULL ポリシーに最も近い複製ポリシーをサポートします。eXtreme Scale 動的キャッシュでは、キャッシュの分散状態はすべてのサーバー間で一貫しています。

グローバル索引の無効化

グローバル索引を使用して、大規模な区画化環境 (例えば、区画数が 40 を超えるような環境) での無効化効率を向上させることができます。グローバル索引フィーチャーを使用しない場合、動的キャッシュ・テンプレートおよび依存関係無効化処理ですべての区画にリモート・エージェント要求を送らなければならないため、結果としてパフォーマンスが低下します。グローバル索引を構成すると、無効化エージェントは、テンプレート ID または依存関係 ID に関連しているキャッシュ・エントリーを含む適用区画にのみ送られます。多数の区画が構成されている環境ではパフォーマンス向上の可能性がより大きくなります。サンプルの動的キャッシュ objectGrid 記述子 XML ファイルに入っている依存関係 ID 索引とテンプレート ID 索引を使用してグローバル索引を構成することができます。

ニア・キャッシュ

アプリケーション・サーバー JVM 内にローカルに存在するニア・キャッシュを作成して維持するように動的キャッシュ・インスタンスを構成することができます。このニア・キャッシュには、リモートの動的キャッシュ・インスタンス内に含まれるエントリーのサブセットが含まれます。詳しくは、[動的キャッシュのニア・キャッシュの構成](#)を参照してください。ニア・キャッシュを調整するためのカスタム・プロパティーもあります。詳しくは、[動的キャッシュ・カスタム・プロパティー](#)を参照してください。

2.5+

マルチマスター複製

WebSphere DataPower XC10 アプライアンス 動的キャッシュ・プロバイダーでマルチマスター複製を使用可能に設定する選択ができます。詳しくは、[動的キャッシュの作成データ・グリッド](#)を参照してください。

追加情報

- [動的キャッシュに関するレッドブック](#)
- 動的キャッシュ文書
 - [WebSphere Application Server 7.0](#)
- DRS 資料
 - [WebSphere Application Server 7.0](#)

親トピック: Java [動的キャッシュの作成データ・グリッド](#)

WebSphere Application Server でのカタログ・サービス・ドメインの作成

WebSphere® DataPower® XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

始める前に

- カatalog・サービス・ドメインの作成は、動的キャッシュ・データ・グリッドの場合にのみ必要です。単純なデータ・グリッドまたはセッション・データ・グリッドを使用する場合には、カタログ・サービス・ドメインを構成する必要はありません。WebSphere Application Server に WebSphere eXtreme Scale クライアントがインストールされている必要があります。

このタスクについて

カタログ・サービス・ドメインを作成することで、カタログ・サーバーの高可用性コレクションが定義されます。カタログ・サービス・ドメインを構成することによって、WebSphere DataPower XC10 アプライアンス で稼働しているカタログ・サーバーへの接続を確立します。このカタログ・サービス・ドメインは、集合内のアプライアンスで実行されているカタログ・サーバーのグループを表します。

手順

1. カatalog・サービス・ドメインを作成します。
 - a. WebSphere Application Server 管理コンソールで、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」 > 「新規」をクリックします。
 - b. カatalog・サービス・ドメインの名前、デフォルト値、JMX 認証資格情報を定義します。
 - c. カatalog・サーバー・エンドポイントを追加します。

リモート・サーバーのグループを指定します。これらは、集合内のアプライアンスで実行されているカタログ・サーバーです。集合内で実行されているカタログ・サーバーを表示するには、「集合」 > 「メンバー」 > 「*member_name*」をクリックします。「カタログ・サーバー」フィールドに、集合内で実行されているカタログ・サーバーがリストされます。IP アドレスまたは完全修飾ホスト名を使用して、エンドポイントを指定する必要があります。アプライアンスには次のポート値を使用します。

- クライアント・ポート: アプライアンス上で稼働するカタログ・サーバーへの接続には必須ではありません。
- リスナー・ポート: 2809

2. カatalog・サービス・ドメイン内のカタログ・サーバーへの接続をテストします。
 - a. WebSphere Application Server 管理コンソールで、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」をクリックします。
 - b. テストするカタログ・サービス・ドメインを選択して、「テスト接続」をクリックします。このボタンをクリックすると、すべての定義されたカタログ・サービス・ドメイン・エンドポイントが1つずつ照会されます。いずれかのエンドポイントが使用可能であれば、カタログ・サービス・ドメインへの接続が成功したことを示すメッセージが返されます。

[カタログ・サービス・ドメイン管理用タスク](#)

Jacl または Jython スクリプト言語を使用して、WebSphere Application Server 構成内のカタログ・サービス・ドメインを管理できます。WebSphere DataPower XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

親トピック: [Java](#) [動的キャッシュの作成データ・グリッド](#)

カタログ・サービス・ドメイン管理用タスク

Jacl または Jython スクリプト言語を使用して、WebSphere® Application Server 構成内のカタログ・サービス・ドメインを管理できます。WebSphere DataPower® XC10 アプライアンス を使用して、アプライアンス上で稼働しているカタログ・サーバーとの接続を確立するようにカタログ・サービス・ドメインを定義します。この構成の作成は、動的キャッシュ・データ・グリッドの場合にのみ必須です。

要件

WebSphere Application Server 環境に WebSphere eXtreme Scale クライアントをインストールしている必要があります。

すべての管理用タスクのリスト

カタログ・サービス・ドメインに関連したすべての管理用タスクのリストを取得するには、**wsadmin** で以下のコマンドを実行します。

```
wsadmin>$AdminTask help XSDomainManagement
```

コマンド

カタログ・サービス・ドメインの管理用タスクには、以下のコマンドが含まれます。

- [createXSDomain](#)
- [deleteXSDomain](#)
- [getDefaultXSDomain](#)
- [listXSDomains](#)
- [modifyXSDomain](#)
- [getTransport](#)
- [testXSDomainConnection](#)
- [testXSServerConnection](#)

createXSDomain

createXSDomain コマンドは、新規カタログ・サービス・ドメインを登録します。

表 1. createXSDomain コマンド引数

引数	説明
-name (必須)	作成するカタログ・サービス・ドメインの名前を指定します。
-default	カタログ・サービス・ドメインがセルでデフォルトかどうかを指定します。デフォルト値は true です。(ブール値: true または false に設定)
-properties	カタログ・サービス・ドメインのカスタム・プロパティを指定します。
-enableXIO	<p>このカタログ・サービス・ドメイン内でのトランスポート通信に IBM eXtreme IO (XIO) が使用されるかオブジェクト・リクエスト・ブローカー (ORB) が使用されるかを指定します。</p> <p>true XIO が使用されることを指定します。</p> <p>false ORB が使用されることを指定します。</p> <p>値を指定しなかった場合、デフォルトは true (XIO 使用可能) です。カタログ・サービス・ドメイン内にリモート・サーバーがある場合、-enableXIO パラメーターはそれらのリモート・サーバーで XIO も ORB も構成しません。リモート・サーバー上にトランスポートを構成するには、リモート・サーバーの始動時にトランスポート・タイプを指定します。</p>

表 2. defineDomainServers ステップ引数

引数	説明
name_endpoint	<p>カタログ・サービス・エンドポイントの名前を指定します。</p> <ul style="list-style-type: none"> • 既存のアプリケーション・サーバーの場合: エンドポイントの名前は、<code>cell_name ¥node_name¥server_name</code> の形式でなければなりません。 • リモート・サーバーの場合: リモート・サーバーのホスト名を指定します。複数のエンドポイントを同じ名前にすることができますが、クライアント・ポートの値はそれぞれのエンドポイントで固有でなければなりません。
custom	カタログ・サービス・ドメイン・エンドポイントのカスタム・プロパティを指定します。カスタム・プロパティがない場合、この引数には一組の二重引用符 (") を使用します。

<i>_properties</i>	
<i>endpoint_ports</i>	<p>カタログ・サービス・ドメイン・エンドポイントのポート番号を指定します。ポートは、<i><client_port></i>,<i><listener_port></i> の順序で指定する必要があります。</p> <p>クライアント・ポート カタログ・サービス・ドメイン内のカタログ・サーバー間の通信に使用するポートを指定します。この値は、WebSphere Application Server プロセスのみで稼働するカタログ・サーバーに必要で、他で使用されていないどのポートにも設定できます。</p> <p>リスナー・ポート クライアントとの通信に使用するポートを指定します。この値はリモート・エンドポイントに必要で、カタログ・サービスが開始されたときに使用された値と一致している必要があります。リスナー・ポートは、カタログ・サービスとの通信のために、クライアントおよびコンテナによって使用されます。</p> <p>WebSphere DataPower XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのリモート・エンドポイントに、値 2809 を使用します。</p>

戻り値:

バッチ・モードの使用例

バッチ・モードの場合、コマンド項目が正しくフォーマットされていることが必要です。入力する値が適切に処理されるように、対話モードの使用を検討してください。バッチ・モードを使用する場合、特定のプロパティ配列を使用して **-defineDomainServers** のステップ引数を定義する必要があります。このプロパティ配列のフォーマットは、*name_of_endpoint_custom_properties_endpoint_ports* です。 *endpoint_ports* 値は、*<client_port>*,*<listener_port>* の順序で指定する必要があるポートのリストです。

- Jacl を使用した、リモート・エンドポイントのカタログ・サービス・ドメインの作成:

```
$AdminTask createXSDomain {-name TestDomain -default true -defineDomainServers
{{xhost1.ibm.com "" ,2809}} }
```

- Jython ストリングを使用した、リモート・エンドポイントのカタログ・サービス・ドメインの作成:

```
AdminTask.createXSDomain('[-name TestDomain -default true
-defineDomainServers [[xhost1.ibm.com "" ,2809]
[xhost2.ibm.com "" ,2809]] ]')
```

- Jacl を使用した、既存のアプリケーション・サーバー・エンドポイントのカタログ・サービス・ドメインの作成:

```
$AdminTask createXSDomain {-name TestDomain -default true -defineDomainServers
{{cellName/nodeName/serverName "" 1109}} }
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask createXSDomain {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.createXSDomain (['-interactive'])
```

deleteXSDomain

deleteXSDomain コマンドは、カタログ・サービス・ドメインを削除します。

必須パラメーター:

-name

削除するカタログ・サービス・ドメインの名前を指定します。

戻り値:

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask deleteXSDomain {-name TestDomain }
```

- Jython ストリングを使用:

```
AdminTask.deleteXSDomain('[-name TestDomain ]')
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask deleteXSDomain {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.deleteXSDomain ('[-interactive]')
```

getDefaultXSDomain

getDefaultXSDomain コマンドは、セルのデフォルト・カタログ・サービス・ドメインを返します。

必須パラメーター: なし

戻り値: デフォルト・カタログ・サービス・ドメインの名前。

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask getDefaultXSDomain
```

- Jython ストリングを使用:

```
AdminTask.getDefaultXSDomain
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask getDefaultXSDomain {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.getDefaultXSDomain ('[-interactive]')
```

listXSDomains

listXSDomains コマンドは、既存のカタログ・サービス・ドメインのリストを返します。

必須パラメーター: なし

戻り値: セル内のすべてのカタログ・サービス・ドメインのリスト。

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask listXSDomains
```

- Jython ストリングを使用:

```
AdminTask.listXSDomains
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask listXSDomains {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.listXSDomains ('[-interactive]')
```

modifyXSDomain

modifyXSDomain コマンドは、既存のカタログ・サービス・ドメインを変更します。

バッチ・モードの場合、コマンド項目が正しくフォーマットされていることが必要です。入力する値が適切に処理されるように、対話モードの使用を検討してください。バッチ・モードを使用する場合、特定のプロパティ配列を使用して **-modifyEndpoints**、**-addEndpoints**、および **-removeEndpoints** ステップ引数を定義する必要があります。このプロパティ配列のフォーマットは、*name_of_endpoint host_name custom_properties endpoint_ports* です。 *endpoint_ports* 値は、*<client_port>*、*<listener_port>* の順序で指定する必要があるポートのリストです。

表 3. *modifyXSDomain* コマンド引数

引数	説明
-name (必須)	編集するカタログ・サービス・ドメインの名前を指定します。
-default	true に設定した場合、選択したカタログ・サービス・ドメインがセルのデフォルトであることを指定します。(ブール値)
-properties	カタログ・サービス・ドメインのカスタム・プロパティを指定します。
-enableXIO	<p>このカタログ・サービス・ドメイン内でのトランスポート通信に IBM eXtreme IO (XIO) が使用されるかオブジェクト・リクエスト・ブローカー (ORB) が使用されるかを指定します。</p> <p>true XIO が使用されることを指定します。</p> <p>false ORB が使用されることを指定します。</p> <p>値を指定しなかった場合、デフォルトは true (XIO 使用可能) です。カタログ・サービス・ドメイン内にリモート・サーバーがある場合は、それらのリモート・サーバーで XIO を構成することはできません。</p>

表 4. *modifyEndpoints* ステップ引数

引数	説明
<i>name_of_endpoint</i>	<p>カタログ・サービス・エンドポイントの名前を指定します。</p> <ul style="list-style-type: none"> 既存のアプリケーション・サーバーの場合: エンドポイントの名前は、<i>cell_name ¥node_name¥server_name</i> の形式でなければなりません。 リモート・サーバーの場合: リモート・サーバーのホスト名を指定します。複数のエンドポイントを同じ名前にすることができますが、リスナー・ポートの値はそれぞれのエンドポイントで固有でなければなりません。アプライアンスを構成する場合、この値は完全修飾のドメイン名である必要があります。
<i>endpoint_ports</i>	<p>カタログ・サービス・ドメイン・エンドポイントのポート番号を指定します。エンドポイントは、<i><client_port></i>、<i><listener_port></i> の順序で指定する必要があります。</p> <p>クライアント・ポート カタログ・サービス・ドメイン内のカタログ・サーバー間の通信に使用するポートを指定します。この値は、WebSphere Application Server プロセスのみで稼働するカタログ・サーバーに必要で、他で使用されていないどのポートにも設定できます。</p> <p>リスナー・ポート クライアントとの通信に使用するポートを指定します。この値はリモート・エンドポイントに必要で、カタログ・サービスが開始されたときに使用された値と一致している必要があります。リスナー・ポートは、カタログ・サービスとの通信のために、クライアントおよびコンテナによって使用されます。 WebSphere DataPower XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのリモート・エンドポイントに、値 2809 を使用します。</p>

表 5. *addEndpoints* ステップ引数

引数	説明
<i>name_of_endpoint</i>	<p>カタログ・サービス・エンドポイントの名前を指定します。</p> <ul style="list-style-type: none"> 既存のアプリケーション・サーバーの場合: エンドポイントの名前は、<i>cell_name ¥node_name¥server_name</i> の形式でなければなりません。

<code>ndp oin t</code>	<ul style="list-style-type: none"> • リモート・サーバーの場合: リモート・サーバーのホスト名を指定します。複数のエンドポイントを同じ名前にすることができますが、リスナー・ポートの値はそれぞれのエンドポイントで固有でなければなりません。アプライアンスを構成する場合、この値は完全修飾のドメイン名である必要があります。
<code>cus tom _pr ope rti es</code>	<p>カタログ・サービス・ドメイン・エンドポイントのカスタム・プロパティを指定します。カスタム・プロパティがない場合、この引数には一組の二重引用符 ("") を使用します。</p>
<code>end poi nt_ por ts</code>	<p>カタログ・サービス・ドメイン・エンドポイントのポート番号を指定します。エンドポイントは、<code><client_port></code>,<code><listener_port></code> の順序で指定する必要があります。</p> <p>クライアント・ポート</p> <p>カタログ・サービス・ドメイン内のカタログ・サーバー間の通信に使用するポートを指定します。この値は、WebSphere Application Server プロセスのみで稼働するカタログ・サーバーに必要で、他で使用されていないどのポートにも設定できます。</p> <p>リスナー・ポート</p> <p>クライアントとの通信に使用するポートを指定します。この値はリモート・エンドポイントに必要で、カタログ・サービスが開始されたときに使用された値と一致している必要があります。リスナー・ポートは、カタログ・サービスとの通信のために、クライアントおよびコンテナによって使用されます。</p> <p>WebSphere DataPower XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのリモート・エンドポイントに、値 2809 を使用します。</p>

表 6. `removeEndpoints` ステップ引数

引数	説明
<code>name_of_endpoint</code>	削除するカタログ・サービス・エンドポイントの名前を指定します。

戻り値:

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask modifyXSDomain [-name TestDomain -default true -modifyEndpoints
{{xhost1.ibm.com "" ,2809}} -addEndpoints {{xhost2.ibm.com "" ,2809}}
-removeEndpoints {{xhost3.ibm.com}}]
```

- Jython スtring を使用:

```
AdminTask.modifyXSDomain('[-name TestDomain
-default false -modifyEndpoints [[xhost1.ibm.com "" ,2809]]
-addEndpoints [[xhost3.ibm.com "" ,2809]]
-removeEndpoints [[xhost2.ibm.com]]]')
```

- IBM eXtremeIO がオンになるように既存のカタログ・サービス・ドメインを変更します。

```
AdminTask.modifyXSDomain('[-name testDomain -enableXIO true]')
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask modifyXSDomain {-interactive}
```

- Jython スtring を使用:

```
AdminTask.modifyXSDomain ('[-interactive]')
```

getTransport

`getTransport` コマンドは、カタログ・サービス・ドメインのトランスポート・タイプ、すなわち IBM eXtremeIO (XIO) またはオブジェクト・リクエスト・ブローカー (ORB) を表示します。リモート・サーバーを含むカタログ・サービス・ドメインでこのコマンドを実行した場合、または `catalogServerName` がリモート・サーバーである場合は、エラーとなります。リモート・サーバーに対しては `xscmd -c showTransport` コマンドを使用する必要があります。

必須パラメーター:

-domainName

トランスポート・タイプを表示するカタログ・サービス・ドメインの名前を指定します。

-catalogServerName

トランスポート・タイプを表示するカタログ・サーバーの名前を指定します。

戻り値: ORB または XIO

カタログ・サービス・ドメインのトランスポートの表示

- Jacl を使用:

```
$AdminTask getTransport {-domainName TestDomain }
```

- Jython スtring を使用:

```
AdminTask.getTransport('[-domainName testDomain]')
```

カタログ・サーバーのトランスポートの表示

- Jacl を使用:

```
$AdminTask getTransport {-catalogServerName myCell01¥myNode01¥container1 }
```

- Jython スtring を使用:

```
AdminTask.getTransport('[-catalogServerName myCell01¥myNode01¥container1]')
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask getTransport {-interactive}
```

- Jython スtring を使用:

```
AdminTask.getTransport ('[-interactive]')
```

testXSDomainConnection

testXSDomainConnection コマンドは、カタログ・サービス・ドメインへの接続をテストします。

必須パラメーター:

-name

接続をテストするカタログ・サービス・ドメインの名前を指定します。

オプション・パラメーター

-timeout

接続されるまで待機する最大時間を秒数で指定します。

戻り値: 接続できた場合、true が返されます。接続できなかった場合は、接続エラー情報が返されます。

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask testXSDomainConnection
```

- Jython スtring を使用:

```
AdminTask.testXSDomainConnection
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask testXSDomainConnection {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.testXSDomainConnection ('[-interactive]')
```

testXSServerConnection

testXSServerConnection コマンドは、カタログ・サーバーへの接続をテストします。このコマンドは、スタンドアロン・サーバーと、カタログ・サービス・ドメインに属するサーバーの両方で機能します。

必須パラメーター:

ホスト (host)

カタログ・サーバーが存在するホストを指定します。

listenerPort

カタログ・サーバーのリスナー・ポートを指定します。

オプション・パラメーター

timeout

カタログ・サーバーに接続されるまで待機する最大時間を秒数で指定します。

戻り値:

バッチ・モードの使用例

- Jacl を使用:

```
$AdminTask testXSServerConnection {-host xhost1.ibm.com -listenerPort 2809}
```

- Jython ストリングを使用:

```
AdminTask.testXSServerConnection('[-host xshost3.ibm.com -listenerPort 2809]')
```

対話モードの使用例

- Jacl を使用:

```
$AdminTask testXSServerConnection {-interactive}
```

- Jython ストリングを使用:

```
AdminTask.testXSServerConnection ('[-interactive]')
```

親トピック: [Java](#) [WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)

動的キャッシュのニア・キャッシュの構成

2.5+ アプライアンスまたは集合で、動的キャッシュ・データ・グリッドと一緒に動作するローカルのニア・キャッシュを構成することができます。ニア・キャッシュは、ローカル Java 仮想マシン (JVM) リソースを使用します。一般に、ニア・キャッシュには、アプライアンスの動的キャッシュ・データ・グリッドにあるデータのサブセットが保持されません。

始める前に

動的キャッシュ・データ・グリッドを作成します。詳しくは、[動的キャッシュの作成データ・グリッド](#)を参照してください。

このタスクについて

動的キャッシュ・データ・グリッドのニア・キャッシュを使用可能にするには、WebSphere Application Server 動的キャッシュ・インスタンスにマップ・テンプレート名を指定するカスタム・プロパティを設定する必要があります。

手順

ニア・キャッシュを使用可能にするには、以下のカスタム・プロパティを設定します。

- `com.ibm.websphere.xs.dynacache.map_template_name`: テンプレート名を変更するには、`IBM_DC_NCI_PARTITIONED_.*` を指定します。キャッシュ・インスタンスでこのカスタム・プロパティを設定すると、そのキャッシュ・インスタンスに対するニア・キャッシュが使用可能になります。

動的キャッシュでのカスタム・プロパティの設定について詳しくは、[カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)を参照してください。

次のタスク

デフォルトでは、動的ニア・キャッシュの最大サイズは、リモートの動的キャッシュ・データ・グリッド内の単一区画の最大エントリー数に相当します。このサイズを維持するために最長未使用時間 (LRU) 除去アルゴリズムが使用されます。ニア・キャッシュのサイズを詳細に制御するには、次のアクションを実行することができます。

- ニア・キャッシュで許容される最大キャッシュ・エントリー数を指定するように `com.ibm.websphere.xs.dynacache.near_cache_size` カスタム・プロパティを構成します。詳しくは、[動的キャッシュ・カスタム・プロパティ](#)を参照してください。

親トピック: [Java](#) [動的キャッシュの作成データ・グリッド](#)

データ・グリッドの最大容量の構成

集合内の各データ・グリッドの最大容量を定義できます。容量限度を構成すると、特定のデータ・グリッドで使用可能なデータ・ストレージの容量が制限されます。この容量限度により、確実に、集合の使用可能なストレージ容量が予測可能な形で使用されるようになります。

始める前に

- ご使用の構成用のデータ・グリッドを作成します。デフォルトでは、データ・グリッドの最大容量限度は構成されません。すべてのタイプのデータ・グリッド (単純データ・グリッド、セッション・データ・グリッド、動的キャッシュ・データ・グリッド) で最大容量を構成できます。

このタスクについて

集合内の各データ・グリッドの最大容量限度の構成後、構成したデータ・グリッド容量限度をデータ・グリッド内の基本データの合計サイズと比較して、容量限度を順守します。データ・グリッドと構成された容量限度とを比較する際には、データのレプリカ・コピーによって使用される容量は比較対象に含まれません。

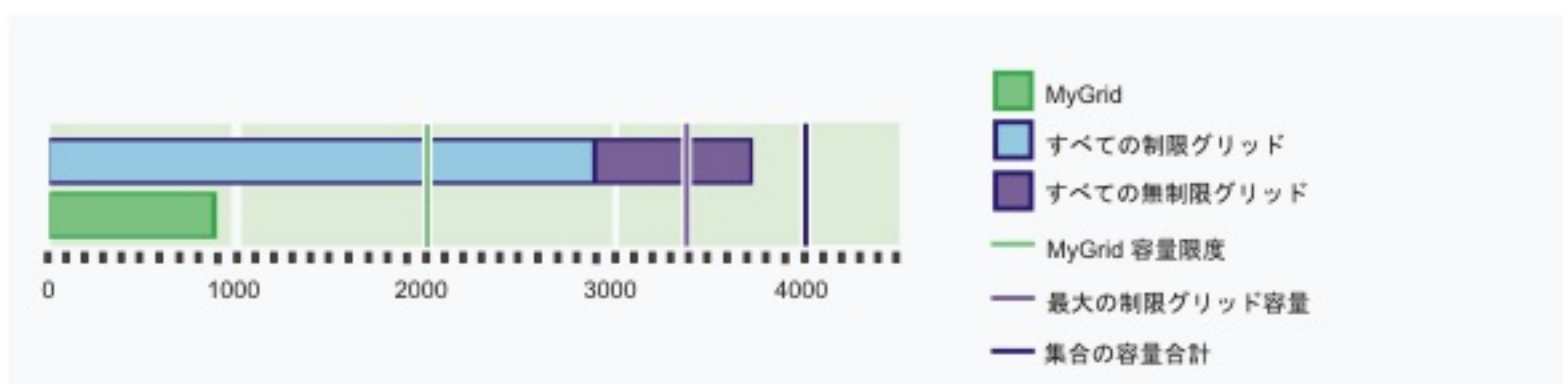
最大容量限度とは、データ・グリッドに挿入可能なデータの最大容量です。この限度を構成しても、データ・グリッドに割り振られるスペース容量は保証されません。したがって、集合にデータを格納できる十分な容量がない場合には、データ・グリッドで構成されている容量限度に達しないことがあります。集合での容量不足の原因としては、データ・グリッドの高い容量限度、または集合内の他のデータ・グリッドに使用された容量の可能性にあります。

特定のデータ・グリッドで容量限度を超えると、グリッドは次のいずれかの方法で挿入操作を処理します。

- デフォルトでは、グリッドの限界容量を超えることが予想される挿入操作は拒否されます。クライアント・プロセスでは、挿入要求に対して例外を受け取ります。ただし読み取り操作、更新操作、および削除操作は、データ・グリッドの容量限度を超える場合でも正常に実行されます。これらの限られた操作では、データ・グリッドは最小機能レベルで稼働できますが、データ・グリッドのサイズが大きくなることが回避されます。
- 単純データ・グリッドでのみ使用可能ですが、「このデータ・グリッドに最長未使用時間 (LRU) の除去を使用」オプションを選択することで、このデフォルトの動作をオーバーライドできます。このオプションを選択すると、最長未使用時間のデータ入力項目を削除することで、単純データ・グリッドへの新しい挿入が受け入れられ、容量限度が保持されます。最大容量をデータ・グリッドに設定しても再始動は必要ありませんが、単純グリッドで LRU オプションを選択した場合は、変更を有効にするためにグリッドが自動的に再始動されます。これは、単純グリッドで LRU オプションをオフにする場合でも同様です。

手順

- ユーザー・インターフェースで「データ・グリッド」 > 「`data_grid_type`」 > 「`data_grid_name`」 > 「拡張属性の表示」とクリックします。
- 「このデータ・グリッドの容量を制限する」を選択します。
- 単純データ・グリッドの最大容量を設定する際、最長未使用時間のデータ入力項目を犠牲にして、グリッドが新規挿入操作を拒否せずに受け入れるようにするには、「このデータ・グリッドに最長未使用時間 (LRU) の除去を使用」を選択します。「変更の適用」をクリックして保存します。再始動を完了するためにグリッド内のデータが失われる旨の警告が表示されます。
- 現在の使用容量を確認し、選択したデータ・グリッドに対して定義する最大容量を決定します。また、集合の合計容量を超えていないことも確認できます。



このグラフでは、構成する現行データ・グリッド (MyGrid) の現在の使用容量が 900 メガバイトです。このデータ・グリッドに対して現在構成されている容量限度は 2000 メガバイトです。集合のレベルでは、その集合の合計容量は 4000 メガバイトです。さらに、容量が制限されているデータ・グリッドの構成の容量限度の合計は 3400 メガバイトです。これらのグリッドの現在の使用容量は 2900 メガバイトです。最後に、この集合内で、容量限度が定義されていないデータ・グリッドが少なくとも 1 つ存在します。容量限度が未定義のこれらのデータ・グリッドによる使用容量は約 900 メガバイトです。

- 基本データの使用量限度の値を MB 単位で入力します。Enter キーを押すと、基本データとレプリカ・データの最大使用可能容量が表示されます。この数値は、定義したレプリカの数に応じて変化します。ただし、レプリカ数は集合内のアプライアンスの数によって制限される点に注意してください。4 つのレプリカが定義されており、集合内に 3 つのアプライアンスがある場合、集合には 1 つの基本データと 2 つのレプリカが含まれます。
- 「変更の適用」をクリックして構成を保存します。新しい限度を有効にするためにデータ・グリッドを再始動する必要はありません。

例

容量限度の例: 複数データ・グリッド

集合内でデータ・グリッド A、B、および C が定義されており、合計ストレージ容量が 600 ギガバイトであるとし、ます。どのデータ・グリッドでもレプリカは定義されていません。データ・グリッド A の容量限度は 100 ギガバイトです。データ・グリッド B の容量限度は 50 ギガバイトです。データ・グリッド C の容量限度は 200 ギガバイトです。このシナリオでは、集合内では常に 250 ギガバイト以上の未使用容量があることになります。3 つのデータ・グリッドの合計サイズが 350 ギガバイトを超えることはできません。

容量限度の例: レプリカ

2 つのアプライアンスが含まれている集合で、データ・グリッド A が定義されています。データ・グリッド A には、3 つのレプリカ (1 つの同期レプリカと 2 つの非同期レプリカ) があります。このグリッドの容量限度は 100 メガバイトとして定義されています。最初はこのグリッドの最大使用容量は 200 メガバイトです。この集合に含まれているアプライアンスが 2 つのみであるため、1 つの基本データと 1 つのレプリカ・コピーが存在します。基本データ・グリッドは 100 メガバイトまで使用できます。レプリカは基本データ・グリッドと同じ割合で増大するため、最大使用容量合計が 200 メガバイトになります。集合に 3 つ目のアプライアンスが追加されると、2 つ目のデータ・レプリカ・コピーが配置されます。グリッドの最大使用量は、1 つの基本データと 2 つのレプリカによって 300 メガバイトになります。

容量限度の例: 容量限度が未定義のデータ・グリッド


集合内でデータ・グリッド A、B、および C が定義されており、合計ストレージ容量が 600 ギガバイトであるとし、ます。データ・グリッド A の容量限度は 100 ギガバイトです。データ・グリッド B の容量限度は 50 ギガバイトです。データ・グリッド C には容量限度が定義されていません。3 つのデータ・グリッドのいずれでもレプリカが定義されていません。データ・グリッド C には限度が定義されていないため、このデータ・グリッドによって、使用可能な容量全体 (600 ギガバイト) が使用される可能性があります。この場合、データ・グリッド A とデータ・グリッド B ではデータを挿入できなくなります。データ・グリッド A またはデータ・グリッド B で挿入されたデータはすべて保持されますが、これらのデータ・グリッドの容量限度に達することは保証されません。データ・グリッド C の最小使用可能容量として 450 ギガバイトが保証されます。これは、システムのその他のデータ・グリッドによる使用容量が、600 ギガバイトの容量のうち、これらのデータ・グリッドの容量限度合計である 150 ギガバイトを超えることがないためです。この 450 ギガバイトを算出する計算では、レプリカ・データによる使用容量が無視されています。2 つ以上の無制限データ・グリッドが集合内にある場合、特定のデータ・グリッドの使用可能容量は保証されません。

親トピック: [データ・グリッドの構成](#)

データ・グリッドのクリア

データ・グリッド内のすべての項目を恒久的に削除することができます。失効した情報またはテスト項目を除去するために、データ・グリッドをクリアできます。

手順

1. ユーザー・インターフェースで、「データ・グリッド」 > 「**data_grid_type**」 > 「**data_grid_name**」をクリックします。
2. 「グリッドのクリア」アイコン () をクリックして、データ・グリッドのすべての項目を削除します。データ・グリッド内のすべての項目を除去することを確認する必要があります。
3. ユーザー・インターフェースで、データ・グリッド項目が削除されたことを確認できます。「モニター」 > 「個別データ・グリッドの概要」 > 「**data_grid_name**」をクリックして、「使用容量対キャッシュ・エントリー数」チャートを表示します。単純データ・グリッドおよびセッション・データ・グリッドの場合、データ・グリッド内の項目数はゼロに近づくはずですが、動的キャッシュ・データ・グリッドでは、データ・グリッドの一部の項目が残ります。これらのデータ・グリッド項目には、動的キャッシュ・データ・グリッドの構成情報が含まれます。

親トピック: [データ・グリッドの構成](#)

データ・グリッドの除去

データ・グリッド (data grid)のデータを初期化する場合は、データ・グリッド (data grid)を除去してから データ・グリッド (data grid)を再作成します。

このタスクについて

手順

1. ユーザー・インターフェースで、「データ・グリッド」 > 「**data_grid_type**」をクリックします。削除する **data_grid_name** を選択します。
2. 除去プロセスを開始するには、「除去」アイコン (✖) をクリックします。メッセージ・ボックスが表示され、このデータ・グリッド (data grid)が完全に除去されることに対する確認が要求されます。「**OK**」をクリックして、除去を確認します。
3. データ・グリッド (data grid)の除去は、「タスク」ビューでモニターできます。

タスクの結果

親トピック: [データ・グリッドの構成](#)

Spring キャッシュ・プロバイダーの構成

Spring Framework バージョン 3.1 では、新しいキャッシュの抽象化が導入されています。この新しい抽象化を使用すると、既存の Spring アプリケーションに対してキャッシュを容易に追加できます。WebSphere® DataPower® XC10 アプライアンス をキャッシュ抽象化のキャッシュ・プロバイダーとして使用できます。

始める前に

- Spring Framework バージョン 3.1 以降を使用するアプリケーションが必要です。
- ご使用のアプリケーションで注釈を使用してキャッシュ方式を宣言する必要があります。キャッシュ抽象化のためにご使用のアプリケーションを更新する方法について、詳しくは [Spring Framework Reference Documentation : Cache abstraction](#) を参照してください。
- ogclient.jar ファイルが Spring アプリケーションのクラスパスにあることを確認してください。
- アプリケーションを実行する JVM が、WebSphere eXtreme Scale クライアントによってインストールされる JVM でない場合、次の JVM 引数を追加して IBM Object Request Broker (ORB) が使用されるようにする必要があります。

```
-Djava.endorsed.dirs=wx_root/lib/endorsed
```

- ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。
- Spring アプリケーションをセキュア・データ・グリッドに接続する場合、適切な client.properties ファイルを **client-security-config** パラメーターの値として指定する必要があります。このパラメーターは、Spring Inversion of Control (IoC) コンテナ構成にある ObjectGridCatalogServiceDomainBean に指定します。クライアント認証を使用し、セキュアなネットワーク・トランスポート用の TLS を使用するように Spring キャッシュ・プロバイダーを構成することができます。詳しくは、[クライアント・プロパティ・ファイル](#)、[クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)、および[データ・グリッド・アプリケーション用の TLS の構成](#)を参照してください。
- カタログ・サーバーのホスト名とアプライアンスのポートが分かっている必要があります。この情報を得るには、ユーザー・インターフェースで「集合」 > 「メンバー」をクリックします。

このタスクについて

Spring Framework でキャッシュ抽象化を使用することで、Spring アプリケーションのメソッドの実行回数を減らすことができます。構成後は、特定のメソッドの結果はキャッシュに保存されます。メソッドが再実行されると、抽象化でキャッシュがチェックされ、メソッドの結果が既にキャッシュに存在しないか確認されます。結果がキャッシュに存在すれば、結果がキャッシュから返され、メソッドは再実行されません。そのため、負荷のかかるメソッドの実行回数を減らすと共に、ご使用のアプリケーションの平均応答時間を短くすることもできます。

手順

WebSphere DataPower XC10 アプライアンス をキャッシュ・プロバイダーとして使用するよう、Spring Inversion of Control (IoC) コンテナを構成します。WebSphere DataPower XC10 アプライアンス のキャッシュ実装環境は、com.ibm.websphere.objectgrid.spring パッケージ内にあります。Spring IoC コンテナ構成で以下の Bean を定義します。

```
<bean id="wxscsDomain"
class="com.ibm.websphere.objectgrid.spring.ObjectGridCatalogServiceDomainBean"
  p:catalog-service-endpoints="CATALOG_SERVICE_ENDPOINTS"
 />

<bean id="wxsgClient" class="com.ibm.websphere.objectgrid.spring.ObjectGridClientBean"
  p:object-grid-name="OBJECT_GRID_NAME"
  p:catalog-service-domain-ref="wxscsDomain" />

<bean id="cacheManager" class="org.springframework.cache.support.SimpleCacheManager">
  <property name="caches">
    <set>
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="CACHE_NAME"
        p:map-name="MAP_NAME "
        p:object-grid-client-ref="wxsgClient" />
    </set>
  </property>
</bean>
```

CATALOG_SERVICE_ENDPOINTS

カタログ・サーバーのホスト名とポートを指定します。

クライアント・サイドで設定を変更する ObjectGrid XML ファイルの絶対パスまたは相対パスを、Spring リソースとして指定します。Spring 内のリソースの指定については、[Spring Framework Reference Documentation: Resources](#) を参照してください。

例:p:client-override-xml="file:/path/to/objectgrid.xml"

例: p:client-override-xml="classpath:com/example/app/override-objectgrid.xml"

例: p:client-override-xml="http://myserver/override-objectgrid.xml"

例: p:client-override-xml="ftp://myserver/override-objectgrid.xml"

CLIENT_SECURITY_CONFIG (オプション)

client.properties ファイルの絶対パスまたは相対パスを、Spring リソースとして指定します。Spring 内のリソースの指定については、[Spring Framework Reference Documentation: Resources](#) を参照してください。WebSphere DataPower XC10 アプライアンスの client.properties の作成については、[クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)を参照してください。

例: p:client-security-config="file:/path/to/client.properties"

OBJECT_GRID_NAME

ObjectGrid 名を指定します。提供された XML 構成ファイルを使用してコンテナ・サーバーを開始する場合、このパラメーターは不要です。この値は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。

CACHE_NAME

Spring キャッシング・アプリケーションで指定されるキャッシュの名前を指定します。

MAP_NAME

キャッシュのバッキング・マップの名前を指定します。この値は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。デフォルト値以外のマップ名を使用する場合は、動的マップを定義することができます。動的マップの作成については、[動的マップの構成オプション](#)を参照してください。

例

以下のスニペットは、myXC10.myhost.com:2809 でアプライアンスがホスティングする default という名前のキャッシュを作成します。この例は、データ・グリッドにちなんで名前が付けられるデフォルト・マップ・インスタンスを使用しています。

```
<bean id="wxsCSDomain"
class="com.ibm.websphere.objectgrid.spring.ObjectGridCatalogServiceDomainBean"
  p:catalog-service-endpoints ="myXC10.myhost.com:2809" />
<bean id="wxsGridClient" class="com.ibm.websphere.objectgrid.spring.ObjectGridClientBean"
  p:object-grid-name="my_simple_data_grid"
  p:catalog-service-domain-ref="wxsCSDomain" />
<bean id="cacheManager" class="org.springframework.cache.support.SimpleCacheManager">
  <property name="caches">
    <set>
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="default"
        p:map-name="my_simple_data_grid"
        p:object-grid-client-ref="wxsGridClient" />
    </set>
  </property>
</bean>
```

親トピック: [データ・グリッドの構成](#)

クライアントの構成

クライアント上の一部のプロパティは、サーバーで設定されるプロパティをオーバーライドすることで構成できます。これらのプロパティは、クライアントのプロパティ・ファイルでオーバーライドするか、またはプログラムでオーバーライドすることができます。

Java クライアントの構成

スタンドアロン環境または WebSphere® Application Server を使用した環境で実行するように WebSphere eXtreme Scale を構成できます。グリッドのサーバー・サイドの構成変更を WebSphere eXtreme Scale デプロイメントに反映するには、動的に適用するのではなく、プロセスを再始動して変更を有効にする必要があります。一方、クライアント・サイドでは、既存のクライアント・インスタンスの構成設定は変更できませんが、XML ファイルを使用するか、プログラムで、新しいクライアント・インスタンスを必要な設定で作成できます。クライアントの作成時には、現行のサーバー構成からのデフォルト設定をオーバーライド可能です。

.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成

WebSphere eXtreme Scale Client for .NET を構成するには、クライアント・プロパティ・ファイルを使用するか、サーバー・サイド XML 構成を使用するか、または一部のサーバー・プロパティをプログラマチックにオーバーライドします。

Java クライアントの構成

スタンドアロン環境または WebSphere® Application Server を使用した環境で実行するように WebSphere eXtreme Scale を構成できます。グリッドのサーバー・サイドの構成変更を WebSphere eXtreme Scale デプロイメントに反映するには、動的に適用するのではなく、プロセスを再始動して変更を有効にする必要があります。一方、クライアント・サイドでは、既存のクライアント・インスタンスの構成設定は変更できませんが、XML ファイルを使用するか、プログラムで、新しいクライアント・インスタンスを必要な設定で作成できます。クライアントの作成時には、現行のサーバー構成からのデフォルト設定をオーバーライド可能です。

次の方法で、eXtreme Scale クライアント (Java クライアントのみ) を構成することができます。いずれの方法も、クライアント・オーバーライド XML ファイルを使用して、もしくはプログラムで実行することができます。

- XML 構成
- プログラマチック構成
- Spring Framework 構成
- ニア・キャッシュの使用不可化

[Java クライアントのオーバーライド](#)

サーバーの設定をオーバーライドすることにより、要件に基づいて WebSphere eXtreme Scale クライアントを構成することができます。いくつかのプラグインおよび属性をオーバーライドすることができます。

[XML 構成を使用した Java クライアントの構成](#)

ObjectGrid 構成 XML ファイルを使用して、クライアント・サイドの設定をオーバーライドできます。

[Java クライアントのプログラマチック構成](#)

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

[要求タイムアウト値および再試行タイムアウト値の構成](#)

eXtreme Scale クライアント・コードが要求の完了をどれくらい待ち、このクライアント・コードがデータ・グリッドへのアクセスと関連付けられる要求をいつまで再試行するかを制御する調整オプションを指定することができます。

[データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

セッション管理を使用し、WebSphere Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、アプライアンス内のデータ・グリッドを使用してそれらのアプリケーションを管理するように Liberty プロファイルを構成することができます。

[REST ゲートウェイのデプロイ](#)

WebSphere Application Server または Liberty プロファイル・サーバーでデータ・グリッド用の REST ゲートウェイをデプロイして構成できます。

親トピック: [クライアントの構成](#)


Java クライアントのオーバーライド

サーバーの設定をオーバーライドすることにより、要件に基づいて WebSphere® eXtreme Scale クライアントを構成することができます。いくつかのプラグインおよび属性をオーバーライドすることができます。

クライアントの設定をオーバーライドするには、XML 構成またはプログラマチック構成のいずれかを使用することができます。クライアント設定のオーバーライドについて詳しくは、[XML 構成を使用した Java クライアントの構成](#)および[Java クライアントのプログラマチック構成](#)を参照してください。

クライアント上で以下のプラグインをオーバーライドできます。

- **BackingMap** プラグイン
 - Evictor プラグイン
 - MapEventListener プラグイン
 - BackingMapLifecycleListener プラグイン
 - MapSerializerPlugin プラグイン
- **BackingMap** 属性
 - numberOfBuckets 属性

 **非推奨:** このプロパティは非推奨です。ニア・キャッシュを使用可能にするには nearCacheEnabled 属性を使用します。

 - timeToLive 属性
 - ttlEvictorType 属性
 - evictionTriggers 属性
 - nearCacheEnabled 属性
 - nearCacheInvalidationEnabled 属性
 - nearCacheLastAccessTTLSyncEnabled 属性
- **ObjectGrid** プラグイン
 - TransactionCallback プラグイン
 - ObjectGridEventListener プラグイン
 - ObjectGridLifecycleListener プラグイン
- **ObjectGrid** 属性
 - entityMetadataXMLFile 属性
 - txTimeout 属性
 - txIsolation 属性

親トピック: [Java クライアントの構成](#)

XML 構成を使用した Java クライアントの構成

ObjectGrid 構成 XML ファイルを使用して、クライアント・サイドの設定をオーバーライドできます。

このタスクについて

WebSphere® eXtreme Scale クライアントの設定を変更するには、コンテナ・サーバーに使用したファイルと似た構造の ObjectGrid XML ファイルを作成します。

クライアントでオーバーライドできるプラグインおよび属性のリストについては、[Java クライアントのオーバーライド](#)を参照してください。

手順

1. コンテナ・サーバー用のファイルと似た構造の ObjectGrid 構成 XML ファイルを クライアント用に作成します。

以下の XML ファイルがデプロイメント・ポリシー XML ファイルと対になっており、これらのファイルを使用してコンテナ・サーバーが開始されたものと想定します。

companyGridServerSide.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<objectGridConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://ibm.com/ws/objectgrid/config ../objectGrid.xsd"
  xmlns="http://ibm.com/ws/objectgrid/config">

  <objectGrids>
    <objectGrid name="CompanyGrid">
      <bean id="TransactionCallback"
        className="com.company.MyTxCallback" />
      <bean id="ObjectGridEventListener"
        className="com.company.MyOgEventListener" />
      <backingMap name="Customer"
        pluginCollectionRef="customerPlugins" />
      <backingMap name="Item" />
      <backingMap name="OrderLine" nearCacheEnabled="true"
        timeToLive="1600" ttlEvictorType="LAST_ACCESS_TIME" />
      <backingMap name="Order" lockStrategy="PESSIMISTIC"
        pluginCollectionRef="orderPlugins" />
    </objectGrid>
  </objectGrids>

  <backingMapPluginCollections>
    <backingMapPluginCollection id="customerPlugins">
      <bean id="Evictor"
        className="com.ibm.websphere.objectgrid.plugins.builtins.LRUEvictor"
      />
      <bean id="MapEventListener"
        className="com.company.MyMapEventListener" />
    </backingMapPluginCollection>
    <backingMapPluginCollection id="orderPlugins">
      <bean id="MapIndexPlugin"
        className="com.company.MyMapIndexPlugin" />
    </backingMapPluginCollection>
  </backingMapPluginCollections>
</objectGridConfig>
```

コンテナ・サーバーでは、CompanyGrid という名前の ObjectGrid インスタンスが companyGridServerSide.xml ファイルによる 定義に従って動作します。デフォルトでは CompanyGrid クライアントの設定は、サーバーで実行している CompanyGrid インスタンスの設定と同じです。

以下の ObjectGrid XML ファイルを使用すると、CompanyGrid クライアントの属性およびプラグインのいくつかを指定できます。

companyGridClientSide.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```



```

<objectGridConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://ibm.com/ws/objectgrid/config ../objectGrid.xsd"
  xmlns="http://ibm.com/ws/objectgrid/config">

  <objectGrids>
    <objectGrid name="CompanyGrid">
      <bean id="TransactionCallback"
        className="com.company.MyClientTxCallback" />
      <bean id="ObjectGridEventListener" className="" />
      <backingMap name="Customer" nearCacheEnabled="true"
        pluginCollectionRef="customerPlugins" />
      <backingMap name="Item" />
      <backingMap name="OrderLine" nearCacheEnabled="true"
        timeToLive="800" ttlEvictorType="LAST_ACCESS_TIME" />
      <backingMap name="Order" lockStrategy="PESSIMISTIC"
        pluginCollectionRef="orderPlugins" />
    </objectGrid>
  </objectGrids>

  <backingMapPluginCollections>
    <backingMapPluginCollection id="customerPlugins">
      <bean id="Evictor"
        className="com.ibm.websphere.objectgrid.plugins.builtins.LRUEvictor"
      />
      <bean id="MapEventListener" className="" />
    </backingMapPluginCollection>
    <backingMapPluginCollection id="orderPlugins">
      <bean id="MapIndexPlugin"
        className="com.company.MyMapIndexPlugin" />
    </backingMapPluginCollection>
  </backingMapPluginCollections>
</objectGridConfig>

```

XML ファイルでは、以下のオーバーライドを定義します。

- クライアントの TransactionCallback Bean は、サーバー・サイドで設定されている com.company.MyTxCallback ではなく、com.company.MyClientTxCallback になります。
- className 値が空ストリングであるため、クライアントには ObjectGridEventListener プラグインがありません。
- クライアントは、Customer backingMap のニア・キャッシュを使用可能にし、その Evictor プラグインを保持し、MapEventListener プラグインを除去します。
- OrderLine backingMap の timeToLive 属性は変更されます。
- 異なる lockStrategy 属性が指定されていても、lockStrategy 属性はクライアント・オーバーライドでサポートされていないため、影響はありません。

2. XML ファイルを使用してクライアントを作成します。

companyGridClientSide.xml ファイルを使用して CompanyGrid クライアントを作成するには、ObjectGrid XML ファイルを URL として、ObjectGridManager インターフェースのいずれかの connect メソッドに渡します。

```

ObjectGridManager ogManager =
    ObjectGridManagerFactory.ObjectGridManager();
ClientClusterContext clientClusterContext =
    ogManager.connect("MyServer1.company.com:2809", null, new URL(
        "file:xml/companyGridClientSide.xml"));

```

親トピック: [Java クライアントの構成](#)

Java クライアントのプログラマチック構成

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

このタスクについて

以下のコード例では、[XML 構成を使用した Java クライアントの構成](#)で説明されているのと同じオーバーライドを作成します。

クライアントでオーバーライドできるプラグインおよび属性のリストについては、[Java クライアントのオーバーライド](#)を参照してください。

手順

以下のコードでは、クライアント・サイドの ObjectGrid インスタンスを作成します。

```
ObjectGridConfiguration companyGridConfig = ObjectGridConfigFactory
    .createObjectGridConfiguration("CompanyGrid");
Plugin txCallbackPlugin = ObjectGridConfigFactory.createPlugin(
    PluginType.TRANSACTION_CALLBACK, "com.company.MyClientTxCallback");
companyGridConfig.addPlugin(txCallbackPlugin);

Plugin ogEventListenerPlugin = ObjectGridConfigFactory.createPlugin(
    PluginType.OBJECTGRID_EVENT_LISTENER, "");
companyGridConfig.addPlugin(ogEventListenerPlugin);

BackingMapConfiguration customerMapConfig = ObjectGridConfigFactory
    .createBackingMapConfiguration("Customer");
customerMapConfig.setNumberOfBuckets(1429);
Plugin evictorPlugin = ObjectGridConfigFactory.createPlugin(PluginType.EVICTOR,
    "com.ibm.websphere.objectgrid.plugins.builtins.LRUEvictor");
customerMapConfig.addPlugin(evictorPlugin);

companyGridConfig.addBackingMapConfiguration(customerMapConfig);

BackingMapConfiguration orderLineMapConfig = ObjectGridConfigFactory
    .createBackingMapConfiguration("OrderLine");
orderLineMapConfig.setNumberOfBuckets(701);
orderLineMapConfig.setTimeToLive(800);
orderLineMapConfig.setTtlEvictorType(TTLType.LAST_ACCESS_TIME);

companyGridConfig.addBackingMapConfiguration(orderLineMapConfig);

List ogConfigs = new ArrayList();
ogConfigs.add(companyGridConfig);

Map overrideMap = new HashMap();
overrideMap.put(CatalogServerProperties.DEFAULT_DOMAIN, ogConfigs);

ogManager.setOverrideObjectGridConfigurations(overrideMap);
ClientClusterContext client = ogManager.connect(catalogServerEndpoints, null, null);
ObjectGrid companyGrid = ogManager.getObjectGrid(client, objectGridName);
```

ObjectGridManager の ogManager インスタンスは、overrideMap マップに組み込まれている ObjectGridConfiguration オブジェクトおよび BackingMapConfiguration オブジェクトのオーバーライドのみをチェックします。例えば、上記のコードは、OrderLine マップ上のバケットの数をオーバーライドします。ただし、そのマップに対する構成が組み込まれていないため、クライアント・サイドの Order マップは変更されないままです。

親トピック: [Java クライアントの構成](#)

要求タイムアウト値および再試行タイムアウト値の構成

eXtreme Scale クライアント・コードが要求の完了をどれくらい待ち、このクライアント・コードがデータ・グリッドへのアクセスと関連付けられる要求をいつまで再試行するかを制御する調整オプションを指定することができます。

このタスクについて

eXtreme Scale クライアントに対して、クライアントがネットワーク接続の作成を試みる時間の長さ、クライアントが区画へのデータ・グリッド要求の処理を試みる時間の長さ、およびクライアントが (アプリケーションに例外を返す前に) 区画へのデータ・グリッド要求の再試行を試みる時間の長さを制御する設定を構成することができます。

XIO および ORB における要求タイムアウト値および再試行タイムアウト値を調整するための要因

一部の調整オプションでは、使用しているトランスポートが eXtremeIO (XIO) であるかオブジェクト・リクエスト・ブローカー (ORB) であるかによって、設定する値が異なります。これらのトランスポート・レベル調整オプションはクライアントとの対話に初期影響を及ぼします。これは、これらのオプションによって、トランスポートがネットワーク・ソケット接続を試みる時間、およびデータ・グリッド操作に類似した個々のリモート・プロシージャ・コール (RPC) が完了するのに認められる時間が調整されるからです。

これらの値を調整する際は、ご使用の環境でどのくらいの最大負荷状態および定常状態が許容できるかを考慮してください。これらの時間間隔をデフォルト値より極端に下に調整すると (例えば、要求再試行タイムアウトのデフォルト値は 30 秒です)、操作は早く失敗することがあります。以下の要因を考慮してください。

- ネットワーク待ち時間
- データベースのような外部リソースとのグリッド対話の結合
- ヒープ・サイズ、ヒープ使用量、およびガーベッジ・コレクション調整ポリシーの組み合わせから生じるガーベッジ・コレクション一時停止

要求タイムアウト値および再試行タイムアウト値を調整するための ORB 設定

ORB 用に以下のタイムアウト設定があります。

com.ibm.CORBA.ConnectionTimeout

ORB がリモート・ロケーションとのソケット接続の作成を試みる時間の長さを指定します。ORB は作成した接続をキャッシュに入れます。したがって、この操作は要求のたびに毎回行われるわけではありません。

com.ibm.CORBA.RequestTimeout

ORB が RPC の完了を待機する時間の長さを指定します。

com.ibm.CORBA.FragmentTimeout

詳しくは、IBM ORB 資料を参照してください。製品はこの値のデフォルト設定を提供します。

com.ibm.CORBA.LocateRequestTimeout

詳しくは、IBM ORB 資料を参照してください。製品はこの値のデフォルト設定を提供します。

RequestTimeout および ConnectionTimeout 設定を調整する際は、デフォルト推奨値に基づく調整が適切である場合があります。要求タイムアウトの長さをどれくらいにしたいかに基づいてこれらの設定を定義する場合は、これらの設定を同じ値にすることもできます。

次の調整レベルは requestRetryTimeout です。各トランスポート・タイプで、RPC が時間内に完了しなかったためにシステム例外をスローした後、データ・グリッドは requestRetryTimeout 設定によって定義される追加時間 (例えば、要求タイムアウトが 10 秒、再試行要求タイムアウトが 20 秒) を使用して、以下のアクションを完了する時間を指定することができます。

- フェイルオーバーのために区画が他のところにある場合に非同期でカタログ・サーバーに最新のルーティング・テーブルを要求する。
- 新しい経路を取り、要求を再試行するか、または試行を停止し、例外をアプリケーションにスローする。

requestRetryTimeout プロパティーはミリ秒単位で指定されます。ゼロより大きい値に設定すると、再試行可能な例外が起こったときに要求は再試行されます。値を 0 に設定すると、例外が起こったときに再試行なしで失敗します。デフォルトの動作を使用するには、このプロパティーを除去するか、値を -1 に設定します。

要求再試行タイムアウトを調整するための XIO 設定

XIO には以下の強化された設定があります。

- xioTimeout 設定は、XIO トランスポートがネットワーク・ソケット接続の確立を試みる時間の長さを決定します。
- ORB の LocateRequest 設定および FragmentTimeout 設定に相当するものではありません。
- requestRetryTimeout 設定は、RPC が成功しようと試みる最大時間を制御するもので、カタログ・サーバーから新しいルーティング情報を取得したときに RPC の再試行が行われる時間の長さを制御するのに役立ちます。XIO との相違は、XIO をトランスポートとして使用すると、多くの場合、XIO は、RPC が失敗しそうになったら ORB よりもずっと早く報告することです。この失敗は、requestRetryTimeout 値が満了するよりも早く起こります。

要求再試行タイムアウトを設定する方法

要求再試行タイムアウト値はクライアント・プロパティ・ファイルまたはセッション内で構成できます。セッションの値がクライアント・プロパティ設定に優先します。ゼロより大きい値に設定された場合、タイムアウト条件が満たされるか、永続的な障害が起こるまで、要求は試行されます。永続的な障害とは、DuplicateKeyException 例外などです。値ゼロはフェイル・ファースト・モード設定を表し、データ・グリッドは、どのようなタイプのトランザクションの後であっても、トランザクションを再試行しません。

トランザクション・タイムアウトと要求再試行タイムアウト

実行時は、トランザクション・タイムアウト値が要求再試行タイムアウト値と一緒に使用され、要求再試行タイムアウトがトランザクション・タイムアウトを超えないようにします。

トランザクションには 2 つのタイプがあります。自動コミット・トランザクションと明示的に begin メソッドと commit メソッドを使用するトランザクションです。再試行の有効な例外は、これら 2 つのタイプのトランザクション間で異なります。

- セッション内で呼び出されるトランザクションの場合は、ORB CORBA SystemException (XIO の場合は TransportException) および eXtreme Scale クライアント TargetNotAvailable 例外であれば、トランザクションが再試行されます。
- 自動コミット・トランザクションは、CORBA SystemException および eXtreme Scale クライアント可用性例外の場合に再試行されます。これらの例外には、ReplicationVotedToRollbackTransactionException、TargetNotAvailable、および AvailabilityException 例外が含まれます。

アプリケーション障害やその他の永続障害はすぐに再発するので、クライアントはトランザクションを再試行しません。このような永続障害には DuplicateKeyException や KeyNotFoundException 例外があります。例外の後にはトランザクションを再試行せず、すべての例外を返すようにするには、フェイル・ファースト設定を使用します。

クライアントがトランザクションを再試行する例外

- ReplicationVotedToRollbackTransactionException (自動コミットの場合のみ)
- TargetNotAvailable
- org.omg.CORBA.SystemException (この ORB システム例外の XIO 相当物は TransportException です。)
- AvailabilityException (自動コミットの場合のみ)
- LockTimeoutException (自動コミットの場合のみ)
- UnavailableServiceException (自動コミットの場合のみ)

トランザクションが再試行されない永続的な例外

- DuplicateKeyException
- KeyNotFoundException
- LoaderException
- TransactionAffinityException
- LockDeadlockException
- OptimisticCollisionException

手順

- 要求再試行タイムアウト値をクライアント・プロパティ・ファイル内に設定します。

クライアントの requestRetryTimeout 値を設定するには、[クライアント・プロパティ・ファイル](#)内の requestRetryTimeout プロパティを追加または変更します。クライアント・プロパティは、デフォルトでは objectGridClient.properties ファイルです。requestRetryTimeout プロパティはミリ秒単位で指定されます。ゼロより大きい値に設定すると、再試行可能な例外が起こったときに要求は再試行されます。値を 0 に設定すると、例外が起こったときに再試行なしで失敗します。デフォルトの動作を使用するには、このプロパティを除去するか、値を -1 に設定します。objectGridClient.properties ファイル内の値の例を次に示します。

```
requestRetryTimeout = 30000
```

requestRetryTimeout 値はミリ秒で指定されます。この例の場合、値が ObjectGrid インスタンスで使用されると、requestRetryTimeout 値は 30 秒です。

- 要求再試行タイムアウト値をプログラマチックに設定します。

クライアント・プロパティをプログラマチックに設定するには、まず最初にアプリケーションの適切な <ロケーション>にクライアント・プロパティ・ファイルを作成します。以下の例では、クライアント・プロパティ・ファイルは、前のセクションの objectGridClient.properties スニペットを参照します。ObjectGridManager インスタンスに接続した後、前述のようにクライアント・プロパティを設定します。その後、ObjectGrid インスタンスを取得すると、ファイルで定義したクライアント・プロパティがインスタンスに設定されています。クライアント・プロパティ・ファイルを変更する場合は、そのたびに新しい ObjectGrid インスタンスを明示的に取得してください。


```

ObjectGridManager manager = ObjectGridManagerFactory.getObjectGridManager();
String objectGridName = "testObjectGrid";
URL clientXML = null;
ClientClusterContext ccc = manager.connect("localhost:2809", null, clientXML);
File file = new File("<location>/objectGridClient.properties");
URL url = file.toURI().toURL();
ccc.setClientProperties(objectGridName, url);
ObjectGrid objectGrid = ogManager.getObjectGrid(ccc, objectGridName);

```

- セッション・コミット時に指定変更ファイルを設定します。

要求再試行タイムアウト をセッションに設定するか、requestRetryTimeout クライアント・プロパティをオーバーライドするには、Session インターフェースの setRequestRetryTimeout(long) メソッドを呼び出します。

```

Session sessionA = objectGrid.getSession();
sessionA.setRequestRetryTimeout(30000);
ObjectMap mapA = sessionA.getMap("payroll");
String key = "key:" + j;
mapA.insert(key, "valueA");

```

クライアント・プロパティ・ファイルに設定されている値に関係なく、このセッションでは現在 30000 ms または 30 秒という requestRetryTimeout 値が使用されています。セッション・インターフェースの詳細については、[セッションを使用したグリッド内データへのアクセス](#)を参照してください。

例

次の例を考えてみてください。この例では、短いタイムアウト値を設定した結果として、クライアントがサーバー上のネットワーク待ち時間、ガーベッジ・コレクション、一般競合を扱えるようになっています。requestRetryTimeout プロパティは 10 秒で、xioTimeout プロパティは ORB ConnectionTimeout 値 (5 秒) と一致します。

表 1. ORB トランスポート・タイプのデータ・グリッド構成と eXtremeIO トランスポート・タイプのデータ・グリッド構成

グリッド・タイプ	ORB	XIO
eXtreme Scale API に直接アクセスする Java™ または .NET クライアント・アプリケーション	<ul style="list-style-type: none"> • クライアント・アプリケーションの orb.properties ファイルを変更します。以下の値を設定します。 <ul style="list-style-type: none"> ◦ com.ibm.CORBA.RequestTimeout=5 ◦ com.ibm.CORBA.ConnectTimeout=5 ◦ com.ibm.CORBA.FragmentTimeout=5 ◦ com.ibm.CORBA.LocateRequestTimeout=5 <p>注: WebSphere Application Server では、ORB 設定の制御は、orb.properties ファイルではなくデプロイメント・マネージャーを使用していきます。</p> <ul style="list-style-type: none"> • クライアント・アプリケーションの objectGridClient.properties ファイルを requestRetryTimeout=7000 で変更します。 	<p>クライアント・アプリケーションの objectGridClient.properties ファイルを以下の値で変更します。</p> <ul style="list-style-type: none"> • xioRequestTimeout=50000。この値はミリ秒単位で、com.ibm.CORBA.RequestTimeout と等価です。 • xioTimeout=5。この値は秒単位で、com.ibm.CORBA.ConnectTimeout と等価です。 • requestRetryTimeout=70000。この値はミリ秒単位で、ORB トランスポートの場合にも使用されます。 • ORB の FragmentTimeout および LocateRequestTimeout には

		XIO 等値がありません。
HTTP セッション	<p>クライアント・プロパティ・ファイルを requestRetryTimeout=10000 でクラスパスに配置します。</p>	<p>クライアント・アプリケーションのクライアント・プロパティ・ファイルを以下の値で変更します。</p> <ul style="list-style-type: none"> • xioRequestTimeout=10000。この値はミリ秒単位で、com.ibm.CORBA.RequestTimeout と等価です。 • xioTimeout=5。この値は秒単位で、com.ibm.CORBA.ConnectTimeout と等価です。 • requestRetryTimeout=10000。この値はミリ秒単位で、ORB トラポートの場合にも使用されます。 • ORB の FragmentTimeout および LocateRequestTimeout には XIO 等値がありません。
動的キャッシュ	<p>キャッシュ・インスタンスに次のプロパティを設定します。</p> <ul style="list-style-type: none"> • com.ibm.websphere.xs.dynacache.request_retry_timeout_override=2000 <p>requestRetryTimeout 設定を、このキャッシュ・インスタンス・プロパティではなくクラスパス内のクライアント・プロパティと一緒に使用することができます (この設定がすべての動的キャッシュ・インスタンスで同じになるようにしたい場合)。</p>	

親トピック: [Java クライアントの構成](#)

データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成

セッション管理を使用し、WebSphere® Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、アプライアンス内のデータ・グリッドを使用してそれらのアプリケーションを管理するように Liberty プロファイルを構成することができます。

始める前に

このタスクには、セッション管理を必要とするアプリケーションをデプロイするための指示が含まれています。セッション管理を使用する Java アプリケーションを作成するには、[Java API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。

このタスクについて

WebSphere Application Server アプリケーションで、セッション管理にアプライアンスを使用する構成ができるように、Liberty プロファイルに対して同じ構成を実行することができます。

動的機能を持つ軽量のサーバーが必要な場合は、アプライアンスで Liberty プロファイルを使用する選択ができます。例えば、機能の単位であるフィーチャーを追加または削除することができ、それにより、特定のサーバーにロードされるランタイム環境の一部を制御することができます。Liberty プロファイルで、例えばセッションを管理するアプリケーションを実行すると、Liberty プロファイル フィーチャーの指定に使用するサーバー定義を作成することができ、これにより、サーバーとアプライアンス上のデータ・グリッドとのやり取りが制御されます。

手順

1. 次のコマンドを実行して、サーバー定義を作成します。

```
wlp_install_root/bin/server create server_name
```

2. サーバー定義の下で server.xml ファイルを見つけて、それを XML エディターで開きます。
3. セッション・アプリケーション、例えば *your_application.jar* を `/wlp/usr/servers/defaultServer/apps` ディレクトリーに配置します。
4. アプライアンスのモニター・コンソールを開始して、「データ・グリッド」 > 「セッション・グリッド」をクリックします。
5. アプライアンスに、`session` という名前のセッション・グリッドを作成します。
6. コマンド行で **JAVA_HOME** プロパティーをエクスポートします。必ず、Liberty プロファイルがインストールされているディレクトリーからコマンドを出してください。
7. 次のコマンドを使用して、Liberty プロファイルを開始します。

```
./server start servername
```

PID が表示されます。

8. 次の URL を使用してセッション・アプリケーションを開きます。

```
http://server:port/A/
```

9. データがアプライアンス内のセッション・グリッドに書き込まれていることを検証するには、セッション・テストを実行します。

SSL を有効に設定していないサーバー定義ファイル

以下に示す、SSL を有効に設定していない基本的な server.xml ファイルの例を参照してください。この例は、印刷の都合上、複数行で表示されています。

注: Web フィーチャーは推奨されません。代わりに webApp フィーチャーを使用してください。webApp フィーチャーをサーバー定義に追加し、セッション・マネージャーを構成した場合、Liberty プロファイルで実行される WebSphere® eXtreme Scale アプリケーションでセッション複製を使用できます。

次の例を参照してください。この例では、webApp フィーチャーを使用しています。

2.5+

```
<server description="new server">
  <!-- Enable features -->
  <featureManager>
```



```
<feature>jsp-2.2</feature>
<feature>eXtremeScale.server-1.1</feature>
<feature>eXtremeScale.webApp-1.1</feature>
</featureManager>

<httpEndpoint id="defaultHttpEndpoint"
  host="*"
  httpPort="9080"
  httpsPort="9443" />

<xsWebApp objectGridName="session" catalogHostPort="remoteHost:2809"
securityEnabled="false" />
<applicationMonitor updateTrigger="mbean"/>
<application id="A" location="A.ear" name="A" type="ear"/>
<httpSession cloneId="A_test"/>

<!-- <com.ibm.ws.xs.server.config catalogServer="true" /> -->
</server>
```

[SSL を有効にしてアクセスするデータ・グリッド用の Liberty プロファイルの構成](#)

WebSphere Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、HTTPS 用の Liberty プロファイルを構成することができ、これにより、保護された Web サーバーでは SSL とデータ暗号化が自動的に使用されます。

[クライアントと一緒に実行する Liberty プロファイルの構成](#)

WebSphere eXtreme Scale クライアント・フィーチャーを使用して、eXtreme Scale クライアントで Liberty プロファイルを実行します。

[Liberty プロファイルでの webApp フィーチャーの使用可能化](#)

Liberty プロファイル・サーバーは、フォールト・トレランス用に HTTP セッション・データを複製するためにアプリケーションのデータをキャッシュに入れるデータ・グリッドをホストできます。

[Liberty プロファイル xsDynacacheApp フィーチャー・プロパティ](#)

データ・グリッドをホストするための Liberty プロファイルを指定します。データ・グリッドは、このフィーチャーを使用して動的キャッシュ・プロバイダーとして構成することができます。

親トピック: [Java クライアントの構成](#)

SSL を有効にしてアクセスするデータ・グリッド用の Liberty プロファイルの構成

WebSphere® Application Server Liberty プロファイルにデプロイされるクライアント・アプリケーションがある場合は、HTTPS 用の Liberty プロファイルを構成することができ、これにより、保護された Web サーバーでは SSL とデータ暗号化が自動的に使用されます。

手順

1. 以下のコマンドを実行すると、SSL 証明書が作成されて HTTPS が使用可能に設定されます。

```
cd to lib_dir¥bin
securityUtility createSSLCertificate --server=defaultServer --password=xc10test
```

2. SSL データ暗号化を実行するように Liberty プロファイルを構成するには、以下の SSL 機能を server.xml ファイルに追加します。

2.5+

```
<featureManager>
  <feature>ssl-1.1</feature>
</featureManager>
<keyStore id="defaultKeyStore" password="{xor}MjowbTI+Kyw=" />
```

3. 次のコマンドを使用して Liberty プロファイルを開始します。

```
./server start servername
```

PID が表示されます。

4. 次の URL を使用してセッション・アプリケーションを開きます。

```
http://server:securedport/A/
```

5. データがアプライアンス内のセッション・グリッドに書き込まれていることを検証するには、セッション・テストを実行します。

SSL 対応のサーバー定義ファイルの例

印刷の都合上、一部のコードが複数行に表示されています。以下は、SSL 機能を使用する高度な server.xml ファイルの構成の例です。この例は、印刷の都合上、複数行で表示されています。

注: バージョン 2.5 以降、フィーチャー・バージョン番号が webApp-1.0 から webApp-1.1 に変わりました。

2.5+

```
<server description="new server">

  <!-- Enable features -->
  <featureManager>
    <feature>jsp-2.2</feature>
    <feature>eXtremeScale.server-1.1</feature>
    <feature>eXtremeScale.webApp-1.1</feature>
    <feature>ssl-1.1</feature>
  </featureManager>

  <httpEndpoint id="defaultHttpEndpoint"
    host="*"
    httpPort="9080"
    httpsPort="9443">
    <!--tcpOptions soReuseAddr="true" / -->
  </httpEndpoint>

  <keyStore id="defaultKeyStore" password="{xor}Jzxubys6LCs=" />

  <xsWebApp objectGridName="session" catalogHostPort="remoteHost:2809"
    securityEnabled="true"
    credentialGeneratorClass="com.ibm.websphere.objectgrid.security.plugins.builtins.UserPasswordCredentialGenerator"
    credentialGeneratorProps="xcadmin xcadmin"/>
```

```
<applicationMonitor updateTrigger="mbean"/>
<application id="A" location="A.ear" name="A" type="ear"/>
<httpSession cloneId="A_test"/>
```

```
</server>
```

次のタスク

Liberty プロファイルとデータ・グリッド・コンテナーの間の SSL 構成を設定するために、クライアント・プロパティ・ファイルでクライアント・トランスポート・タイプを指定します。可能な値は以下のとおりです。

- **TCP/IP:** クライアントが、TCP/IP 接続のみをサポートすることを示します。
- **SSL-Supported:** クライアントが TCP/IP 接続と Secure Sockets Layer (SSL) 接続の両方をサポートすることを示します。(デフォルト)
- **SSL-Required:** クライアントが SSL 接続を必要とすることを示します。

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

関連タスク:

[クライアント・セキュリティーの構成](#)

クライアントと一緒に実行する Liberty プロファイルの構成

WebSphere® eXtreme Scale クライアント・フィーチャーを使用して、eXtreme Scale クライアントで Liberty プロファイルを実行します。

始める前に

Liberty プロファイルを構成する前に以下のタスクを実行してください。

- [Liberty プロファイルおよび WebSphere eXtreme Scale をインストールします。](#)

このタスクについて

この構成では、クライアント機能のみが提供されます。サーバー機能は別のプロセスで実行されます。クライアント・フィーチャーを追加すると、アプリケーションは eXtreme Scale API にアクセスでき、またリモート・グリッドに接続できます。

このクライアント構成により、eXtreme Scale データ・グリッドを使用して Web アプリケーションの実行を行うために必要な機能が組み込まれた単一のプロセスが提供されます。クライアント・フィーチャーの追加後は、アプリケーションは eXtreme Scale API に書き込むことができます。

手順

クライアント・フィーチャーを Liberty プロファイル サーバーに追加します。以下のコードを Liberty プロファイル サーバーに追加します。

```
eXtremeScale.client-1.1
```

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

Liberty プロファイルでの webApp フィーチャーの使用可能化

Liberty プロファイル・サーバーは、フォールト・トレランス用に HTTP セッション・データを複製するためにアプリケーションのデータをキャッシュに入れるデータ・グリッドをホストできます。

このタスクについて

インストールした WebSphere® Application Server Liberty プロファイルには、セッション複製は含まれていません。ただし、Liberty プロファイルとともにデータ・グリッドを使用した場合は、セッションを複製できます。これにより、サーバーがダウンした場合に、アプリケーション・ユーザーのセッション・データが失われなくなります。

webApp フィーチャーをサーバー定義に追加し、セッション・マネージャーを構成した場合、Liberty プロファイルで実行されるデータ・グリッド・アプリケーションでセッション複製を使用できます。

手順

以下の webApp フィーチャーを Liberty プロファイル server.xml ファイルに追加します。webApp フィーチャーには、クライアント・フィーチャーが含まれていますが、サーバー・フィーチャーは含まれていません。恐らく、Web アプリケーションはデータ・グリッドから分離する必要があります。例えば、Web アプリケーション用に 1 つの Liberty プロファイルサーバーを使用し、データ・グリッドをホストするために別の Liberty プロファイルサーバーを使用します。

```
<featureManager>
<feature>eXtremeScale_webapp-1.1</feature>
</featureManager>
```

タスクの結果

Web アプリケーションは、データ・グリッドにセッション・データを永続化できるようになりました。

例

server.xml ファイルの以下の例を参照してください。これには、データ・グリッドにリモート側で接続する場合に使用する webApp フィーチャーが含まれています。

```
<server description="Airport Entry eXtremeScale Getting Started Client Web Server">
<!--
This sample program is provided AS IS and may be used, executed, copied and modified
without royalty payment by customer
(a) for its own instruction and study,
(b) in order to develop applications designed to run with an IBM WebSphere product,
either for customer's own internal use or for redistribution by customer, as part of such
an
application, in customer's own products.
Licensed Materials - Property of IBM
5724-X67, 5655-V66 (C) COPYRIGHT International Business Machines Corp. 2012
-->
<!-- Enable features -->
<featureManager>
<feature>eXtremeScale.webapp-1.1</feature>
</featureManager>

<httpEndpoint id="defaultHttpEndpoint"
host="*"
httpPort="${default.http.port}"
httpsPort="${default.https.port}" />

<xsWebApp objectGridName="session" catalogHostPort="remoteHost:2809"
securityEnabled="false" />

</server>
```

次のタスク

webApp フィーチャーには、server.xml ファイルの xsWebApp エレメントで設定できるメタ・タイプ・プロパティが含まれています。詳しくは、[Liberty プロファイル xsWebApp フィーチャー・プロパティ](#)を参照してください。

[Liberty プロファイル xsWebApp フィーチャー・プロパティ](#)


webApp フィーチャーを指定して、Liberty プロファイル Web アプリケーションを拡張します。フォールト・

トランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを追加してください。

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

Liberty プロファイル xsWebApp フィーチャー・プロパティ

webApp フィーチャーを指定して、Liberty プロファイル Web アプリケーションを拡張します。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを追加してください。

 Web フィーチャーは推奨されません。フォールト・トレランスのために HTTP セッション・データを複製するときは、webApp フィーチャーを使用してください。

server.xml ファイルの xsWebApp エレメントで以下の属性を設定することができます。

パラメーター

objectGridName

特定の Web アプリケーションで使用する ObjectGrid インスタンスの名前を定義する文字列値。デフォルトの名前は「session」です。

eXtreme Scale コンテナ・サーバーの始動に使用する ObjectGrid XML ファイルとデプロイメント XML ファイルの両方で、このプロパティは objectGridName を反映する必要があります。

catalogHostPort

カタログ・サーバーに接続して、クライアント・サイドの ObjectGrid インスタンスを取得できます。値の形式は、host:port<,host:port> でなければなりません。host は、カタログ・サーバーが実行されているリスナー・ホストです。port は、そのカタログ・サーバー・プロセスのリスナー・ポートです。このリストは任意の長さにすることができ、ブートストラッピングにのみ使用されます。最初の実行可能なアドレスが使用されます。このパラメーターは、**catalog.services.cluster** プロパティが構成されている場合は、WebSphere® Application Server 内でオプションです。

replicationInterval

更新されたセッションを ObjectGrid に書き込む時間間隔を秒数で定義する整数値。デフォルトは 10 秒です。0 から 60 までの値を設定できます。0 の場合は、更新されたセッションを ObjectGrid に書き込むのは、各要求のサーブレット・サービス・メソッド呼び出しの最後ということになります。replicationInterval 値が高いほどパフォーマンスは向上します。これは、データ・グリッドに書き込まれるアップデートの数が少ないためです。ただし、値が高いとそれだけ構成のフォールト・トレラント性が低くなります。

この設定は、objectGridType が REMOTE に設定されている場合のみ適用されます。

sessionTableSize

メモリー内に保持するセッション参照数を定義する整数値。デフォルトは 1000 です。

EMBEDDED トポロジーは既に Web コンテナと同じ層にセッション・データを持っているため、この設定は REMOTE トポロジーのみに関するものです。

セッションは、最長未使用時間 (LRU) ロジックに基づいて、メモリー内のテーブルから除去されます。メモリー内のテーブルから除去されたセッションは Web コンテナから無効化されます。しかし、データ自体はグリッドから削除されないため、そのセッションに対する後続の要求は引き続きデータを検索することができます。この値は、Web コンテナの最大スレッド・プール値よりも高く設定されなければなりません。そうすると、セッション・キャッシュでの競合を削減できます。

fragmentedSession

true または false のいずれかの文字列値。デフォルト値は true です。この設定を使用して、製品がセッション・データをエントリ全体として保管するか、それぞれの属性を個別に保管するかを制御します。

Web アプリケーションのセッションが持っている属性の数が多い場合や属性のサイズが大きい場合は、fragmentedSession パラメーターを true に設定してください。すべての属性はデータ・グリッド内の同じキーに保管されるため、セッションが持っている属性の数が多い場合は fragmentedSession を false に設定してください。

以前のフィルター・ベースの実装環境では、このプロパティは「persistenceMechanism」と呼ばれており、設定可能な値は ObjectGridStore (フラグメント化されている場合) および ObjectGridAtomicSessionStore (フラグメント化されていない場合) でした。

securityEnabled

true または false のいずれかの文字列値。デフォルト値は false です。この設定により、eXtreme Scale クライアント・セキュリティーを使用可能にすることができます。この設定は、eXtreme Scale サーバー・プロパティ・ファイルの **securityEnabled** 設定と一致していなければなりません。これらの設定が一致しないと、例外が発生します。

credentialAuthentication

資格情報の認証を実施するか、またはサポートするかを示します。

常になし

クライアント証明書認証は実施されません。

必須

資格情報の認証は必ず実施されます。サーバーが資格情報の認証をサポートしない場合、クライアントはサーバーに接続できません。

サポートされる

(デフォルト) 資格情報の認証は、クライアントとサーバーの両方が資格情報の認証をサポートする場合のみ実施されます。

authenticationRetryCount

`com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator` インターフェースを実装するクラスの名前を指定します。このクラスを使用して、クライアントの資格情報が取得されます。デフォルト値は 0 です。

credentialGeneratorClass

`com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator` インターフェースを実装するクラスの名前。このクラスを使用して、クライアントの資格情報が取得されます。

credentialGeneratorProps

`CredentialGenerator` 実装クラスのプロパティ。このプロパティが、`setProperties(String)` メソッドを使用してオブジェクトに設定されます。`credentialGeneratorProps` 値は、**credentialGeneratorClass** プロパティの値が非ヌルの場合にのみ使用されます。

shareSessionsAcrossWebApps

セッションが Web アプリケーション間で共有されるかどうかを指定します。`true` または `false` のいずれかのストリング値として指定されます。デフォルトは `false` です。サーブレット仕様では、HTTP セッションを Web アプリケーション間で共有できないとされています。この共有を可能にするため、サーブレット仕様の拡張が提供されます。

親トピック: [Liberty プロファイルでの webApp フィーチャーの使用可能化](#)

Liberty プロファイル xsDynacacheApp フィーチャー・プロパティ

データ・グリッドをホストするための Liberty プロファイルを指定します。データ・グリッドは、このフィーチャーを使用して動的キャッシュ・プロバイダーとして構成することができます。

server.xml ファイルの xsDynacacheApp エレメントで以下の属性を設定することができます。

パラメーター

gridName

特定の動的キャッシュ・インスタンスで使用するデータ・グリッド・インスタンスの名前を定義する文字列値。

cacheName

テンプレート・マップの名前として使用される固有の接尾部の名前を指定します。例えば、次のとおりです。

```
IBM_DC_PARTITIONED.cache_name
```

mapName

動的キャッシュ・プロバイダーであるデータ・グリッドのマップの名前を指定します。

remoteDomain

Liberty プロファイル内のデータ・グリッド動的キャッシュ・プロバイダーのリモートのクライアント・ドメイン名を指定します。

clientObjectgridXML

eXtreme Scale のクライアント objectgrid.xml ファイルのファイル・ロケーションを指定します。

requestRetryTimeout

要求がタイムアウトにならないで実行できる時間 (ミリ秒) を指定します。このプロパティは、クライアント・プロパティ・ファイルで **requestRetryTimeout** プロパティが設定されている場合、その値をオーバーライドします。動的キャッシュ・インスタンスのデフォルト値は 2000 ミリ秒です。

templateName

テンプレート・マップ接頭部の名前を指定します。以下のマップ・テンプレートのいずれかを使用できます。

- IBM_DC_PARTITIONED_.* (デフォルト)
- IBM_DC_NCI_PARTITIONED_.* (このキャッシュがニア・キャッシュを使用することを示します)

親トピック: [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)

REST ゲートウェイのデプロイ

WebSphere® Application Server または Liberty プロファイル・サーバーでデータ・グリッド用の REST ゲートウェイをデプロイして構成できます。

始める前に

Liberty プロファイル・サーバーが作成されていることを確認します。詳しくは、[Liberty プロファイルのインストール](#)を参照してください。

このタスクについて

REST ゲートウェイは、wxsRESTGateway.war Web アーカイブ (WAR) ファイルに定義されたサーブレットです。この REST ゲートウェイでは、Uniform Resource Identifier (URI) を使用して、データ・グリッド内のデータにアクセスします。

手順

1. server.xml ファイルを手動で編集するか、Liberty Profile Developer Tools を使用して、REST ゲートウェイ機能を使用可能にします。
 - Liberty プロファイル server.xml ファイルで REST ゲートウェイを使用可能にします。

```
<featureManager>
  <feature>eXtremeScale.rest-1.1</feature>
</featureManager>
```

- Liberty Profile Developer Tools を使用して、Liberty プロファイル server.xml ファイルで REST ゲートウェイを使用可能にします。
 - IBM® WebSphere Application Server バージョン 8.6 Liberty Profile Developer Tools を開始します。
 - 「設計」タブで、「フィーチャー・マネージャー」を選択します。「フィーチャー・マネージャーの詳細」セクションで「追加」をクリックします。「**eXtremeScale.rest-1.1**」フィーチャーを選択して追加します。
 - 選択したフィーチャー・マネージャーの「フィーチャー・マネージャーの詳細」セクションで「追加」をクリックします。「**servlet-3.0**」フィーチャーを選択して追加します。
 - server.xml ファイルを保存します。
 - WebSphere Application Server で REST ゲートウェイを使用可能にします。
 - WebSphere Application Server とともに WebSphere eXtreme Scale をインストールします。詳しくは、[WebSphere Application Server での WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
 - `was_install_root¥optionalLibraries¥ObjectGrid¥restgateway¥wxsRESTGateway.war` ファイルを WebSphere Application Server にデプロイします。
2. REST ゲートウェイを構成します。
 - a. server.xml ファイルで REST ゲートウェイを構成します。以下のコード行を入力します。

```
<xsREST contextRoot="myContextRoot" remoteDomain="myDomain"/>
```

重要: 属性 contextRoot および remoteDomain はオプションです。デフォルトのコンテンツ・ルートは resources です。

- b. eXtreme Scale サーバーを構成します。
- c. コンテナ・サービスを構成します。

コンテナ・サービスを構成するために以下のオプションが使用可能です。

- 有効な objectgrid.xml ファイルを (objectGridDeployment.xml ファイルとともに、またはこのファイルなしで) `wlp_home/usr/servers/server_name/grids` ディレクトリーにコピーします。この grids ディレクトリーは、実行時に製品によってモニターされます。このディレクトリー内のファイルが変更されると、Liberty プロファイル ランタイム環境でイベントが開始されます。例えば、新規 objectgrid.xml または objectGridDeployment.xml、あるいはその両方のファイルが見つかった場合は、新規コンテナ・サーバーが作成されます。これらのファイルの 1 つが削除されると、eXtreme Scale はそのコンテナ・サーバーを停止します。ファイルが変更されると、eXtreme Scale はコンテナを停止して再始動します。複数の断片コンテナが同じ eXtreme Scale 内に存在できます。この場合、grids ディレクトリー内にサブディレクトリーが存在する必要があります。

3. Liberty プロファイル・サーバーを始動して、REST クライアント・ゲートウェイを実行します。

次のタスク

REST ゲートウェイが使用可能になっている場合は、サーブレットに対するアクセス権を備えた任意のユーザーが、データ・グリッドのデータにアクセスできます。そのため、WebSphere Application Server で Web アプリケーション・セキュリティーを使用して、許可を制御する必要があります。この REST ゲートウェイを使用する Web アプリケーションの保護について詳しくは、WebSphere Application Server インフォメーション・センターの『[Web アプリケーションの保護](#)』を参照してください。

wxsRESTGateway.war ファイルは、ご使用のインストール済み環境に応じて、以下の場所にあります。このファイルには、セキュリティー構成用の web.xml ファイルが含まれています。

- `wlp_install_root/wxs/web/rest`
- `was_install_root/optionalLibraries/ObjectGrid/restgateway`
- `wxs_standalone_install_root/ObjectGrid/restgateway`

これで、Liberty プロファイルで REST データ・サービスの使用を開始して、URI を使用してデータ・グリッドと通信できるようになりました。詳しくは、[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。

親トピック: [Java クライアントの構成](#)

WebSphere eXtreme Scale Client for .NET の構成

WebSphere® eXtreme Scale Client for .NET を構成するには、クライアント・プロパティ・ファイルを使用するか、サーバー・サイド XML 構成を使用するか、または一部のサーバー・プロパティをプログラマチックにオーバーライドします。

このタスクについて

クライアントがデータ・グリッドへの Connect メソッドを呼び出すと、指定されたクライアント・プロパティ・ファイルを通じて構成が行われます。プロパティ・ファイルを指定しないと、Client.Net.properties ファイルが使用されます。

クライアントが Connect メソッドを呼び出した後でクライアント・プロパティ・ファイルに加えられた変更は、データ・グリッドへのクライアント接続を再始動して初めて有効になります。

Client.Net.properties ファイル内の一部のプロパティについては動的構成を構成することができます。

クライアント・プロパティ・ファイルに正しくないプロパティ値、正しくないクライアント・プロパティ・ロケーション、またはその他のエラーが含まれていると、Connect メソッドから例外が発生することがあります。また、WebSphere eXtreme Scale Client for .NET は、Client for .NET API を使用してプログラマチックにオーバーライドできるサーバー・サイド XML 構成プロパティ値をいくつか受け取ります。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET オーバーライド](#)

サーバー・プロパティをオーバーライドして WebSphere eXtreme Scale Client for .NET を構成することができます。これらのサーバー・プロパティは objectgrid.xml 構成ファイルを使用して設定されます。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET のプログラマチックな構成](#)

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

.NET 2.5+ [WebSphere eXtreme Scale Client for .NET の動的構成を使用可能に設定](#)

クライアント・プロパティ・ファイル内のプロパティ値に対して行われた変更を動的に検出するように WebSphere eXtreme Scale Client for .NET を構成することができます。データ・グリッドへの既存の接続を再始動しなくても変更が有効になります。

親トピック: [クライアントの構成](#)

.NET

WebSphere eXtreme Scale Client for .NET オーバーライド

サーバー・プロパティをオーバーライドして WebSphere® eXtreme Scale Client for .NET を構成することができます。これらのサーバー・プロパティは objectgrid.xml 構成ファイルを使用して設定されます。

クライアントの設定をオーバーライドするには、WebSphere eXtreme Scale Client for .NET でプログラマチック構成を使用することができます。プログラマチック・オーバーライドについて詳しくは、[WebSphere eXtreme Scale Client for .NET のプログラマチックな構成](#)を参照してください。

クライアントで以下の属性をオーバーライドすることができます。

BackingMap 属性

- timeToLive 属性
- lockTimeout 属性

ObjectGrid 属性

- txTimeout 属性
- txIsolation 属性

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成](#)

WebSphere eXtreme Scale Client for .NET のプログラマチックな構成

クライアント・サイドの設定をプログラマチックにオーバーライドすることができます。サーバー・サイド ObjectGrid インスタンスと同様の構造を持つ ObjectGridConfiguration オブジェクトを作成します。

このタスクについて

以下のコード例では、[WebSphere eXtreme Scale Client for .NET オーバーライド](#)で説明されているのと同じオーバーライドを作成します。

手順

以下のコードでは、クライアント・サイドの ObjectGrid インスタンスを作成します。

```
IGridManager gm = GridManagerFactory.GetGridManager();
ICatalogDomainInfo cdi =
gm.CatalogDomainManager.CreateCatalogDomainInfo("localhost:2809");
ctx = gm.Connect(cdi, "Sample.Client.Net.properties");
IGrid grid = gm.GetGrid( ctx, "Grid");

//Overriding grid's txTimeout value
grid.TransactionTimeout = new TimeSpan(0, 0, 30);

IGridMapPessimisticTxObject, Object gridMap;
gridMap = grid.GetGridMapPessimisticTx<Object, Object>("Map1");

//Overriding timeToLive value
gridMap.TimeToLive = new TimeSpan(0, 0, 50);

//Overriding lockTimeout value
gridMap.LockTimeout = new TimeSpan(0, 0, 20);

//Overriding txTimeout value
gridMap.Transaction.TransactionTimeout = new TimeSpan(0, 0, 40);

//Overriding txIsolation value
gridMap.Transaction.TransactionIsolationLevel = TxnIsolationLevel.ReadUncommitted;

//Calling ResetToDefaults(), resets the value of timeToLive, lockTimeout,
//txTimeout, txIsolation back to values when .NET client initialized the grid.

gridMap.ResetToDefaults();
```

プログラムで **timeToLive** 値をオーバーライドする前に、objectgrid.xml ファイル内の **ttlEvictorType** 値を LAST_ACCESS_TIME または LAST_UPDATE_TIME に設定します。WebSphere eXtreme Scale Client for .NET が **ttlEvictorType** 値を設定することなく **timeToLive** のプログラマチックなオーバーライドを試みると、例外がスローされます。TimeSpan.Zero に設定されている **txTimeout** 値は、タイムアウト制限が無いことを示しています。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成](#)

関連情報:

[IGridTransaction.TransactionTimeout プロパティ](#)

[IGridMapPessimisticAutoTx\(Of TKey, TValue\).LockTimeout プロパティ](#)

[IGridMapPessimisticAutoTx\(Of TKey, TValue\).TimeToLive プロパティ](#)

[IGridMapPessimisticTx\(Of TKey, TValue\).LockTimeout プロパティ](#)

[IGridMapPessimisticTx\(Of TKey, TValue\).TimeToLive プロパティ](#)

[IGrid.TransactionTimeout プロパティ](#)

WebSphere eXtreme Scale Client for .NET の動的構成を使用可能に設定

クライアント・プロパティ・ファイル内のプロパティ値に対して行われた変更を動的に検出するように WebSphere® eXtreme Scale Client for .NET を構成することができます。データ・グリッドへの既存の接続を再始動しなくても変更が有効になります。

このタスクについて

例外が発生した後いつまで (ミリ秒) 要求の処理を続行するかを指定するように、requestRetryTimeout プロパティを動的に構成することができます。他のプロパティはすべて読み取り専用です。読み取り専用プロパティの値を変更しても変更は有効になりません。システム・ログ・ファイルにエラーが書き込まれます。

手順

1. WebSphere eXtreme Scale Client for .NET のクライアント・プロパティ・ファイルにある enableDynamicConfiguration プロパティの値を true に設定します。

.NET 2.5+ enableDynamicConfiguration

これが true に設定されると、クライアント・プロパティ・ファイル内の requestRetryTimeout プロパティに対して行われた変更が動的に検出されます。新しい要求再試行タイムアウト値を計算するために新しいプロパティ値がただちに使用されます。

デフォルト: false

2. クライアント・プロパティ・ファイルにある requestRetryTimeout プロパティの値を更新します。

Java .NET requestRetryTimeout

例外が発生してから要求の処理をいつまで続行するかをミリ秒で指定します。以下の有効値のいずれかを使用します。

- 値が 0 の場合、要求は、フェイル・ファーストになり、内部の再試行ロジックは飛ばされます。
- 値が -1 の場合、要求再試行のタイムアウトは設定されません。すなわち、要求の所要時間は、トランザクション・タイムアウトによって制御されます。(デフォルト)
- 0 より大きい値は、要求再試行のタイムアウト値をミリ秒で示しています。正常に作成されない例外が返されます。DuplicateException などの例外が再試行されるときでも、成功しないときはその例外も返されます。トランザクション・タイムアウトは、最大待ち時間として引き続き使用されます。

タスクの結果

requestRetryTimeout クライアント・プロパティの値は、データ・グリッドへの接続を再始動しなくても動的に更新することができます。

親トピック: [.NET 2.5+ WebSphere eXtreme Scale Client for .NET の構成](#)

データ・グリッドの管理

コンソール、コマンド行、および HTTP コマンド・インターフェースを使用して、データ・グリッドを管理できます。

データの照会、表示、および無効化

モニター・コンソールおよび **xscmd** ユーティリティの QUERY インターフェースを使用して、マップから小規模なキーと値のセットを取得し、一連のデータを無効化することができます。

xscmd ユーティリティによる管理

xscmd ユーティリティを使用して、などの管理タスクを環境内で実行することができます。

HTTP コマンド・インターフェースでの管理

HTTP コマンド・インターフェースを使用して、アプライアンスでの操作の実行、アプライアンス設定の構成、およびデータ・グリッド、集合、ゾーンの管理を行えます。

データの照会、表示、および無効化

モニター・コンソールおよび **xscmd** ユーティリティの QUERY インターフェースを使用して、マップから小規模なキーと値のセットを取得し、一連のデータを無効化することができます。

始める前に

- **xscmd** を使用してデータの照会、表示、および無効化を行う場合は、**xscmd** ユーティリティをセットアップします。詳しくは、[xscmd ユーティリティによる管理](#)を参照してください。

このタスクについて

データ・グリッドの内容を照会するには、コンソールまたは **xscmd** ユーティリティを使用できます。データ・キーで正規表現を実行することで、データを照会できます。その後、同じ照会を使用してデータを無効にすることができます。正規表現の例については、[正規表現の構文](#)を参照してください。

手順

- コンソールでデータの照会、表示、または無効化を行います。
 1. コンソールで、照会ページに進みます。ユーザー・インターフェースで、「データ管理」 > 「データ・グリッド内容の照会」とクリックします。
 2. マップ内のデータを検索またはフィルター処理します。データの検索またはフィルター処理を行うには、以下のいずれかのオプションを使用します。
 - フィールドに正規表現を入力し、「検索」ボタン (🔍) をクリックします。その正規表現に一致するキーのリストが表示されます。このデータのリストは、一致するすべてのデータのサブセットである場合があります。
 - パーティション・セットで結果のフィルター処理をするには、「フィルター」ボタン (🔍) をクリックします。次に正規表現を入力して、結果のフィルター処理をするパーティションの範囲を選択します。
 3. 表示されているキーの値を表示します。「値の表示 (Show values)」を選択します。値が表に表示されます。値が長すぎて表示できない場合は、省略符号 (...) で値が切り捨てられます。完全フィールドを表示する値をクリックします。値は、テキスト・ストリングとして返されます。一部の値は人間が理解できるストリングには変換されないことがあり、16 進数が表示されます。

重要: アプリケーションは、Java™ クラスがサーバーにとって不明なオブジェクト値を保管することがあります。アプリケーションが eXtreme Data Format (XDF) を使用している場合は、これらの値が表示されます。XDF が使用されておらず、Java クラスがサーバーにとって不明な場合は、オブジェクトのクラスがサーバーで使用できなかったというメッセージが返されます。
 4. データを無効化します。データを無効化すると、データは、データ・グリッドから永続的に削除されます。

選択したキー

無効化するキーをテーブルから選択することができます。エントリーを個々にクリックすることも、あるいは、「すべて選択」チェック・ボックスをクリックする (これによりテーブル内の最大 500 個のエントリーが選択される) こともできます。削除したいエントリーを選択したら、「無効化」 > 「選択したキー」とクリックします。

照会に一致するすべてのキー

指定した正規表現に一致するすべてのデータを無効化することもできます。このオプションを使用すると、コンソールに表示される最大 500 のエントリーだけでなく、正規表現に一致する、データ・グリッド内のすべてのデータが削除されます。正規表現を使用して選択したエントリーを無効化するには、「無効化」 > 「照会に一致するすべてのキー」とクリックします。

5. マップのコンテンツ全体を削除します。「マップをクリア」をクリックします。選択したマップ内のすべての項目を削除することを確認する必要があります。
- **xscmd** ユーティリティでデータの照会、表示、または無効化を行います。

データの照会:

```
xscmd.sh -c findbykey -g <data_grid> -m <map>
-fs <find_string> [-fp <partitionid>]
```

検索ストリング値のデータ・グリッド、マップ、および正規表現を含める必要があります。区画 ID でフィルターリングすることもできます。結果は照会全体のサブセットを返します。

データの無効化:

照会によって選択されたデータを無効化するには、コマンドに **-inv** 引数を含めます。


```
xscmd -c findbykey -g <data_grid> -m <map>
-fs <find_string> [-fp <partitionid>] -inv
```

検索STRING値のデータ・グリッド、マップ、および正規表現を含める必要があります。区画 ID でフィルタリングすることもできます。無効化を実行する際、照会によって返される小規模なセットだけでなく、一致するすべての値が無効化されます。

以下のように、照会されたデータの値を表示します。

コマンドに **-rv** 引数を含めて、照会で選択されたデータの値を表示します。

```
xscmd.sh -c findbykey -g <data_grid> -m <map>
-fs <find_string> -rv
```

検索STRING値のデータ・グリッド、マップ、および正規表現を含める必要があります。区画 ID でフィルタリングすることもできます。結果では、照会全体のサブセットが返され、各キーの値が含まれています。

UNIX | **Linux** **重要:** 使用する正規表現が文字 `.*` で始まる場合、コマンドを実行する際にその文字は正常に処理されない可能性があります。この問題を解決するには、ご使用の正規表現を次のいずれかの方法でフォーマットしてください。

- 正規表現をアポストロフィ文字で囲みます: `-fs '.*'`
- 円記号 (`¥`) を使用してアスタリスク文字をエスケープします: `-fs .¥*`

例:

次の例は、Grid データ・グリッドと Map1 マップにあるすべてのエントリーを検索します。

```
xscmd -c findbykey -g Grid -m Map1 -fs ".*"
```

このコマンドは、次の結果を戻します。

```
3 matching keys were found.

Partition Key
-----
2          keyghi
4          keydef
6          keyabc
```

親トピック: [データ・グリッドの管理](#)

関連タスク:

[xscmd ユーティリティーによる管理](#)

関連資料:

[xscmd ユーティリティー・リファレンス](#)

xscmd ユーティリティーによる管理

xscmd ユーティリティーを使用して、などの管理タスクを環境内で実行することができます。

始める前に

- アプライアンスを開始する必要があります。
- クライアント・インストールから **xscmd** ユーティリティーを実行している場合: アクティブなカタログ・サーバーの IP アドレスとポート番号が必要です。ユーザー・インターフェースで、「集合」 > 「メンバー」をクリックします。集合メンバーを選択します。カタログ・サーバーの IP とポート番号が表示されます。
- クライアント・インストールから **xscmd** ユーティリティーを実行している場合: `JAVA_HOME`環境変数が、製品と一緒にインストールされたランタイム環境を使用するように設定されているか、確認してください。製品の試用版を使用している場合は、`JAVA_HOME` 環境変数を設定する必要があります。

このタスクについて

2.5+ **xscmd** ユーティリティーは、クライアント・インストールまたはアプライアンス・コマンド行インターフェースから実行できます。アプライアンス・コマンド行インターフェースから **xscmd** ユーティリティーを実行すると、ユーティリティーは集合内のカタログ・サーバーに自動的に接続します。**xscmd** ユーティリティーをアプライアンス・コマンド行インターフェースから実行するときは、環境変数の設定とトラストストアのインポートは不要です。アプライアンス・コマンド行インターフェースを使用するときは、ローカル集合にしか接続できません。リモート集合に接続するには、クライアント・インストールを使用するか、またはリモート集合内のアプライアンスの 1 つでアプライアンス・コマンド行インターフェースを使用することができます。

手順

1. クライアント・インストールから **xscmd** ユーティリティーを実行している場合: アプライアンス用のアクティブ・トラストストアをクライアントにダウンロードしてください。アプライアンス・ユーザー・インターフェースで、「集合」 > 「設定」 > 「**Transport Layer Security**」 > 「アクティブなトラストストアのダウンロード」をクリックします。デフォルトのトラストストア・ファイルは、`xsatruststore.jks` ファイルです。このファイルのデフォルトのパスワードは、`xc10pass` です。
2. オプション: クライアント認証が使用可能な場合: コマンド行ウィンドウをクライアントのインストール済み環境で開きます。コマンド行で、適切な環境変数を設定します。
3. **xscmd** ユーティリティーをアプライアンスに接続します。
 - クライアント・インストールから **xscmd** ユーティリティーを実行している場合:

クライアントのインストール済み環境の `bin` ディレクトリーから、以下のコマンドを実行します。

```
xscmd.bat|sh -ts xsatruststore.jks -tst jks -tsp xc10pass -user xadmin -pwd xadmin -cep myxc10.mycompany.com -prot TLS -cxpv IBMJSSE2 -tt TCP/IP [additional parameters]
```

- **2.5+** アプライアンス・コマンド行インターフェースから **xscmd** ユーティリティーを実行する場合:
 - a. コマンド行インターフェースに接続します。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
 - b. **xscmd** ユーティリティーを実行します。

コマンドの一般構文は以下のとおりです。

```
Console> xscmd -c <command_name> -opt1 [arg1] -opt2 [arg2] -opt3
```

次のコマンドはアプライアンスのヘルプを表示します。

```
Console> xscmd -h
```

4. さまざまな **xscmd** オプションのヘルプを表示します。アプライアンス・コマンド行インターフェースから **xscmd** ユーティリティーを実行する場合、`.bat|.sh` 拡張子は不要です。
 - 一般ヘルプを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -h`
 - **Windows** `xscmd.bat -h`
 - すべてのコマンドのリストを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -lc`
 - **Windows** `xscmd.bat -lc`
 - 特定のコマンドのヘルプを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -h command_name`
 - **Windows** `xscmd.bat -h command_name`

- コマンド・グループのリストを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -lcg`
 - **Windows** `xscmd.bat -lcg`
 - コマンド・グループ内のコマンドのリストを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -lc command_group_name`
 - **Windows** `xscmd.bat -lc command_group_name`
5. 特定のカタログ・サーバーに接続するコマンドを実行します。 アプライアンス上で実行されるデータ・グリッドに関する情報を取得するには、カタログ・サーバーの IP アドレスとポートの組み合わせを 1 つ以上指定する必要があります。アプライアンス・コマンド行インターフェースを使用するときは、ローカル集合にしか接続できません。 リモート集合に接続するには、クライアント・インストールを使用するか、またはリモート集合内のアプライアンスの 1 つでアプライアンス・コマンド行インターフェースを使用することができます。
- 接続するのカタログ・サーバーのリストを指定します。
 - **UNIX** `./xscmd.sh -c <command_name> -cep hostname:port(,hostname:port)`
 - **Windows** `xscmd.bat -c <command_name> -cep hostname:port(,hostname:port)`
 上記のコマンドで、*command_name* は実行しようとしているコマンドの名前です。*hostname:port* 値は、カタログ・サーバーのホスト名とリスナー・ポートです。

注意:

WebSphere® DataPower® XC10 アプライアンス 環境では、次のコマンドは使用しないでください。

- **-c releaseShard**
 - **-c reserveShard**
 - **-c swapShardWithPrimary**
 - **-c suspendBalancing**
 - **-c resumeBalancing**
 - **-c teardown**
 - **-c triggerPlacement**
 - **-c enableForPlacement**
6. オプション: コマンドの実行時にタイムアウト値を設定します。 任意のコマンドでグローバル・パラメーターとして **-to** または **--timeout** オプションを使用できます。この値は、コマンドでカタログ・サーバーに接続しているときにタイムアウトになるまでの秒数を指定します。オペレーティング・システムまたは他のネットワークのタイムアウトが原因で使用可能でないことがあるカタログ・サーバーに接続している場合は、このオプションを使用すると、制御した時間に待機時間を減らすために役に立つ場合があります。

デフォルトのタイムアウト値は 30 秒に設定されています。

[xscmd ユーティリティーのためのセキュリティー・プロファイルの構成](#)

セキュリティー・プロファイルを作成すると、保存されたセキュリティー・パラメーターを使用して、セキュアな環境で **xscmd** ユーティリティーを使用できます。

親トピック: [データ・グリッドの管理](#)

関連タスク:

[データの照会、表示、および無効化](#)

関連資料:

[xscmd ユーティリティー・リファレンス](#)

xscmd ユーティリティーのためのセキュリティー・プロファイルの構成

セキュリティー・プロファイルを作成すると、保存されたセキュリティー・パラメーターを使用して、セキュアな環境で **xscmd** ユーティリティーを使用できます。

始める前に

xscmd ユーティリティーのセットアップについては、[xscmd ユーティリティーによる管理](#)を参照してください。

このタスクについて

xscmd コマンドの残り部分で **-ssp profile_name** または **--saveSecProfile profile_name** パラメーターを使用して、セキュリティー・プロファイルを保存できます。プロファイルには、ユーザー名とパスワード、資格情報生成プログラム、鍵ストア、トラストストア、およびトランスポート・タイプについての設定を含めることができます。

xscmd ユーティリティー内の **ProfileManagement** コマンド・グループには、セキュリティー・プロファイルの管理のためのコマンドが含まれます。

手順

- セキュリティー・プロファイルを保存します。

セキュリティー・プロファイルを保存するには、コマンドの残り部分で **-ssp profile_name** または **--saveSecProfile profile_name** パラメーターを使用します。このパラメーターをコマンドに追加すると、次のパラメーターが保存されます。

```
-al,--alias <alias>
-arc,--authRetryCount <integer>
-ca,--credAuth <support>
-cgc,--credGenClass <className>
-cgp,--credGenProps <property>
-cxpv,--contextProvider <provider>
-ks,--keyStore <filePath>
-ksp,--keyStorePassword <password>
-kst,--keyStoreType <type>
-prot,--protocol <protocol>
-pwd,--password <password>
-ts,--trustStore <filePath>
-tsp,--trustStorePassword <password>
-tst,--trustStoreType <type>
-tt,--transportType <type>
-user,--username <username>
```

セキュリティー・プロファイルは、`user_home¥.xscmd¥profiles¥security ¥<profile_name>.properties` ディレクトリーに保存されます。

重要: `profile_name` パラメーターでは、ファイル名拡張子 `.properties` を含めないようにしてください。この拡張子は自動的にファイル名に付加されます。

- 保存したセキュリティー・プロファイルを使用します。

保存したセキュリティー・プロファイルを使用するには、実行するコマンドに、**-sp profile_name** または **--securityProfile profile_name** パラメーターを追加します。

コマンド例: `xscmd -c listHosts -cep myhost.mycompany.com -sp myprofile`

- **ProfileManagement** コマンド・グループの中のコマンドをリストします。

次のコマンドを実行します。 **xscmd -lc ProfileManagement**

- 既存のセキュリティー・プロファイルをリストします。

次のコマンドを実行します。 **xscmd -c listProfiles -v**

- セキュリティー・プロファイルの中に保存されている設定を表示します。

次のコマンドを実行します。 **xscmd -c showProfile -pn profile_name**

- 既存のセキュリティー・プロファイルを削除します。

次のコマンドを実行します。 **xscmd -c RemoveProfile -pn profile_name**

親トピック: [xscmd ユーティリティーによる管理](#)

関連資料:

[xscmd ユーティリティー・リファレンス](#)

HTTP コマンド・インターフェースでの管理


HTTP コマンド・インターフェースを使用して、アプライアンスでの操作の実行、アプライアンス設定の構成、およびデータ・グリッド、集合、ゾーンの管理を行えます。

このタスクについて

HTTP コマンド・インターフェースを使用して、HTTP POST JSON ステートメントで操作を実行することができます。これらのステートメントを結合してスクリプトにし、構成タスクや管理タスクを自動化することも可能です。

手順

1. ユーザー・インターフェースの HTTP コマンド・インターフェースで使用可能なコマンドを表示します。

ユーザー・インターフェースで使用可能な全コマンドの表を表示するには、 (ヘルプ) > 「**HTTP コマンド・インターフェースのヘルプ**」をクリックします。コマンド名をクリックすると、「**使用法の詳細**」と「**JSON の実行依頼例**」を表示できます。

インフォメーション・センター内の全コマンドのリストを表示するには、[HTTP コマンド・インターフェースのリファレンス](#)を参照してください。

2. 実行する操作の JSON ステートメントを作成します。このステートメントには次の内容を含める必要があります。

- 実行するコマンド。
- コマンドに適切なパラメーター。

cURL ツールで次のコマンドを使用します。

```
curl -v -k -u XC_ADMIN_USER:XC_ADMIN_PWD -H "Content-Type: application/json" --data-binary 'HTTP_INTERFACE_JSON_STATEMENT' https://XC10_HOST_NAME/resources/appTaskInterface
```

以下の変数を定義します。

XC_ADMIN_USER:XC_ADMIN_PWD

WebSphere® DataPower® XC10 アプライアンス 管理者のユーザー名およびパスワードを指定します。

HTTP_INTERFACE_JSON_STATEMENT

選択可能な JSON ステートメントのいずれかを指定します。これらは、HTTP コマンド・インターフェースに対して実行依頼されます。実行したい特定のコマンドの「**HTTP コマンド・インターフェースのヘルプ**」から「**JSON の実行依頼**」ステートメントの例をコピーすることができます。コマンドは単一引用符 (') で囲む必要があります。

XC10_HOST_NAME

WebSphere DataPower XC10 アプライアンス の完全修飾ホスト名または IP アドレスを指定します。

例えば、「HTTP コマンド・インターフェースのヘルプ」から ViewAllUsers コマンドの JSON の実行依頼例をコピーすることができます。その後、次のコマンドを実行することができます。

```
curl -v -k -u xcadmin:xcadmin -H "Content-Type: application/json" --data-binary '{"task":{"stopOnTaskFailure":"true","command":"ViewAllUsers"}}' https://myXC10.mycompany.com/resources/appTaskInterface
```

3. コマンドを実行して出力を表示します。

親トピック: [データ・グリッドの管理](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

関連情報:

[cURL ツールおよび libcurl ツール](#)

[HTTP コマンド・インターフェースのリファレンス](#)

単純データ・グリッドにアクセスするアプリケーションの開発

マップの作成や単純データ・グリッドの読み取り、書き込み、および更新には、2つの異なるオプションを使用できます。ObjectMap API を使用する Java クライアント・アプリケーションを作成できます。また、REST ゲートウェイを使用して、単純データ・グリッドにアクセスする非 Java アプリケーションを開発することもできます。

始める前に

既存の単純データ・グリッドが必要です。単純データ・グリッドの作成については、[単純データ・グリッドの作成](#)を参照してください。

動的キャッシュまたはセッション・データ・グリッドを使用する場合、既存のアプリケーションを変更せずに使用できます。

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

カタログ・サーバーに接続したり、ObjectGrid インスタンスを取得したり、ObjectMap API を使用したりすることができます。

[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

Representational State Transfer (REST) ゲートウェイを使用して、集合がホスティングする単純データ・グリッドにアクセスできます。この REST ゲートウェイは、非 Java 環境からデータ・グリッドにアクセスする必要があるときに便利です。

.NET 2.5+ [.NET API を使用したデータ・グリッド・アプリケーションの開発](#)

Java™ アプリケーションと同じデータ・グリッドを使用する Microsoft .NET アプリケーションを開発できます。

Java API を使用したデータ・グリッド・アプリケーションの開発

カタログ・サーバーに接続したり、ObjectGrid インスタンスを取得したり、ObjectMap API を使用したりすることができます。

開始する前に

ユーザー・インターフェースで単純データ・グリッドを作成する必要があります。詳しくは、[単純データ・グリッドの作成](#)を参照してください。

サンプル・データ・グリッド・アプリケーション・ウォークスルー

1. **ClientClusterContext** インスタンスを取得することによって、カタログ・サービスに 接続します。

指定すべきカタログ・サーバーが、作成した単純データ・グリッドのユーザー・インターフェースのページに表示されます。「データ・グリッド」 > 「単純データ・グリッド」 > 「**my_simple_data_grid**」をクリックし、「カタログ・サービス」フィールドにある値を使用します。

カタログ・サーバーに 接続するには、ObjectGridManager API の connect メソッドを使用します。使用する connect メソッドが必要とするのは、*hostname:port* という形式のカタログ・サーバー・エンドポイントのみです。*hostname:port* 値のリストをコンマで区切って、複数のカタログ・サーバー・エンドポイントを示すことができます。以下のコード・スニペットは、カタログ・サーバーへの接続方法 と ClientClusterContext インスタンスの取得方法を示します。

```
ClientClusterContext ccc =
ObjectGridManagerFactory.getObjectGridManager().connect("myXC10.myhost.com:2809",
null, null);
```

connect メソッドは、接続が成功するまで リスト内の各アプライアンスに接続しようとします。他のアプライアンスの 1 つが応答しなければ、自動フェイルオーバーが提供されます。

カタログ・サーバーへの 接続が成功すれば、connect メソッド は ClientClusterContext インスタンスを戻します。この ClientClusterContext インスタンス は、ObjectGridManager API から ObjectGrid を取得するのに必要です。

2. **ObjectGrid** インスタンスを取得します。

ObjectGrid インスタンスを 取得するには、ObjectGridManager API の getObjectGrid メソッドを使用します。getObjectGrid メソッドは、ClientClusterContext インスタンスと、データ・グリッド・インスタンスの名前との両方を必要とします。ClientClusterContext インスタンスは、カタログ・サーバーへの接続中に取得されます。データ・グリッド・インスタンスの名前は、ユーザー・インターフェースで作成した単純データ・グリッドの名前です。以下のコード・スニペットは、ObjectGridManager API の getObjectGrid メソッドを呼び出すことによってデータ・グリッドを取得する方法を示します。

```
ObjectGrid grid = ObjectGridManagerFactory.getObjectGridManager().getObjectGrid(ccc,
"my_simple_data_grid");
```

3. 必要なセキュリティー資格情報を設定します。

クライアント・セキュリティー構成を作成するほか、アプリケーションに提供するユーザー名とパスワードを使用して資格情報生成プログラムを作成します。使用するユーザー名とパスワードには、アプライアンス上のデータ・グリッドにアクセスするための 権限が必要です。権限があるユーザーの作成について詳しくは、[ユーザーおよびグループの管理](#)を参照してください。

```
file // Creates a ClientSecurityConfiguration object using the specified
ClientSecurityConfiguration clientSC =
ClientSecurityConfigurationFactory.getClientSecurityConfiguration();
clientSC.setSecurityEnabled(true);
// Creates a CredentialGenerator using the passed-in user and
password.
CredentialGenerator credGen = new
UserPasswordCredentialGenerator(username,password);
clientSC.setCredentialGenerator(credGen);
return clientSC;
```

4. **Session** インスタンスを取得します。

取得した ObjectGrid インスタンス から、Session を取得することができます。Session インスタンスは、ObjectMap インスタンスの取得とトランザクション区分の実行のために必要です。以下のコード・スニペットは、ObjectGrid API の getSession メソッドを 呼び出すことによって Session インスタンスを取得する方法を示します。

```
Session sess = grid.getSession();
```

5. ObjectMap インスタンスを取得します。

Session を取得した後、Session API の getMap メソッドを呼び出すことによって、Session インスタンスから ObjectMap インスタンスを取得することができます。getMap メソッドに渡す マップ・インスタンス名は、ユーザー・インターフェースで作成したデータ・グリッドと同じ名前です。以下のコード・スニペットは、Session API の getMap メソッドを呼び出すことによって ObjectMap を取得する方法を示します。

```
ObjectMap map1 = sess.getMap("my_simple_data_grid");
```

前の例は、データ・グリッドにちなんで名前が付けられるデフォルト・マップ・インスタンスを使用しています。次の例のように、新規マップ名を指定することもできます。

```
ObjectMap map2 = sess.getMap("my_simple_data_grid.CT.P");  
ObjectMap map3 = sess.getMap("my_new_map.NONE");
```

my_simple_data_grid.CT.P マップは作成時除去と ペシミスティック・ロックを使用するマップです。my_new_map.NONE マップには、除去もロック設定もありません。詳しくは、[動的マップの構成オプション](#)を参照してください。

6. ObjectMap メソッドを使用します。

ObjectMap インスタンスを取得した後、ObjectMap API を使用できます。ObjectMap インターフェースはトランザクション・マップであり、Session API の begin メソッドおよび commit メソッドを使用することによるトランザクション区分を必要とすることに注意してください。アプリケーションに明示的なトランザクション区分がない場合、ObjectMap 操作は自動コミット・トランザクションで実行します。

以下のコード・スニペットは、自動コミット・トランザクションでの ObjectMap API の使用方法を示しています。

```
map1.insert(key1, value1);
```

使用する鍵は、java.lang.String または Integer といった既存の Java 型にできます。値は、任意の直列化可能オブジェクト・タイプから構成できます。

以下のコード・スニペットは、明示的なトランザクション区分がある場合の ObjectMap API の使用方法を示しています。

```
sess.begin();  
map1.insert(key1, value1);  
sess.commit();
```

[Java API 資料へのアクセス](#)

WebSphere DataPower XC10 アプライアンス用の Java™ API 資料は、zip ファイルのアーカイブをダウンロードしてご使用の開発環境に取り込んで利用するか、インフォメーション・センターで閲覧できます。

[Java 例: 単純データ・グリッド \(data grid\) ・アプリケーション](#)

この例では、ObjectMap API を使用して、データ・グリッド (data grid) に対する単純な作成、取得、更新、および削除操作を実行します。

[Java API を使用した動的マップの作成](#)

データ・グリッドがインスタンス化されたら、Java API を使用して動的マップを作成できます。マップ・テンプレートの事前定義セットに基づいて、動的にマップをインスタンス化することができます。

[2.5+ Java アプリケーションにおけるトランザクションのためのプログラミング](#)

トランザクションを必要とする Java アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

[データの索引付けのためのプラグイン](#)

WebSphere eXtreme Scale クライアントは、作成する索引のタイプに応じて、索引の作成のために BackingMap に追加できる組み込みプラグインを提供します。

[2.5+ クライアントへの連続照会を使用したマップ更新の通知](#)

データ・グリッドでオブジェクトまたはエントリが挿入または更新されたときに、クライアント Java 仮想マシン (JVM) で通知を受けることができます。

親トピック: [単純データ・グリッドにアクセスするアプリケーションの開発](#)

関連タスク:

[単純データ・グリッドの作成](#)

関連資料:

[例: 単純データ・グリッド \(data grid\) ・アプリケーション
クライアント・プロパティ・ファイル](#)

関連情報:

[クライアント API 仕様](#)

Java API 資料へのアクセス

WebSphere® DataPower® XC10 アプライアンス 用の Java™ API 資料は、zip ファイルのアーカイブをダウンロードしてご使用の開発環境に取り込んで利用するか、インフォメーション・センターで閲覧できます。

このタスクについて

Java API 資料は、次のいずれかの場所からアクセスできます。

インフォメーション・センター

情報の検索には、WebSphere DataPower XC10 アプライアンス の製品情報の他に、インフォメーション・センターの API 資料を利用すると便利です。

zip ファイル・アーカイブ

このファイルは、リリースごとにダウンロードできます。その後、比較ツールを使ってリリースごとの API の変更内容を確認できます。また、objectgrid.jar ファイルをコンパイルしている場合は、ご使用の Eclipse プロジェクトに圧縮ファイルを直接リンクできます。このリンクを使って、統合開発環境に API 資料を組み込みます。

オンライン形式

オンライン形式は、IBM® の Web サイトにある API 資料の公開版です。サイトの URL は、Eclipse に直接リンクできます。現行バージョンに対するリンクは、常に最新バージョンにアップグレードされるので、修正や変更が加えられた資料を自動的に閲覧できます。

手順

- インフォメーション・センターの API 資料を閲覧します。詳しくは、

[API 資料 \(API documentation\)](#) を参照してください。

- API 資料の zip アーカイブをダウンロードします。

API 資料をダウンロードしてオフラインでご覧になる場合は、次のページから該当するリリースの zip ファイルをダウンロードしてください。[WebSphere DataPower XC10 アプライアンス wiki: API 資料 \(API documentation\)](#)

- オンライン形式の API 資料を閲覧します。常に最新バージョンにアップグレードされるリンクをブックマークするか、特定のバージョンに対するリンクを選んでご利用ください。リンク一覧については、[WebSphere DataPower XC10 アプライアンス wiki: API 資料 \(API documentation\)](#) を参照してください。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

例: 単純データ・グリッド (data grid)・アプリケーション

この例では、ObjectMap API を使用して、データ・グリッド (data grid)に対する 単純な作成、取得、更新、および削除操作を実行します。

アプリケーション例 SimpleGrid.java

```
import java.net.MalformedURLException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Date;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Random;
import java.util.HashSet;
import java.util.BitSet;
import java.util.concurrent.ConcurrentLinkedQueue;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.io.File;
import java.io.PrintWriter;
import java.io.FileWriter;
import java.io.PrintWriter;
import java.io.Serializable;
import java.io.IOException;
import java.text.DecimalFormat;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import com.ibm.websphere.objectgrid.ClientClusterContext;
import com.ibm.websphere.objectgrid.ConnectException;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectMap;
import com.ibm.websphere.objectgrid.ObjectGridException;
import com.ibm.websphere.objectgrid.ObjectGridManager;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
import com.ibm.websphere.objectgrid.ObjectGridRuntimeException;
import com.ibm.websphere.objectgrid.Session;
import com.ibm.websphere.objectgrid.config.BackingMapConfiguration;
import com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory;
import com.ibm.websphere.objectgrid.config.ObjectGridConfiguration;
import com.ibm.websphere.objectgrid.plugins.TransactionCallbackException;
import com.ibm.websphere.objectgrid.security.config.ClientSecurityConfiguration;
import com.ibm.websphere.objectgrid.security.config.ClientSecurityConfigurationFactory;
import com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator;
import
com.ibm.websphere.objectgrid.security.plugins.builtins.UserPasswordCredentialGenerator;
import com.ibm.websphere.objectgrid.server.CatalogServerProperties;

public class SimpleGrid {

    static String gridName = "test";
    static String mapName = gridName;
    static String username="xadmin";
    static String password="xadmin";
    static String hostName="localhost";
    static ObjectGrid clientGrid=null;
    static ConcurrentLinkedQueue<Session> sessions = new
ConcurrentLinkedQueue<Session>();

    static synchronized public ObjectGrid getObjectGrid() {
        if (clientGrid == null) {
            ClientClusterContext ccc = null;
            try {
                new
```

```

java.io.File(System.getProperty("java.io.tmpdir")).mkdirs();
    } catch (Throwable t) {
        t.printStackTrace();
    }
    ObjectGridManager ogm =
ObjectGridManagerFactory.getObjectGridManager();
    ClientSecurityConfiguration clientSC = getAdminClientConfig();
    List<ObjectGridConfiguration> ogConfigs=new
ArrayList<ObjectGridConfiguration>();
    ObjectGridConfiguration lclGridConfig =
ObjectGridConfigFactory.createObjectGridConfiguration(gridName);
    BackingMapConfiguration bmc =
ObjectGridConfigFactory.createBackingMapConfiguration(mapName);
    bmc.setNumberOfBuckets(0);
    lclGridConfig.addBackingMapConfiguration(bmc);
    ogConfigs.add(lclGridConfig);
    try {
        ccc = ogm.connect(hostName+":2809", clientSC, null);
    } catch (Throwable e) {
        e.printStackTrace();
    }
    if (ccc != null) {
        HashMap<String,List<ObjectGridConfiguration>> overrideMap = new
HashMap<String,List<ObjectGridConfiguration>>();
        overrideMap.put(ccc.getClusterName(),ogConfigs);
        ogm.setOverrideObjectGridConfigurations(overrideMap);
        try {
            clientGrid = ogm.getObjectGrid(ccc, gridName);
        } catch (ObjectGridRuntimeException ogre) {
            ogre.printStackTrace();
        }
    }
    }
    return clientGrid;
}

static public Session getSession() throws TransactionCallbackException,
ObjectGridException {
    Session session = sessions.poll();
    if (session == null && getObjectGrid()!=null) {
        session = getObjectGrid().getSession();
    }
    if (session == null)
        throw new IllegalStateException("unable to initialize connection
to objectgrid");
    return session;
}

static public void putSession(Session session) {
    if (session.isTransactionActive()) {
        try {
            session.rollback();
        } catch (Exception e) {
        }
    }
    sessions.add(session);
}

public static ClientSecurityConfiguration getAdminClientConfig() {
    // Creates a ClientSecurityConfiguration object using the specified file
    ClientSecurityConfiguration clientSC =
ClientSecurityConfigurationFactory.getClientSecurityConfiguration();
    clientSC.setSecurityEnabled(true);
    // Creates a CredentialGenerator using the passed-in user and password.
    CredentialGenerator credGen = new
UserPasswordCredentialGenerator(username,password);
    clientSC.setCredentialGenerator(credGen);
    return clientSC;
}

```



```

    }

    public static void main(String args[]) throws Exception {
    for (int i=0;i<args.length;i++) {
        if(args[i].startsWith("-username:")) {
            username = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-password:")) {
            password = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-gridname:")) {
            gridName = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-mapname:")) {
            mapName = args[i].substring(args[i].indexOf(":")+1);
        } else if(args[i].startsWith("-hostname:")) {
            hostName = args[i].substring(args[i].indexOf(":")+1);
        } else {
            System.out.println("usage: SimpleGrid [optional args]");
            System.out.println("    -username:<username>");
            System.out.println("    -password:<password>");
            System.out.println("    -gridname:<gridname>");
            System.out.println("    -mapname:<mapname>");
            System.out.println("    -hostname:<hostname>");
            System.exit(1);
        }
    }
    System.out.println("-----");
    System.out.println("Simple Grid Test");
    System.out.println("-----");
    System.out.println("username      : "+username);
    System.out.println("password      : "+password);
    System.out.println("gridname      : "+gridName);
    System.out.println("mapname       : "+mapName);
    System.out.println("hostname      : "+hostName);
    System.out.println("-----");

    if (getObjectGrid() == null) {
        System.out.println("ERROR: unable to connect to objectgrid at "+hostName);
        System.exit(1);
    }

    Session session = getSession();
    ObjectMap map=session.getMap(mapName);
    session.begin();
    Object data = map.get("TestKey");
    if (data!=null)
        map.remove("TestKey");
    map.insert("TestKey", "TestValue");
    session.commit();
    putSession(session);
}
}

```

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連タスク:

[単純データ・グリッドの作成](#)

関連情報:

[クライアント API 仕様](#)

Java API を使用した動的マップの作成

データ・グリッドがインスタンス化されたら、Java API を使用して動的マップを作成できます。マップ・テンプレートの事前定義セットに基づいて、動的にマップをインスタンス化することができます。

始める前に

動的マップで、どの構成オプションを使用する必要があるかを決定します。詳しくは、[動的マップの構成オプション](#)を参照してください。

手順

Session.getMap(String) メソッドを呼び出します。

事前定義テンプレートの 1 つの正規表現と一致するストリングを受け渡すと、適切なマップが作成されます。

```
ObjectMap map2 = sess.getMap("my_simple_data_grid.CT.P");  
ObjectMap map3 = sess.getMap("my_new_map.NONE");
```

my_simple_data_grid.CT.P マップは、作成時除去、ペシミスティック・ロック、および、ニア・キャッシュ無効化不可を使用するマップです。my_new_map.NONE マップには、除去設定も、ロック設定も、ニア・キャッシュ無効化設定もありません。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

[Java API を使用したデータ・グリッド・アプリケーションの開発](#)

関連資料:

[動的マップの構成オプション](#)

[REST ゲートウェイの例: 動的マップの作成](#)

Java アプリケーションにおけるトランザクションのためのプログラミング

トランザクションを必要とする Java™ アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

Java

2.5+ [Java アプリケーションのためのトランザクション内のデータとの対話](#)

セッションを使用してデータと対話します (挿入操作や更新操作など)。

2.5+ [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

データがデータ・グリッド内の複数の区画に分散されている場合は、単一のトランザクションで複数の区画を読み取って更新することができます。このタイプのトランザクションは複数区画トランザクションと呼ばれ、障害の際は 2 フェーズ・コミット・プロトコルを使用してトランザクションの調整とリカバリーを行います。

2.5+ [ロックの使用](#)

ロックにはライフサイクルがあり、さまざまな種類のロックはさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

Java アプリケーションのためのトランザクション内のデータとの対話

セッションを使用してデータと対話します (挿入操作や更新操作など)。

このタスクについて

ObjectMap インターフェースには、put、get、および remove などの一般的なマップ操作が含まれています。しかし、get、getForUpdate、insert、update、および remove といった、より具体的な操作名を使用してください。これらのメソッドは、従来のマップ API より意図を正確に伝えます。

注: upsert および upsertAll メソッドが ObjectMap の put および putAll メソッドに取って代わります。データ・グリッド内のエントリがキーと値をグリッドに挿入する必要があることを BackingMap に知らせるには、upsert メソッドを使用します。BackingMap は、insert または update のいずれかを行って値をグリッドに挿入します。

また、フレキシブルな 索引付けサポートを使用することもできます。

手順

- データを挿入します。

セッションを取得した後、次のコード・フラグメントを使用して、データの挿入に Map API を使用することができます。

```
Session session = ...;
ObjectMap personMap = session.getMap("PERSON");
session.begin();
Person p = new Person();
p.name = "John Doe";
personMap.insert(p.name, p);
session.commit();
```

- データを更新します。

データの更新に Map API を使用するには、次のコード・フラグメントを使用します。

```
session.begin();
Person p = (Person)personMap.getForUpdate("John Doe");
p.name = "John Doe";
p.age = 30;
personMap.update(p.name, p);
session.commit();
```

アプリケーションでは、通常は、単純な get ではなく、getForUpdate メソッドを使用してレコードをロックします。update メソッドは、更新済みの値を実際にマップに提供するために呼び出す必要があります。update を呼び出さないと、そのマップは変更されません。

- **2.5+** 次のメソッドを呼び出すことによって、2 フェーズ・コミットメント・プロトコルでデータを挿入します。session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE); session.begin(); 次のコード・スニペットは、2 フェーズ・コミット・プロトコルを使用してグリッドで操作を作成、取得、更新、および削除する方法を示しています。

```
Session session = og.getSession();
Objectmap map1 = session.getMap("Map1");
Objectmap map2 = session.getMap("Map2");
Objectmap map3 = session.getMap("Map3");
session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE);
session.begin();
map1.insert("randKey345", "HelloMap1");
map2.insert("randKey58901", "HelloMap2");
map3.insert("randKey58", "HelloMap3");
session.commit();
```

親トピック: **2.5+** [Java アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[データ・アクセスおよびトランザクション](#)

2.5+ [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

単一のトランザクションで複数の区画を更新するアプリケーションの開発

データがデータ・グリッド内の複数の区画に分散されている場合は、単一のトランザクションで複数の区画を読み取って更新することができます。このタイプのトランザクションは複数区画トランザクションと呼ばれ、障害の際は 2 フェーズ・コミット・プロトコルを使用してトランザクションの調整とリカバリーを行います。

2.5+ [2 フェーズ・コミットとエラー・リカバリー](#)

2 フェーズ・コミット・プロトコルは、分散トランザクションに関与するすべての区画を、そのトランザクションをコミットするかロールバックするかに基づいて調整します。

2.5+ [スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクションに書き込むアプリケーションの開発](#)

スタンドアロン WebSphere® eXtreme Scale 環境で複数の区画を備えた分散データ・グリッド用のアプリケーションを作成できます。

2.5+ [トランザクションを使用する eXtreme Scale クライアント・コンポーネントの開発](#)

WebSphere eXtreme Scale リソース・アダプターは、クライアント接続管理およびローカル・トランザクション・サポートを提供します。このサポートによって、Java Platform, Enterprise Edition (Java EE) アプリケーションは、Java EE ローカル・トランザクションや eXtreme Scale API を使用して、eXtreme Scale のクライアント接続を調べたり、ローカル・トランザクションを区別したりすることができます。

親トピック: **2.5+** [Java アプリケーションにおけるトランザクションのためのプログラミング](#)

関連タスク:

Java [Java アプリケーションのためのトランザクション内のデータとの対話](#)

スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクションに書き込むアプリケーションの開発

2.5+ スタンドアロン WebSphere® eXtreme Scale 環境で複数の区画を備えた分散データ・グリッド用のアプリケーションを作成できます。

始める前に

eXtremeIO プロトコルを使用可能にします。詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

制約事項: マルチ区画トランザクションへの書き込みを行うアプリケーションの開発を開始する前に、以下の制約事項に注意してください。

- 複数の区画に書き込むトランザクションでマルチマスター複製を使用することはできません。
- .NET 環境において WebSphere eXtreme Scale クライアントでマルチ区画を使用することはできません。
- ロダー・プラグインで構成された BackingMap は、マルチ区画トランザクションのマップを読み取ることはできますが、書き込むことはできません。
- ロック・ストラテジー NONE を使用している BackingMap は、マルチ区画トランザクションに参加できません。

このタスクについて

TxCommitProtocol 設定セッション API を使用して、スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクション・サポートを使用可能にします。新規 API では、以下の 2 つのオプションが用意されています。

- TxCommitProtocol.ONEPHASE: デフォルトの 1 フェーズ・コミットでトランザクションをコミットする必要があることを指示するトランザクション・コミット・プロトコル定数。このオプションを使用すると、トランザクションは複数の区画から読み取ることができ、書き込みは単一の区画にしか行えません。トランザクションが複数の区画に書き込むと、TransactionException 例外が発生します。
- TxCommitProtocol.TWOPHASE: 1 フェーズ・コミットまたは 2 フェーズ・コミットのいずれかでトランザクションをコミットする必要があることを指示するトランザクション・コミット・プロトコル定数。トランザクションが単一の区画に書き込む場合は、1 フェーズ・コミット・プロトコルが使用されます。そうではない場合は、トランザクションのコミットに 2 フェーズ・プロトコルが使用され、複数の区画への書き込み操作が行われます。

WebSphere Application Server 内で WebSphere eXtreme Scale のマルチトランザクション・サポートを構成することもできます。詳しくは、[トランザクションを使用する eXtreme Scale クライアント・コンポーネントの開発](#)を参照してください。

手順

1. データ・グリッドに接続します。詳しくは、[Java API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。
2. ObjectGrid.getSession メソッドを使用して、データ・グリッド・セッション・インスタンスを取得します。詳しくは、[Java API を使用したデータ・グリッド・アプリケーションの開発](#)を参照してください。
3. 以下のコード・スニペットを設定することで、2 フェーズ・コミット・プロトコルを使用可能にします。session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE); session.begin(); 以下のコード・スニペットで、2 フェーズ・コミット・プロトコルを使用してグリッドで操作を作成、取得、更新、および削除する方法を示します。

```
Session session = og.getSession();
Objectmap map1 = session.getMap("Map1");
Objectmap map2 = session.getMap("Map2");
Objectmap map3 = session.getMap("Map3");
session.setTxCommitProtocol(Session.TxCommitProtocol.TWOPHASE);
session.begin();
map1.insert("randKey345", "HelloMap1");
map2.insert("randKey58901", "HelloMap2");
map3.insert("randKey58", "HelloMap3");
session.commit();
```

次のタスク

マルチ区画トランザクションに対してトレースを使用可能にすることができます。詳しくは、[ログおよびトレース・データの分析](#)を参照してください。

親トピック: **2.5+** [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

関連概念:

トランザクションを使用する eXtreme Scale クライアント・コンポーネントの開発

WebSphere® eXtreme Scale リソース・アダプターは、クライアント接続管理およびローカル・トランザクション・サポートを提供します。このサポートによって、Java™ Platform, Enterprise Edition (Java EE) アプリケーションは、Java EE ローカル・トランザクションや eXtreme Scale API を使用して、eXtreme Scale のクライアント接続を調べたり、ローカル・トランザクションを区分したりすることができます。

始める前に

eXtreme Scale 接続ファクトリーのリソース参照を作成してください。

このタスクについて

eXtreme Scale データ・アクセス API を使用した作業に関するオプションがいくつかあります。いずれの場合も、eXtreme Scale 接続ファクトリーをアプリケーション・コンポーネントに注入するか、Java Naming Directory Interface (JNDI) で検索する必要があります。接続ファクトリーが検索された後、トランザクションを区分し、eXtreme Scale API にアクセスするための接続を作成することができます。

オプションで、接続ハンドルを取得するための追加オプションを提供する `com.ibm.websphere.xs.ra.XSConnectionFactory` に `javax.resource.cci.ConnectionFactory` インスタンスをキャストすることができます。結果の接続ハンドルは、`getSession` メソッドを提供する `com.ibm.websphere.xs.ra.XSConnection` インターフェイスにキャストする必要があります。`getSession` メソッドは `com.ibm.websphere.objectgrid.Session` オブジェクト・ハンドルを返します。このハンドルにより、アプリケーションが eXtreme Scale データ・アクセス API (ObjectMap API や EntityManager API など) をどれでも使用できるようになります。

Session ハンドルと派生オブジェクトは XSConnection ハンドルが存続している限り有効です。

以下の手順を使用して eXtreme Scale トランザクションを区分することができます。それぞれの手順を混合することはできません。例えば、同じアプリケーション・コンポーネントという状況下でグローバル・トランザクション区分とローカル・トランザクション区分を混用することはできません。

手順

- 自動コミット・ローカル・トランザクションを使用します。データ・アクセス操作、またはアクティブ・トランザクションをサポートしない操作を自動的にコミットするには、以下のステップに従います。
 1. グローバル・トランザクションのコンテキスト外で `com.ibm.websphere.xs.ra.XSConnection` 接続を取得します。
 2. `com.ibm.websphere.objectgrid.Session` セッションを取得し、このセッションを使用してデータ・グリッドと対話します。
 3. 自動コミット・トランザクションをサポートするデータ・アクセス操作を呼び出します。
 4. 接続をクローズします。
- ObjectGrid セッションを使用してローカル・トランザクションを区分します。Session オブジェクトを使用して ObjectGrid トランザクションを区分するには、以下のステップに従います。
 1. `com.ibm.websphere.xs.ra.XSConnection` 接続を取得します。
 2. `com.ibm.websphere.objectgrid.Session` セッションを取得します。
 3. `Session.begin()` メソッドを使用してトランザクションを開始します。
 4. セッションを使用してデータ・グリッドと対話します。
 5. `Session.commit()` メソッドまたは `rollback()` メソッドを使用してトランザクションを終了します。
 6. 接続をクローズします。
- `javax.resource.cci.LocalTransaction` トランザクションを使用してローカル・トランザクションを区分します。`javax.resource.cci.LocalTransaction` インターフェイスを使用して ObjectGrid トランザクションを区分するには、以下のステップに従います。
 1. `com.ibm.websphere.xs.ra.XSConnection` 接続を取得します。
 2. `XSConnection.getLocalTransaction()` メソッドを使用して `javax.resource.cci.LocalTransaction` トランザクションを取得します。
 3. `LocalTransaction.begin()` メソッドを使用してトランザクションを開始します。
 4. `com.ibm.websphere.objectgrid.Session` セッションを取得し、このセッションを使用してデータ・グリッドと対話します。
 5. `LocalTransaction.commit()` メソッドまたは `rollback()` メソッドを使用してトランザクションを終了します。

6. 接続をクローズします。
- 接続をグローバル・トランザクションに登録します。この手順はコンテナ管理トランザクションにも適用されます。
 1. javax.transaction.UserTransaction インターフェースを介するか、コンテナ管理トランザクションを使用するかして、グローバル・トランザクションを開始します。
 2. com.ibm.websphere.xs.ra.XSConnection 接続を取得します。
 3. com.ibm.websphere.objectgrid.Session セッションを取得して使用します。
 4. 接続をクローズします。
 5. グローバル・トランザクションをコミットまたはロールバックします。
 - **2.5+** トランザクションで複数の区画を作成するように接続を構成します。Session オブジェクトを使用して ObjectGrid トランザクションを区分するには、以下のステップに従います。
 1. 新規 com.ibm.websphere.xs.ra.XSConnectionSpec オブジェクトを作成します。
 2. XSConnectionSpec メソッドおよび setMultiPartitionSupportEnabled メソッドを、引数 true を指定して呼び出します。
 3. com.ibm.websphere.xs.ra.XSConnection 接続を取得して、XSConnectionSpec を ConnectionFactory.getConnection メソッドに渡します。
 4. com.ibm.websphere.objectgrid.Session セッションを取得して使用します。

例

次のコード例を参照してください。このコード例は、eXtreme Scale トランザクションを区分する先行ステップを示しています。

```
// (C) Copyright IBM Corp. 2001, 2012.
// All Rights Reserved. Licensed Materials - Property of IBM.
package com.ibm.ws.xs.ra.test.ee;

import javax.naming.InitialContext;
import javax.resource.cci.Connection;
import javax.resource.cci.ConnectionFactory;
import javax.resource.cci.LocalTransaction;
import javax.transaction.Status;
import javax.transaction.UserTransaction;

import junit.framework.TestCase;

import com.ibm.websphere.objectgrid.ObjectMap;
import com.ibm.websphere.objectgrid.Session;
import com.ibm.websphere.xs.ra.XSConnection;

/**
 * This sample requires that it runs in a J2EE context in your
 * application server. For example, using the JUnitEE framework servlet.
 *
 * The code in these test methods would typically reside in your own servlet,
 * EJB, or other web component.
 *
 * The sample depends on a configured WebSphere eXtreme Scale connection
 * factory registered at of JNDI Name of "eis/embedded/wxscf" that defines
 * a connection to a grid containing a Map with the name "Map1".
 *
 * The sample does a direct lookup of the JNDI name and does not require
 * resource injection.
 */
public class DocSampleTests extends TestCase {
    public final static String CF_JNDI_NAME = "eis/embedded/wxscf";
    public final static String MAP_NAME = "Map1";

    Long          key = null;
    Long          value = null;
    InitialContext ctx = null;
    ConnectionFactory cf = null;

    public DocSampleTests() {
```

```

}
public DocSampleTests(String name) {
    super(name);
}
protected void setUp() throws Exception {
    ctx = new InitialContext();
    cf = (ConnectionFactory)ctx.lookup(CF_JNDI_NAME);
    key = System.nanoTime();
    value = System.nanoTime();
}
/**
 * This example runs when not in the context of a global transaction
 * and uses autocommit.
 */
public void testLocalAutocommit() throws Exception {
    Connection conn = cf.getConnection();
    try {
        Session session = ((XSConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
    }
    finally {
        conn.close();
    }
}

/**
 * This example runs when not in the context of a global transaction
 * and demarcates the transaction using session.begin()/session.commit()
 */
public void testLocalSessionTransaction() throws Exception {
    Session session = null;
    Connection conn = cf.getConnection();
    try {
        session = ((XSConnection)conn).getSession();
        session.begin();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
        session.commit();
    }
    finally {
        if (session != null && session.isTransactionActive()) {
            try { session.rollback(); }
            catch (Exception e) { e.printStackTrace(); }
        }
        conn.close();
    }
}

/**
 * This example uses the LocalTransaction interface to demarcate
 * transactions.
 */
public void testLocalTranTransaction() throws Exception {
    LocalTransaction tx = null;
    Connection conn = cf.getConnection();
    try {
        tx = conn.getLocalTransaction();
        tx.begin();
        Session session = ((XSConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
        tx.commit(); tx = null;
    }
    finally {
        if (tx != null) {
            try { tx.rollback(); }
            catch (Exception e) { e.printStackTrace(); }
        }
    }
}

```

```

        }
        conn.close();
    }
}

/**
 * This example depends on an externally managed transaction,
 * the externally managed transaction might typically be present in
 * an EJB with its transaction attributes set to REQUIRED or REQUIRES_NEW.
 * NOTE: If there is NO global transaction active, this example runs in auto-
commit
 * mode because it doesn't verify a transaction exists.
 */
public void testGlobalTransactionContainerManaged() throws Exception {
    Connection conn = cf.getConnection();
    try {
        Session session = ((XSCConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
    }
    catch (Throwable t) {
        t.printStackTrace();
        UserTransaction tx =
(UserTransaction)ctx.lookup("java:comp/UserTransaction");
        if (tx.getStatus() != Status.STATUS_NO_TRANSACTION) {
            tx.setRollbackOnly();
        }
    }
    finally {
        conn.close();
    }
}

/**
 * This example demonstrates starting a new global transaction using the
 * UserTransaction interface. Typically the container starts the global
 * transaction (for example in an EJB with a transaction attribute of
 * REQUIRES_NEW), but this sample will also start the global transaction
 * using the UserTransaction API if it is not currently active.
 */
public void testGlobalTransactionTestManaged() throws Exception {
    boolean started = false;
    UserTransaction tx = (UserTransaction)ctx.lookup("java:comp/UserTransaction");
    if (tx.getStatus() == Status.STATUS_NO_TRANSACTION) {
        tx.begin();
        started = true;
    }
    // else { called with an externally/container managed transaction }
    Connection conn = null;
    try {
        conn = cf.getConnection(); // Get connection after the global tran starts
        Session session = ((XSCConnection)conn).getSession();
        ObjectMap map = session.getMap(MAP_NAME);
        map.insert(key, value); // Or various data access operations
        if (started) {
            tx.commit(); started = false; tx = null;
        }
    }
    finally {
        if (started) {
            try { tx.rollback(); }
            catch (Exception e) { e.printStackTrace(); }
        }
        if (conn != null) { conn.close(); }
    }
}
/**
/**

```

```

* This example demonstrates a multi-partition transaction.
*/

        public void
testGlobalTransactionTestManagedMultiPartition() throws Exception {
    boolean started = false;
    XSConnectionSpec connSpec = new
XSConnectionSpec();

    connSpec.setWriteToMultiplePartitions(true);

    UserTransaction tx =
(UserTransaction)ctx.lookup("java:comp/UserTransaction");
    if (tx.getStatus() ==
Status.STATUS_NO_TRANSACTION) {
        tx.begin();
        started = true;
    }
    // else { called with an externally/container managed
transaction }

    Connection conn = null;
    try {
        conn = cf.getConnection(connSpec); // Get
connection after the global tran starts

        Session session =
((XSConnection)conn).getSession();

        access operations

        if (started) {
            tx.commit(); started =
false; tx = null;
        }
    }
    finally {
        if (started) {
            try {
                tx.rollback(); }
            catch (Exception
e) { e.printStackTrace(); }
        }
        if (conn != null) { conn.close();
    }
}
}
}

```

親トピック: **2.5+** [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)

ロックの使用

ロックにはライフサイクルがあり、さまざまな種類のロックはさまざまな方法で他のロックと互換性を持ちます。ロックはデッドロック・シナリオにならないように、正しい順序で処理する必要があります。

2.5+ [Java アプリケーションにおけるロックの構成と実装](#)

WebSphere® eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。

2.5+ [ロック・シナリオでの例外処理の実装 \(Java アプリケーション\)](#)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

Java **2.5+** [例: flush メソッドのロック順序](#)

ObjectMap インターフェイスで、コミット前に flush メソッドを呼び出すと、ロック順序に関する追加の考慮事項が発生する場合があります。通常、flush メソッドは、マップに対して行われた変更を、Loader プラグインを通じてバックエンドに強制するために使用されます。

Java **2.5+** [Java のトランザクション分離の例](#)

トランザクション分離は、1つの操作で行われた変更がどのように他の並行操作に可視になるのかを定義します。以下の例を使用して、Java アプリケーションのトランザクション分離レベルを定義することができます。

親トピック: **2.5+** [Java アプリケーションにおけるトランザクションのためのプログラミング](#)

Java アプリケーションにおけるロックの構成と実装

WebSphere® eXtreme Scale 構成の各 BackingMap に対するオプティミスティック、ペシミスティック、あるいはロックなしのストラテジーを定義できます。

始める前に

- 使用したいロック・ストラテジーを決めます。詳しくは、[ロック・ストラテジー](#)を参照してください。
- 動的マップの構成によってロック・ストラテジーを構成することもできます。詳しくは、[ロック・ストラテジーの構成](#)を参照してください。

このタスクについて

java.lang.IllegalStateException 例外を避けるために、ObjectGrid インスタンスで initialize メソッドまたは getSession メソッドを呼び出す前に、setLockStrategy メソッドを呼び出す必要があります。

手順

1. Java™ アプリケーションでロック・ストラテジーを構成します。
 - オプティミスティック・ロック・ストラテジーの構成。setLockStrategy メソッドを使用します。

```
import com.ibm.websphere.objectgrid.BackingMap;
import com.ibm.websphere.objectgrid.LockStrategy;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
...
ObjectGrid og =
    ObjectGridManagerFactory.getObjectGridManager().createObjectGrid("test"
);
BackingMap bm = og.defineMap("optimisticMap");
bm.setLockStrategy( LockStrategy.OPTIMISTIC );
```

- ペシミスティック・ロック・ストラテジーの構成。setLockStrategy メソッドを使用します。

```
import com.ibm.websphere.objectgrid.BackingMap;
import com.ibm.websphere.objectgrid.LockStrategy;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
...
ObjectGrid og =
    ObjectGridManagerFactory.getObjectGridManager().createObjectGrid("test"
);
BackingMap bm = og.defineMap("pessimisticMap");
bm.setLockStrategy( LockStrategy.PESSIMISTIC );
```

- ロックなしストラテジーの構成。setLockStrategy メソッドを使用します。

注: ロックなしストラテジーを使用するように構成された BackingMap は、複数区画トランザクションに
関与できません。

```
import com.ibm.websphere.objectgrid.BackingMap;
import com.ibm.websphere.objectgrid.LockStrategy;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
...
ObjectGrid og =
    ObjectGridManagerFactory.getObjectGridManager().createObjectGrid("test"
);
BackingMap bm = og.defineMap("noLockingMap");
bm.setLockStrategy( LockStrategy.NONE );
```

2. ロック・タイムアウト値を構成します。BackingMap インスタンスの setLockTimeout メソッドを使用しま
す。

```
bm.setLockTimeout( 60 );
```

setLockTimeout メソッドのパラメーターは、Java プリミティブの整数で、eXtreme Scale がロック・モードを
認可されるのを待たなければならない秒数を指定します。BackingMap に構成されているロック待ちタイムアウト

ト値よりも長くトランザクションが待つ場合は、`com.ibm.websphere.objectgrid.LockTimeoutException` 例外が発生します。

3. ペシミスティック・ロック・ストラテジーを使用している場合は、`lock` メソッドを使用して、データ・グリッド内のキーをロックするか、またはキーをロックして値がデータ・グリッド内に存在するかどうかを確認することができます。これまでのリリースでは、`get` API および `getForUpdate` API を使用してデータ・グリッド内のキーをロックしていました。しかし、クライアントからデータを取得する必要がなかった場合は、大きくなる可能性がある値オブジェクトを取得してクライアントに渡すことになるため、パフォーマンスが低下します。`containsKey` メソッドはロックを保持しないため、ペシミスティック・ロックを使用しているときには、`get` および `getForUpdate` メソッドを使用して適切なロックを取得しなければなりません。現在は、ロックを保持しているとき、`lock` API により `containsKey` メソッドが提供されるようになりました。次の例を参照してください。
 - 次のメソッドはマップ内のキーをロックします。キーが存在する場合は `true` を返し、キーが存在しない場合は `false` を返します。

```
boolean ObjectMap.lock(Object key, LockMode lockMode);
```

- 次のメソッドはマップ内のキーのリストをロックし、`true` または `false` 値のリストを返します。キーが存在する場合は `true` を返し、キーが存在しない場合は `false` を返します。

```
List<Boolean> ObjectMap.lockAll(List keys, LockMode lockMode);
```

`LockMode` は列挙型で以下に示す値を取ることができ、ここでロックしたいキーを指定できます。

- `SHARED`、`UPGRADABLE`、および `EXCLUSIVE`

`LockMode` パラメーターの設定例を次に示します。

```
session.begin();
map.lock(key, LockMode.UPGRADABLE);
map.upsert();
session.commit();
```

親トピック: **2.5+** [ロックの使用](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[ロック・ストラテジーの構成](#)

関連資料:

例: [flush](#) メソッドのロック順序

ロック・シナリオでの例外処理の実装 (Java™ アプリケーション)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

手順

1. 例外をキャッチし、結果のメッセージを表示します。

```
try {  
    ...  
} catch (ObjectGridException oe) {  
    System.out.println(oe);  
}
```

結果、次の例外が表示されます。

```
com.ibm.websphere.objectgrid.plugins.LockDeadlockException: Message
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるメッセージを表示します。

2. 例外の後、トランザクションをロールバックします。

```
Session sess = ...;  
ObjectMap person = sess.getMap("PERSON");  
boolean activeTran = false;  
try  
{  
    sess.begin();  
    activeTran = true;  
    Person p = (IPerson)person.get("Lynn");  
    // Lynn had a birthday, so we make her 1 year older.  
    p.setAge( p.getAge() + 1 );  
    person.put( "Lynn", p );  
    sess.commit();  
    activeTran = false;  
}  
finally  
{  
    if ( activeTran ) sess.rollback();  
}
```

コード・スニペットの finally ブロックは、予期しない例外が発生したときにトランザクションがロールバックされるようにしています。LockDeadlockException 例外のみでなく、発生する可能性のあるその他の予期しない例外もすべて処理します。finally ブロックは、commit メソッドの呼び出し時に例外が発生するケースも処理します。この例は、予期しない例外を処理する唯一の方法ではありません。アプリケーションが、発生する予期しない例外のいくつかをキャッチし、そのアプリケーション例外の 1 つを表示するケースも存在するかもしれません。適宜 catch ブロックを追加できますが、アプリケーションは、コード・スニペットがトランザクションを完了せずに終了しないようにする必要があります。

親トピック: [2.5+ ロックの使用](#)

例: flush メソッドのロック順序

ObjectMap インターフェースで、コミット前に flush メソッドを呼び出すと、ロック順序に関する追加の考慮事項が発生する場合があります。通常、flush メソッドは、マップに対して行われた変更を、Loader プラグインを通じてバックエンドに強制するために使用されます。

この状況では、バックエンドは独自のロック・マネージャーを使用して並行性を制御するので、ロック待ち状態とデッドロックが WebSphere® eXtreme Scale クライアント ロック・マネージャーではなくバックエンドで起こることがあります。次のトランザクションについて検討します。

```
Session sess = ...;
ObjectMap person = sess.getMap("PERSON");
boolean activeTran = false;
try
{
    sess.begin();
    activeTran = true;
    Person p = (IPerson)person.get("Lynn");
    p.setAge( p.getAge() + 1 );
    person.put( "Lynn", p );
    person.flush();
    ...
    p = (IPerson)person.get("Tom");
    p.setAge( p.getAge() + 1 );
    sess.commit();
    activeTran = false;
}
finally
{
    if ( activeTran ) sess.rollback();
}
```

何かほかのトランザクションが Tom も更新し、flush メソッドを呼び出し、次に Lynn を更新したとします。この状態が発生した場合、2 つのトランザクションの以下のインターリーピングの結果、データベースはデッドロック状態になります。

```
flush の実行時に "Lynn" のトランザクション 1 に対して X ロックが認可されます。
flush の実行時に "Tom" のトランザクション 2 に対して X ロックが認可されます。
コミット処理中に "Tom" のトランザクション 1 によって、X ロックが要求されます。
(トランザクション 1 は、
トランザクション 2 によって所有されたロックを待機するのをブロックします。)
コミット処理中に "Lynn" のトランザクション 2 によって、X ロックが要求されます。
(トランザクション 2 は、
トランザクション 1 によって所有されたロックを待機するのをブロックします。)
```

この例は、flush メソッドの使用により、WebSphere eXtreme Scale クライアント 内ではなくデータベース内でデッドロックが発生することを示しています。このデッドロック例は、どのロック・ストラテジーを使用しても発生する可能性があります。アプリケーションで flush メソッドを使用している場合や、Loader が BackingMap にプラグインされている場合は、この種のデッドロックが起こらないように注意する必要があります。上記の例は、WebSphere eXtreme Scale クライアント がロック待ちタイムアウト機構を備えている もう 1 つの理由を示しています。データベース・ロックを待機するトランザクションは、WebSphere eXtreme Scale クライアント マップ・エントリーのロックを所有している間、待機し続ける可能性があります。データベース・レベルの問題により、WebSphere eXtreme Scale クライアント ロック・モードの待機時間が過大になり、LockTimeoutException 例外が発生する可能性があります。

親トピック: [2.5+ ロックの使用](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[ロック・ストラテジーの構成](#)

Java のトランザクション分離の例

トランザクション分離は、1つの操作で行われた変更がどのように他の並行操作に可視になるのかを定義します。以下の例を使用して、Java™ アプリケーションのトランザクション分離レベルを定義することができます。

ペシミスティック・ロックでの反復可能読み取り

```
map = session.getMap("Order");
session.setTransactionIsolation(Session.TRANSACTION_REPEATABLE_READ);
session.begin();

// An S lock is requested and held and the value is copied into
// the transactional cache.
Order order = (Order) map.get("100");
// The entry is evicted from the transactional cache.
map.invalidate("100", false);

// The same value is requested again. It already holds the
// lock, so the same value is retrieved and copied into the
// transactional cache.
Order order2 = (Order) map.get("100");

// All locks are released after the transaction is synchronized
// with cache map.
session.commit();
```

ペシミスティック・ロックでの読み取りコミット済み

```
map1 = session1.getMap("Order");
session1.setTransactionIsolation(Session.TRANSACTION_READ_COMMITTED);
session1.begin();

// An S lock is requested but immediately released and
//the value is copied into the transactional cache.

Order order = (Order) map1.get("100");

// The entry is evicted from the transactional cache.
map1.invalidate("100", false);

// A second transaction updates the same order.
// It acquires a U lock, updates the value, and commits.
// The ObjectGrid successfully acquires the X lock during
// commit since the first transaction is using read
// committed isolation.

Map orderMap2 = session2.getMap("Order");
session2.begin();
order2 = (Order) orderMap2.getForUpdate("100");
order2.quantity=2;
orderMap2.update("100", order2);
session2.commit();

// The same value is requested again. This time, they
// want to update the value, but it now reflects
// the new value
Order order1Copy = (Order) map1.getForUpdate("100");
```

親トピック: [2.5+ ロックの使用](#)

関連概念:

[トランザクション分離](#)

データの索引付けのためのプラグイン

WebSphere® eXtreme Scale クライアント は、作成する索引のタイプに応じて、索引の作成のために BackingMap に追加できる組み込みプラグインを提供します。

HashIndex

組み込み HashIndex である `com.ibm.websphere.objectgrid.plugins.index.HashIndex` クラスは、動的索引を作成するために BackingMap に追加可能な `MapIndexPlugin` プラグインです。このクラスは、`MapIndex` と `MapRangeIndex` の両方のインターフェースをサポートします。索引を定義し、実装すると、照会のパフォーマンスを大幅に改善できます。

[索引によるデータへのアクセス \(索引 API\)](#)

より効率的なデータ・アクセスのために索引付けを使用します。

[セッションを使用したグリッド内データへのアクセス](#)

アプリケーションは、`Session` インターフェースを介してトランザクションを開始および終了できます。`Session` インターフェースは、アプリケーションを基にした `ObjectMap` および `JavaMap` インターフェースへのアクセスも提供します。

[HashIndex プラグインの構成](#)

組み込み HashIndex である `com.ibm.websphere.objectgrid.plugins.index.HashIndex` クラスは、動的索引を使用してプログラマチックに、構成することができます。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

索引によるデータへのアクセス (索引 API)

より効率的なデータ・アクセスのために索引付けを使用します。

このタスクについて

HashIndex クラスは、組み込みアプリケーション索引インターフェースである MapIndex と MapRangeIndex の両方をサポートすることのできる組み込み索引プラグイン実装です。ユーザー独自の索引を作成することもできます。HashIndex 動的索引としてバックアップ・マップに追加し、MapIndex または MapRangeIndex の索引プロキシ・オブジェクトを取得し、その索引プロキシ・オブジェクトを使用してキャッシュ・オブジェクトを検索することができます。

ローカル・マップ内のキーを反復処理する場合は、デフォルトの索引を使用できます。この索引はまったく構成を必要としませんが、エージェントを使用するか ShardEvents.shardActivated(ObjectGrid shard) メソッドから取得した ObjectGrid インスタンスを使用して、断片に対して使用しなければなりません。

注: 分散環境では、索引オブジェクトがクライアント ObjectGrid から取得された場合、その索引のタイプはクライアント索引オブジェクトになり、すべての索引操作はリモート・サーバー ObjectGrid で実行されます。マップが区画化されている場合、索引操作は各区画でリモートに実行されます。各区画の結果はマージされてからアプリケーションに戻されます。パフォーマンスは、区画数と、各区画が戻す結果のサイズによって決まります。これらの要因が両方とも大きいと、パフォーマンスが低下することがあります。

手順

索引を使用してマップのキーと値にアクセスします。

- ローカル索引:

ローカル・マップ内のキーと値を反復処理する場合は、デフォルトの索引を使用できます。デフォルトの索引は、エージェントを使用するか、ShardEvents.shardActivated(ObjectGrid shard) メソッドから取得した ObjectGrid インスタンスを使用するかして、断片に対してのみ機能します。次の例を参照してください。

```
MapIndex keyIndex = (MapIndex)
objMap.getIndex(MapIndexPlugin.SYSTEM_KEY_INDEX_NAME);
Iterator keyIterator = keyIndex.findAll();
```

- 動的索引:

BackingMap インスタンスから動的索引を、いつでもプログラマチックに作成および除去することができます。動的索引と静的索引の違いは、動的索引は、索引を含む ObjectGrid インスタンスが初期化されたあとでも作成できる、という点です。動的索引付けは、静的索引付けとは違って非同期プロセスであり、使用される前に作動可能状態になっている必要があります。このメソッドは、動的索引の取得および使用に、静的索引と同じアプローチを使用します。動的索引は、不要になると除去できます。BackingMap インターフェースには、動的索引を作成および除去するためのメソッドがあります。

createDynamicIndex メソッドおよび removeDynamicIndex メソッドの詳細については、[BackingMap API](#) を参照してください。

```
import com.ibm.websphere.objectgrid.ObjectGridManagerFactory;
import com.ibm.websphere.objectgrid.ObjectGridManager;
import com.ibm.websphere.objectgrid.ObjectGrid;
import com.ibm.websphere.objectgrid.BackingMap;

ObjectGridManager ogManager =
ObjectGridManagerFactory.getObjectGridManager();
BackingMap bm = og.getMap("person");

// create index after ObjectGrid initialization without
DynamicIndexCallback.
bm.createDynamicIndex("CODE", true, "employeeCode", null);

try {
// If not using DynamicIndexCallback, need to wait for the Index to be
ready.
// The waiting time depends on the current size of the map
Thread.sleep(3000);
} catch (Throwable t) {
// ...
}
```

```

// When the index is ready, applications can try to get application index
// interface instance.
// Applications have to find a way to ensure that the index is ready to use,
// if not using DynamicIndexCallback interface.
// The following example demonstrates the way to wait for the index to be
ready
// Consider the size of the map in the total waiting time.

Session session = og.getSession();
ObjectMap m = session.getMap("person");
MapRangeIndex codeIndex = null;

int counter = 0;
int maxCounter = 10;
boolean ready = false;
while (!ready && counter < maxCounter) {
    try {
        counter++;
        codeIndex = (MapRangeIndex) m.getIndex("CODE");
        ready = true;
    } catch (IndexNotReadyException e) {
        // implies index is not ready, ...
        System.out.println("Index is not ready. continue to wait.");
        try {
            Thread.sleep(3000);
        } catch (Throwable tt) {
            // ...
        }
    } catch (Throwable t) {
        // unexpected exception
        t.printStackTrace();
    }
}

if (!ready) {
    System.out.println("Index is not ready. Need to handle this
situation.");
}

// Use the index to perform queries
// Refer to the MapIndex or MapRangeIndex interface for supported
operations.
// The object attribute on which the index is created is the EmployeeCode.
// Assume that the EmployeeCode attribute is Integer type: the
// parameter that is passed into index operations has this data type.

Iterator iter = codeIndex.findLessEqual(new Integer(15));

// remove the dynamic index when no longer needed

bm.removeDynamicIndex("CODE");
// Close the session (optional in Version 7.1.1 and later) for improved
performance
session.close();

```

次のタスク

DynamicIndexCallback インターフェースを使用して、索引付けイベントの発生時に通知を受けることができます。詳しくは、[DynamicIndexCallback インターフェース](#)を参照してください。

[DynamicIndexCallback インターフェース](#)

DynamicIndexCallback インターフェースは、作動可能、エラー、または破棄という索引付けイベントの発生時に、そのことを通知してもらう必要のあるアプリケーションのために設計されています。

DynamicIndexCallback は、BackingMap の createDynamicIndex メソッドのオプション・パラメーターです。アプリケーションは、索引付けイベントの通知を受け取ると、登録済みの DynamicIndexCallback インスタンスを使用して、ビジネス・ロジックを実行することができます。

親トピック: [データの索引付けのためのプラグイン](#)

DynamicIndexCallback インターフェース

DynamicIndexCallback インターフェースは、作動可能、エラー、または破棄という索引付けイベントの発生時に、そのことを通知してもらう必要のあるアプリケーションのために設計されています。DynamicIndexCallback は、BackingMap の createDynamicIndex メソッドのオプション・パラメーターです。アプリケーションは、索引付けイベントの通知を受け取ると、登録済みの DynamicIndexCallback インスタンスを使用して、ビジネス・ロジックを実行することができます。

索引付けイベント

例えば、作動可能イベントは、索引を使用する準備が整ったことを意味します。アプリケーションは、このイベントの通知を受け取ると、アプリケーション索引インターフェースのインスタンスの取得および使用を試行することができます。

例: DynamicIndexCallback インターフェースの使用

```
BackingMap personBackingMap = ivObjectGrid.getMap("person");
DynamicIndexCallback callback = new DynamicIndexCallbackImpl();
personBackingMap.createDynamicIndex("CODE", true, "employeeCode", callback);

class DynamicIndexCallbackImpl implements DynamicIndexCallback {
    public DynamicIndexCallbackImpl() {
    }

    public void ready(String indexName) {
        System.out.println("DynamicIndexCallbackImpl.ready() -> indexName = " +
indexName);

        // Simulate what an application would do when notified that the index is
ready.
        // Normally, the application would wait until the ready state is reached and
then proceed
        // with any index usage logic.
        if("CODE".equals(indexName)) {
            ObjectGridManager ogManager =
ObjectGridManagerFactory.getObjectGridManager();
            ObjectGrid og = ogManager.createObjectGrid( "grid" );
            Session session = og.getSession();
            ObjectMap map = session.getMap("person");
            MapIndex codeIndex = (MapIndex) map.getIndex("CODE");
            Iterator iter = codeIndex.findAll(codeValue);
            // Close the session (optional in Version 7.1.1 and later) for improved
performance
            session.close();
        }
    }

    public void error(String indexName, Throwable t) {
        System.out.println("DynamicIndexCallbackImpl.error() -> indexName = " +
indexName);
        t.printStackTrace();
    }

    public void destroy(String indexName) {
        System.out.println("DynamicIndexCallbackImpl.destroy() -> indexName = " +
indexName);
    }
}
```

親トピック: [索引によるデータへのアクセス \(索引 API\)](#)

セッションを使用したグリッド内データへのアクセス

アプリケーションは、Session インターフェースを介してトランザクションを開始および終了できます。Session インターフェースは、アプリケーションを基にした ObjectMap および JavaMap インターフェースへのアクセスも提供します。

各 ObjectMap または JavaMap インスタンスは、特定のセッション・オブジェクトに直接結合しています。eXtreme Scale にアクセスしたい各スレッドは、まず最初に ObjectGrid オブジェクトからセッションを取得しなければなりません。セッション・インスタンスは、スレッド間で同時に共有することはできません。WebSphere® eXtreme Scale は、スレッドのローカル・ストレージをまったく使用しませんが、プラットフォームの制約事項により、あるスレッドから別のスレッドへのセッションの受け渡しの機会が制限されることがあります。

方式

get メソッド

アプリケーションは ObjectGrid.getSession メソッドを使用して、セッション・インスタンス を ObjectGrid オブジェクトから取得します。次の例は、Session インターフェースを取得する方法を示しています。

```
ObjectGrid objectGrid = ...; Session sess = objectGrid.getSession();
```

セッションを取得した後、スレッドはそのセッションへの参照を専用保持します。getSession メソッドを複数回呼び出すと、その度に新規セッション・オブジェクトが戻されます。

トランザクション・メソッドとセッション・メソッド

セッションは、トランザクションの開始、コミット、またはロールバックに使用できます。ObjectMap と JavaMap を使用した BackingMap に対する操作は、セッション・トランザクション内では非常に効率よく実行されます。トランザクションが開始された後は、そのトランザクションの有効範囲にある 1 つ以上の BackingMap に対するすべての変更は、そのトランザクションがコミットされるまで、特別のトランザクション・キャッシュに保管されます。トランザクションがコミットされると、保留になっている変更内容は BackingMap とローダーに適用され、その ObjectGrid のその他のクライアントから見えるようになります。

WebSphere eXtreme Scale は、トランザクションを自動的にコミットする機能 (自動コミットともいう) もサポートします。すべての ObjectMap オペレーションがアクティブ・トランザクションのコンテキストの外部で実行される場合は、暗黙のトランザクションはそのオペレーションの前に開始され、そのトランザクションはアプリケーションに制御が戻される前に自動的にコミットされます。

```
Session session = objectGrid.getSession();
ObjectMap objectMap = session.getMap("someMap");
session.begin();
objectMap.insert("key1", "value1");
objectMap.insert("key2", "value2");
session.commit();
objectMap.insert("key3", "value3"); // auto-commit
```

Session.flush メソッド

Session.flush メソッドは、ローダーが BackingMap に関連付けられているときにのみ意味があります。flush メソッドは、トランザクション・キャッシュ内の変更内容の現行セットを使用してローダーを呼び出します。ローダーは、変更内容をバックエンドに適用します。これらの変更内容は、flush が呼び出される時はコミットされません。flush 呼び出しの後、セッション・トランザクションがコミットされると、flush 呼び出しの後で発生する更新のみがローダーに適用されます。flush 呼び出しの後、セッション・トランザクションがロールバックされると、フラッシュされた変更内容はトランザクション内のその他すべての保留している変更内容と一緒に廃棄されます。flush メソッドは、ローダーに対するバッチ操作の機会を制限するので、慎重に使用してください。以下は、Session.flush メソッドの使用例です。

```
Session session = objectGrid.getSession();
session.begin();
// make some changes
...
session.flush(); // push these changes to the Loader, but don't commit yet
// make some more changes
...
session.commit();
```

NoWriteThrough メソッド

いくつかのマップはローダーによってバックアップされます。ローダーはマップ内のデータ用に永続ストレージを提供

します。eXtreme Scale マップのみにデータをコミットし、ローダーにデータをプッシュアウトしないことが有益な場合があります。Session インターフェースは、この目的のために beginNoWriteThrough メソッドを提供します。beginNoWriteThrough メソッドは、begin メソッドのようなトランザクションを開始します。beginNoWriteThrough メソッドでは、トランザクションがコミットされると、データはメモリー内のマップにのみコミットされ、ローダーが提供する永続ストレージにはコミットされません。このメソッドが非常に役立つのは、データがマップにプリロードされることです。

分散 ObjectGrid インスタンスを使用する場合、サーバーで遠くのキャッシュは変更せずにニア・キャッシュのみを変更するには、beginNoWriteThrough メソッドが役立ちます。ニア・キャッシュでデータが不整合であると認識されている場合は、beginNoWriteThrough メソッドを使用すると、エントリーをサーバーでは無効にせずに、ニア・キャッシュで無効にすることができます。

Session インターフェースは、現在活動中のトランザクション・タイプを判別する isWriteThroughEnabled メソッドも提供します。

```
Session session = objectGrid.getSession();
session.beginNoWriteThrough();
// make some changes ...
session.commit(); // these changes will not get pushed to the Loader
```

TxID オブジェクト・メソッドの取得

TxID オブジェクトは、内部が見えないオブジェクトで、活動中のトランザクションを識別します。以下の目的には、TxID オブジェクトを使用します。

- ある特定のトランザクションを検索している場合の比較用
- TransactionCallback とローダーのオブジェクト間で共有データを保管するため
- 1 フェーズ・コミット・プロトコルまたは 2 フェーズ・コミット・プロトコルを使用しているセッション・トランザクションからこのトランザクションが開始されたかどうかを判別します。TxID.toString() 出力を調べることによって、このトランザクションが単一区画トランザクションに対するものであったか複数区画トランザクションに対するものであったかを決定することができます。ストリングがキーワード「Local」で始まっている場合、これは単一区画トランザクションを示します。例えば、Local-40000139-72B2-C037-E000-1C271366B073 などです。ストリングがキーワード「WXS」で始まっている場合、これは複数区画トランザクションを示します。例えば WXS-40000139-72B2-BD3A-E000-1C271366B073 などです。

オブジェクト・スロット・フィーチャーについての追加情報は、『TransactionCallback プラグイン』と『ローダー』を参照してください。

パフォーマンス・モニター・メソッド

eXtreme Scale を WebSphere Application Server 内で使用する場合、パフォーマンス・モニタリング用にトランザクション・タイプをリセットすることが必要になることがあります。トランザクション・タイプの設定には、setTransactionType メソッドを使用できます。setTransactionType メソッドについて詳しくは、『WebSphere Application Server Performance Monitoring Infrastructure (PMI) を使用した ObjectGrid パフォーマンスのモニター』を参照してください。

完全な LogSequence メソッドの処理

WebSphere eXtreme Scale は、ある Java™ 仮想マシンから別のマシンへマップを配布する手段として、マップ変更セットを ObjectGrid リスナーに伝搬できます。リスナーが受信済み LogSequences を処理するのを容易にするために、Session インターフェースは processLogSequence メソッドを提供します。このメソッドは LogSequence 内で各 LogElement を検査し、LogSequence MapName によって識別される BackingMap に対して適切なオペレーション（例えば、挿入、更新、無効化など）を実行します。ObjectGrid セッションは、processLogSequence メソッドが呼び出される前に使用可能になっていなければなりません。アプリケーションは、セッションを完了するために適切な commit または rollback 呼び出しを実行する役割があります。自動コミット処理は、このメソッド呼び出しには使用できません。リモート JVM での受信側 ObjectGridEventListener による通常の処理では、この processLogSequence メソッドの呼び出しが続く beginNoWriteThrough メソッド（変更内容のエンドレスな伝搬を防止するもの）を使用し、次にトランザクションをコミットまたはロールバックすることで、セッションを開始することになります。

```
// Use the Session object that was passed in during
//ObjectGridEventListener.initialization...
session.beginNoWriteThrough();
// process the received LogSequence
try {
    session.processLogSequence(receivedLogSequence);
} catch (Exception e) {
    session.rollback(); throw e;
}
// commit the changes
session.commit();
```


markRollbackOnly メソッド

このメソッドを使用して、現行トランザクションに「rollback only」とマークを付けます。トランザクションに「rollback only」とマークを付けると、アプリケーションで commit メソッドが呼び出された場合でも、トランザクションはロールバックされます。このメソッドは、通常、トランザクションのコミットが許可されている場合にデータ破壊が発生する可能性があるとして認識されているとき、ObjectGrid 自体またはアプリケーションで使用されます。このメソッドが呼び出されると、このメソッドに渡される Throwable オブジェクトが com.ibm.websphere.objectgrid.TransactionException 例外にチェーニングされます。この例外は、以前に「rollback only」とマーク付けされたセッションで commit メソッドが呼び出された場合の結果です。既に「rollback only」とマーク付けされているトランザクションのこのメソッドに対する以降の呼び出しは、無視されます。つまり、ヌル以外の Throwable 参照を渡す最初の呼び出しのみが使用されます。マークされたトランザクションが完了すると、「rollback only」マークは除去されるため、セッションで開始される次のトランザクションはコミットされます。

isMarkedRollbackOnly メソッド

セッションが現在「rollback only」とマークされている場合に返されます。 markRollbackOnly メソッドが以前このセッションで呼び出されており、セッションで開始されたトランザクションがアクティブな場合、かつこの場合に限り、このメソッドによってブール値 true が返されます。

setTransactionTimeout メソッド

このセッションで開始される次のトランザクションのトランザクション・タイムアウトを特定の秒数に設定します。このメソッドは、このセッションで以前に開始されたトランザクションのトランザクション・タイムアウトには影響を与えません。このメソッドが呼び出された後に開始されたトランザクションにのみ影響を与えます。このメソッドが呼び出されない場合は、com.ibm.websphere.objectgrid.ObjectGrid メソッドの setTxTimeout メソッドに渡されたタイムアウト値が使用されます。

getTransactionTimeout メソッド

このメソッドは、トランザクション・タイムアウト値 (秒単位) を返します。タイムアウト値として setTransactionTimeout メソッドに渡された最後の値は、このメソッドによって返されます。 setTransactionTimeout メソッドが呼び出されない場合は、com.ibm.websphere.objectgrid.ObjectGrid メソッドの setTxTimeout メソッドに渡されたタイムアウト値が使用されます。

transactionTimedOut

このメソッドは、このセッションで開始された現行トランザクションがタイムアウトになると、ブール値 true を返します。

isFlushing メソッド

このメソッドは、呼び出されたセッション・インターフェースの flush メソッドの結果として、すべてのトランザクション変更が Loader プラグインにフラッシュされる場合、かつこの場合に限り、ブール値 true を返します。 Loader プラグインでは、batchUpdate メソッドが呼び出された理由を確認する必要がある場合にこのメソッドが役立ちます。

isCommitting メソッド

このメソッドは、呼び出されたセッション・インターフェースの commit メソッドの結果として、すべてのトランザクション変更がコミットされる場合、かつこの場合に限り、ブール値 true を返します。 Loader プラグインでは、batchUpdate メソッドが呼び出された理由を確認する必要がある場合にこのメソッドが役立ちます。

setRequestRetryTimeout メソッド

このメソッドは、セッションの要求再試行タイムアウト値 (ミリ秒) を設定します。クライアントが要求再試行タイムアウトを設定してある場合、セッション設定値がクライアント値をオーバーライドします。

getRequestRetryTimeout メソッド

このメソッドは、セッションの現行の要求再試行タイムアウト設定を取得します。値 -1 は、タイムアウトが設定されていないことを表します。値 0 は、フェイル・ファースト・モードであることを表します。0 より大きい値は、ミリ秒単位のタイムアウト設定値です。

親トピック: [データの索引付けのためのプラグイン](#)

HashIndex プラグインの構成

組み込み HashIndex である `com.ibm.websphere.objectgrid.plugins.index.HashIndex` クラスは、動的索引を使用してプログラマチックに、構成することができます。

このタスクについて

複合索引の構成は、XML を使用した 通常の索引の構成と同じですが、**attributeName** プロパティ値は例外です。複合索引の場合、**attributeName** プロパティの値は、コンマ区切りの属性のリストです。例えば、値クラス `Address` は、`city`、`state`、および `zipcode` の 3 つの 属性を持つとします。この場合、"`city,state,zipcode`" という **attributeName** プロパティ値を 使用して複合索引を定義し、複合索引に `city`、`state`、および `zipcode` が含まれていることを 示すことができます。

また、複合 HashIndexes は、範囲検索をサポートしないため、`RangeIndex` プロパティを `true` に設定しないよう注意してください。

手順

プログラムで複合索引を構成します。動的索引にのみ適用されます。

次のサンプル・コードも同じ複合索引を作成します。

```
        HashIndex mapIndex = new HashIndex();
        mapIndex.setName("Address.CityStateZip");
        mapIndex.setAttributeName(("city,state,zipcode"));
        mapIndex.setRangeIndex(true);

BackingMap bm = objectGrid.getMap("mymap");
    bm.createDynamicIndex(mapIndex, null);

    try {
        // If not using DynamicIndexCallback, need to wait for the Index to be ready.
        // The waiting time depends on the current size of the map
        Thread.sleep(3000);
    } catch (Throwable t) {
        // ...
    }

    // When the index is ready, applications can try to get application index
    // interface instance.
    // Applications have to find a way to ensure that the index is ready to use,
    // if not using DynamicIndexCallback interface.
    // The following example demonstrates the way to wait for the index to be ready
    // Consider the size of the map in the total waiting time.

    Session session = objectGrid.getSession();
    ObjectMap m = session.getMap("mymap");
    MapRangeIndex codeIndex = null;

    int counter = 0;
    int maxCounter = 10;
    boolean ready = false;
    while (!ready && counter < maxCounter) {
        try {
            counter++;
            codeIndex = (MapRangeIndex) m.getIndex("Address.CityStateZip");
            ready = true;
        } catch (IndexNotReadyException e) {
            // implies index is not ready, ...
            System.out.println("Index is not ready. continue to wait.");
            try {
                Thread.sleep(3000);
            } catch (Throwable tt) {
                // ...
            }
        } catch (Throwable t) {
            // unexpected exception
        }
    }
}
```

```
        t.printStackTrace();
    }
}

if (!ready) {
    System.out.println("Index is not ready. Need to handle this situation.");
}
```

HashIndex プラグイン属性

次の属性を使用して、HashIndex プラグインを構成できます。

複合索引の使用

複合 HashIndex により、照会のパフォーマンスが向上し、高いコストがかかる マップのスキャンを避けることができます。また、この機能は、検索条件に多くの属性が関係する際、キャッシュ・オブジェクトを検索するための便利な方法を HashIndex API に提供します。

2.5+ グローバル索引の使用

グローバル索引を実装することで、大規模な区画化環境 (例えば、100 個の区画を含む環境) におけるデータ検索パフォーマンスを向上させることができます。

親トピック: [データの索引付けのためのプラグイン](#)

HashIndex プラグイン属性

次の属性を使用して、HashIndex プラグインを構成できます。

属性

Name

索引の名前を指定します。名前は各マップで固有でなければなりません。この名前は、バッキング・マップのオブジェクト・マップ・インスタンスから索引オブジェクトを取り出すのに使用されます。

AttributeName

索引に対する属性の名前をコンマで区切ったリストを指定します。フィールド・アクセス索引の場合、属性名はフィールド名と同じです。プロパティ・アクセス索引の場合、属性名は JavaBean 互換のプロパティ名です。属性名が 1 つだけであれば、HashIndex は単一属性索引です。この属性がリレーションシップの場合は、リレーションシップ索引でもあります。複数の属性名が含まれている場合は、HashIndex は複合索引です。

FieldAccessAttribute

非エンティティ・マップに使用されます。true の場合、フィールドを使用してオブジェクトに直接アクセスします。指定されていないか、false の場合、データのアクセスには、属性の getter メソッドが使用されます。

GlobalIndexEnabled

これが true に設定されると、グローバル索引が使用可能になり、アプリケーションは取得した索引オブジェクトを MapGlobalIndex インターフェースにキャストすることができます。

HashIndex の GlobalIndexEnabled プロパティが true に設定されると、HashIndex 構成のほかに MapGlobalIndex インターフェースもサポートするように HashIndex のグローバル索引機能が使用可能になります。大規模な区画化環境でデータを効率的に検索する方法を提供します。

POJOKeYIndex

非エンティティ・マップに使用されます。true の場合、索引はマップのキー部分でオブジェクトをイントロスペクトします。この設定は、キーが複合キーで、値にキーが組み込まれていない場合に役立ちます。指定されていないか、false の場合、索引はマップの値部分でオブジェクトをイントロスペクトします。

RangIndex

true の場合、範囲索引付けが使用可能にされ、アプリケーションは取り出された索引オブジェクトを MapRangIndex インターフェースにキャストできます。RangeIndex プロパティが false と構成されている場合は、アプリケーションは取り出された索引オブジェクトを MapIndex インターフェースにしかキャストできません。

単一属性 HashIndex と複合 HashIndex

HashIndex の AttributeName プロパティに複数の属性名が含まれている場合、HashIndex は複合索引です。そうではなく、含まれている属性名が 1 つのみの場合は、単一属性索引です。例えば、AttributeName プロパティ値が city, state, zipcode であるような、複合 HashIndex が考えられます。この例では、3 つの属性がコンマで区切られています。もし AttributeName プロパティ値が単に zipcode であれば、属性は 1 つだけなので、単一属性 HashIndex であるということになります。

複合 HashIndex は、検索条件に多くの属性が関係するような場合に、キャッシュ・オブジェクトを検索する効果的な方法を提供します。ただし、範囲索引はサポートしないため、RangIndex プロパティは false に設定されている必要があります。

詳しくは、[複合索引の使用](#)を参照してください。

リレーションシップ HashIndex

単一属性 HashIndex の索引属性が、単一値または複数値のリレーションシップの場合、その HashIndex はリレーションシップ HashIndex です。リレーションシップ HashIndex の場合、HashIndex の RangIndex プロパティは「false」に設定されている必要があります。

キー HashIndex

非エンティティ・マップの場合、HashIndex の POJOKeYIndex プロパティが true に設定されていると、HashIndex はキー HashIndex であり、エントリーのキー部分が索引付けに使用されます。HashIndex の AttributeName プロパティが指定されていないと、キー全体に索引が付けられます。指定されていると、キー HashIndex は単一属性 HashIndex にしかありません。

範囲 HashIndex

HashIndex の RangIndex プロパティが true に設定されている場合、その HashIndex は範囲索引であり、MapRangIndex インターフェースをサポートできます。MapRangIndex は、範囲関数 greater than や less than、あるいは両方を使用するデータ検出をサポートしますが、MapIndex は equals 関数のみをサポートします。単

一属性索引の場合、**RangeIndex** プロパティを true に設定できるのは、索引付けられる属性のタイプが Comparable の場合に限りです。単一属性索引が照会によって使用される場合、RangeIndex プロパティは true に設定されていなければならない、索引付けられる属性のタイプは Comparable でなければなりません。リレーションシップ HashIndex および複合 HashIndex の場合、RangeIndex プロパティは false に設定されていなければならない。

RangeIndex プロパティの値が true なので、前記の例は範囲 HashIndex です。

以下の表に、範囲索引の使用についての要約を示します。

表 1. 範囲索引のサポート。HashIndex のタイプが範囲索引をサポートするかどうかを記述します。

HashIndex タイプ	範囲索引のサポート
単一属性 HashIndex: 索引付けられるキーまたは属性のタイプは Comparable である	はい
単一属性 HashIndex: 索引付けられるキーまたは属性のタイプは Comparable でない	いいえ
複合 HashIndex	いいえ
リレーションシップ HashIndex	いいえ

親トピック: [HashIndex プラグインの構成](#)

複合索引の使用

複合 HashIndex により、照会のパフォーマンスが向上し、高いコストがかかる マップのスキャンを避けることができます。また、この機能は、検索条件に多くの属性が関係する際、キャッシュ・オブジェクトを検索するための便利な方法を HashIndex API に提供します。

パフォーマンスの改善

複合 HashIndex を使用すると、一致検索条件に入れた複数の属性によって、キャッシュ・オブジェクトを高速かつ簡単に見つけることができます。複合索引は、完全属性一致検索をサポートしますが、範囲検索はサポートしません。

注: 複合索引は ObjectGrid 照会言語での BETWEEN 演算子の使用をサポートしません。BETWEEN は範囲サポートを必要とすることがあるためです。より大 (>)、より小 (<) 条件も、範囲索引を必要とするため機能しません。

複合索引の構成

複合索引は動的索引としてプログラマチックに構成することができます。

プログラマチック構成

次の例は複合索引を作成するものです。

```
HashIndex mapIndex = new HashIndex();
mapIndex.setName("Address.CityStateZip");
mapIndex.setAttributeName(("city,state,zipcode"));
mapIndex.setRangeIndex(false);

BackingMap bm = objectGrid.getMap("mymap");
bm.createDynamicIndex(mapIndex, null);
```

複合索引の構成は、XML を使用した 通常の索引の構成と同じですが、attributeName プロパティ値は例外なので注意してください。複合索引の場合、attributeName の値は、コンマ区切りの属性のリストです。例えば、値クラス Address は、city、state、および zipcode の 3 つの属性を持つとします。この場合、"city,state,zipcode" という attributeName プロパティ値を使用して複合索引を定義し、複合索引に city、state、および zipcode が含まれていることを示すことができます。

複合 HashIndexes は、範囲検索をサポートしないため、true に設定された RangeIndex プロパティを持つことができません。

複合索引の検索の実行

複合索引が構成されたら、アプリケーションは、MapIndex インターフェースの findAll(Object) メソッドを使用して、検索を実行できます。

2.5+ 制約事項: MapIndex.EMPTY_VALUE は複合グローバル索引に対してはサポートされません。

```
Session sess = objectgrid.getSession();
ObjectMap map = sess.getMap("MAP_NAME");
MapIndex codeIndex = (MapIndex) map.getIndex("INDEX_NAME");
Object[] compositeValue = new Object[]{ MapIndex.EMPTY_VALUE,
                                         "MN", "55901"};
Iterator iter = mapIndex.findAll(compositeValue);
// Close the session (optional in Version 7.1.1 and later) for improved performance
sess.close();
```

MapIndex.EMPTY_VALUE は compositeValue[0] に割り当てられ、評価から city 属性が除外されることを示します。結果には、state 属性が"MN" に等しく、zipcode 属性が"55901" に等しいオブジェクトのみが含まれます。

マイグレーションおよびインターオペラビリティ

複合索引の使用に関する唯一の制約は、異種のコンテナがある分散環境では、アプリケーションが複合索引を構成できないことです。古いコンテナ・サーバーと新しいコンテナ・サーバーを混用することはできません。というのは、古いほうのコンテナ・サーバーは複合索引構成を認識しないからです。複合索引は、既存の通常の属性索引とよく似ていますが、複合索引では、複数の属性にまたがる索引付けが許可される点が異なります。通常の属性索引のみを使用する場合、コンテナ混在環境はそのまま存続できます。

親トピック: [HashIndex プラグインの構成](#)

グローバル索引の使用

2.5+ グローバル索引を実装することで、大規模な区画化環境 (例えば、100 個の区画を含む環境) におけるデータ検索パフォーマンスを向上させることができます。

このフィーチャーはまた、索引付き属性の所在を検索する手段を提供し、索引付き属性に関連するエージェント操作や照会操作を向上させることができます。グローバル索引の機能については、MapGlobalIndex API 資料を参照してください。

パフォーマンスの改善

大規模な区画化環境では、キャッシュ・オブジェクトがすべての区画に分散されます。索引、照会、またはエージェントを使用したデータの検索で完全な結果を得るためには、すべてのサーバーに対して検索を実行する必要があります。この種の検索は、各区画をロードして検索するために必要なリモート・コールが発生するので低速になります。その上、検索条件を満たすデータがすべての区画にあるとは限りません。グローバル索引は、一致するデータを実際に含む区画に対してのみ検索を実行するため、検索パフォーマンスを向上させます。グローバル索引フィーチャーは、索引付き属性の所在を追跡したり、すべての区画から属性に合った適用区画を決定したりすることができます。通常、適用区画は全区画のサブセットです。したがって、適用区画での索引、照会、およびエージェントの実行は、グローバル索引によって相殺されることでさえ、すべての区画でこれらの項目を実行するときよりもはるかに速くなります。

データの検索

アプリケーションはキーを使用してデータを検索することができます。アプリケーションは索引を使用したデータを検索することもできますが、これは、当該データが 1 つ以上の属性を持ち、それらの属性に対して索引が定義されていることが前提となります。従来から、アプリケーションは、クライアント索引プロキシを使用してすべての区画からエントリー・キーを取得することもできれば、エージェントを使用してすべての区画で索引検索を実行し、キャッシュ・キー、値、またはその両方を返すこともできます。グローバル索引フィーチャーを使用すれば、アプリケーションは、MapGlobalIndex API を通じて、操作を適用区画のみで実行する効率的な方法によりエントリー・キー、値、またはその両方を検索することができます。

エージェント操作

例えば、エージェント操作が索引付き属性に関連している場合は、索引付き属性を使用してエントリーを無効化することにより、アプリケーションはグローバル索引を使用してまず属性で適用区画を検索することができます。その後、アプリケーションはエージェントをこれらの適用区画に送ることができます。MapGlobalIndex.findPartitions() メソッドを使用して、属性によって適用区画を検索することができます。

グローバル索引の使用可能化

グローバル索引は HashIndex プラグインの拡張機能であり、既存の任意の HashIndex 構成で使用可能にすることができます。

グローバル索引検索の実行

グローバル索引機能は MapGlobalIndex API で定義されます。HashIndex プラグインでグローバル索引が使用可能になれば、アプリケーションは、取得した索引プロキシを MapGlobalIndex タイプにキャストし、グローバル索引を開始することができます。

```
// in client ObjectGrid process
MapGlobalIndex mapGlobalIndexCODE = (MapGlobalIndex)m.getIndex("CODE", false);
Object[] attributes = new Object[] {new Integer(1)};
Collection partitions = mapGlobalIndexCODE.findPartitions(attributes);
Set keys = mapGlobalIndexDependency.findKeys(attributes);
Set values = mapGlobalIndexDependency.findValues(attributes);
Map entries = mapGlobalIndexDependency.findEntries(attributes);
```

マイグレーションおよびインターオペラビリティ

グローバル索引を使用する際の唯一の制約は、異種コンテナを含む分散環境ではアプリケーションがグローバル索引を構成できないということです。古いコンテナ・サーバーと新しいコンテナ・サーバーは混在できません。これは、古いコンテナ・サーバーはグローバル索引構成および複合グローバル索引構成を認識しないからです。

グローバル索引または複合グローバル索引を使用するためには、まず、アプリケーションで使用しているコンテナ・サーバーとコンテナ・クライアントをすべて停止する必要があります。その後、HashIndex 構成でグローバル索引を使用可能にし、コンテナ・サーバーとコンテナ・クライアントを再始動します。

親トピック: [HashIndex プラグインの構成](#)

クライアントへの連続照会を使用したマップ更新の通知

2.5+ データ・グリッドでオブジェクトまたはエントリーが挿入または更新されたときに、クライアント Java™ 仮想マシン (JVM) で通知を受けることができます。

始める前に

連続照会を使用する場合は、コンテナ・サーバーとクライアント間の通信に使用されるトランスポート・メカニズムである IBM® eXtremeIO を使用可能にする必要があります。eXtremeIO の使用可能化について詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。

このタスクについて

データ・グリッドと対話するクライアント・アプリケーションを開発する場合は、フィルター条件に一致するエントリーが挿入、更新、または削除されたときに自動的にリアルタイムの結果を取得する照会が必要になることがあります。例えば、頻繁な更新が必要な株価アプリケーションを開発するとします。この更新は、株式市場で行われた変更を反映します。そのため、正確でタイムリーな結果を提供できるように、アプリケーションが変更について即時に通知を受けるのが重要になります。連続照会は低メモリー・フットプリントであり、データ・グリッド内で変更が行われたときに能動的にクライアントに通知することができます。

以下の手順を使用して、連続照会を使用するようにクライアント・アプリケーションをプログラミングします。

制約事項: 値属性パス null を指定する照会は、値オブジェクトがストリングや整数などのプリミティブ Java 型でなければサポートされません。null が指定されると、値オブジェクト全体を照会するために照会フィルターが使用されます。

手順

1. クライアント・アプリケーションで連続照会マネージャーを呼び出します。例えば、以下のコード行を挿入します。

```
ContinuousQueryManager cqMan = ContinuousQueryManagerFactory.getManager(og);
```

2. フィルターまたはフィルター・チェーンを定義します。独自のフィルターを実装するか、提供されている基本フィルター AND、OR、LT、GT、EQ などを使用できます。インスタンス化されたフィルターまたはフィルター・チェーンには、固有 ID が付与されます。サポートされるすべてのフィルターについては、[Java API 資料へのアクセス](#)を参照して、連続照会 API を確認してください。

以下のコード例で、等価 (EQ) 基本フィルターを使用する一つの方法を示します。データ・グリッドに、フィールド firstName がある Customer オブジェクトが含まれているものと仮定します。フィルターは、firstName が Larry に等しいと、true を返します。

```
EQFilter<String, String> equalsFilter = new EQFilter<String, String>("firstName", "Larry");
```

3. 前のステップで作成したフィルターを使用して照会を定義します。例えば、以下のようになります。

```
ContinuousQueryTopic<String, Customer> topic =  
    cqMan.<String, Customer> defineContinuousQuery("myMapName", equalsFilter, true,  
    true, true);
```

4. オプション: 連続照会キャッシュを取得して、連続照会のクライアント・サイドの結果にアクセスします。照会が鍵のみの照会として定義されている場合は、照会を満たす鍵のみが、連続照会キャッシュに入ります。例:

```
ContinuousQueryCache cache = topic.getCache();
```

5. オプション: さらに、ContinuousQueryTopic インスタンスを使用して ContinuousQueryListener インターフェースを実装するクラスを登録して、連続照会の結果が変更されたときに通知を受け取ることができます。addListener メソッドを呼び出してリスナーを登録します。例えば、以下のようになります。

```
ContinuousQueryListener<String, Customer> listener = new MyCQListener<String,  
Customer>();  
topic.addListener(listener);
```

次のタスク

連続照会 API について詳しくは、[API 資料: パッケージ com.ibm.websphere.objectgrid.continuousquery](#) を参照してください。

親トピック: [Java API を使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発

Representational State Transfer (REST) ゲートウェイを使用して、集合がホスティングする単純データ・グリッドにアクセスできます。この REST ゲートウェイは、非 Java 環境からデータ・グリッドにアクセスする必要があるときに便利です。

始める前に

- REST ゲートウェイは、WebSphere® DataPower® XC10 アプライアンス、バージョン 8.6 以降で使用できます。
- アプライアンス上に単純データ・グリッドを作成する必要があります。単純データ・グリッドの作成について詳しくは、[単純データ・グリッドの作成](#)を参照してください。

このタスクについて

REST ゲートウェイを使用して、DataPower XI50 アプライアンスや .NET アプリケーションなどの非 Java 環境から単純データ・グリッドのデータにアクセスします。REST ゲートウェイを使用すると、Java ベースの ObjectMap API が使用する IBM® オブジェクト・リクエスト・ブローカー (ORB) をホスティングできない Java™ 仮想マシンからマップ・データにアクセスすることもできます。

トランザクション

WebSphere DataPower XC10 アプライアンスに対する各 REST 操作により、データ・グリッドへの独立したトランザクションが開始して終わります。複数の操作をチェンニングして、単一トランザクションにまとめることはできません。

ロード・バランシング

REST ゲートウェイを使用するときは、クライアントの責任において、WebSphere DataPower XC10 アプライアンス集合への要求のロード・バランスを取る必要があります。クライアント・プログラムでは、外部ロード・バランサーを使用するか、使用する HTTP クライアントにロジックを追加できます。

セキュリティ

REST ゲートウェイを介した通信では、データ・グリッドでセキュリティを有効にしていなくても、常にセキュアな構成が実現します。データ・グリッドへのアクセスに使用するユーザー・グループを構成し、このグループにデータ・グリッドへの all 許可限を割り当ててください。

WebSphere eXtreme Scale REST データ・サービスへの関係

REST ゲートウェイは、Microsoft ADO.NET データ・サービス・インターフェースを実装する WebSphere eXtreme Scale REST データ・サービスとは別個のエンティティです。

2.5+ グリッド別名

複数のデータ・グリッドに同時にデータを取り込む必要がある場合、REST ゲートウェイを使用して、グリッド別名を作成し、管理することができます。グリッド別名を使用すると、グリッドを切り替え、それらのグリッドに同時にデータを取り込むことができます。例えば、グリッド A に対してデータ・グリッド操作を実行したいが、もう 1 つのグリッドであるグリッド B に切り替えてデータを取り込むための間接層も必要であるという場合があります。この場合、グリッド A を指すグリッド C という別名を作成することができます。この別名を使用して、グリッド A に対してデータ・グリッド操作を実行し、同時にグリッド B にデータを取り込むことができます。グリッド A に対して (グリッド C という別名を使用して) データ・グリッド操作を実行すると同時に、別名をグリッド A からグリッド B を指すように切り替えることができます。グリッド別名は、作成、照会、あるいは REST リソース `resources/gridaliases` から削除することができます。REST ゲートウェイを使用したグリッド別名の管理について詳しくは、[REST ゲートウェイ: REST 操作](#)を参照してください。

[REST ゲートウェイ: URI フォーマット](#)

特定のフォーマットで URI を指定すると、単純データ・グリッドにアクセスし、操作を実行できます。

[REST ゲートウェイ: データ・フォーマット](#)

REST ゲートウェイは、HTTP 要求内の Content-Type ヘッダーを使用して、データ・グリッドに保管するデータのデータ・フォーマットを判定します。

[REST ゲートウェイ: REST 操作](#)

HTTP POST、GET、および DELETE 操作を使用して、データ・グリッドのデータの挿入、更新、取得、および削除を行います。REST ゲートウェイは、ご使用のデータ・グリッドを指すグリッド別名を管理するための HTTP 要求もサポートします。グリッド別名は、複数のデータ・グリッドに一度にデータを取り込んだり、データ・グリッドを切り替えたりする必要があるときに便利です。グリッド別名は作成、照会、および削除することができます。REST リソース `/resource/gridaliases` を使用します。

[REST ゲートウェイの例: データ・グリッドのマップ項目の挿入と取得](#)

HTTP メソッドの POST および GET を使用して、データ・グリッドのマップ項目を挿入および取得できます。

REST ゲートウェイの例: Java クライアントから REST マップへのデータの挿入およびデータへのアクセス (ObjectMap API を使用)

REST ゲートウェイを使用してデータがマップに挿入される場合、提供されたコンテンツ・タイプおよび要求本体をラップするためにタイプ `com.ibm.websphere.xsa.RestValue` のラッパー・クラスが使用されます。同じ `RestValue` クラスを使用して、Java クライアントからデータを挿入したり、マップからデータを取得したりすることができます。これには ObjectMap API を使用します。

REST ゲートウェイの例: データ・グリッドのマップ項目のクリア

REST ゲートウェイの HTTP DELETE メソッドを使用して、データ・グリッド内の マップをクリアできます。

REST ゲートウェイの例: 動的マップの作成

単純データ・グリッドを作成すると、同じ名前のデフォルト・マップがデフォルトで作成されます。アプリケーションの必要に応じて、マップ・テンプレートを使用して追加のマップを作成することもできます。

REST ゲートウェイの例: 存続時間 (TTL) の有効期限

TTL 値は、最終更新時間 (*.LUT) マップと最終アクセス時間 (*.LAT) マップの 両方に設定できます。デフォルトの TTL は、いずれのマップ・タイプも 1 時間です。

REST ゲートウェイ: セキュリティー構成

REST ゲートウェイを介してデータ・グリッドにアクセスするには、データ・グリッドのセキュリティーが使用可能に設定されているかどうかに関わらず、ユーザーは WebSphere DataPower XC10 アプライアンス に認証される必要があります。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを 常に提供しなければなりません。REST ゲートウェイを介してデータ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。

REST ゲートウェイ: HTTP セッションと Cookie

Set-Cookie: ヘッダーを指定し、HTTP セッションと Cookie を REST ゲートウェイで使用します。

親トピック: [単純データ・グリッドにアクセスするアプリケーションの開発](#)

関連タスク:

[単純データ・グリッドの作成](#)

関連資料:

[クライアント・プロパティ・ファイル](#)

[動的マップの構成オプション](#)

REST ゲートウェイ: URI フォーマット

特定のフォーマットで URI を指定すると、単純データ・グリッドにアクセスし、操作を実行できます。

URI フォーマット

WebSphere® DataPower® XC10 アプライアンス上の単純データ・グリッドにアクセスするための REST URI フォーマットは次のとおりです。

```
/resources/datacaches/[grid_name]/[map_name]/[key]
```

デフォルトのコンテキスト・ルートは resources です。

MyDataGrid という名前の単純データ・グリッドを、ホスト名 myxc10.ibm.com のアプライアンス上に作成した場合、キー名 my.data.item にアクセスするための URL は、結果として次のようになります。

```
http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyMap/my.data.item
```

前の例では、MyDataGrid グリッド内の MyMap デフォルト・マップが使用されます。このデフォルト・マップに 存続時間 (TTL) の期限はありません。データ・グリッド内に配置された項目は、明示的に削除されるまでデータ・グリッド内にとどまります。TTL の期限を構成するには、[REST ゲートウェイの例: 存続時間 \(TTL\) の有効期限](#)を参照してください。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイ: データ・フォーマット

REST ゲートウェイは、HTTP 要求内の Content-Type ヘッダーを使用して、データ・グリッドに 保管するデータのデータ・フォーマットを判定します。

データ・フォーマット

REST ゲートウェイは、HTTP 要求内の Content-Type ヘッダーを使用して、データ・グリッド内に 保管するデータのデータ・フォーマットを判定します。タイプが application/xml のコンテンツを挿入した場合に、アプリケーションが同じキャッシュ・キーに対して GET 操作を 実行すると、応答の本体と Content-type は同じフォーマット・タイプになります。この例の場合、応答の本体は application/xml フォーマットになります。複数のコンテンツ・タイプのデータを同一データ・グリッドに保管できます。以下に、有効なコンテンツ・タイプの例をいくつか示します。

表 1. HTTP 要求内の content-type ヘッダーのコンテンツ・タイプ

コンテンツ・タイプ	用途
application/xml	XML
application/json	JavaScript データ
application/octet-stream	シリアライズド・オブジェクト、汎用データ

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイ: REST 操作

HTTP POST、GET、および DELETE 操作を使用して、データ・グリッドのデータの挿入、更新、取得、および削除を行います。REST ゲートウェイは、ご使用のデータ・グリッドを指すグリッド別名を管理するための HTTP 要求もサポートします。グリッド別名は、複数のデータ・グリッドに一度にデータを取り込んだり、データ・グリッドを切り替えたりする必要のあるときに便利です。グリッド別名は作成、照会、および削除することができ、REST リソース `/resource/gridalias` を使用します。

データ・グリッドにデータを取り込むための REST 操作

表 1. 操作、それに相当する HTTP メソッド、および応答コードの定義

操作	HTTP メソッド	応答コード
挿入または更新	POST	<ul style="list-style-type: none">200 CREATED: データは正常にデータ・グリッドに挿入されたか、更新されました。400 BAD REQUEST: データの挿入操作または更新操作は正常に完了しませんでした。
取得	GET	<ul style="list-style-type: none">200 OK: 直前の挿入操作または更新操作の応答の本体と content-type が取得されました。404 NOT FOUND: 指定されたキーがデータ・グリッド内に存在しません。400 BAD REQUEST: アプライアンスは要求を処理できませんでした。
削除	DELETE	<ul style="list-style-type: none">200 NO CONTENT: 項目がデータ・グリッドから削除されました。400 BAD REQUEST: アプライアンスは要求を処理できませんでした。

2.5+

データ・グリッドの別名を管理するための REST 操作

表 2. 操作、それに相当する HTTP メソッド、および応答コードの定義

操作	HTTP メソッド	応答コード
別名の追加または更新	POST <code>/resources/gridalias/ <aliasname>?src= <sourceGridName></code>	<ul style="list-style-type: none">204 SUCCESS: グリッド別名が正常に作成されました。401 SOURCE GRID DOES NOT EXIST: 別名が指すデータ・グリッドが存在しないため、別名を作成できませんでした。409 DATA GRID EXISTS: 同じ名前を持つデータ・グリッドが既に存在しているため、別名を作成できませんでした。
指定した別名を持つ現行データ・グリッドを取得	GET <code>/resources/gridalias/ <aliasname></code>	<ul style="list-style-type: none">200 OK
すべての別名のリストを取得	GET <code>/resources/gridalias/</code>	<ul style="list-style-type: none">200 OK
別名の削除	DELETE <code>/resources/gridalias/ <aliasname></code>	<ul style="list-style-type: none">204 SUCCESS: 別名が削除されます。404 BAD REQUEST: 別名が存在していません。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: データ・グリッドのマップ項目の挿入と取得

HTTP メソッドの POST および GET を使用して、データ・グリッドのマップ項目を挿入および取得できます。

例: 挿入操作

定義済み URI とデータ・フォーマットを使用して、データ・グリッドに情報を挿入できます。次の例では、キー「bob」を MyGrid グリッドと MyGrid マップに挿入します。

```
POST /resources/datacaches/MyGrid/MyGrid/bob
Content-type: application/xml
<mydata>this is some data</mydata>
```

例: 取得操作

前の例で挿入したキーを取得するには、次の URI を使用できます。

```
GET /resources/datacaches/MyGrid/MyGrid/bob
```

GET 操作は、個々のキーに対して実行しなければなりません。すべてのマップ項目を取得することはできません。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: Java クライアントから REST マップへのデータの挿入およびデータへのアクセス (ObjectMap API を使用)

REST ゲートウェイを使用してデータがマップに挿入される場合、提供されたコンテンツ・タイプおよび要求本体をラップするためにタイプ `com.ibm.websphere.xsa.RestValue` のラッパー・クラスが使用されます。同じ `RestValue` クラスを使用して、Java クライアントからデータを挿入したり、マップからデータを取得したりすることができます。これには `ObjectMap` API を使用します。

REST マップにアクセスするための Java クライアント・コード

```
RestValue rv = new RestValue();
rv.setContentType("application/xml");
String myXml("<customer>brian</customer>");
rv.setValue(myXml.getBytes("UTF8"));
ogSession.begin();
ObjectMap map = ogSession.getMap("myMap.LUT");
map.insert("brian", rv);
ogSession.commit();
```

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: データ・グリッドのマップ項目のクリア

REST ゲートウェイの HTTP DELETE メソッドを使用して、データ・グリッド内の マップをクリアできます。

個々の項目のクリア

個々の項目を削除するには、DELETE メソッドとオブジェクトのキー名を使用します。

```
DELETE http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyDataGrid/my.data.item
```

データ・グリッド上のマップ全体のクリア

データ・グリッド内のマップ全体をクリアするには、URI のキー部分を省略して HTTP DELETE メソッドを使用します。例えば、MyDataGrid データ・グリッド上の MyDataMap.LUT マップをクリアするには、次の操作を使用します。

```
DELETE http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyDataMap.LUT
```

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: 動的マップの作成

単純データ・グリッドを作成すると、同じ名前のデフォルト・マップがデフォルトで作成されます。アプリケーションの必要に応じて、マップ・テンプレートを使用して追加のマップを作成することもできます。

動的マップの作成

あるマップに対して最初の操作を実行したとき、そのマップがマップ・テンプレートと一致しても、まだ作成されていないマップであると、新規動的マップが作成されます。例えば、*.LUT テンプレートを使用する新規動的マップを MyMap.LUT という名前で作成するには、GET、DELETE、または POST 操作に次の URI を使用します。

```
http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyMap.LUT/a.key
```

動的マップを命名する方法については、[動的マップの構成オプション](#) を参照してください。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイの例: 存続時間 (TTL) の有効期限

TTL 値は、最終更新時間 (*.LUT) マップと最終アクセス時間 (*.LAT) マップの両方に設定できます。デフォルトの TTL は、いずれのマップ・タイプも 1 時間です。

例

最終更新時間 (*.LUT) マップと最終アクセス時間 (*.LAT) マップのいずれに TTL 値を設定する場合も、値を秒数で指定した TTL 要求パラメーターを提供します。例えば、a.key キーに対して TTL 値 600 秒を設定するには、HTTP POST メソッドを使用して値をデータ・グリッドに挿入または更新するときに ttl 要求パラメーターを指定します。

```
http://myxc10.ibm.com/resources/datacaches/MyDataGrid/MyMap.LUT/a.key?ttl=600
```

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

REST ゲートウェイ: セキュリティー構成

REST ゲートウェイを介してデータ・グリッドにアクセスするには、データ・グリッドのセキュリティーが使用可能に設定されているかどうかに関わらず、ユーザーは WebSphere® DataPower® XC10 アプライアンス に認証される必要があります。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを常に提供しなければなりません。REST ゲートウェイを介してデータ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。

認証と許可

REST ゲートウェイを介してデータ・グリッド・マップにアクセスするには、ユーザーまたはユーザー・グループが認証される必要があります。また URI 内で指定したデータ・グリッドにアクセスする権限が必要です。データ・グリッドでセキュリティーを構成していない場合でも、REST ゲートウェイを介した通信に使用するユーザー・グループを構成し、このグループにデータ・グリッドへの all アクセス権限を割り当てる必要があります。データ・グリッドへのアクセス権限の構成について詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。アプリケーション・クライアントは、許可があるユーザー ID とパスワードを HTTP 要求の HTTP ヘッダー内に指定し、Basic 許可ヘッダーを提供しなければなりません。

```
Authorization: Basic <base64 encoded string of "userid:password">
```

Basic 許可ヘッダーの形式について詳しくは、[Wikipedia: Basic access authentication](#) を参照してください。

セキュア・データ・グリッド

セキュア・データ・グリッドの構成で REST ゲートウェイを使用できます。セキュア・データ・グリッドにアクセスするには、許可ヘッダー内にユーザー ID とパスワードを指定します。ユーザーが認証される必要があります。また URI 内で指定したデータ・グリッドにアクセスする権限が必要です。

表 1. セキュア・データ・グリッド

許可	取得	ポスト	削除
READ	X		
WRITE	X		
CREATE	X	X	
ALL	X	X	X

トランスポート・セキュリティー

REST ゲートウェイを使用するクライアントは、トランスポート・セキュリティーが必要な場合、HTTPS プロトコルを使用できます。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

[ユーザー許可](#)

[xcadminパスワード](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

REST ゲートウェイ: HTTP セッションと Cookie

Set-Cookie: ヘッダーを指定し、HTTP セッションと Cookie を REST ゲートウェイで使⽤します。

REST ゲートウェイ・コードは、現在セッション中でないクライアントから要求を受け取ると、HTTP セッションを作成します。不必要なセッションの作成を避け、最高のパフォーマンスを得るため、REST クライアントは、REST ゲートウェイから Set-Cookie: ヘッダーを使用して返される Cookie を保存し、後続の要求ではそれらと同じ Cookie を、Cookie: ヘッダーを使用して REST ゲートウェイに提供しなければなりません。

親トピック: [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

.NET API を使用したデータ・グリッド・アプリケーションの開発

2.5+ Java™ アプリケーションと同じデータ・グリッドを使用する Microsoft .NET アプリケーションを開発できます。

.NET [.NET 開発環境の設定](#)

Microsoft Visual Studio で WebSphere® eXtreme Scale クライアント for .NET を使用するには、開発環境をインストールし、WebSphere eXtreme Scale クライアント for .NET アセンブリーを使用するようにプロジェクトを構成する必要があります。

.NET [.NET API を使用した動的マップの作成](#)

データ・グリッドがインスタンス化されたら、.NET API を使用して動的マップを作成できます。マップ・テンプレートの事前定義セットに基づいて、動的にマップをインスタンス化することができます。

.NET [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)

ClassAlias および FieldAlias 注釈を使用して、Java と .NET クラス間でのデータ・グリッド・データの共有を使用可能にします。

.NET [PartitionKey 注釈を使用したキーから区画へのマップ](#)

PartitionKey 別名を使用して、データが保存される区画を判別するためにハッシュ・コード計算を実行するフィールドまたは属性を識別します。PartitionKey 注釈は、キー属性でのみ有効です。

.NET [.NET アプリケーションにおけるトランザクションのためのプログラミング](#)

トランザクションを必要とする .NET アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

.NET [WebSphere eXtreme Scale Client for .NET 用のデータ・グリッド・セキュリティーの構成](#)

Secure Sockets Layer (SSL) を介して通信し、ユーザー/パスワード認証ロジックを使用するように .NET および Java を構成できます。

.NET [WebSphere eXtreme Scale Client for .NET 用の TLS の構成](#)

WebSphere eXtreme Scale Client for .NET 用に Transport Layer Security (TLS) を構成することができます。

.NET [WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

WebSphere eXtreme Scale Client for .NET からサーバー・サイドに資格情報を送信するには、ICredentialGenerator および ICredential インターフェースを実装する必要があります。これらのインターフェースでは、データ・グリッドに渡されてサーバー・サイドで解釈される資格情報オブジェクトが生成されます。サーバー・サイドで、対応するプラグインが資格情報オブジェクトを解釈します。

.NET [WebSphere eXtreme Scale Client for .NET のためのカスタム資格情報のプログラミング](#)

マップのユーザー資格情報を指定することができます。マップのユーザー資格情報があれば、2 人のユーザーが Web アプリケーションを介して同じデータ・グリッドと対話するようにすることができます。

親トピック: [単純データ・グリッドにアクセスするアプリケーションの開発](#)

.NET 開発環境の設定

Microsoft Visual Studio で WebSphere® eXtreme Scale クライアント for .NET を使用するには、開発環境をインストールし、WebSphere eXtreme Scale クライアント for .NET アセンブリーを使用するようにプロジェクトを構成する必要があります。

始める前に

- サポートされる Microsoft Visual Studio リリースのリストについては、[Microsoft .NET に関する考慮事項](#)を参照してください。
- WebSphere eXtreme Scale クライアント for .NET をインストールします。インストール・ウィザードで、「カスタム」パスを選択し、開発環境を選択します。詳しくは、[WebSphere eXtreme Scale Client for .NET のインストール](#)を参照してください。

手順

1. Microsoft Visual Studio 環境で、プロジェクトを開きます。
2. WebSphere eXtreme Scale クライアント for .NET アセンブリーへの参照を追加します。アセンブリーは、[net_client_home](#)¥bin ディレクトリーにあります。IBM.WebSphere.Caching.dll ファイルを選択します。
3. WebSphere eXtreme Scale クライアント for .NET API を使用するために、以下の行をアプリケーションに追加します。

```
using IBM.WebSphere.Caching;  
using IBM.WebSphere.Caching.Map;
```

タスクの結果

アセンブリーを開発環境に統合すると、WebSphere eXtreme Scale クライアント for .NET API 用に IntelliSense が使用可能になります。

次のタスク

クライアント・アプリケーションで WebSphere eXtreme Scale クライアント for .NET API を使用します。API 資料へのアクセスについて詳しくは、[WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)を参照してください。

[WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)

WebSphere eXtreme Scale クライアント for .NET API 資料へのアクセスは、.chm ファイル内でも、インフォメーション・センターの API 資料を表示しても行うことができます。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

WebSphere eXtreme Scale Client for .NET API 資料へのアクセス

WebSphere® eXtreme Scale クライアント for .NET API 資料へのアクセスは、.chm ファイル内でも、インフォメーション・センターの API 資料を表示しても行うことができます。

手順

次のいずれかのオプションを使用して WebSphere eXtreme Scale クライアント for .NET API 資料を開きます。

- 製品と一緒にインストールされる .NET Client API 資料を使用します。 .NET クライアント API 資料をローカルで開くには、 [net_client_home¥doc¥IBM.WebSphere.Caching.chm](#) ファイルを開きます。
- インフォメーション・センターの API 資料を閲覧します。 詳しくは、 [Client for .NET API 資料](#) を参照してください。

親トピック: [.NET .NET 開発環境の設定](#)

.NET API を使用した動的マップの作成

2.5+ データ・グリッドがインスタンス化されたら、.NET API を使用して動的マップを作成できます。マップ・テンプレート
の事前定義セットに基づいて、動的にマップをインスタンス化することができます。

始める前に

動的マップで、どの構成オプションを使用する必要があるかを決定します。詳しくは、[動的マップの構成オプション](#)を参照してください。

手順

GetGridMapPessimisticTx メソッドを呼び出します。

```
IGridManager gm = GridManagerFactory.GetGridManager( );
ICatalogDomainInfo cdi =
    gm.CatalogDomainManager.CreateCatalogDomainInfo( catalogServerHostsList );
IClientConnectionContext ccc = gm.Connect( cdi, "SimpleClient.properties" );
grid = gm.GetGrid( ccc, "Grid" );
IGridMapPessimisticTx<Object, Object> map =
    grid.GetGridMapPessimisticTx<Object, Object>( "SessionState.LAT.P" );
```

SessionState.LAT.P マップは、最終アクセス時刻除去、ペシミスティック・ロック、および、ニア・キャッシュ無効化不可を使用するマップです。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連資料:

[動的マップの構成オプション](#)

.NET

.NET アプリケーションにおけるトランザクションのためのプログラミング

トランザクションを必要とする .NET アプリケーションを作成するときには、ロック処理、衝突処理、およびトランザクション分離などの問題を考慮する必要があります。

.NET

[.NET アプリケーションのためのトランザクション内のデータとの対話](#)

WebSphere® eXtreme Scale クライアントの API では、各スレッドが別々の IGridMapPessimisticTx または IGridMapPessimisticAutoTx オブジェクトを持つ必要があります。IGridMapPessimisticTx オブジェクトの場合は、トランザクションの開始、コミット、またはロールバックを明示的に行うために Transaction プロパティが使用されます。IGridMapPessimisticAutoTx オブジェクトでは、トランザクションの開始、コミット、およびロールバック操作が自動的に行われます。セッションを使用してデータと対話します (Add、Put、および Replace 操作など)。

.NET

[.NET アプリケーションにおけるロックの構成と実装](#)

WebSphere eXtreme Scale Client for .NET からアクセスするバッキング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。マップ・インスタンスのロック・タイムアウト値をオーバーライドすることもできます。ロックを構成したら、マップ内の個々のキーまたはキーのリストをロックすることができます。

.NET

[ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連概念:

[トランザクション](#)

.NET アプリケーションのためのトランザクション内のデータとの対話

WebSphere® eXtreme Scale クライアントの API では、各スレッドが別々の IGridMapPessimisticTx または IGridMapPessimisticAutoTx オブジェクトを持つ必要があります。IGridMapPessimisticTx オブジェクトの場合は、トランザクションの開始、コミット、またはロールバックを明示的に行うために Transaction プロパティが使用されます。IGridMapPessimisticAutoTx オブジェクトでは、トランザクションの開始、コミット、およびロールバック操作が自動的に行われます。セッションを使用してデータと対話します (Add、Put、および Replace 操作など)。

このタスクについて

IGridMapPessimisticTx および IGridMapPessimisticAutoTx インターフェースは、データを処理する Add、Get、Put、Replace、および Remove などの操作を提供します。IGridMapPessimisticTx インターフェースは、データへの同時アクセスを制御する Lock および GetAndLock などの追加操作を提供します。

手順

- データを追加します。

次のコード・フラグメントは、IGridMapPessimisticTx インターフェースを使用して、新しいトランザクションを開始し、データ・グリッドの項目を作成し、そしてトランザクション全体をコミットする方法を示しています。

```
IGridMapPessimisticTx<String,Person> ptmap;  
ptmap = grid.GetGridMapPessimisticTx<String,Person>("PERSON");  
ptmap.Transaction.Begin();  
Person p = new Person();  
p.name = "John Doe";  
ptmap.Add(p.name, p);  
ptmap.Transaction.Commit();
```

- データを置き換えます。

次のコード・フラグメントは、IGridMapPessimisticTx インターフェースを使用して、新しいトランザクションを開始し、データ・グリッド内の項目をロックしてその値を取得し、項目値を置き換え、そしてトランザクションをコミットする方法を示しています。

```
IGridMapPessimisticTx<String,Person> ptmap;  
ptmap = grid.GetGridMapPessimisticTx<String,Person>("PERSON");  
ptmap.Transaction.Begin();  
Person p = ptmap.GetAndLock("John Doe", LockMode.Upgradable);  
p.age = 30;  
ptmap.Replace(p.name, p);  
ptmap.Transaction.Commit();
```

アプリケーションでは、通常は、単純な get ではなく、GetAndLock メソッドを使用してレコードをロックします。メソッドは、更新済みの値をマップに提供するために呼び出す必要があります。Replace メソッドが呼び出されないと、マップは変更されません。

親トピック: [.NET アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[データ・アクセスおよびトランザクション](#)

.NET アプリケーションにおけるロックの構成と実装

WebSphere® eXtreme Scale Client for .NET からアクセスするバッキング・マップの場合は、ペシミスティック・ロック・ストラテジーを定義する必要があります。マップ・インスタンスのロック・タイムアウト値をオーバーライドすることもできます。ロックを構成したら、マップ内の個々のキーまたはキーのリストをロックすることができます。

始める前に

- 使用したいロック・ストラテジーを決めます。詳しくは、[ロック・ストラテジー](#)を参照してください。
- 動的マップでペシミスティック・ロック・ストラテジーを構成します。詳しくは、[ロック・ストラテジーの構成](#)を参照してください。

手順

1. バッキング・マップでペシミスティック・ロック・ストラテジーを構成します。WebSphere eXtreme Scale Client for .NET はペシミスティック・ロック・ストラテジーのみをサポートします。詳しくは、[ロック・ストラテジーの構成](#)を参照してください。
2. 単一 IGridMapPessimisticTx インスタンスのロック待ちタイムアウトをオーバーライドします。**IGridMapPessimisticTx.LockTimeout** プロパティを使用して、特定の IGridMapPessimisticTx インスタンスのロック・タイムアウト値をオーバーライドします。ロック・タイムアウト値は、新規のタイムアウト値の設定後に開始されたすべてのトランザクションに影響します。このメソッドは、ロック競合が選択トランザクションで起こりうる、あるいは予想される場合に便利です。
3. マップ内の個々のキーまたはキーのリストをロックします。Lock メソッドを使用して、データ・グリッド内のキーをロックするか、またはキーをロックして値がデータ・グリッド内に存在するかを確認します。
 - 次のメソッドはマップ内のキーをロックします。キーが存在する場合は true を返し、キーが存在しない場合は false を返します。

```
bool IGridMapPessimisticTx.Lock(Tkey key, LockMode lockMode);
```

- 次のメソッドはマップ内のキーのリストをロックし、true または false 値のリストを返します。キーが存在する場合は true を返し、キーが存在しない場合は false を返します。

```
IList<bool> IGridMapPessimisticTx.LockAll(IList<TKey> keyList, LockMode lockMode);
```

LockMode は列挙型で以下に示す値を取ることができ、ここでロックしたいキーを指定できます。

- Shared、Upgradable、Exclusive

LockMode パラメーターの設定例を次に示します。

```
ptmap.Transaction.Begin();  
ptmap.Lock(key, LockMode.Upgradable);  
ptmap.Put(key, value);  
ptmap.Transaction.Commit();
```

親トピック: [.NET .NET アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[ロック・タイプ](#)
[ロック・ストラテジー](#)
[デッドロック](#)

関連タスク:

[.NET ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#)
[ロック・ストラテジーの構成](#)

ロック・シナリオでの例外処理の実装 (.NET アプリケーション)

LockTimeoutException 例外または LockDeadlockException 例外が発生したときにロックが過度に長い時間保持されないようにするには、予期しないイベントが発生したとき、アプリケーションが予期しない例外をキャッチし、rollback メソッドを呼び出す必要があります。

手順

1. 例外をキャッチし、結果のメッセージを表示します。

```
try {  
    ...  
} catch (GridException ge) {  
    System.Console.WriteLine(ge.ToString());  
}
```

LockDeadlockException 例外がスローされると、この例外は内部例外として別の例外の中に含まれることがあります。前のコード・スニペットは、内部例外チェーン全体 (存在する場合) を含む最上位例外を表示します。LockDeadlockException 例外に固有の例外メッセージに、ロック競合の詳細が含まれています。このメッセージの解釈の仕方については、[デッドロックのトラブルシューティング](#)を参照してください。

```
IBM.WebSphere.Caching.Map.LockDeadlockException: Message
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるメッセージを表示します。

2. 例外の後、トランザクションをロールバックします。

```
IGridMapPessimisticTx<String,Person> ptmap;  
ptmap = grid.GetGridMapPessimisticTx<String,Person>("PERSON");  
try {  
    ptmap.Transaction.Begin();  
    Person p = ptmap.Get("Lynn");  
    // Lynn had a birthday, so we make her 1 year older.  
    p.Age++;  
    ptmap.Put(p.name, p);  
    ptmap.Transaction.Commit();  
}  
catch (GridException ge) {  
    System.Console.WriteLine(ge.ToString());  
}  
finally {  
    if ( ptmap.Transaction.Active )  
        ptmap.Transaction.Rollback();  
}
```

コード・スニペットの finally ブロックは、予期しない例外が発生したときにトランザクションがロールバックされるようにしています。LockDeadlockException 例外のみでなく、発生する可能性のあるその他の予期しない例外もすべて処理します。finally ブロックは、commit メソッドの呼び出し時に例外が発生するケースも処理します。この例は、予期しない例外を処理する唯一の方法ではありません。アプリケーションが、発生する予期しない例外のいくつかをキャッチし、そのアプリケーション例外の 1 つを表示するケースも存在するかもしれません。適宜 catch ブロックを追加できますが、アプリケーションは、コード・スニペットがトランザクションを完了せずに終了しないようにする必要があります。

親トピック: [.NET .NET アプリケーションにおけるトランザクションのためのプログラミング](#)

関連概念:

[ロック・タイプ](#)

[ロック・ストラテジー](#)

[デッドロック](#)

関連タスク:

[.NET .NET アプリケーションにおけるロックの構成と実装](#)

[ロック・ストラテジーの構成](#)

WebSphere eXtreme Scale Client for .NET 用のデータ・グリッド・セキュリティの構成

Secure Sockets Layer (SSL) を介して通信し、ユーザー/パスワード認証ロジックを使用するように .NET および Java™ を構成できます。

始める前に

ご使用の環境用の key.jks ファイルおよび trust.jks ファイルを用意しておく必要があります。

手順

サーバーでセキュリティを使用可能にして構成します。セキュリティがサーバーでまだ構成されていない場合は、以下の手順を使用して、外部オーセンティケーター・サンプルでセキュリティを構成します。

- サンプル・セキュリティ・ファイルを取得します。 security_extauth.zip ファイルのサンプル・ファイルを、[WebSphere eXtreme Scale wiki](#) からダウンロードします。
 - xsjaas3.config: Java Authentication and Authorization Service (JAAS) 構成を定義します。
 - sampleKS3.jks: JAAS ユーザーおよびパスワードの値の鍵ストアが入っています。
 - security3.xml: セキュリティに使用するオーセンティケーターを定義します。
- xsjaas3.config ファイルを編集し、sampleKS3.jks ファイルのパスを修正します。
- サンプル sampleKS3.jks ファイルではなく、独自の秘密鍵ストアを生成する場合は、**keytool** ユーティリティを使用して秘密鍵を生成します。

```
keytool -genkey -alias myalias -keysize 2048 -keystore key.jks -keyalg rsa -dname "CN=www.mydomain.com" -storepass password -keypass password -validity 3650
```

- sampleServer.properties を編集して、セキュリティを使用可能にします。 sampleServer.properties ファイルは、[wxs_install_root](#)¥properties ディレクトリーにあります。以下のプロパティ値のコメントを外して編集します。

```
securityEnabled=true
secureTokenManagerType=none
alias=ogsample
contextProvider=IBMJSSE2
protocol=SSL
keyStoreType=JKS
keyStore=../../../../../xio.test/etc/test/security/key.jks
keyStorePassword=ogpass
trustStoreType=JKS
trustStore=../../../../../xio.test/etc/test/security/trust.jks
trustStorePassword=ogpass
```

次のタスク

WebSphere® eXtreme Scale Client for .NET 用の Transport Layer Security (TLS) を構成します。詳しくは、[WebSphere eXtreme Scale Client for .NET 用の TLS の構成](#)を参照してください。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

WebSphere eXtreme Scale Client for .NET 用の TLS の構成

WebSphere® eXtreme Scale Client for .NET 用に Transport Layer Security (TLS) を構成することができます。

始める前に

- WebSphere eXtreme Scale Client for .NET 構成に追加する、鍵ストアと、関連付けられたパスワードが必要です。

手順

1. オプション: keytool ユーティリティを使用して、key.jks ファイルから公開証明書を抽出します。

```
keytool -export -alias myalias -keystore key.jks -file public.cer -storepass password
```

証明書管理ツール certmgr.msc を使用して、この公開鍵を Windows の証明書ストアにインポートして、鍵を「信頼されたルート証明機関」または「信頼されたユーザー」証明書フォルダーにインポートします。(client.properties ファイルの **keyStore** プロパティは、このファイルを指すことができます)

2. Client.Net.properties ファイルを編集して、以下のプロパティの値を含めます。

```
securityEnabled=true
credentialAuthentication=supported
authenticationRetryCount=3
credentialGeneratorAssembly=IBM.WebSphere.Caching.CredentialGenerator,Version=8.6.0.0,
Culture=neutral,PublicKeyToken=b439a24ee43b0816
credentialGeneratorProps=manager manager1
transportType=ssl-required
publicKeyFile=<name>.cer
```

credentialGeneratorProps プロパティの値 manager manager1 は、Credential オブジェクトでサーバーに提供されるユーザー名とパスワードの値として使用されます。

publicKeyFile プロパティは、.NET ランタイムの相対パスとして設定されます。**publicKeyFile** プロパティが設定されていない場合は、Windows 証明書ストアで public.cer ファイルが検索されます。

publicKeyFile プロパティが設定されている場合は、指定されたファイルが SSL 公開証明書ファイルとして使用されます。指定されたファイルが見つからない場合は、.NET クライアントは、一致する public.cer ファイルを証明書ストアで見つけようとしています。

3. オプション: credentialGeneratorProps プロパティの値をエンコードします。プロパティ値をエンコードするには、Client.Net.properties ファイルを、Java ベースの WebSphere eXtreme Scale クライアントインストール済み環境を持つコンピューターに転送します。**FilePasswordEncoder** ユーティリティを実行して credentialGeneratorProps プロパティをエンコードします。

```
FilePasswordEncoder.bat Client.Net.Properties credentialGeneratorProps
```

このユーティリティをプロパティ・ファイルに対して実行すると、ファイル内のすべてのコメントが削除されます。

4. `net_client_home`¥IBM.WebSphere.Caching.CredentialGenerator.dll を `net_client_home`¥sample¥SimpleClient¥bin¥<ConfigurationName> ディレクトリーにコピーします。
5. `ConfigurationName` プロジェクト・コンテキストを使用してサンプルをビルドします。サンプルをサーバーに対して実行します。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング

WebSphere® eXtreme Scale Client for .NET からサーバー・サイドに資格情報を送信するには、ICredentialGenerator および ICredential インターフェースを実装する必要があります。これらのインターフェースでは、データ・グリッドに渡されてサーバー・サイドで解釈される資格情報オブジェクトが生成されます。サーバー・サイドで、対応するプラグインが資格情報オブジェクトを解釈します。

このタスクについて

認証を実行するには、.NET アプリケーションが以下のインターフェースを実装している必要があります。

- ICredential: Credential は、クライアント資格情報 (ユーザー ID とパスワードのペアなど) を表します。
- ICredentialGenerator: CredentialGenerator は、資格情報を生成するための資格情報ファクトリーを表します。

.NET クライアント・アプリケーションが認証を必要とするサーバーに接続すると、クライアントはクライアント資格情報を提供するように要求されます。クライアント資格情報は、ICredential インターフェースによって表されます。クライアント資格情報には、ユーザー名とパスワードのペア、Kerberos チケット、クライアント証明書、またはクライアントとサーバーが同意する任意の形式でのデータがあります。このインターフェースでは、equals(Object) メソッドおよび hashCode メソッドを定義します。Credential オブジェクトをサーバー・サイドの鍵として使用することによって認証済み Subject オブジェクトがキャッシュされるため、この 2 つのメソッドは重要です。ICredentialGenerator インターフェースを使用して資格情報を生成することもできます。このインターフェースは、資格情報に有効期限がある場合に役立ちます。Credential プロパティを取得することにより、新規資格情報が生成されます。

また、提供されている CredentialGenerator プラグインを使用して、Client.Net.Properties ファイルの **credentialGeneratorProps=** 設定に基づいた資格情報を作成することもできます。資格情報プラグインを定義する追加設定は、**credentialGeneratorAssembly** および **credentialGeneratorClass** です。

手順

.NET アプリケーションで、ICredentialGenerator および ICredential インターフェースを実装します。

要確認: このクライアントの資格情報プラグインを実装する場合は、WebSphere eXtreme Scale Client for .NET からの認証資格情報を解釈して受け取ることができる、対応サーバーの資格情報プラグインも実装する必要があります。

以下の例を使用して、アプリケーションを開発できます。

- [例: .NET アプリケーション用のユーザー・パスワード資格情報の実装](#)
- [例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装](#)

.NET

[例: .NET アプリケーション用のユーザー・パスワード資格情報の実装](#)

この例を使用して、ICredential インターフェースの実装を独自に作成することができます。ユーザー・パスワードの資格情報では、ユーザー ID とパスワードを保管します。

.NET

[例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装](#)

この例を使用して、ICredentialGenerator インターフェースの実装を独自に作成することができます。このインターフェースはユーザー ID およびパスワードを受け取ります。UserPasswordCredential オブジェクトには、読み取り専用の資格情報プロパティから取得される、ユーザー ID およびパスワードが含まれています。

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

関連資料:

.NET

[例: .NET アプリケーション用のユーザー・パスワード資格情報の実装](#)

.NET

[例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装](#)[クライアント・プロパティ・ファイル](#)

関連情報:

[ICredential インターフェース](#)[ICredentialGenerator インターフェース](#)

例: .NET アプリケーション用のユーザー・パスワード資格情報の実装

この例を使用して、ICredential インターフェースの実装を独自に作成することができます。ユーザー・パスワードの資格情報では、ユーザー ID とパスワードを保管します。

UserPasswordCredential.cs

```
// Module : UserPasswordCredential.cs

using System;
using IBM.WebSphere.Caching.Security;

namespace com.ibm.websphere.objectgrid.security.plugins.builtins
{
    public class UserPasswordCredential : ICredential
    {
        private String ivUserName;

        private String ivPassword;

        /// <summary>
        ///Creates a UserPasswordCredential with the specified user name and
        /// password.
        ///
        /// ArgumentException if userName or password is null
        /// </summary>
        /// <param name="userName">the user name for this credential</param>
        /// <param name="password">the password for this credential</param>
        public UserPasswordCredential(String userName, String password)
        {
            if (userName == null || password == null) {
                throw new ArgumentException("User name and password cannot be null.");
            }
            this.ivUserName = userName;
            this.ivPassword = password;
        }

        /// <summary>Gets the user name for this credential.</summary>
        /// <returns>the user name argument that was passed to the constructor
        ///or the setUsername(String) method of this class </returns>
        public String GetUserName() {
            return ivUserName;
        }

        /// <summary>Sets the user name for this credential.
        ///ArgumentException if userName is null
        /// </summary>
        /// <param name="userName">userName the user name to set.</param>
        public void SetUserName(String userName) {
            if (userName == null) {
                throw new ArgumentException("User name cannot be null.");
            }
            this.ivUserName = userName;
        }

        /// <summary>Gets the password for this credential.
        /// </summary>
        /// <returns>the password argument that was passed to the constructor or the
        setPassword(String) method of this class</returns>
        public String GetPassword() {
            return ivPassword;
        }

        /// <summary>Sets the password for this credential.
        ///ArgumentException if password is null
        /// </summary>
        /// <param name="password">the password to set.</param>
```

```

public void SetPassword(String password) {
    if (password == null)
    {
        throw new ArgumentException("Password cannot be null.");
    }
    this.ivPassword = password;
}

/// <summary>Checks two UserPasswordCredential objects for equality.
///<p>
/// Two UserPasswordCredential objects are equal if and only if their user names
/// and passwords are equal.
/// </summary>
/// <param name="o">the object we are testing for equality with this object.
</param>
/// <returns>>true if both UserPasswordCredential objects are equivalent.</returns>
public bool Equals(ICredential credential)
{
    if (this == credential) {
        return true;
    }
    if (credential is UserPasswordCredential) {
        UserPasswordCredential other = (UserPasswordCredential)credential;
        return other.ivPassword.Equals(ivPassword) &&
other.ivUserName.Equals(ivUserName);
    }
    return false;
}

/// <summary>Returns the hashCode of the UserPasswordCredential object.
/// </summary>
/// <returns>return the hash code of this object</returns>
public override int GetHashCode() {
    int ret = ivUserName.GetHashCode() + ivPassword.GetHashCode();
    return ret;
}

/// <summary>this.Object as a string
/// </summary>
/// <returns>return the string presentation of the UserPasswordCredential object.
</returns>
public override String ToString() {
    return typeof(UserPasswordCredential).FullName + "[" + ivUserName +
",xxxxxx]";
}

}
}

```

親トピック: [.NET WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連タスク:

[.NET WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連情報:

[ICredential インターフェース](#)

[ICredentialGenerator インターフェース](#)

例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装

この例を使用して、ICredentialGenerator インターフェースの実装を独自に作成することができます。このインターフェースはユーザー ID およびパスワードを受け取ります。UserPasswordCredential オブジェクトには、読み取り専用の資格情報プロパティから取得される、ユーザー ID およびパスワードが含まれています。

UserPasswordCredentialGenerator.cs

```
// Module : UserPasswordCredentialGenerator.cs
//
// Source File Description: Reference Documentation
//
using System;
using System.Security.Authentication;
using IBM.WebSphere.Caching.Security;
using com.ibm.websphere.objectgrid.security.plugins.builtins;

namespace IBM.WebSphere.Caching.Security
{
    public class UserPasswordCredentialGenerator : ICredentialGenerator
    {
        private String ivUser;

        private String ivPwd;

        public ICredential Credential { get { return _getCredential(); } }

        public string Properties { set { _setProperties(value);} }

        public UserPasswordCredentialGenerator()
        {
            ivUser = null;
            ivPwd = null;
        }

        public UserPasswordCredentialGenerator(String user=null, String pwd=null)
        {
            ivUser = user;
            ivPwd = pwd;
        }

        /// <summary>Creates a new UserPasswordCredential object using this object's user
        name and password.
        /// </summary>
        /// <returns>new UserPasswordCredential instance</returns>
        private ICredential _getCredential()
        {
            try
            {
                ICredential MyCredential = new UserPasswordCredential(ivUser, ivPwd) as
                ICredential;
                return (ICredential) MyCredential;
            }
            catch (Exception e)
            {
                AuthenticationException CannotGenerateCredentialException = new
                AuthenticationException(e.ToString());
                throw CannotGenerateCredentialException;
            }
        }

        /// <summary>Gets the password for this credential generator.
        /// </summary>
        /// <returns>the password argument that was passed to the constructor</returns>
        public String getPassword()
    }
}
```



```

    {
        return ivPwd;
    }

    /// <summary>Gets the user name for this credential.
    /// </summary>
    /// <returns>the user argument that was passed to the constructor of this
class</returns>
    public String getUsername()
    {
        return ivUser;
    }

    /// <summary>Sets additional properties namely a user name and password.
    ///Throws ArgumentException if the format is not valid
    /// </summary>
    /// <param name="properties">properties a properties string with a user name and
a password separated by a blank.</param>
    private void _setProperty(string properties)
    {
        String token = properties;
        char[] Separator = { ' ' };
        String[] StringProperty = properties.Split(Separator);
        if (StringProperty.Length != 2)
        {
            throw new ArgumentException(
                "The properties should have a user name and password and separated by
a space.");
        }

        ivUser = StringProperty[0];
        ivPwd = StringProperty[1];
    }

    /// <summary>Checks two UserPasswordCredentialGenerator objects for equality.
    ///<p>
    ///Two UserPasswordCredentialGenerator objects are equal if and only if
    ///their user names and passwords are equal.
    /// </summary>
    /// <param name="obj">the object we are testing for equality with this object.
</param>
    /// <returns><code>>true</code> if both UserPasswordCredentialGenerator objects are
equivalent</returns>
    public override bool Equals(Object obj)
    {
        if (obj == this)
        {
            return true;
        }

        if (obj != null && obj is UserPasswordCredentialGenerator)
        {
            UserPasswordCredentialGenerator other =
(UserPasswordCredentialGenerator)obj;

            Boolean bothUserNull = false;
            Boolean bothPwdNull = false;

            if (ivUser == null)
            {
                if (other.ivUser == null)
                {
                    bothUserNull = true;
                }
                else
                {
                    return false;
                }
            }
        }
    }

```



```
    }
    if (ivPwd == null)
    {
        if (other.ivPwd == null)
        {
            bothPwdNull = true;
        }
        else
        {
            return false;
        }
    }
    return (bothUserNull || ivUser.Equals(other.ivUser)) && (bothPwdNull ||
ivPwd.Equals(other.ivPwd));
}
return false;
}

/// <summary>Returns the hashcode of the UserPasswordCredentialGenerator object.
/// </summary>
/// <returns>the hash code of this object</returns>
public override int GetHashCode()
{
    return ivUser.GetHashCode() + ivPwd.GetHashCode();
}

}

}
```

親トピック: [.NET](#) [WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連タスク:

[.NET](#) [WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連情報:

[ICredential インターフェイス](#)

[ICredentialGenerator インターフェイス](#)

WebSphere® eXtreme Scale Client for .NET のためのカスタム資格情報のプログラミング

マップのユーザー資格情報を指定することができます。マップのユーザー資格情報があれば、2人のユーザーが Web アプリケーションを介して同じデータ・グリッドと対話することができます。

手順

1. client.properties ファイルでユーザー資格情報を設定します。

```
credentialAuthentication=required
authenticationRetryCount=3
credentialGeneratorAssembly=IBM.WebSphere.Caching.CredentialGenerator,
Version=8.6.0.0, Culture=neutral, PublicKeyToken=b439a24ee43b0816
credentialGeneratorClass=IBM.WebSphere.Caching.Security.UserPasswordCredentialGenerator
credentialGeneratorProps=manager manager1
```

2. アプリケーション・プロジェクト内の IBM.WebSphere.Caching.CredentialGenerator.dll ファイルへの参照を追加します。このプラグイン DLL には ICredentialGenerator 実装が含まれています。
3. .NET アプリケーションで ICredentialGenerator API を使用します。

```
//GridManagerFactory.GetGridManager
IGridManager gm = GridManagerFactory.GetGridManager();
//IGridManager.Connect
ICatalogDomainInfo cdi =
gm.CatalogDomainManager.CreateCatalogDomainInfo(hostAndPort);
ctx = gm.Connect(cdi, "client.properties");
//IGridManager.GetGrid
IGrid grid = gm.GetGrid( ctx, "Grid");
ICredentialGenerator credGenManager = new UserPasswordCredentialGenerator("manager",
"manager1");
ICredentialGenerator credGenOperator = new
UserPasswordCredentialGenerator("operator", "operator1");
//IGrid.GetGridMap
IGridMapPessimisticAutoTx<Object, Object> gridMap1 =
grid.GetGridMapPessimisticAutoTx<Object, Object>("Map1", credGenManager);

IGridMapPessimisticAutoTx<Object, Object> gridMap2 =
grid.GetGridMapPessimisticAutoTx<Object, Object>("Map1", credGenOperator);
```

親トピック: [.NET 2.5+ .NET API を使用したデータ・グリッド・アプリケーションの開発](#)

モニター

ご使用の WebSphere® DataPower® XC10 アプライアンス に関するさまざまな局面をモニターでき、「タスク」ビューによる構成変更の状況や、ユーザー・インターフェース内の「モニター」ビューで見る データ・グリッド のパフォーマンスが含まれます。

始める前に

ユーザー・インターフェース内の「タスク」ビューまたは「モニター」ビューでデータを見るには、ご使用のアプライアンスが構成されており、アプリケーションからの新規エントリーを受け取るように データ・グリッド がアプリケーションからの新規エントリーを受け入れるように作成しなければなりません。

2.5+ [アプライアンスの始動状況のモニター](#)

アプライアンスの始動状況をモニターすることができます。始動状況をモニターすることは、問題の判別とトラブルシューティングに役立ちます。

[ユーザー・インターフェースでのデータ・グリッドのモニター](#)

WebSphere DataPower XC10 アプライアンス のグラフ機能を使用して、ご使用の環境における データ・グリッド の全体的なパフォーマンスを表示することができます。

[タスクによるアクティビティのモニター](#)

例えば集合にアプライアンスを追加するなど、管理上の変更の進行をモニターするためにタスクが使用できます。

[xscmd ユーティリティーによるモニター](#)

xscmd ユーティリティーを使用すれば、ご使用のアプライアンス上で 実行されるデータ・グリッドに関するテキスト情報を表示できます。

[CSV ファイルによるモニター](#)

モニター・データは自動的に CSV ファイルに書き込まれます。これらの CSV ファイルには、サーバー、マップ、またはデータ・グリッドに関する情報を含めることができます。

[xsadmin ユーティリティーによるモニター](#)

xsadmin ユーティリティーを使用して、WebSphere DataPower XC10 アプライアンス上で 実行しているデータ・グリッドに関するテキスト情報をフォーマット設定し、表示できます。このサンプル・ユーティリティーは現在のデプロイメント・データの解析とディスクバリーの方法を提供するもので、カスタム・ユーティリティーの作成基盤として使用することができます。

2.5+ [環境のヘルスのモニター](#)

メッセージ・センターでは、ログおよび初期障害データ・キャプチャー機能 (FFDC) メッセージのイベント通知の集約ビューが示されます。これらのイベント通知は、Web コンソールのメッセージ・センター、**xscmd** ユーティリティー、ヘルス・ログ・ファイル、HTTP コマンド・インターフェース、または MBeans を使用したプログラマチック方式で表示することができます。

[Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

Simple Network Monitoring Protocol (SNMP) サポートにより、IBM® WebSphere DataPower XC10 アプライアンス の状況を、データ・センターのシステムのより大きなグループの一部として、モニターできます。SNMP モニターは、システム問題を素早く通知し、それらに迅速に応答する機能を強化します。

[リモート・ロギングの構成](#)

リモート・ロギングを使用可能にすれば、ログ・エントリーをアプライアンス外のリモート・サーバーに保存することができます。リモート・ロギングは、問題の分離や長期にわたる動作のモニターを容易にするために詳細デバッグ・ログ・レベルを設定する必要があるときに役立つことがあります。

アプライアンスの始動状況のモニター

2.5+ アプライアンスの始動状況をモニターすることができます。始動状況をモニターすることは、問題の判別とトラブルシューティングに役立ちます。

始める前に

- アプライアンスを初期化、始動、または再始動します。
- これにはxcadmin ユーザー名およびパスワードが必要です。

このタスクについて

始動ユーザー・インターフェース・パネルまたは **start-progress** コマンドを使用して、アプライアンスの始動状況をモニターすることができます。

アプライアンス始動ユーザー・インターフェース・パネル

このユーザー・インターフェース・パネルを使用して、始動状況 (完了パーセントや個々のサーバーおよびプロセスの状況など) を表示することができます。状況は 15 秒ごとに自動的にリフレッシュします。

start-progress コマンド

start-progress コマンドはコマンド行インターフェースで実行します。このコマンドは始動状況の現在のスナップショットを返します。最新の始動状況を取得するには、このコマンドをもう一度実行してください。

手順

- ユーザー・インターフェースでアプライアンス始動状況パネルを表示します。
 1. 次の URL を使用してアプライアンス始動状況パネルにアクセスします。

```
https://<appliance_hostname>:9443/
```

始動の進行状況がパーセントで表示されます。

2. 100 パーセントになる前に進行が止まった場合は、ログ・ファイルをダウンロードして問題のトラブルシューティングを行うことができます。「[ログ・ファイルのダウンロード](#)」をクリックします。
- コマンド行インターフェースでアプライアンス始動状況を表示します。
 1. コマンド行インターフェースにログインします。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。
 2. **start-progress** コマンドを実行します。アプライアンス始動シーケンスの状況が表示されます。
 3. 最新の状況を取得するには、**start-progress** コマンドをもう一度実行します。

例

次の例は成功した始動シーケンスです。

```
Volume 1 mounted
Volume 2 mounted
Grid administrative service started.
Catalog server started.
Grid configuration service started.
Container server 01 started.
Container server 02 started.
Container server 03 started.
Container server 04 started.
Container server 05 started.
Container server 06 started.
Container server 07 started.
Container server 08 started.
Administrative console started.
```

- 集合内の最初の 3 つのアプライアンスでカタログ・サーバーが実行されます。アプライアンスが 3 つを超えている場合は、カタログ・サーバー始動メッセージが表示されないことがあります。
- 始動されるコンテナ・サーバーの数は、使用しているトランスポート・タイプによって決まります。IBM eXtremeIO (XIO) の場合は、デフォルトで 8 つのコンテナ・サーバーが始動されます。オブジェクト・リクエスト・ブローカー (ORB) の場合は、デフォルトで 16 のコンテナ・サーバーが始動されます。トランスポート・タイプについて詳しくは、[IBM eXtremeIO \(XIO\) の構成](#)を参照してください。
- Administrative console started. 状況が表示されたら、ユーザー・インターフェースにログインすることができます。

次のタスク

- **request suspend** および **request force-suspend** コマンドを実行すると、アプライアンス始動パネルも延期状態を示します。

親トピック: [モニター](#)

ユーザー・インターフェースでのデータ・グリッドのモニター

WebSphere® DataPower® XC10 アプライアンス のグラフ機能を使用して、ご使用の環境における データ・グリッドの全体的なパフォーマンスを表示することができます。

始める前に

データ・グリッドを作成し、アプリケーションで データ・グリッドを使用するように構成したら、統計が使用可能になるまで少し時間を置きます。例えば、動的キャッシュ データ・グリッド (data grid)を使用する場合、動的キャッシュを実行する WebSphere Application Server がアプライアンス上の動的キャッシュ データ・グリッド (data grid) に接続するまで、統計は使用できません。集合を使用している場合は、統計が使用可能になる前に、集合の初期設定が完了していなければなりません。一般的には、統計における変化を見るために、主要な構成変更の後で 1 分ほど待ちます。

このタスクについて

グラフの性質

日、週、および月の一連のデータは、長期間保持されます。しかし、「最終時刻」の一連のデータは、メモリー内のみ保管されます。最終時刻データは、次の場所に保管されます。

- データ・グリッドのメモリー内
- コンソール・キャッシュのメモリー内

これらのメモリー内のどちらかの場所でのみ統計データの部分が逸失した場合、そのデータはまだ使用可能です。しかし、両方の場所でデータが逸失した場合は、グラフの中にデータは表示されません。

メモリー内のデータ損失の条件:

- データ・グリッドが移動された場合、統計が集計されていないときに期間が発生します。
- アプライアンスが再始動したとき、データ・グリッドとコンソール・キャッシュの両方のメモリー・データが消去されます。
- コンソール・キャッシュのメモリーがいっぱいになると、最長未使用時間データはコンソール・キャッシュから除去されます。コンソール・キャッシュは、2048 の統計項目を保持します。
- コンソールが 30 分間非アクティブの場合、コンソール・プロセスが再始動し、メモリーは消去されます。

統計の収集では、最終時刻の一連のデータの再集計を続行します。

ヒント: グラフ内の任意のデータ・ポイントに関するより具体的な情報を表示するには、そのデータ・ポイントの上にマウス・ポインターを移動させてください。

手順

- データ・グリッド全体のパフォーマンスを表示するには、「モニター」 > 「データ・グリッドの概要」をクリックします。このページには、以下の情報が含まれています。

「使用容量」タブ

現行データ・グリッドの使用容量の分布

このグラフには、アプライアンスまたは集合の使用可能容量の合計、基本データおよびレプリカ・データ・コピーに使用されている容量の合計、容量限度が設定されている場合の残りの制限容量、使用容量が最も大きいデータ・グリッドのビューが含まれています。次のオプションを使用してデータをソートできます。上位 25 のデータ・グリッドのみが表示されます。

- 最大使用容量コンシューマー (Largest used capacity consumers)
- 最大使用済み制限容量の割合 (Largest percent of limited capacity used)

一定時間の使用容量

このグラフには、選択した期間の使用容量が表示されます。

「平均スループット」タブ

トランザクション/秒で測定した平均スループットによるトップ 5 の最もアクティブな データ・グリッド

このグラフには、平均スループット (トランザクション/秒で測定) で編成された、トップ 5 のデータ・グリッドのリストが含まれています。

一定時間の平均スループット

このグラフには、選択した期間の平均スループット (トランザクション/秒で測定) が表示されます。

「トランザクション平均時間」タブ

平均トランザクション時間 (ミリ秒) が下位 5 位までの低速なデータ・グリッド

このグラフには、平均トランザクション時間で編成された、下位 5 の最も低速なデータ・グリッドのリストが含まれています。

一定時間の平均トランザクション

このグラフには、選択した期間の平均トランザクションが表示されます。

- 個々のデータ・グリッドを見るには、「モニター」 > 「個別データ・グリッドの概要」 > 「`data_grid_name`」をクリックします。このページには、最後の 30 秒間のキャッシュ・エントリー数、平均トランザクション時間、平均スループット、キャッシュのヒット率、および制限容量の割合を含むサマリーが表示されます。以下のグラフも表示できます。

使用容量

このグラフには、キャッシュの使用容量と実際のエントリー数の比較、および構成されているキャッシュの容量限度が示されます。使用容量には基本データとレプリカ・データの両方が含まれます。表示される時刻範囲を、次のように編集できます。最終時刻、最終日付、先週、先月。グラフの表示内容の詳細レベルは、選択した時刻範囲によって変化します。

キャッシュ・エントリー数の解釈:

- **すべてのデータ・グリッドの場合:** キャッシュ・エントリー数はプライマリー断片のみを表します。プライマリー断片とレプリカ断片の両方を組み合わせた断片のキャッシュ・エントリー数を表示するには、`xscmd -c showMapSizes` コマンドを使用します。
- **動的キャッシュ・データ・グリッドの場合:** 動的キャッシュ・データ・グリッドごとに、デフォルトで 166 個のキャッシュ・エントリーが作成されます。各動的キャッシュ・グリッドには 83 のパーティションがあり、各パーティションまたは断片は、動的キャッシュ・グリッドの 2 個のエントリーで初期化されます。したがって、初期化時のキャッシュ・エントリーの数は、166 になります。これらのキャッシュ・エントリーには、WebSphere Application Server の動的キャッシュ・プロバイダー統計および動的キャッシュ構成が含まれます。そのため、データを動的キャッシュ・データ・グリッドに追加する前に、モニター・パネルに 166 個のキャッシュ・エントリーが表示されます。
- **セッション・データ・グリッドの場合:** データ・グリッド内のエントリー数には、セッション数、すべてのセッションの属性数、および追い出し表エントリーのカウントが含まれます。Web コンテナによって無効化されていないタイムアウト・セッションが存在する場合に、追い出し表にデータが追加されます。追い出し表には、アプリケーション・サーバーのフェイルオーバー時にのみデータが追加されます。

キャッシュの使用

このグラフは、キャッシュに対する成功した照会の数の視覚化に便利です。キャッシュの試行数、キャッシュのヒット数、およびキャッシュのヒット率をグラフに表示できます。

平均スループット

このグラフは、指定された時刻範囲で処理された、1 秒当たりの平均トランザクション数と、各トランザクションの平均時間長を示します。

- 特定のデータ・グリッドに関するより詳細な情報を表示するには、「モニター」 > 「データ・グリッドの詳細レポート」をクリックします。ツリーにはご使用の構成におけるすべてのデータ・グリッドが表示されています。特定のデータ・グリッドをドリルダウンして、そのデータ・グリッドの一部であるマップを表示します。詳細情報は、データ・グリッド名またはマップをクリックします。

データ・グリッドの詳細

使用容量と、データ・グリッドが属するゾーンのリストを表示できます。使用容量には基本データとレプリカ・データの両方が含まれます。データ・グリッド表示内の上位 25 のマップによって消費された容量を示すグラフです。合計プールも表示でき、これにはゾーンによる容量が含まれます。グラフは、上位 25 ゾーン全体で消費されるデータ・グリッド容量を示します。

マップの詳細

各グリッド内のマップに関する詳細情報を表示でき、マップ内のプライマリー・キャッシュ・エントリーの合計数、平均スループット、平均トランザクション時間、および上位 25 区画ごとに分割されたマップの合計容量などの詳細が含まれます。

親トピック: [モニター](#)

関連情報:

[入門チュートリアル・レッスン 4: 環境のモニター](#)

タスクによるアクティビティのモニター

例えば集合にアプライアンスを追加するなど、管理上の変更の進行をモニターするためにタスクが使用できます。

このタスクについて

初めから終わりまでタスクの状態を表示できます。タスクには、待機、成功、または失敗という全般的な状況があります。

表 1. 全般的なタスク状況のアイコン

アイコン	説明
待機 (🕒)	タスクはタスク・キューに入りましたが、まだ実行されていません。
実行中 (🕒)	タスクは実行中です。
成功 (✅)	タスクはエラーなしで完了しました。
失敗 (❌)	タスクは完了しませんでした。発生したエラーについては、タスク内のメッセージを参照してください。

各タスクは、タスク状況の詳細を示す 1 つ以上のメッセージから成ります。

表 2. タスク・メッセージのアイコン

アイコン	説明
情報 (ℹ️)	メッセージには、実行中のタスクのステップに関する情報が含まれます。
エラー (❌)	タスクは完了しませんでした。発生したエラーについては、タスク内のメッセージを参照してください。
2.5+ 警告 (⚠️)	メッセージには、タスクが失敗する可能性のあることを知らせる情報が含まれています。ただし、タスクは成功する可能性もあります。

手順

1. ユーザー・インターフェースで、「タスク」をクリックします。
2. 詳しくは、タスク名をクリックすると特定のタスクを表示できます。各タスクには、タスクのタイプ、タスクの開始および完了日時、さらにタスクの進行に関するより詳しい情報を表示するための特定の状況メッセージのリストが含まれます。タスクが合格または失敗の状況で完了するまで、ページが 10 秒ごとにリフレッシュして、タスクの状況を更新します。
3. **2.5+** オプション: 完了したタスクを削除します。完了したタスクの状況は成功または失敗です。「完了したタスクをすべて削除」(🗑️) をクリックします。このアイコンをクリックする場合は、すべてのタスクが削除されてもかまわないことを確認してください。「OK」をクリックすると、完了したタスクがすべて削除されます。状況が待機または実行中のタスクは削除されません。

親トピック: [モニター](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

xscmd ユーティリティーによるモニター

xscmd ユーティリティーを使用すれば、ご使用のアプライアンス上で実行されるデータ・グリッドに関するテキスト情報を表示できます。

始める前に

- xscmd ユーティリティーの開始について詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

このタスクについて

xscmd ユーティリティーを使用して、マップの内容など、データ・グリッドの現在のレイアウトおよび特定の状態を表示できます。この例では、このタスクのデータ・グリッドのレイアウトは、MapSetA マップ・セットに属する 1 つの MapA マップを持つ 1 つの ObjectGridA データ・グリッドから成ります。この例では、データ・グリッドにすべてのアクティブ・コンテナを表示する方法と、MapA マップのマップ・サイズに関するフィルタリングされたメトリックを印刷する方法を説明します。使用できるコマンド・オプションをすべて知りたい場合は、引数なしか、または **-help** オプションを付けて xscmd ユーティリティーを実行してください。

手順

1. xscmd ユーティリティーを使用して、環境をモニターします。

- すべてのサーバーの統計を使用可能にするには、以下のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c setStatsSpec -spec ALL=enabled -g ObjectGridA`
 - **Windows** `xscmd.bat -c setStatsSpec -spec ALL=enabled -g ObjectGridA`
- データ・グリッドのすべてのオンライン・コンテナ・サーバーを表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showPlacement -g ObjectGridA -ms MapSetA`
 - **Windows** `xscmd.bat -c showPlacement -g ObjectGridA -ms MapSetA`

すべてのコンテナ情報が表示されます。

重要: Transport Layer Security/Secure Sockets Layer (TLS/SSL) が使用可能であるときにこの情報を取得するには、JMX サービス・ポートを設定してカタログ・サーバーおよびコンテナ・サーバーを始動する必要があります。JMX サービス・ポートを設定するには、**startOgServer** または **startXsServer** スクリプトで **-JMXServicePort** オプションを使用するか、ServerProperties インターフェースで `setJMXServicePort` メソッドを呼び出すことができます。

- ObjectGridA データ・グリッドのマップについての情報を表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showMapSizes -g ObjectGridA -ms MapSetA`
 - **Windows** `xscmd.bat -c showMapSizes -g ObjectGridA -ms MapSetA`
- カタログ・サービスに接続して、カタログ・サービス・ドメイン全体の MapA マップに関する情報を表示するには、次のコマンドを実行します。
 - **UNIX** `./xscmd.sh -c showMapSizes -g ObjectGridA -ms MapSetA -m MapA -cep CatalogMachine:6645`
 - **Windows** `xscmd.bat -c showMapSizes -g ObjectGridA -ms MapSetA -m MapA -cep CatalogMachine:6645`
- 構成内の構成されている配置とランタイムの配置を表示するには、次のいずれかのコマンドを実行します。
 - `xscmd -c placementServiceStatus`
 - `xscmd -c placementServiceStatus -g ObjectGridA -ms MapSetA`
 - `xscmd -c placementServiceStatus -ms MapSetA`
 - `xscmd -c placementServiceStatus -g ObjectGridA`

配置情報については、全体の構成、1 つのデータ・グリッド、1 つのマップ・セット、またはデータ・グリッドとマップ・セットの組み合わせを表示するそれぞれのコマンドを使用できます。

2. 環境内の複製状態の要約を表示します。

- 各コンテナ・サーバーの未処理レビジョンの要約を表示します。**-ct** 引数を指定して特定のコンテナ・サーバーに対してコマンドを実行できます。または、引数を含めずに、すべてのコンテナ・サーバーに対して実行できます。
 - **UNIX** `./xscmd.sh -c showReplicationState -ct container1`
 - **Windows** `xscmd.bat -c showReplicationState -ct container1`

このコマンドの出力の情報には、アウトバウンド複製およびインバウンド複製が含まれています。アウトバウンド複製には、当該コンテナ・サーバー上のプライマリ断片から他のコンテナ・サーバー上のレプリカ断片にプッシュアウトする必要がある変更が含まれています。インバウンド複製には、他のコンテナ・サーバー上のプライマリ断片から当該コンテナ・サーバー上のレプリカにプッシュする必要がある変更が含まれています。これらの統計により、複製のヘルスを把握できます。コンテナ・サーバー上の未処理レビジョンの数が突然多くなった場合は、コンテナの問題が存在する可能性があります。

- カタログ・サービス・ドメイン間の断片の未処理レビジョンの要約を表示します。特定のコンテナ・サーバーおよびカタログ・サービス・ドメインに対してコマンドを実行できます。または、引数を含めずに、構成全体に対して実行できます。

- **UNIX** `./xscmd.sh -c showDomainReplicationState -dom domainA -ct container1`

- **Windows** `xscmd.bat -c showDomainReplicationState -dom domainA -ct container1`
このコマンドの出力の情報には、リンクされた各カタログ・サービス・ドメインの各コンテナ・サーバーの未処理リビジョンの要約が含まれています。このコマンドは、各プライマリ断片と、別のカタログ・サービス・ドメイン内にある対応するリモート・プライマリ断片との間で複製する必要がある変更を返します。

親トピック: [モニター](#)

関連タスク:

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

CSV ファイルによるモニター

モニター・データは自動的に CSV ファイルに書き込まれます。これらの CSV ファイルには、サーバー、マップ、またはデータ・グリッドに関する情報を含めることができます。

このタスクについて

デフォルトで CSV ファイルに書き込まれるモニター・データ。アプライアンス上で実行中のサーバーの履歴データをダウンロードし、分析することができます。サーバーが始動すると、データの収集が開始されます。その後は、この CSV ファイルをいつでもダウンロードでき、自由にファイルを使用できます。

手順

1. CSV ファイルをダウンロードします。アプライアンスからログ・ファイルをダウンロードする際、CSV ファイルは trace.zip ファイルに含まれています。このファイルをダウンロードするには、「アプライアンス」 > 「トラブルシューティング」 > 「ロギング」 > 「ログ・ファイルのダウンロード」をクリックします。trace.zip ファイル内で、CSV ファイルは server_name/logs ディレクトリーにあります。ファイル名は jvmstats.log、mapstats.log、および ogstats.log です。
2. データの処理に使用するプログラム (スプレッドシートなど) に CSV ファイルをインポートします。

CSV ファイルの統計定義

サーバーにダウンロード可能な CSV ファイルには、ヒストリカル・チャートを作成するために使用可能な統計やその他の情報が含まれています。

親トピック: [モニター](#)

関連タスク:

[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)

関連資料:

[CSV ファイルの統計定義](#)

CSV ファイルの統計定義

サーバーにダウンロード可能な CSV ファイルには、ヒストリカル・チャートを作成するために使用可能な統計やその他の情報が含まれています。

Java 仮想マシン (JVM) の統計ログ

TimeStamp (列 1)

Java 仮想マシン (JVM) に対して取られた統計スナップショットの日時を示します。

ServerName (列 2)

JVM のサーバー名を指定します。

Hostname (列 3)

JVM のホスト名を指定します。

FreeMemory (列 4)

JVM で使用可能なバイト数を指定します。

MaxMemory (列 5)

JVM に割り振り可能な最大バイト数を指定します。

TotalMemory (列 6)

サーバー実行時における実メモリ使用量を表示します。

AvailProcs (列 7)

このカタログ・サービスおよびそのマップで使用可能なプロセッサの数を表示します。最高の安定度を実現するために、60% のプロセッサ負荷でサーバー、さらに 60% のヒープ負荷で JVM ヒープを稼働してください。スパイクでプロセッサ使用量を 80% から 90% の間に引き上げることができますが、通常はこのレベルより高いレベルでサーバーを稼働しないようにしてください。

マップの統計ログ

TimeStamp (列 1)

マップに対して取られた統計スナップショットの日時を指定します。

MapName (列 2)

マップの名前を指定します。

OgName (列 3)

このマップが属しているデータ・グリッドの名前を指定します。

PartitionId (列 4)

区画 ID を指定します。

MapSetName (列 5)

このマップが属しているマップ・セットの名前を指定します。

HitRate (列 6)

選択したマップのヒット・レート (ヒット率) を表示します。高ヒット・レートが望ましい状態です。ヒット・レートは、パーシスタント・ストアへのアクセスの回避にデータ・グリッドがどのくらい役立っているかを示します。

Count (列 7)

サーバーの始動以降におけるデータ・グリッド内の項目数を示します。例えば、100 の値は、その項目がサーバーの始動以降収集された 100 番目のサンプル項目であることを示します。

TotalGetCount (列 8)

データを取得するためにマップがパーシスタント・ストアにアクセスする必要があった回数の合計を表示します。

TotalHitCount (列 9)

要求データがマップ内で検出され、パーシスタント・ストアにアクセスする必要がなかった回数の合計を表示します。

StartTime (列 10)

最後のリセット呼び出しからカウンターが開始する時刻を示します。リセットは、サーバーが始動または再始動したときに行われます。

LastCount (列 11)

最後のデータ・サンプルが取られてから経過した時間を示します。

LastTotalGetCount (列 12)

現在のキャッシュからの取得操作の総数から、前の期間の取得操作の数を引いた数を表示します。

LastTotalHitCount (列 13)

現在のキャッシュからのヒット総数から、前の期間のヒット数を引いた数を示します。

UsedBytes (列 14)

このマップによるメモリー消費量を表示します。使用バイト統計は、単純なオブジェクトまたは COPY_TO_BYTES コピー・モードを使用している場合に限り正確です。

MinUsedBytes (列 15)

このカタログ・サービスおよびそのマップによるメモリー消費量の最低点を表示します。使用バイト統計は、単純なオブジェクトまたは COPY_TO_BYTES コピー・モードを使用している場合に限り正確です。

MaxUsedBytes (列 16)

このカタログ・サービスおよびそのマップによるメモリー消費量の最高点を表示します。使用バイト統計は、単純なオブジェクトまたは COPY_TO_BYTES コピー・モードを使用している場合に限り正確です。

LastUsedBytes (列 17)

現在の UsedBytes 値から以前の統計収集期間の UsedBytes 値を引いた数を示します。

SampleLen (列 18)

データのサンプルが取られた期間の長さをミリ秒単位で示します。

ObjectGrid 統計ログ

TimeStamp (列 1)

データ・グリッドに対して取られた統計スナップショットの日時を指定します。

OgName (列 2)

データ・グリッドの名前を指定します。

PartitionId (列 3)

区画 ID を指定します。

Count (列 4)

サーバーの始動以降に収集されたデータ・グリッド項目の数を示します。例えば、100 の値は、その項目がサーバーの始動以降収集された 100 番目のサンプル項目であることを示します。

Hostname (列 5)

ホスト名を指定します。

DomainName (列 6)

このデータ・グリッドが属しているカタログ・サービス・ドメインを指定します。

MaxTime (列 7)

このサーバーで最も時間がかかった トランザクションで費やした時間を表示します。

MinTime (列 8)

このサーバーで最も時間がかからなかった トランザクションで費やした時間を表示します。

MeanTime (列 9)

トランザクションにかかった平均時間を示します。

TotalTime (列 10)

このサーバーでトランザクションに費やした、このサーバーの初期設定時からの合計時間を表示します。

AvgTransTime (列 11)

このサーバーでトランザクションを完了するために必要な平均時間を表示します。

AvgThroughPut (列 12)

このサーバーの 1 秒当たりのトランザクションの平均数を表示します。

SumOfSquares (列 13)

トランザクション時間の二乗値の合計を示します。この値は、ある時点での平均からの偏差を測定します。

SampleLen (列 14)

データのサンプルが取られた期間の長さをミリ秒単位で示します。

LastDataSample (列 15)

最後のサンプルが取られてから経過した時間を示します。

LastTotalTime (列 16)

現在のデータ・サンプルの合計時間から前の合計時間を引いた値を示します。

StartTime (列 17)

データの最後のリセット以降に統計の収集が開始された時刻を示します。サーバーが再始動すると、データはリセットされます。

親トピック: [CSV ファイルによるモニター](#)


関連タスク:

[WebSphere DataPower XC10 アプリアンス ログ・ファイルの操作](#)
[CSV ファイルによるモニター](#)

xsadmin ユーティリティーによるモニター

xsadmin ユーティリティーを使用して、WebSphere® DataPower® XC10 アプライアンス上で実行しているデータ・グリッドに関するテキスト情報をフォーマット設定し、表示できます。このサンプル・ユーティリティーは現在のデプロイメント・データの解析とディスクバリーの方法を提供するもので、カスタム・ユーティリティーの作成基盤として使用することができます。



始める前に

-  **2.5+** **xsadmin** ユーティリティーは現在非推奨になっています。代わりに、**xscmd** ユーティリティーを使用します。**xscmd** ユーティリティーは、環境のモニターおよび管理のための、サポートされるユーティリティーとして提供されています。詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

このタスクについて

xsadmin ユーティリティーは、Managed Bean (MBeans) の実装を使用します。[com.ibm.websphere.objectgrid.management](#) パッケージ内のインターフェースを使用して、このユーティリティーの機能を拡張できます。**xsadmin** アプリケーションのソース・コードは、スタンドアロン・インストールの場合は `wxs_client_home/samples/xsadmin.jar` ファイルで確認でき、WebSphere Application Server インストールの場合は `wxs_client_home/optionalLibraries/ObjectGrid/xsadmin.jar` ファイルで確認できます。

手順

1. アプライアンスのアクティブなトラストストアをクライアントにダウンロードします。WebSphere DataPower XC10 アプライアンスのユーザー・インターフェースで、「アプライアンス」 > 「設定」 > 「**Transport Layer Security**」 > 「アクティブなトラストストアのダウンロード」をクリックします。デフォルトのトラストストア・ファイルは、`xsatruststore.jks` ファイルです。このファイルのデフォルトのパスワードは、`xc10pass` です。
2. コマンド行で、`JAVA_HOME` 環境変数を設定します。
 -  `export JAVA_HOME=javaHome`
 -  `set JAVA_HOME=javaHome`
3. `bin` ディレクトリーに移動します。

```
cd wxs_client_home/bin
```

4. **xsadmin** ユーティリティーを実行します。アプライアンスに接続するには、ダウンロードしたトラストストアのセキュリティー引数、アプライアンスへのログインに使用するユーザー名とパスワード、およびアプライアンスのホスト名をコマンドを実行するたびに組み込む必要があります。

```
xsadmin.sh -trustPath xsatruststore.jks -trustType jks -ssl -trustPass xc10pass  
-username xcadmin -password xcadmin -ch myxc10.mycompany.com  
[additional_xsadmin_parameters]
```

構成ファイルを作成して、これらのパラメーターを保存することもできます。必要なパラメーターが組み込まれたプロパティー・ファイルの例を次に示します。

```
XSADMIN_TRUST_PATH=xsatruststore.jks  
XSADMIN_TRUST_TYPE=JKS  
XSADMIN_TRUST_PASS=xc10pass  
XSADMIN_USERNAME=xcadmin  
XSADMIN_PASSWORD=xcadmin
```

プロパティー・ファイルを指定して **xsadmin** ユーティリティーを実行するには、**-profile** 引数を使用してプロパティー・ファイルの場所を示します。

```
xsadmin.sh -profile myxc10.properties -ssl -ch myxc10.mycompany.com  
[additional_xsadmin_parameters]
```

[xsadmin ユーティリティー・リファレンス](#)

xsadmin ユーティリティーには、コマンド行引数を使用する方式、またはプロパティー・ファイルを使用する方式で、引数を渡すことができます。

[xsadmin ツールから xscmd ツールへのマイグレーション](#)


これまでのリリースでは、**xsadmin** ツールは環境の状態をモニターするサンプルのコマンド行ユーティリティーでした。**xscmd** ツールは、管理およびモニター用の、正式にサポートされるコマンド行ツールとして導入されました。これまで **xsadmin** ツールを使用していた場合は、新しい **xscmd** ツールにコマンドをマイグレーションすることを検討してください。

親トピック: [モニター](#)

xsadmin ユーティリティー・リファレンス

xsadmin ユーティリティーには、コマンド行引数を使用する方式、またはプロパティ・ファイルを使用する方式で、引数を渡すことができます。

xsadmin 引数

 **2.5+** 注: **xsadmin** ユーティリティーは現在非推奨になっています。代わりに、**xscmd** ユーティリティーを使用します。**xscmd** ユーティリティーは、環境のモニターおよび管理のための、サポートされるユーティリティーとして提供されています。詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

xsadmin ユーティリティー用のプロパティ・ファイルは、WebSphere® eXtreme Scale クライアント、バージョン 7.1 フィックス 1 以上で定義できます。プロパティ・ファイルを作成すると、ユーザー名などの使用頻度の高い引数の一部を保存できます。プロパティ・ファイルに追加できるプロパティは、次の表のとおりです。プロパティ・ファイル内のプロパティとそれに相当するコマンド行引数を両方指定した場合、コマンド行引数値がプロパティ・ファイルの値に優先します。

表 1. **xsadmin** ユーティリティーの引数

コマンド行引数	プロパティ・ファイル内で相当するプロパティ名	説明と有効な値
-bp	n/a	リスナー・ポートを指示します。 デフォルト:2809
-ch	n/a	カタログ・サーバーの JMX ホスト名を指示します。 デフォルト:localhost
-clear	n/a	指定されたマップをクリアします。 使用できるフィルター: -fm
-containers	n/a	データ・グリッドおよびマップ・セットのそれぞれについて、コンテナ・サーバーのリストを表示します。 使用できるフィルター: -fnp
-continuous	n/a	データ・グリッドをモニターするためにマップ・サイズの結果を継続的に表示させたい場合は、このフラグを指定してください。-mapsizes 引数を指定してこのコマンドを実行すると、20 秒ごとにマップ・サイズが表示されます。
-coregroups	n/a	カタログ・サーバーのすべてのコア・グループを表示します。この引数は、拡張診断に使用されます。
-dismissLink <catalog_service_domain>	n/a	このコマンド行引数は、WebSphere DataPower® XC10 アプライアンス構成には適用されません。
-dmgr	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
-empties	n/a	出力に空のコンテナを表示する場合、このフラグを指定します。
-establishLink <foreign_domain_name> <host1:port1, host2:port2...>	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
-fc	n/a	このコンテナのみにフィルタリングします。 WebSphere Application Server Network Deployment 環境でコンテナ・サーバーをフィルタリングする場合は、次の形式を使用します。

		<p><cell_name>/<node_name>/<serverName_containerSuffix></p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec</p>
-fh	n/a	<p>このホストのみにフィルタリングします。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec、-routetable</p>
-fm	n/a	<p>このマップのみにフィルタリングします。</p> <p>この形式を指定できる引数: -clear、-mapsizes</p>
-fnp	n/a	<p>プライマリ断片を持たないサーバーをフィルタリングします。</p> <p>この形式を指定できる引数: -containers</p>
-fp	n/a	<p>この区画のみにフィルタリングします。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec、-routetable</p>
-fs	n/a	<p>このサーバーのみにフィルタリングします。</p> <p>WebSphere Application Server Network Deployment 環境でアプリケーション・サーバーをフィルタリングする場合は、次の形式を使用します。</p> <p><cell_name>/<node_name>/<server_name></p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec</p>
-fst	n/a	<p>この断片タイプのみでフィルタリングします。プライマリ断片のみ場合は P を、非同期レプリカ断片のみ場合は A を、同期レプリカ断片のみ場合は S を指定します。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec</p>
-fz	n/a	<p>このゾーンのみにフィルタリングします。</p> <p>この形式を指定できる引数: -mapsizes、-teardown、-revisions、-getTraceSpec、-setTraceSpec、-getStatsSpec、-setStatsSpec、-routetable</p>
-force	n/a	<p>いずれのプリエンティブなプロンプトも無効にして、コマンド内のアクションを強制します。この引数はバッチ・コマンドを実行するときに便利です。</p>
-g	n/a	<p>ObjectGrid 名を指定します。</p>
-getstatsspec	n/a	<p>現在の統計仕様を表示します。統計仕様は、-setstatsspec 引数で設定できます。</p> <p>使用できるフィルター: -fst -fc -fz -fs -fh -fp</p>
-getTraceSpec	n/a	<p>現在のトレース仕様を表示します。トレース仕様</p>

<code>gettracespec</code>	n/a	<p>構成内のトレースは、<code>-settracespec</code> 引数で設定できます。</p> <p>使用できるフィルター: <code>-fst -fc -fz -fs -fh -fp</code></p>
<code>-h</code>	n/a	xsadmin ユーティリティのヘルプを表示します。ヘルプには引数のリストも含まれます。
<code>-hosts</code>	n/a	構成内のすべてのホストを表示します。
<code>-jmxUrl</code>	XSADMIN_JMX_URL	<p>JMX API コネクタ・サーバーのアドレスを次の形式で指定します: <code>service:jmx:protocol:sap</code>。 <code>protocol</code> および <code>sap</code> 変数の定義は、次のとおりです。</p> <p>protocol</p> <p>コネクタ・サーバーに接続するために使用するトランスポート・プロトコルを指定します。</p> <p>sap</p> <p>コネクタ・サーバーが検出されるアドレスを指定します。</p> <p>JMX サービス URL の形式の詳細については、Class JMXServiceURL (Java™ 2 Platform SE 5.0) を参照してください。</p>
<code>-l</code>	n/a	すべての既知のデータ・グリッドとマップ・セットを表示します。
<code>-m</code>	n/a	マップ・セットの名前を指定します。
<code>-mapsizes</code>	n/a	<p>断片間でキーの分布が一樣か確認するため、カタログ・サーバー上の各マップのサイズを表示します。</p> <p>使用できるフィルター: <code>-fm -fst -fc -fz -fs -fh -fp</code></p>
<code>-mbeanservers</code>	n/a	すべての MBean サーバー・エンドポイントのリストを表示します。
<code>-overridequorum</code>	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
<code>-password</code>	XSADMIN_PASSWORD	xsadmin ユーティリティにログインするためのパスワードを指定します。パスワードを保護する必要がある場合、プロパティ・ファイル内にパスワードは指定しないでください。
<code>-p</code>	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
	n/a	<p>構成内の構成されている配置とランタイムの配置を表示します。出力の有効範囲はデータ・グリッドとマップ・セットの組み合わせにしたり、構成全体にしたりできます。</p> <ul style="list-style-type: none"> 構成全体: <pre>-placementStatus</pre> 特定のデータ・グリッドの場合: <pre>-placementStatus -g my_grid</pre> 特定のマップ・セットの場合: <pre>-placementStatus -m my_mapset</pre> 特定のデータ・グリッドとマップ・セットの場合: <pre>-placementStatus -g my_grid -m my_mapset</pre>

-primaries	n/a	プライマリー断片のリストを表示します。
-profile	n/a	xsadmin ユーティリティー用のプロパティ・ファイルへの完全修飾パスを指定します。
-quorumstatus	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。
-releaseShard <container_server_name> <objectgrid_name> <map_set_name> > <partition_name>	n/a	-reserveShard 引数と一緒に使用されます。 -releaseShard 引数は、断片が予約されて配置された後に呼び出されます。 -releaseShard 引数は、ContainerMBean.release() メソッドを呼び出します。
-reserved	n/a	-reserveShard 引数によって予約された断片のみを表示するために -containers 引数と一緒に使用されます。
-reserveShard <container_server_name> <objectgrid_name> <map_set_name> > <partition_name>	n/a	プライマリー断片を指定されたコンテナ・サーバーに移動します。この引数によって、ContainerMBean.reserve() メソッドが呼び出されます。
-resumeBalancing <objectgrid_name> <map_set_name> >	n/a	要求のバランスを取るを試みます。指定された ObjectGrid とマップ・セットの将来の再バランシング試行を可能にします。
-revisions	n/a	カタログ・サービス・ドメインの改訂 ID を表示します。これには、特定の断片ごとの各データ・グリッド、区画番号、区画タイプ (基本またはレプリカ)、カタログ・サービス・ドメイン、存続期間 ID、およびデータ改訂の回数が含まれます。この引数を使用して、非同期レプリカまたはリンクされたドメインがキャッチアップされたかどうかを判別できます。この引数は、ObjectGridMBean.getKnownRevisions() メソッドを呼び出します。 使用できるフィルター: -fst -fc -fz -fs -fh -fp
-routetable	n/a	クライアント・サーバーの観点から、データ・グリッドの現在の状態を表示します。経路テーブルは、ObjectGrid クライアント・サーバーがデータ・グリッドとの通信に使用する情報です。接続問題または TargetNotAvailable 例外を特定する場合に、診断援助として経路テーブルを使用します。 必須引数: スタンドアロン環境では、カタログ・サーバー・ホストのブートストラップ・リスナー・ポートおよび JMX ポートにデフォルト値を使用していない場合、この引数と一緒に -bp パラメーターおよび -p パラメーターを指定しなければなりません。 使用できるフィルター: -fz -fh -fp
-settracespec <trace_string> >	n/a	このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。アプライアンスのトレースの構成について詳しくは、 WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作

		<p>を参照してください。</p> <p>使用できるフィルター: -fst -fc -fz -fs -fh -fp</p>
-swapShardWithPrimary <container_server_name> <objectgrid_name> <map_set_name> > <partition_name>	n/a	指定されたコンテナ・サーバーからの指定されたレプリカ断片を、プライマリ断片と交換します。このコマンドを実行することで、必要なときに手動でプライマリ断片をバランシングできます。
-setstatsspec <stats_spec>	n/a	統計の収集を有効にします。この引数は、DynamicServerMBean.setStatsSpec メソッドと DynamicServerMBean.getStatsSpec メソッドを呼び出します。 使用できるフィルター: -fm -fst -fc -fz -fs -fh -fp
-suspendBalancing <objectgrid_name> <map_set_name> >	n/a	指定された ObjectGrid およびマップ・セットの将来のバランシング試行をさせません。
-ssl	n/a	Secure Sockets Layer (SSL) が有効なことを示します。
-teardown	n/a	<p>使用できるフィルター: -fst -fc -fz -fs -fh -fp</p> <p>サーバーのリストを指定する場合の形式:</p> <pre>server_name_1,server_name_2 ...</pre> <p>特定ゾーン内のすべてのサーバーを停止する場合、-fz 引数を組み込みます。</p> <pre>-fz <zone_name></pre> <p>特定ホスト上のすべてのサーバーを停止する場合、-fh 引数を組み込みます。</p> <pre>-fh <host_name></pre> <p>このコマンド行引数は、WebSphere DataPower XC10 アプライアンス構成には適用されません。</p>
	n/a	デプロイメント XML ファイルに構成されている numInitialContainers の値を無視して、断片配置の実行を強制します。サーバーのメンテナンスを実行中、この引数を使用すると、 numInitialContainers の値が構成値より小さい場合でも断片配置の実行を続行できるようになります。
-trustPass	XSADMIN_TRUST_PASS	指定したトラストストアのパスワードを指定します。
-trustPath	XSADMIN_TRUST_PATH	トラストストア・ファイルへのパスを指定します。 例: etc/test/security/server.public
-trustType	XSADMIN_TRUST_TYPE	トラストストアのタイプを指定します。

	<code>-type</code>	有効値: JKS、JCEK、PKCS12 など。
<code>-unassigned</code>	n/a	データ・グリッド上に配置できない断片のリストを表示します。配置サービスに配置を妨げる制約があると、断片は配置できません。
<code>-username</code>	XSADMIN_USERNAME	xsadmin ユーティリティーにログインするためのユーザー名を指定します。
<code>-v</code>	n/a	詳細コマンド行アクションを使用可能にします。環境変数、プロパティ・ファイル、またはその両方を使用して特定のコマンド行引数を指定し、それらの値を表示する必要がある場合、このフラグを使用します。
<code>-xml</code>	n/a	PlacementServiceMBean.listObjectGridPlacement() メソッドからのフィルタリングされていない出力を表示します。他の xsadmin 引数は、このメソッドの出力をフィルタリングし、データをさらに消費可能な形式に編成します。

親トピック: [xsadmin ユーティリティーによるモニター](#)

xsadmin ツールから xscmd ツールへのマイグレーション

これまでのリリースでは、**xsadmin** ツールは環境の状態をモニターするサンプルのコマンド行ユーティリティでした。 **xscmd** ツールは、管理およびモニター用の、正式にサポートされるコマンド行ツールとして導入されました。これまで **xsadmin** ツールを使用していた場合は、新しい **xscmd** ツールにコマンドをマイグレーションすることを検討してください。

xsadmin および xscmd コマンド同等


 **2.5+ 重要:** **xsadmin** ユーティリティは現在非推奨になっています。代わりに、**xscmd** ユーティリティを使用します。 **xscmd** ユーティリティは、環境のモニターおよび管理のための、サポートされるユーティリティとして提供されています。詳しくは、[xscmd ユーティリティによる管理](#)を参照してください。

表 1. **xsadmin** ユーティリティの引数と、**xscmd** 同等コマンド。一部の **xscmd** コマンドは、短い形式と長い形式を持ちます。短い形式のダッシュ (-) を持ち、長い形式のコマンドは、2 つのダッシュ (--) を持ちます。どちらの形式も区別なく使用できます。

xsadmin コマンド行引数	xscmd 同等コマンド	xscmd コマンド・パラメーター
-bp	<ul style="list-style-type: none"> • -cep <i>hostname:listener_port</i> • --catalogEndpoint <i>hostname:listener_port</i> 	n/a
-ch	<ul style="list-style-type: none"> • -cep <i>hostname:listener_port</i> • --catalogEndpoint <i>hostname:listener_port</i> 	n/a
-clear	-c clearGrid	-g, -ms, -v, -m, (-cep)
-containers	<ul style="list-style-type: none"> • -c showPlacement -container <i>containerName</i> • -c showPlacement -server <i>serverName</i> 	-e, -i, , -st, -snp, -ct, -s, -p, -hf,
-continuous	n/a	n/a
-coregroups	<ul style="list-style-type: none"> • -c listCoreGroupMembers -cg <i>core_group</i> 	n/a
-dismissLink <i><catalog_service_domain></i>	-c dismissLink	<ul style="list-style-type: none"> • -fd <i><foreignCatalogServiceDomain></i> • --foreignCatalogServiceDomain <i><foreignCatalogServiceDomain></i>
-dmgr	n/a - この引数は、 xscmd で自動的に判別される	n/a
-empties	新規コマンドに固有の引数	n/a
-establishLink <i><foreign_domain_name></i> <i><host1:port1,host2:port2...></i>	-c establishLink	<ul style="list-style-type: none"> • -fd <i><foreignCatalogServiceDomain></i> <i><host1:port1,host2:port2...></i> • --foreignCatalogServiceDomain <i><foreignCatalogServiceDomain></i> -fd <i><host1:port1,host2:port2...></i>
-fc	<ul style="list-style-type: none"> • -ct • --container 	n/a
-fh	<ul style="list-style-type: none"> • -hf • --hostFilter 	n/a
-fm	<ul style="list-style-type: none"> • -m • --map 	n/a
-fnp	<ul style="list-style-type: none"> • -snp • --serversWithNoPrimaries 	n/a
-fp	<ul style="list-style-type: none"> • -p • --partitionId 	n/a
-fs	<ul style="list-style-type: none"> • -s • --server 	n/a

-fst	<ul style="list-style-type: none"> • -st <shard_type> • --shardType <shard_type> 断片値: P=primary A=asyncReplica S=syncReplica	n/a
-fz	<ul style="list-style-type: none"> • -z • --zone 	n/a
-force	新規コマンドに固有の引数	
-g	<ul style="list-style-type: none"> • -g • --objectGrid 	n/a
-getstatsspec	-c getStatsSpec	n/a
-getTraceSpec	-c getTraceSpec	n/a
-h	特定のコマンド名を付けても付けなくても help を実行できる <ul style="list-style-type: none"> • -h • --help • -h <command_name> • --help <command_name> 	n/a
-hosts	-c listHosts	-g, -ms, -st, -c, -s, -hf, -z
-jmxUrl	<ul style="list-style-type: none"> • -cep hostname:listener_port • --catalogEndpoint hostname:listener_port 	n/a
-l	-c listObjectGridNames	n/a
-m	<ul style="list-style-type: none"> • -ms • --mapSet 	n/a
-mapsizes	-c showMapSizes	-g, -ms, -i, [-ct, -z, -s, -hf, sht [P
-mbeanservers	-c listAllJMXAddresses	n/a
-overridequorum	-c overrideQuorum	n/a
-password	<ul style="list-style-type: none"> • -pwd • --password 	n/a
-p	<ul style="list-style-type: none"> • -cep hostname:listener_port • --catalogEndpoint hostname:listener_port 	n/a
-placementStatus	-c placementServiceStatus	-g, -ms
-primaries	-c showPlacement -sf P	-e, -i, , -st, -snp, -ct, -s, -p, -hf,
-profile	現行セキュリティー設定をセキュリティー・ プロファイルとして保存する場合: <ul style="list-style-type: none"> • -ssp profile_name • --saveSecProfile profile_name 指定されたセキュリティー・プロファイル を使用する場合: <ul style="list-style-type: none"> • -sp profile_name • --securityProfile profile_name 	
-quorumstatus	-c showQuorumStatus	n/a
-releaseShard	-c releaseShard	-c, -g, -ms, -p
重要: このコマンド行引数は、 WebSphere® DataPower® XC10 アプ ライアンス構成には適用され ません。	重要: このコマンド行引数は、 WebSphere DataPower XC10 アプ ライアンス構成には適用され ません。	

-reserved	<ul style="list-style-type: none"> • -sf R • --shardFilter R 	n/a
-reserveShard 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c reserveShard 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c, -g, -ms, -p
-resumeBalancing <objectgrid_name> <map_set_name> 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c resumeBalancing 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-g, -ms
-revisions	-c revisions	-s, -p, -g, -m
-routetable	-c routetable	-z, -hf, -p, -g, -ms
-settracespec <trace_string>	-c setTraceSpec	-spec <trace_string>
-swapShardWithPrimary 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c swapShardWithPrimary 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c -g, -ms, -p
-setstatsspec <stats_spec>	-c setStatsSpec	-spec <stats_spec>
-suspendBalancing <objectgrid_name> <map_set_name> 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c suspendBalancing 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-g, -ms
-ssl	<ul style="list-style-type: none"> • -ssl • --enableSSL 	n/a
-teardown 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c teardown 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-f, , -st, -snp, -c, -s, -p, -hf, -z,
-triggerPlacement 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-c triggerPlacement 重要: このコマンド行引数は、WebSphere DataPower XC10 アプリケーション構成には適用されません。	-g, -ms
-trustPass	<ul style="list-style-type: none"> • -tsp • --trustStorePassword 	n/a
-trustPath	<ul style="list-style-type: none"> • -ts 	n/a

	<ul style="list-style-type: none"> • <code>--trustStore</code> 	
<code>-trustType</code>	<ul style="list-style-type: none"> • <code>-tst</code> • <code>--trustStoreType</code> 	n/a
<code>-unassigned</code>	<code>-c showPlacement -sf U</code>	<code>-e, -i, , -st, -snp, -ct, -s, -p, -hf,</code>
<code>-username</code>	<ul style="list-style-type: none"> • <code>-user</code> • <code>--username</code> 	n/a
<code>-v</code>	<ul style="list-style-type: none"> • <code>-v</code> • <code>--verbose</code> 	n/a
<code>-xml</code>	<code>-c showPlacement</code>	n/a

親トピック: [xsadmin ユーティリティによるモニター](#)

環境のヘルスのモニター

2.5+ メッセージ・センターでは、ログおよび初期障害データ・キャプチャー機能 (FFDC) メッセージのイベント通知の集約ビューが示されます。これらのイベント通知は、Web コンソールのメッセージ・センター、**xscmd** ユーティリティ、ヘルス・ログ・ファイル、HTTP コマンド・インターフェース、または MBeans を使用したプログラマチック方式で表示することができます。

2.5+ [ヘルス・モニターの概要](#)

メッセージ・センターは、集合内のすべてのコンテナ・サーバーおよびカタログ・サーバーからヘルス状況イベントをリアルタイムで収集します。メッセージ・センターが構成されていると、さまざまなサーバーで発生している現行クリティカル・イベントの概要を、各サーバーのログを収集せずに表示することができます。

2.5+ [メッセージ・センターでのヘルス・イベント通知の表示](#)

Web コンソールでメッセージ・センターを使用して、データ・グリッドと 集合全体のリアルタイム・ヘルスを評価できます。メッセージ・センターに表示されるイベントは、最も重大な問題を表示するようにフィルターが掛けられた、イベントのサブセットです。

2.5+ [xscmd ユーティリティを使用したヘルス通知の表示](#)

xscmd ユーティリティを使用して、現在のイベント通知を表示したり、イベント通知履歴を表示したり、メッセージ・センターから通知フィルターを設定したりすることができます。

次のタスク

アプライアンスのハードウェアおよびソフトウェアのヘルス状況全体の要約を表示することができます。以下のいずれかのオプションを使用します。

- ヘルス・ログ・ファイルを表示してダウンロードする。詳しくは、[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)を参照してください。
- HTTP コマンド・インターフェースの **GetHealthStatus** コマンドを使用して、アプライアンスのハードウェアおよびソフトウェアのヘルス状況全体を表示する。詳しくは、[HTTP コマンド・インターフェースでの管理](#)および [HTTP コマンド・インターフェースのリファレンス](#)を参照してください。

親トピック: [モニター](#)

ヘルス・モニターの概要

2.5+ メッセージ・センターは、集合内のすべてのコンテナ・サーバーおよびカタログ・サーバーからヘルス状況イベントをリアルタイムで収集します。メッセージ・センターが構成されていると、さまざまなサーバーで発生している現行クリティカル・イベントの概要を、各サーバーのログを収集せずに表示することができます。

メッセージ・センターの実装

メッセージ・センターはデフォルトで使用可能です。ユーザー・インターフェースでメッセージ・センターを使用不可にすることができます。

データ・グリッド・デプロイメントは多数の分散サーバー・プロセスを伴うことがあります。問題が発生した場合は、影響を受けたコンテナ・サーバーの現行ログ・ファイルを開いて、その問題を詳しく分析することができます。

メッセージ・センターは以下のコンポーネントから成ります。

イベント集約

カタログ・サーバーに対するヘルス・モニターを構成すると、カタログ・サービス・ドメイン全体のヘルスに影響を及ぼしている集約イベントを受け取ります。この枠組みには、次に示すタイプのイベントの原因と重大度の指示が含まれます。

- すべての FFDC イベント
- すべての WARNING または SEVERE ログ・エントリ
- フィルタリングされたすべてのログ・エントリのリスト (INFO、WARNING、および SEVERE ログ・エントリを含む)
- サーバー始動操作およびサーバー停止操作
- データ・グリッドがほぼ満杯である
- クォーラムの損失または回復
- 使用可能に設定された SNMP トラップがトリガーされる
- 複製が 15 分間遅れる

Web コンソール内のメッセージ・センター

Web コンソール内のメッセージ・センターは集約されたイベント・レコードを表示します。これらのイベントには、最近発生したイベントと、コンソールが開かれた後から発生したイベントのリアルタイム更新通知の両方が含まれます。

xscmd ユーティリティーでのイベント

xscmd ユーティリティーを使用して最近発生したイベントのリストを表示することもできます。イベントの発生について、イベント・レコードをリダイレクトして自動スクリプティング・ユーティリティーを作成することができます。

他のモニター・ソフトウェアとの統合のための MBean

有効な管理 MBean を使用してメッセージ・センターを他の Java Management Extensions (JMX) モニター・ソフトウェアにプラグインすることもできます。この MBean の資料は API 資料に含まれています。

2.5+

ヘルスの要約

ヘルス・ログおよび HTTP コマンド・インターフェースの **GetHealthStatus** コマンドを使用して、アプライアンスのソフトウェアおよびハードウェアのヘルス状況の要約を取得することができます。この情報には、データ・グリッドの配置と複製の状況、ハードウェア警告、システムとネットワークのヘルス状況、およびその他の状況メッセージなどがあります。

ログ・アナライザーと対比したメッセージ・センター

ログ・アナライザーは、一連のログ・メッセージを分析するもう 1 つのツールです。このツールを使用するには、環境内のさまざまなサーバーからログを手動で収集する必要があります。その後、このツールを実行して問題条件のレポートを作成することができます。メッセージ・センターで表示できるメッセージ数 (1000 メッセージ) を超える一連のメッセージを分析する必要があるときには、ログ・アナライザーを使用してログの事後分析を実施してください。発生している問題を素早く特定するには、メッセージ・センターを使用してデータ・グリッドのヘルスをリアルタイムでモニターしてください。そうすれば、関連するコンテナ・サーバーのログ・ファイルを検討することもできれば、ログ・アナライザーを使用して問題を詳しく調査することもできます。

ヘルス・モニターの構成とアーキテクチャー

メッセージ・センターはデフォルトで使用可能です。ユーザー・インターフェースでメッセージ・センターを使用不可にすることができます。メッセージ・センターが使用可能に設定されると、集合内のカタログ・サーバーがヘルス・モニターのためのハブとしてアクティブ化されます。一般に、メッセージ・センター内のメッセージの送信元は、最初に集合を作成したときのアプライアンス上にあるカタログ・サーバーです。各ハブには、独自のサブスクリプションと個別のイベント・ヒストリーがあります。ヒストリー内の各イベントにはシーケンス番号が付いています。個々のカタログ・サーバーにあるイベント・ヒストリーは同期されず、それぞれ異なります。カタログ・サーバーは、他のカタログ

グ・サーバーからのログ・イベントおよび FFDC イベントにサブスクライブすることができます。

親トピック: **2.5+** [環境のヘルスのモニター](#)

メッセージ・センターでのヘルス・イベント通知の表示

Web コンソールでメッセージ・センターを使用して、データ・グリッドと 集合全体のリアルタイム・ヘルスを評価できます。メッセージ・センターに表示されるイベントは、最も重大な問題を表示するようにフィルターが掛けられた、イベントのサブセットです。

手順

- Web コンソールで、重大エラー、FFDC メッセージ、およびサーバーの始動と停止のイベントの通知を表示します。これらの通知は、Web コンソールで任意のページにログインすると、自動的に表示されます。
- メッセージ・センターでメッセージを表示します。Web コンソールで、「モニター」 > 「メッセージ・センター」をクリックします。メッセージ・センターには、カタログ・サーバー・メッセージ・ハブを介して送信された、最新の 1000 件の重大メッセージが表示されます。

カタログ・サーバー・メッセージ・ハブは、1000 件のメッセージが表示されるように、表示されるメッセージをフィルターに掛けます。そのため、メッセージ・センターには、カタログ・サーバーで発生している全重大イベントのサブセットのみが含まれています。ページを開いた状態のときに新しいメッセージが使用可能になると、ページを最新表示するオプションを備えた情報メッセージが、ページの上部に表示されます。

- メッセージ・センターに表示されるメッセージをフィルターに掛けます。最大で 3 つのフィルター・ルールを追加できます。ルールは、列、条件、および値から成ります。
 1. Web コンソールで、「モニター」 > 「メッセージ・センター」をクリックします。
 2. フィルター・ボタン (🔍) をクリックします。
 3. ルールを追加します。
 - a. 追加ボタン (+) をクリックします。
 - b. メッセージ・センターで、フィルターに掛ける列を以下のものから選択します。

ID

メッセージ・センターで生成されたイベント ID。

タイプ

メッセージの重大度を示す、メッセージのタイプ。有効な値は、「重大」、「警告」、「エラー」、および「通知」です。

日付

メッセージが生成された日時。

ソース

メッセージが発信されたサーバー。

メッセージ

メッセージ・イベントのメッセージ・テキスト。

- c. フィルターを適用する条件を選択します。日付およびタイプ以外の、ほとんどの列で有効な条件を以下のリストに示します。
 - 次を含む (Contains)
 - 次と等しい (Is)
 - 次で始まる (Starts with)
 - 次で終わる (Ends with)
 - d. 列をフィルターに掛ける値を入力します。

例: server1 のメッセージのみを表示するには、「ソース」列を選択します。次に、「次と等しい (Is)」条件を選択します。値として、「server1」を入力します。
4. 定義したルールの一部またはすべてと突き合わせるように選択できます。
 5. 「フィルター」をクリックして、構成されているフィルターをメッセージ・センターの出力に適用します。
- **2.5+** メッセージ・センターを使用不可に設定します。メッセージ・センターを使用不可に設定すると、新しいメッセージのポーリングが停止されます。「メッセージ・センターおよびトースター・アラートを使用不可に設定」をクリックします。メッセージ・センターを使用不可に設定した後、設定を適用する必要があります。

次のタスク

ログ・ファイルについて詳細な分析を実行できます。詳しくは、[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)を参照してください。

親トピック: **2.5+** [環境のヘルスのモニター](#)

xscmd ユーティリティーを使用したヘルス通知の表示

2.5+ xscmd ユーティリティーを使用して、現在のイベント通知を表示したり、イベント通知履歴を表示したり、メッセージ・センターから通知フィルターを設定したりすることができます。

始める前に

- xscmd ユーティリティーを開始し、カタログ・サービス・ドメインに接続します。詳しくは、[xscmd ユーティリティーによる管理](#)を参照してください。

手順

- xscmd ユーティリティーを使用して、イベント通知履歴を表示します。出力が表形式で表示されます。

```
xscmd -c showNotificationHistory -cep hostname:port(,hostname:port)
```

- 新規通知を listen します。

```
xscmd -c listenForNotifications -cep hostname:port(,hostname:port)
```

出力はロー形式であり、コマンドを停止するまで出され続けます。追加スクリプトを作成して、出力を構文解析できます。

- 今後作成されるすべてのログ・エントリ (INFO、WARNING、SEVERE ログ・エントリなど) に対してフィルタリングを使用可能に設定します。。デフォルトでは、メッセージ・センターとコマンドでは、WARNING エラー、SEVERE エラー、およびイベントのみが表示されます。環境内のすべてのサーバー、または単一のサーバーに対してフィルターを設定できます。この設定は今後の結果にのみ影響します。

```
xscmd -c setNotificationFilter -fs <regular expression> [-server <servername>]
```

- 環境内のすべてのサーバーまたは単一のサーバーの現在の通知フィルターを表示します。

```
xscmd -c getNotificationFilter [-s servername]
```

親トピック: **2.5+** [環境のヘルスのモニター](#)

Simple Network Monitoring Protocol (SNMP) によるモニター

Simple Network Monitoring Protocol (SNMP) サポートにより、IBM® WebSphere® DataPower® XC10 アプライアンス の状況を、データ・センターのシステムのより大きなグループの一部として、モニターできます。SNMP モニターは、システム問題を素早く通知し、それらに迅速に応答する機能を強化します。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成しなくても以下の手順を実行できますが、SNMP モニターを使用できるのは、クライアントを構成した場合のみです。WebSphere DataPower XC10 アプライアンス は、SNMP バージョン v2c をサポートします。

このタスクについて

以下の手順に従って、IBM WebSphere DataPower XC10 アプライアンスで SNMP モニターを使用可能にします。SNMP 設定はアプライアンスに固有であり、集合内の他の アプライアンスには伝搬されません。

要確認: 1 つのアプライアンスに関する SNMP 統計が示されるため、報告される値は、ユーザー・インターフェースのモニター・パネルで報告される値と異なります。ユーザー・インターフェースでは、集合に関するデータが報告されます。例えば、SNMP gridStats 表、mapStats 表、または jvmStats 表の値は、ユーザー・インターフェースで報告される合計値の一部を報告します。

1. [アプライアンスでの SNMP モニターの使用可能化](#)

IBM WebSphere DataPower XC10 アプライアンス では、Simple Network Monitoring Protocol (SNMP) モニターを使用可能にすることができます。SNMP クライアントが使用できる SNMP データを指定するには、アプライアンスで提供される管理情報ベース (MIB) をダウンロードします。

2. [SNMP コミュニティーの構成](#)

1 つ以上の Simple Network Monitoring Protocol (SNMP) コミュニティーを作成することで、アプライアンス上の SNMP データへのアクセスを定義できます。モニターが使用可能な場合は、SNMP コミュニティーが必要です。

3. [SNMP トラップ・サブスクリプションの構成](#)

トラップ・サブスクリプションは、アプライアンスに組み込まれた Simple Network Monitoring Protocol (SNMP) エージェントとの通信に使用される SNMP クライアントです。

4. [SNMP トラップ・サブスクリプションの管理](#)

Simple Network Monitoring Protocol (SNMP) トラップは、アプライアンスに組み込まれた SNMP エージェントによって開始されるイベント通知または状態通知です。SNMP トラップ・サブスクリプションを使用して、エージェントが SNMP クライアントに伝達する SNMP トラップを定義します。WebSphere DataPower XC10 アプライアンス は、SNMP バージョン v2c をサポートします。

親トピック: [モニター](#)

アプライアンスでの **SNMP** モニターの使用可能化

IBM® WebSphere® DataPower® XC10 アプライアンスでは、Simple Network Monitoring Protocol (SNMP) モニターを使用可能にすることができます。SNMP クライアントが使用できる SNMP データを指定するには、アプライアンスで提供される管理情報ベース (MIB) をダウンロードします。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成しなくても以下の手順を実行できますが、SNMP モニターを使用できるのは、クライアントを構成した場合のみです。WebSphere DataPower XC10 アプライアンスは、SNMP バージョン v2c をサポートします。

このタスクについて

デフォルトでは、IBM WebSphere DataPower XC10 アプライアンスでは、SNMP モニターが使用可能になっていません。SNMP 設定はアプライアンスに固有であり、集合内の他のアプライアンスには伝搬されません。

手順

1. 「モニター」パネルにナビゲートします。IBM WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースで、「アプライアンス」 > 「**SNMP 設定**」をクリックします。
2. SNMP モニターを有効にするには、チェック・ボックスにチェック・マークを付けます。SNMP モニターを無効にする場合は、チェック・ボックスのチェックを外します。
3. アプライアンスで使用可能な管理情報ベース (MIB) ファイルをダウンロードします。

エンタープライズ MIB ファイルは、組み込み SNMP エージェントで 使用可能な機能とデータを記述して、クライアントが適切にこれらの機能とデータにアクセスできるようにします。クライアントは、SNMP GET、GET-NEXT、および GET-BULK コマンドを発行できます。MIB ファイルをダウンロードし、クライアントにインポートすると、基本 MIB-II データ定義以外のデータにアクセスできます。アプライアンスから MIB をダウンロードするには、エンタープライズ **MIB** を展開し、各 MIB の名前をクリックします。

統計 MIB

統計 MIB には、ユーザー・インターフェースのモニター機能を使用して 確認できる統計と類似した情報が含まれます。この MIB には Java™ 仮想マシン (JVM) の統計も含まれます。

HW 状況 MIB

HW 状況 MIB には、温度や日時などのハードウェアの状態に関する情報が含まれます。

HW 通知 MIB

HW 通知 MIB は、SNMP TRAP コマンドで使用可能な情報を定義するために使用されます。

タスクの結果

上記手順を終えると、アプライアンスで SNMP モニターが使用可能になり、SNMP クライアント用の MIB データがダウンロードされます。

次のタスク

SNMP クライアントを使用して MIB データを表示します。

要確認: 1 つのアプライアンスに関する SNMP 統計が示されるため、報告される値は、ユーザー・インターフェースのモニター・パネルで報告される値と異なります。ユーザー・インターフェースでは、集合に関するデータが報告されます。例えば、SNMP 統計の gridStats 表、mapStats 表、または jvmStats 表の値は、ユーザー・インターフェースで報告される合計値の一部です。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

次のトピック: [SNMP コミュニティーの構成](#)

SNMP コミュニティーの構成

1 つ以上の Simple Network Monitoring Protocol (SNMP) コミュニティーを作成することで、アプライアンス上の SNMP データへのアクセスを定義できます。モニターが使用可能な場合は、SNMP コミュニティーが必要です。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成しなくても以下の手順は実行できますが、SNMP モニターを使用できるのは、クライアントを構成した場合のみです。WebSphere® DataPower® XC10 アプライアンスは SNMP バージョン v2c をサポートします。

このタスクについて

アプライアンスに組み込まれた SNMP エージェントを認証し、SNMP データにアクセスするには、コミュニティが必要です。アプライアンスに組み込まれた SNMP エージェントが要求されたデータを返すには、エージェントを認証するための定義済みのコミュニティ名をクライアントが指定する必要があります。コミュニティが指定されていない場合、SNMP エージェントはクライアントからの要求を無視します。SNMP コミュニティーはアプライアンスに固有であり、集合内の他のアプライアンスには伝搬されません。

手順

1. 「モニター」パネルにナビゲートします。IBM® WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース上部のメニュー・バーで、「アプライアンス」>「**SNMP モニター**」にナビゲートします。
2. 「**SNMP コミュニティー**」を展開します。
3. 「**コミュニティの作成**」をクリックします。
4. 作成する SNMP コミュニティーを記述するためのフォームに入力します。

名前

このフィールドには、SNMP コミュニティーの説明に使用する名前を指定します。

ホスト制限

このフィールドには、通信で使用する IP アドレスを IPv4 形式および IPv6 形式で指定します。また、このフィールドはホストに解決されるホスト名を受け入れます。クラスレス・ドメイン間ルーティング (CIDR) IP アドレスまたはホスト名を使用して、指定したサブネットにさらにアクセスを制限することができます。ホスト制限が指定されている場合、他の IP アドレスまたはサブネットとの通信はすべて拒否されます。このフィールドを空にすると、すべての IP アドレスとの通信が許可されます。

5. SNMP コミュニティーを除去するには、除去アイコン (✖) をクリックします。既存のコミュニティは変更できません。コミュニティを変更する必要がある場合は、削除してから再作成する必要があります。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

前のトピック: [アプライアンスでの SNMP モニターの使用可能化](#)

次のトピック: [SNMP トラップ・サブスクリバターの構成](#)

SNMP トラップ・サブスクリバの構成

トラップ・サブスクリバは、アプライアンスに組み込まれた Simple Network Monitoring Protocol (SNMP) エージェントとの通信に使用される SNMP クライアントです。

始める前に

SNMP トラップ・サブスクリバは、既存の SNMP クライアントである必要があります。IBM® WebSphere® DataPower® XC10 アプライアンス ユーザー・インターフェース でトラップ・サブスクリバを作成する前に、トラップを受信するように SNMP クライアントを構成します。

このタスクについて

サブスクリバした SNMP クライアントに、アプライアンスに関する通知を送信するには SNMP トラップ・サブスクリバが必要です。トラップ・サブスクリバの作成時に、SNMP トラップを受信する SNMP クライアントに関する情報を指定する必要があります。トラップは、IBM WebSphere DataPower XC10 アプライアンス 管理者が注意を払う必要のある状況を記述します。SNMP エージェントは、サブスクリバとして定義されていないクライアントに SNMP トラップを送信しません。

SNMP トラップ・サブスクリバはすべて、データ・グリッドのストレージ制限とアプライアンスのストレージ制限に関するトラップを自動受信します。設定可能な具体的なトラップについては、[SNMP トラップ・リファレンス](#) を参照してください。

手順

1. 「SNMP 設定」パネルにナビゲートします。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェースで、「アプライアンス」 > 「**SNMP 設定**」をクリックします。
2. 「トラップ・サブスクリバ」を展開します。
3. 「トラップ・サブスクリバの作成」をクリックします。
4. 作成する SNMP トラップ・サブスクリバを記述するためのフォームに入力します。

ホスト

SNMP クライアントがトラップ情報を listen する対象の IP アドレスを指定します。

クライアント・ポート番号

トラップ情報を listen している SNMP クライアントの所属元のポートを指定します。

コミュニティ

クライアントがメンバーとして所属している SNMP コミュニティを指定します。

5. 除去アイコン (✖) をクリックして、SNMP トラップ・サブスクリバを削除します。既存のサブスクリバは変更できません。サブスクリバを変更する必要がある場合は、削除してから再作成する必要があります。

次のタスク

アプライアンスにアクセスできる少なくとも 1 つの SNMP トラップ・サブスクリバを指定すると、SNMP エージェントに伝達させるトラップを構成できます。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

前のトピック: [SNMP コミュニティの構成](#)

次のトピック: [SNMP トラップ・サブスクリプションの管理](#)

SNMP トラップ・サブスクリプションの管理

Simple Network Monitoring Protocol (SNMP) トラップは、アプライアンスに組み込まれた SNMP エージェントによって開始されるイベント通知または状態通知です。SNMP トラップ・サブスクリプションを使用して、エージェントが SNMP クライアントに伝達する SNMP トラップを定義します。WebSphere® DataPower® XC10 アプライアンスは、SNMP バージョン v2c をサポートします。

始める前に

以下の手順を実行する前に、SNMP クライアントを構成します。SNMP クライアントを構成せずに以下の手順を実行できますが、有効な SNMP クライアントが構成されていない場合は、SNMP トラップを受信できません。

このタスクについて

以下の手順を使用して、通知されるトラップを具体的に設定します。

手順

1. 「SNMP 設定」パネルにナビゲートします。WebSphere DataPower XC10 アプライアンス ユーザー・インターフェイス上部のメニュー・バーで、「アプライアンス」>「**SNMP 設定**」>「トラップ・サブスクリプション」にナビゲートします。
2. 個別のトラップを選択またはクリアします。個々のトラップを選択およびクリアすることで、SNMP トラップを選択できます。SNMP トラップ・サブスクリイバーはすべて、データ・グリッドのストレージ制限とアプライアンスのストレージ制限に関するトラップを自動受信します。設定可能な具体的なトラップについては、[SNMP トラップ・リファレンス](#)を参照してください。

タスクの結果

上記手順を終えると、サブスクライブした SNMP クライアントに報告される SNMP トラップの集合が定義されます。

[SNMP トラップ・リファレンス](#)

以下のイベントは、IBM WebSphere DataPower XC10 アプライアンスの Simple Network Management Protocol (SNMP) エージェントによってトラップされ、サブスクライブした SNMP クライアントに伝達されます。

親トピック: [Simple Network Monitoring Protocol \(SNMP\) によるモニター](#)

前のトピック: [SNMP トラップ・サブスクリイバーの構成](#)

SNMP トラップ・リファレンス

以下のイベントは、IBM® WebSphere® DataPower® XC10 アプライアンスの Simple Network Management Protocol (SNMP) エージェントによってトラップされ、サブスクライブした SNMP クライアントに伝達されます。

IBM WebSphere DataPower XC10 アプライアンスのモニターに使用可能な SNMP トラップ

表 1. IBM WebSphere DataPower XC10 アプライアンス の SNMP トラップ。

次のトラップはデフォルトで構成されています。これらのトラップは使用不可にできません。

説明
この通知は、特定の XC10 データ・グリッドがその容量の 60 パーセントに到達したことを示しています。
この通知は、指定された XC10 データ・グリッドがその容量の 60 パーセントを下回る状態に戻ったことを示しています。
この通知は、特定の XC10 データ・グリッドがその容量の 80 パーセントに到達したことを示しています。
この通知は、指定された XC10 データ・グリッドがその容量の 80 パーセントを下回る状態に戻ったことを示しています。
この通知は、特定の XC10 データ・グリッドがその容量に到達したことを示しています。
この通知は、指定された XC10 データ・グリッドがその容量を下回る状態に戻ったことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 80 パーセントに到達したことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 80 パーセントを下回る状態に戻ったことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 90 パーセントに到達したことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量の 90 パーセントを下回る状態に戻ったことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量に到達したことを示しています。
この通知は、XC10 アプライアンス・ストレージがその容量を下回る状態に戻ったことを示しています。

表 2. アプライアンスの SNMP トラップ。

以下のトラップは、デフォルトで使用可能になっています。ユーザー・インターフェースで、これらのトラップを使用不可にすることができます。

デフォルト	説明
Y	イーサネット・インターフェースは、非アクティブ状態に遷移しています。
Y	イーサネット・インターフェースは、アクティブ状態に遷移しました。
Y	SNMP エージェントが、潜在的な構成変更で再初期設定されました。
Y	SNMP エージェントがシャットダウンしています。
Y	ファン・センサーが元の正常な状態に遷移しました。
Y	ファン・センサーが望ましくない状態に遷移しました。
Y	電圧センサーが元の正常な状態に遷移しました。
Y	電圧センサーが望ましくない状態に遷移しました。
Y	温度センサーが元の正常な範囲に遷移しました。
Y	温度センサーが望ましくない範囲に遷移しました。
Y	電源機構センサーが正常な状態に遷移しました。
Y	電源機構センサーが望ましくない状態に遷移しました。

親トピック: [SNMP トラップ・サブスクリプションの管理](#)

リモート・ロギングの構成

リモート・ロギングを使用可能にすれば、ログ・エントリをアプライアンス外のリモート・サーバーに保存することができます。リモート・ロギングは、問題の分離や長期にわたる動作のモニターを容易にするために詳細デバッグ・ログ・レベルを設定する必要があるときに役立つことがあります。

始める前に

- イベントを listen してキャプチャーするためには、syslog サーバーが使用可能でなければなりません。
- カタログ・サーバー、コンテナ・サーバー、およびアプリケーション・サーバー (WebSphere® Application Server を使用している場合) の名前は、英数字のみからなる名前ではなければなりません。Syslog RFC 1364 では、TAG フィールドに非英数字を使用することは許可されていません。TAG フィールドには syslog メッセージ内のサーバー名が含まれています。

このタスクについて

履歴データの分析のためにリモート・ロギングを使用します。アプライアンスは、システム内で、限られた数のログ・ファイルしか保持しません。さらなる分析に備えてより多くのログ・ファイルを保存する必要がある場合は、リモート・ロギングを構成してください。リモート・ロギング・サーバーは複数のアプライアンスからのデータを集約するので、ファイルが同じリモート・ロギング・サーバーに送られるように集合全体を構成することができます。

手順

アプライアンスのユーザー・インターフェースでリモート・ロギング・サーバーを構成します。

- a. ユーザー・インターフェースで、「アプライアンス」 > 「トラブルシューティング」 > 「ロギング」 > 「リモート・ロギングの構成」をクリックします。
- b. リモート・ロギング・サーバーのリモート・ホストおよびポートを入力します。
- c. リモート・ロギング・サーバーに送るメッセージの重大度のしきい値を入力します。警告メッセージと重大メッセージの両方を送る場合は、**WARNING** を選択します。重大メッセージのみを送る場合は、**SEVERE** を選択します。

タスクの結果

保存と分析のために構成したリモート・ロギング・サーバーにメッセージが送られます。

親トピック: [モニター](#)

クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成

アプリケーション用に構成しているデータ・グリッドでセキュリティーを使用する場合、client.properties ファイルを構成して、そのデータ・グリッド・アプリケーションに受け渡す設定を含むようにします。

始める前に

WebSphere® DataPower® XC10 アプライアンス を使用するデータ・グリッド・アプリケーションが必要です。

手順

1. アプリケーションが使用しているデータ・グリッドを保護します。詳しくは、[データ・グリッドのセキュリティーを使用可能に設定する](#)を参照してください。
2. ユーザーの資格情報を提供するようにデータ・グリッド・アプリケーションを構成します。ユーザーの資格情報を構成するには、client.properties ファイルを使用する必要があります。WebSphere eXtreme Scale クライアント インストール済み環境の [wxs_home/properties](#) ディレクトリーにある sampleClient.properties ファイルを使用して、プロパティー・ファイルを作成することができます。client.properties ファイルに次の変更を行います。
 - **securityEnabled: securityEnabled** プロパティーをデフォルト値の true に設定して、認証を含むクライアント・セキュリティーを使用可能にします。
 - **credentialAuthentication: credentialAuthentication** プロパティーをデフォルト値の Supported に設定します。これは、クライアントが資格情報の認証をサポートすることを示します。
 - **transportType: transportType** プロパティーを TCP/IP に設定します。これは、Secure Sockets Layer (SSL) を使用しないことを意味します。
 - **singleSignOnEnabled: singleSignOnEnabled** プロパティーをデフォルト値の false に設定します。シングル・サインオンは使用不可になります。
 - データ・グリッドへのアクセス権限を付与されているユーザーの資格情報:

```
credentialGeneratorClass=com.ibm.websphere.objectgrid.security.plugins.builtins
.UserPasswordCredentialGenerator
credentialGeneratorProps=<username> <password>
```

例:

```
credentialGeneratorClass=com.ibm.websphere.objectgrid.security.plugins.builtins
.UserPasswordCredentialGenerator
credentialGeneratorProps=xcadmin xcadmin
```

- エンコードされたパスワードを使用する場合は、**FilePasswordEncoder** ユーティリティーを使用できます。このユーティリティーは、[wxs_home/ObjectGrid/bin](#) ディレクトリーにあります。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)

[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[データ・グリッド・アプリケーション用の TLS の構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

データ・グリッド・アプリケーション用の TLS の構成

ご使用の構成の鍵ストアおよびトラストストアを変更または置換し、証明書別名を選択することによって、Transport Layer Security (TLS) を構成できます。

始める前に

- WebSphere® DataPower® XC10 アプライアンス を使用するデータ・グリッド・アプリケーションが必要で
す。
- アプライアンス管理権限が割り当てられている必要があります。
- アプライアンス構成に追加する、鍵ストアまたはトラストストアと、関連付けられたパスワードが必要です。既存
のトラストストアを変更したければ、アプライアンスからトラストストアをダウンロードすることができます。
クライアントの公開証明書を使用してトラストストアを更新する必要があります。詳しくは、[WebSphere
Application Server 用の Transport Layer Security \(TLS\) の構成](#)を参照してください。
- **keytool** ツールに対するアクセス権限が必要です。このツールは、[java_home/bin](#) ディレクトリーにありま
す。

このタスクについて

アプライアンスは、データ・グリッドに接続するクライアントを信頼する必要があります。TLS 設定は、ユーザー・イ
ンターフェースおよびデータ・グリッドに適用されます。設定は、集合内のすべてのアプライアンスに適用されます。

手順

1. アクティブなトラストストアをダウンロードします。ユーザー・インターフェースで、「集合」 > 「設定」 >
「**Transport Layer Security (TLS)**」をクリックします。「アクティブなトラストストアのダウンロード」を
クリックし、ディスク上のファイルを保存した場所 (例えば/downloads/trustStore.jks ディレクトリーな
ど) を覚えておきます。
2. 必要に応じて、証明書を作成し、公開証明書をエクスポートします。
 - a. 鍵ストアに秘密鍵を作成します。次のコマンドは、key.jks 鍵ストアと、そこに保管される鍵
「ogsample」を作成します。この key.jks 鍵ストアが、SSL 鍵ストアとして使用されます。以下のコマ
ンドを実行します。

```
keytool -genkey -alias ogsample -keystore key.jks -storetype JKS -keyalg rsa  
-dname "CN=ogsample, U=Your Organizational Unit, O=Your Organization, L=Your  
City, S=Your State, C=Your Country" storepass ogpass -keypass ogpass -validity  
3650
```

- b. 公開証明書をエクスポートします。次のコマンドは、「ogsample」という鍵の公開証明書を抽出して、そ
の鍵を temp.key ファイルに格納します。

```
keytool -export -alias ogsample -keystore key.jks -file temp.key -storepass  
ogpass
```

3. クライアントの証明書をトラストストアに追加します。 **keytool** ツールを実行して、クライアントの公開証明書
をトラストストアにインポートします。

```
keytool -import -noprompt -alias "ogsample" -keystore /downloads/trustStore.jks  
-file temp.key -storepass xc10pass -storetype jks
```

4. トラストストアの情報をアプライアンスにアップロードします。ユーザー・インターフェースで、「アプライア
ンス」 > 「設定」 > 「**Transport Layer Security (TLS)**」をクリックします。更新した
/downloads/trustStore.jks ファイルをアップロードします。「**TLS 設定の送信**」をクリックして、構成を
保存します。
5. client.properties ファイルを更新します。このファイルの場所およびこのファイル内のプロパティーについ
て詳しくは、[クライアント・プロパティー・ファイル](#)を参照してください。client.properties ファイル内の
以下のプロパティーを設定します。

```
securityEnabled=true  
transportType=SSL-Required  
alias=ogsample  
contextProvider=IBMJSSE2  
protocol=TLS  
keyStoreType=JKS  
keyStore=key.jks  
keyStorePassword=ogpass
```



```
trustStoreType=JKS
trustStore=/downloads/trustStore.jks
trustStorePassword=xc10pass
```

次のタスク

- 新しい構成を使用してアプリケーションを再始動し、Transport Layer Security に接続します。

親トピック: [セキュリティー](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)
[ユーザーおよびグループの管理](#)

[データ・グリッドのセキュリティーを使用可能に設定する](#)

[Java WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)

[Java クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)

[LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)

関連資料:

[REST ゲートウェイ: セキュリティー構成](#)

トラブルシューティング

ご使用の環境で遭遇した問題のトラブルシューティングには、ログならびにトレース、監査データ、ハードウェア・データ、およびリリース情報が役立ちます。

[WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

ご使用の IBM 製品の問題を切り分け、解決するために、トラブルシューティング情報とサポート情報を利用できます。この情報には、ご使用の IBM 製品 (WebSphere® DataPower® XC10 アプライアンス を含む) と共に提供される問題判別リソースを使用するための説明が示されています。

[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)

WebSphere DataPower XC10 アプライアンス と関連付けられているログ・ファイルはアプライアンスに保管されます。表示可能なログは、ユーザー・インターフェースを使用してアプライアンスから直接表示するか、または検討するために ローカル・ファイル・システムへダウンロードすることができます。

[監査データのダウンロード](#)

監査アクティビティは WebSphere DataPower XC10 アプライアンス によって取り込まれます。適切な範囲の監査が有効になるように、一連の Auditable オブジェクトに対する ユーザー・アクティビティが保存されます。

[ログおよびトレース・データの分析](#)

ログ分析ツールを使用して、ランタイム環境のパフォーマンスを分析したり、環境内で発生した問題を解決したりできます。

[ハードウェア温度のモニター](#)

温度センサーは、アプライアンスの各種内部構成装置の温度を常に測定しています。これらの温度は、ユーザー・インターフェースを使用すると、単一のパネルで容易にモニターできます。

[シリアル接続を使用したイーサネット・インターフェースの状況詳細のモニター](#)

IBM® WebSphere DataPower XC10 アプライアンス でネットワーク問題が発生した場合、イーサネット・インターフェースのアクティビティおよび状況の追加詳細情報を表示できます。このトピックでは、アプライアンスとのシリアル接続を使用したこれらの詳細の表示方法を説明しますが、ユーザー・インターフェースでこれらの詳細を表示することもできます。

[アプライアンスからのアウトバウンド接続の検査](#)

アウトバウンド接続機能を使用すると、アプライアンスからネットワーク・アドレスに 到達可能かどうかを確認できます。

[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)

コマンドを実行して、アプライアンス・ハードウェアを再始動したり、アプライアンスを元の工場出荷時設定に再設定したり、アプライアンスをシャットダウンしたりすることができます。

[デッドロックのトラブルシューティング](#)

以下のセクションでは、いくつかの最も一般的なデッドロック・シナリオを説明し、その回避方法を提示します。

2.5+ [マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

説明するシナリオは、ロック・タイムアウト例外を引き起こしているマルチ区画トランザクションの例です。トランザクションの状態に応じて、解決策で、この問題を手動で解決する方法を説明します。

[キャッシュ統合のトラブルシューティング](#)

HTTP セッションや動的キャッシュ構成など、キャッシュ統合構成の問題をトラブルシューティングする場合、この情報を使用してください。

[インストールのトラブルシューティング](#)

インストールおよび更新の問題をトラブルシューティングする場合、この情報を使用してください。

[管理のトラブルシューティング](#)

サーバーの開始や停止、**xscmd** ユーティリティの使用など、管理についてのトラブルシューティングを行う場合、この情報を使用してください。

[リリース情報](#)

製品のサポート Web サイト、製品資料、および製品の最新の更新、制限、および既知の問題へのリンクが提供されています。

WebSphere DataPower XC10 アプライアンス のトラブルシューティング およびサポート

ご使用の IBM 製品の問題を切り分け、解決するために、トラブルシューティング情報とサポート情報を利用できます。この情報には、ご使用の IBM 製品 (WebSphere® DataPower® XC10 アプライアンス を含む) と共に提供される問題判別リソースを使用するための説明が示されています。

[問題のトラブルシューティングのための手法](#)

トラブルシューティングとは、体系的な方法で問題を解決することです。トラブルシューティングの目的は、想定どおりに機能しない理由を明らかにして、問題の解決方法を説明することです。ある種の一般的な手法が、トラブルシューティングのタスクに役立つこともあります。

[知識ベースの検索](#)

以下は英語のみの対応となります。多くの場合、IBM 知識ベースを検索することによって、問題の解決策を見つけることができます。使用可能なリソース、サポート・ツール、および検索方式を使用して、最適の結果を得ることができます。

[フィックスの入手](#)

以下は英語のみの対応となります。お客様の問題の解決に、プロダクトのフィックスが有効な場合があります。

[IBM サポートへの連絡](#)

IBM サポートでは、FAQ での回答や、製品に関する問題解決でユーザーのお手伝いをするにより、製品の障害に関する支援を提供します。

[IBM サポート用の診断情報の収集](#)

コマンド行インターフェースを使用して、IBM サポートに送信するログを収集できます。

[IBM との情報の交換](#)

以下は英語のみの対応となります。問題の診断や特定を行うには、ご使用のシステムからのデータと情報を IBM サポートに提供していただく必要がある場合があります。また、IBM サポートから、問題判別に使用するツールまたはユーティリティーをご提供する場合があります。

[サポート更新情報のサブスクライブ](#)

ご使用の IBM 製品に関する重要な情報を常に入手するために、更新情報をサブスクライブすることができます。

親トピック: [トラブルシューティング](#)

問題のトラブルシューティングのための手法

トラブルシューティングとは、体系的な方法で問題を解決することです。トラブルシューティングの目的は、想定どおりに機能しない理由を明らかにして、問題の解決方法を説明することです。ある種の一般的な手法が、トラブルシューティングのタスクに役立つこともあります。

トラブルシューティング・プロセスの最初のステップは、問題の内容を完全に記述することです。問題の記述によって、問題の原因を突き止めるためにどこから着手すべきかを、お客様自身と IBM 技術サポート担当者がわかるようになります。このステップでは、以下の基本的な項目について明確にする必要があります。

- 問題の症状はどのようなものか。
- 問題の発生場所はどこか。
- 問題はいつ発生したか。
- 問題が発生する条件は何か。
- 問題は再現可能か。

通常は、これらの質問に回答することで問題が適切に記述され、問題解決につながります。

問題の症状はどのようなものか。

問題の記述を始めるときの最も明確な質問は「問題は何か」ということです。この質問は単純なように思われますが、問題をより具体的に説明する、さまざまな観点からの質問に細分化することができます。細分化した質問には以下のようなものがあります。

- だれが、または何が問題を報告するか。
- エラー・コードおよびメッセージの内容は何か。
- システムにどのような障害が発生したか。ループ、ハング、異常終了、パフォーマンス低下、不適切な結果などがあります。

問題の発生場所はどこか。

問題の発生源を特定することは、必ずしも簡単ではありませんが、これは問題を解決する上で最も重要な段階の 1 つです。報告元のコンポーネントと障害が発生しているコンポーネントの間に、いくつかのテクノロジー層が存在している場合があります。ネットワーク、データ・グリッド、およびサーバーは、問題を調査するときに考慮すべきコンポーネントの一部にすぎません。

次のような質問は、問題の層の切り分けのため、問題の発生箇所に焦点が集まるようにします。

- 問題は 1 つのプラットフォームまたはオペレーティング・システムに固有か、それとも複数のプラットフォームまたはオペレーティング・システムに共通か。
- 現在の環境と構成はサポートされているか。
- その問題はどのユーザーに対しても起こるのか。
- (マルチサイトのインストール済み環境の場合。) その問題はすべてのサイトで起こるのか。

ある層で問題が報告されても、必ずしもその層が問題の発生源とは限りません。問題の発生源の特定作業には、問題が存在する環境を理解することが含まれます。しばらく時間をかけて、問題のある環境のすべての内容 (オペレーティング・システムとバージョン、対応するすべてのソフトウェアとバージョン、ハードウェア情報など) を記述してください。サポートされる構成の環境で実行していることを確認してください。多くの場合、問題をトレースすると、ソフトウェア・レベルに互換性がないことがわかります (一緒に実行することを意図していないソフトウェア・レベルであったり、完全には一緒にテストされていないソフトウェア・レベルであったりします)。

問題はいつ発生したか。

障害発生に至るイベントについて、特に発生が 1 回限りのケースについて、詳しい時系列対照表を作成してください。最も簡単に時系列対照表を作成する方法は、逆方向に作業することです。エラーが報告された時点 (できればミリ秒単位に至るまで精密に) から開始して、使用可能なログと情報を通じて逆方向に作業します。一般に、診断ログ内で最初に見つかる疑わしいイベントまで調べる必要があります。

イベントの詳細な時系列対照表を作成するには、以下の質問に答えます。

- 問題が昼間または夜間の特定の時刻のみに発生するか。
- どれくらいの頻度で問題が発生するか。
- どのような一連のイベントを経て、問題が報告された時点に至ったのか。
- 環境の変更 (ソフトウェアまたはハードウェアのアップグレードやインストールなど) の後に問題が発生したか。

このようなタイプの質問に回答することで、問題調査のための基準の枠組みが得られます。

問題が発生する条件は何か。

問題の発生時に稼働していたシステムとアプリケーションを知ることは、トラブルシューティングの重要な部分です。ご使用の環境に関する次のような質問は、問題の根本原因の特定に役立ちます。

- 問題は同じタスクの実行時に常に発生するか。
- この問題が発生するときに、必ず発生する一連のイベントがあるか。
- 他のいずれかのアプリケーションでも同時に障害が発生したか。

このようなタイプの質問に回答することによって、問題が発生した環境が明らかになり、依存関係の相関付けができません。同じ時間の頃に複数の問題が発生したからといって、それらの問題が必ずしも関連しているとは限らないことを覚えておいてください。

問題は再現可能か。

トラブルシューティングの観点からすると、理想的な問題とは、再現できる問題であるということです。通常、問題を再現できる場合は、調査に役立つために自由に使用できるツールまたは手順の数が多くなります。このため、再現できる問題は、多くの場合デバッグや解決が簡単です。

しかし、再現できる問題には欠点があります。すなわち、その問題がビジネスに大きな影響を与える場合、問題が再発生することは好ましくありません。可能であれば、テスト環境または開発環境で問題を再現してください。一般に、このようにすると、調査時の柔軟性と管理能力が向上します。

- 問題をテスト・システムで再現することができるか。
- 複数のユーザーまたはアプリケーションで同じタイプの問題が発生しているか。
- 単一のコマンド、一連のコマンド、特定のアプリケーションのいずれを実行することによって問題を再現できるか。

親トピック: [WebSphere DataPower XC10 アプライアンスのトラブルシューティングおよびサポート](#)

知識ベースの検索

以下は英語のみの対応となります。多くの場合、IBM 知識ベースを検索することによって、問題の解決策を見つけることができます。使用可能なリソース、サポート・ツール、および検索方式を使用して、最適の結果を得ることができます。

このタスクについて

WebSphere® DataPower® XC10 アプライアンス のインフォメーション・センターを検索すれば、役立つ情報を見つけることができます。ただし、疑問の回答を得たり問題を解決するために、インフォメーション・センター以外で情報を検索する必要があることもあります。

手順

必要な情報を知識ベースで検索するには、以下のいずれか 1 つまたは複数の方法を使用します。

- IBM® Support Assistant (ISA) を使用してコンテンツを検索します。

ISA は、IBM ソフトウェア製品に関する質問への回答や問題の解決に役立つ、無料のソフトウェア保守ワークベンチです。ISA のダウンロードとインストールの手順については、[ISA Web サイト](#)を参照してください。

- [IBM Support Portal](#) を使用して必要なコンテンツを検索します。

IBM Support Portal は、IBM のすべてのシステム、ソフトウェア、およびサービスの、すべての技術サポート・ツールと情報が 1 つにまとめられた、中心となる場所です。この IBM Support Portal という 1 つの場所から、IBM エレクトロニック支援ポートフォリオにアクセスできます。各ページを調整して、問題の予防と迅速な問題解決に必要な情報やリソースに焦点を当てることができます。IBM Support Portal をよく理解するためには、このツールに関する[デモ・ビデオ](https://www.ibm.com/blogs/SPNA/entry/the_ibm_support_portal_videos) (https://www.ibm.com/blogs/SPNA/entry/the_ibm_support_portal_videos) をご覧ください。このビデオは、IBM Support Portal について紹介し、トラブルシューティングやその他のリソースについて説明し、ポートレットの移動、追加、削除によってページを調整する方法のデモを示します。

- WebSphere DataPower XC10 アプライアンス に関するコンテンツを検索するには、以下の追加の技術資料のいずれかを使用します。
 - [WebSphere DataPower XC10 アプライアンス リリース情報](#)
 - [WebSphere DataPower XC10 アプライアンス サポート Web サイト](#)
 - [WebSphere DataPower XC10 アプライアンスフォーラム \(forum\)](#)
- IBM マストヘッド検索を使用してコンテンツを検索します。IBM マストヘッド検索を使用するには、ibm.com® の任意のページの最上部にある「検索」フィールドに検索文字列を入力します。
- 外部の検索エンジン (Google、Yahoo、Bing など) を使用して、コンテンツを検索します。外部検索エンジンを使用すると、ibm.com ドメイン以外の情報が結果に含まれる可能性が高くなります。ただし、ibm.com 以外の場所にあるニュースグループ、フォーラム、およびブログで、IBM 製品に関する問題解決のための有用な情報が見つかることもあります。

ヒント: IBM 製品に関する情報を検索する場合は、「IBM」と製品の名称を検索に含めてください。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

フィックスの入手

以下は英語のみの対応となります。 お客様の問題の解決に、プロダクトのフィックスが有効な場合があります。

手順

フィックスを見つけてインストールするには、以下のようにします。

1. フィックスを入手するために必要なツールを取得します。 [IBM Update Installer](#) を使用して、WebSphere eXtreme Scale または WebSphere eXtreme Scale クライアントのさまざまなタイプの保守パッケージをインストールして、適用します。Update Installer は定期的に保守されるため、そのツールの最新バージョンを使用する必要があります。
2. 必要なフィックスを特定します。 [WebSphere DataPower® XC10 アプライアンスの推奨フィックス](#) を参照して、最新のフィックスを選択してください。フィックスを選択すると、そのフィックスのダウンロード資料が開きます。
3. フィックスをダウンロードします。ダウンロード資料で、「Download package」セクションの最新フィックスのリンクをクリックします。
4. フィックスを適用します。ダウンロード資料の「Installation Instructions」セクションに記載されている説明に従ってください。
5. フィックスやその他の IBM サポート情報について週次の E メール通知を受信できるよう、登録してください。

[Fix Central](#) からのフィックスの入手

以下は英語のみの対応となります。Fix Central を使用して、IBM サポートが推奨する、WebSphere DataPower XC10 アプライアンス を含むさまざまな製品のフィックスを見つけることができます。Fix Central では、ご使用のシステム用のフィックスを検索、選択、注文、およびダウンロードすることができ、その際に配信オプションを選択できます。以下は英語のみの対応となります。 お客様の問題の解決に、WebSphere DataPower XC10 アプライアンス プロダクトのフィックスが有効な場合があります。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

Fix Central からのフィックスの入手

以下は英語のみの対応となります。Fix Central を使用して、IBM サポートが推奨する、WebSphere® DataPower® XC10 アプライアンス を含むさまざまな製品のフィックスを見つけることができます。Fix Central では、ご使用のシステム用のフィックスを検索、選択、注文、およびダウンロードすることができ、その際に配信オプションを選択できます。以下は英語のみの対応となります。 お客様の問題の解決に、WebSphere DataPower XC10 アプライアンス プロダクトのフィックスが有効な場合があります。

手順

フィックスを見つけてインストールするには、以下のようにします。

1. フィックスを入手するために必要なツールを取得します。製品アップデート・インストーラーがインストールされていない場合は、それを取得します。インストーラーは [Fix Central](#) からダウンロードできます。このサイトでは、アップデート・インストーラーのダウンロード、インストール、および構成の手順を提供しています。
2. 製品として選択し、解決したい問題に関連のある 1 つ以上のチェック・ボックスを選択します。
3. 必要なフィックスを特定して選択します。
4. フィックスをダウンロードします。
 - a. ダウンロード資料を開き、「Download Package」セクションのリンクをたどります。
 - b. ファイルのダウンロード時に、保守ファイルの名前が変更されていないことを確認してください。このような変更は、意図的に行われる場合もあれば、特定の Web ブラウザーやダウンロード・ユーティリティーによって偶発的に行われる場合もあります。
5. フィックスを適用します。
 - a. ダウンロード資料の「Installation Instructions」セクションに記載されている説明に従ってください。
 - b. 詳しくは、製品資料の『Update Installer を使用したフィックスのインストール』のトピックを参照してください。
6. オプション: フィックスおよびその他の IBM サポートの更新については、登録して各週の E メール通知を受け取ってください。

親トピック: [フィックスの入手](#)

IBM サポートへの連絡

IBM サポートでは、FAQ での回答や、製品に関する問題解決でユーザーのお手伝いをするにより、製品の障害に関する支援を提供します。

始める前に

問題点に関する回答や解決方法について、リリース情報などのセルフ・ヘルプ・オプションを使用しても判明しない場合は、IBM サポートにご連絡ください。IBM サポートに連絡するには、お客様の会社または組織が有効な IBM 保守契約を締結しており、お客様が IBM に問題を送信する許可を受けている必要があります。使用可能なサポートのタイプについては、「*Software Support Handbook*」の『[Support portfolio](#)』のトピックを参照してください。

手順

問題について IBM サポートに連絡を取るには、次のようにしてください。

1. 問題を定義し、バックグラウンド情報を収集して、問題の重大度を判別します。詳しくは、「*Software Support Handbook*」の『[Getting IBM support](#)』のトピックを参照してください。
2. 診断情報を収集します。
3. IBM サポートに以下のいずれかの方法で問題を送信します。
 - IBM® Support Assistant (ISA) を使用する。を参照してください。
 - [IBM サポート・ポータル](#)によるオンライン・サポート: 「Service Request」ページの「Service Request」ポートレットから、お客様のすべてのサービス要求をオープン、更新、および表示できます。
 - 電話: 自国での連絡先の電話番号を調べるには、[Directory of worldwide contacts](#) の Web ページを参照してください。

タスクの結果

送信した問題がソフトウェア障害に関するものである場合、または文書の不備や不正確さに関するものである場合、IBM サポートはプログラム診断依頼書 (APAR) を作成します。APAR には問題が詳細に記載されます。IBM サポートでは、APAR が解決されてフィックスが配信されるまで、お客様が実施可能な回避策を可能な限り提供します。IBM では、解決された APAR を IBM サポート Web サイトに毎日公開しているため、同じ問題が発生した他のユーザーも同じ解決策を利用できます。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

IBM サポート用の診断情報の収集

コマンド行インターフェースを使用して、IBM サポートに送信するログを収集できます。

始める前に

コマンド行インターフェースの接続が必要です。詳しくは、[コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)を参照してください。

手順

- アプライアンス構成と状況情報のみを収集します。コマンド行インターフェースで、次のコマンドを実行します。

```
platform collect-pd <PDFFileName>
```

ファイル名を指定しない場合は、デフォルトで collect-pd.txt ファイルが作成されます。このファイルには、アプライアンスの状況コマンドからの出力とネットワーク構成の詳細が含まれます。

- アプライアンスのログ・ファイルとトレース・ファイルを収集します。コマンド行インターフェースで、次のコマンドを実行します。

```
platform must-gather <tarfilename> [<PDFFileName>]
```

このコマンドにより、アプライアンスのログ・ファイルとトレース・ファイルが含まれる tar ファイルが作成されます。コマンドを実行するには、tar ファイル名を指定する必要があります。このコマンドは、tar ファイルを作成する前に **platform collect-pd** コマンドを実行します。

- ログ・ファイルをクリアします。アプライアンスが長期間、実行を継続している場合は、ログ・ファイルが大きくなっている可能性があります。 **platform must-gather** コマンドからの tar ファイルが 1 ギガバイトを超える場合は、次のコマンドを実行して、ログ・ファイルをすべて、長さゼロにリセットすることができます。

```
clear-logs
```

- IBM サポートに送信するため、アプライアンスから生成済みファイルをコピーします。
 - 以下のコマンドを使用して生成したファイルをリストします。

```
Console> file list
```

```
logsTest.tgz 1810748334 bytes created 2011-06-08 08:22:17-0500
```

```
collect-pd.txt 17477 bytes created 2011-06-08 08:18:34-0500
```

- 以下のコマンドを使用して、アプライアンスのログ・ファイルをコピーします。

```
Console> file put logsTest.tgz
```

```
scp://root@linux010.myco.com:/opt/cp/logsTest.tgz
```

```
Password:*****
```

次のタスク

ファイルを IBM サポートに送信します。詳しくは、[IBM との情報の交換](#)を参照してください。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

IBM との情報の交換

以下は英語のみの対応となります。問題の診断や特定を行うには、ご使用のシステムからのデータと情報を IBM サポートに提供していただく必要がある場合があります。また、IBM サポートから、問題判別に使用するツールまたはユーティリティをご提供する場合があります。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

IBM サポートへの情報の送信

問題解決に必要な時間を減らすため、トレースおよび診断情報を IBM サポートに送信していただくことができます。

手順

IBM サポートに診断情報を提出するには、以下のようになります。

1. 問題管理レコード (PMR) を開きます。
2. 必要な診断データを収集します。診断データは、PMR の解決にかかる時間の節約に役立ちます。診断データは、手動で収集することも自動的に収集することもできます。
 - データを手動で収集する。
 - データを自動的に収集する。
3. .zip または .tar ファイル形式を使用してファイルを圧縮します。
4. ファイルを IBM に転送します。以下のいずれかの方法を使用して、IBM にファイルを転送することができます。
 - [IBM® Support Assistant](#)
 - [Service Request ツール](#)
 - 標準的なデータのアップロード方法: FTP、HTTP
 - 機密保護機能のあるデータのアップロード方法: FTPS、SFTP、HTTPS
 - E メール

z/OS 製品を使用していて、PMR の送信に ServiceLink / IBMLink を使用している場合は、E メールまたは FTP を使用して IBM サポートに診断データを送信できます。

これらのデータ交換方法は、すべて [IBM サポートの Web サイトで説明されています。](#)

IBM サポートからの情報の受信

IBM 技術サポートの担当者から、診断ツールやその他のファイルのダウンロードをお願いする場合があります。FTP を使用して、これらのファイルをダウンロードすることができます。

始める前に

IBM 技術サポートの担当者から、ファイルのダウンロードに使用する推奨サーバーと、アクセス先の正確なディレクトリおよびファイル名が指定されていることを確認してください。

手順

IBM サポートからファイルをダウンロードするには、以下のようになります。

1. FTP を使用して、IBM 技術サポートの担当者が指定したサイトに接続し、anonymous としてログインします。パスワードとして E メール・アドレスを使用してください。
2. 以下のように、適切なディレクトリに移動します。

- a. /fromibm ディレクトリに移動します。

```
cd fromibm
```

- b. IBM 技術サポートの担当者が指定したディレクトリに移動します。

```
cd nameofdirectory
```

3. セッションのバイナリー・モードを有効にします。

```
バイナリー
```

4. **get** コマンドを使用して、IBM 技術サポートの担当者が指定したファイルをダウンロードします。

```
get filename.extension
```

5. FTP セッションを終了します。

quit

サポート更新情報のサブスクライブ

ご使用の IBM 製品に関する重要な情報を常に入手するために、更新情報をサブスクライブすることができます。

このタスクについて

製品の更新情報を受け取るようにサブスクライブすることによって、特定の IBM サポート・ツールおよびリソースに関する重要な技術情報と更新情報を受け取ることができます。次の 2 つの方法のうちいずれかを使用して、更新情報をサブスクライブできます。

ソーシャル・メディアのサブスクリプション

製品に関し、次の RSS フィードが使用可能です。

- [WebSphere® DataPower® XC10 アプライアンスフォーラム](#)の RSS フィード

RSS の一般情報 (入門情報、RSS を使用できる IBM Web ページのリストなど) については、[IBM ソフトウェア・サポート RSS フィード](#)のサイトをご覧ください。

マイ通知 (My Notifications)

「マイ通知 (My Notifications)」では、任意の IBM 製品のサポート更新情報をサブスクライブできます。「マイ通知 (My Notifications)」は、以前に使用されていた同様のツール「マイ・サポート」に置き換わるものです。「マイ通知 (My Notifications)」を使用すると、E メールによる告知を、毎日、または週 1 回受信するように指定できます。発表、ヒント、製品フラッシュ (アラートとも言う)、ダウンロード、ドライバーなど、受信したい情報のタイプを指定できます。「マイ通知 (My Notifications)」では、通知を受信する製品、およびユーザーのニーズに最も適した配信方法をカスタマイズして分類することができます。

手順

サポート更新情報をサブスクライブするには、次のようにします。

1. [WebSphere DataPower XC10 アプライアンスフォーラム](#)の RSS フィードをサブスクライブします。
 - a. 「サブスクリプション」ページで、RSS フィード・アイコンをクリックします。
 - b. フィードのサブスクライブに使用するオプションを選択してください。
 - c. 「サブスクライブ」をクリックします。
2. 「マイ通知 (My Notifications)」をサブスクライブするには、[IBM® サポート・ポータル](#)にアクセスし、「通知」ポートレットの「**マイ通知 (My Notifications)**」をクリックします。
3. IBM ID とパスワードを使用してサインインし、「送信」をクリックします。
4. 更新を受け取る内容と方法を指定します。
 - a. 「サブスクライブ」タブをクリックします。
 - b. 該当するソフトウェア・ブランドまたはハードウェアのタイプを選択します。
 - c. 1 つ以上の製品名を選択して、「**続行**」をクリックします。
 - d. 更新を E メールで受け取るか、オンラインで指定のフォルダーに受け取るか、RSS または Atom フィードで受け取るか、更新の受信方法に関する設定を選択します。
 - e. 受信する資料更新のタイプ (製品ダウンロードに関する新規情報、ディスカッション・グループのコメントなど) を選択します。
 - f. 「**実行依頼**」をクリックします。

タスクの結果

RSS フィードと「マイ通知 (My Notifications)」の設定を変更するまで、要求した更新情報に関する通知を受け取ることになります。必要に応じて、この設定内容は変更することができます (ある製品の使用を中止し、別の製品の使用を開始した場合など)。

親トピック: [WebSphere DataPower XC10 アプライアンス のトラブルシューティングおよびサポート](#)

関連情報

🔗 [IBM ソフトウェア・サポート RSS フィード](#)

🔗 [「マイ通知 \(My Notifications\)」サポート・コンテンツの更新情報のサブスクライブ](#)

🔗 [IBM 技術サポートの「マイ通知 \(My Notifications\)」](#)

🔗 [IBM 技術サポート概要の「マイ通知 \(My Notifications\)」](#)

WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作

WebSphere® DataPower® XC10 アプライアンス と関連付けられているログ・ファイルはアプライアンスに保管されます。表示可能なログは、ユーザー・インターフェースを使用してアプライアンスから直接表示するか、または検討するために ローカル・ファイル・システムへダウンロードすることができます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

ログ・データは、アプライアンス上に直接保管されます。 trace.zip ファイルを IBM® のサポート・チームに提供できます。

手順

1. ユーザー・インターフェースで、「アプライアンス」「トラブルシューティング」をクリックし、「ロギング」を展開します。 ログ・データはアプライアンスに保管されます。「ロギング」セクションを展開すると、ログ・ビューアーを使用して使用可能なログにアクセスできるようになります。また、使用可能なログをファイル・システムにダウンロードして、さらに検討することもできます。
2. 現在のログ・ファイルを表示します。 次のいずれかのリンクをクリックしてログを表示します。
 - 現行エラー・ファイルの表示
 - 現行トレース・ファイルの表示
 - **2.5+** 現行の正常性ファイルの表示

ログ・ビューアー用に新しい Web ブラウザー・ウィンドウが開きます。ログ・ビューアーは、選択したログの末尾 10 行を表示するために使用されます。新しいログ項目が発生すると、ログ・ビューアーに付加されます。ログ・ビューアーには、ログ・ビューアーの動作を制御するいくつかのアクションがあります。

- a. 新しいログ項目の付加を停止するには、「一時停止」をクリックします。このアクションは、ログ・ビューアーが新しい項目を受け入れている場合にのみ使用できます。
 - b. 新しい項目を付加できるようにするには、「再始動」をクリックします。このアクションは、ログ・ビューアーが新しい項目を受け入れていない場合にのみ使用できます。
 - c. ログ・ビューアーからすべてのデータをクリアするには、「クリア」をクリックします。このアクションは、ログ・ビューアーが新しい項目を受け入れているかどうかにかかわらず使用できます。
3. 使用可能なすべてのログをファイル・システムに保存するには、「ログ・ファイルのダウンロード」をクリックします。発生したイベントに関する情報を表示したい場合は、このリンクを使用する必要があります。ウィンドウが表示され、圧縮ファイルを開いたり、ファイル・システムに保存したりすることができます。 trace.zip ファイルには、サポート・チームに提供するログ・ファイルがすべて含まれています。

trace.zip ファイルには一連の CSV ファイルが含まれていますが、これはアプライアンス上のサーバーの履歴データをトラッキングする際に使用できます。 trace.zip ファイル内で、CSV ファイルは `server_name/logs` ディレクトリにあります。ファイル名は `jvmstats.log`、`mapstats.log`、および `ogstats.log` です。マップまたはサーバーによるメモリー使用量、マップのヒット率、トランザクション時間、スループットなどの情報を CSV ファイルから収集することができます。

4. 「トレース・レベルの構成」セクションで、トレース・レベルの表示または変更を行うことができます。「管理コンソール」または「データ・グリッド」のトレース・レベルが編集できます。管理コンソールの場合、デフォルト・ロガーの出力を以下のトレース・レベルのいずれか 1 つに変更できます。
 - OFF
 - SEVERE
 - WARNING
 - INFO
 - FINE
 - a. トレース・ストリングを追加します。「トレース設定の追加」をクリックし、有効なトレース・ストリングを入力します。新規トレース・ストリングのトレース・レベルは、デフォルトで *INFO* に設定されます。
 - b. トレース・ストリングを削除します。トレース・ストリングを削除するには、そのトレース・ストリングの横にある「除去」アイコン (✖) をクリックします。
 - c. トレース・レベルを変更します。トレース・レベル値をクリックして、新しいトレース・レベルを選択します。指定したトレース・ストリングの新しいトレース・レベルをコミットするには、「保存」をクリックします。

次のタスク

Web コンソールが使用不可でも、trace.zip ファイルを取り出すことができます。コマンド行から以下のステップを実行して、IBM サポートに提供するログ・ファイルを取得してください。

1. ログ・ファイルを収集します。


```
Console> platform must-gather
```

2. 生成されたログ・ファイルをリストします。

```
Console> file list
```

3. アプライアンスからログをコピーします。

```
Console> file put <whateverNameIWant>.tgz  
scp://root@remoteMachine.myco.com:/myRemoteMachinePath/<whateverNameIWant>.tgz
```

[syslog を使用して WebSphere DataPower XC10 アプライアンス ログ・レコードをリモートの UNIX システムに送信する](#)

リモート・ロギングを使用可能に設定して、ログ・エントリをリモート・サーバーに送信することができます。リモート・ロギングは監査目的で情報を保存する場合に便利です。

親トピック: [トラブルシューティング](#)

関連タスク:

[CSV ファイルによるモニター](#)

関連資料:

[CSV ファイルの統計定義](#)

syslog を使用して WebSphere DataPower XC10 アプライアンス ログ・レコードをリモートの UNIX システムに送信する

リモート・ロギングを使用可能に設定して、ログ・エントリをリモート・サーバーに送信することができます。リモート・ロギングは監査目的で情報を保存する場合に便利です。

始める前に

- アプライアンス管理権限が必要です。
- syslog デーモンのセットアップ方法と syslog レコードの受信方法についての具体的な説明は、ご使用のオペレーティング・システムの資料を参照してください。
- WebSphere® DataPower® XC10 アプライアンス は syslog プロトコル RFC 3164 をサポートします。
- ログ・レコードはユーザー・データグラム・プロトコル (UDP) で送信され、したがって応答不要送信です。UDP プロトコルのとおり、メッセージ送達は保証されません。
- どのコンテナ・サーバー・プロセスがログに記録されて、WebSphere DataPower XC10 アプライアンス 上のリモート・システムに送信される対象になるか注意してください。

- **cs**
- **xsServer00-xsServerXY**: ここで **XY** は、アプライアンス上のコンテナ・プロセスの数を表します。
- **xsa.admin**

リモート・ロギングを使用可能に設定すると、すべてのプロセスが、syslogd への送信対象となります。リモート・ロギングが使用不可になっていると、どのプロセスも syslogd に送信されません。いずれかのプロセスが停止していたり、あるいは一時的に使用不可になっている場合、ロギングは、まだアクティブなプロセス上で続行されます。WebSphere DataPower XC10 アプライアンス 上の syslog に問題の可能性が疑われ、トラブルシューティングを行う必要がある場合は、アプライアンス・ユーザー・インターフェースでトレース・レベル **SYSLOG=all** を追加することができます。

このタスクについて

アーカイブ・ログ・イベントのアーカイブ用にリモート・ロギングを使用します。WebSphere DataPower XC10 アプライアンス では、スペースを節約するために、保存するログ・ファイルの数および保存期間を制限してログ・レコードを保存しています。ログ・レコードを長期間アーカイブしたり、ログ有効期限を過ぎたアーカイブ・ログ・レコードを分析したりするには、WebSphere DataPower XC10 アプライアンス でリモート・ロギングを構成し、使用可能に設定してください。

手順

1. アプライアンス・ユーザー・インターフェースで、「アプライアンス」 > 「トラブルシューティング」 > 「ロギング」をクリックします。
2. 「リモート・ロギングの構成」セクションで、「リモート・ロギングの有効化」を選択します。この設定により、履歴データ分析のためのリモート・ロギングが使用可能になります。イベントを listen してキャプチャーするためには、syslogd サーバーが使用可能でなければなりません。以下の設定を指定します。
 - a. リモート・ホスト - ログ・レコードの送信先である、リモート syslogd サーバーのホスト名または IP アドレスを指定します。値を「localhost」、「localhost-v6」、「127.0.0.1」、「::1」にすることはできません。アプライアンスから、そのホスト、または IP アドレスに ping できることを確認してください。
 - b. リモート・ポート - ログ・レコードの送信先の syslogd サーバーのポート番号を指定します。有効な値は、0-65535 で、デフォルト値は 512 です。
 - c. しきい値: リモート・ロギング・サーバーに送るメッセージの重大度のしきい値を指定します。警告メッセージと重大メッセージの両方を送る場合は、**WARNING** という値を入力してください。重大メッセージのみを送る場合は、**SEVERE** を選択します。
 - d. **syslog 機能**: ログ・メッセージの送信に使用される syslog ロギング機能を指定します。この設定により、リモート・システム上で実行されている syslog デーモンがログ・メッセージを入力するファイルが決定されます。例えば、この機能を「ユーザー」に設定すると、ほとんどの syslog デーモンはメッセージを /var/log/user.log ファイルに保管します。「メール」を選択すると、syslog デーモンはメッセージを /var/log/mail.log ファイルに保管します。ログ・レコードの宛先は、syslog デーモンをどのように構成したかによって決まります。syslogd 構成については、ご使用のオペレーティング・システムのための指示を参照してください。

設定を編集するには、「変更の適用」をクリックします。変更は syslog メッセージの影響を受けるすべてのプロセスに適用されます。

タスクの結果

保存と分析のために、アプライアンスのログ・レコードが、構成済みのリモート・ロギング・サーバーに送信されます。

親トピック: [WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)

監査データのダウンロード

監査アクティビティは WebSphere® DataPower® XC10 アプライアンス によって取り込まれます。適切な範囲の監査が有効になるように、一連の Auditable オブジェクトに対する ユーザー・アクティビティが保存されます。

このタスクについて

特定の Auditable オブジェクトについてのユーザー・アクティビティを収集した監査データが、アプライアンス上に保管されます。監査データに含まれる Auditable オブジェクトについては、下の表を参照してください。

表 1. Auditable オブジェクト

オブジェクト	追加/作成	削除	更新	その他
同化	Y	Y	N/A	
グループ	Y	Y	Y	
Session	N/A	N/A	N/A	タイムアウト、ログイン
ユーザー	Y	Y	Y	
ゾーン	Y	Y	Y	
ゾーン・アクティベーション	Y	Y	Y	

適切な監査手順のためには、データ安全性が非常に重要です。データ安全性を保証するため、監査データが必要になったときに、アプライアンスから監査データを取得する必要があります。監査データのダウンロード後は、ユーザーがそれぞれの方法でデータ安全性を維持する責任があります。以下の手順で、アプライアンスから監査データを取得してください。

手順

1. ユーザー・インターフェースの「監査」セクションを開きます。「アプライアンス」>「トラブルシューティング」をクリックします。「監査」セクションを展開します。「監査」セクションを展開すると、アプライアンスで使用可能な監査データにアクセスできます。
2. 監査データをダウンロードします。すべてのデータまたはフィルターに掛けられたデータをダウンロードできます。

- 「データをすべてダウンロード」

「データをすべてダウンロード」をクリックして、audit.zip ファイルをダウンロードします。audit.zip ファイルには、アプライアンスで現在使用可能なすべてのデータが含まれています。コンマ区切り値 (CSV) フォーマットの監査データが必要な場合、または使用可能な監査データのサブセットのみが必要な場合は、日付範囲を調整して、フィルター処理を行ったデータをダウンロードできます。

- 「フィルターに掛けられたデータのダウンロード」

日付範囲を調整し、時間帯を設定します。日付範囲および時間帯の設定は、フィルター処理されたデータの生成に使用されます。これらの設定は、データをフィルターに掛ける場合にのみ適用され、すべてのデータをダウンロードする際には適用されません。時間帯設定のデフォルト値は、アプライアンスに関連付けられた時間帯です。デフォルトの日付範囲には、最新の 30 日分のデータが含まれます。時間帯と日付範囲を調整することで、データに組み込む監査イベントを指定します。

「フィルターに掛けられたデータのダウンロード」をクリックして、指定した日付範囲の監査データを CSV フォーマットでダウンロードします。audit.csv ファイルはコンマ区切り値フォーマットのファイルであり、ファイル・システムにダウンロードできます。CSV フォーマットのデータをベンダー・ソフトウェアにインポートすれば、フォーマットや表示の設定をさらに操作できます。

親トピック: [トラブルシューティング](#)

ログおよびトレース・データの分析

ログ分析ツールを使用して、ランタイム環境のパフォーマンスを分析したり、環境内で発生した問題を解決したりできます。

このタスクについて

環境内の既存のログ・ファイルやトレース・ファイルからレポートを生成できます。これらのビジュアル・レポートには次のような用途があります。

- **ランタイム環境の状況およびパフォーマンスの分析**
 - デプロイメント環境の整合性
 - ログの頻度
 - 実行中トポロジーと構成されているトポロジーの比較
 - 予定外のトポロジー変更
 - クォーラム状況
 - 区画の複製の状況
 - メモリー、スループット、プロセッサ使用量などの統計
- **環境内の問題のトラブルシューティング**
 - 特定時点でのトポロジー・ビュー
 - クライアント障害時のメモリー、スループット、プロセッサ使用量などの統計
 - 現在のフィックスパック・レベル、チューニング設定
 - クォーラム状況

[ログ分析の概要](#)

環境内の問題のトラブルシューティングに役立つ **xsLogAnalyzer** ツールを使用できます。

[ログ・ファイルの保管ポリシー](#)

[ログ分析の実行](#)

任意のコンピューターのログ・ファイルやトレース・ファイルのセットに対して **xsLogAnalyzer** ツールを実行できます。

[ログ分析用カスタム・スキャナーの作成](#)

ログ分析用のカスタム・スキャナーを作成できます。スキャナーを構成してから、**xsLogAnalyzer** ツールを実行すると、結果がレポート内に生成されます。カスタム・スキャナーは、指定された正規表現に基づいてイベント・レコードのログをスキャンします。

[ログ分析のトラブルシューティング](#)

xsLogAnalyzer ツールおよびこのツールで生成されるレポートに関する問題を診断し、修正する場合、次のトラブルシューティング情報を使用してください。

親トピック: [トラブルシューティング](#)

ログ分析の概要

環境内の問題のトラブルシューティングに役立つ **xsLogAnalyzer** ツールを使用できます。

すべてのフェイルオーバー・メッセージ

フェイルオーバー・メッセージの総数を一定時間のチャートで表示します。また、影響を受けたサーバーを含む、フェイルオーバー・メッセージのリストも表示します。

すべての **eXtreme Scale** 重大メッセージ

メッセージ ID を関連する説明およびユーザー処置と一緒に表示します。これにより、メッセージを検索する時間を節約できます。

すべての例外

メッセージ、発生回数、例外の影響を受けたサーバーも含めて、上位 5 つの例外を表示します。

トポロジーの要約

ログ・ファイルに基づいて、どのようにトポロジーが構成されているかをダイアグラムで表示します。この要約を使用して実際の構成と比較することができ、構成エラーを特定できる場合があります。

トポロジーの整合性: オブジェクト・リクエスト・ブローカー (**ORB**) 比較表

環境での ORB 設定を表示します。環境全体で設定が整合しているか判断するのに助けるために、この表を使用できます。

イベント・タイムライン・ビュー

データ・グリッドで発生したライフサイクル・イベント、例外、重大なメッセージ、初期障害データ・キャプチャー機能 (FFDC) イベントなどのさまざまなアクションのタイムライン図を表示します。

親トピック: [ログおよびトレース・データの分析](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析用カスタム・スキャナーの作成](#)

[ログ分析のトラブルシューティング](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ログ・ファイルの保管ポリシー

下記の表は、WebSphere® DataPower® XC10 アプライアンス 上でのログ・ファイルの保管に関するポリシーについて説明したものです。

表 1. xsa.app トレース・ファイル

xsa.app トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	5	true
audit.log	30	2097152	5	true
*stats.log	N/A	N/A	N/A	N/A

表 2. xsa.admin トレース・ファイル

xsa.admin トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	2	true
audit.log	30	2097152	5	true
health.log	30	2097152	2	true
*stats.log	N/A	N/A	N/A	N/A

表 3. cs トレース・ファイル

cs トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	2	true
audit.log	30	2097152	5	true
health.log	30	2097152	2	true
*stats.log	N/A	N/A	N/A	N/A

表 4. xsServer* トレース・ファイル

xsServer* トレース・ファイル	削除前の最大最終変更継続期間 (日数)	最大サイズ (バイト)	ログ・ファイルの最大数	最大サイズへの圧縮
trace.log	30	20971520	10	true
error.log	30	2097152	2	true
audit.log	30	2097152	5	true
health.log	30	2097152	2	true
*stats.log	N/A	104857600	10	true

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析用カスタム・スキャナーの作成](#)

[ログ分析のトラブルシューティング](#)

ログ分析の実行

任意のコンピューターのログ・ファイルやトレース・ファイルのセットに対して **xsLogAnalyzer** ツールを実行できます。

始める前に

- WebSphere® eXtreme Scale クライアント インストール済み環境が必要です。詳しくは、[WebSphere eXtreme Scale クライアントのインストール](#)を参照してください。
- アプライアンスからログ・ファイルとトレース・ファイルを集めます。詳しくは、[WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)を参照してください。
- 生成されるレポートのカスタム・スキャナーを作成する場合は、ツールを実行する前に スキャナー仕様プロパティ・ファイルと構成ファイルを作成してください。詳しくは、[ログ分析用カスタム・スキャナーの作成](#)を参照してください。

手順

1. **xsLogAnalyzer** ツールを実行します。

スクリプトは次の場所にあります。

- スタンドアロン・インストールの場合: [wxs_install_root](#)/ObjectGrid/bin
- WebSphere Application Server と統合されたインストールの場合: [was_root](#)/bin

ヒント: ログ・ファイルが大きい場合、レポートを実行するときに **-startTime**、**-endTime**、および **-maxRecords** パラメーターを使用して、スキャンするログ・エントリーの数 を制限することを検討してください。レポートを実行するときにこれらのパラメーターを使用すると、レポートが見やすくなるうえ、レポートをより効率的に実行できます。同一セットのログ・ファイルを対象に複数のレポートを実行できます。

```
xsLogAnalyzer.sh|bat -logsRoot c:¥myxslogs -outDir c:¥myxslogs¥out  
-startTime 11.09.27_15.10.56.089 -endTime 11.09.27_16.10.56.089 -maxRecords 100
```

-logsRoot

評価するログ・ディレクトリーへの絶対パスを指定します (必須)。

-outDir

レポートの出力を書き込む既存のディレクトリーを指定します。値を指定しないと、レポートは **xsLogAnalyzer** ツールの ルート・ロケーションに書き込まれます。

-startTime

ログ内の評価する開始時刻を指定します。日付のフォーマットは、*year.month.day_hour.minute.second.millisecond*です。

-endTime

ログ内の評価する終了時刻を指定します。日付のフォーマットは、*year.month.day_hour.minute.second.millisecond*です。

-trace

トレース・ストリング (ObjectGrid*=all=enabled など) を指定します。

-maxRecords

レポート内に生成するレコードの最大数を指定します。デフォルトは 100 です。値を 50 と指定した場合、指定された期間の最初の 50 レコードが生成されます。

2. 生成されたファイルを開きます。出力ディレクトリーを定義しなかった場合、レポートは *report_date_time* というフォルダー内に生成されます。レポートのメインページを開くには、*index.html* ファイルを開きます。
3. レポートを使用して、ログ・データを分析します。次のヒントを使用して、レポート表示のパフォーマンスを最大にしてください。
 - ログ・データの照会のパフォーマンスを最大にするには、できるだけ具体的な情報を使用します。例えば、*server_host_name* の照会より *server* の照会のほうが実行時間が長くなり、返される結果も多くなります。
 - 一部のビューでは、一度に表示されるデータ・ポイントの数が制限されます。ビュー内の現在のデータを変更して (開始時刻や終了時刻を変更するなどして)、表示される時間セグメントを調整できます。

次のタスク

xsLogAnalyzer ツールや生成されるレポートのトラブルシューティングの詳細については、[ログ分析のトラブルシューティング](#)を参照してください。

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析用カスタム・スキャナーの作成](#)

[ログ分析のトラブルシューティング](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ログ分析用カスタム・スキャナーの作成

ログ分析用のカスタム・スキャナーを作成できます。スキャナーを構成してから、**xsLogAnalyzer** ツールを実行すると、結果がレポート内に生成されます。カスタム・スキャナーは、指定された正規表現に基づいてイベント・レコードのログをスキャンします。

手順

1. カスタム・スキャナーで実行する正規表現を指定したスキャナー仕様プロパティ・ファイルを作成します。
 - a. プロパティ・ファイルを作成し、保存します。ファイルは `loganalyzer_root/config/custom` ディレクトリ内に存在しなければなりません。ファイルには好きな名前を付けることができます。ファイルは新規スキャナーで使用されるので、スキャナーの名前をプロパティ・ファイルの名前の一部にすると (例えば、`my_new_server_scanner_spec.properties`) 便利です。
 - b. 次のプロパティを `my_new_server_scanner_spec.properties` ファイルに組み込みます。

```
include.regular_expression = REGULAR_EXPRESSION_TO_SCAN
```

`REGULAR_EXPRESSION_TO_SCAN` 変数は、ログ・ファイルのフィルタリングに使用する正規表現です。

例: `xception` と `rror` 両方のストリングを含んでいる行 (ストリングの出現順序は問いません) のインスタンスをスキャンするには、**include.regular_expression** プロパティを次の値に設定します。

```
include.regular_expression = (xception.+rror)|(rror.+xception)
```

この正規表現によって、`rror` ストリング または `xception` ストリングのいずれかが存在する場合、イベントが記録されます。

例:

ログ内の各行をスキャンして、句 `xception` または句 `rror` のいずれかのストリングを含んでいる行のインスタンスを探すには、**include.regular_expression** プロパティを次の値に設定します。

```
include.regular_expression = (xception)|(rror)
```

`rror` ストリングまたは `xception` ストリングのいずれかが存在する場合、この正規表現によってイベントが記録されます。

2. **xsLogAnalyzer** ツールがスキャナーを作成するために使用する構成ファイルを作成します。
 - a. 構成ファイルを作成し、保存します。ファイルは `loganalyzer_root/config/custom` ディレクトリ内に存在しなければなりません。ファイルの名前は、`scanner_nameScanner.config` のようにします。ここで、`scanner_name` は、新規スキャナーの固有の名前です。例えば、このファイルは `serverScanner.config` という名前にできます。
 - b. 次のプロパティを `scanner_nameScanner.config` ファイルに組み込みます。

```
scannerSpecificationFiles = LOCATION_OF_SCANNER_SPECIFICATION_FILE
```

`LOCATION_OF_SCANNER_SPECIFICATION_FILE` 変数は、前のステップで作成した仕様ファイルの場所 (パス) です。例: `loganalyzer_root/config/custom/my_new_scanner_spec.properties`。セミコロンで区切ったリストを使用して、複数のスキャナー仕様ファイルを指定することもできます。

```
scannerSpecificationFiles =  
LOCATION_OF_SCANNER_SPECIFICATION_FILE1;LOCATION_OF_SCANNER_SPECIFICATION_FILE2
```

3. **xsLogAnalyzer** ツールを実行します。詳しくは、[ログ分析の実行](#)を参照してください。

タスクの結果

xsLogAnalyzer ツールを実行すると、構成した カスタム・スキャナー用の新しいタブがレポートに含まれています。各タブには、次のビューがあります。

チャート

記録されたイベントを示すプロット・グラフ。イベントは検出された順序で表示されます。

テーブル

記録されたイベントのテーブル表示。

要約レポート

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析のトラブルシューティング](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ログ分析のトラブルシューティング

xsLogAnalyzer ツールおよびこのツールで生成されるレポートに関する問題を診断し、修正する場合、次のトラブルシューティング情報を使用してください。

手順

- **問題:** **xsLogAnalyzer** ツールを使用してレポートを生成中、メモリー不足状態が発生する。発生する可能性があるエラーの例は、次のとおりです: `java.lang.OutOfMemoryError: GC overhead limit exceeded`。

解決策: **xsLogAnalyzer** ツールは Java 仮想マシン (JVM) 内で実行されます。**xsLogAnalyzer** ツールの実行時に、ある設定を指定することで、ヒープ・サイズを大きくするよう JVM を構成してから、ツールを実行することができます。ヒープ・サイズを大きくすることで、より多くの イベント・レコードを JVM メモリー内に保管できるようになります。オペレーティング・システムに十分なメイン・メモリーがあれば、最初は 2048M を設定してはじめてください。**xsLogAnalyzer** ツールを実行するのと同じコマンド行インスタンスで、次のように最大 JVM ヒープ・サイズを設定します。

```
java -XmxHEAP_SIZEm
```

`HEAP_SIZE` 値は、任意の整数にでき、JVM ヒープに割り振られるメガバイト数を表します。

例えば、`java -Xmx2048m` を実行できます。メモリー不足メッセージが続く場合、または 2048m 以上のメモリーを割り振るためのリソースがない場合は、ヒープ内に保持するイベントの数を制限してください。

`-maxRecords` パラメーターを **xsLogAnalyzer** コマンドに渡すと、ヒープ内のイベントの数を制限できます。

- **問題:** **xsLogAnalyzer** ツールで生成されたレポートを開くと、ブラウザーがハングするか、ページが読み込まれない。

原因: 生成された HTML ファイルが大きすぎるため、ブラウザーが読み込むことができません。これらのファイルが大きすぎる理由は、分析対象のログ・ファイルの範囲が広すぎるためです。

解決策: **xsLogAnalyzer** ツールを実行するときに `-startTime`、`-endTime`、および `-maxRecords` パラメーターを使用して、スキャンするログ・エントリーの数を制限することを検討してください。レポートを実行するときにこれらのパラメーターを使用すると、レポートが見やすくなるうえ、レポートをより効率的に実行できます。同一セットのログ・ファイルを対象に複数のレポートを実行できます。

親トピック: [ログおよびトレース・データの分析](#)

関連概念:

[ログ分析の概要](#)

関連タスク:

[ログ分析の実行](#)

[ログ分析用カスタム・スキャナーの作成](#)

関連資料:

[ログ・ファイルの保管ポリシー](#)

ハードウェア温度のモニター

温度センサーは、アプライアンスの各種内部構成装置の温度を常に測定しています。これらの温度は、ユーザー・インターフェースを使用すると、単一のパネルで容易にモニターできます。

始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

IBM® WebSphere® DataPower® XC10 アプライアンス は、各種内部構成装置をモニターして、アプライアンスが安全な温度で作動していることを確認します。

手順

1. 「アプライアンス」 > 「トラブルシューティング」にナビゲートします。
2. 「ハードウェア温度」を展開します。
3. ハードウェア温度を確認します。

各温度センサーの最新の読み取り値が、3桁の有効数字を使用して表示されます。

- 構成装置の温度が標準的な作動温度である場合は、緑色のアイコン (✅) が表示されます。
- 構成装置の温度が安全な作動レベルを超えている場合は、黄色の警告アイコン (⚠️) が表示されます。

タスクの結果

上記の手順が正常に完了すると、有効な温度読み取り値が表示され、アプライアンスが安全な温度で作動していることを確認できます。

次のタスク

引き続き、発生している問題の[トラブルシューティング](#)を行います。

親トピック: [トラブルシューティング](#)

シリアル接続を使用したイーサネット・インターフェースの状況詳細のモニター

IBM® WebSphere® DataPower® XC10 アプライアンス でネットワーク問題が発生した場合、イーサネット・インターフェースのアクティビティおよび状況の追加詳細情報を表示できます。このトピックでは、アプライアンスとのシリアル接続を使用したこれらの詳細の表示方法を説明しますが、ユーザー・インターフェースでこれらの詳細を表示することもできます。

始める前に

アプライアンスに対する物理的なアクセス権限を保持しており、xcadmin としてログインできる必要があります。

手順

1. xcadmin ユーザーとして、アプライアンスとのシリアル接続を確立します。

ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC を、アプライアンスのシリアル・ポートにシリアル接続する必要があります。シリアル・コンソールとして PC を使用している場合は、Windows 用または Linux 用の PC ベースのシリアル通信プログラムを使用します。提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して、アプライアンスに接続します。

重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。

シリアル・コンソール設定は `9600.8.n.1` です。

2. イーサネット・インターフェースの状況を表示します。以下のコマンドを使用して、イーサネット・インターフェースの状況を表示します。

```
netif status <interface>
```

ここで、<interface> は、照会するイーサネット・インターフェースの名前です。出力例を以下に示します。

```
Console> netif status mgt0
mgt0    generic MTU:1500 flags:UP BROADCAST RUNNING MULTICAST
        inet addr:127.0.0.1 mask:255.255.255.128
        inet6 addr: 2001:DB8:0:0:0:0:0:0/32 mask: ffff:ffff:ffff:ffff::
        ethernet MAC: 00:1a:64:88:a0:6c autoneg:on duplex:Full port:TP
        speed:100Mbps
        statistics collisions:0 multicast:13 rx_bytes:101069093
          rx_compressed:0 rx_crc_errors:0 rx_dropped:0 rx_errors:0
          rx_fifo_errors:0 rx_frame_errors:0 rx_length_errors:0
          rx_missed_errors:0 rx_over_errors:0 rx_packets:1515625
          tx_aborted_errors:0 tx_bytes:10045272 tx_carrier_errors:0
          tx_compressed:0 tx_dropped:0 tx_errors:0 tx_fifo_errors:0
          tx_heartbeat_errors:0 tx_packets:20104 tx_window_errors:0
```

タスクの結果

これらのステップを完了すると、アプライアンス上のイーサネット・インターフェースに関する可用性状況の詳細の検討は完了です。

親トピック: [トラブルシューティング](#)

アプライアンスからのアウトバウンド接続の検査

アウトバウンド接続機能を使用すると、アプライアンスからネットワーク・アドレスに到達可能かどうかを確認できます。



始める前に

以下の手順を実行するには、アプライアンス管理権限が割り当てられている必要があります。

このタスクについて

アウトバウンド接続機能は、アプライアンスからターゲットのネットワーク・アドレスに到達可能であるかを確認することによって、問題を切り分けるためのトラブルシューティング作業を支援します。このツールは、あらゆる問題をデバッグする際に適したものではありませんが、ターゲット・ネットワーク・アドレスが使用可能であること、およびそのネットワーク・アドレスへの接続がファイアウォールまたはネットワーク問題によって妨害されていないことを確認するための便利な方法として機能します。

手順

1. 「アプライアンス」 > 「トラブルシューティング」にナビゲートします。
2. 「アウトバウンド接続」を展開します。
3. IP アドレスまたは完全修飾ホスト名を入力します。このフィールドに入力されたネットワーク・アドレスは、ping コマンドの発行時にターゲット・アドレスとして使用されます。
4. 指定したネットワーク・アドレスに対して ping コマンドを実行するには、「**ping**」をクリックします。
 - 入力したネットワーク・アドレスへの接続試行が成功すると、が表示されます。
 - 入力したネットワーク・アドレスへの接続試行が成功しなかった場合は、が表示されます。

タスクの結果

上記の手順が正常に完了すると、ターゲット・アドレスが現在、アプライアンスからネットワーク経由で到達可能かどうか判別されます。

次のタスク

引き続き、発生している問題の[トラブルシューティング](#)を行います。

親トピック: [トラブルシューティング](#)

コマンド行インターフェースを使用したアプライアンスでの操作の実行

コマンドを実行して、アプライアンス・ハードウェアを再始動したり、アプライアンスを元の工場出荷時設定に再設定したり、アプライアンスをシャットダウンしたりすることができます。

このタスクについて

コマンド行インターフェースを使用して、アプライアンス構成データを再設定したり、アプライアンスを再始動したり、アプライアンスをシャットダウンしたりすることができます。

手順

1. xcadmin ユーザーを使用して、アプライアンスとの接続を確立します。 シリアル接続を使用するか、セキュア・シェル (SSH) でコマンド行インターフェースに接続することができます。
 - **シリアル接続:** ASCII 端末、または端末エミュレーション・ソフトウェアを実行している PC を、アプライアンスのシリアル・ポートにシリアル接続する必要があります。 シリアル・コンソールとして PC を使用している場合は、Windows 用または Linux 用の PC ベースのシリアル通信プログラムを使用する必要があります。 提供されたシリアル・ケーブルまたは PL-2303 USB シリアル・ケーブルを使用して、アプライアンスに接続します。
重要: PL-2303 USB シリアル・ケーブルを使用する場合は、先に進む前に、ケーブル用のドライバーをダウンロードしてインストールする必要があります。
 - **リモート・ログイン:** SSH でアプライアンスに接続するには、アプライアンスの URL を SSH クライアントに指定します。
2. アプライアンスとの接続を確立したら、コマンドを実行することができます。

2.5+ 実行できるコマンドについては、[CLI コマンド・リファレンス \(CLI command reference\)](#) を参照してください。

次のタスク

アプライアンスを再設定した場合には、アプライアンスを再初期化できます。 アプライアンスの構成について詳しくは、『[IBM WebSphere DataPower XC10 アプライアンスの初期設定および構成](#)』を参照してください。

親トピック: [トラブルシューティング](#)

関連タスク:

[集約インターフェースの編集](#)

[集約インターフェースの削除](#)

[集約インターフェースの追加](#)

[IBM WebSphere DataPower XC10 アプライアンスのイーサネット・インターフェースの管理](#)

[ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)

[タスクによるアクティビティのモニター](#)

デッドロックのトラブルシューティング

以下のセクションでは、いくつかの最も一般的なデッドロック・シナリオを説明し、その回避方法を提示します。

始める前に

アプリケーションに例外処理を実装します。詳しくは、[ロック・シナリオでの例外処理の実装 \(Java アプリケーション\)](#) および [ロック・シナリオでの例外処理の実装 \(.NET アプリケーション\)](#) を参照してください。

結果、次の例外が表示されます。

```
com.ibm.websphere.objectgrid.plugins.LockDeadlockException: Message
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるストリングを表します。

手順

- **問題:** LockTimeoutException 例外が発生する。

説明: トランザクションまたはクライアントが特定のマップ・エントリーに対するロックの付与を求めると、その要求は、現在のクライアントがロックを解放するまで待機させられ、その後、要求が実行依頼されることがしばしばあります。ロック要求が長い期間アイドル状態のままになり、いつまでもロックが付与されない場合、デッドロックを回避するために LockTimeoutException 例外が作成されます。デッドロックについては、次のセクションで詳しく説明します。ペシミスティック・ロック・ストラテジーを構成すると、ロックはトランザクションがコミットするまで解放されないため、この例外が発生する可能性がより高くなります。

詳細の取得

- **Java** LockTimeoutException 例外は、ストリングを返す getLockRequestQueueDetails メソッドを含んでいます。このメソッドを使用して、例外のトリガーとなった状態についての詳細説明を確認できます。以下に、例外をキャッチして、エラー・メッセージを表示するサンプル・コードを示します。

```
try {
    ...
}
catch (LockTimeoutException lte) {
    System.out.println(lte.getLockRequestQueueDetails());
}
```

ObjectGridException 例外 catch ブロック内で例外を受け取る場合、次のコードのように例外を判定し、キューの詳細を表示できます。コードでは、findRootCause ユーティリティ・メソッドも使用します。

```
try {
    ...
}
catch (ObjectGridException oe) {
    Throwable Root = findRootCause( oe );
    if (Root instanceof LockTimeoutException) {
        LockTimeoutException lte = (LockTimeoutException)Root;
        System.out.println(lte.getLockRequestQueueDetails());
    }
}
```

- **.NET** LockTimeoutException 例外は、ストリングを返す getMessage メソッドを含んでいます。このメソッドを使用して、例外のトリガーとなった状態についての詳細説明を確認できます。

解決策: LockTimeoutException 例外を使用して、アプリケーションでデッドロックが発生する可能性を回避できます。このタイプの例外は、例外が一定時間待機すると発生します。ロック・タイムアウト値を設定することで、例外が待機する時間を設定できます。アプリケーションで実際にはデッドロックが発生していない場合は、ロック・タイムアウトを調整して、LockTimeoutException を回避してください。

ロック・タイムアウト値をプログラマチックに構成してください。

- **Java** [Java アプリケーションにおけるロックの構成と実装](#)
- **.NET** [.NET アプリケーションにおけるロックの構成と実装](#)

- **問題:** 単一キーでデッドロックが発生する。

説明: 以下のシナリオでは、S ロックを使用して単一キーにアクセスし、その後、そのキーを更新するときにデッドロックがどのように発生するかを示しています。このアクションが 2 つのトランザクションから同時に実行されたとき、デッドロックが発生します。

表 1. 単一キーのデッドロックのシナリオ

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	get key1	get key1	key1 に対して S ロックが両方のトランザクションに認可されます。
3	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> Put key1 		U ロックはありません。更新はトランザクション・キャッシュで実行されます。
4		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> put key1 	U ロックはありません。更新はトランザクション・キャッシュで実行されます。
5	トランザクションのコミット		ブロックされます。スレッド 2 が S ロックを保有しているため、key1 に対する S ロックは X ロックにアップグレードできません。
6		トランザクションのコミット	デッドロック: T1 が S ロックを保有しているため、key1 に対する S ロックは X ロックにアップグレードできません。

表 2. 単一キーのデッドロック (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	get key1		key1 に対して S ロックが認可されます。
3	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	get key1	key1 に対して S ロックが U ロックにアップグレードされます。
4		get key1	key1 に対して S ロックが認可されます。
5		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	ブロックされます。T1 が既に U ロックを保有しています。
6	トランザクションのコミット		デッドロック: key1 に対する U ロックはアップグレードできません。
7		トランザクションのコミット	デッドロック: key1 に対する S ロックはアップグレードできません。

表 3. 単一キーのデッドロック (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。

2	get key1		key1 に対して S ロックが認可されます。
3	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 		key1 に対して S ロックが U ロックにアップグレードされます。
4		get key1	key1 に対して S ロックが認可されます。
5		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	ブロックされます。スレッド 1 が既に U ロックを保有しています。
6	トランザクションのコミット		デッドロック: スレッド 2 が S ロックを保有しているため、key1 に対する U ロックは X ロックにアップグレードできません。

ObjectMap.getForUpdate を使用して S ロックを回避すれば、デッドロックは回避されます。

表 4. 単一キーのデッドロック (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 		key1 のスレッド 1 に対して U ロックが認可されます。
3		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> getForUpdate key1 ◦ <code>.NET</code> GetAndLock key1 	U ロック要求がブロックされます。
4	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key1 ◦ <code>.NET</code> Put key1 	<blocked>	
5	トランザクションのコミット	<blocked>	key1 に対する U ロックは正常に X ロックにアップグレードできます。
6		<released>	スレッド 2 に対して U ロックが最終的に key1 に認可されます。
7		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> update key2 ◦ <code>.NET</code> Put key2 	key2 に対して U ロックがスレッド 2 に認可されます。
8		トランザクションのコミット	key1 に対する U ロックは正常に X ロックにアップグレードできます。

解決策:

- get メソッドではなく getForUpdate または GetAndLock メソッドを使用し、S ロックではなく U ロックを獲得します。
 - 読み取りコミット済みのトランザクション分離レベルを使用し、S ロックの保有を回避します。トランザクション分離レベルを下げると、非反復可能読み取りの可能性が増します。しかし、いずれかのクライアントからの非反復可能読み取りが可能になるのは、同じクライアントによってトランザクション・キャッシュが明示的に無効化された場合に限られます。
 - **Java** オプティミスティック・ロック・ストラテジーを使用します。オプティミスティック・ロック・ストラテジーを使用するには、オプティミスティック競合例外を処理する必要があります。(Java アプリケーションのみ)
- **問題:** 順序付けられた複数のキーでデッドロックが発生する。

説明: このシナリオでは、2 つのトランザクションが同一エントリを直接更新しようとしたときに、他のエントリに対して S ロックを保有しているとうなるかを説明します。

表 5. 順序付けされた複数のキーのデッドロックのシナリオ

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	get key1	get key1	key1 に対して S ロックが両方のトランザクションに認可されます。
3	get key2	get key2	key2 に対して S ロックが両方のトランザクションに認可されます。
4	次のいずれか 1 つ: ◦ Java update key1 ◦ .NET Put key1		U ロックはありません。更新はトランザクション・キャッシュで実行されます。
5		次のいずれか 1 つ: ◦ Java update key2 ◦ .NET Put key2	U ロックはありません。更新はトランザクション・キャッシュで実行されます。
6	トランザクションのコミット		ブロックされます。スレッド 2 が S ロックを保有しているため、key1 に対する S ロックは X ロックにアップグレードできません。
7		トランザクションのコミット	デッドロック: スレッド 1 が S ロックを保有しているため、key2 に対する S ロックはアップグレードできません。

ObjectMap.getForUpdate メソッドを使用して、S ロックを回避すれば、デッドロックを回避できます。

表 6. 順序付けされた複数のキーのデッドロックのシナリオ (続き)

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	次のいずれか 1 つ: ◦ Java getForUpdate key1 ◦ .NET GetAndLock key1		key1 に対して U ロックがトランザクション T1 に認可されます。
3		次のいずれか 1 つ: ◦ Java getForUpdate	U ロック要求がブロックされます。

		<ul style="list-style-type: none"> ◦ <code>key1</code> ◦ <code>.NET</code> <code>GetAndLock key1</code> 	
4	<code>get key2</code>	<blocked>	key2 に対して S ロックが T1 に認可されます。
5	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> <code>update key1</code> ◦ <code>.NET</code> <code>Put key1</code> 	<blocked>	
6	トランザクションのコミット	<blocked>	key1 に対する U ロックは正常に X ロックにアップグレードできます。
7		<released>	T2 に対して U ロックが最終的に key1 に認可されます。
8		<code>get key2</code>	key2 に対して S ロックが T2 に認可されます。
9		次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> <code>update key2</code> ◦ <code>.NET</code> <code>Put key2</code> 	key2 に対して U ロックが T2 に認可されます。
10		トランザクションのコミット	key1 に対する U ロックは正常に X ロックにアップグレードできます。

解決策:

- `get` メソッドではなく `getForUpdate` または `GetAndLock` メソッドを使用して、最初のキーに対する U ロックを直接取得します。この戦略が機能するのは、メソッド順序が決定論的な場合に限られます。
 - 読み取りコミット済みのトランザクション分離レベルを使用し、S ロックの保有を回避します。この解決策は、メソッド順序が決定論的でない場合に、最も簡単に実装できます。トランザクション分離レベルを下げると、非反復可能読み取りの可能性が増します。しかし、非反復可能読み取りが起こりうるのは、トランザクション・キャッシュが明示的に無効化された場合に限られます。
 - `Java` オプティミスティック・ロック・ストラテジーを使用します。オプティミスティック・ロック・ストラテジーを使用するには、オプティミスティック競合例外を処理する必要があります。(Java アプリケーションのみ)
- **問題:** 順序付けられていない U ロックからデッドロックが発生する。

説明: キーが要求される順序が保証できない場合でも、デッドロックは起きる可能性があります。

表 7. U ロックで順序付けがないシナリオ

	スレッド 1	スレッド 2	
1	トランザクションの開始	トランザクションの開始	各スレッドが独立したトランザクションを確立します。
2	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> <code>getForUpdate key1</code> ◦ <code>.NET</code> <code>GetAndLock key1</code> 	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> <code>getForUpdate key2</code> ◦ <code>.NET</code> <code>GetAndLock key2</code> 	key1 と key2 に対して U ロックが正常に認可されます。
3	<code>get key2</code>	<code>get key1</code>	key1 と key2 に対して S ロックが認可されます。
4	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> <code>update key1</code> ◦ <code>.NET</code> <code>Put key1</code> 	次のいずれか 1 つ: <ul style="list-style-type: none"> ◦ <code>Java</code> <code>update key2</code> ◦ <code>.NET</code> <code>Put key2</code> 	
5	トランザクションのコミット		T2 が S ロックを保有しているため、U ロックは X ロックにアップグレードできません。
6		トランザクション	T1 が S ロックを保有しているため、U ロック

解決策:

- すべての作業を単一のグローバル U ロックでラップします (mutex)。この方法は、並行性を低下させますが、アクセスおよび順序が決定論的でない場合に、すべてのシナリオを処理できます。
- 読み取りコミット済みのトランザクション分離レベルを使用し、S ロックの保有を回避します。この解決策は、メソッド順序が決定論的でない場合に、最も簡単に実装でき、最大の並行性を提供します。トランザクション分離レベルを下げると、非反復可能読み取りの可能性が増します。しかし、非反復可能読み取りが起こりうるのは、トランザクション・キャッシュが明示的に無効化された場合に限られます。
- **Java** オプティミスティック・ロック・ストラテジーを使用します。オプティミスティック・ロック・ストラテジーを使用するには、オプティミスティック競合例外を処理する必要があります。(Java アプリケーションのみ)

親トピック: [トラブルシューティング](#)

マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング

2.5+ 説明するシナリオは、ロック・タイムアウト例外を引き起こしているマルチ区画トランザクションの例です。トランザクションの状態に応じて、解決策で、この問題を手動で解決する方法を説明します。

始める前に

アプリケーションに例外処理を実装します。詳しくは、[ロック・シナリオでの例外処理の実装 \(Java アプリケーション\)](#) を参照してください。

結果、次の例外が表示されます。

```
Caused by: com.ibm.websphere.objectgrid.LockTimeoutException:
Local-40000139-DEF8-05EA-E000-64A856931719 timed out waiting
for lock mode S to be granted for map name: TS2_MapP, key: key12
granted = X
lock request queue
->[WXS-40000139-DEF6-FA84-E000-1CB456931719, state = Granted, requested
73423 milli-seconds ago, marked to keep current mode false,
snapshot mode 0, mode = X, thread name = xIOReplicationWorkerThreadPool : 29]
->[Local-40000139-DEF8-05EA-E000-64A856931719, state
= Waiting for 5000 milli-seconds, marked to keep current mode false,
snapshot mode 0, mode = S, thread name = xIOWorkerThreadPool : 28]
dump of all locks for WXS-40000139-DEF6-FA84-E000-1CB456931719
Key: key12, map: TS2_MapP
strongest currently granted mode for key is X
->[WXS-40000139-DEF6-FA84-E000-1CB456931719, state = Granted,
requested 73423 milli-seconds ago, marked to keep current mode false,
snapshot mode 0, mode = X, thread name = xIOReplicationWorkerThreadPool : 29]
dump of all locks for Local-40000139-DEF8-05EA-E000-64A856931719
```

このメッセージは、例外が作成されてスローされるときに、パラメーターとして渡されるストリングを表します。

手順

問題: ロック・タイムアウト例外が発生し、ロックの所有者がマルチ区画トランザクションである。または、ログ・フォルダーがログ・メッセージで増大している。

診断:

以下のようなログ・メッセージが繰り返し現れ、ログ・フォルダーが満たされています。

```
00000099 TransactionLog I CW0BJ8705I:

Automatic resolution of transaction
WXS-40000139-DF01-216D-E002-1CB456931719
at RM:TestGrid:TestSet2:20 is still waiting for a decision.
Another attempt to resolve the transaction will occur in 30 seconds.
```

ロックを引き起こしているトランザクションのタイプを判別します。トランザクション ID のプレフィックスが WXS- の場合は、マルチ区画トランザクションです。トランザクション ID のプレフィックスが Local- の場合は、トランザクションは単一区画トランザクションです。

原因: 恐らく、コミットまたはロールバックが行われなかったため、アプリケーションがロックを保持しています。

解決策: トランザクションの状態およびその状態が続いている期間を判別します。オプション `-d` (詳細出力) を指定してコマンド・ユーティリティ `xscmd -c listindoubts` を使用するか、トランザクション MBean を使用します。

2.5+ [ロック・タイムアウト例外の解決](#)

`xscmd -c listindoubt` コマンドを使用して、トランザクションの状態を表示して、アクションの方針を判別できます。

親トピック: [トラブルシューティング](#)

関連概念:

関連タスク:

[Java](#) [ロック・タイムアウト例外の解決](#)

ロック・タイムアウト例外の解決

2.5+ `xscmd -c listindoubt` コマンドを使用して、トランザクションの状態を表示して、アクションの方針を判別できます。

親トピック: **2.5+** [マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

関連概念:

[ロック・ストラテジー](#)

[2 フェーズ・コミットとエラー・リカバリー](#)

関連タスク:

[マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)

`xscmd -c listindoubts` コマンドを使用したロック・タイムアウト例外の解決

手順

- 以下のコマンドで、環境内のトランザクションの詳細リストを表示します。 `xscmd -c listindoubt -d`
- 適切なアクションを実行して、トランザクションを解決します。 **問題:** トランザクションは、TM でコミット済みとマークを付けられますが、RM は未確定です。

```
[1] WXS-40000139-DEF8-EF60-E002-1CB456931719
Timestamp          Partition  Role  State      Container      Resync
Attempts
-----
--
2012-09-19 10:40:19.824 TestSet1:11 TM    COMMIT    MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.824 TestSet1:7  RM    PREPARED  MPTBasic0_C-1 Primary  0
2012-09-19 10:40:19.839 TestSet2:20 RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.824 TestSet2:6  RM    PREPARED  MPTBasic0_C-1 Primary  0
```

解決策: リソース・マネージャー (**RM**) 区画をコミットしてから、トランザクションを無視します。

- 以下のコマンドを発行してトランザクション WXS-40000139-DEF8-EF60-E002-1CB456931719 の RM 区画をコミットします。 `xscmd -c listIndoubts -xid WXS-40000139-DEF8-EF60-E002-1CB456931719 -cm -rm`
- 以下のコマンドを発行して、このトランザクションを無視します。 `xscmd -c listIndoubts -xid WXS-40000139-DEF8-EF60-E002-1CB456931719 -f`

問題: トランザクションは、すべての区画で未確定です。

```
[1] WXS-40000139-DEF6-FA84-E000-1CB456931719
Timestamp          Partition  Role  State      Container      Resync
Attempts
-----
--
2012-09-19 10:38:11.603 TestSet1:10 RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:38:11.588 TestSet1:5  TM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:38:11.603 TestSet2:11 RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:38:11.619 TestSet2:13 RM    PREPARED  MPTBasic2_C-0 Primary  0
```

解決策: **TM** 区画をまずロールバックしてから、後続の **RM** 区画をロールバックします。次に、トランザクションを無視します。

- 以下のコマンドを発行してトランザクション WXS-40000139-DEF6-FA84-E000-1CB456931719 の TM 区画をロールバックします。 `xscmd -c listIndoubts -xid WXS-40000139-DEF6-FA84-E000-1CB456931719 -r -tm`
- 以下のコマンドを発行してこのトランザクションの RM 区画をロールバックします。 `xscmd -c listIndoubts -xid WXS-40000139-DEF6-FA84-E000-1CB456931719 -r -rm`
- 以下のコマンドを発行してこのトランザクションを無視します。 `xscmd -c listIndoubts -xid WXS-40000139-DEF6-FA84-E000-1CB456931719 -f`

問題: トランザクションは、すべての RM 区画で未確定ですが、トランザクションの決定は、TM では不明です。

```
[1] WXS-40000139-DEF8-EF31-E000-1CB456931719
Timestamp          Partition  Role  State      Container      Resync  Attempts
-----
2012-09-19 10:40:19.777 TestSet1:11 RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.792 TestSet2:5  RM    PREPARED  MPTBasic2_C-0 Primary  0
2012-09-19 10:40:19.777 TestSet2:6  RM    PREPARED  MPTBasic2_C-1 Primary  0
```

解決策: **RM** 区画をロールバックします。

- 以下のコマンドを発行してトランザクション WXS-40000139-DEF8-EF31-E000-1CB456931719 の RM 区画をロールバックします。xscmd -c listIndoubts -xid WXS-40000139-DEF8-EF31-E000-1CB456931719 -r

キャッシュ統合のトラブルシューティング

HTTP セッションや動的キャッシュ構成など、キャッシュ統合構成の問題をトラブルシューティングする場合、この情報を使用してください。

手順

- **問題:** HTTP セッション ID が再使用されない。

原因: セッション ID は再使用できます。セッション・パーシスタンスのデータ・グリッドをバージョン 7.1.1 以上で作成すると、セッション ID の再使用は自動的に有効になります。しかし、それより前の構成で作成した場合、この設定が既に誤った値で設定されている可能性があります。

解決策: 次の設定をチェックして、HTTP セッション ID の再使用が有効であるか確認します。

- splicer.properties ファイルの reuseSessionId プロパティは、true に設定する必要があります。
- HttpSessionIdReuse カスタム・プロパティ値は、true に設定する必要があります。このカスタム・プロパティは、WebSphere® Application Server 管理コンソールの次のいずれかのパスで設定されている可能性があります。
 - 「サーバー」 > 「*server_name*」 > 「セッション管理」 > 「カスタム・プロパティ」
 - 「動的クラスター」 > 「*dynamic_cluster_name*」 > 「サーバー・テンプレート」 > 「セッション管理」 > 「カスタム・プロパティ」
 - 「サーバー」 > 「サーバー・タイプ」 > 「WebSphere Application Server」 > 「*server_name*」を選択してから、「サーバー・インフラストラクチャー」セクションの下で次の順にクリックします。「Java およびプロセス管理」 > 「プロセス定義」 > 「Java 仮想マシン」 > 「カスタム・プロパティ」
 - 「サーバー」 > 「サーバー・タイプ」 > 「WebSphere Application Server」 > 「*server_name*」 > 「Web コンテナ設定」 > 「Web コンテナ」

カスタム・プロパティ値を更新した場合、eXtreme Scale セッション管理を再構成し、splicer.properties ファイルが変更を認識できるようにしてください。

- **問題:** データ・グリッドを使用して HTTP セッションを保管する際、トランザクションの負荷が高いと SystemOut.log ファイルに CWOBJ0006W メッセージが出力される。

```
CWOBJ0006W: 例外が発生しました:  
com.ibm.websphere.objectgrid.ObjectGridRuntimeException:  
java.util.ConcurrentModificationException
```

このメッセージは、Web アプリケーションが HttpSession の属性として設定された List オブジェクトを変更した場合にのみ発生します。

解決策: 変更された List オブジェクトを含んでいる属性を複製し、複製した属性をセッション・オブジェクトに設定します。

- **問題:** Web アプリケーションを Servlet 3.0 仕様で実行しているとき、Web アプリケーションのフィルターとリスナーが WebSphere eXtreme Scale セッション管理によって起動されません。例えば、WebSphere eXtreme Scale でリモート・コンテナ除去を使用してセッションが無効化されると、リスナーがコールバックされません。

原因: WebSphere eXtreme Scale は、注釈を使用して定義されたりプログラマチックに定義されたりしたフィルターやリスナーを特定しません。

解決策: フィルターやリスナーは、Web アプリケーションの web.xml ファイルで明示的に宣言する必要があります。

親トピック: [トラブルシューティング](#)

関連概念:

[アプライアンスのトポロジー: 集合、ゾーン、およびデータ・グリッド](#)

インストールのトラブルシューティング

インストールおよび更新の問題をトラブルシューティングする場合、この情報を使用してください。

手順

- **問題:** インストール・コマンドをリモート・コンピューター (¥¥mymachine¥downloads¥ など) から実行すると、次のメッセージが表示される: 現在のディレクトリとして上記のパスで CMD.EXE を開始しました。UNC パスはサポートされません。 Windows ディレクトリを既定で使用します。結果として、インストールが正常に完了しない。

解決策: リモート・コンピューターをネットワーク・ドライブにマップしてください。例えば、Windows の場合、「マイ コンピュータ」を右クリックし、「ネットワーク ドライブの割り当て」を選択して、リモート・コンピューターへの 汎用名前付け規則 (UNC) パスを組み込むことができます。その後は、ネットワーク・ドライブから正常にインストール・スクリプトを実行できます。例: y:¥mymachine¥downloads¥WXS¥install.bat。

- **問題:** インストールが正常に完了しない。

解決策: ログ・ファイルをチェックして、インストールがどこで失敗したか 確認します。インストールが正常に完了しない場合は、ログは [wxs_install_root/logs/wxs](#) ディレクトリにあります。

- **問題:** インストール時に重大な障害が発生する。

解決策: ログ・ファイルをチェックして、インストールがどこで失敗したか 確認します。インストールが部分的に完了した時点でインストールが失敗した場合、ログは 一般的に `user_root/wxs_install_logs/` ディレクトリにあります。

- **Windows** **問題:** Windows 上で WebSphere® eXtreme Scale クライアント をインストールする場合、インストールの結果に 次のテキストが表示されることがある。

```
Success: The installation of the following product was successful:
WebSphere eXtreme Scale Client. Some configuration steps have errors.
For more information, refer to the following log file:
<WebSphere Application Server install root>¥logs¥wxs_client¥install¥log.txt"
Review the installation log (log.txt) and review the deployment manager
augmentation log.
```

解決策: `iscdeploy.sh` ファイルで発生した障害の場合、エラーは 無視してかまいません。このエラーでは、問題は生じません。

- **Linux** **問題:**

フルインストールの環境で、WebSphere eXtreme Scale クライアント に対してアップデート・インストーラーを使用して保守のみを適用しようとする、以下のメッセージが表示されます。

```
Prerequisite checking has failed. Click Back to select a different package,
or click Cancel to exit.
```

Failure messages are:

```
Required feature wxs.client.primary is not found.
```

WebSphere eXtreme Scale クライアント がインストールされていて、アップデート・インストーラーを使用してすべての保守パッケージを適用しようとする、次のメッセージが表示されます。

```
Prerequisite checking has failed. Click Back to select a different package,
or click Cancel to exit.
```

Failure messages are:

```
Required feature wxs.primary is not found.
```

解決策: インストールする保守パッケージはインストール済み環境のタイプと一致している必要があります。ご使用のインストール済み環境のタイプに適合する保守パッケージをダウンロードし、適用してください。

- **Linux** **問題:** インストールがハングします。

解決策: 非 root ユーザーで Linux に WebSphere eXtreme Scale をインストールしている時に、インストーラーがハングする場合があります。これは、オープン・ファイルの最大数が Linux オペレーティング・システム上であまりに低く設定されていることが原因と考えられます。 `/etc/limits.conf` ファイルまたは `/etc/security/limits.conf` ファイル (ファイルの場所は、個別の Linux 配布版によって異なります) 内で許

可される限度を少なくとも 8192 まで上げる必要があります。

親トピック: [トラブルシューティング](#)

関連タスク:

[Update Installer を使用して保守パッケージをインストールする](#)

管理のトラブルシューティング

サーバーの開始や停止、**xscmd** ユーティリティの使用など、管理についてのトラブルシューティングを行う場合、この情報を使用してください。

手順

- **問題:** **xscmd** コマンドの実行時に次のメッセージが画面に表示される。

```
java.lang.IllegalStateException: Placement service MBean not available.
[]
    at
com.ibm.websphere.samples.objectgrid.admin.OGAdmin.main(OGAdmin.java:1449)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:60)
    at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:37)
)
    at java.lang.reflect.Method.invoke(Method.java:611)
    at com.ibm.ws.bootstrap.WSLauncher.main(WSLauncher.java:267)
Ending at: 2011-11-10 18:13:00.000000484
```

原因: カタログ・サーバーで接続の問題が発生しました。

解決策: カタログ・サーバーが実行中であり、ネットワーク経由で使用可能なことを確認します。このメッセージは、カタログ・サービス・ドメインが定義されている場合に、実行中のカタログ・サーバーが2つより少ないとき発生することもあります。そのような環境は、2つのカタログ・サーバーが開始されるまで使用できません。

- **問題:** **xscmd** コマンドの実行時に次のメッセージが画面に表示される。

```
CWXSI0066E: 一致しない引数 argument_name が検出されました。
```

原因: 入力されたコマンド・フォーマットは **xscmd** ユーティリティによって認識されませんでした。

解決策: コマンドのフォーマットを確認してください。 **-c findbyKey** コマンドを使用して正規表現を実行するときに、この問題が発生する可能性があります。詳しくは、[データの照会、表示、および無効化](#)を参照してください。

親トピック: [トラブルシューティング](#)

参照

参照情報は、特定の項目を迅速に見つけることができるように編成されています。

[xscmd ユーティリティー・リファレンス](#)

xscmd ユーティリティーを使用する場合、次のコマンドのリストをリファレンスとして使用できます。

2.5+ [HTTP コマンド・インターフェースのリファレンス](#)

[クライアント・プロパティ・ファイル](#)

WebSphere® eXtreme Scale クライアント・プロセスの要件に基づいて、プロパティ・ファイルを作成します。

[正規表現の構文](#)

正規表現は、他のストリングとのマッチングに使用される構造化ストリングです。正規表現は、データを無効化する場合およびメッセージに表示されるメッセージをフィルタリングする場合に使用することができます。

[動的マップの構成オプション](#)

ご使用のクライアント・アプリケーションを特定の名前の付いたマップに接続させることで、データ・グリッド内に追加のマップを作成できます。この接続が発生すると、マップが自動的に作成されます。

[ユーザー・インターフェース設定](#)

この参照情報では、WebSphere Application Server 管理コンソールなどのページで表示および構成できる設定について説明します。

xscmd ユーティリティー・リファレンス

xscmd ユーティリティーを使用する場合、次のコマンドのリストをリファレンスとして使用できます。

汎用コマンド・パラメーター

xscmd ユーティリティー・コマンドの一般的な形式は次のとおりです。オプション・パラメーターは、大括弧 [] で囲まれています。

注意:

WebSphere® DataPower® XC10 アプライアンス 環境では、次のコマンドは使用しないでください。

- **-c releaseShard**
- **-c reserveShard**
- **-c swapShardWithPrimary**
- **-c suspendBalancing**
- **-c resumeBalancing**
- **-c teardown**
- **-c triggerPlacement**
- **-c enableForPlacement**

```
Usage: xscmd -c <cmdName> | -h <cmdName> | -lc [<cmdGroupName>] | -lcg
      [-tt <type>] [-prot <protocol>] [-trf <filePath>] [-ks
      <filePath>] [-ksp <password>] [-user <username>] [-al <alias>]
      [-cgp <property>] [-kst <type>] [-cep <endpoints>] [-ssp
      <profileName>] [-tsp <password>] [-arc <integer>] [-trs
      <traceSpec>] [-tst <type>] [-to <serverTimeout>] [-cxpv
      <provider>] [-sp <profileName>] [-pwd <password>] [-ca <support>]
      [-cgc <className>] [-ts <filePath>]
```

Options:

- | | |
|--------------------------------------|--|
| -al, --alias <alias> | Alias name in the keystore. |
| -arc, --authRetryCount <integer> | The retry count for authentication if the credential is expired. If the value is set to 0, then authentication retries do not occur. |
| -c, --command <cmdName> | Specifies the name of the command to execute |
| -ca, --credAuth <support> | Set the client credential authentication support [Never, Supported, Required]. |
| -cep, --catalogEndPoints <endpoints> | Specifies one or more catalog service endpoints in the format <host>[:<listenerPort>][,<host>[:<listenerPort>]]. Default endpoint: localhost:2809 |
| -cgc, --credGenClass <className> | Specifies the name of the class that implements the CredentialGenerator interface. This class is used to get credentials for clients. |
| -cgp, --credGenProps <property> | Specifies the properties for the CredentialGenerator implementation class. The properties are set to the object with the setProperties(String) method. |
| -cxpv, --contextProvider <provider> | Context provider. Examples: IBMJSSE2, IBMJSSE, IBMJSSEFIPS. |
| -h, --help <cmdName> | Invokes general command-line help |
| -ks, --keyStore <filePath> | Absolute path to keystore. Example: /etc/test/security/server.public |

-ksp,--keyStorePassword <password>	Specifies the password to the keystore.
-kst,--keyStoreType <type>	Keystore type. Example: JKS, JCEK, PKCS12.
-lc,--listCommands <cmdGroupName>	List all commands for a command group
-lcg,--listCommandGroups	List all command groups
-lpc,--listPrivateCommands	List all private commands.
-prot,--protocol <protocol>	Protocol. Examples: SSL, SSLv2, SSLv3, TLS, TLSv1.
-pwd,--password <password>	eXtreme Scale password security credential
-sp,--secProfile <profileName>	Specifies a profile name.
-ssp,--saveSecProfile <profileName>	Save security parameter values in security profile.
-to,--timeout <serverTimeout>	Server connection timeout in seconds
-trf,--traceFile <filePath>	Specifies the absolute path to the generated trace file for xscmd command output
-trs,--traceSpec <traceSpec>	Specifies the trace specification for xscmd command output
-ts,--trustStore <filePath>	Absolute path to truststore. Example: /etc/test/security/server.public
-tsp,--trustStorePassword <password>	Truststore password
-tst,--trustStoreType <type>	Truststore type. Examples: JKS, JCEK, PKCS12.
-tt,--transportType <type>	Transport layer security configuration type. Examples: TCP/IP, SSL-Supported, SSL-Required.
-user,--username <username>	eXtreme Scale user name security credential

すべてのコマンド

次に、**xscmd** ユーティリティ・コマンドをすべてリストします。

注: 列名は変わることがあります (列のタイプと位置は変わりません)。また、コマンドが拡張された場合、必要に応じて新しい列が表の最後に追加されることがあります。

特定のコマンドの詳細およびパラメーターを表示するには、**xscmd -h command_name** を実行してください。これらのコマンドを使用してカスタマイズ・スクリプトを作成する計画の場合は、**./xscmd.sh -c commandName -Options 2> /dev/null** を実行して、標準エラー・パスをヌル・デバイスに指定変更してください。

例

以下の例は、**xscmd** ユーティリティからコマンドを実行する方法を示しています。

```
./xscmd.sh -lc (このコマンドにより、使用可能なコマンドがすべてリストされます。)
```

```
./xscmd.sh -h showMapSizes (このコマンドにより、showMapSizes コマンドのヘルプがリストされます。)
```

重要: 以下のコマンドの出力と使用法については、将来変更される可能性があります。コマンドにアスタリスク (*) が付いている場合は、そのコマンドの出力のみ、変更の可能性があることを意味しています。

- listHosts
- showPlacement
- placementServiceStatus

- showDomainReplicationState
- showReplicationState
- *osgiAll
- *osgiCheck
- *osgiCurrent
- *osgiUpdate
- *showLinkedPrimaries
- *triggerPlacement

Command Name	Description
-----	-----
balanceShardTypes	Attempt shard redistribution so that the ratio of primaries and replicas within each container server are within one shard.
balanceStatus	Check the balance status of the data grid for the ObjectGrid and map set specified.
clearGrid	Clears data from the data grid.
dismissLink	Disconnects from the specified catalog service domain.
enableForPlacement	Re-enable shard placement to a container that is disabled for shard placement because of the failure of a previous shard placement operation.
establishLink	Connects to the specified catalog service domain with the specified catalog service endpoints.
findByKey	Finds matching keys in a map.
getCatTraceSpec	Retrieves the trace specification for all the catalog servers that are known by this process.
getNotificationFilter	Displays the notification filters for the servers in the environment.
getStatsSpec	Retrieves the statistics specification.
getTraceSpec	Retrieves the trace specification.
listAllJMXAddresses	Displays all JMX MBean server addresses.
listCoreGroups	List all core groups.
listDisabledForPlacement	List the containers that are disabled for shard placement because shard placement operations failed for them.
listHosts	Lists all hosts. The command usage and output is subject to change in the future.
listIndoubts	List and resolve indoubt transactions.
listObjectGridNames	Lists all known ObjectGrid instances and map sets.
listProfiles	Lists profiles.
listenForNotifications	Subscribes to notifications received by the messaging hub. Prints the errors, warnings and other messages as they are received.
osgiAll	Show all available OSGi service rankings. Use the -sn option to display a single service. The command output only is subject to change in the future.
osgiCheck	Check whether specified OSGi service rankings are

	available. The command output only is subject to change in the future.
osgiCurrent	Show all OSGi service rankings currently in use. Use the -sn option to display a single service. The command output only is subject to change in the future.
osgiUpdate	Update the OSGi services to the specified rankings. The command output only is subject to change in the future.
overrideQuorum	Notify the catalog server to override quorum.
placementServiceStatus	Displays the ObjectGrid placement operation status. The command usage and output is subject to change in the future.
releaseShard	Releases the specified primary shard from the specified container server.
removeProfile	Remove profile from file system.
reserveShard	Reserves the specified primary shard on the specified container server.
resumeBalancing	Attempts to balance and allow future balancing attempts for the specified ObjectGrid instance and map set.
revisions	Displays all known revision history.
routetable	Displays the current routing table.
setCatTraceSpec	Sets the trace specification for all the catalog servers known by this process.
setNotificationFilter	Sets the notification filter. The messaging hub processes INFO, WARNING, and SEVERE messages that match the regular expression.
setStatsSpec	Sets the statistics specification.
setTraceSpec	Trace specification in the form: traceType1=traceLevel1=traceState1[: traceTypeN=traceLevelN=traceStateN]*
showCoreGroupMembers	Shows all core group members.
showDomainReplicationState	Displays the outbound and inbound revisions yet to be replicated between primary shards on linked domains, for each container across all domains. The command usage and output is subject to change in the future.
showInfo	Retrieves the environment specifications including installed versions and JVM information.
showLinkedDomains	Displays the linked foreign domains
showLinkedPrimaries	Displays the primary shards and all of their foreign or domestic linked primary shards. The command output only is subject to change in the future.
showMapSizes	Displays all map sizes.
showNotificationHistory	Displays the most recent environment errors,

	warnings and messages that are stored in the messaging hub.
showPlacement	Lists all container servers and their shards. The command usage and output is subject to change in the future.
showProfile	Show the details of the specified profile.
showQuorumStatus	Displays catalog server quorum status.
showReplicationState	Displays the outbound and inbound revisions yet to be replicated between the primary and replica shards for each container. The command usage and output is subject to change in the future.
showTransport	Displays the transport used by the catalog service domain. Types include ORB and eXtremeIO.
suspendBalancing	Prevents future attempts to balance the specified ObjectGrid instance and map set.
swapShardWithPrimary	Swaps the specified replica shard for the specified container server with its primary shard.
teardown	Stops a list of catalog and container servers.
triggerPlacement	Triggers a placement operation for the specified ObjectGrid instance and map set. The numInitialContainers value is ignored. The command output only is subject to change in the future.

親トピック: [参照](#)

関連タスク:

[xscmd ユーティリティーによる管理](#)

[xscmd ユーティリティーのためのセキュリティー・プロファイルの構成](#)

[データの照会、表示、および無効化](#)

HTTP command interface reference

With the HTTP command interface, you can run operations on your appliance, configure appliance settings, and administer data grids, collectives, and zones.

Table of contents

- [List of APPLIANCE commands](#)
- [List of COLLECTIVE commands](#)
- [List of GRID commands](#)
- [List of TASK commands](#)

List of APPLIANCE commands

CreateAdminTrace

D
e
s
c
r
i
p
t
i
o
n:

Adds or modifies an existing administrative trace string.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**tr
a**

**c
e
N
a
m
e** Specifies the name of the trace.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage** A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult** A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e** Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"AutoCustomLogger", "command": "CreateAdminTrace" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

CreateAggregateInterface

D
e

description:

Creates a new Aggregate Interface.

Required Parameters:

member

Lists all ethernet interface members of this aggregation.

stopontaskfailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

primary-member

Specifies which of the aggregated ethernet interfaces is the primary one.

r

agg
r
e
g
a
t
i
o
n
-
p
o
l
i
c
y

Specifies the aggregation policy of the interface.

a
d
d
r
e
s
s

Specifies IP address of the interface.

n
a
m
e

Specifies the interface name.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "member": "eth0,eth1", "stopOnTaskFailure": "true",  
"primary_member": "eth0", "aggregation_policy": "active-backup",  
"address": "1:2:3:4/24", "name": "agg1", "command":  
"CreateAggregateInterface" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

CreateDataCacheTrace

D
e
s
c
r
i
p
t
i
o
n:

Adds or modifies an existing data cache trace string.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**F
a
i
l
u
r
e**

**tr
a
c
e
N
a
m
e**

Specifies the name of the trace.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"autoDataCacheLogger", "command": "CreateDataCacheTrace" } }
```

C
o
m
m
a
n
d
T
y
p
e

appliance

p
e:

CreateSNMPCommunity

D
e
s
c
r
i
p
t
i
o
n:

Creates a Simple Network Management Protocol (SNMP) community.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**h
o
s
t
R
e
s
t
r
i
c
t
i
o
n**

(Optional) Specifies an IP address on which to restrict SNMP communication. Communication with any other IP address or subnet is denied.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
c
o
m
m
u
n
i
t
y
N
a
m
e**

Specifies the name of the SNMP community to create.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "hostRestriction": "autorestriction", "stopOnTaskFailure":  
"true", "command": "CreateSNMPCommunity", "communityName":  
"autocommunity" } }
```

**C
o
m
m
a
n
d
T
y
p
e**

appliance

p
e:

DeleteSNMPCommunity

D
e
s
c
r
i
p
t
i
o
n:

Deletes a Simple Network Management Protocol (SNMP) community.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

c
o
m
m
u
n
i
t
y
N
a
m

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies the name of SNMP community to delete.

e

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"DeleteSNMPCommunity", "communityName": "autocommunity" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

EnableCreateAccount

D
e
s
c
r
i
p

Enables a setting that allows users to initiate the creation of their

tion:

own accounts.

Required Parameters:

stopOnTaskFailure
enable

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Enables account creation. Set the value to false to disable account creation. Set the value to true to enable account creation.

Result Parameter

rs
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "enable": "true", "command": "EnableCreateAccount" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

EnablePasswordReset

D
e
s
c
r
i
p
t
i
o
n:

Enables the ability to reset the xadmin password with a serial connection. No other credentials or SMTP messages are required.

R
e
q
u
i
r
e
d
P
a
r
a
m
e

ters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

enable

Enables the password to be reset. Set the value to true to enable password resets.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

E

X
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "enable": "true", "command":  
"EnablePasswordReset" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

EnableSNMP

D
e
s
c
r
i
p
t
i
o
n:

Enables or disables Simple Network Management Protocol (SNMP).

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**il
u
r
e

e
n
a
b
l
e**

Enables SNMP. Set the value to false to disable SNMP. Set the value to true to enable SNMP.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

- errorMessage** A message that explains the reason for the failure, if the command failed.
- status** Specifies the status of the command that was run. The result can be either success or failed.
- commandResult** A JSON-formatted statement that contains the result of the command that was run.
- commandName** Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "stopOnTaskFailure": "true", "enable": "true", "command": "EnableSNMP" } }
```

**C
o
m
m
a
n
d
T
y
p
e:**

appliance

ModifyAdminDefaultTrace

D
e
s
c
r
i
p
t
i
o
n:

Changes the trace level for the administrative default logger.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**tr
a
c
e
L
e
v
e
l**

Specifies the trace level. Valid values: OFF, SEVERE, WARNING, or INFO.

R
e

S
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "traceLevel": "SEVERE",  
"command": "ModifyAdminDefaultTrace" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyAdminTrace

D
e
s
c
r
i
p
t
i
o
n:

Changes an administrative logger trace level.

R
e

q
u
i
r
e
d
p
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

tr
a
c
e
L
e
v
e
l

Specifies the trace level. Valid values: OFF, SEVERE, WARNING, INFO, FINE, FINER, FINEST, ALL.

tr
a
c
e
N
a
m
e

Specifies the name of the trace to modify.

R
e
s
u
l
t
P
a
r
a

parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceLevel": "SEVERE",  
"traceName": "autoCustomLogger", "command": "ModifyAdminTrace"  
} }
```

Command Type:

appliance

ModifyAdministratorEmail

Description:

Changes the default administrator email address.

Required P

a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**a
d
m
i
n
i
s
t
r
a
t
o
r
E
m
a
i
l**

Specifies a new email address for the default administrator account.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "administratorEmail": "somenewname@us.ibm.com", "command": "ModifyAdministratorEmail" } }
```

Command Type:

appliance

ModifyAdministratorName

Description:

Changes the default administrator account name.

Required Parameters:

st

**o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**a
d
m
i
n
i
s
t
r
a
t
o
r
N
a
m
e**

Specifies the new name for the default administrator account.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E

X
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "administratorName":  
"somenewname", "command": "ModifyAdministratorName" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyAdministratorPassword

D
e
s
c
r
i
p
t
i
o
n:

Changes the password for the default administrator account.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**st
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**n
e
w
P
a
s
s
w
o
r
d**

Specifies a new default administrator account password.

**p
a
s
s
w
o
r
d
V
e
r
i
f
i
c
a
t
i
o
n**

Specifies a new default administrator account password, entered a second time for verification.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "newPassword": "somepass",  
"passwordVerification": "somepass", "command":  
"ModifyAdministratorPassword" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyAggregateInterface

D
e
s
c
r
i
p
t
i
o
n:

Modifies Aggregate Interface parameters.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

n

a
m
e Specifies the interface name.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorM
essage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

comma
ndRes
ult A JSON-formatted statement that contains the result of the command that was run.

comma
ndNam
e Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "mtu": "1500", "member": "eth0,eth1",  
"stopOnTaskFailure": "true", "transmit_hash_policy": "layer2",  
"dad_retransmit_timer": "1000", "use_slaac": "true", "mode": "Auto",  
"use_dhcp": "false", "use_arp": "true", "primary_member": "eth0",  
"aggregation_policy": "active-backup", "address": "1:2:3:4/24",  
"name": "agg1", "command": "ModifyAggregateInterface",  
"dad_transmits": "1", "lACP_selection_logic": "stable",  
"ipv4_default_gateway": "1:2:3:5" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyDataCacheTrace

D
e
s

cr
ip
ti
o
n:

Adds or modifies an existing data cache trace string.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**tr
a
c
e
L
e
v
e
l**

Specifies the trace level. Valid values: OFF, SEVERE, WARNING, INFO, FINE, FINER, FINEST, ALL.

**tr
a
c
e
N
a
m
e**

Specifies the name of the trace.

R

Result Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceLevel": "SEVERE",  
"traceName": "autoDataCacheLogger", "command":  
"ModifyDataCacheTrace" } }
```

Command Type:

appliance

ModifyEthernetInterface

Description:

Modifies Ethernet Interface parameters.

R

Required Parameters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

name

Specifies the interface name.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

comma

ndResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "mtu": "1500", "use_dhcp": "false", "use_arp": "true", "stopOnTaskFailure": "true", "dad_retransmit_timer": "1000", "use_slaac": "true", "address": "1:2:3:4/24", "name": "eth0", "dad_transmits": "1", "command": "ModifyEthernetInterface", "ipv4_default_gateway": "1:2:3:5", "mode": "Auto" } }
```

Command Type:

appliance

ModifyLDAP

Description:

Configures the appliance to use Lightweight Directory Access Protocol (LDAP) for user login authentication.

Required Parameters:

jndiSecurityPrincipal Specifies the JNDI security principal.

**nci
pal**

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

ldapUserSearchFilterPattern LDAP user search filter pattern.

newPassword Specifies the new LDAP password.

groupBaseDn Specifies the LDAP JNDI base distinguished name (DN) for groups.

passwordVerification Specifies the new LDAP password a second time for password verification. The value must be the same as the newPassword parameter.

userBaseDn Specifies the LDAP JNDI base distinguished name (DN) for users.

enable Specifies if LDAP user authentication is enabled. Set the value to true to enable LDAP user authentication.

jndiProviderURL Specifies the URL for the LDAP provider.

rs
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "jndiSecurityPrincipal":  
"CN=Administrator,CN=users,DC=mycompany,DC=com",  
"ldapUserIdSearchFilterPattern": "thefilterpattern", "newPassword":  
"somepassword", "passwordVerification": "somepassword",  
"groupBaseDn": "cn=group", "userBaseDn": "cn=user", "enable":  
"true", "command": "ModifyLDAP", "jndiProviderURL": "someurl" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ModifyNtpServers

D
e
s
c
r
i
p
t
i
o
n:

Changes the list of Network Time Protocol (NTP) servers.

R
e
q
u
i
r
e
d
P
a
r
a
m

eters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

listOfServers

Specifies a list of comma-separated Network Time Protocol (NTP) servers.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "listOfServers":  
"time.nist.gov,10.10.2.2", "command": "ModifyNtpServers" } }
```

Command Type:

appliance

ModifySMTPEmail

Description:

Changes the reply-to email address that is used for sending passwords that are reset by users.

Required Parameters:

stopOn

Specifies whether to stop running the batch routine

**T
a
s
k
F
a
i
l
u
r
e** when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
m
a
i
l
A
d
d
r
e
s
s** Specifies the reply-to email address. Use the email address for the administrator.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage** A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult** A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e** Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ModifySMTPEmail", "emailAddress": "admin@mycompany.com" } }
```

C
o

m
m
a
n
d
T
y
p
e:

appliance

ModifySMTPServer

D
e
s
c
r
i
p
t
i
o
n:

Changes the Simple Mail Transfer Protocol (SMTP) server.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

**h
o
s
t**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Name Specifies the SMTP host address.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command": "ModifySMTPServer", "hostName": "sicdsjc" } }
```

Command Type:

appliance

ModifySearchFilterUsers

De

S
c
r
i
p
t
i
o
n:

Changes the Lightweight Directory Access Protocol (LDAP) search filter for users.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

ldapUserSearchFilterPattern LDAP user search filter pattern.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true",
"ldapUserIdSearchFilterPattern": "(&(uid={0})
(objectclass=inetOrgPerson))", "command":
"ModifySearchFilterUsers" } }
```

Command Type:

appliance

ModifyTimeZone

Description:

Changes the time zone for the appliance.

Required Parameters:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**l
o
c
a
l
T
i
m
e
Z
o
n
e**

Specifies the new time zone as an abbreviated string. Valid values: EST : Eastern Standard Time
CST : Central Standard Time MST : Mountain Standard Time PST : Pacific Standard Time HAST : Hawaii AKST : Alaska AST : Atlantic UTC : Coordinated Universal Time GMT : Greenwich Mean Time CET : Central Europe EET : Eastern Europe MSK : Moscow AST_ARABIA : Arabia KRT : Pakistan IST : India NOV : Novosibirsk CST_CHINA : China JST : Japan AWST : Australia West ACST : Australia Central AEST : Australia East

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e**

```
{ "task": { "stopOnTaskFailure": "true", "command":
```

pl "ModifyTimeZone", "localTimeZone": "EST" } }

e:

C
o
m
m
a
n
d
T
y
p
e:

appliance

PingRemoteHost

D
e
s
c
r
i
p
t
i
o
n:

Tests the visibility of a host from this appliance.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

e
h
o
s
t
N
a
m
e

Specifies the IP address or host name of the remote host.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"PingRemoteHost", "hostName": "io03.rtp.raleigh.ibm.com" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

RemoveAdminTrace

Description:

Removes an existing administrative console trace string.

Required Parameters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

traceName

Specifies the name of the trace.

Result

Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"AutoCustomLogger", "command": "RemoveAdminTrace" } }
```

Command Type:

appliance

RemoveAggregateInterface

Description:

Deletes Aggregate Interface of a given name.

Require

readParameters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

name

Specifies the interface name.

ResultParameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "name": "agg1", "command": "RemoveAggregateInterface" } }
```

Command Type:

appliance

RemoveDataCacheTrace

Description:

Removes an existing data cache trace string.

Required Parameters:

stopOnT

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the

stopOnTaskFailure batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

traceName Specifies the name of the trace.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "traceName":  
"autoDataCacheLogger", "command": "RemoveDataCacheTrace" } }
```

Command:

appliance

d
T
y
p
e:

RestartAppliance

D
e
s
c
r
i
p
t
i
o
n:

Restarts or shuts down the appliance.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**i
m
m
e
d
i
a
t
e**

Specifies if the appliance waits to complete active tasks before restarting. If the value is set to true, the appliance restarts immediately. If the value is set to false, the appliance waits for all active tasks to complete before restarting.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
s
h
u
t
d
o
w
n**

Specifies if the appliance restarts or shuts down. If the value is set to true, the appliance is shut down. If the value is set to false, the appliance restarts.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "immediate": "false", "stopOnTaskFailure": "true",  
"command": "RestartAppliance", "shutdown": "true" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

ViewAllAggregateInterfaces

D
e
s
c
r
i
p
t
i
o
n:

Displays the information for every aggregate interface.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**S
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t

ers
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ViewAllAggregateInterfaces" } }
```

Command Type:

appliance

ViewAllEthernetInterfaces

Description:

Displays the information for every Ethernet Interface.

Required Parameters

me
te
rs
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ViewAllEthernetInterfaces" } }
```

C
o
m
m
a
n
d
T
y
p
e:

appliance

List of COLLECTIVE commands

AddApplianceToCollective

D
e
s
c
r
i
p
t
i
o
n:

Adds an existing appliance to the current collective.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**a
p
p
l
i
a
n
c
e
l
P**

Specifies the IP address of the appliance to add to the collective.

**s
t
o
p
O**

n Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

T
a
s
k
F
a
i
l
u
r
e

s
e
c
r
e
t
K
e
y

Specifies the secret key of the appliance.

w
a
i
t
O
n
T
a
s
k

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

z
o
n
e
N
a
m
e

Specifies the name of the zone to which you want to assign the appliance.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorM
essage

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "applianceIP": "192.168.222.128", "stopOnTaskFailure": "true", "secretKey": "P5Xa4F8MQeSxuuhKxnaStw==", "waitOnTask": "true", "command": "AddApplianceToCollective", "zoneName": "DefaultZone" } }
```

Command Type:

collective

AddRoleToGroup

Description:

Assigns an access role to a group. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

Required Parameters:

**r
o
u
p
N
a
m
e**

Specifies the name of the group that is being assigned to a role.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**r
o
l
e**

Specifies the role to assign to the group. Valid values: 2 = appliance administration, 3 = appliance monitoring, 5 = data cache creation.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
X

sample: { "task": { "stopOnTaskFailure": "true", "groupName": "autoGroup1003", "command": "AddRoleToGroup", "role": "5" } }

Command Type: collective

AddRoleToUser

Description: Assigns an access role to a user. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

Required Parameters:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**r
e**

**r
o
l
e**

Specifies the role to assign to the user. Valid values:
2 = appliance administration, 3 = appliance
monitoring, 5 = data cache creation.

**u
s
e
r
N
a
m
e**

Specifies the name of the user that is being
assigned a role.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the
failure, if the command failed.

status

Specifies the status of the command that was
run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the
result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command
interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command": "AddRoleToUser",  
"role": "2", "userName": "autoUser1003" } }
```

C
o
m
m
a
n
d
T

collective

y
p
e:

AddUserToGroup

D
e
s
c
r
i
p
t
i
o
n:

Adds a user to a group. After the user is added to the group, the user has the roles that are assigned to the selected group only. Any individual user roles are lost.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of a defined group.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**u
s
e
r
N
a
m
e**

Specifies the name of a defined user.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName":  
"autoGroup1003", "command": "AddUserToGroup", "userName":  
"autoUser1003" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

CreateCollectiveLink

D
e
s
c
r
i
p
t
i
o
n:

Creates a link to another collective.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**a
p
p
l
i
a
n
c
e
N
a
m
e**

Specifies the host name or IP address of an appliance that is a member of another collective.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

r

**e
m
o
t
e
P
a
s
s
w
o
r
d**

Specifies the password that is used to log in to the remote appliance.

**r
e
m
o
t
e
U
s
e
r
N
a
m
e**

Specifies the user name that is used to log in to the remote appliance.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s**
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
X**

```
{ "task": { "applianceName": "myappliance.mydomain.com",
```

example: "stopOnTaskFailure": "true", "command": "CreateCollectiveLink",
"remotePassword": "somePassword", "remoteUserName": "someUser"
} }

Command
Type:

collective

CreateGroup

Description:

Creates a new group. Groups are useful for assigning multiple users the same set of roles.

Required Parameters:

**group
Name**

Specifies a name for the new group. Group names must be unique.

st

**o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
o
u
p
D
e
s
c
r
i
p
t
i
o
n**

Specifies a description of the group and its users.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
X**

example: { "task": { "stopOnTaskFailure": "true", "groupName": "autoGroup1003", "groupDescription": "this group is huge", "command": "CreateGroup" } }

Command Type: collective

CreateUser

Description: Creates a new user.

Required Parameters:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**as
s
w
o
r
d
V
e
r
i
f
i
c
a
t
i
o
n**

Specifies the password, entered a second time for verification.

**us
er
P
a
s
s
w
o
r
d**

Specifies a new password for the selected user.

**e
m
a
i
l
A
d
d
r
e
s
s**

Specifies an email address for the new user.

**fu
ll
U
s
e
r
N
a
m
e
us
er
N
a
m
e**

Specifies a full name for the user. Example: John E. Doe.

Specifies a short name for the new user. Example: xadmin.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e**

rs
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "passwordVerification":  
"someuserpass", "userPassword": "someuserpass", "command":  
"CreateUser", "emailAddress": "someuser@us.ibm.com",  
"fullUserName": "someuser isme", "userName": "someuser" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

DeleteCollectiveLink

D
e
s
c
r
i
p
t
i
o
n:

Removes the link between the local collective and the specified collective.

R
e
q
u
i
r
e
d
P
a
r
a
m
e

ters:

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

name

Specifies the name of the collective to delete.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "name": "myCollectiveName",  
"command": "DeleteCollectiveLink" } }
```

e:
C
o
m
m
a
n
d
T
y
p
e:

collective

DeleteGroup

D
e
s
c
r
i
p
t
i
o
n:

Deletes a group. Users that belonged to the group no longer belong to the group and might lose any assigned roles.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

**s
t
o
p
O
n**

Specifies the name of a defined group.

T Specifies whether to stop running the batch routine
a when the task fails. If the value is set to true, the
s batch routine stops. If the value is false or blank, the
k batch routine does not stop. Applies to commands
F that are run in batch processes only.
a
i
l
u
r
e

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorM
essage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

comma
ndRes
ult A JSON-formatted statement that contains the result of the command that was run.

comma
ndNam
e Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName":  
"autoGroup1003", "command": "DeleteGroup" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

DeleteUser

Description:

Deletes a user. If the user belongs to a group, the user is removed from the group.

Required Parameters:

stopOnTaskFailure
userName

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies the name of a defined user.

Result

Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command": "DeleteUser",  
"userName": "autoUser1003" } }
```

Command Type:

collective

GetCollectiveName

Description:

Retrieves the collective name.

Requires:

e
d
p
a
r
a
m
e
t
e
r
s
:

s
t
o
p
o
n
t
a
s
k
f
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x

Example: { "task": { "stopOnTaskFailure": "true", "command": "GetCollectiveName" } }

Command Type: collective

GetHealthStatus

Description: Displays the hardware and software Health Status information for the appliance.

Required Parameters:

stopOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

il u r e

R e s u l t P a r a m e t e r s :

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E x a m p l e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"GetHealthStatus" } }
```

C o m m a n d T y p e:

collective

ModifyGroupDescription

D e

s
c
r
i
p
t
i
o
n
:

Changes the description of a selected group.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of a defined group.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
o
u
p
D
e
s
c**

Specifies a description of the group and its users.

ri p t i o n

R e s u l t P a r a m e t e r s :

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E x a m p l e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "somegroup",  
"groupDescription": "somegroupdesc", "command":  
"ModifyGroupDescription" } }
```

C o m m a n d T y p e:

collective

ModifyScheduleExportSettings

D e

S
c
r
i
p
t
i
o
n:

Enable and disable the scheduled exports feature. Also, specify an interval to generate a configuration export.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**d
a
y
l
n
t
e
r
v
a
l**

Specifies how many days to wait in between scheduled configuration exports.

**t
i
m
e**

Specifies the scheduled date that the first export will occur. Must be in the following format: YYYY-MM-DD. For example, "2014-09-08".

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**e
n
a
b
l
e**

Specifies if the scheduled configuration export feature is enabled or disabled.

d
a
t CLI.parameter.date.desc
e

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "dayInterval": "3", "stopOnTaskFailure": "true", "time": "16:25", "enable": "true", "command": "ModifyScheduleExportSettings", "date": "2014-02-14" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

ModifyTransportAndStorageMode

D
e

scription:

Changes the transport and storage mode setting for the collective. When you update the transport and storage mode, a task is created to update the collective settings and all appliances in the collective are restarted.

Required Parameters:

transport Specifies the transport and storage mode. Possible values are ORB, XIO, XIO_XM. Transport and storage modes enable the exchange of objects and data between different server processes. When you enable IBM eXtremeIO (XIO), relative response time is faster than the Object Request Broker (ORB). ORB is deprecated. With XIO enabled, you can also create an enterprise data grid. With an enterprise data grid, client applications that are written in different programming languages, such as Java and .NET, can access the same data grid. When you use IBM eXtremeMemory, cache entries are stored in native memory. When you use heap memory, cache entries are stored in the Java heap.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "command": "ModifyTransportAndStorageMode", "mode": "XIO" } }
```

Command Type:

collective

ModifyUserEmail

Description:

Changes the email address for a user.

Required Parameters:

s
t

**o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

e
m
a
i
l
A
d
d
r
e
s
s

u
s
e
r
N
a
m
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies a new email address for the user.

Specifies the user name of the user to update.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes**

A JSON-formatted statement that contains the

ult result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ModifyUserEmail", "emailAddress": "user1@mycompany.com",  
"userName": "user1" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

ModifyUserPassword

D
e
s
c
r
i
p
t
i
o
n:

Changes the password for a specified user.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

stopOn Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the

Task Failure batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

new Password Specifies a new password for the selected user.

password Verification Specifies the new password for the selected user, entered a second time for verification.

username Specifies the name of the user to update.

Result Parameters:

errorMessage A message that explains the reason for the failure, if the command failed.

status Specifies the status of the command that was run. The result can be either success or failed.

comma

ndResult A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "newPassword": "newpass",  
"passwordVerification": "newpass", "command":  
"ModifyUserPassword", "userName": "autoUser1003" } }
```

Command Type:

collective

RemoveApplianceFromCollective

Description:

Removes an existing appliance from the current collective.

Required Parameters:

appliance

a n c e i p	Specifies the IP address of the appliance to be removed.
s t o p o n T a s k F a i l u r e	Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.
w a i t O n T a s k	Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

Result Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
comma	

ndName Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "applianceIP": "192.168.222.128", "stopOnTaskFailure":  
"true", "waitOnTask": "true", "command":  
"RemoveApplianceFromCollective" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

RemoveRoleFromGroup

D
e
s
c
r
i
p
t
i
o
n:

Removes an access role from a group. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a**

Specifies the name of the group for which you are removing an access role.

stopOnTaskFailure

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

role

Specifies the role to remove from the group. Valid values: 2 = appliance administration, 3 = appliance monitoring, 5 = data cache creation.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "groupName":  
"autoGroup1003", "command": "RemoveRoleFromGroup", "role": "5"  
} }
```

C

o
m
m
a
n
d
T
y
p
e:

collective

RemoveRoleFromUser

D
e
s
c
r
i
p
t
i
o
n:

Removes an access role from a user. The defined roles are: appliance administration, appliance monitoring, or data cache creation.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**r
o
l
e**

Specifies the role to remove from the user. Valid values: 2 = appliance administration, 3 = appliance monitoring, 5 = data cache creation.

**u
s
e
r
N
a
m
e**

Specifies the name of for which you are removing an access role.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"RemoveRoleFromUser", "role": "2", "userName": "autoUser1003" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

RemoveUserFromGroup

Description:

Removes a user from a group. After a user is removed from a group, the user continues to have the same roles as the group. However, the user roles for the user no longer change when the group roles change. You must modify the user roles for the selected user individually.

Required Parameters:

**g
r
o
u
p
N
a
m
e**

Specifies the name of a defined group.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**u
s
e
r
N**

Specifies the name of a defined user.

ame

Result Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "autoGroup1003", "command": "RemoveUserFromGroup", "userName": "autoUser1003" } }
```

Command Type:

collective

ViewAllGroups

Description:

ip
ti
o
n:

Displays information about all groups.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "command": "ViewAllGroups" } }
```

Command Type:

collective

ViewAllUsers

Description:

Displays the information for every user.

Required Parameters:

s

**t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command": "ViewAllUsers" } }
```

C
o
m
m
a
n
d
T
y
p
e

collective

p
e:

ViewCollectiveLinks

D
e
s
c
r
i
p
t
i
o
n:

Displays the information for every collective link.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

R
e
s
u
l
t
P
a
r
a

m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "command":  
"ViewCollectiveLinks" } }
```

C
o
m
m
a
n
d
T
y
p
e:

collective

ViewGroup

D
e
s
c
r
i
p
t
i
o
n:

Displays information about a selected group.

R
e
q
u
i
r
e
d
P
a

parameters:

**group
Name**

Specifies the name of a defined group.

**stop
OnTask
Failure**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "autoGroup1003", "command": "ViewGroup" } }
```

Command Type:

collective

ViewMemberDetails

Description:

Displays information about a member in a collective.

Required Parameters:

ipAddress

Specifies the IP Address of the member.

**c
e
l
p**

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

- errorMessage** A message that explains the reason for the failure, if the command failed.
- status** Specifies the status of the command that was run. The result can be either success or failed.
- commandResult** A JSON-formatted statement that contains the result of the command that was run.
- commandName** Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "applianceIP": "192.168.222.128", "stopOnTaskFailure": "true", "command": "ViewMemberDetails" } }
```

**C
o**

m
m
a
n
d
T
y
p
e:

collective

ViewTransportAndStorageMode

D
e
s
c
r
i
p
t
i
o
n:

Displays the transport and storage mode.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "command": "ViewTransportAndStorageMode" } }
```

Command Type:

collective

ViewUser

Description:

Displays information about a selected user.

Required Parameters:

s

**t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

u
s
e
r
N
a
m
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Specifies the name of a defined user.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

**E
x
a
m
p
l
e:**

```
{ "task": { "stopOnTaskFailure": "true", "command": "ViewUser",  
"userName": "autoUser1003" } }
```


C
o
m
m
a
n
d
T
y
p
e:

collective

List of GRID commands

ClearGrid

D
e
s
c
r
i
p
t
i
o
n:

Clears data from a data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

gridName

Specifies the name of the data grid to be cleared.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"command": "ClearGrid" } }
```

Command Type:

grid

CreateGrid

D
e
s
c
r
i
p
t
i
o
n:

Creates a new simple, dynamic cache, or session data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

g
r
i
d
N
a
m
e

Specifies the name of the data grid to create.

w
a
i
t
O
n
T
a
s
k

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

k

gridType

Specifies the type of data grid to create. Valid values: simple, dynamic, or session.

Result Parameters:

errorMessage

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

commandResult

A JSON-formatted statement that contains the result of the command that was run.

commandName

Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"waitOnTask": "true", "command": "CreateGrid", "gridType": "simple"  
} }
```

Command Type:

grid

DeleteGrid

D
e
s
c
r
i
p
t
i
o
n:

Deletes a data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

g
r
i
d
N
a
m
e

Specifies the name of the data grid to delete.

w
a
i
t
O
n
T
a
s
k

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

k

Result Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"command": "DeleteGrid" } }
```

Command Type:

grid

ExportGrid

Description:

(Simple data grids only) Exports data grid information to XML.

o
n:
R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to export.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",
"command": "ExportGrid" } }
```

Command Type:

grid

GrantGroupAccess

Description:

Gives a group access rights to a data grid.

Required Parameter:

rs
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of the group to grant the given access rights.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to grant access rights.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

comma

A JSON-formatted statement that contains the

ndResult result of the command that was run.
commandName Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "somegroup",  
"gridName": "myGrid", "command": "GrantGroupAccess" } }
```

Command Type:

grid

GrantUserAccess

Description:

Gives a user access rights to a data grid.

Required Parameters:

stop

**O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to which the user is assigned access.

**u
s
e
r
N
a
m
e**

Specifies the name of the user to give access rights.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

**errorM
essage**

A message that explains the reason for the failure, if the command failed.

status

Specifies the status of the command that was run. The result can be either success or failed.

**comma
ndRes
ult**

A JSON-formatted statement that contains the result of the command that was run.

**comma
ndNam
e**

Specifies the name of the HTTP command interface command that was run.

E

X
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"command": "GrantUserAccess", "userName": "someuser" } }
```

C
o
m
m
a
n
d
T
y
p
e:

grid

ModifyGridCapacity

D
e
s
c
r
i
p
t
i
o
n:

Changes the capacity for a simple data grid.

R
e
q
u
i
r
e
d
P
a
r
a
m
e
t
e
r
s
:

**s
t
o
p
O
n
T
a
s
k
F**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**ai
lu
r
e**

**u
s
e
L
R
U**

(Simple data grids only) Enables least recently used (LRU) eviction when set to true.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to update.

**w
a
i
t
O
n
T
a
s
k**

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

**g
r
i
d
C
a
p
a
c
i
t
y
L
i
m
i
t**

Specifies the maximum capacity for the selected data grid in megabytes.

**R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:**

errorMessage

A message that explains the reason for the failure, if the command failed.

status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "useLRU": "false",
"gridName": "myGrid", "waitOnTask": "true", "command":
"ModifyGridCapacity", "gridCapLimit": "10" } }
```

Command Type:

grid

ModifyGridReplication

Description:

Changes the replication settings for a data grid.

Required Parameters:

st

opOnTaskFailure Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

gridName Specifies the name of the data grid to update.

maximumSyncReplicas Specifies the maximum number of synchronous replicas for this data grid.

waitOnTask Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

maximumAsyncreplicas Specifies the maximum number of asynchronous replicas for this data grid.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorM

message	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",
"waitOnTask": "true", "maximumSyncReplicas": "4", "command":
"ModifyGridReplication", "maximumAsyncReplicas": "2" } }
```

Command Type:

grid

ModifyGridSecurity

Description:

Changes the security settings for a data grid.

Required Parameters:

rs
:

stopOnTaskFailure
authorizationEnabled

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

Enables authorization for the data grid when set to true.

gridName
waitOnTask

Specifies the name of the data grid to update.

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

securityEnabled

Enables security for the data grid when set to true.

R
e
s
u
l
t

Parameters:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

Example:

```
{ "task": { "stopOnTaskFailure": "true", "authorizationEnabled": "false", "gridName": "myGrid", "waitOnTask": "true", "command": "ModifyGridSecurity", "securityEnabled": "false" } }
```

Command Type:

grid

ModifyGridTTL

Description:

(Simple data grids only) Changes time to live (TTL) settings for the data grid.

Requirements:

r
e
d
P
a
r
a
m
e
t
e
r
s
:

**de
fa
ul
tM
ap
Ev
ict
or
Ty
pe**

(Optional) Specifies the time to live eviction type used for the default map. Valid values: NONE, CREATION_TIME, LAST_ACCESS_TIME, LAST_UPDATE_TIME.

**st
op
O
nT
as
kF
ail
ur
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**gr
id
N
a
m
e**

Specifies the name of the data grid to update.

**w
ait
O
nT
as
k**

Specifies whether to wait on the completion of the task associated with the command. If the value is set to true, wait on the completion of the task. If the value is set to false, do not wait on the completion of the task.

**gr
id
TT
L**

Specifies the amount of time, in seconds, to keep data before evicting the data from data grid.

R
e
s
u
l
t
P
a
r
a

m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "defaultMapEvictorType": "CREATION_TIME",  
"stopOnTaskFailure": "true", "gridName": "myGrid", "waitOnTask":  
"true", "command": "ModifyGridTTL", "gridTTL": "60000" } }
```

C
o
m
m
a
n
d
T
y
p
e:

grid

ModifyGroupAccess

D
e
s
c
r
i
p
t
i
o
n:

Changes the group access rights for a data grid.

R
e
q
u
i
r
e
d
P

a
r
a
m
e
t
e
r
s
:

**g
r
o
u
p
N
a
m
e**

Specifies the name of the group for which you want to modify access rights.

**s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e**

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

**g
r
i
d
N
a
m
e**

Specifies the name of the data grid to update.

**a
c
c
e
s
s
T
y
p
e**

Specifies the type of access to give to the group.
Valid values: 1 = read, 2 = write, 3 = create, 4 = all.

R
e
s
u
l

t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "groupName": "somegroup",  
"gridName": "myGrid", "command": "ModifyGroupAccess",  
"accessType": "1" } }
```

C
o
m
m
a
n
d
T
y
p
e:

grid

ModifyUserAccess

D
e
s
c
r
i
p
t
i
o
n:

Changes user access rights for a data grid.

R
e
q

u
r
e
d
P
a
r
a
m
e
t
e
r
s
:

s
t
o
p
O
n
T
a
s
k
F
a
i
l
u
r
e

Specifies whether to stop running the batch routine when the task fails. If the value is set to true, the batch routine stops. If the value is false or blank, the batch routine does not stop. Applies to commands that are run in batch processes only.

g
r
i
d
N
a
m
e

Specifies the name of the data grid to update.

u
s
e
r
N
a
m
e

Specifies the name of the user for which access rights are being updated.

a
c
c
e
s
s
T
y
p
e

Specifies the type of access to give to the user. Valid values: 1 = read, 2 = write, 3 = create, 4 = all.

R
e
s
u
l
t
P
a
r
a
m
e
t
e
r
s
:

errorMessage	A message that explains the reason for the failure, if the command failed.
status	Specifies the status of the command that was run. The result can be either success or failed.
commandResult	A JSON-formatted statement that contains the result of the command that was run.
commandName	Specifies the name of the HTTP command interface command that was run.

E
x
a
m
p
l
e:

```
{ "task": { "stopOnTaskFailure": "true", "gridName": "myGrid",  
"command": "ModifyUserAccess", "userName": "someuser",  
"accessType": "1" } }
```

C
o
m
m
a
n

クライアント・プロパティ・ファイル

WebSphere® eXtreme Scale クライアント・プロセスの要件に基づいて、プロパティ・ファイルを作成します。

サンプル・クライアント・プロパティ・ファイル

Java `wxs_home/properties` ディレクトリーにある `sampleClient.properties` ファイルを使用して、プロパティ・ファイルを作成できます。

.NET `net_client_home¥config¥` ディレクトリーにある `Client.Net.properties` ファイルを使用して、プロパティ・ファイルを作成することができます。

Java

Java クライアントのクライアント・プロパティ・ファイルの指定

クライアント・プロパティ・ファイルは、以下のいずれかの方法で指定できます。このリスト内の項目のいずれかを使用して後で設定を指定すると、前の設定はオーバーライドされます。例えば、クライアント・プロパティ・ファイルのシステム・プロパティ値を指定すると、そのファイルのプロパティにより、クラスパスにある `objectGridClient.properties` ファイルの値が指定変更されます。

1. クラスパス内の任意の場所にあるわかりやすい名前のファイルで指定。このファイルをシステム現行ディレクトリーに置くことはサポートされていません。

```
objectGridClient.properties
```

2. スタンドアロンまたは WebSphere Application Server 構成のいずれかでシステム・プロパティとして指定。この値に、システム現行ディレクトリー内のファイルを指定することはできますが、クラスパス内のファイルは指定できません。

```
-Dobjectgrid.client.props=file_name
```

注: WebSphere Application Server 構成で、特定のクライアント・プロパティ・ファイルをシステム・プロパティで使いたい場合は、クライアント・プロパティ・ファイル (プロパティ・ファイルを特定のプロファイルに適用されたいのか、またはインストール済み環境全体に適用したいのか) に応じて、`was_root/properties` または `profile_root/properties` など) をクラスパスに置く必要があります。

3. `ClientClusterContext.getClientProperties` メソッドを使用したプログラマチックな指定変更。オブジェクトのデータに、プロパティ・ファイルからのデータが取り込まれます。セキュリティ・プロパティは、このメソッドで構成できません。

.NET

WebSphere eXtreme Scale Client for .NET のクライアント・プロパティ・ファイルの指定

デフォルトのクライアント・プロパティ・ファイルは `net_client_home¥config¥Client.Net.properties` ディレクトリーにあります。独自のプロパティ・ファイルを指定したい場合は、各 `Connect` メソッド呼び出しへの絶対プロパティ・ファイル・パスを指定します。

```
IGridManager gm = GridManagerFactory.GetGridManager( );
ICatalogDomainInfo cdi = gm.CatalogDomainManager.CreateCatalogDomainInfo( "localhost:2809" );
ccc = gm.Connect( cdi, @"C:¥MyLocation¥MyClient.properties" );
grid = gm.GetGrid( ccc, gridName );
```

クライアント・プロパティ

クライアント・プロパティ

.NET **2.5+** `enableDynamicConfiguration`

これが `true` に設定されると、クライアント・プロパティ・ファイル内の `requestRetryTimeout` プロパティに対して行われた変更が動的に検出されます。新しい要求再試行タイムアウト値を計算するために新しいプロパティ値がただちに使用されます。

デフォルト: `false`

Java

listenerHost

オブジェクト・リクエスト・ブローカー (ORB) または eXtremeIO (XIO) トランスポート・プロトコルが通信用にバインドするバインド先のホスト名を指定します。この値は、完全修飾ドメイン名または IP アドレスである必要があります。ご使用の構成に複数のネットワーク・カードが含まれている場合は、JVM 内のトランスポート・メカニズムがソバ

インド先の IP アドレスを知ることができるようにリスナー・ホストとリスナー・ポートを設定してください。使用する IP アドレスを指定しなければ、接続タイムアウトや異常な API 障害、クライアントがハングしたように見える状態などの症状が現れることがあります。

Java listenerPort

オブジェクト・リクエスト・ブローカーまたは eXtremeIO (XIO) トランスポート・プロトコルが通信用にバインドするバインド先のポート番号を指定します。listenerPort に対して定義されたポート番号は、クライアントとカタログ・サーバーの間の通信、および同じドメイン内にあるコンテナ・サーバーとカタログ・サーバーの間の通信に使用されます。また、ドメイン間およびドメイン内のカタログ・サーバー間通信にも使用されます。

デフォルト: 2809

注: WebSphere Application Server 内でデータ・グリッド・サーバーが実行され、ORB トランスポート・プロトコルが使用されている場合は、別のポート ORB_LISTENER_ADDRESS も開く必要があります。BOOTSTRAP_ADDRESS ポートは要求をこのポートに転送します。XIO トランスポート・プロトコルを使用している場合は、XIO_ADDRESS ポートを開く必要があります。

Java preferLocalProcess

このプロパティは、現在使用されていません。将来の使用のために予約済みです。

Java preferLocalHost

このプロパティは、現在使用されていません。将来の使用のために予約済みです。

Java .MET preferZones

優先ルーティング・ゾーンのリストを指定します。各ゾーンは、preferZones=ZoneA,ZoneB,ZoneC の形式でコマンドで区切って指定します。

デフォルト: 値なし

Java .MET requestRetryTimeout

例外が発生してから要求の処理をいつまで続行するかをミリ秒で指定します。以下の有効値のいずれかを使用します。

- 値が 0 の場合、要求は、フェイル・ファーストになり、内部の再試行ロジックは飛ばされます。
- 値が -1 の場合、要求再試行のタイムアウトは設定されません。すなわち、要求の所要時間は、トランザクション・タイムアウトによって制御されます。(デフォルト)
- 0 より大きい値は、要求再試行のタイムアウト値をミリ秒で示しています。正常に作成されない例外が返されます。DuplicateException などの例外が再試行されるときでも、成功しないときはその例外も返されます。トランザクション・タイムアウトは、最大待ち時間として引き続き使用されます。

Java .MET shuffleBootstrapAddresses

データ・グリッドへのブートストラップ時にクライアントによってカタログ・サービス・アドレスが使用されるとき、カタログ・サービス・アドレスをランダム化するかどうかを指定します。値は true または false のいずれかでなければなりません。

デフォルト: true

Java 2.5+ xioTimeout

eXtremeIO によるアウトバウンド・ソケット接続試行のタイムアウト値を秒で指定します。これは、内部 eXtremeIO ロジックのデフォルト・タイムアウトとしても使用されます。

Java 2.5+ xioRequestTimeout

要求が応答を待つ秒数を指定します (この秒数を過ぎると待つのを止めます)。このプロパティは、ネットワーク停止の障害が発生した場合にクライアントがフェイルオーバーするまでに要する時間に影響します。このプロパティの値を極端に低く設定すると、要求が誤ってタイムアウトになる可能性があります。不用意なタイムアウトを回避するためにこのプロパティの値は慎重に考慮してください。

セキュリティー・クライアント・プロパティ

一般セキュリティー・プロパティ

Java .MET securityEnabled

WebSphere eXtreme Scale クライアント・セキュリティーを使用可能にします。この設定は、WebSphere eXtreme Scale サーバー・プロパティ・ファイルの securityEnabled 設定と一致している必要があります。設定が一致しないと、データ・グリッドへのクライアント接続は失敗します。

デフォルト: false

資格情報認証の構成プロパティ

Java .MET credentialAuthentication

クライアントの資格情報認証のサポートを指定します。以下の有効値のいずれかを使用します。

- Never: クライアントは、資格情報認証をサポートしません。
- Supported: サーバーも資格情報認証をサポートする場合に、クライアントは資格情報認証をサポートします。(デフォルト)
- Required: クライアントは資格情報認証を必要とします。

Java .NET authenticationRetryCount

資格情報の有効期限が切れている場合に認証を試行できる回数を指定します。この値が 0 に設定されていると、認証の再試行はできません。

デフォルト: 3

.NET credentialGeneratorAssembly

WebSphere eXtreme Scale Client for .NET の資格情報を生成するために使用されるアセンブリーの名前を指定します。このプロパティを指定するためには、credentialAuthentication プロパティを Supported または Required に設定する必要があります。値は、バージョン、国別情報その他のプロパティが含まれている有効な C# .dll アセンブリー名でなければなりません。

例: IBM.WebSphere.Caching.CredentialGenerator, Version=8.6.0.0, Culture=neutral, PublicKeyToken=b439a24ee43b0816

Java .NET credentialGeneratorClass

com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator インターフェースを実装するクラスの名前を指定します。このプロパティを指定するためには、credentialAuthentication プロパティを Supported または Required に設定する必要があります。このクラスを使用して、クライアントの資格情報が取得されます。

デフォルト: 値なし

Java .NET credentialGeneratorProps

CredentialGenerator 実装クラスの プロパティを指定します。このプロパティが、setProperty(String) メソッドを使用してオブジェクトに設定されます。このプロパティを指定するためには、credentialAuthentication プロパティを Supported または Required に設定する必要があります。credentialGeneratorProps 値は、credentialGeneratorClass プロパティの値が非ヌルの場合にのみ使用されます。

Transport Layer Security の構成プロパティ

Java .NET transportType

クライアントのトランスポート・タイプを指定します。可能な値は以下のとおりです。

- TCP/IP: クライアントが、TCP/IP 接続のみをサポートすることを示します。
- SSL-Supported: クライアントが TCP/IP 接続と Secure Sockets Layer (SSL) 接続の両方をサポートすることを示します。(デフォルト)
- SSL-Required: クライアントが SSL 接続を必要とすることを示します。

SSL 構成プロパティ

Java 別名 (alias)

鍵ストア内の別名を指定します。鍵ストアに複数の鍵ペア証明書があり、いずれか 1 つの証明書を選択したい場合は、このプロパティを使用します。

デフォルト: 値なし

Java contextProvider

トラスト・サービスのコンテキスト・プロバイダーの名前を指定します。有効でない値を指定すると、コンテキスト・プロバイダー・タイプが正しくないことを示すセキュリティー例外が発生します。

有効値: IBMJSSE2、IBMJSSE、IBMJSSEFIPS など。

Java keyStore

鍵ストア・ファイルへの完全修飾パスを指定します。

例:

etc/test/security/client.private

Java keyStoreType

鍵ストアのタイプを示します。有効でない値を指定すると、ランタイム・セキュリティー例外が発生します。

有効値: JKS、JCEK、PKCS12 など。

Java .NET プロトコル

クライアントに使用するセキュリティー・プロトコルのタイプを指定します。このプロトコルの値は、使用している

セキュリティー・プロバイダーに基づいて設定します。有効でない値を指定すると、プロトコル値が正しくないことを示すセキュリティー例外が発生します。

Java 有効値: SSL、SSLv3、TLS、TLSv1 など。

.NET 有効な値: SSLv2、SSLv3、TLS、または Default (SSLv3 または TLS1.0)

.NET publicKeyFile

サーバーからエクスポートされた公開鍵が入っているファイルへの完全修飾パス名を指定します。

例: c:\tmp\wxs\serverA.cer

Java trustStoreType

トラストストアのタイプを示します。有効でない値を指定すると、ランタイム・セキュリティー例外が発生します。

有効値: JKS、JCEK、PKCS12 など。

Java trustStore

トラストストア・ファイルへの完全修飾パスを指定します。

例:

etc/test/security/server.public

Java keyStorePassword

鍵ストアのストリング・パスワードを指定します。この値はエンコードすることも、実際の値を使用することもできます。

Java trustStorePassword

トラストストアのストリング・パスワードを指定します。この値はエンコードすることも、実際の値を使用することもできます。

親トピック: [参照](#)

関連概念:

[IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)

関連タスク:

[クライアント・セキュリティーの構成](#)

.NET [WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)

関連情報:

[ICredential インターフェース](#)

[ICredentialGenerator インターフェース](#)

正規表現の構文

正規表現は、他の文字列とのマッチングに使用される構造化文字列です。正規表現は、データを無効化する場合およびメッセージに表示されるメッセージをフィルタリングする場合に使用することができます。

正規表現の例

正規表現	説明
.	任意の 1 文字を検索します。
.*	任意の 1 文字を 0 回以上検索します。
A.*	「A」で始まるすべての文字列を検索します。
.*A	「A」で終わるすべての文字列を検索します。
A?	「A」を 0 回または 1 回検索します。
A*	「A」を 0 回または複数回検索します。
A+	「A」を 1 回または複数回検索します。
A?B*C+	後に 0 個または 1 個の A が続き、その後に 0 個または複数個の B が続き、その後に 1 個または複数個の C が続く任意の 1 文字を検索します
[ABC].*	「A」、「B」、「C」のいずれかの文字で始まる文字列を検索します。
.*ABC.*	文字列のどこかに「ABC」という文字列を含む文字列を検索します。
(ABC DEF).*	文字列「ABC」または「DEF」で始まる文字列を検索します。

詳しい正規表現の構文については `java.util.regex.Pattern` API の資料を参照してください。

親トピック: [参照](#)

動的マップの構成オプション

ご使用のクライアント・アプリケーションを特定の名前の付いたマップに接続させることで、データ・グリッド内に追加のマップを作成できます。この接続が発生すると、マップが自動的に作成されます。

動的マップの命名

データ・グリッドを作成すると、デフォルトでは2つのマップが作成されます。最初のマップには、ご使用のデータ・グリッドと同じ名前が付けられます。例えば、myGrid というデータ・グリッドを作成すると、自動的に myGrid というマップが作成されます。2番目のマップはニア・キャッシュ用で、myGrid.NCI という名前が付けられます。しかし、データ・グリッドにマップを追加することも可能です。クライアント・アプリケーションが次の命名規則を使用したマップに接続する際、1つのマップが自動作成されます。

```
<map_name>.<template>.<locking_option>.<near_cache_invalidation>
```

各部の意味は、次のとおりです。

map_name (必須)

マップの名前を指定します。

template (必須)

存続時間 (TTL) の動作を定義することで、エントリーの有効期限がマップから切れる時期を定義するテンプレートを指定します。使用できるオプションのリストについては、[マップ・テンプレート](#) を参照してください。

locking_option

マップに使用されるロック・メカニズムを指定します。使用できるオプションのリストについては、[ロック・オプション](#) を参照してください。

2.5+ near_cache_invalidation

ニア・キャッシュの無効化をマップに使用するかを指定します。ニア・キャッシュの無効化は、マップで、ロック・タイプを NONE または OPTIMISTIC に設定している場合にのみ使用されます。できるだけ速やかにニア・キャッシュから失効データが削除されるようにニア・キャッシュの無効化を構成することができます。リモート・データ・グリッドに対して更新、削除、または無効化操作が実行されると、ニア・キャッシュで非同期無効化がトリガーされます。使用可能なオプションのリストは、[ニア・キャッシュ無効化オプション](#) を参照してください。

マップのテンプレート名を含める必要があります。ロック・オプションを指定しなければ、マップ上でロックは発生しません。

マップ・テンプレート

表 1. 動的マップ・テンプレート

マップ・テンプレート	説明
*.NONE	存続時間 (TTL) に有効期限がないマップを指定します。
*.LUT	項目の最終更新時間に基づいて項目の有効期限が切れる マップを指定します。デフォルトの TTL は 1 時間です。
*.LAT	項目の最終アクセス時間に基づいて項目の有効期限が切れるマップです。デフォルトの TTL は 1 時間です。
*.CT	エントリーの作成時間と TTL 値に基づいてエントリーの有効期限が切れるマップです。デフォルトの TTL は 1 時間です。

ロック・オプション

表 2. 動的マップのロック・オプション

ロック・オプション	説明
(ブランク)	ロック・オプションを指示しなければ、ロック・メカニズムは使用されません。
.P	マップにペシミスティック・ロックがあるよう指定します
.O	マップにオプティミスティック・ロックがあることを指定します

2.5+

ニア・キャッシュ無効化オプション

表 3. ニア・キャッシュ無効化オプション

ロック・オ	
-------	--

プシヨン	説明
(プランク)	値を指定しないと、ニア・キャッシュ無効化は使用不可に設定されます。
.NCI	マップでニア・キャッシュ無効化を使用することを指定します。お使いのマップで、オプティミスティック・ロックまたは非ロック式が使用可能に設定されている必要があります。

親トピック: [参照](#)

関連概念:

[Evictor](#)

関連タスク:

[存続時間 \(TTL\) Evictor の構成](#)

[動的マップの構成](#)

[Java 2.5+ ニア・キャッシュ無効化の構成](#)

[REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)

[単純データ・グリッドの作成](#)

[.NET .NET API を使用した動的マップの作成](#)

CLI コマンド解説書

このページは、WebSphere® DataPower® XC10 アプライアンスにより提供されるコマンド行インターフェース (CLI) を記載しています。以下の 3 セットのコマンドがサポートされています。

- 様々な構成オブジェクトを処理する共通コマンド。 [共通コマンド](#) を参照してください。
- 特殊目的コマンド。 [特殊目的コマンド](#) を参照してください。
- フェイルセーフ・モードで使用可能なコマンド。

共通コマンド

このセクションに記載されているコマンドは、様々な構成オブジェクトを処理するために使用されます。共通コマンドは、指定された構成オブジェクトを処理するか、または構成オブジェクトの編集セッション内で使用されます。編集セッションとは、構成オブジェクトの値を変更できる CLI のモードです。編集セッションは「edit」または「create」コマンドで始まり、「exit」または「cancel」コマンドで終わります。これらのコマンドの一部、例えば「append」や「set」などは、編集セッション内でのみ使用可能です。

現在サポートされている構成オブジェクトは以下のとおりです。

- aggregate-interface
- ethernet-interface
- ldap-auth
- logging-settings
- netsec-settings
- network-settings
- vlan-interface

編集セッション内で機能するコマンドには、以下のコマンドが含まれます。

[**cancel**](#)

[**create**](#)

[**delete**](#)

[**disable**](#)

[**domain list**](#)

[**domain show**](#)

[**domain switch**](#)

[**edit**](#)

[**enable**](#)

[**exit**](#)

[**list**](#)

[**reset**](#)

[**show**](#)

[**show properties**](#)

[**show types**](#)

[**status**](#)

親トピック: [CLI コマンド解説書](#)

cancel

目的

編集セッション内で、編集セッション中またはネストされた編集セッション中に行われた構成オブジェクトへの変更すべてを廃棄します。

構文

cancel

パラメーター

なし。

使用上の注意

ネストされた編集セッションでは、**cancel** は編集サブモード内で行われた変更のみを廃棄します。

関連コマンド

[exit](#) を参照してください。

例

編集セッション中に network-settings 構成オブジェクトに行った変更をキャンセルします。

```
Console> edit network-settings
Console network-settings> set tcp-retries 10
Console network-settings> cancel
Cancelled
Console>
```

親トピック: [共通コマンド](#)

create

目的

編集セッションに入り、構成オブジェクトのインスタンスを作成します。

構文

create *object-type object-instance*

パラメーター

object-type

作成されるオブジェクトのタイプ

object-instance

作成されるオブジェクト・インスタンスの名前

使用上の注意

- 一部のオブジェクトは、コマンド行インターフェースで作成できません。これらのオブジェクトには以下のものが含まれます。
 - シングルトン・オブジェクトと呼ばれる、インスタンスが1つだけのオブジェクト
 - ethernet-interface オブジェクトなど、事前定義された有限のセットのあるオブジェクト
- デフォルトでは、新規作成されたオブジェクトの AdminState は使用可能になっています。

関連コマンド

[delete](#)、[edit](#)、および [list](#) を参照してください。

例

v1000 という名前の vlan-interface を作成します。

```
Console> create vlan-interface v1000
Console vlan-interface:v1000>
...
```

親トピック: [共通コマンド](#)

delete

目的

構成オブジェクトのインスタンスを削除します。

構文

delete *object-type object-instance*

パラメーター

object-type

削除されるオブジェクトのタイプ

object-instance

削除されるオブジェクト・インスタンスの名前

使用上の注意

一部のオブジェクトは削除できません。これらのオブジェクトには以下のものが含まれます。

- シングルトン・オブジェクトと呼ばれる、インスタンスが 1 つだけのオブジェクト
- ethernet-interface オブジェクトなど、事前定義された有限のセットのあるオブジェクト

関連コマンド

[create](#)、[edit](#)、および [list](#) を参照してください。

例

v1000 という名前の vlan-interface を削除します。

```
Console> delete vlan-interface v1000
Console>
```

親トピック: [共通コマンド](#)

disable

目的

- 構成オブジェクト、または構成オブジェクトのインスタンスを使用不可に設定します。
- このコマンドは、管理状態 (AdminState) を「disabled」に変更します。エラーが発生しなければ、操作状態 (OpState) が「down」に変わります。
- このコマンドの結果、他の構成値は変更されません。

構文

disable *object-type* [*object-instance*]

パラメーター

object-type

使用不可に設定されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、使用不可に設定されるオブジェクト・インスタンスの名前

使用上の注意

構成変更は管理上使用不可のオブジェクトにも行えますが、オブジェクトが使用可能になるまでは変更は適用されません。

関連コマンド

[enable](#) を参照してください。

例

v1000 という名前の vlan-interface を使用不可に設定します。

```
Console> disable vlan-interface v1000
Console>
```

親トピック: [共通コマンド](#)

domain list

目的

サポートされるドメインを表示します。

構文

domain list

パラメーター

なし。

使用上の注意

現在は 1 つのドメイン「default」のみがサポートされています。

関連コマンド

[domain show](#) および [domain switch](#) を参照してください。

例

現在サポートされているドメインを表示します。

```
Console> domain list
default
Console>
```

親トピック: [共通コマンド](#)

domain show

目的

現在アクティブなドメインを表示します。

構文

domain show

パラメーター

なし。

使用上の注意

現在は 1 つのドメイン「default」のみがサポートされています。

関連コマンド

[domain list](#) および [domain switch](#) を参照してください。

例

現在アクティブなドメインを表示します。

```
Console> domain show
In domain "default"
Console>
```

親トピック: [共通コマンド](#)

domain switch

目的

現在アクティブなドメインを変更します。

構文

```
domain switch domain
```

パラメーター

domain

要求されるアクティブ・ドメイン

使用上の注意

現在は 1 つのドメイン「default」のみがサポートされています。

関連コマンド

[domain list](#) および [domain show](#) を参照してください。

例

現在アクティブなドメインを変更します。

```
Console> domain switch default
Switched to "default" domain
Console>
```

親トピック: [共通コマンド](#)

edit

目的

構成モードに入り、構成オブジェクト、または構成オブジェクトのインスタンスを変更します。

構文

edit *object-type* [*object-instance*]

パラメーター

object-type

編集されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、編集されるオブジェクト・インスタンスの名前

使用上の注意

edit コマンドによって、構成されたオブジェクトの編集セッションに入ります。この編集セッション内では、**edit** サブコマンドにより、構成されたオブジェクトのプロパティの値を変更することができます。

関連コマンド

[create](#)、[delete](#)、および [list](#) を参照してください。

例

ethernet-interface eth3 を編集して MTU を 4096 に設定します。

```
Console> edit ethernet-interface eth3
Console ethernet-interface:eth3> set mtu 4096
Console ethernet-interface:eth3> exit
Console>
```

親トピック: [共通コマンド](#)

enable

目的

- 構成オブジェクト、または構成オブジェクトのインスタンスを使用可能に設定します。
- このコマンドは、管理状態 (AdminState) を「enabled」に変更します。エラーが発生しなければ、操作状態 (OpState) が「up」に変わります。
- このコマンドの結果、他の構成値は変更されません。

構文

enable *object-type* [*object-instance*]

パラメーター

object-type

使用可能に設定されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、使用可能に設定されるオブジェクト・インスタンスの名前

関連コマンド

[disable](#) を参照してください。

例

v1000 という名前の vlan-interface を使用可能に設定します。

```
Console> enable vlan-interface v1000
Console>
```

親トピック: [共通コマンド](#)

exit

目的

編集モードまたは編集サブモードを終了します。

構文

exit

パラメーター

なし。

使用上の注意

構成変更は、編集サブモードの終了時ではなく、編集セッションの終了時に保存されます。このため、検証エラーが予想より遅く表示されることがあります。

関連コマンド

[cancel](#) を参照してください。

例

ethernet-interface オブジェクトの編集セッションに入り、ip サブモードに入ります。IP アドレスを追加し、その後 ip サブモードを終了し、編集セッションを終了します。

```
Console> edit ethernet-interface eth2
Console ethernet-interface:eth2> ip
Entering "ip" mode
Console ethernet-interface:eth2 ip> append address 192.168.0.200/24
Console ethernet-interface:eth2 ip> exit
Console ethernet-interface:eth2> exit
Console>
```

親トピック: [共通コマンド](#)

list

目的

指定されたオブジェクト・タイプの構成されたオブジェクトの名前を表示します。

構文

list *object-type*

パラメーター

object-type

リストされるオブジェクトのタイプ

使用上の注意

オブジェクト・タイプのリストを取得するには、[show types](#) を使用してください。

関連コマンド

[create](#)、[delete](#)、[edit](#)、および [show types](#) を参照してください。

例

ethernet-interfaces のすべてのインスタンスをリストします。

```
Console> list ethernet-interface
    eth0
    eth1
    eth2
    eth3
Console>
```

親トピック: [共通コマンド](#)

reset

目的

構成オブジェクト、または構成オブジェクトの一部をデフォルト設定にリストアします。

構文

- シングルトン構成オブジェクトをリセットする: **reset object-type**
- 構成オブジェクトのインスタンスをリセットする: **reset object-type object-instance**
- 編集セッション内の構成オブジェクトのフィールドをリセットする: **reset property**

パラメーター

object-type

リセットされるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、リセットされるオブジェクトのインスタンス

property

リセットされる構成オブジェクト内のフィールド

例

- ethernet-interface eth3 の IP アドレスをリセットします。

```
Console> edit ethernet-interface eth3
Console ethernet-interface:eth3> ip
Entering "ip" mode
Console ethernet-interface:eth3 ip> reset address
Console ethernet-interface:eth3 ip> exit
Console ethernet-interface:eth3> exit
Console>
```

- ethernet-interface eth3 の構成オブジェクト全体をデフォルト値にリセットします。

```
Console> reset ethernet-interface eth3
Console>
```

親トピック: [共通コマンド](#)

show

注: **show** コマンドは、構成オブジェクトとその他の特定データの両方を表示するのに使用されます。このセクションは、構成オブジェクトに対する **show** コマンドの使用について記載しています。[特殊目的コマンド](#) セクション内の **show** コマンドも参照してください。

目的

指定された構成オブジェクト、または構成オブジェクトのインスタンスに構成されている値を表示します。

構文

```
show [ object type [ object instance ] ]
```

パラメーター

object type

表示されるオブジェクトのタイプ

object instance

構成オブジェクトがシングルトン・オブジェクトでない場合、表示されるオブジェクト・インスタンスの名前

使用上の注意

- 編集セッション内では、**show** コマンドは編集集中のオブジェクト・インスタンスで現在構成されている値を表示します。
- オブジェクトの操作状態も表示されます。必須のプロパティと、割り当てられたデフォルト値と異なる値が構成されたプロパティのみが表示されます。明示的に構成されていないオプションのプロパティは表示されません。
- ここに表示される値は、オブジェクトに要求される構成です。対照的に、[status](#) コマンドはオブジェクトの実際のランタイム値を表示します。

例

network-settings (シングルトン) オブジェクト用に構成された値を表示します。

```
Console> show network-settings
network-settings : [Up]

name "network-settings"
AdminState "Enabled"
icmp-options " "
explicit-congestion-notification "false"
destination-based-routing "false"
interface-isolation "relaxed"
tcp-retries "5"
arp-retries "8"
arp-retry-interval "500"
tso-offload "true"
reverse-path-filtering "false"
tcp-window-scaling "true"
```

親トピック: [共通コマンド](#)

show properties

目的

編集セッション内で、オブジェクトの構成可能プロパティを表示します。

構文

show properties

パラメーター

なし。

使用上の注意

ネストされたレベルのプロパティを表示するには、**edit** サブモードに入り、コマンドを繰り返します。

例

ethernet-interface の構成可能プロパティを表示します。

```
Console> edit ethernet-interface eth3
Console ethernet-interface:eth3> show properties
Basic properties:
  AdminState
  mac-address
  mode
  mtu
  use-arp
  userdata

Sub-mode (complex) properties:
  ip

Console ethernet-interface:eth3> ip
Entering "ip" mode
Console ethernet-interface:eth3 ip> show properties
Basic properties:
  dad-retransmit-timer
  dad-transmits
  use-dhcp
  use-slaac

Array (list) properties:
  address
  static-route

Sub-mode (complex) properties:
  ipv4-default-gateway
  ipv6-default-gateway

Console ethernet-interface:eth3 ip>
```

親トピック: [共通コマンド](#)

show types

目的

アプライアンスでサポートされている構成オブジェクトのタイプを表示します。これは構成されたインスタンスを反映せず、サポートされるタイプを示すだけです。

構文

show types

パラメーター

なし。

使用上の注意

なし。

例

アプライアンスの構成オブジェクトを表示します。

```
Console> show types
The following types are defined:
  aggregate-interface
  ethernet-interface
  logging-settings
  network-settings
  vlan-interface
```

親トピック: [共通コマンド](#)

status

注: **status** コマンドは、構成されたオブジェクトの状況と特定の機能状況情報の両方を表示するのに使用されます。このセクションは、構成オブジェクトに対する **status** コマンドの使用について記載しています。[特殊目的コマンド](#) セクション内の status コマンドも参照してください。

目的

指定された構成オブジェクト、または構成オブジェクトのインスタンスの状況を表示します。

構文

```
status object-type [ object-instance ]
```

パラメーター

object-type

表示されるオブジェクトのタイプ

object-instance

構成オブジェクトがシングルトン・オブジェクトでない場合、表示されるオブジェクト・インスタンスの名前

使用上の注意

- シングルトン・オブジェクトでない構成オブジェクトで *object-instance* が省略されている場合、指定された *object-type* のすべてのインスタンスの状況が表示されます。
- [status](#) コマンドは、選択された構成オブジェクトのみでサポートされます。
- ここに表示される値は、オブジェクトのランタイム値です。対照的に、[show](#) コマンドはオブジェクトの構成されたプロパティを表示します。

例

ethernet-interface eth0 の状況を表示します。

```
Console> status ethernet-interface eth0
eth0      OpState:[Up]
          generic MTU:1500 carrier:true flags:UP BROADCAST RUNNING MULTICAST
          index:5
          inet addr:9.42.77.48 flags:PERMANENT mask:255.255.255.0
          scope:GLOBAL
          inet6 addr: 2002:92a:8f7a:901:9:42:77:48 flags:PERMANENT
          mask: ffff:ffff:ffff:ffff:: scope:GLOBAL
          inet6 addr: fe80::20c:29ff:fedd:7baf flags:PERMANENT
          mask: ffff:ffff:ffff:ffff:: scope:LINK
          ethernet Link:on MAC: 00:0c:29:dd:7b:af autoneg:on duplex:Full
          port:TP speed:1000Mbps
          statistics collisions:0 multicast:0 rx_bytes:69088545
          rx_compressed:0 rx_crc_errors:0 rx_dropped:0 rx_errors:0
          rx_fifo_errors:0 rx_frame_errors:0 rx_length_errors:0
          rx_missed_errors:0 rx_over_errors:0 rx_packets:842919
          tx_aborted_errors:0 tx_bytes:3512656 tx_carrier_errors:0
          tx_compressed:0 tx_dropped:0 tx_errors:0 tx_fifo_errors:0
          tx_heartbeat_errors:0 tx_packets:18144 tx_window_errors:0
```

親トピック: [共通コマンド](#)

特殊目的コマンド

このセクションに記載されているコマンドは、アプライアンスで特定のアクションを行うか、特定の情報を表示します。

[add-jvm-args](#)

[alias](#)

[clear-all](#)

[clear-jvm-args](#)

[clear-logs](#)

[clear-tls-config](#)

[collect-logs](#)

[component firmware update](#)

[config](#)

[datetime get](#)

[datetime set](#)

[deleteExport](#)

[device RESET](#)

[device battery-replaced](#)

[device intrusion allow](#)

[device intrusion clear](#)

[device intrusion disallow](#)

[device nvdim clear](#)

[device clear-intrusion](#)

[device restart](#)

[device shutdown](#)

[echo](#)

[export](#)

[file delete](#)

[file get](#)

[file list](#)

[file put](#)

[firmware pristine-install](#)

[firmware rollback](#)

[firmware upgrade](#)

[force recycle](#)

[get-dns-search](#)

[get-dns-servers](#)

[get-ntp-servers](#)

[help](#)

[import](#)

[license accept](#)

[license get](#)

[license reset](#)

[listExports](#)

[locale get](#)

[locale set](#)

[locate-led](#)

[log message](#)

[netif](#)

[net-test available](#)

[net-test dns](#)

[net-test ping](#)

[net-test tcp](#)

[net-test traceroute](#)

[nodename get](#)

[nodename set](#)

[packet-capture clear](#)

[packet-capture list](#)

[packet-capture start](#)

[packet-capture stop](#)

[platform collect-pd](#)

[platform log-level get](#)

[platform log-level set](#)

[platform service disable ssh](#)

[platform service disable telnet](#)

[platform service enable ssh](#)

[platform service enable telnet](#)

[platform must-gather](#)

[raid delete](#)

[raid re-use](#)

[raid foreign-config import](#)

[raid foreign-config clear](#)

2.5+ [request force-suspend](#)

2.5+ [request resume](#)

2.5+ [request suspend](#)

[set-dns-search](#)

[set-dns-servers](#)

[set-ntp-servers](#)

[show commandref](#)

[show components](#)

[show locales](#)

[show log](#)

[show timezones](#)

[show version](#)

[sshkey fingerprint](#)

[sshkey reset](#)

[sshkey verify](#)

[start-progress](#)

[status battery](#)

[status cpu-usage](#)

[status cpu-utilization](#)

[status dynamic-tunnels](#)

[status fan](#)

[status flash](#)

[status intrusion](#)

[status memory](#)

[status nvdimmm](#)

[status power-supply](#)

[status raid all](#)

[status raid battery](#)

[status raid physical](#)

[status voltage](#)

[status volume](#)

[status temperature](#)

[status uptime](#)

[timezone get](#)

[timezone set](#)

[unalias](#)

[usage](#)

[user add](#)

[user delete](#)

[user known-hosts list](#)

[user known-hosts delete](#)

[user list](#)

[user password](#)

[user sshkey add](#)

[user sshkey delete](#)

add-jvm-args

目的

事前設定された JVM 構成パラメーターのセットを、始動中にアプライアンスのプロセスのいずれかに追加します。

構文

add-jvm-args process name listOfOneOrMoreJVMArgs

パラメーター

このコマンドには、以下の 2 つの必須パラメーターがあります。

process name

構成される必要があるプロセスの名前。有効な processName 引数は以下のとおりです。

- all - xsa.app、カタログ・サーバー、およびすべてのコンテナ・サーバーを含む、すべてのプロセス。
- console - xsa.app プロセスのみを含む。
- grids - すべてのコンテナ・サーバー・プロセスのみ。
- gridNN - ここで NN は単一のコンテナ・サーバーを指定する数値 (つまり 01)。

listOfOneOrMoreJVMArgs

プロセスで有効な JVM 引数をリストします。指定可能である有効な JVM 引数は以下のとおりです。

- -Dcom.ibm.CORBA.Debug
- -Dcom.ibm.CORBA.CommTrace
- -Djavax.net.debug
- -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true
- -verbose:gc
- -Xverbosegclog
- -Xhealthcenter
- -Xcheck:jni

注: リストに含めたいパラメーターが複数ある場合は、`!!` で区切ります。(例: `add-jvm-args grid01 -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true!!`)

```
-Dcom.ibm.CORBA.CommTrace add-jvm-args all -Dcom.ibm.CORBA.CommTrace
```

関連コマンド

[device restart](#) を参照してください。

例

```
Console> add-jvm-args grid01 -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true
Updating JVM Args to include new args...
Adding the following JVM Arg: -Dcom.ibm.websphere.xs.ProduceJavaCoreOnLockTimeout=true
[Wed Apr 10 2013 17:00:22] Run device restart to enable the new JVM args settings.
```

親トピック: [特殊目的コマンド](#)

alias

目的

- コマンド行インターフェース (CLI) コマンドの代替名またはショートカットを定義します。
- 構成された別名を表示します。

構文

- 定義された別名のリストを表示する: **alias**
- 既存の別名の定義を表示する: **alias name**
- 指定された CLI コマンドの別名を定義する: **alias name value**

パラメーター

name

コマンド行インターフェース (CLI) コマンドの代替名。

value

指定された名前への代替値。代替値にスペースが含まれる場合は、値を二重引用符で囲みます。

関連コマンド

[unalias](#) を参照してください。

例

特定のアドレスへの接続のテストにショートカットを定義します。それから構成された別名を表示します。

```
Console> alias test-connectivity "net-test ping 9.42.106.2"
Console> alias
alias test-connectivity "net-test ping 9.42.106.2"
Console>
```

親トピック: [特殊目的コマンド](#)

clear-all

目的

アプライアンスの構成データをリセットします (すべてのデータ・グリッドおよび新規ユーザーの削除を含む)。アプライアンスで実行中のすべての内部プロセスは、再始動されます。このコマンドを実行しても、アプライアンス・ハードウェアは再始動されません。

構文

clear-all

パラメーター

なし。

関連コマンド

なし。

例

```
Console> clear-all
Force Stopped all XC-10 processes
Deleting configuration data and logs
Deleting grid data
```

親トピック: [特殊目的コマンド](#)

clear-jvm-args

目的

始動中にアプライアンスに適用されたプロセスのいずれかに対する JVM 構成パラメーターをクリアします。

構文

clear-jvm-argsprocessName

パラメーター

process name

構成される必要があるプロセスの名前。有効な processName 引数は以下のとおりです。

- all - xsa.app、カタログ・サーバー、およびすべてのコンテナ・サーバーを含む、すべてのプロセス。
- console - xsa.app プロセスのみを含む。
- grids - すべてのコンテナ・サーバー・プロセスのみ。
- gridNN - ここで NN は単一のコンテナ・サーバーを指定する数値 (つまり 01)。

関連コマンド

[add-jvm-args](#) を参照してください。

例

```
Console> clear-jvm args all
```

親トピック: [特殊目的コマンド](#)

clear-logs

目的

WebSphere® DataPower® XC10 アプライアンスのログ・ファイルをすべて削除します。

構文

clear-logs

パラメーター

なし。

関連コマンド

なし。

例

```
Console> clear-logs  
Cleared All Logs
```

親トピック: [特殊目的コマンド](#)

clear-tls-config

目的

Transport Layer Security (TLS) 構成をリセットします。すべての構成データを失いたくないので clear-all は選択肢にならない、しかし TLS 構成が破損したかデフォルトの TLS 値をリストアしたいという場合に、このコマンドを実行します。ほとんどの場合には、TLS 変更用のユーザー・インターフェースを使用するか、**clear-all** コマンドを使用して、アプライアンスの元の構成に戻します。集合内の各アプライアンスで、**clear-tls-config** コマンドを実行します。各アプライアンスでコマンドを実行した後、集合内の各アプライアンスの中のプロセスを再始動します。集合が正常に通信している場合、**device restart** コマンドを使用します。集合内のすべてのアプライアンスがユーザー・インターフェースを通してアクセス可能で、「集合」パネルで開始済みであることが確認できる時、集合は正しく通信しています。しかし、TLS 構成によって集合の通信が妨げられていて、**device restart** コマンドでアプライアンスがバックアップしない場合、**force-recycle** コマンドを使用して、データを保存せずにアプライアンスの中のすべてのプロセスを強制的に停止して始動することができます。

構文

clear-tls-config

パラメーター

なし。

関連コマンド

[clear-all](#)、[device restart](#)、および [force recycle](#)。

例

```
Console> clear-tls-config
Cleared Transport Layer Security Configuration
Run clear-tls-config on the other appliances in the collective and restart each appliance
for changes to take effect
```

親トピック: [特殊目的コマンド](#)

collect-logs

このコマンドは推奨されません。代わりに、[platform must-gather](#) を使用してください。

親トピック: [特殊目的コマンド](#)

component firmware update

目的

コンポーネントの現行のファームウェア・バージョンが構成されたバージョンと一致しない場合に、ファームウェア・バージョンを更新します。

構文

component firmware update component type

パラメーター

component type

- コンポーネントのタイプ名。サポートされるコンポーネントのタイプの完全なリストを取得するには、このコマンドのヘルプを参照してください。
- [show components](#) を使用していて、コンポーネントのファームウェア・バージョンが構成されたバージョンと一致しない場合、このコマンドを使用してファームウェア・バージョンを更新できます。

関連コマンド

[show components](#) を参照してください。

例

BMC のコンポーネント・ファームウェアを更新します。

```
Console> component firmware update bmc
CWZBR03003I: Are you sure you want to update component firmware for bmc
After update, device will reboot
CWZBR03012I: yes/no:yes
CWZBR03013I: Component firmware update is starting.....
CWZBR03006I: Component firmware update succeeded and reboot device
```

親トピック: [特殊目的コマンド](#)

config

目的

アプライアンスの構成をインポートまたはエクスポートします。

構文

```
config <export|import|usage|listExports> [-file] <filename> [-keyStore] <keystore>
[-trustStore] <truststore> [-silent] [-noRestart]
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

-truststore

-trustStore フラグの後に、アプライアンスの構成の一部であるトラストストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-keystore

-keyStore フラグの後に、アプライアンスの構成の一部である鍵ストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-noRestart

特定のアプライアンス設定では、アプライアンスを再始動することが要求されます。このフラグを設定すると、アプライアンスはどれも再始動されません。しかし、**-noRestart** パラメーターを設定した場合には、再始動が要求される設定を保存するために、アプライアンスを手動で再始動する必要があります。

-silent

これが指定されると、**config** コマンドの状況メッセージが表示されません。これはデフォルトの動作です。

使用上の注意

なし

関連コマンド

このコマンドには、以下のサブコマンドがあります。[usage](#)、[export](#)、[import](#)、[listExports](#)、および [deleteExport](#) を参照してください。

例

アプライアンスの構成をインポートまたはエクスポートします。

```
config export -file foo.json -silent
config import -file foo.json -keyStore keyStore.jks -trustStore trustStore.jks
config import -file foo.json -noRestart
config usage
config listExports
config deleteExport export_1942-09-06_13-50-14.036.json
```

親トピック: [特殊目的コマンド](#)

datetime get

目的

アプライアンスに構成された時刻を、グリニッジ標準時と地方時の両方で表示します。

構文

datetime get

パラメーター

なし。

関連コマンド

[datetime set](#)、[timezone get](#)、および [timezone set](#) を参照してください。

例

アプライアンスに構成された時刻を表示します。

```
Console> datetime get
GMT: 2012-03-08T20:20:21Z
Local: Mar 8, 2012 3:20:21 PM
Console>
```

親トピック: [特殊目的コマンド](#)

datetime set

目的

アプライアンスに現在時刻を構成します。

構文

datetime set *time*

パラメーター

time

- 地方時 (時間帯設定により決まる) での現在時刻
- 形式は以下の仕様による YYYY-MM-DD hh:mm:ss または YYYY-MM-DD hh:mm:ssZ です。

YYYY

年を示します

MM

月を示します

DD

日付を示します

hh

時を示します

mm

分を示します

ss

秒を示します

Z

時刻に付加してグリニッジ標準時 (GMT) を指定します。

関連コマンド

[datetime get](#)、[timezone get](#)、および [timezone set](#) を参照してください。

例

構成された時間帯を表示し、アプライアンスの現在時刻を設定し、それから構成された時刻を表示します。

```
Console> timezone get
Timezone is EST
Console> datetime set 2012-03-22 10:01:00
Console> datetime get
GMT: 2012-03-22T15:01:03Z
Local: Mar 22, 2012 10:01:03 PM
Console>
```

親トピック: [特殊目的コマンド](#)

deleteExport

目的

指定されたエクスポート・ファイルを、ローカルに保管されたエクスポートのリストから削除します。

構文

```
config deleteExport
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

使用上の注意

deleteExport コマンドを受け渡し、**-file** フラグの付いたファイルを 1 つのみ指定する必要があります。

例

エクスポートされたアプライアンスの構成を削除します。

```
config deleteExport export_1942-09-06_13-50-14.036.json
```

親トピック: [特殊目的コマンド](#)

device RESET

目的

アプライアンスのすべてのデータを消去して、工場出荷状態に戻します。SSH サーバー鍵、ライセンス同意、およびログを含めて、すべての構成が削除されます。すべてのユーザー ID とパスワードも削除され、工場で設定されたデフォルトのユーザー ID とパスワードに置換されます。アプライアンスはこのコマンドを実行した後で自動的にリブートします。

コマンドの「RESET」部分は、間違っって入力してしまう可能性を減らすために、すべて大文字で入力する必要があります。このコマンドは、このデバイスで実行されているファームウェアを変更しないため、**firmware pristine-install**とは異なります。したがって、このコマンドを実行するときにファームウェア・イメージは必要ありません。このコマンドは構成のみを変更します。

構文

device RESET [noprompt]

オプション

noprompt

確認のプロンプトを表示しません。このコマンドはすぐに開始されます。このオプションが存在しない場合、次に表示される確認プロンプトに対してユーザーが肯定的に回答しない限り、コマンドは開始されません。

パラメーター

なし。

使用上の注意

ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、構成がクリアされるたびにネットワーク接続が終了します。このコマンドのすべてのメッセージを表示したい場合には、シリアル・コンソールに接続する必要があります。

関連コマンド

[firmware pristine-install](#) を参照してください。

例

```
Console> device RESET

This command will result in the network configuration being reset; this may end your
current connection to the appliance.

Undo all existing configuration and reset the appliance to factory-new condition? y/n: y
You have chosen to reset the device, proceeding with reset...
Reset complete; ready to reboot
Console>
```

親トピック: [特殊目的コマンド](#)

device battery-replaced

目的

内部バッテリーが交換された日付を記録します。バッテリーは、アプライアンスが電源オフで AC 電源に接続していないときに、ハードウェア構成と RAID ディスク書き込み可能キャッシュにバックアップ電源を提供します。

バッテリーは特定の期間有効であり、このコマンドはその期間の開始を記録します。このコマンドは工場で行われており、ユーザーが実行する必要があるのは現場でバッテリーが交換された場合のみです。既存のバッテリーの現行の取り付け日と有効期限は、**status battery** コマンドで表示できます。

構文

device battery-replaced

オプション

なし。

パラメーター

なし。

関連コマンド

[status battery](#) を参照してください。

例

```
Console> device battery-replaced
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device intrusion allow

目的

- ケースを開くことが許可されるデバイスにおいて、それを許可します。
- [device intrusion clear](#) を参照してください。

構文

device intrusion allow

パラメーター

なし。

例

```
Console> device intrusion allow
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device intrusion clear

目的

- アプライアンスのケースが物理的に開いている場合に、記録されたデバイスの侵入状態をクリアし、ケースが侵入されたと見なされなくなるようにします。
- アプライアンスのお客様交換可能部品や現場交換可能部品の一部には、ケースを開くことが必要なものがあります。部品を交換した後でこのコマンドを実行して、ケースを閉じます。

構文

device intrusion clear

パラメーター

なし。

使用上の注意

- このコマンドは、7198 または 7199 マシン・タイプのみで有効です。
- 9235 マシン・タイプでケースを開くと、アプライアンスが永続的に操作不能になり、アプライアンスを工場に戻さなければならなくなるので、9235 マシン・タイプのケースを開いてはいけません。これらの物理アプライアンスには、ケースが開かれたときに検出するセンサーがあります。物理アプライアンスの他のマシン・タイプでは、このコマンドは作用しません。このコマンドは仮想アプライアンスには適用されません。
- [status intrusion](#) も参照してください。

例

```
Console> device intrusion clear
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device intrusion disallow

目的

以前に [device intrusion allow](#) を呼び出した後で、侵入検出を再び使用可能にします。

構文

device intrusion disallow

パラメーター

なし。

使用上の注意

- このコマンドは、マシン・タイプ 7198 および 7199 にのみ作用します。詳細については、[device intrusion clear](#)を参照してください。
- このコマンドは、侵入センサーのリセットを行いません。一般に、アプライアンスをリブートする前に [device intrusion clear](#) を呼び出すことをお勧めします。

例

```
Console> device intrusion disallow
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

device nvdimm clear

目的

nvDIMM ボリューム・ラベルをクリアします。

構文

device nvdimm clear

パラメーター

なし。

使用上の注意

このコマンドは 9006 マシン・タイプでのみ許可されています。

関連コマンド

なし。

例

```
Console> device nvdimm clear
CWZBR02923I: Volume labels cleared
Console>
```

親トピック: [特殊目的コマンド](#)

device clear-intrusion

このコマンドは推奨されません。代わりに、[device intrusion clear](#) を使用してください。

親トピック: [特殊目的コマンド](#)

device restart

目的

アプライアンスをシャットダウンして、リブートします。リブート時に、構成されたすべてのアプリケーションが開始します。これは、電源を入れ直さずにアプライアンスをリブートする、望ましい方法です。

構文

device restart

オプション

なし。

パラメーター

なし。

使用上の注意

ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、シャットダウン中にネットワーク接続が終了します。再始動サイクル全体のすべてのメッセージを表示したい場合には、シリアル・コンソールに接続する必要があります。

関連コマンド

[device shutdown](#) を参照してください。

例

```
Console> device restart
Ok
Console>
Preparing system for shutdown: s1 hb hn 2 0 s2 rs nt hc ns sy ul r6
machine restart
Starting system... dn un nvm 7198 r1 r2 lo wf lu pt tc sr in cd ln mu ku up us
t1 t2 md mm mcg mt ml na fi
login:
```

親トピック: [特殊目的コマンド](#)

device shutdown

目的

アプライアンスをシャットダウンして、電源オフ状態のままにします。自動的にリブートしません。このコマンドの実行後にアプライアンスをリブートしたい場合には、電源ボタンが単一プッシュ・ボタンの場合はボタンを押し、電源ボタンが2ポジションのロッカー・スイッチの場合はオンに切り替えます。これは、電源ボタンのみを使用するのではなくアプライアンスをシャットダウンする、望ましい方法です。

構文

device shutdown

オプション

なし。

パラメーター

なし。

使用上の注意

ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、シャットダウン中にネットワーク接続が終了します。シャットダウン・サイクル全体のすべてのメッセージを表示したい場合には、シリアル・コンソールに接続する必要があります。

関連コマンド

[device restart](#) を参照してください。

例

```
Console> device shutdown
Ok
Console>
Preparing system for shutdown:  s1  hb  hn  2  0  s2  rs  nt  hc  ns  sy  ul  r0
machine shutdown
```

親トピック: [特殊目的コマンド](#)

echo

目的

コマンド行インターフェース (CLI) の出力ストリームに書き込みます。

構文

echo *text*

パラメーター

text

コマンド行インターフェース (CLI) のストリームに書き込むテキスト。このコマンドが発行されたコマンド行インターフェース (CLI) のインスタンスにのみ表示されます。

使用上の注意

echo コマンドは、コマンド行インターフェース (CLI) のスクリプトで役に立ちます。

例

echo コマンドを使用して、**datetime get** 表示の出力を前に付加します。

```
Console> echo Current time is: ; datetime get
Current time is:
GMT: 2012-03-22T15:32:06Z
Local: Mar 22, 2012 3:32:06 PM
Console>
```

親トピック: [特殊目的コマンド](#)

export

目的

アプライアンス構成をローカルに、またはリモート・ファイル・サーバーにアップロードするための一時ロケーションに、エクスポートします。

構文

```
config export
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

使用上の注意

-file フラグを export コマンドに受け渡す必要があります。

例

アプライアンス構成をエクスポートします。

```
config export -file foo.json -silent
```

親トピック: [特殊目的コマンド](#)

file delete

目的

アプリケーションからファイルを削除します。

構文

```
file delete filename
```

パラメーター

filename

削除されるファイル

関連コマンド

[file get](#)、[file list](#)、および [file put](#) を参照してください。

例

platform collect-pd コマンドによって以前に作成されたファイルを削除します。

```
Console> file list  
collect-pd.txt 103576 bytes created Mar 22, 2012 3:38:32 PM  
Console> file delete collect-pd.txt  
Console>
```

親トピック: [特殊目的コマンド](#)

file get

目的

リモート・ロケーションからアプライアンスにファイルをコピーします。

構文

```
file get url localfile
```

パラメーター

url

以下の2つの形式のうちの1つで指定された Uniform Resource Locator。以下の形式では、読みやすくするための空白文字が付いています。コマンドの URL には空白文字を含めないでください。

- プロトコル HTTP および FTP の場合:

```
protocol: // [username [:password] @]hostname [:port]/[%2F abspath/][relpath /] filename
```

ここで、

protocol

FTP または HTTP。

username

リモート・ホストのユーザー名。これは URL のオプション・コンポーネントです。

password

リモート・ホストでの *username* のパスワード。これは URL のオプション・コンポーネントです。

hostname

リモート・ホスト名または IP アドレス。

port

リモート・サーバーの 10 進数のポート番号。これは URL のオプション・コンポーネントです。

abspath

'%2F' または '%2f' の 3 文字の接頭部がついた、リモート・ファイルの絶対パス。これはプロトコル FTP および SCP でサポートされ、URL のオプション・コンポーネントです。

relpath

リモート・ファイルの相対パス。このパスは、リモート・システムのデフォルト・ディレクトリーに対して相対です。*username* が指定されている場合、デフォルト・ディレクトリーは一般にそのユーザーのホーム・ディレクトリーです。このパスは、URL のオプション・コンポーネントです。

filename

リモート・ファイル名。

- プロトコル SCP の場合:

```
protocol: // username@hostname: [%2F abspath/][relpath /] filename
```

ここで、

protocol

SCP。

username

リモート・ホストのユーザー名。これは URL のオプション・コンポーネントです。

hostname

リモート・ホスト名または IP アドレス。IPv6 アドレスに解決するホスト名がサポートされます。しかし、*hostname* がファイル・パスとコロンの区切られるため、コロンを含むリテラル IPv6 アドレスはサポートされません。

abspath

'%2F' または '%2f' の 3 文字の接頭部がついた、リモート・ファイルの絶対パス。これはプロトコル FTP および SCP でサポートされ、URL のオプション・コンポーネントです。

relpath

リモート・ファイルの相対パス。このパスは、リモート・システムのデフォルト・ディレクトリーに対して相

対です。 *username* が指定されている場合、デフォルト・ディレクトリーは一般にそのユーザーのホーム・ディレクトリーです。このパスは、URL のオプション・コンポーネントです。

filename

リモート・ファイル名。

localfile

ファイルのローカル・コピーに割り当てられる名前。絶対パス・エレメント (頭にスラッシュが付いている) またはサブディレクトリー (スラッシュ) は許可されません。相対の、階層なしの名前でなければなりません。

関連コマンド

[file delete](#)、[file list](#)、および [file put](#) を参照してください。

例

リモート・ロケーションからファームウェア・ファイルを取得して、アプライアンスで newfirmware という名前を付けます。

```
Console> file get scp://user1@server1.raleigh.ibm.com:~/dev_bedrock.scrypt2
newfirmware.scrypt2
user1@server1.raleigh.ibm.com's password:
dev_bedrock.scrypt2          100% 136MB 34.0MB/s 00:04
Wrote 142553856 bytes to local storage
Console>
```

親トピック: [特殊目的コマンド](#)

file list

目的

アプライアンス上のファイルをリストします。

構文

file list

パラメーター

なし。

関連コマンド

[file delete](#)、[file get](#)、および [file put](#) を参照してください。

例

現在アプライアンス上にあるファイルを表示します。

```
Console> file list
collect-pd.txt 103794 bytes created Mar 22, 2012 5:30:56 PM
newfirmware 142553856 bytes created Mar 22, 2012 4:41:17 PM
Console>
```

親トピック: [特殊目的コマンド](#)

file put

目的

アプライアンスからリモート・ロケーションにファイルをコピーします。

構文

```
file put localfile url
```

パラメーター

localfile

転送されるローカル・ファイルの名前。このファイルは存在している必要があります。

url

[file get](#) に記載された、Uniform Resource Locator。ただし、**file put** コマンドでは HTTP プロトコルはサポートされません。

関連コマンド

[file delete](#)、[file get](#)、および [file list](#) を参照してください。

例

ローカル・ファイル collect-pd.txt を diag.txt としてリモート・システムにコピーします。

```
Console> file put collect-pd.txt scp://user1@system1.rtp.raleigh.ibm.com:~/diag.txt
user1@system1.rtp.raleigh.ibm.com's password:
collect-pd.txt                100% 101KB 101.4KB/s   00:00
Console>
```

親トピック: [特殊目的コマンド](#)

firmware pristine-install

目的

ファームウェアをインストールし、アプライアンスを工場出荷時に近い状態にリストアします。

構文

firmware pristine-install *image*

パラメーター

image

インストールするファームウェア・イメージ (scrypt2 ファイルの形式)

使用上の注意

- **pristine-install** コマンドは、一般的なファームウェアのアップグレードにコピーされる構成やデータ・ファイルのコピーせずに、新規ファームウェア・イメージをインストールします。
- **pristine-install** コマンドは、アプライアンスに古いバージョンのファームウェアをインストールするのに使用できます。
- **pristine-install** コマンドは、影響がより少ない方法ではうまくいかなかった場合に、エラー状況のリカバリーに使用してみることもできます。
- この操作の一部としてアプライアンスがリブートされるので、ネットワーク接続は中断されます。したがって、通常シリアル・コンソールからこのコマンドを実行することが推奨されます。

関連コマンド

[firmware upgrade](#)、[firmware rollback](#)、および [device RESET](#) を参照してください。

例

ファームウェアを古いファームウェア・レベルに戻します。

```
Console> firmware pristine-install oldfirmware.scrypt2
Upgrading firmware...
Verifying image signature
Executing dynamic loader
Executing dynamic loader
Validating image
Extracting firmware from image
Extracting firmware manifest
Executing pre-installation
Deleting previous installation
Linking common files.
Extracting files
Verifying installation
Copying configuration from existing installation
Switching to new installation
Upgrade or rollback succeeded. Rebooting
```

親トピック: [特殊目的コマンド](#)

firmware rollback

目的

ファームウェア・レベルを、アプライアンスにインストールされたファームウェアの以前のレベルに戻します。

構文

firmware rollback

パラメーター

なし。

使用上の注意

- アプライアンスは最大で 2 レベル (アクティブおよび代替) のファームウェアを保持できます。 **firmware rollback** コマンドを呼び出す都度、2 つのイメージが交互に切り替わります。
- このコマンドは構成をロールバックします。ロールバック後、構成は最後のアップグレード前の状態になります。
- このコマンドはコンポーネント・ファームウェアをロールバックします。各コンポーネントが保持できるコンポーネント・ファームウェアは 1 コピーのみであるため、ロールバックされたバージョンのコンポーネント・ファームウェアがコンポーネントに再インストールされます。
- この操作の一部としてアプライアンスがリブートされるので、ネットワーク接続は中断されます。したがって、通常シリアル・コンソールからこのコマンドを実行することが推奨されます。

関連コマンド

[firmware upgrade](#)、[firmware pristine-install](#)、および [device RESET](#) を参照してください。

例

アプライアンスを、インストールされている以前のバージョンのファームウェアに戻します。

```
Console> firmware rollback
Rolling back firmware...
Upgrade or rollback succeeded. Rebooting...
```

親トピック: [特殊目的コマンド](#)

firmware upgrade

目的

アプライアンスに新規レベルのファームウェアをインストールします。

構文

firmware upgrade *image*

パラメーター

image

インストールするファームウェア・イメージ (scrypt2 ファイルの形式)

使用上の注意

この操作の一部としてアプライアンスがリブートされるので、ネットワーク接続は中断されます。したがって、通常シリアル・コンソールからこのコマンドを実行することが推奨されます。

関連コマンド

[firmware pristine-install](#)、[firmware rollback](#)、および [device RESET](#) を参照してください。

例

アプライアンスに新規レベルのファームウェアをインストールします。

```
Console> firmware upgrade myfirmware.scrypt2
Upgrading firmware...
Verifying image signature
Executing dynamic loader
Validating image
Extracting firmware from image
Extracting firmware manifest
Executing pre-installation
Deleting previous installation
Linking common files
Extracting files
Verifying installation
Copying configuration from existing installation
Switching to new installation
Upgrade or rollback succeeded. Rebooting
```

親トピック: [特殊目的コマンド](#)

force recycle

目的

データを何も保存せずに、WebSphere DataPower XC10 アプライアンスのプロセスを再始動します。データ損失が発生する可能性があるため、データ損失があっても問題ない場合、または device restart コマンドを試してもアプライアンスが使用可能にならなかった場合に限り、このコマンドを実行してください。

構文

force recycle

パラメーター

なし。

関連コマンド

[device restart](#)

例

```
Console> force-recycle
Force Stopped all XC-10 processes
Forced recycle of appliance
```

親トピック: [特殊目的コマンド](#)

get-dns-search

目的

構成されたドメイン・ネーム・サーバーの検索ドメインを表示します。

構文

get-dns-search

パラメーター

なし。

関連コマンド

[set-dns-search](#)、[get-dns-servers](#)、および [set-dns-servers](#) を参照してください。

例

ドメイン・ネーム・サーバーの検索ドメインを構成してから表示します。

```
Console> set-dns-search short.example.com example.com
Ok
Console> get-dns-search
DNS search domains:
    short.example.com
    example.com
Console>
```

親トピック: [特殊目的コマンド](#)

get-dns-servers

目的

構成されたドメイン・ネーム・サーバーの IP アドレスを表示します。

構文

get-dns-servers

パラメーター

なし。

関連コマンド

[set-dns-servers](#)、[get-dns-search](#)、および [set-dns-search](#) を参照してください。

例

ドメイン・ネーム・サーバーを構成してから表示します。

```
Console> set-dns-servers 9.42.106.2 2002:92a:8f7a:106:9:42:106:42
Ok
Console> get-dns-servers
Domain (DNS) servers:
    9.42.106.2
    2002:92a:8f7a:106:9:42:106:42
Console>
```

親トピック: [特殊目的コマンド](#)

get-ntp-servers

目的

構成された Network Time Protocol サーバーを表示します。

構文

get-ntp-servers

パラメーター

なし。

関連コマンド

[set-ntp-servers](#) を参照してください。

例

NTP サーバーを構成してから表示します。

```
Console> set-ntp-servers example.server.com 127.0.0.1
Ok
Console> get-ntp-servers
NTP servers:   example.server.com
               127.0.0.1
Console>
```

親トピック: [特殊目的コマンド](#)

help

目的

コマンドのヘルプ情報を表示します。

構文

- サポートされるコマンドのリストを表示する: **help command**
- 特定のコマンドのヘルプを表示する: **help command**

パラメーター

command

コマンド行インターフェース (CLI) コマンド

例

platform コマンドのサブコマンドを表示します。

```
Console> help platform
The following platform commands are available:
  platform collect-pd [PDFfilename | console]
  platform log-level ...
  platform must-gather filename [PDFfilename]
```

親トピック: [特殊目的コマンド](#)

import

目的

アプライアンス構成をインポートします。

構文

```
config import
```

パラメーター

-file

-file フラグの後に、構成コンテンツを保管したい場所のファイル名が続きます。

-truststore

-trustStore フラグの後に、アプライアンスの構成の一部であるトラストストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-keystore

-keyStore フラグの後に、アプライアンスの構成の一部である鍵ストア・ファイル名が続きます。構成リストに保管されている設定以外の SSL 設定 (**export_TIMESTAMP.json** 設定など) をインポートする場合は、この値を指定する必要があります。

-noRestart

特定のアプライアンス設定では、アプライアンスを再始動することが要求されます。このフラグを設定すると、アプライアンスはどれも再始動されません。しかし、**-noRestart** パラメーターを設定した場合には、再始動が要求される設定を保存するために、アプライアンスを手動で再始動する必要があります。

-silent

これが指定されると、**config** コマンドの状況メッセージが表示されません。これはデフォルトの動作です。

使用上の注意

-file フラグの付いた **import** コマンドを受け渡す必要があります。

例

アプライアンス構成をインポートします。

```
config import -file foo.json -silent
```

親トピック: [特殊目的コマンド](#)

license accept

目的

ライセンスを表示し、コマンド行インターフェース (CLI) ユーザーがライセンスに同意するか拒否するよう要求します。

構文

license accept

パラメーター

なし。

例

ライセンス同意プロセスを実行します。

```
Console> license accept
Bedrock Project, WebSphere Technical Institute
Copyright 2009-2010, IBM Corporation
```

```
This is your sample license.
```

```
Accept; Reject: accept
Console>
```

親トピック: [特殊目的コマンド](#)

license get

目的

アプライアンスに新規または変更されたライセンスを転送します。

構文

license get url

パラメーター

url

取得するライセンスのリモート・ロケーション

使用上の注意

URL は以下のプロトコルのうちの 1 つで指定する必要があります: HTTP、FTP、または SCP。URL 構文の説明については、[URL 構文](#)を参照してください。

例

新規ライセンス lic-test をライセンス・リポジトリに取得します。

```
Console> license get scp://user1@server1.raleigh.ibm.com:~/temp/lic
user1@server1.raleigh.ibm.com's password:
lic                               100%   14     0.0KB/s   00:00
Wrote 14 bytes to local storage
Console>
```

親トピック: [特殊目的コマンド](#)

license reset

目的

ライセンス同意状況をリセットします。

構文

license reset

パラメーター

なし。

使用上の注意

コマンド行インターフェースを次に呼び出すには、ユーザーがライセンスに同意する必要があります。

例

ライセンス同意状況をリセットします。

```
Console> license reset  
Ok
```

親トピック: [特殊目的コマンド](#)

listExports

目的

ローカルに保管された、エクスポートされた構成のリストを表示します。

構文

```
config listExports
```

パラメーター

なし

使用上の注意

なし

例

エクスポートされたアプライアンスの構成をリストします。

```
config listExports
```

親トピック: [特殊目的コマンド](#)

locale get

目的

コマンド行インターフェース (CLI) でメッセージの表示の制御に使用されるロケール設定を取得します。

構文

locale get

パラメーター

なし。

例

ロケール設定を取得します。

```
Console> locale get  
Locale is en_US  
Console>
```

親トピック: [特殊目的コマンド](#)

locale set

目的

コマンド行インターフェース (CLI) でメッセージの表示に使用されるロケールを設定します。

構文

```
locale set language [country]
```

パラメーター

language

言語の省略形

country

国の省略形

使用上の注意

ロケール設定は、コマンドが発行された CLI ですぐに有効になります。既にアクティブな可能性のある別の CLI インスタンスでは、新規ロケール設定を有効にするために、ユーザーは CLI を終了して再入力する必要があります。

例

ロケールを米国の表記規則を使用して英語に設定します。

```
Console> locale set en US  
Console>
```

親トピック: [特殊目的コマンド](#)

locate-led

目的

場所の特定 LED をオンまたはオフにします。場所の特定 LED は物理アプライアンスの前部にあるライトです。灯台の形のアイコンで表示され、LED の色は青です。場所の特定 LED の目的は、データ・センターの特定のアプライアンスを視覚的にお客様に見つけやすくすることです。場所の特定 LED はデフォルトでオフであり、このコマンドを使用して明示的にオンにする必要があります。LED の状態は持続しないので、アプライアンスがリブートした後で、場所の特定 LED はオフになります。場所の特定 LED は、手動で視覚的に位置を示す以外の機能はありません。これは物理アプライアンスにのみ適用されます。仮想アプライアンスには場所の特定 LED はありません。

構文

```
locate-led { on | off }
```

オプション

なし。

パラメーター

on

場所の特定 LED をオンにします。

off

場所の特定 LED をオフにします。

使用上の注意

場所の特定 LED の状態を照会することはできません。場所の特定 LED の状態を、現在なっている状態に設定することはエラーとは見なされません。

例

```
Console> locate-led on
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

log message

目的

- テキスト・ストリングを診断ログ・ファイルに書き込みます。
- 例えば、このコマンドを使用して、特定のアクティビティの開始または終了を示します。ログ・ファイルの手動による検査や自動化された後処理で、このテキストを見つけることができ、さらに分析するための開始点として使用できます。
- このメッセージは、INFO (通知) レベルで記録されます。

構文

log message *text*

パラメーター

text

ログに書き込むメッセージ・テキスト

例

```
Console> log message About to run the acceptance tests
Console>
```

親トピック: [特殊目的のコマンド](#)

netif

netif コマンドは推奨されず、デフォルトでは存在しなくなりました。

代わりに、ethernet-interface 構成オブジェクトの [共通コマンド](#) に記載されているコマンドを使用してください。

親トピック: [特殊目的コマンド](#)

net-test available

目的

ネットワーク・インターフェースにアクティブなキャリアがあるかどうかをテストします。

構文

net-test available

パラメーター

なし。

関連コマンド

他の診断コマンドについては、[net-test dns](#)、[net-test ping](#)、および [net-test tcp](#) を参照してください。

例

アクティブなネットワーク・インターフェースの使用可能性をテストします。

```
Console> net-test available
Network available
Console>
```

親トピック: [特殊目的コマンド](#)

net-test dns

目的

指定されたホスト名でドメイン・ネーム・サーバー検索を行います。

構文

```
net-test dns hostname
```

パラメーター

hostname

ドメイン・ネーム・サーバー検索で使用するホスト名

関連コマンド

[net-test available](#)、[net-test ping](#)、および [net-test tcp](#) を参照してください。

例

system1.rtp.ibm.com で DNS 検索を行います。

```
Console> net-test dns system1.rtp.raleigh.ibm.com
9.42.77.42
Console>
```

親トピック: [特殊目的コマンド](#)

net-test ping

目的

指定されたホスト名または IP アドレスへの接続をテストします。

構文

```
net-test ping host
```

パラメーター

host

リモート・システムのホスト名または IP アドレス

関連コマンド

他の診断コマンドについては、[net-test available](#)、[net-test dns](#)、および [net-test tcp](#) を参照してください。

例

指定されたホスト名への接続をテストします。リモート・システムに到達できます。

```
Console> net-test ping system1.rtp.raleigh.ibm.com
Ok
Console>
```

指定されたアドレスへの接続をテストします。リモート・システムに到達できません。

```
Console> net-test ping 4.4.4.4
ping failed
Console>
```

親トピック: [特殊目的コマンド](#)

net-test tcp

目的

指定されたホスト名または IP アドレスにあるポートへの TCP 接続を開くことができるかをテストします。

正常に開かれた場合、TCP 接続経由でデータは送信されず、閉じられます。

構文

```
net-test tcp host port
```

パラメーター

host

リモート・システムのホスト名または IP アドレス

port

リモート・システムの TCP ポート。

関連コマンド

他の診断コマンドについては、[net-test available](#)、[net-test dns](#)、および [net-test ping](#) を参照してください。

例

ホスト system1.rtp.raleigh.ibm.com のポート 80 に到達できるかをテストします。

```
Console> net-test tcp system1.rtp.raleigh.ibm.com 80
Ok
```

親トピック: [特殊目的コマンド](#)

net-test traceroute

目的

指定されたホスト名または IP アドレスへのパスをレポートします。

正常に開かれた場合、TCP 接続経由でデータは送信されず、閉じられます。

構文

```
net-test traceroute host
```

パラメーター

host

リモート・システムのホスト名または IP アドレス

関連コマンド

他の診断コマンドについては、[net-test available](#)、[net-test dns](#)、[net-test ping](#)、および [net-test tcp](#) を参照してください。

例

指定されたホストへのルートを判別します。

```
Console> net-test traceroute localhost
localhost (127.0.0.1)  0.097 ms
Ok
Console>
```

```
Console> net-test traceroute 127.0.0.1
localhost (127.0.0.1)  0.099 ms
Ok
Console>
```

指定されたアドレスへの接続をテストします。リモート・システムに到達できません。

```
Console> net-test traceroute host.that.does.not.exist
bad address 'host.that.does.not.exist'
traceroute failed
Console>
Console> net-test traceroute 9.65.47.137
sendto: Network is unreachable
traceroute failed
```

親トピック: [特殊目的コマンド](#)

nodename get

目的

アプライアンスの構成名を取得します。

構文

nodename get

パラメーター

なし。

関連コマンド

[nodename set](#) を参照してください。

例

アプライアンス名を設定してから取得します。

```
Console> nodename set appliance1
Console> nodename get
nodename is appliance1
Console>
```

親トピック: [特殊目的コマンド](#)

nodename set

目的

アプライアンスの構成名を設定します。

構文

```
nodename set name
```

パラメーター

name

アプライアンスの名前

関連コマンド

[nodename get](#) を参照してください。

例

アプライアンス名を設定してから取得します。

```
Console> nodename set appliance1
Console> nodename get
nodename is appliance1
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture clear

目的

指定されたネットワーク・インターフェースのパケット・トレース・キャプチャー・ファイルを削除します。

構文

```
packet-capture clear interface
```

パラメーター

interface

パケット・トレース・ファイルをクリアする必要があるネットワーク・インターフェース

使用上の注意

- インターフェースでパケット・トレースが実行された後、同じインターフェースで別のパケット・キャプチャーを開始するためには、その前に、ファイルをクリアする必要があります。

関連コマンド

[packet-capture list](#)、[packet-capture start](#)、および [packet-capture stop](#) を参照してください。

例

パケット・キャプチャー・アクティビティを表示し、それからインターフェース eth0 で収集されたファイルをクリアします。

```
Console> packet-capture list
interface      status          file(s)
eth1    Finished      eth1pc1
                               eth1pc2
-----
eth0    Finished      eth0pc
Ok
Console> packet-capture clear eth0
Packet capture dump files cleared for eth0
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture list

目的

アプライアンスのネットワーク・キャプチャー・アクティビティを表示します。

構文

packet-capture list

パラメーター

なし。

関連コマンド

[packet-capture clear](#)、[packet-capture start](#)、および [packet-capture stop](#) を参照してください。

例

アプライアンスのネットワーク・キャプチャー・アクティビティを表示します。この例では、インターフェース **eth0** のネットワーク・キャプチャーが完了しています。インターフェース **mgt0** のトレースは、リング・バッファ・ファイル 3 つにまだアクティブに書き込み中です。

```
Console> packet-capture list
interface      status          file(s)
eth0    Finished      eth0pc
-----
mgt0    Capturing     mgt0pc1
                               mgt0pc2
                               mgt0pc3
-----
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture start

目的

パケット・トレース収集をアクティブ化します。

構文

```
packet-capture start interface filename duration filesize [ ring-buffer-number ] ["filter "]
```

パラメーター

interface

ネットワーク・インターフェース名 (例えば **eth0**)

filename

出力トレース・ファイル名。ring-buffer-number も指定されている場合は、ファイル名に索引番号が追加されま
す。

duration

パケット・キャプチャーの期間 (単位は秒)。値 **0** は、**packet-capture stop** コマンドが発行されるまでパケット・
キャプチャーを続行することを示します。

filesize

トレース・ファイルのサイズ (単位はメガバイト (MB))。値は 0 から 100 の範囲でなければなりません。

ring-buffer-number

指定されると、リング・バッファ機能が使用可能に設定され、リング・バッファ・ファイルの数が設定されます。
値は 0 から 10 の範囲でなければなりません。

filter

収集されるパケットを選択するのに使用するフィルター (二重引用符 (" ") で囲む)。拡張フィルター機能がサポート
され、例えばパケット・ソース、宛先、ポート、およびプロトコルのフィルタリングが含まれます。サポートされるフ
ィルター構文の完全な詳細については、インターネットで pcap-filter を検索してください。

使用上の注意

- インターフェースでパケット・キャプチャーを開始するためには、以下の条件が必要です。
 - インターフェースはアクティブであることが必要です。
 - インターフェースで他のパケット・キャプチャーはアクティブにはできません。
 - aggregate-interface にトレースが要求される場合、aggregate-interface 自体でトレースを開始しま
す。メンバー・リンクのパケット・キャプチャーは完全なデータを含まない場合があります。
 - 以前のパケット・キャプチャーからのファイルはすべて、**packet-capture clear** コマンドを使用して
クリアする必要があります。
- パケット・キャプチャーは、以下のいずれかの条件が発生した場合に停止します。
 - リング・バッファ機能が使用されていない場合、ファイル・サイズの制限に達したとき。
 - パケット・キャプチャー期間が終了したとき。
 - **packet-capture stop** コマンドが発行されたとき。
- 出力トレース・ファイルは、Wireshark など、libpcap フォーマットのデータを読むことができる市販の既製ツ
ールで分析できます。

関連コマンド

[packet-capture clear](#)、[packet-capture list](#)、および [packet-capture stop](#) を参照してください。

例

インターフェース eth1 でパケット・キャプチャーを開始して 60 秒間行います。トレース・ファイルのサイズは最大
10 MB まで可能です。パケット・キャプチャーを pc11 と pc12 という名前の 2 つのリング・バッファ・ファイルに
書き込みます。

```
Console> packet-capture start eth1 pc1 60 10 2 "icmp"  
Packet capture started on eth1  
Ok  
Console>
```

親トピック: [特殊目的コマンド](#)

packet-capture stop

目的

指定されたネットワーク・インターフェースで進行中のパケット・キャプチャーを停止します。

構文

```
packet-capture stop interface
```

パラメーター

interface

パケット・キャプチャーを停止する必要があるネットワーク・インターフェース

使用上の注意

- パケット・キャプチャー・ファイルは、**packet-capture clear** コマンドが発行されるまで存在し続けます。
- パケット・キャプチャー・ファイルを分析用にアプライアンスの外に転送するには、**file put** を使用します。

関連コマンド

[packet-capture clear](#)、[packet-capture list](#)、および [packet-capture start](#) を参照してください。

例

アクティブなパケット・キャプチャーを表示し、その後インターフェース eth1 のパケット・キャプチャー・アクティビティを停止します。

```
Console> packet-capture list
interface      status          file(s)
eth1    Capturing     eth1pc1
                               eth1pc2
-----
eth0    Finished      eth0pc
-----
Ok
Console> packet-capture stop eth1
Stopping capture on eth1
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

platform collect-pd

目的

問題診断データの標準セットを単一のファイルに収集します。このファイルは、表示または IBM サポートに送信するために、アプライアンスから転送できます。この結果できる出力ファイルは、コマンド行インターフェース (CLI) コマンドの事前定義リストの出力です。このコマンドの目的は、サポート担当者が問題診断の一環としてお客様に実行を依頼する可能性がある多数のコマンド行インターフェース (CLI) コマンドを「記す」ことです。

構文

```
platform collect-pd [ PDfilename ]
```

パラメーター

PDfilename

問題診断コマンドの出力が書き込まれるテキスト・ファイルの名前を指定します。指定されない場合、デフォルト値は collect-pd.txt です。出力ファイルが存在する場合、上書きされます。

使用上の注意

- 出力はお客様が読める平文と、IBM サポート向けのみ意図された隠されたテキスト・データの両方を含むことがあります。
- このコマンドの出力は、**platform must-gather** コマンドにより収集されたデータにも自動的に含まれています。そのためユーザーが **platform must-gather** を実行する場合は、別途 **platform collect-pd** を実行する必要はありません。
- 出力はすべて印刷可能なテキストであるため、出力ファイルの内容は、**show log** コマンドを使用してコマンド行インターフェース (CLI) セッション内で表示することができます。これはアプライアンスがすべてのネットワーク接続を失った場合に役立ちます。

関連コマンド

[platform must-gather](#)、[file put](#)、および [show log](#) を参照してください。

例

診断コマンド・データを収集し、出力をデフォルトの出力ファイルに送信します。

```
Console> platform collect-pd
Console> file list
collect-pd.txt 79336 bytes created Feb 6, 2012 9:32:17 PM
Console>
```

親トピック: [特殊目的コマンド](#)

platform log-level get

このコマンドは推奨されません。代わりに、**show logging-settings** を使用してください。

親トピック: [特殊目的コマンド](#)

platform log-level set

このコマンドは推奨されません。代わりに、**edit logging-settings** を使用してください。

親トピック: [特殊目的コマンド](#)

platform service disable ssh

目的

- SSH プラットフォーム・サービスがセキュリティー・ポリシーに影響する場合に、使用不可に設定します。

構文

```
platform service ssh ... platform service ssh disable
```

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service enable ssh](#)、[platform service enable telnet](#)、[platform service disable telnet](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service ssh ...
CWZBR02901I: platform service ssh disable
```

親トピック: [特殊目的コマンド](#)

platform service disable telnet

目的

- Telnet プラットフォーム・サービス・セキュリティー・プロトコルを使用不可に設定します。

構文

platform service telnet ... platform service telnet disable

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service enable telnet](#)、[platform service enable ssh](#)、[platform service disable ssh](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service telnet ...
CWZBR02901I: platform service telnet disable
```

親トピック: [特殊目的コマンド](#)

platform service enable ssh

目的

- SSH プラットフォーム・サービス・セキュリティー・プロトコルを使用可能に設定します。

構文

platform service ssh ... platform service ssh enable

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service disable ssh](#)、[platform service enable telnet](#)、[platform service disable telnet](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service ssh ...
CWZBR02901I: platform service ssh enable
```

親トピック: [特殊目的コマンド](#)

platform service enable telnet

目的

- Telnet プラットフォーム・サービス・セキュリティー・プロトコルを使用可能に設定します。

構文

platform service telnet ... platform service telnet enable

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[platform service disable telnet](#)、[platform service enable ssh](#)、[platform service disable ssh](#) のいずれかを参照してください。

例

```
Console> help platform service
CWZBR02449I: platform service telnet ...
CWZBR02901I: platform service telnet enable
```

親トピック: [特殊目的コマンド](#)

platform must-gather

目的

IBM サポートに問題を報告するために必要な、集めておくべき診断データを収集します。

構文

```
platform must-gather filename [ PDfilename ]
```

パラメーター

filename

すべての診断データを含む出力ファイル (圧縮 (.tgz) 形式)

pdfilename

問題診断コマンドの出力が書き込まれるテキスト・ファイルの名前。このファイルは、platform collect-pd コマンドにより収集された同じ出力を含みます。指定されない場合、デフォルト値は collect-pd.txt です。出力ファイルが存在する場合、ファイルは上書きされます。

使用上の注意

- 出力は平文と、IBM サポートのみが読むように意図された隠されたデータの両方を含むことがあります。

関連コマンド

[platform collect-pd](#) を参照してください。

例

診断データを problem-data.tgz という名前のファイルに収集します。

```
Console> platform must-gather problem-data.tgz
Console> file list
problem-data.tgz 3449276 bytes created Mar 27, 2012 11:15:48 AM
collect-pd.txt 572878 bytes created Mar 27, 2012 11:15:48 AM
Console>
```

親トピック: [特殊目的コマンド](#)

raid delete

目的

RAID カード・コントローラーの構成を削除し、ボリューム情報とパーティション情報もクリーンアップして、物理ディスクを工場出荷状態に戻します。

構文

raid delete

パラメーター

なし。

使用上の注意

- ディスク上のデータは失われます。
- RAID カード・コントローラーの構成は消去され、次のブートで自動的に再構成されます。
- パーティションは除去され、パーティション情報はディスクから消去されます。
- すべての物理ディスクの状態はリセットされます。
- RAID カード・コントローラーにより管理されていたボリュームのみが削除されます。
- 管理対象でないボリュームは削除されません。

例

RAID コントローラーの構成を削除します。

```
Console> raid delete
Delete volume
Delete volume successfully
Console> status volume
No volume
Console> status raid physical
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 0
  Logical drive ID: 0
  Logical drive name: raid0
  State: unconfigured good drive
...
```

親トピック: [特殊目的コマンド](#)

raid re-use

目的

以前別の RAID カード・コントローラーで使用された物理ディスクを、アプライアンスで使用できるようにします。以前に使用されたディスクがアプライアンスに挿入されると、RAID コントローラーにより外部ディスクと認識されることがあります。このコマンドは、ディスクを RAID コントローラーが使用できるようにして、ディスクを RAID アレイに再ビルドするプロセスを自動的に開始します。

構文

raid re-use

パラメーター

なし。

使用上の注意

- 物理ディスク上の古いデータは失われます。
- このコマンドは、使用された物理ディスクの外部状態をクリアし、さらに再ビルド・プロセスをトリガーします。ビルド・プロセスが完了すると、物理ディスクはオンラインに移行します。
- このコマンドは、1つの物理ディスクのみに機能します。複数の外部物理ディスクが同時に挿入された場合、このコマンドはそれらを RAID 構成に入れることができません。

例

使用された物理ディスクの外部状態をクリーンアップし、再ビルドを開始します。

```
Console> raid re-use
Bring physical drive back online after hot swap
Done
Console> status raid physical
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: rebuild
  Progress percent: 3
...
```

親トピック: [特殊目的コマンド](#)

raid foreign-config import

目的

外部構成のインポートは 9005 でサポートされています。外部構成とは、ドライブの交換セットに既に存在する RAID 構成です。ドライブの外部構成は、クリアするか、またはドライブが取り付けられているアプライアンスにインポートすることができます。デフォルトでは、9005 マシンで外部構成が見つかったら、それはクリアされます。ユーザーは必要に応じて、CLI コマンドまたは C++/Java API を使用した外部構成のインポートを使用可能に設定することができます。

構文

raid foreign-config import

パラメーター

なし。

使用上の注意

交換ディスクの外部構成をインポートするには、以下のステップを順番に行います。

1. コマンド **raid foreign-config import** を実行して、外部構成のインポートを使用可能に設定します。
2. アプライアンスをシャットダウンします。
3. ドライブを、外部構成が既に存在するものに交換します。

アプライアンスを電源オンにした後、`status volume` を実行して RAID ボリュームがオンラインになっていることを確認します。そして外部構成が正常にインポートされたかどうか確認します。デフォルトでは、外部構成のインポートは使用不可に設定されています。使用不可に設定されていると、ディスク上のいかなる外部構成もクリアされます。ディスク上のデータも、RAID 構成がクリアされたときに失われます。

関連コマンド

[raid foreign-config clear](#) を参照してください。

例

RAID コントローラーの外部構成をインポートします。

```
Console> raid foreign-config import
CWZBR02804I: Import of foreign configuration is enabled. Foreign configurations will be
imported at reboot.
```

親トピック: [特殊目的コマンド](#)

raid foreign-config clear

目的

外部構成のインポートを使用不可に設定します。外部構成はリブート時にクリアされます。外部構成のインポートは 9005 でサポートされています。外部構成とは、ドライブの交換セットに既に存在する RAID 構成です。ドライブの外部構成は、クリアするか、またはドライブが取り付けられているアプライアンスにインポートすることができます。デフォルトでは、9005 マシンで外部構成が見つかると、それはクリアされます。

構文

```
raid foreign-config clear
```

パラメーター

なし。

関連コマンド

[raid foreign-config import](#) を参照してください。

例

RAID コントローラーの外部構成のインポートを使用不可に設定します。

```
Console> raid foreign-config clear
CWZBR02805I: Import of foreign configuration is disabled. Foreign configurations will be
cleared at reboot.
```

親トピック: [特殊目的コマンド](#)

request force-suspend

目的

グリッドへのネットワーク接続を維持しながら、グリッド・アクティビティーを突然中断するよう要求します。ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、ネットワーク接続は使用可能なままです。

構文

request force-suspend

オプション

なし。

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[request resume](#) および [request suspend](#) を参照してください。

例

```
Console> request force-suspend
Force Stopped all XC-10 processes
Forced recycle of appliance
[Mon Feb 25 2013 09:46:59] request force-suspend completed
Console> start-progresss
CWZBR02115E: Unknown command "start-progress"
Console> start-progress
DNS 10.42.229.86 10.42.229.86
DNS 10.1.1.105 10.1.1.105
DNS 9.42.94.15 9.42.94.15
SUSPENDED
```

親トピック: [特殊目的コマンド](#)

request resume

目的

以前に **request suspend** または **request force suspend** コマンドのいずれかでグリッド・アクティビティーの中断を要求した場合、このコマンドはグリッド・アクティビティーをリストアします。

構文

request resume

オプション

なし。

パラメーター

なし。

使用上の注意

このコマンドは、アクティビティーを完全にリストアするのに時間がかかります。ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、ネットワーク接続は使用可能なままです。

関連コマンド

[request suspend](#) および [request force-suspend](#) を参照してください。

例

```
Console> request resume
[Mon Feb 25 2013 09:48:09] request resume completed
Console> start-progress
DNS 10.42.229.86 10.42.229.86
DNS 10.1.1.105 10.1.1.105
DNS 9.42.94.15 9.42.94.15
Volume 1 mounted
Volume 2 mounted
STARTING (4% complete)
```

親トピック: [特殊目的コマンド](#)

request suspend

目的

グリッドへのネットワーク接続を維持しながら、グリッド・アクティビティーを安全に中断するよう要求します。ネットワーク接続 (つまり SSH や Telnet) 経由でコマンド行インターフェース (CLI) に接続している場合、ネットワーク接続は使用可能なままです。

構文

request suspend

オプション

なし。

パラメーター

なし。

使用上の注意

なし。

関連コマンド

[request resume](#) および [request force-suspend](#) を参照してください。

例

```
Console> request suspend
Stopped all XC-10 processes
Recycle of appliance
[Mon Feb 25 2013 09:46:59] request suspend completed
Console> start-progresss
CWZBR02115E: Unknown command "start-progress"
Console> start-progress
DNS 10.42.229.86 10.42.229.86
DNS 10.1.1.105 10.1.1.105
DNS 9.42.94.15 9.42.94.15
SUSPENDED
```

親トピック: [特殊目的コマンド](#)

set-dns-search

目的

ドメイン・ネーム・サーバーの検索ドメインを構成します。

構文

```
set-dns-search [domain [... domain ]]
```

パラメーター

domain

ドメイン・ネーム・サーバーの検索ドメイン。

使用上の注意

- ドメインを指定せずに **set-dns-search** コマンドを呼び出すと、リストがクリアされます。
- それぞれのコマンド呼び出しによって、以前に構成されたドメインが置換されます。

関連コマンド

[get-dns-search](#)、[get-ntp-servers](#)、および [set-dns-servers](#) を参照してください。

例

ドメイン・ネーム・サーバーの検索ドメインを構成してから表示します。

```
Console> set-dns-search rtp.raleigh.ibm.com raleigh.ibm.com ibm.com
Ok
Console> get-dns-search
DNS search domains:
    rtp.raleigh.ibm.com
    raleigh.ibm.com
    ibm.com
Console>
```

親トピック: [特殊目的コマンド](#)

set-dns-servers

目的

1つ以上のドメイン・ネーム・サーバーの IP アドレスを構成します。

構文

```
set-dns-servers [server [... server ]]
```

パラメーター

server

ドメイン・ネーム・サーバーの IP アドレス

使用上の注意

- サーバーを指定せずに **set-dns-servers** コマンドを開始すると、リストがクリアされます。
- それぞれのコマンド呼び出しによって、以前に構成されたサーバーが置換されます。

関連コマンド

[get-dns-servers](#)、[get-dns-search](#)、および [set-dns-search](#) を参照してください。

例

ドメイン・ネーム・サーバーを構成してから表示します。

```
Console> set-dns-servers 9.42.106.2 2002:92a:8f7a:106:9:42:106:42
Ok
Console> get-dns-servers
Domain (DNS) servers:
    9.42.106.2
    2002:92a:8f7a:106:9:42:106:42
Console>
```

親トピック: [特殊目的コマンド](#)

set-ntp-servers

目的

Network Time Protocol サーバーを構成します。

構文

```
set-ntp-servers [server [... server ]]
```

パラメーター

server

Network Time Protocol サーバーの IP アドレスまたはホスト名

使用上の注意

- サーバーを指定せずに **set-ntp-servers** コマンドを開始すると、リストがクリアされます。
- それぞれのコマンド呼び出しによって、以前に構成されたサーバーが置換されます。

関連コマンド

[get-ntp-servers](#) を参照してください。

例

NTP サーバーを構成してから表示します。

```
Console> set-ntp-servers rtpgsa.ibm.com 9.42.106.44
Ok
Console> get-ntp-servers
NTP servers:    rtpgsa.ibm.com
                9.42.106.44
Console>
```

親トピック: [特殊目的コマンド](#)

show commandref

目的

コマンド行インターフェース (CLI) でサポートされるすべてのコマンドの参照リストを表示します。

構文

show commandref

パラメーター

なし。

使用上の注意

- このコマンドからの出力は、プログラムにより構文解析されることを意図しています。
- 出力は非常に長い場合があるので、シリアル・コンソールからコマンドを使用することは推奨されません。

例

コマンドをリストします。

```
Console> show commandref
<reference>

<commands>
  <entry>
    <name>alias</name>
    <brief>alias [name [value]]/brief>
    <detailed/>
  </entry>
  .....

```

親トピック: [特殊目的コマンド](#)

show components

目的

アプライアンス上のコンポーネントのファームウェア・バージョンを表示します。

構文

show components

パラメーター

なし。

使用上の注意

なし。

例

様々なコンポーネントのファームウェア・バージョンを表示します。

```
Console> show components
BMC Firmware Version: 2.53
ServeRaid Firmware Version: 12.12.0-0039
```

親トピック: [特殊目的コマンド](#)

show locales

目的

locale set コマンドで使用可能な、言語、国、および文字セットの値のリストを表示します。言語と国の有効な組み合わせが、その組み合わせのデフォルトの文字セットと共に表示されます。さらに、サポートされる文字セットの全体リストが表示されます。

構文

show locales

パラメーター

なし。

使用上の注意

一部の言語には複数の方言があります。例えば、中国語 (zh) には簡体字 (Hans) と繁体字 (Hant) の方言があります。中国語 (繁体字) を指定するには、言語 zh_Hant を使用します。アプライアンスとの通信に使用されるクライアント・ソフトウェアでサポートされている文字セットを指定するように注意してください。

関連コマンド

なし。

例

locale set コマンドで使用可能なロケール値をリストします。

```
Console> show locales
CWZBR02866I: af
CWZBR02866I: af NA UTF-8
CWZBR02866I: af ZA UTF-8
CWZBR02866I: agq
CWZBR02866I: agq CM UTF-8
...
CWZBR02866I: zh TW BIG5
CWZBR02866I: zu
CWZBR02866I: zu ZA UTF-8
CWZBR02867I: supported charsets: Big5 EUC-JP EUC-KR GB18030 GB2312 IBM00858 IBM850
IBM857 IBM860
IBM861 IBM862 IBM863 IBM864 IBM865 IBM866 IBM868 IBM869 ISO-2022-JP ISO-2022-JP-1 ISO-
2022-JP-2
ISO-2022-KR ISO-8859-1 ISO-8859-10 ISO-8859-13 ISO-8859-15 ISO-8859-2 ISO-8859-3 ISO-
8859-4
ISO-8859-5 ISO-8859-6 ISO-8859-7 ISO-8859-8 ISO-8859-9 KOI8-R KSC_5601 Shift_JIS US-
ASCII UTF-16
UTF-16BE UTF-16LE UTF-32 UTF-7 UTF-8 cp1363 cp851 macintosh x-mac-centraleurroman x-
mac-cyrillic
x-mac-greek x-mac-turkish
```

親トピック: [特殊目的コマンド](#)

show log

目的

ログ・ファイルのリスト、またはログ・ファイルの内容を表示します。

構文

- ログ・ファイルのリストを表示する: **show log list**
- ログ・ファイルの内容を表示する: **show log filename**

パラメーター

filename

コンソールにリストされるファイルの名前

使用上の注意

- IBM サポートに問題を報告するには、[platform must-gather](#) コマンドを使用してください。
- ネットワーク問題が原因で集めておくべき出力を取得できない場合に、このコマンドを使用してログをシリアル・コンソールにリストすることができます。しかし、シリアル・ポートは速度が遅く、ログ・ファイルは非常に大きいことがあるのに注意してください。この方法でリストするには長時間かかることがあります。

例

表示可能なログ・ファイルをリストし、次にデフォルトのログ・ファイルの内容をリストします。

```
Console> show log list
default-log
default-trace
Console> show log default-log
2012-02-17T19:25:43+00:00 CWZBR00028 [info][firmwaremgr(3929)]: The Firmware Manager has
started.
2012-02-17T19:25:43+00:00 CWZBR00031 [debug][firmwaremgr(3929)]: The Firmware Manager was
created.
...
```

親トピック: [特殊目的コマンド](#)

show timezones

目的

timezone set コマンドに使用可能な時間帯のリストを表示します。

構文

show timezones

パラメーター

なし。

例

timezone set コマンドに使用可能な時間帯の値をリストします。

```
Console> show timezones
EST5EDT
PST8PDT
...
```

親トピック: [特殊目的コマンド](#)

show version

目的

アプライアンスでアクティブなファームウェア・バージョンを表示します。

構文

show version

パラメーター

なし。

例

アクティブなファームウェア・バージョンを表示します。

```
Console> show version
Installation date: Mar 9, 2012 8:06:27 PM
Platform version: 4.9.4.0
Platform build ID: build12-20120309-1307
Platform build date: 2012-03-09 18:09:44+00:00
Machine type/model: 923572X
Serial number: 68A0553
Firmware type: Development
Console>
```

親トピック: [特殊目的コマンド](#)

sshkey fingerprint

目的

アプライアンスの SSH サーバーにより使用されるサーバー鍵を表示します。

構文

sshkey fingerprint

パラメーター

なし。

例

サーバーの SSH 鍵を表示します。

```
Console> sshkey fingerprint
SSH RSA Key fingerprint 10:c6:c6:5c:44:72:d1:e9:84:15:df:65:47:96:1a:c1
SSH DSA Key fingerprint cd:a9:8e:a8:0b:d0:45:ce:ac:9e:67:92:06:cf:63:1b
Console>
```

親トピック: [特殊目的コマンド](#)

sshkey reset

目的

アプライアンスの SSH サーバーにより使用されるサーバー鍵を削除して再生成します。

構文

sshkey reset

パラメーター

なし。

関連コマンド

なし。

例

サーバーの SSH 鍵をリセットします。

```
Console> sshkey fingerprint
CWZBR02154I: SSH RSA Key fingerprint 10:c6:c6:5c:44:72:d1:e9:84:15:df:65:47:96:1a:c1
CWZBR02154I: SSH DSA Key fingerprint cd:a9:8e:a8:0b:d0:45:ce:ac:9e:67:92:06:cf:63:1b
Console> sshkey reset
CWZBR02196I: Ok
Console> sshkey fingerprint
CWZBR02154I: SSH RSA Key fingerprint 7c:43:c7:36:2e:4a:ed:2d:a9:af:62:cc:44:f0:a5:7b
CWZBR02154I: SSH DSA Key fingerprint c3:fb:08:aa:c8:81:81:57:f1:ff:9b:c3:70:df:4a:bd
Console>
```

親トピック: [特殊目的コマンド](#)

sshkey verify

目的

アプライアンスの SSH サーバーにより使用されるサーバー鍵が強いか弱いかを判断します。

構文

sshkey verify

パラメーター

なし。

関連コマンド

なし。

例

サーバーの SSH 鍵の強度を検証し、最適な強度でない場合にはリセットします。

```
Console> sshkey verify
CWZBR02797E: SSH keys are not optimally strong
Console> sshkey reset
CWZBR02196I: Ok
Console> sshkey verify
CWZBR02798I: Host SSH keys are strong
Console>
```

親トピック: [特殊目的コマンド](#)

start-progress

目的

始動の進行中にこのコマンドを発行できます。このコマンドは、完了した始動プロセスのパーセントを表示します。

構文

start-progress

なし。

パラメーター

なし。

関連コマンド

なし。

例

```
Console> start-progress
DNS 9.42.139.134 9.42.139.134
DNS 9.42.139.139 9.42.139.139
DNS 9.42.139.152 xsa23e2.rtp.raleigh.ibm.com
DNS 9.42.139.142 localhost
CWXSA0014I: Volume 1 mounted
CWXSA0014I: Volume 2 mounted
CWXSA0006I: Catalog server started.
CWXSA0018I: Grid configuration service started.
CWXSA0017I: Container server 02 started.
CWXSA0017I: Container server 07 started.
CWXSA0017I: Container server 04 started.
CWXSA0017I: Container server 06 started.
CWXSA0017I: Container server 08 started.
CWXSA0017I: Container server 05 started.
CWXSA0017I: Container server 01 started.
CWXSA0017I: Container server 03 started.
CWXSA0002I: Grid administrative service started.
CWXSA0004I: Administrative console started.
STARTED
```

親トピック: [特殊目的コマンド](#)

status battery

目的

予想されるバッテリーの有効期限を表示します。

構文

status battery

パラメーター

なし。

使用上の注意

- バッテリーは、アプライアンスが電源オフで AC 電源に接続していない間、ハードウェア構成と RAID ディスク書き込み可能キャッシュにバックアップ電源を提供します。バッテリーは特定の期間有効です。このコマンドは、その期間の開始と終了を表示します。

関連コマンド

[status battery](#) も参照してください。

例

予想されるバッテリーの有効期限を表示します。

```
Console> status battery
Battery installed: Feb 17, 2012 7:25:45 PM
Battery expires: Feb 16, 2014 7:25:45 PM
Console>
```

親トピック: [特殊目的コマンド](#)

status cpu-usage

このコマンドは推奨されません。代わりに、**status cpu-utilization** を使用してください。

親トピック: [特殊目的コマンド](#)

status cpu-utilization

目的

過去 10 秒間、1 分間、10 分間、1 時間、および 1 日間のインターバルの CPU 使用率を表示します。

構文

```
status cpu-utilization
```

パラメーター

なし。

使用上の注意

なし。

例

CPU 使用率を表示します。

```
Console> status cpu-utilization
CPU utilization over time:
0% over 10 seconds
49% over 1 minute
25% over 10 minutes
14% over 1 hour
6% over 1 day
Console>
```

親トピック: [特殊目的コマンド](#)

status dynamic-tunnels

目的

現在アクティブな動的 IPsec トンネルの詳細を表示します。

構文

```
status dynamic-tunnels [tunnelID]
```

パラメーター

tunnelID

表示する特定の動的トンネルのオプションの ID (Ynnn の形式)。トンネル ID が指定されていない場合、既存の動的 IPsec トンネルすべてが表示されます。

使用上の注意

この状況プロバイダーにより戻されたフィールドについての詳細は、「ネットワーク・セキュリティー構成 (Network Security Configuration)」を参照してください。

例

動的トンネル Y2 の状況を表示します。

```
Console> status dynamic-tunnels y2
Dynamic tunnel name: Y2
  AssociatedFiltCode: n/a
  AssociatedFiltCodeRange: n/a
  AssociatedFiltDestPort: n/a
  AssociatedFiltDestPortRange: n/a
  AssociatedFiltProtocol: ALL(0)
  AssociatedFiltSrcPort: n/a
  AssociatedFiltSrcPortRange: n/a
  AssociatedFiltType: n/a
  AssociatedFiltTypeRange: n/a
  AuthAlgorithm: HMAC-SHA1
  AuthInboundSpi: 4258054942 (0xFDCCC31E)
  AuthOutboundSpi: 796243132 (0x2F75B4BC)
  CurrentTime: 2012/06/01 15:27:14
  DiffieHellmanGroup: 14
  EncryptInboundSpi: 4258054942 (0xFDCCC31E)
  EncryptOutboundSpi: 796243132 (0x2F75B4BC)
  Generation: 1
  HowActivated: OnDemand
  HowToAuth: ESP
  HowToEncap: Transport
  HowToEncrypt: 3DES-CBC
  IKEVersion: 2.0
  InboundBytes: 192
  InboundPackets: 3
  IpFilterRule: tcpcs-ipsec-ifr
  KeyLength: n/a
  Lifesize: 0K
  LifesizeRefresh: 0K
  LifetimeExpires: 2012/06/01 19:17:54
  LifetimeRefresh: 2012/06/01 19:13:05
  LocalAddressBase: 9.42.90.6
  LocalAddressPrefix: n/a
  LocalAddressRange: n/a
  LocalDynVpnRule: n/a
  LocalEndPoint: 9.42.90.6
  OutboundBytes: 192
  OutboundPackets: 3
  PFS: Yes
  ParentIKETunnelID: K1
  PendingNewActivation: n/a
  RemoteAddressBase: 9.42.105.155
  RemoteAddressPrefix: n/a
  RemoteAddressRange: n/a
  RemoteEndPoint: 9.42.105.155
  RmtIpSpecExIDPayload: n/a
```



```
State: DONE  
TunnelID: Y2  
VpnActionName: tcpcs-idva  
Console>
```

親トピック: [特殊目的コマンド](#)

status fan

目的

ファンの速度と状況を表示します。

構文

status fan

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームやブレード・ベースのプラットフォームには適用されません。

例

ファンの状況を表示します。

```
Console> status fan
Fan Speed Tray 1 Fan 1: 4800 (ok)
Fan Speed Tray 1 Fan 2: 5840 (ok)
Fan Speed Tray 1 Fan 3: 0.000000 (alert)
Fan Speed Tray 1 Fan 4: 6960 (ok)
Fan Speed Tray 2 Fan 1: 4720 (ok)
Fan Speed Tray 2 Fan 2: 5520 (ok)
Fan Speed Tray 2 Fan 3: 4720 (ok)
Fan Speed Tray 2 Fan 4: 6080 (ok)
Hard Disk Tray Fan 1: 4720 (ok)
Hard Disk Tray Fan 2: 6000 (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status flash

目的

フラッシュ・ドライブのサイズと使用可能なスペースを表示します。

構文

status flash

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームには適用されません。

例

フラッシュ・メモリー使用状況を表示します。

```
Console> status flash
Flash drive status:
    Total size = 3908 MB, free size = 1894 MB
Console>
```

親トピック: [特殊目的コマンド](#)

status intrusion

目的

物理的なアプライアンスのケースの健全性状況を表示します。

構文

status intrusion

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームやブレード・ベースのプラットフォームには適用されません。

関連コマンド

[device intrusion clear](#) を参照してください。

例

ケースの健全性状況を表示します。

```
Console> status intrusion
Case has not been opened and is secure
Console>
```

親トピック: [特殊目的コマンド](#)

status memory

目的

RAM メモリー使用状況を表示します。

構文

status memory

パラメーター

なし。

例

メモリー使用状況を表示します。

```
Console> status memory
Memory size 8193636K, free 7696564K
Console>
```

親トピック: [特殊目的コマンド](#)

status nvdimm

目的

nvDIMM の状態を表示します。

構文

```
status nvdimm
```

パラメーター

なし。

例

nvDIMM 状況を表示します。

```
Console> status nvdimm
CWZBR02911I: All nvDIMMs are ready to go
Console>
```

親トピック: [特殊目的コマンド](#)

status power-supply

目的

電源機構ユニットの状況を表示します。

構文

status power-supply

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームやブレード・ベースのプラットフォームには適用されません。

例

電源機構ユニットの状況を表示します。

```
Console> status power-supply
Power Supply 1 Output Failure: true (alert)
Power Supply 2 Output Failure: false (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status raid all

目的

RAID カード・コントローラーによりコントロールされる物理ディスクと論理ディスクの詳細を表示します。

構文

```
status raid all
```

パラメーター

なし。

使用上の注意

- この論理ディスク情報はまだサポートされないため、物理ディスクの詳細のみが表示されます。

関連コマンド

[status volume](#) および [status raid physical](#) を参照してください。

例

RAID カード・コントローラーによりコントロールされるすべての物理ディスクと論理ディスクを表示します。

```
Console> status raid all
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286968d50000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE2CE1F
Position: HDD0
  Controller ID: 1
  Device ID: 9
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286970090000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE28HV9
Console>
```

親トピック: [特殊目的コマンド](#)

status raid battery

目的

RAID BBU 情報を表示します。

構文

status raid battery

パラメーター

なし。

使用上の注意

BBU 情報は、MegaRAID コントローラーのあるアプライアンスでのみ使用可能で、MPT コントローラーのあるものでは使用できません。

例

RAID BBU 情報を表示します。

```
Console> status raid battery
CWZBR02808I: BBU status for Adapter '0':
CWZBR02809I:  Battery Type                : iBBU
CWZBR02810I:  Voltage                      : 4055 mV
CWZBR02811I:  Current                      : 0 mA
CWZBR02812I:  Temperature                  : 29 C
CWZBR02816I: BBU Firmware Status:
CWZBR02819I:  Charging Status              : None
CWZBR02820I:  Voltage                      : OK
CWZBR02821I:  Temperature                  : OK
CWZBR02822I:  Learn Cycle Requested       : No
CWZBR02823I:  Learn Cycle Active          : No
CWZBR02824I:  Learn Cycle Status         : OK
CWZBR02825I:  Learn Cycle Timeout        : No
CWZBR02826I:  I2c Errors Detected        : No
CWZBR02827I:  Battery Pack Missing       : No
CWZBR02828I:  Battery Replacement required : No
CWZBR02829I:  Remaining Capacity Low     : No
CWZBR02830I:  Periodic Learn Required    : No
CWZBR02831I:  Transparent Learn         : No
CWZBR02832I:  No space to cache offload   : No
CWZBR02833I:  Pack is about to fail. Should be replaced : No
CWZBR02834I:  Cache Offload premium feature required : No
CWZBR02835I:  Module microcode update required : No
CWZBR02847I: BBU Capacity Info:
CWZBR02848I:  Relative State of Charge    : 98 %
CWZBR02849I:  Absolute State of Charge   : 96 %
CWZBR02850I:  Remaining Capacity         : 1163 mAh
CWZBR02851I:  Full Charge Capacity       : 1187 mAh
CWZBR02853I:  Run time to empty (min)         : Battery is not being
discharged
CWZBR02855I:  Average time to empty (min)       : Battery is not being
discharged
CWZBR02857I:  Average Time to full (min)      : Battery is not being
charged
CWZBR02858I:  Cycle Count                     : 21
CWZBR02859I:  Max Error                       : 6 %
CWZBR02860I:  Remaining Capacity Alarm         : 120 mAh
CWZBR02861I:  Remaining Time Alarm            : 10 Min
CWZBR02836I: BBU Design Info:
CWZBR02837I:  Device Name                     : 3150301
CWZBR02838I:  Serial Number                   : 4010
CWZBR02839I:  Manufacturer Name               : LS1121001A
CWZBR02840I:  Date of Manufacture             : 07-03-2010
CWZBR02841I:  Design Capacity                 : 1215 mAh
CWZBR02843I:  Design Voltage                  : 3700 mV
CWZBR02844I:  Transparent Learn               : 0
CWZBR02845I:  Specification Info              : 33
CWZBR02846I:  Pack Stat Configuration        : 0x6490
Console>
```

親トピック: [特殊目的コマンド](#)

status raid physical

目的

RAID カード・コントローラーによりコントロールされる物理ディスクの詳細を表示します。

構文

status raid physical

パラメーター

なし。

使用上の注意

- 表示される詳細情報は、様々な RAID カード・コントローラーによって異なります。
- *ロケーション* または *位置* 情報は、アプライアンス上のディスクの物理ロケーションを示します。アプライアンスのハード・ディスク・トレイまたはフロント・パネル上で、このロケーション情報とともにテキスト・マッピングがあります。

関連コマンド

[status volume](#) および [status raid all](#) を参照してください。

例

RAID カード・コントローラーによりコントロールされるすべての物理ディスクを表示します。

```
Console> status raid physical
Raid physical disk status:
Position: HDD1
  Controller ID: 1
  Device ID: 8
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286968d50000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE2CE1F
Position: HDD0
  Controller ID: 1
  Device ID: 9
  Array ID: 1
  Logical drive ID: 1
  Logical drive name: raid0
  State: online
  Progress percent: 0
  Raw size: 286102
  Normalized size: 285148
  Interface type: SAS
  Interface speed: 6.0Gb/s
  SAS address: 5000c500286970090000000000000000
  Vendor ID: SEAGATE
  Product ID: ST9300603SS
  Raid Firmware Version: 12.12.0-0065
  Revision: 0006
  Vendor specific information: 3SE28HV9
Console>
```

親トピック: [特殊目的コマンド](#)

status voltage

目的

電圧の測定値と状況を表示します。

構文

status voltage

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- 仮想プラットフォームには適用されません。

例

電圧の測定値を表示します。

```
Console> status voltage
Voltage +12: 11.8 (ok)
Voltage +5 Standby: 4.90 (ok)
Voltage +5: 4.87 (ok)
Voltage +3.3: 3.33 (ok)
Voltage +1.8: 1.79 (ok)
Voltage +1.5: 1.49 (ok)
Voltage CPU1 Core: 1.07 (ok)
Voltage CPU2 Core: 1.08 (ok)
Voltage Bus Termination: 1.08 (ok)
Voltage Battery: 3.14 (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status volume

目的

システムのすべてのボリュームとパーティション情報を表示します。

構文

status volume

パラメーター

なし。

使用上の注意

- このコマンドは、管理対象および管理対象でない両方のタイプのボリュームとパーティションを表示します。
- 管理対象でないパーティションでは、暗号化情報は表示されません。
- 管理対象でないパーティションは、マウント・ポイントが存在しない場合、オフラインとして表示されます。

関連コマンド

[status raid physical](#) および [status raid all](#) を参照してください。

例

すべてのボリュームの詳細を表示します。

```
Console> status volume
Storage Volume Status:
Volume 'raid0' is [Online] : size = 285148 MB, num partitions = 2
  Partition 1 'raid0-1' is [Online], encryption is [None]
    Total size = 91 MB,   free size = 86 MB
  Partition 2 'raid0-2' is [Online], encryption is [None]
    Total size = 184 MB,  free size = 179 MB
Console>
```

親トピック: [特殊目的コマンド](#)

status temperature

目的

物理ハードウェア・コンポーネントの温度状況を表示します。

構文

status temperature

パラメーター

なし。

使用上の注意

- 基本となるハードウェア・プラットフォームによって、表示は異なります。
- このコマンドは仮想プラットフォームには適用されません。

例

温度状況を表示します。

```
Console> status temperature
Power Supply 1 Over-Temperature: false (ok)
Power Supply 1 High Temperature: false (ok)
Power Supply 2 Over-Temperature: false (ok)
Power Supply 2 High Temperature: false (ok)
Temperature System 1: 30.0 (ok)
Temperature System 2: 25.0 (ok)
Temperature CPU1: 27.0 (ok)
Temperature CPU2: 30.0 (ok)
Temperature Memory 00: 45.0 (ok)
Temperature Memory 10: 41.0 (ok)
Temperature Memory 20: 39.0 (ok)
Temperature Memory 30: 39.0 (ok)
Temperature Ethernet Ambient: 29 (ok)
Temperature Ethernet MAC: 50 (ok)
Console>
```

親トピック: [特殊目的コマンド](#)

status uptime

目的

アプライアンスがブートされてからの時間の長さを表示します。

構文

status uptime

パラメーター

なし。

例

アプライアンスが稼働している時間の長さを表示します。

```
Console> status uptime  
13 days 17 hours 13 minutes 15 seconds  
Console>
```

親トピック: [特殊目的コマンド](#)

timezone get

目的

構成された時間帯を表示します。

構文

```
timezone get
```

パラメーター

なし。

使用上の注意

- なし。

関連コマンド

[timezone set](#)、[datetime get](#)、および [datetime set](#) を参照してください。

例

構成された時間帯を表示します。

```
Console> timezone get
Timezone is EST
Console>
```

親トピック: [特殊目的コマンド](#)

timezone set

目的

地方時の時間帯を設定します。

構文

```
timezone set zone
```

パラメーター

zone

時間帯の値

使用上の注意

- なし。

関連コマンド

[timezone get](#)、[datetime get](#)、および [datetime set](#) を参照してください。

例

時間帯を東部標準時 (EST) に設定して、設定を確認します。

```
Console> timezone set EST
Console> timezone get
Timezone is EST
Console>
```

親トピック: [特殊目的コマンド](#)

unalias

目的

構成されたコマンド行インターフェース (CLI) コマンド別名を削除します。

構文

unalias *name*

パラメーター

name

削除されるコマンド行インターフェース (CLI) コマンドの代替名

関連コマンド

[alias](#) を参照してください。

例

構成された別名を表示し、それから test-connectivity という別名を削除します。

```
Console> alias
alias test-connectivity "net-test ping 9.42.106.2"
Console> unalias test-connectivity
Console> alias
No aliases defined
Console>
```

親トピック: [特殊目的コマンド](#)

usage

目的

config コマンドの使用方法の詳細を表示します。

構文

```
config usage
```

パラメーター

なし

使用上の注意

なし

例

config コマンドの詳細を表示します。

```
config usage
```

親トピック: [特殊目的コマンド](#)

user add

目的

新規ユーザーをアプライアンスに対して定義します。

構文

```
user add username [ defaultpass ]
```

パラメーター

username

追加されるユーザー名。ユーザー名は英数字、またはドット(.)、ダッシュ(-)、アンダースコア(_) またはプラス(+)
の特殊文字で構成する必要があります。

defaultpass

新規ユーザー名に割り当てるデフォルトのパスワード

使用上の注意

- 初回のコマンド呼び出しでパスワードが提供されない場合、コマンド行インターフェース (CLI) はパスワードとパスワードの確認を求めるプロンプトを出します。

関連コマンド

[user list](#)、[user delete](#)、[user password](#) を参照してください。

例

ユーザー fred および barney を追加します。

```
Console> user add fred
enter default password:*****
confirm password:*****
Ok
Console> user add barney rubble
Ok
Console> user list
User names:
    admin
    barney
    fred
Console>
```

親トピック: [特殊目的コマンド](#)

user delete

目的

アプリケーションからユーザーを削除します。

構文

```
user delete username
```

パラメーター

username

削除されるユーザー名

関連コマンド

[user add](#) および [user list](#) も参照してください。

例

ユーザー fred を削除します。

```
Console> user list
User names:
    admin
    fred

Console> user delete fred
Ok
Console> user list
User names:
    admin
Console>
```

親トピック: [特殊目的コマンド](#)

user known-hosts list

目的

現行ユーザーの既知のホストをリストします。ユーザーがコマンド `file get|put scp://user@host/dir` を発行すると、ホストの公開鍵がローカル・ファイルに保存されます。このコマンドを使用して、そのファイルに保存されるホストをリストします。

構文

`user known-hosts list`

パラメーター

なし。

関連コマンド

例

現在ログインしているユーザーの既知のホストをリストします。

```
Console> user known-hosts list
aaa.example.com,192.168.10.200
bbb.example.com,192.168.11.201
Console>
```

親トピック: [特殊目的コマンド](#)

user known-hosts delete

目的

現行ユーザーの既知のホスト・ファイルから、指定されたホスト用の保管された公開鍵ファイルを削除します。ユーザーがコマンド `file get|put scp://user@host/dir` を発行すると、ホストの公開鍵がローカル・ファイルに保存され、ホストの ID を後続の `file get|put scp://` コマンドで検証するのに使用されます。ホストが鍵を変更すると、その検証が失敗することがあります。このコマンドを使用して既知のホストに保管された公開鍵を削除し、ホストの新しい鍵を知ることができます。

構文

```
user known-hosts delete hostname
```

パラメーター

hostname

削除される既知のホスト・エントリーのホスト名または IP アドレス

関連コマンド

[user known-hosts list](#) を参照してください。

例

既知のホスト `aaa.example.com` の公開鍵を、現在ログインしているユーザーの既知のホストのリストから削除します。

```
<p>Console> user known-hosts delete aaa.example.com  
CWZBR02196I: Ok  
Console></p>
```

親トピック: [特殊目的コマンド](#)

user list

目的

アプライアンスに定義されたユーザーをリストします。

構文

user list

パラメーター

なし。

関連コマンド

[user add](#) および [user delete](#) を参照してください。

例

現在定義されているユーザーをリストします。

```
Console> user list
User names:
    admin
    fred
Console>
```

親トピック: [特殊目的コマンド](#)

user password

目的

現在アクティブなユーザーのパスワードを変更します。

構文

```
user password oldpass newpass
```

パラメーター

oldpass

古い (現在アクティブな) パスワード

newpass

新規パスワード

使用上の注意

- 初回のコマンド呼び出しで古いパスワードまたは新規パスワードが提供されない場合、コマンド行インターフェース (CLI) は適切なパスワードと新規パスワードの確認を求めるプロンプトを出します。

関連コマンド

[user add](#) を参照してください。

例

パスワードを 3 回変更します。

```
Console> user password aaaa bbbb
Changing password...
Ok
Console> user password bbbb
new password:****
confirm password:****
Changing password...
Ok
Console> user password
current password:****
new password:*****
confirm password:*****
Changing password...
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

user sshkey add

目的

現行ユーザーによる使用のために、指定された URL から SSH 公開鍵ファイルを取得します。

構文

```
user sshkey add url
```

パラメーター

url

公開鍵ファイルの URL

使用上の注意

- 鍵ファイルが保管されると、対応する秘密鍵のある SSH クライアントは、パスワードを指定せずにそのユーザーとして SSH でアクセスすることができます。
- サポートされる公開鍵はアプライアンス・ユーザーごとに 1 つのみです。後続の **user sshkey add** コマンドは、そのアプライアンス・ユーザーの公開鍵ファイルを置換します。

関連コマンド

[user sshkey delete](#) を参照してください。

例

現在ログインしているユーザーの公開鍵を取得します。

```
Console> user sshkey add scp://user1@system1.rtp.raleigh.ibm.com:~/mypublickey
user1@system1.rtp.raleigh.ibm.com's password:
mypublickey                               100% 411      0.4KB/s   00:00
Wrote 411 bytes to local storage
Adding ssh public key file for user wilma
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

user sshkey delete

目的

現行ユーザーの保管された SSH 公開鍵ファイルを削除します。

構文

```
user sshkey delete
```

パラメーター

なし。

関連コマンド

[user sshkey add](#) を参照してください。

例

現在ログインしているユーザーの公開鍵を削除します。

```
Console> user sshkey delete
Deleting ssh public key file for user wilma
Ok
Console>
```

親トピック: [特殊目的コマンド](#)

IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification

Packages	
com.ibm.websphere.objectgrid	These are the main application APIs for the ObjectGrid.
com.ibm.websphere.objectgrid.client	This package contains the primary interfaces and classes for setting behaviors for ObjectGrid clients for this process.
com.ibm.websphere.objectgrid.config	This package contains the interfaces and a factory class for creating ObjectGrid configuration objects programatically.
com.ibm.websphere.objectgrid.continuousquery	
com.ibm.websphere.objectgrid.continuousquery.exception	
com.ibm.websphere.objectgrid.continuousquery.filter	These are the Filter implementations which are used to build continuous query definitions.
com.ibm.websphere.objectgrid.management	This package contains the interfaces for all ObjectGrid MBeans.
com.ibm.websphere.objectgrid.plugins	These are the interfaces for adding plugins to the Grid core framework.
com.ibm.websphere.objectgrid.plugins.builtins	This package contains built-in plugins for ObjectGrid core.
com.ibm.websphere.objectgrid.plugins.index	
com.ibm.websphere.objectgrid.plugins.io.annotations	
com.ibm.websphere.objectgrid	This package has the class MapPermission and class AdminPermission which represents the permissions for to access the ObjectGrid maps and

security	ObjectGrid administration respectively.
com.ibm.websphere.objectgrid.security.config	This package contains the ObjectGrid client security configurations.
com.ibm.websphere.objectgrid.security.plugins	This package contains the interfaces for adding plug-ins to the ObjectGrid security framework and associated Exception classes.
com.ibm.websphere.objectgrid.security.plugins.builtins	This package contains the built-in implementation for the security plugins.
com.ibm.websphere.objectgrid.spring	This package holds the Spring specific APIs for ObjectGrid.
com.ibm.websphere.xs.ra	These are the eXtreme Scale Resource Adapter APIs that allows integration with a Java EE application.
com.ibm.websphere.xsa	

[Overview](#)
[Package](#)
[Class Tree](#)
[Serialized](#)
[Deprecated](#)
[Index](#)
[Help](#)

**IBM WebSphere® DataPower®
 XC10 Appliance
 Release 2.5 Client API
 Specification**

[PREV](#)
[NEXT](#)
[FRAMES](#)
[NO FRAMES](#)
[All Classes](#)

Package com.ibm.websphere.objectgrid

These are the main application APIs for the ObjectGrid.

See:

[Description](#)

Interface Summary	
BackingMap	This is the public interface to the BackingMap.
CatalogDomainInfo	Identifies the configuration attributes of a catalog service domain.
CatalogDomainManager	Provides access to catalog domain configuration information for the current environment.
ClientClusterContext	This interface is a context to represent which cluster/domain the client connected to using one of the ObjectGridManager.connect methods.
ClientReplicableMap	Deprecated. <i>The client replicated map function is deprecated in version 8.6.</i>
IObjectGridException	This interface is used to ensure JDK 1.4 Throwable chaining behavior for all exceptions thrown by ObjectGrid even when an earlier JDK is used (e.g.
JavaMap	This interface is a handle to a named Map.
ObjectGrid	This object is used for creating sessions to the ObjectGrid.
ObjectGridManager	ObjectGridManager is responsible for creating or retrieving local ObjectGrid instances and connecting to distributed ObjectGrid servers.
ObjectMap	This is a handle to a named Map.
PartitionManager	This interface will be used for calculating the proper partition for a given input key.
Session	This interface represents a session container for ObjectMaps.
StateManager	The StateManager can be used to retrieve the availability state of an ObjectGrid.
TxID	This interface is an opaque identifier for a transaction.

Class Summary	
AvailabilityState	Each shard in a distributed ObjectGrid has an availability state associated with it.
ClientReplicableMapMode	Client Replication mode
CopyMode	This class is used to define the "copy" mode when the setCopyMode method of the BackingMap interface is used.
LockStrate	LockStrategy provides an enumerated type idiom for use on the

gy	BackingMap.setLockStrategy(LockStrategy) method.
ObjectGridManagerFactory	This factory class is a high level helper class to get ObjectGridManager instances.
TTLType	Every BackingMap in ObjectGrid has a built in timed based evictor that is referred to as "time to live" evictor or TTL evictor.

Enum Summary

ObjectMap.PutMode	Identifies the operation mode of the ObjectMap.put(Object, Object) , ObjectMap.putAll(Map) , JavaMap.put(Object, Object) and JavaMap.putAll(Map) methods.
Session.TransactionCommitProtocol	The commit protocols that can be used to commit the Session's transaction

Exception Summary

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException	This exception is thrown when a method call to the Client/Server TransactionCallback detects the user is attempting to perform a write against multiple maps in different Map Sets, Partition Sets or Replication groups.
ConnectException	This exception is used to indicate that the client was unable to connect to the server
DuplicateKeyException	A DuplicateKeyException exception is thrown if a key cannot be inserted into a BackingMap because an object with the same key already exists.
KeyNotFoundException	Normally, record not found means a null is returned.
LockDeadlockException	This exception is used by the lock manager to indicate that it detected a deadlock.
LockException	A general locking exception indicating something went wrong with locking operations.
LockInternalFailureException	This exception is used by the lock manager to indicate it detected some internal programming error while processing a lock or unlock request.
LockTimeoutException	This exception is used by the lock manager to indicate that the maximum wait time for a lock has been exceeded.
NoActiveTransactionException	An exception indicating there is no active transaction.
ObjectGridException	Base exception class for all checked exceptions thrown by the ObjectGrid product.
ObjectGrid	

<u>RuntimeException</u>	This exception is the base class for all runtime exceptions thrown by the cache.
<u>ReadOnlyException</u>	This exception is thrown when an attempt is made to modifying operations on a read only maps.
<u>ReplicationVotedToRollbackTransactionException</u>	This exception is thrown when a transaction was rolled back because some/all of the replicas failed to apply the transaction when in synchronous replication mode.
<u>SessionNotReentrantException</u>	A Session object can only be used by a single thread concurrently to perform map operations.
<u>TargetNotAvailableException</u>	A TargetNotAvailableException indicates the ObjectGrid target is not available.
<u>TransactionAffinityException</u>	This exception is thrown for inflight transaction when server fails over.
<u>TransactionAlreadyActiveException</u>	An exception indicating a transaction is already active for the current session.
<u>TransactionException</u>	A general transaction exception indicating something went wrong with a transaction.
<u>TransactionQuiesceException</u>	This exception is thrown when partition/shard/mapset/replication group/replication group member/server/cluster/objectgrid is entered quiesce process for various reasons such as shard movement, partition relocation, system update, server shutdown, and others.
<u>TransactionTimeoutException</u>	This exception is thrown when a transaction exceeds the transaction timeout that was specified on the ObjectGrid or Session.
<u>UnavailableServiceException</u>	This exception is thrown when all servers are dead or when all services are unavailable even though servers are running.
<u>UndefinedMapException</u>	This exception indicates that the map which an application tries to access is not defined in the ObjectGrid.

Package **com.ibm.websphere.objectgrid** Description

These are the main application APIs for the ObjectGrid. The main interface here is the ObjectGrid interface. A JVM needs to create at least one instance.

Introduction

The WebSphere ObjectGrid is designed as a data caching tier that can be used to hold data from multiple sources and then make it available to the clients of the ObjectGrid. The clients access the data through the ObjectGrid APIs. The ObjectGrid is designed to be able to store large quantities of data.

Programming Tutorial

The following sections show snippets on the usage of the ObjectGrid APIs.

Obtaining a ObjectGrid instance.

The application needs to construct an ObjectGrid reference first. An application can choose to make several ObjectGrid instances. Each instance is independent, however, and has its own configuration file. For now, use the following code and programmatically initialize it using the setter methods on the ObjectGrid.

```
ObjectGrid objectGrid = new ObjectGridImpl();
```

The instance can then have a Map defined on it using the following snippet:

```
BackingMap bm = objectGrid.defineMap("TABLE1");
```

Again, setter methods on BackingMap allow it to be configured once it's defined.

Working with an ObjectGrid, Sessions

Each thread that wants to access the ObjectGrid must have its own Session instance. The ObjectGrid class has a getSession method that returns one. Once the thread has a Session then it can obtain ObjectMap instances for manipulating data in the ObjectGrid as well as use the begin/commit/rollback methods on the Session to handle transactions.

```
Session session = objectGrid.getSession();
ObjectMap table1 = session.getMap("TABLE1");
session.begin();
MyData d = (MyData)table1.get("key1");
session.commit();
```

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Package	Next Package	Frames	No Frames	All Classes				

Package com.ibm.websphere.objectgrid.client

This package contains the primary interfaces and classes for setting behaviors for ObjectGrid clients for this process.

See:

[Description](#)

Interface Summary

ClientProperties	The set of properties used to define various preference for ObjectGrid clients.
----------------------------------	---

Package com.ibm.websphere.objectgrid.client Description

This package contains the primary interfaces and classes for setting behaviors for ObjectGrid clients for this process.

Overview

The interfaces in this package should not be implemented directly but are used by the [ClientClusterContext](#) to set default behaviors for application clients for an ObjectGrid instance.

The properties available for use are defined in the [ClientProperties](#) interface.

There are two ways to configure client properties:

1. Create a properties file named `objectGridClient.properties` and store it in the root of your classpath.
2. Create a properties file on your file system in the directory where the client is started from named `objectGridClient.properties`.
3. Create a properties file with any path and name and use the following system property to detect it: `-Dcom.ibm.websphere.objectgrid.ClientProperties=<fileName>`
4. Create a properties file with any path and name and set load it programmatically and pass url to `ClientClusterContext.getClientProperties(ogname, url)`.
5. Programmatically define the properties the `ClientProperties` set methods.

In the following example we set the proximity routing defaults for all clients that use this `ClientClusterContext`:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ClientClusterContext ccc = ogMgr.connect(...);
ClientProperties props = ccc.getClientProperties("myOGName");
props.setPreferLocalHost(true);
props.setPreferLocalProcess(true);
props.setPreferZones(new String[]{"New York", "Texas"});

// The ClientProperties are now applied to the ObjectGrid client connection:
ObjectGrid og=ogMgr.get(ccc, "myOGName");
```

The following example uses a custom client properties file:

```
ClientClusterContext ccc = ogMgr.connect(...);
```

```
URL clientPropsURL = Thread.currentThread().getContextClassLoader().getResource("etc/myObjectGridClient.properties");
ClientProperties props = ccc.setClientProperties("myOGName", clientPropsURL);

// The ClientProperties are now applied to the ObjectGrid client connection:
ObjectGrid og=ogMgr.get(ccc, "myOGName");
```

The following file is an example of a properties file that matches the preceding API:

```
preferLocalProcess=true
preferLocalhost=true
preferZones=New York,Texas
```

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.client

Interface ClientProperties

public interface **ClientProperties**

The set of properties used to define various preference for ObjectGrid clients.

See the [package summary](#) for details on how to use the ClientProperties class and properties file.

Since:

WAS XD 6.1.0.3, XC10

Field Summary	
s t a t i c S t r i n g	<p>CLIENT_PROPS_FILE_PATH_KEY The system property key to override the location of the client properties file.</p>
s t a t i c S t r i n g	<p>DEFAULTCLIENTPROPERTYFILE The default name of client property file</p>
s t a t i c S t r i n g	<p>PROP_LISTENER_HOST Listener host property key for the client properties file.</p>
s t a t i c S t r i n g	<p>PROP_LISTENER_PORT Listener port property key for the client properties file.</p>

r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

PROP_MAXIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

PROP_MAXIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.

PROP_MINIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

PROP_MINIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.

PROP_PREFER_LOCAL_HOST

Currently, this property is not used.

PROP_PREFER_LOCAL_PROCESS

Currently, this property is not used.

n g	
s t a t i c S t r i n g	<p>PROP_PREFER_ZONES Prefer zones property key for the client properties file.</p>
s t a t i c S t r i n g	<p>PROP_REQUEST_RETRY_TIMEOUT The requestRetryTimeout which indicates how long to retry a request (in milliseconds).</p>
s t a t i c S t r i n g	<p>PROP_SHUFFLE_BOOTSTRAP_ADDRESSES The shuffleBootstrapAddresses property is used to determine if the catalog service grid addresses should be randomized when used by a client when bootstrapping to the grid.</p>
s t a t i c S t r i n g	<p>PROP_XIO_REQUEST_TIMEOUT The xioRequestTimeout indicates how long the eXtreme IO transport will wait for cross-process call to complete.</p>
s t a t i c S t r i n g	<p>PROP_XIO_TIMEOUT Sets the timeout for server requests using the eXtremeIO transport.</p>

Method Summary

S t r i n g	<p>getListenerHost() Retrieves the host to be used by the ORB.</p>

i n t	getListenerPort() Retrieves the port to be used by the ORB.
i n t	getMaximumXIONetworkThreads() Gets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.
i n t	getMaximumXIOWorkerThreads() Sets the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.
i n t	getMinimumXIONetworkThreads() Gets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.
i n t	getMinimumXIOWorkerThreads() Retrieves the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.
S t r i n g []	getPreferZones() Retrieve the preferred zones.
l o n g	getRequestRetryTimeout() Retrieves the current request retry timeout.
i n t	getXIORequestTimeout() Gets the current timeout that is allowed for requests to servers over the XIO transport to complete.
i n t	getXIOTimeout() Returns the current timeout for server requests using the XIO transport.
b o o l e a n	isBootStrapListShuffled() Retrieves the value of the BootStrapListShuffled property.
b o o l e a n	isPreferLocalHost() This method is reserved for future use.
b o o l e a n	isPreferLocalProcess() This method is reserved for future use.
v o i d	setBootStrapListShuffled(boolean shuffle) Sets the value of the BootStrapListShuffled property.

v o i d	setMaximumXIONetworkThreads (int maxThreads) Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.
v o i d	setMaximumXIOWorkerThreads (int maxThreads) Retrieves the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.
v o i d	setMinimumXIONetworkThreads (int minThreads) Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.
v o i d	setMinimumXIOWorkerThreads (int minThreads) Sets the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.
v o i d	setPreferLocalHost (boolean localHost) This method is reserved for future use.
v o i d	setPreferLocalProcess (boolean localProcess) This method is reserved for future use.
v o i d	setPreferZones (String[] zones) Prefer routing to specific zones.
v o i d	setRequestRetryTimeout (long requestRetryTimeout) Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions.
v o i d	setXIORequestTimeout (int timeout) Sets the current timeout that is allowed for requests to servers over the XIO transport to complete.
v o i d	setXIOTimeout (int timeout) Sets the current timeout for server requests using the XIO transport.

Field Detail

DEFAULTCLIENTPROPERTYFILE

static final [String](#) DEFAULTCLIENTPROPERTYFILE

The default name of client property file

See Also:

[Constant Field Values](#)

CLIENT_PROPS_FILE_PATH_KEY

static final [String](#) CLIENT_PROPS_FILE_PATH_KEY

The system property key to override the location of the client properties file.

Since:

7.0

See Also:

[Constant Field Values](#)

PROP_PREFER_LOCAL_PROCESS

static final [String](#) PROP_PREFER_LOCAL_PROCESS

Currently, this property is not used. It is reserved for future use.

See Also:

[setPreferLocalProcess\(boolean\)](#), [Constant Field Values](#)

PROP_PREFER_LOCAL_HOST

static final [String](#) PROP_PREFER_LOCAL_HOST

Currently, this property is not used. It is reserved for future use.

See Also:

[setPreferLocalHost\(boolean\)](#), [Constant Field Values](#)

PROP_PREFER_ZONES

static final [String](#) PROP_PREFER_ZONES

Prefer zones property key for the client properties file. Each specified zone is separated by a comma in the form: preferZones=ZoneA,ZoneB,ZoneC

See Also:

[setPreferZones\(String\[\]\)](#), [Constant Field Values](#)

PROP_REQUEST_RETRY_TIMEOUT

static final [String](#) PROP_REQUEST_RETRY_TIMEOUT

The requestRetryTimeout which indicates how long to retry a request (in milliseconds). A 0 indicates that the request should fail fast and skip over in internal retry logic. Exceptions that cannot succeed even if tried again such as DuplicateException will be returned immediately.

Since:

7.0

See Also:

[setRequestRetryTimeout\(long\)](#), [Constant Field Values](#)

PROP_XIO_REQUEST_TIMEOUT

static final [String](#) PROP_XIO_REQUEST_TIMEOUT

The xioRequestTimeout indicates how long the eXtreme IO transport will wait for cross-process call to complete. The value is expressed in milliseconds. The default value is 30,000 or 30 seconds. When custom tuning client side retry of eXtreme Scale operations, this value will determine how long a single network operation is given, and then the

PROP_REQUEST_RETRY_TIMEOUT / requestRetryTimeout will control how long those individual network operations are retried.

Since:

8.6.0.2, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_LISTENER_HOST

static final [String](#) PROP_LISTENER_HOST

Listener host property key for the client properties file.

Since:

XS 7.1

See Also:

[getListenerHost\(\)](#), [Constant Field Values](#)

PROP_LISTENER_PORT

static final [String](#) PROP_LISTENER_PORT

Listener port property key for the client properties file.

Since:

XS 7.1

See Also:

[getListenerPort\(\)](#), [Constant Field Values](#)

PROP_SHUFFLE_BOOTSTRAP_ADDRESSES

static final [String](#) PROP_SHUFFLE_BOOTSTRAP_ADDRESSES

The shuffleBootstrapAddresses property is used to determine if the catalog service grid addresses should be randomized when used by a client when bootstrapping to the grid. The default value of the property is true.

Since:

7.1.0.3

See Also:

[Constant Field Values](#)

PROP_XIO_TIMEOUT

static final [String](#) PROP_XIO_TIMEOUT

Sets the timeout for server requests using the eXtremeIO transport. The timeout is set in seconds. The default xioTimeout for server requests is set to 30 seconds. The valid range is timeout >= 1.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MAXIMUM_XIO_NETWORK_THREAD_POOL_SIZE

static final [String](#) PROP_MAXIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MAXIMUM_XIO_WORKER_THREAD_POOL_SIZE

static final [String](#) PROP_MAXIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MINIMUM_XIO_NETWORK_THREAD_POOL_SIZE

static final [String](#) PROP_MINIMUM_XIO_NETWORK_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

PROP_MINIMUM_XIO_WORKER_THREAD_POOL_SIZE

static final [String](#) PROP_MINIMUM_XIO_WORKER_THREAD_POOL_SIZE

Sets the minimum number of threads to allocate in the eXtremeIO transport request processing thread pool.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

Method Detail

setPreferZones

void setPreferZones([String](#) [] zones)

Prefer routing to specific zones.

When zones are enabled on an ObjectGrid, requests will be routed to the specified zones.

Parameters:

zones - array of zone names. If null or an empty array, then requests are routed to all zones.

setPreferLocalProcess

void **setPreferLocalProcess**(boolean localProcess)

This method is reserved for future use. Calls to the method will not result in any performed operation.

Parameters:

localProcess -

setPreferLocalHost

void **setPreferLocalHost**(boolean localHost)

This method is reserved for future use. Calls to the method will not result in any performed operation.

Parameters:

localHost -

getPreferZones

[String](#)[] **getPreferZones**()

Retrieve the preferred zones.

Returns:

the preferred zones.

isPreferLocalProcess

boolean **isPreferLocalProcess**()

This method is reserved for future use. The returned value should be ignored by the user.

Returns:

false

isPreferLocalHost

boolean **isPreferLocalHost**()

This method is reserved for future use. The returned value should be ignored by the user.

Returns:

false

setRequestRetryTimeout

void **setRequestRetryTimeout**(long requestRetryTimeout)

Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions. A request will timeout when either the request timeout expires or the transaction timeout expires, whichever expires first.

A value of 0 indicates that all requests should fail immediately and avoid any retry logic.

Exceptions that cannot succeed even if tried again such as `DuplicateKeyException` exceptions will be thrown immediately.

A value of -1 indicates that the request retry timeout is not set, meaning that the request duration is governed by the transaction timeout.

The request retry timeout can be overridden using the [Session.setRequestRetryTimeout\(long\)](#) method.

Parameters:

`requestRetryTimeout` - the duration in milliseconds retry a client request, 0 if the request should fail immediately or -1 if the request timeout is not set.

Since:

7.0

See Also:

[Session.setRequestRetryTimeout\(long\)](#), [ObjectGrid.setTxTimeout\(int\)](#)

getRequestRetryTimeout

long `getRequestRetryTimeout()`

Retrieves the current request retry timeout. Returns -1 if it was not set.

Returns:

`requestRetryTimeout` in milliseconds, 0 to fail immediately or -1 if not set.

Since:

7.0

getListenerHost

String `getListenerHost()`

Retrieves the host to be used by the ORB. The listener host property defaults to 'localhost'. This property can only be set in the `client.properties` file.

Returns:

The host that the ORB will bind to.

Since:

7.1

getListenerPort

int `getListenerPort()`

Retrieves the port to be used by the ORB. The listener port property defaults to the `corbaloc` port, 2809. This property can only be set in the `client.properties` file.

Returns:

The port that the ORB will bind to.

Since:

7.1

isBootStrapListShuffled

boolean `isBootStrapListShuffled()`

Retrieves the value of the `BootStrapListShuffled` property.

Returns:

true if the value of `BootStrapListeShuffled` was set to true. false if the value of `BootStrapListeShuffled` was set to false.

Since:

7.1.0.3

setBootStrapListShuffled

```
void setBootStrapListShuffled(boolean shuffle)
```

Sets the value of the `BootStrapListShuffled` property.

Parameters:

`shuffle` - true the bootstrap list will be shuffled providing each client a random distribution of catalog servers to select from. false the first viable address in the list of catalog servers will be used.

Since:

7.1.0.3

getMinimumXIOWorkerThreads

```
int getMinimumXIOWorkerThreads()
```

Retrieves the minimum number of threads to allocate in the `eXtremeIO` transport request processing thread pool.

Returns:

the minimum number of threads.

Since:

8.6, XC10 2.5

setMinimumXIOWorkerThreads

```
void setMinimumXIOWorkerThreads(int minThreads)
```

Sets the minimum number of threads to allocate in the `eXtremeIO` transport request processing thread pool.

Parameters:

`minThreads` - the minimum number of threads.

Since:

8.6, XC10 2.5

getMaximumXIOWorkerThreads

```
int getMaximumXIOWorkerThreads()
```

Sets the maximum number of threads to allocate in the `eXtremeIO` transport request processing thread pool.

Returns:

the maximum number of threads.

Since:

8.6, XC10 2.5

setMaximumXIOWorkerThreads

```
void setMaximumXIOWorkerThreads(int maxThreads)
```

Retrieves the maximum number of threads to allocate in the eXtremeIO transport request processing thread pool.

Parameters:

maxThreads - the maximum number of threads.

Since:

8.6, XC10 2.5

getMinimumXIONetworkThreads

```
int getMinimumXIONetworkThreads()
```

Gets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

Returns:

the minimum number of threads

Since:

8.6, XC10 2.5

setMinimumXIONetworkThreads

```
void setMinimumXIONetworkThreads(int minThreads)
```

Sets the minimum number of threads to allocate in the eXtremeIO transport network thread pool.

Parameters:

minThreads - the minimum number of threads

Since:

8.6, XC10 2.5

getMaximumXIONetworkThreads

```
int getMaximumXIONetworkThreads()
```

Gets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

Returns:

the maximum number of threads.

Since:

8.6, XC10 2.5

setMaximumXIONetworkThreads

```
void setMaximumXIONetworkThreads(int maxThreads)
```

Sets the maximum number of threads to allocate in the eXtremeIO transport network thread pool.

Parameters:

maxThreads - the maximum number of threads.

Since:

8.6, XC10 2.5

getXIOTimeout

int `getXIOTimeout()`

Returns the current timeout for server requests using the XIO transport.

Returns:

the current timeout in seconds

Since:

8.6, XC10 2.5

setXIOTimeout

void `setXIOTimeout(int timeout)`

Sets the current timeout for server requests using the XIO transport. The timeout is set in seconds. The valid range is `timeout >= 1`.

Parameters:

`timeout` - the timeout in seconds

Since:

8.6, XC10 2.5

getXIORequestTimeout

int `getXIORequestTimeout()`

Gets the current timeout that is allowed for requests to servers over the XIO transport to complete. The timeout is set in milliseconds.

Since:

8.6.0.2, XC10 2.5

setXIORequestTimeout

void `setXIORequestTimeout(int timeout)`

Sets the current timeout that is allowed for requests to servers over the XIO transport to complete. The timeout is set in milliseconds. The valid range is greater than 0.

Parameters:

`timeout` - the timeout in milliseconds

Since:

8.6.0.2, XC10 2.5

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package **com.ibm.websphere.objectgrid.config**

This package contains the interfaces and a factory class for creating ObjectGrid configuration objects programatically.

See:

[Description](#)

Interface Summary	
BackingMapConfiguration	A BackingMapConfiguration object can be used to override BackingMap settings on the client side.
ConfigProperty	ConfigProperty can be used to attach properties to a Plugin.
ObjectGridConfiguration	An ObjectGridConfiguration object can be used to override ObjectGrid plugins on the client side.
Plugin	This interface represents an ObjectGrid or BackingMap plugin.
PluginType	Every Plugin has a PluginType.

Class Summary	
ConfigPropertyType	ConfigPropertyType is used to set the type of an attribute on a Plugin.
ObjectGridConfigFactory	This is the configuration factory for ObjectGrid configuration entities.
QueryConfig	This QueryConfig represents a schema configuration for a QueryManager.
QueryMapping	A QueryMapping maps a Java class to a BackingMap and allows a map to be included in a query.
QueryRelationship	A QueryRelationship represents a relationship between two BackingMap value classes.

Exception Summary	
ObjectGridConfigurationException	Thrown when a problem with the current configuration is found.

Package **com.ibm.websphere.objectgrid.config** Description

This package contains the interfaces and a factory class for creating ObjectGrid configuration objects programatically. The main use of this is by the objectgrid client to override serverside configuration.

Overview

ObjectGridManagerFactory has static methods to create the configuration objects. Using these configuration objects in conjunction with ObjectGridManager methods

- `setOverrideObjectGridConfigurations(Map)`
- `putOverrideObjectGridConfigurations(String, List)`

to override client configuration, before connecting to the objectgrid server.

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
ew	ge		ed	ted				
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.config

Class QueryRelationship

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.config.QueryRelationship

All Implemented Interfaces:

[Serializable](#)

```
public class QueryRelationship
extends Object
implements Serializable
```

A QueryRelationship represents a relationship between two BackingMap value classes. A BackingMap must have one class type defined in the value part of the map. A relationship can be established between two maps by mapping the source and target map's value classes.

The cardinality of the relationship is automatically determined by the type of the attribute field.

A relationship requires two classes, a relationship field, and optionally an inverse relationship field.

For example: Two entities; Department and Employee, have the following bi-directional relationship:

1. One department has many employees. the collection field in the Department class is "emps"
2. An employee belongs to one department

```
public class Department {
    private int id;
    private Collection emps;

    public void setEmps(Collection emps) {
        this.emps = emps;
    }

    public Collection getEmps() {
        return emps;
    }
    ...
}
```

```
public class Employee {
    private int id;
    private Department dept;

    public void setDept(Department dept) {
        this.dept = dept;
    }

    public Department getDept() {
        return dept;
    }
}
```

```
}
```

Use the following method call to establish this bi-directional relationship.

```
queryConfig.addRelationship(new QueryRelationship(  
    Department.class.getName(), Employee.class.getName(), "emps", "dept"));
```

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary	
QueryRelationship (String sourceClass, String targetClass, String relationshipField, String invRelationshipField)	Constructor for creating a QueryRelationship instance.

Method Summary	
boolean	equals (Object o)
String	getInvRelationshipField () Retrieve the inverse relationship attribute name of a bi-directional relationship.
String	getRelationshipField () Retrieve the name of the attribute in the source class that references the key of the target class.
String	getSourceClass () Retrieve the name of the class representing the source of a relationship.
String	getTargetClass () Retrieve the name of the class representing the target of a relationship.
int	hashCode ()
void	setInvRelationshipField (String invRelationshipField) Set the inverse relationship attribute name of a bi-directional relationship.

v o i d	setRelationshipField (String relationshipField) Set the name of the attribute in the source class that references the key of the target class.
v o i d	setSourceClass (String sourceClass) Set the name of the class representing the source of a relationship.
v o i d	setTargetClass (String targetClass) Set the name of the class representing the target of a relationship.
S t r i n g	toString ()

Methods inherited from class [java.lang.Object](#)

[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

QueryRelationship

```
public QueryRelationship(String sourceClass,
                        String targetClass,
                        String relationshipField,
                        String invRelationshipField)
```

Constructor for creating a QueryRelationship instance.

The sourceClass, targetClass, and relationshipField must not be null.

Parameters:

sourceClass - the source class of the relationship

targetClass - the target class of the relationship

relationshipField - the attribute in the source class that references the key of the target class.

invRelationshipField - the attribute in the target class that references the key of the source class. This value is null if a bi-directional relationship does not exist.

Method Detail

getInvRelationshipField

```
public String getInvRelationshipField()
```

Retrieve the inverse relationship attribute name of a bi-directional relationship.

Returns:

the attribute name of the inverse side of a bi-directional relationship or null if the relationship is uni-directional.

setInvRelationshipField

```
public void setInvRelationshipField(String invRelationshipField)
```

Set the inverse relationship attribute name of a bi-directional relationship.

Parameters:

invRelationshipField - the attribute name of the inverse side of a bi-directional relationship or null if the relationship is uni-directional.

getRelationshipField

```
public String getRelationshipField()
```

Retrieve the name of the attribute in the source class that references the key of the target class.

Returns:

the name of the relationship attribute.

setRelationshipField

```
public void setRelationshipField(String relationshipField)
```

Set the name of the attribute in the source class that references the key of the target class.

Parameters:

relationshipField - the name of the relationship attribute.

getSourceClass

```
public String getSourceClass()
```

Retrieve the name of the class representing the source of a relationship.

Returns:

the source class

setSourceClass

```
public void setSourceClass(String sourceClass)
```

Set the name of the class representing the source of a relationship.

Parameters:

sourceClass - the source class

getTargetClass

```
public String getTargetClass()
```

Retrieve the name of the class representing the target of a relationship.

Returns:

the target class

setTargetClass

public void **setTargetClass**([String](#) targetClass)

Set the name of the class representing the target of a relationship.

Parameters:

targetClass - the target class

equals

public boolean **equals**([Object](#) o)

Overrides:

[equals](#) in class [Object](#)

See Also:

[Object.equals\(java.lang.Object\)](#)

hashCode

public int **hashCode**()

Overrides:

[hashCode](#) in class [Object](#)

See Also:

[Object.hashCode\(\)](#)

toString

public [String](#) **toString**()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Class QueryMapping

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.config.QueryMapping

All Implemented Interfaces:

[Serializable](#)

```
public class QueryMapping
extends Object
implements Serializable
```

A QueryMapping maps a Java class to a BackingMap and allows a map to be included in a query. It also indicates whether the query engine should use a getter method or direct field access to access fields in the value class.

For example, class Department is the value class that is stored in the "DepartmentMap" BackingMap and the key is an Integer.

```
public class Department {
    private int id;
    private Collection emps;

    public void setEmps(Collection emps) {
        this.emps = emps;
    }

    public Collection getEmps() {
        return emps;
    }
    ...
}
```

The QueryMapping would be created as follows:

```
...
QueryConfig queryConfig = new QueryConfig();
queryConfig.addMapping(new QueryMapping(
    "DepartmentMap", Department.class.getName(), "id", QueryMapping.PROPERTY_ACCESS)
objectGrid.setQueryConfig(queryConfig);
...
```

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Field Summary

s
t
a

t i c	FIELD_ACCESS This constant indicates to use direct field access to read the field values
i n t	
s t a t i c	PROPERTY_ACCESS This constant indicates to use JavaBean property-style get methods to read the field values from the Java object stored in the BackingMap.
i n t	

Constructor Summary	
i n t	QueryMapping () Default constructor.
i n t	QueryMapping (String mapName, String valueClass, String primaryKeyField) Constructor for creating a basic QueryMapping instance with a default access type of PROPERTY_ACCESS .
i n t	QueryMapping (String mapName, String valueClass, String primaryKeyField, int accessType) Constructor for creating a QueryMapping instance.

Method Summary	
b o o l e a n	equals (Object o)
i n t	getAccessType () Retrieve the method in which the query engine will access the value class object stored in the BackingMap.
S t r i n g	getMapName () Retrieve the BackingMap name associated with this mapping.
S t r i n g	getPrimaryKeyField () Retrieve the name of the attribute in the value class object that is also the primary key of the BackingMap.
S t r i n g	getValueClass () Retrieve the type of object that is stored in the BackingMap.
i n t	hashCode ()

v o i d	setAccessType (int accessType) Set the method in which the query engine will access the value class object stored in the BackingMap.
v o i d	setMapName (String mapName) Set the BackingMap name associated with this mapping.
v o i d	setPrimaryKeyField (String primaryKeyField) Set the name of the attribute in the value class object that is also the primary key of the BackingMap.
v o i d	setValueClass (String valueClass) Set the type of object that is stored in the BackingMap.
S t r i n g	toString ()

Methods inherited from class [java.lang.Object](#)

[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

FIELD_ACCESS

```
public static final int FIELD_ACCESS
```

This constant indicates to use direct field access to read the field values

See Also:

[setAccessType\(int\)](#), [getAccessType\(\)](#), [Constant Field Values](#)

PROPERTY_ACCESS

```
public static final int PROPERTY_ACCESS
```

This constant indicates to use JavaBean property-style get methods to read the field values from the Java object stored in the BackingMap. PROPERTY_ACCESS is the default.

See Also:

[setAccessType\(int\)](#), [getAccessType\(\)](#), [Constant Field Values](#)

Constructor Detail

QueryMapping

```
public QueryMapping()
```

Default constructor.

QueryMapping

```
public QueryMapping(String mapName,  
                   String valueClass,  
                   String primaryKeyField)
```

Constructor for creating a basic QueryMapping instance with a default access type of [PROPERTY_ACCESS](#).

The mapName and valueClass must not be null.

Parameters:

mapName - the name of the BackingMap to map
valueClass - the class of object stored in the BackingMap's value.
primaryKeyField - the optional name of the primary key field of the class.

QueryMapping

```
public QueryMapping(String mapName,  
                   String valueClass,  
                   String primaryKeyField,  
                   int accessType)
```

Constructor for creating a QueryMapping instance. The mapName and valueClass must not be null.

Parameters:

mapName - the name of the BackingMap to map
valueClass - the class of object stored in the BackingMap's value.
primaryKeyField - the optional name of the primary key field of the class.
accessType - the method ([PROPERTY_ACCESS](#) or [FIELD_ACCESS](#)) in which the query engine will access the persistent data in the value object.

Method Detail

getMapName

```
public String getMapName()
```

Retrieve the BackingMap name associated with this mapping.

Returns:

the BackingMap name.

setMapName

```
public void setMapName(String mapName)
```

Set the BackingMap name associated with this mapping.

getValueClass

```
public String getValueClass()
```

Retrieve the type of object that is stored in the BackingMap.

Returns:

the object type that is stored in the BackingMap's value.

setValueClass

```
public void setValueClass(String valueClass)
```

Set the type of object that is stored in the BackingMap.

getAccessType

```
public int getAccessType()
```

Retrieve the method in which the query engine will access the value class object stored in the BackingMap.

Returns:

Returns the accessType.

See Also:

[PROPERTY_ACCESS](#), [FIELD_ACCESS](#)

setAccessType

```
public void setAccessType(int accessType)
```

Set the method in which the query engine will access the value class object stored in the BackingMap.

Parameters:

accessType - the accessType.

See Also:

[PROPERTY_ACCESS](#), [FIELD_ACCESS](#)

getPrimaryKeyField

```
public String getPrimaryKeyField()
```

Retrieve the name of the attribute in the value class object that is also the primary key of the BackingMap.

This value is optional.

Returns:

the primaryKeyField.

setPrimaryKeyField

```
public void setPrimaryKeyField(String primaryKeyField)
```

Set the name of the attribute in the value class object that is also the primary key of the BackingMap.

Parameters:

primaryKeyField - the name of the primary key attribute or null if not set.

equals

```
public boolean equals(Object o)
```

Overrides:

[equals](#) in class [Object](#)

See Also:

[Object.equals\(java.lang.Object\)](#)

hashCode

public int hashCode()

Overrides:

[hashCode](#) in class [Object](#)

See Also:

[Object.hashCode\(\)](#)

toString

public [String](#) toString()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Class QueryConfig

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public class QueryConfig
extends Object
implements Serializable
```

This QueryConfig represents a schema configuration for a QueryManager. A QueryConfig object contains a set of QueryMapping objects and a set of QueryRelationship objects.

Users use addQuerySchema method to add a QueryMapping object and use addRelationship method to add a QueryRelationship object into this QueryConfig object.

Since:

WAS XD 6.1, XC10

See Also:

[ObjectGrid.setQueryConfig\(QueryConfig\)](#), [QueryMapping](#), [QueryRelationship](#), [Serialized Form](#)

Constructor Summary

[QueryConfig\(\)](#)
Default constructor.

Method Summary

[addQueryMapping\(QueryMapping mapping\)](#)
Add a QueryMapping to this QueryConfig

[addQueryRelationship\(QueryRelationship relation\)](#)
Add a QueryRelationship to this QueryConfig

[getQueryMappings\(\)](#)
Retrieve all added QueryMappings

Q
u
e
r
y
R
e
l
a
t
i
o
n
s
h
i
p
[
]

[getQueryRelationships\(\)](#)
Retrieve all added QueryRelationships

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

QueryConfig

public [QueryConfig](#)()

Default constructor.

Method Detail

addQueryMapping

public void [addQueryMapping](#)([QueryMapping](#) mapping)

Add a QueryMapping to this QueryConfig

Parameters:

mapping - the QueryMapping to add.

addQueryRelationship

public void [addQueryRelationship](#)([QueryRelationship](#) relation)

Add a QueryRelationship to this QueryConfig

Parameters:

relation - the QueryRelationship to add.

getQueryMappings

public [QueryMapping](#)[] [getQueryMappings](#)()

Retrieve all added QueryMappings

Returns:

array of QueryMapping

getQueryRelationships

```
public QueryRelationship[] getQueryRelationships()
```

Retrieve all added QueryRelationships

Returns:

array of QueryRelationship

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.config

Interface PluginType

All Superinterfaces:

[Serializable](#)

```
public interface PluginType
extends Serializable
```

Every Plugin has a PluginType. The PluginType is specified during Plugin creation.

ObjectGridConfiguration objects support the following PluginTypes for overriding client-side ObjectGrid plugins:

- PluginType.TRANSACTION_CALLBACK
- PluginType.OBJECTGRID_EVENT_LISTENER
- PluginType.COLLISION_ARBITER

BackingMapConfiguration objects support the following PluginTypes for overriding client-side BackingMap plugins:

- PluginType.EVICTOR
- PluginType.MAP_EVENT_LISTENER

Since:

WAS XD 6.0.1.2, XC10

See Also:

[ObjectGridConfigFactory.createPlugin\(PluginType, String\)](#)

Field Summary

s
t
a
t
i
c
P
L
U
G
I
N
T
Y
P
E

[COLLISION_ARBITER](#)

PluginType.COLLISION_ARBITER can be used to attach a CollisionArbiter plugin to an ObjectGrid.

s
t
a
t
i
c
P
L
U
G
I
N
T
Y
P
E

[EVICTOR](#)

PluginType.EVICTOR can be used to attach an [Evictor](#) to a BackingMap.

q
i
n
T
Y
p
e

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[MAP_EVENT_LISTENER](#)

PluginType.MAP_EVENT_LISTENER can be used to attach a [MapEventListener](#) to a BackingMap.

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[MAP_LIFECYCLE_LISTENER](#)

PluginType.MAP_LIFECYCLE_LISTENER can be used to attach a BackingMapLifecycleListener plugin to an BackingMap

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[MAP_SERIALIZER_PLUGIN](#)

PluginType.MAP_SERIALIZER_PLUGIN can be used to attach a MapSerializerPlugin to a BackingMap to serialize keys.

s
t
a
t
i
c

P
L
U
G
I
N
T
Y
P
E

[OBJECTGRID_EVENT_LISTENER](#)

PluginType.OBJECTGRID_EVENT_LISTENER can be used to attach an [ObjectGridEventListener](#) plugin to an ObjectGrid

s
t
a
t

i
c
P
L
U
G
I
N
T
Y
P
E

[OBJECTGRID_LIFECYCLE_LISTENER](#)

PluginType.OBJECTGRID_LIFECYCLE_LISTENER can be used to attach an [ObjectGridLifecycleListener](#) plugin to an ObjectGrid

s
t
a
t
i
c
P
L
U
G
I
N
T
Y
P
E

[TRANSACTION_CALLBACK](#)

PluginType.TRANSACTION_CALLBACK can be used to attach a [TransactionCallback](#) plugin to an ObjectGrid.

Field Detail

OBJECTGRID_EVENT_LISTENER

static final [PluginType](#) OBJECTGRID_EVENT_LISTENER

PluginType.OBJECTGRID_EVENT_LISTENER can be used to attach an [ObjectGridEventListener](#) plugin to an ObjectGrid

TRANSACTION_CALLBACK

static final [PluginType](#) TRANSACTION_CALLBACK

PluginType.TRANSACTION_CALLBACK can be used to attach a [TransactionCallback](#) plugin to an ObjectGrid.

COLLISION_ARBITER

static final [PluginType](#) COLLISION_ARBITER

PluginType.COLLISION_ARBITER can be used to attach a CollisionArbiter plugin to an ObjectGrid.

OBJECTGRID_LIFECYCLE_LISTENER

static final [PluginType](#) OBJECTGRID_LIFECYCLE_LISTENER

PluginType.OBJECTGRID_LIFECYCLE_LISTENER can be used to attach an [ObjectGridLifecycleListener](#) plugin to an ObjectGrid

Since:

7.1.1

EVICTOR

static final [PluginType](#) EVICTOR

`PluginType.EVICTOR` can be used to attach an [Evictor](#) to a `BackingMap`.

MAP_EVENT_LISTENER

static final [PluginType](#) MAP_EVENT_LISTENER

`PluginType.MAP_EVENT_LISTENER` can be used to attach a [MapEventListener](#) to a `BackingMap`.

MAP_SERIALIZER_PLUGIN

static final [PluginType](#) MAP_SERIALIZER_PLUGIN

`PluginType.MAP_SERIALIZER_PLUGIN` can be used to attach a `MapSerializerPlugin` to a `BackingMap` to serialize keys.

Since:

7.1.1

MAP_LIFECYCLE_LISTENER

static final [PluginType](#) MAP_LIFECYCLE_LISTENER

`PluginType.MAP_LIFECYCLE_LISTENER` can be used to attach a `BackingMapLifecycleListener` plugin to an `BackingMap`

Since:

7.1.1

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
OD

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.config

Interface Plugin

All Superinterfaces:

[Serializable](#)

```
public interface Plugin
extends Serializable
```

This interface represents an ObjectGrid or BackingMap plugin. An ObjectGridConfiguration object supports the following Plugins:

- PluginType#OBJECTGRID_EVENT_LISTENER
- PluginType#TRANSACTION_CALLBACK

A BackingMapConfiguration object supports the following Plugins:

- PluginType#EVICTOR
- PluginType#MAP_EVENT_LISTENER

A Plugin object has following attributes:

- pluginType: the support PluginTypes are listed above
- className: the plugin implementation class name. This class name will be used to construct the class object. A default constructor must be implemented.
- configProperties: the JavaBean properties for this plugin implementation object.

A plugin object can be created by using ObjectGridConfigFactory.createPlugin(PluginType, String) method. Please refer to ObjectGridConfigFactory for detailed example.

Since:

WAS XD 6.0.1.2, XC10

Method Summary	
V o i d	addConfigProperty (ConfigProperty configProp) Add a ConfigProperty to this object.
S t r i n g	getClassName () Get the String representation of the class name of this Plugin
L i s t	getConfigProperties () Get the ConfigProperty objects that have been set on this object.
S t r	getOSGiService ()

i n g	Get the OSGi service name configured for this Plugin.
P L U G I N T Y P E	getPluginType() Get the PluginType for this Plugin.
v o i d	setClassName(String className) The class name that is set must be an implementor of the PluginTypeImpl for this Plugin.
v o i d	setConfigProperties(List configPropList) Set the ConfigProperties for this object
v o i d	setOSGiService(String osgiService) Set the OSGi service name configured for this Plugin.
v o i d	setPluginType(PluginType pluginType) Set the PluginType for this Plugin.

Method Detail

addConfigProperty

void **addConfigProperty**([ConfigProperty](#) configProp)

Add a ConfigProperty to this object.

Parameters:

configProp - - ConfigProperty to add to this object

setConfigProperties

void **setConfigProperties**([List](#) configPropList)

Set the ConfigProperties for this object

Parameters:

configPropList -

getConfigProperties

[List](#) **getConfigProperties**()

Get the ConfigProperty objects that have been set on this object.

Returns:

a List of the ConfigProperty objects that have been added to this object.

See Also:

[ConfigProperty](#)

getPluginType

[PluginType](#) `getPluginType()`

Get the PluginType for this Plugin.

Returns:
the PluginType for this Plugin

`setPluginType`

`void setPluginType(PluginType pluginType)`

Set the PluginType for this Plugin.

The ObjectGridConfiguration plugins include

- `PluginType.TRANSACTION_CALLBACK`
- `PluginType.OBJECTGRID_EVENT_LISTENER`

The BackingMapConfiguration plugins include

- `PluginType.EVICTOR`
- `PluginType.MAP_EVENT_LISTENER`

Parameters:
pluginType -

`getClassName`

[String](#) `getClassName()`

Get the String representation of the class name of this Plugin

Returns:
the String representation of the class name

`setClassName`

`void setClassName(String className)`

The class name that is set must be an implementor of the PluginTypeImpl for this Plugin. For example, if the type of this Plugin is PluginType.EVICTOR, then the className must be an implementor of the com.ibm.websphere.objectgrid.plugins.Evictor interface.

Parameters:
className - - the class name of the Class that implements the PluginType

`getOSGiService`

[String](#) `getOSGiService()`

Get the OSGi service name configured for this Plugin. If an OSGi service name is configured for this Plugin, the className configured for this Plugin is ignored.

Returns:
the OSGi service name configured for this Plugin.

Since:
7.1.1

`setOSGiService`

`void setOSGiService(String osgiService)`

Set the OSGi service name configured for this Plugin. If an OSGi service name is configured for this Plugin, the className configured for this Plugin is ignored.

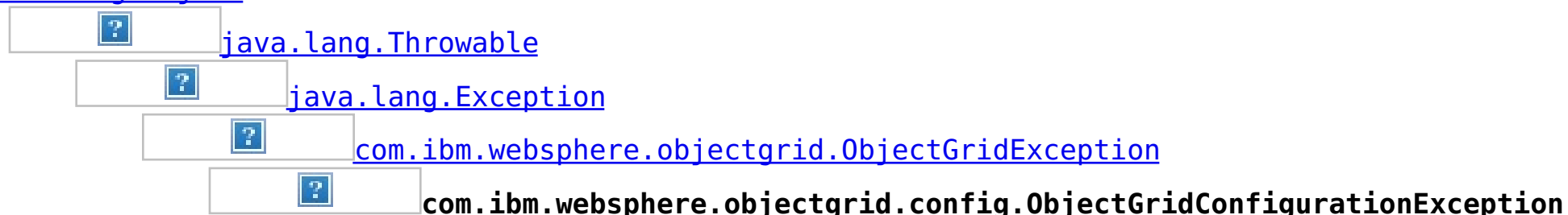
Parameters:
osgiService - the OSGi service name configured for this Plugin.

Since:
7.1.1

com.ibm.websphere.objectgrid.config

Class ObjectGridConfigurationException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class ObjectGridConfigurationException
extends ObjectGridException
```

Thrown when a problem with the current configuration is found. This exception may be thrown when the configuration specified in the deployment policy, ObjectGrid descriptor or security descriptor is incorrect.

Since:

7.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridConfigurationException](#)()

Constructs a new ObjectGridConfigurationException with null as its detail message.

[ObjectGridConfigurationException](#)(String message)

Constructs a new ObjectGridConfigurationException with the specified detail message.

[ObjectGridConfigurationException](#)(String message, [Throwable](#) cause)

Constructs a new ObjectGridConfigurationException with the specified detail message and cause.

[ObjectGridConfigurationException](#)([Throwable](#) cause)

Constructs a new ObjectGridConfigurationException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridConfigurationException

```
public ObjectGridConfigurationException()
```

Constructs a new ObjectGridConfigurationException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ObjectGridConfigurationException

```
public ObjectGridConfigurationException(String message)
```

Constructs a new ObjectGridConfigurationException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridConfigurationException

```
public ObjectGridConfigurationException(Throwable cause)
```

Constructs a new ObjectGridConfigurationException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for ObjectGridConfigurationException that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

ObjectGridConfigurationException

```
public ObjectGridConfigurationException(String message,  
                                         Throwable cause)
```

Constructs a new ObjectGridConfigurationException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in

this ObjectGridConfigurationException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the getMessage method).

cause - the cause (which is saved for later retrieval by the getCause method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Class	Next Class	Frames No Frames All					
			Classes				
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD				

com.ibm.websphere.objectgrid.config

Interface ObjectGridConfiguration

public interface **ObjectGridConfiguration**

An ObjectGridConfiguration object can be used to override ObjectGrid plugins on the client side. The com.ibm.websphere.objectgrid.plugins.ObjectGridEventListener and the com.ibm.websphere.objectgrid.plugins.TransactionCallback Plugins can be overridden.

Since:

WAS XD 6.0.1.2, XC10

See Also:

[ObjectGridEventListener](#), [TransactionCallback](#)

Method Summary	
V o i d	<p>addBackingMapConfiguration(BackingMapConfiguration backingMapConfiguration) Add a BackingMapConfiguration to this ObjectGridConfiguration.</p>
V o i d	<p>addPlugin(Plugin plugin) Add a Plugin to this ObjectGridConfiguration.</p>
L i s t	<p>getBackingMapConfigurations() Get the List of BackingMapConfiguration objects that are attached to this ObjectGridConfiguration object</p>
S t r i n g	<p>getName() Get the name of this ObjectGridConfiguration</p>
L i s t	<p>getPlugins() Get the Plugins that have been attached to this ObjectGridConfiguration.</p>
i n t	<p>getTxIsolation() Retrieves the default transaction isolation level.</p>
i n t	<p>getTxTimeout() Gets the transaction timeout.</p>
V o i d	<p>setBackingMapConfigurations(List backingMapConfigList) Set the BackingMapConfiguration objects for this ObjectGridConfiguration.</p>

void [setPlugins](#)([List](#) pluginList)
Set the Plugins for this ObjectGridConfiguration.

void [setTxIsolation](#)(int level)
Sets the default transaction isolation level for all sessions created by the ObjectGrid.

void [setTxTimeout](#)(int seconds)
Sets the transaction timeout

Method Detail

getName

[String](#) [getName](#)()

Get the name of this ObjectGridConfiguration

Returns:
the name of this ObjectGridConfiguration

addBackingMapConfiguration

void [addBackingMapConfiguration](#)([BackingMapConfiguration](#) backingMapConfiguration)

Add a BackingMapConfiguration to this ObjectGridConfiguration.

Parameters:
backingMapConfiguration -

setBackingMapConfigurations

void [setBackingMapConfigurations](#)([List](#) backingMapConfigList)

Set the BackingMapConfiguration objects for this ObjectGridConfiguration. Any BackingMapConfiguration objects that were previously attached to this ObjectGridConfiguration object will be overridden.

Parameters:
backingMapConfigList - - A List of BackingMapConfiguration objects.

See Also:
[BackingMapConfiguration](#)

getBackingMapConfigurations

[List](#) [getBackingMapConfigurations](#)()

Get the List of BackingMapConfiguration objects that are attached to this ObjectGridConfiguration object

Returns:
a List of BackingMapConfiguration objects

See Also:

addPlugin

void **addPlugin**([Plugin](#) plugin)

Add a Plugin to this ObjectGridConfiguration. The Plugins that can be overridden on a client-side ObjectGrid are `com.ibm.websphere.objectgrid.plugins.ObjectGridEventListener` and `com.ibm.websphere.objectgrid.plugins.TransactionCallback`.

Parameters:

plugin -

See Also:

[setPlugins\(List\)](#), [Plugin](#)

setPlugins

void **setPlugins**([List](#) pluginList)

Set the Plugins for this ObjectGridConfiguration. Any Plugins that were previously attached to this ObjectGridConfiguration object will be overridden.

Parameters:

pluginList - - a List of Plugins

See Also:

[addPlugin\(Plugin\)](#), [Plugin](#)

getPlugins

[List](#) **getPlugins**()

Get the Plugins that have been attached to this ObjectGridConfiguration.

Returns:

a List of Plugin objects

See Also:

[Plugin](#)

setTxTimeout

void **setTxTimeout**(int seconds)

Sets the transaction timeout

Parameters:

seconds -

Since:

7.1.0.3

See Also:

[ObjectGrid.setTxTimeout\(int\)](#)

getTxTimeout

int **getTxTimeout**()

Gets the transaction timeout. The value is in seconds.

Returns:
the transaction timeout in seconds

Since:
7.1.0.3

See Also:
[ObjectGrid.getTxTimeout\(\)](#)

setTxIsolation

void **setTxIsolation**(int level)

Sets the default transaction isolation level for all sessions created by the ObjectGrid. The constants defined in the Session interface are the possible transaction isolation levels. The default is [Session.TRANSACTION_REPEATABLE_READ](#).

Parameters:
level - one of the following Session constants: [Session.TRANSACTION_READ_UNCOMMITTED](#), [Session.TRANSACTION_READ_COMMITTED](#) or [Session.TRANSACTION_REPEATABLE_READ](#) or 0 if the TransactionIsolation should not be set.

Since:
7.1.1

getTxIsolation

int **getTxIsolation**()

Retrieves the default transaction isolation level.

Returns:
the current transaction isolation level.

Since:
7.1.1

See Also:
[setTxIsolation\(int\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

`com.ibm.websphere.objectgrid.config`

Class `ObjectGridConfigFactory`

[java.lang.Object](#)

 `com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory`

```
public final class ObjectGridConfigFactory  
extends Object
```

This is the configuration factory for ObjectGrid configuration entities. Users are expected to use static methods of this factory to create ObjectGrid configuration objects.

Here is a list of static methods used to create configuration objects:

- `createObjectGridConfiguration(String)`: create an `ObjectGridConfiguration` object
- `createBackingMapConfiguration(String)`: create a `BackingMapConfiguration` object by passing a backing map name
- `createConfigProperty(ConfigPropertyType, String, String)`: create a `ConfigProperty` object
- `createPlugin(PluginType, String)`: create a `Plugin` object

Below is an example of creating an ObjectGrid configuration. A `Plugin` is added to the `ObjectGridConfiguration` object: `com.ibm.websphere.objectgrid.plugins.ObjectGridEventListener` plugin.

A `BackingMapConfiguration` called "myBackingMap" is then created and added to the `ObjectGridConfiguration`. This `BackingMapConfiguration` also has an `Evictor Plugin` configured.

Once the `ObjectGridConfiguration` object has been created, it can be used to call either of these methods

- `com.ibm.websphere.objectgrid.ObjectGridManager.putOverrideObjectGridConfigurations(String, List)`
- `com.ibm.websphere.objectgrid.ObjectGridManager.setOverrideObjectGridConfigurations(Map)`

to set configuration objects, prior to connecting.

```
// Create an ObjectGridConfiguration object  
ObjectGridConfiguration ogConfig = ObjectGridConfigFactory.createObjectGridConfiguration(ogName);  
  
// create ObjectGridEventListener plugin  
Plugin eventListener = ObjectGridConfigFactory.createPlugin(PluginType.OBJECTGRID_EVENT_LISTENER,  
"com.ibm.test.MyOgEventListener");  
  
// Add plugin to ObjectGridConfiguration object  
ogConfig.addPlugin(eventListener);  
  
// Create a BackingMapConfiguration object  
BackingMapConfiguration bmConfig = ObjectGridConfigFactory.createBackingMapConfiguration("mybacki  
ngMap");  
  
// Add BackingMapConfiguration object to ObjectGridConfiguration object  
ogConfig.addBackingMapConfiguration(bmConfig);  
  
// Set the number of buckets to 1000  
bmConfig.setNumberOfBuckets(1000);
```

```

// Create a Evictor plugin for this backing map.
Plugin evictor = ObjectGridConfigFactory.createPlugin(
BackingMapConfiguration.PLUGIN_EVICTOR,
com.acme.myEvictorImpl.class.getName());

// add Evictor Plugin to the BackingMapConfiguration
bmConfig.addPlugin(evictor);

// Create a ConfigProperty for the Evictor plugin
ConfigProperty sizeProperty = ObjectGridConfigFactory.createConfigProperty(
    ConfigPropertyType.INT_PRIM, "size", "153");

// Add the ConfigProperty to the Evictor plugin
evictor.setConfigProperty(sizeProperty);

```

Since:

WAS XD 6.0.1.2, XC10

See Also:

[ObjectGridConfiguration](#), [BackingMapConfiguration](#), [Plugin](#), [ConfigProperty](#)

Constructor Summary

[ObjectGridConfigFactory](#)()

Method Summary

[createBackingMapConfiguration](#)([String](#) backingMapConfigName)
 Create a BackingMapConfiguration object

[createConfigProperty](#)([ConfigPropertyType](#) configPropType, [String](#) name, [String](#) value)
 Create a ConfigProperty object for use on a Plugin.

P
r
o
p
e
r
t
y

s
t
a
t
i
c
O
b
j
e
c
t
G
r
i
d
C
o
n
f
i
g
u
r
e
t
i
o
n

[createObjectGridConfiguration](#)(String objectGridConfigName)
Create an ObjectGridConfiguration object.

s
t
a
t
i
c
P
l
u
g
i
n

[createOSGiServicePlugin](#)(PluginType pluginType, String osgiServiceName)
Create an OSGi Service Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

s
t
a
t
i
c
P
l
u
g
i
n

[createPlugin](#)(PluginType pluginType, String className)
Create a Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridConfigFactory

```
public ObjectGridConfigFactory()
```

Method Detail

createPlugin

```
public static Plugin createPlugin(PluginType pluginType,  
                                   String className)
```

Create a Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

Parameters:

pluginType - the PluginType.
className - of the Plugin implementation class to instantiate

Returns:

a Plugin instance

See Also:

[PluginType.OBJECTGRID_EVENT_LISTENER](#), [PluginType.TRANSACTION_CALLBACK](#),
[PluginType.COLLISION_ARBITER](#), [PluginType.EVICTOR](#), [PluginType.MAP_EVENT_LISTENER](#),
[PluginType.OBJECTGRID_LIFECYCLE_LISTENER](#), [PluginType.MAP_LIFECYCLE_LISTENER](#)

createObjectGridConfiguration

```
public static ObjectGridConfiguration createObjectGridConfiguration(String objectGridConfigName)
```

Create an ObjectGridConfiguration object.

Parameters:

objectGridConfigName - the name that will be assigned to this ObjectGridConfiguration object

Returns:

the ObjectGridConfiguration object

createBackingMapConfiguration

```
public static BackingMapConfiguration createBackingMapConfiguration(String backingMapConfigName)
```

Create a BackingMapConfiguration object

Parameters:

backingMapConfigName - the name to assign to this BackingMapConfiguration

Returns:

the BackingMapConfiguration object

createConfigProperty

```
public static ConfigProperty createConfigProperty(ConfigPropertyType configPropType,  
                                                  String name,  
                                                  String value)
```

Create a ConfigProperty object for use on a Plugin.

The Plugin should have a set method that corresponds to the name of this ConfigProperty. The method must accept a parameter of the ConfigPropertyType that is specified on this ConfigProperty. For example, if the name of this ConfigProperty is set to "size", and the type is ConfigPropertyType.INT_PRIM, then the Plugin must have the method setSize(int). The value of the ConfigProperty will be passed to the setter of the Plugin when an ObjectGrid is created based on this configuration.

Parameters:

configPropType - ConfigPropertyType of the ConfigProperty. Part of the set method's signature, the type of parameter the set method requires. Valid name - of the ConfigProperty. It must correspond to the name of a set method on the Plugin.

value - of the ConfigProperty. This value will be passed to the set method on the Plugin.

Returns:

the ConfigProperty object

See Also:

[ConfigPropertyType.STRING_JAVA_LANG](#), [ConfigPropertyType.BOOLEAN_JAVA_LANG](#), [ConfigPropertyType.BOOLEAN_PRIM](#), [ConfigPropertyType.BYTE_JAVA_LANG](#), [ConfigPropertyType.BYTE_PRIM](#), [ConfigPropertyType.CHARACTER_JAVA_LANG](#), [ConfigPropertyType.CHAR_PRIM](#), [ConfigPropertyType.DOUBLE_JAVA_LANG](#), [ConfigPropertyType.DOUBLE_PRIM](#), [ConfigPropertyType.FLOAT_JAVA_LANG](#), [ConfigPropertyType.FLOAT_PRIM](#), [ConfigPropertyType.INTEGER_JAVA_LANG](#), [ConfigPropertyType.INT_PRIM](#), [ConfigPropertyType.LONG_JAVA_LANG](#), [ConfigPropertyType.LONG_PRIM](#), [ConfigPropertyType.SHORT_JAVA_LANG](#)

createOSGiServicePlugin

```
public static Plugin createOSGiServicePlugin(PluginType pluginType,  
                                             String osgiServiceName)
```

Create an OSGi Service Plugin for a BackingMapConfiguration or an ObjectGridConfiguration

Parameters:

pluginType - the PluginType.
osgiServiceName - the OSGi service name

Returns:

a Plugin instance

Since:

7.1.1

See Also:

[PluginType.OBJECTGRID_EVENT_LISTENER](#), [PluginType.TRANSACTION_CALLBACK](#), [PluginType.COLLISION_ARBITER](#), [PluginType.EVICTOR](#), [PluginType.MAP_EVENT_LISTENER](#), [PluginType.OBJECTGRID_LIFECYCLE_LISTENER](#), [PluginType.MAP_LIFECYCLE_LISTENER](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Class	Next Class	Frames	No Frames	All			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid.config
Class ConfigPropertyType

[java.lang.Object](#)



All Implemented Interfaces:
[Serializable](#)

```
public final class ConfigPropertyType
extends Object
implements Serializable
```

ConfigPropertyType is used to set the type of an attribute on a Plugin. The Java primitives, their java.lang counterparts, and java.lang.String are the supported types.

Since:
 WAS XD 6.0.1.2, XC10

See Also:
[ObjectGridConfigFactory.createConfigProperty\(ConfigPropertyType, String, String\)](#),
[Plugin.setPluginType\(PluginType\)](#), [Serialized Form](#)

Field Summary

s t a t i c C o n f i g P r o p e r t y T y p e	<p>BOOLEAN_JAVA_LANG ConfigPropertyType.BOOLEAN_JAVA_LANG can be used to set a property of type java.lang.Boolean on a plugin.</p>
s t a t i c C o n	(Empty field)

f
i
j
p
r
o
p
e
r
t
y
T
Y
P
e

BOOLEAN_PRIM

ConfigPropertyType.BOOLEAN_PRIM can be used to set a property of type boolean on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
y
T
Y
P
e

BYTE_JAVA_LANG

ConfigPropertyType.BYTE_JAVA_LANG can be used to set a property of type java.lang.Byte on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
y
T
Y
P
e

BYTE_PRIM

ConfigPropertyType.BYTE_PRIM can be used to set a property of type byte on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r

CHAR_PRIM

ConfigPropertyType.CHAR_PRIM can be used to set a property of type char on a

r
o
p
e
r
t
i
v
e

plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
i
v
e

CHARACTER_JAVA_LANG

ConfigPropertyType.CHARACTER_JAVA_LANG can be used to set a property of type java.lang.Character on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
i
v
e

DOUBLE_JAVA_LANG

ConfigPropertyType.DOUBLE_JAVA_LANG can be used to set a property of type java.lang.Double on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p

DOUBLE_PRIM

ConfigPropertyType.DOUBLE_PRIM can be used to set a property of type double on a plugin.

e
r
t
i
c
l
e

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
y

FLOAT_JAVA_LANG

ConfigPropertyType.FLOAT_JAVA_LANG can be used to set a property of type java.lang.Float on a plugin.

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
y

FLOAT_PRIM

ConfigPropertyType.FLOAT_PRIM can be used to set a property of type float on a plugin.

s
t
a
t
i
c
c
o
n
f
i
g
p
r
o
p
e
r
t
y

INT_PRIM

ConfigPropertyType.INT_PRIM can be used to set a property of type int on a plugin.

I
V
P
E

s
t
a
t
i
c
C
O
N
F
I
G
P
R
O
P
E
R
T
Y
T
Y
P
E

INTEGER_JAVA_LANG

ConfigPropertyType.INTEGER_JAVA_LANG can be used to set a property of type java.lang.Integer on a plugin.

s
t
a
t
i
c
C
O
N
F
I
G
P
R
O
P
E
R
T
Y
T
Y
P
E

LONG_JAVA_LANG

ConfigPropertyType.LONG_JAVA_LANG can be used to set a property of type java.lang.Long on a plugin.

s
t
a
t
i
c
C
O
N
F
I
G
P
R
O
P
E
R
T
Y
T
Y
P
E

LONG_PRIM

ConfigPropertyType.LONG_PRIM can be used to set a property of type long on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
Y
T
Y
P
e

SHORT_JAVA_LANG

ConfigPropertyType.SHORT_JAVA_LANG can be used to set a property of type java.lang.Short on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
Y
T
Y
P
e

SHORT_PRIM

ConfigPropertyType.SHORT_PRIM can be used to set a property of type short on a plugin.

s
t
a
t
i
c
C
o
n
f
i
g
P
r
o
p
e
r
t
Y
T
Y
P
e

STRING_JAVA_LANG

ConfigPropertyType.STRING_JAVA_LANG can be used to set a property of type java.lang.String on a plugin.

Method Summary

b o o l e a n	equals(Object o)
i n t	getId() Get the raw value of this ConfigPropertyType.
i n t	hashCode()
S t r i n g	toString()
s t a t i c C o n f i g P r o p e r t y T y p e	valueOf(byte id) Given the raw value of a ConfigPropertyType, this method returns a ConfigPropertyType object, or null if the raw value does not match an existing type.

Methods inherited from class java.lang.[Object](#)
[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

INTEGER_JAVA_LANG

public static final [ConfigPropertyType](#) INTEGER_JAVA_LANG

ConfigPropertyType.INTEGER_JAVA_LANG can be used to set a property of type java.lang.Integer on a plugin.

INT_PRIM

public static final [ConfigPropertyType](#) INT_PRIM

ConfigPropertyType.INT_PRIM can be used to set a property of type int on a plugin.

BOOLEAN_JAVA_LANG

public static final [ConfigPropertyType](#) **BOOLEAN_JAVA_LANG**

`ConfigPropertyType.BOOLEAN_JAVA_LANG` can be used to set a property of type `java.lang.Boolean` on a plugin.

BOOLEAN_PRIM

public static final [ConfigPropertyType](#) **BOOLEAN_PRIM**

`ConfigPropertyType.BOOLEAN_PRIM` can be used to set a property of type `boolean` on a plugin.

CHARACTER_JAVA_LANG

public static final [ConfigPropertyType](#) **CHARACTER_JAVA_LANG**

`ConfigPropertyType.CHARACTER_JAVA_LANG` can be used to set a property of type `java.lang.Character` on a plugin.

CHAR_PRIM

public static final [ConfigPropertyType](#) **CHAR_PRIM**

`ConfigPropertyType.CHAR_PRIM` can be used to set a property of type `char` on a plugin.

BYTE_JAVA_LANG

public static final [ConfigPropertyType](#) **BYTE_JAVA_LANG**

`ConfigPropertyType.BYTE_JAVA_LANG` can be used to set a property of type `java.lang.Byte` on a plugin.

BYTE_PRIM

public static final [ConfigPropertyType](#) **BYTE_PRIM**

`ConfigPropertyType.BYTE_PRIM` can be used to set a property of type `byte` on a plugin.

SHORT_JAVA_LANG

public static final [ConfigPropertyType](#) **SHORT_JAVA_LANG**

`ConfigPropertyType.SHORT_JAVA_LANG` can be used to set a property of type `java.lang.Short` on a plugin.

SHORT_PRIM

public static final [ConfigPropertyType](#) **SHORT_PRIM**

`ConfigPropertyType.SHORT_PRIM` can be used to set a property of type `short` on a plugin.

LONG_JAVA_LANG

public static final [ConfigPropertyType](#) LONG_JAVA_LANG

`ConfigPropertyType.LONG_JAVA_LANG` can be used to set a property of type `java.lang.Long` on a plugin.

LONG_PRIM

public static final [ConfigPropertyType](#) LONG_PRIM

`ConfigPropertyType.LONG_PRIM` can be used to set a property of type `long` on a plugin.

FLOAT_JAVA_LANG

public static final [ConfigPropertyType](#) FLOAT_JAVA_LANG

`ConfigPropertyType.FLOAT_JAVA_LANG` can be used to set a property of type `java.lang.Float` on a plugin.

FLOAT_PRIM

public static final [ConfigPropertyType](#) FLOAT_PRIM

`ConfigPropertyType.FLOAT_PRIM` can be used to set a property of type `float` on a plugin.

DOUBLE_JAVA_LANG

public static final [ConfigPropertyType](#) DOUBLE_JAVA_LANG

`ConfigPropertyType.DOUBLE_JAVA_LANG` can be used to set a property of type `java.lang.Double` on a plugin.

DOUBLE_PRIM

public static final [ConfigPropertyType](#) DOUBLE_PRIM

`ConfigPropertyType.DOUBLE_PRIM` can be used to set a property of type `double` on a plugin.

STRING_JAVA_LANG

public static final [ConfigPropertyType](#) STRING_JAVA_LANG

`ConfigPropertyType.STRING_JAVA_LANG` can be used to set a property of type `java.lang.String` on a plugin.

Method Detail

valueOf

public static final [ConfigPropertyType](#) `valueOf`(byte id)

Given the raw value of a `ConfigPropertyType`, this method returns a `ConfigPropertyType` object, or null if the raw value does not match an existing type. This method is used to

deserialize this object.

Parameters:

id - the raw value of a ConfigPropertyType

Returns:

the ConfigPropertyType corresponding to the raw input value

Since:

8.6, XC10 2.5

equals

public boolean equals([Object](#) o)

Overrides:

[equals](#) in class [Object](#)

hashCode

public int hashCode()

Overrides:

[hashCode](#) in class [Object](#)

getId

public int getId()

Get the raw value of this ConfigPropertyType. This method is used to serialize this object.

Returns:

the raw value of this ConfigPropertyType.

toString

public [String](#) toString()

Overrides:

[toString](#) in class [Object](#)

com.ibm.websphere.objectgrid.config

Interface ConfigProperty

All Superinterfaces:

[Serializable](#)

```
public interface ConfigProperty
extends Serializable
```

ConfigProperty can be used to attach properties to a Plugin. A ConfigProperty has the following attributes:

- name: the property name
- value: the property value
- configPropertyType: the configuration property type

This ConfigProperty can be used to set the properties of a plugin. The name of the property should follow JavaBean convention. That is, for every property, there should be a corresponding set method in the plugin class.

Users can use

```
com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory.createConfigProperty(ConfigPropertyType, String, String) to create a ConfigProperty object.
```

```
ConfigProperty evictorNameProp =
ObjectGridConfigFactory.createConfigProperty(ConfigProperty.STRING_JAVA_LANG, "evictorName",
"evictor1"); This creates a property "evictorName" with value "evictor1", and the type is
java.lang.String. Use the
com.ibm.websphere.objectgrid.config.Plugin#addConfigProperty(ConfigProperty) method to attach a
ConfigProperty to a Plugin. When the Plugin is created, each ConfigProperty will have its
corresponding set method called.
```

Continuing with the example above, attach the ConfigProperty to an Evictor Plugin.

```
evictorPlugin.addConfigProperty(evictorNameProp);
```

When this Evictor Plugin is created, the setEvictorName(String) method will be called with the value "evictor1".

Since:

WAS XD 6.0.1.2, XC10

Method Summary

C
o
n
f
i
g
P
r
o
p
e
r
t
y

```
getConfigPropertyType\(\)
    Get the ConfigPropertyType of this object.
```

t y p e	
S t r i n g	getName() Get the name of this object.
S t r i n g	getValue() Get the value that is assigned to this ConfigProperty.
V o i d	setConfigPropertyType(ConfigPropertyType configPropType) Set the ConfigPropertyType for this object.
V o i d	setName(String name) Set the name of this object.
V o i d	setValue(String value) Set the value of this ConfigProperty.

Method Detail

setConfigPropertyType

void **setConfigPropertyType**([ConfigPropertyType](#) configPropType)

Set the ConfigPropertyType for this object. The Java primitives, their java.lang counterparts, and java.lang.String are the supported ConfigPropertyTypes.

Parameters:

configPropType -

See Also:

[ConfigPropertyType.INTEGER_JAVA_LANG](#), [ConfigPropertyType.INT_PRIM](#),
[ConfigPropertyType.BOOLEAN_JAVA_LANG](#), [ConfigPropertyType.BOOLEAN_PRIM](#),
[ConfigPropertyType.CHARACTER_JAVA_LANG](#), [ConfigPropertyType.CHAR_PRIM](#),
[ConfigPropertyType.BYTE_JAVA_LANG](#), [ConfigPropertyType.BYTE_PRIM](#),
[ConfigPropertyType.SHORT_JAVA_LANG](#), [ConfigPropertyType.SHORT_PRIM](#),
[ConfigPropertyType.LONG_JAVA_LANG](#), [ConfigPropertyType.LONG_PRIM](#),
[ConfigPropertyType.FLOAT_JAVA_LANG](#), [ConfigPropertyType.FLOAT_PRIM](#),
[ConfigPropertyType.DOUBLE_JAVA_LANG](#), [ConfigPropertyType.DOUBLE_PRIM](#),
[ConfigPropertyType.STRING_JAVA_LANG](#)

getConfigPropertyType

[ConfigPropertyType](#) **getConfigPropertyType**()

Get the ConfigPropertyType of this object.

Returns:

the ConfigPropertyType for this object

setValue

void **setValue**([String](#) value)

Set the value of this ConfigProperty. This String value will be converted to the proper type, based on ConfigPropertyType assigned to this ConfigProperty

Parameters:

value - - will be converted to type and passed to the setter on the plugin

getValue

[String](#) **getValue**()

Get the value that is assigned to this ConfigProperty. This is the value that will be passed to the set method on the plugin.

Returns:

Returns the value.

setName

void **setName**([String](#) name)

Set the name of this object. The Plugin that this ConfigProperty is attached to should have a setter that corresponds to this name. For example, if "size" is passed in as the name, then the Plugin must have a "setSize" method.

Parameters:

name - - name of the property

getName

[String](#) **getName**()

Get the name of this object. The name must have a corresponding set method on the Plugin.

Returns:

Returns the name.

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

Release 2.5 Client API Specification

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

com.ibm.websphere.objectgrid.config

Interface BackingMapConfiguration

public interface **BackingMapConfiguration**

A BackingMapConfiguration object can be used to override BackingMap settings on the client side. The com.ibm.websphere.objectgrid.plugins.Evictor and the com.ibm.websphere.objectgrid.plugins.MapEventListener Plugins can be overridden. Other Evictor related settings can be adjusted as well as client, near-cache specific options.

Use the

com.ibm.websphere.objectgrid.config.ObjectGridConfigFactory.createBackingMapConfiguration(String) method to create a BackingMapConfiguration

Since:

WAS XD 6.0.1.2, XC10

See Also:

[Evictor](#), [MapEventListener](#), [Plugin](#), [ObjectGridConfigFactory](#)

Method Summary

v o i d	<p>addPlugin(Plugin plugin) Add a Plugin to this BackingMapConfiguration.</p>
S t r i n g	<p>getEvictionTriggers() Gets the list of eviction triggers for this BackingMapConfiguration.</p>
c o m . i b m . w e b s p h e r e . o b j e c t	<p>getKeyOutputFormat() Retrieves the data format for all data access APIs that return cache keys.</p>

g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

S
t
r
i
n
g

[getName\(\)](#)
Get the name of this BackingMapConfiguration

i
n
t

[getNumberOfBuckets\(\)](#)
Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

L
i
s
t

[getPlugins\(\)](#)
Get the Plugins that have been attached to this BackingMapConfiguration.

i
n
t

[getTimeToLive\(\)](#)
Gets the "time to live" for each map entry.

I
T
L
E
V
i
c
t
o
r
T
y
p
e

[getTtlEvictorType\(\)](#)
Gets the "time to live" Evictor type for this BackingMapConfiguration.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d

[getValueOutputFormat\(\)](#)
Retrieves the data format for all data access APIs that return cache values.

·
O
u
t
p
u
t
F
o
r
m
a
t

B
o
o
l
e
a
n

[isNearCacheEnabled\(\)](#)

If true, the client local cache is enabled for supported configurations.

B
o
o
l
e
a
n

[isNearCacheInvalidationEnabled\(\)](#)

If true, clients with local caches are automatically invalidated when the data grid map is updated.

B
o
o
l
e
a
n

[isNearCacheLastAccessTTLSyncEnabled\(\)](#)

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

V
o
i
d

[setEvictionTriggers\(String evictionTriggers\)](#)

Sets the eviction triggers for this BackingMapConfiguration.

V
o
i
d

[setKeyOutputFormat\(com.ibm.websphere.objectgrid.OutputFormat dataFormat\)](#)

Sets the data format for all data access APIs that return cache keys.

V
o
i
d

[setNearCacheEnabled\(Boolean nearCacheEnabled\)](#)

Enables or disables the client local cache for supported configurations.

V
o
i
d

[setNearCacheInvalidationEnabled\(Boolean nearCacheInvalidationEnabled\)](#)

Set to true to enable client near cache invalidation.

V
o
i
d

[setNearCacheLastAccessTTLSyncEnabled\(Boolean nearCacheLastAccessTTLSyncEnabled\)](#)

Enables or disables time-to-live access information synchronization to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

V
o
i
d

[setNumberOfBuckets\(int numBuckets\)](#)

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

V
o
i
d

[setPlugins\(List pluginList\)](#)

Set the Plugins for this BackingMapConfiguration.

v o i d	setTimeToLive (int seconds) Sets "time to live" of each BackingMap entry in seconds.
v o i d	setTtlEvictorType (TTLType ttlEvictorType) Set the "time to live" Evictor type for this BackingMapConfiguration.
v o i d	setValueOutputFormat (com.ibm.websphere.objectgrid.OutputFormat dataFormat) Sets the data format for all data access APIs that return cache values.

Method Detail

getName

[String](#) getName()

Get the name of this BackingMapConfiguration

Returns:

The name of this BackingMapConfiguration

addPlugin

void addPlugin([Plugin](#) plugin)

Add a Plugin to this BackingMapConfiguration. The Plugins that can be overridden on a client-side BackingMap are com.ibm.websphere.objectgrid.plugins.Evictor and com.ibm.websphere.objectgrid.plugins.MapEventListener.

Parameters:

plugin -

See Also:

[setPlugins\(List\)](#)

setPlugins

void setPlugins([List](#) pluginList)

Set the Plugins for this BackingMapConfiguration. Any Plugins that were previously attached to this BackingMapConfiguration object will be overridden.

Parameters:

pluginList - - a List of Plugins

See Also:

[addPlugin\(Plugin\)](#)

getPlugins

[List](#) getPlugins()

Get the Plugins that have been attached to this BackingMapConfiguration.

Returns:

getNumberOfBuckets

int `getNumberOfBuckets()`

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Gets the number of buckets defined for this BackingMapConfiguration.

Returns:
the number of buckets defined

setNumberOfBuckets

void `setNumberOfBuckets(int numBuckets)`

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Sets the number of buckets for this BackingMapConfiguration. This will be used by the BackingMap.

The BackingMap implementation uses a hash map for its implementation. If there are a lot of entries in the BackingMap then more buckets means better performance because the risk of collisions is lower as the number of buckets grows. More buckets also means more concurrency.

Parameters:
numBuckets -

See Also:
[BackingMap.setNumberOfBuckets\(int\)](#)

getTimeToLive

int `getTimeToLive()`

Gets the "time to live" for each map entry. The value is in seconds.

Returns:
the "time to live" in seconds

setTimeToLive

void `setTimeToLive(int seconds)`

Sets "time to live" of each BackingMap entry in seconds.

If this method is not called, the lifetime of an entry is forever (or until the application explicitly removes or invalidates the entry, or a user defined Evictor evicts the entry).

Parameters:
seconds -

getTtlEvictorType

[TTLType](#) `getTtlEvictorType()`

Gets the "time to live" Evictor type for this BackingMapConfiguration. If `setTtlEvictorType` was not called, this method will return null and the BackingMap based off this BackingMapConfiguration will use `TTLType.NONE`

Returns:

the "time to live" Evictor type or null if `setTtlEvictorType(TTLType)` was not called

See Also:

[setTimeToLive\(int\)](#)

`setTtlEvictorType`

`void setTtlEvictorType(TTLType ttlEvictorType)`

Set the "time to live" Evictor type for this BackingMapConfiguration. This is used to determine how expiration time of a BackingMap entry is computed.

If this method is not called, `TTLType.NONE` is used to indicate the map entry has no expiration time (e.g. is allowed to live until explicitly removed or invalidated by the application, or evicted by a user defined Evictor).

Parameters:

ttlEvictorType -

See Also:

[BackingMap.setTtlEvictorType\(TTLType\)](#)

`getEvictionTriggers`

`String getEvictionTriggers()`

Gets the list of eviction triggers for this BackingMapConfiguration.

See [BackingMap](#) for a list of valid eviction triggers.

Returns:

a semicolon separated list of eviction triggers or null if `setEvictionTriggers(String)` was not called

Since:

WAS XD 6.1.0.3

`setEvictionTriggers`

`void setEvictionTriggers(String evictionTriggers)`

Sets the eviction triggers for this BackingMapConfiguration. All evictors will use the eviction supplied triggers.

See [BackingMap](#) for a list of valid eviction triggers.

Parameters:

evictionTriggers - a semicolon separated list of eviction triggers

Since:

WAS XD 6.1.0.3

`isNearCacheInvalidationEnabled`

`Boolean isNearCacheInvalidationEnabled()`

If true, clients with local caches are automatically invalidated when the data grid map is updated.

Returns:

true if client near cache invalidation is enabled, false if client near cache invalidation is disabled, or null if the override is not specified.

Since:

8.6, XC10 2.5

setNearCacheInvalidationEnabled

void **setNearCacheInvalidationEnabled**([Boolean](#) nearCacheInvalidationEnabled)

Set to true to enable client near cache invalidation. When enabled, the client will receive events from the remote data grid to invalidate data from the local cache.

Parameters:

nearCacheInvalidationEnabled - If true, the client near cache invalidation is enabled. If false, invalidation is disabled. If null, the override is not specified and the client will use the setting from the remote data grid.

Since:

8.6, XC10 2.5

isNearCacheLastAccessTTLSyncEnabled

[Boolean](#) **isNearCacheLastAccessTTLSyncEnabled**()

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

Returns:

True if last-access time-to-live information is sent to the remote data grid, false if last-access time information is not sent, or null if the override is not specified.

Since:

8.6, XC10 2.5

setNearCacheLastAccessTTLSyncEnabled

void **setNearCacheLastAccessTTLSyncEnabled**([Boolean](#) nearCacheLastAccessTTLSyncEnabled)

Enables or disables time-to-live access information synchronization to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

Parameters:

nearCacheLastAccessTTLSyncEnabled - If true, the last-access time-to-live information is sent to the remote data grid. If false, the last-access information is not sent. If null, the override is not specified and the client will use the setting from the remote data grid.

Since:

8.6, XC10 2.5

isNearCacheEnabled

[Boolean](#) **isNearCacheEnabled**()

If true, the client local cache is enabled for supported configurations. The client near cache can only be enabled when using optimistic locking or when locking is disabled.

Returns:

True if the client near cache is enabled, false if the near cache is disabled, or null if the override is not specified.

Since:

8.6, XC10 2.5

setNearCacheEnabled

void **setNearCacheEnabled**([Boolean](#) nearCacheEnabled)

Enables or disables the client local cache for supported configurations. The client near cache can only be enabled when using optimistic locking or when locking is disabled.

Parameters:

nearCacheEnabled - If true, the client local cache is enabled for supported configurations. If false, the client local cache is disabled. If null, the override is not specified and the client will use the setting from the remote data grid.

getKeyOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getKeyOutputFormat()**

Retrieves the data format for all data access APIs that return cache keys.

Returns:

the data format or null if the default should be used.

Since:

8.6, XC10 2.5

setKeyOutputFormat

void **setKeyOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat dataFormat)

Sets the data format for all data access APIs that return cache keys.

Parameters:

dataFormat - the data format to use or null to use the default.

Since:

8.6, XC10 2.5

getValueOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getValueOutputFormat()**

Retrieves the data format for all data access APIs that return cache values.

Returns:

the data format.

Since:

8.6, XC10 2.5

setValueOutputFormat

void **setValueOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat dataFormat)

Sets the data format for all data access APIs that return cache values.

Parameters:

dataFormat - the data format to use or null to use the default.

Since:

8.6, XC10 2.5

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

PREV CLASS [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#) DETAIL: FIELD | CONSTR | [METHOD](#)

Release 2.5 Client API Specification

Package com.ibm.websphere.objectgrid.continuousquery

Interface Summary	
ContinuousQueryCache<KeyType, ValueType>	A Continuous Query Cache contains the keys and optionally the values that match a defined continuous query.
ContinuousQueryFilter<KeyType, ValueType, AttributeType, MatchType>	An interface which provides an abstraction to check if an object matches the criteria defined in a filter.
ContinuousQueryListener<KeyType, ValueType>	This interface should be implemented to get a callback for a continuous query.
ContinuousQueryManager	The management interface used to define new continuous queries, retrieve an existing query, or destroy a query.
ContinuousQueryNotificationEvent<KeyType, ValueType>	An interface which defines notification metadata that is sent to continuous query listeners when the result set of a defined continuous query changes.
ContinuousQueryTopic<KeyType, ValueType>	A Continuous Query Topic embodies the client side view of a defined Continuous Query.

Class Summary	
ContinuousQueryManagerFactory	A factory class for obtaining the ContinuousQueryManager.

Enum Summary	
ContinuousQueryNotificationEvent.Reason	Indicates the reason a notification event has occurred for this cache entry.

com.ibm.websphere.objectgrid.continuousquery

Interface ContinuousQueryTopic<KeyType, ValueType>

Type Parameters:

KeyType - Type of the key object for the map being queried

ValueType - Type of the value object for the map being queried

```
public interface ContinuousQueryTopic<KeyType,ValueType>
```

A Continuous Query Topic embodies the client side view of a defined Continuous Query. The topic gives access to the client-side cached results of the continuous query and allows for management of Continuous Query Listeners for this topic.

Since:

8.6, XC10 2.5

Method Summary

void [addListener](#)([ContinuousQueryListener](#)<KeyType,ValueType> listener)
Add a Continuous Query Listener to this topic.

Collection [getAllListeners](#)()
Returns a shallow copy of the current set of listeners registered for this query topic.

Value
Type
,
Value
Type
,
Value
Type
,
>

Continuous
Query
Cache
Object
Methods
<
Key
Value
Type
,
Value
Type
,
>

[getCache\(\)](#)
Return a continuous query cache object with methods to access the keys and values stored in the cache.

Static
Method
<

[getName\(\)](#)
Returns the generated unique name for this continuous query.

com
ibm
websphere

e . o b j e c t g r i d . O u t p u t F o r m a t	<p>getOutputFormat() Return the format this query will use for keys and values.</p>
L i s < I n t e r >	<p>getPartitions() Return the list of partitions this query is defined against.</p>
b o o l e a n	<p>isKeysOnlyCache() Return true if the cache for this continuous query contains only keys.</p>
b o o l e a n	<p>noCache() Return true if the no caching option is enabled for this Continuous Query.</p>
v o i d	<p>removeAllListeners() Removes all registered listeners from this query topic.</p>
b o o l e a n	<p>removeListener(ContinuousQueryListener<KeyType,ValueType> listener) Remove the listener from this continuous query's the set of listeners.</p>

Method Detail

addListener

void **addListener**([ContinuousQueryListener](#)<KeyType,ValueType> listener)

Add a Continuous Query Listener to this topic. The listener will be called when this topic receives updates to the query. There is no guarantee of the order that listeners will be called. Multiple adds of the same listener will result in only one call to the listener.

Parameters:

listener - the listener to add to this continuous query's set of listeners.

removeListener

boolean **removeListener**([ContinuousQueryListener](#)<KeyType,ValueType> listener)

Remove the listener from this continuous query's the set of listeners. In flight notifications to this listener may continue after this method returns.

Parameters:

listener - - the continuous query listener to remove.

Returns:

true if the listener was removed.

getCache

[ContinuousQueryCache](#)<KeyType,ValueType> **getCache**()

Return a continuous query cache object with methods to access the keys and values stored in the cache.

Returns:

a Continuous Query Cache object for this topic.

getName

[String](#) **getName**()

Returns the generated unique name for this continuous query.

Returns:

a unique string.

isKeysOnlyCache

boolean **isKeysOnlyCache**()

Return true if the cache for this continuous query contains only keys. Return false if it also contains the value associated with the key.

Returns:

true if the cache does not contain values.

noCache

boolean **noCache**()

Return true if the no caching option is enabled for this Continuous Query.

Returns:

true if no keys or values are being cached.

getOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getOutputFormat()**

Return the format this query will use for keys and values.

Returns:

the OutputFormat specified when this query was defined.

See Also:

OutputFormat

getAllListeners

[Collection](#)<[ContinuousQueryListener](#)<[KeyType](#),[ValueType](#)>> **getAllListeners()**

Returns a shallow copy of the current set of listeners registered for this query topic.

Returns:

a new collection object containing all currently registered listeners.

removeAllListeners

void **removeAllListeners()**

Removes all registered listeners from this query topic. In flight notifications to listeners may continue after this method returns.

getPartitions

[List](#)<[Integer](#)> **getPartitions()**

Return the list of partitions this query is defined against. If the return value is null or an empty list, this query is defined against all partitions of the map.

Returns:

the array of partitions

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.continuousquery

Interface

ContinuousQueryNotificationEvent<KeyType,ValueType>

Type Parameters:

KeyType - Type of the key object in this event, from the key type used in the map being queried
 ValueType - Type of the value object in this event, from the value type used in the map being queried

```
public interface ContinuousQueryNotificationEvent<KeyType,ValueType>
```

An interface which defines notification metadata that is sent to continuous query listeners when the result set of a defined continuous query changes.

Since:

8.6, XC10 2.5

Nested Class Summary

s t a t i c c l a s s	ContinuousQueryNotificationEvent.Reason Indicates the reason a notification event has occurred for this cache entry.
---	---

Method Summary

K e y T y p e	getKey() Retrieve the key of the cache entry that triggered this notification.
C o n t i n u o u s Q u e r y	

e
r
r
o
r
N
o
t
i
f
i
c
a
t
i
o
n
E
v
e
n
t
R
e
a
s
o
n

[getReason\(\)](#)

Retrieve the reason for this notification event

c
o
m
·
i
b
m
·
w
e
b
s
p
h
e
r
e
·
o
b
j
e
c
t
g
r
i
d
·
S
e
s
s
i
o
n
H
a
n
d
l
e

[getSessionHandle\(\)](#)

Retrieve a SessionHandle object which can be used to retrieve the cache entry that triggered this notification.

V
a
l
u
e

[getValue\(\)](#)

Retrieve the value of the cache entry that triggered this notification, if the query was

[y](#) configured to return the value object for matches.

[b](#)
[o](#)
[o](#) [isKeysOnly\(\)](#)
[l](#) If true, this notification only includes the key of the matching cache entry.
[e](#)
[a](#)
[n](#)

Method Detail

getKey

[KeyType](#) `getKey()`

Retrieve the key of the cache entry that triggered this notification. This object is a reference to the object stored in the [ContinuousQueryCache](#), if the cache has been enabled. Therefore it should not be modified.

Returns:
the cache entry key

getValue

[ValueType](#) `getValue()`

Retrieve the value of the cache entry that triggered this notification, if the query was configured to return the value object for matches. This object is a reference to the object stored in the [ContinuousQueryCache](#), if the cache has been enabled. Therefore it should not be modified.

Returns:
the cache entry value

getSessionHandle

`com.ibm.websphere.objectgrid.SessionHandle` `getSessionHandle()`

Retrieve a `SessionHandle` object which can be used to retrieve the cache entry that triggered this notification. This method can only be invoked when using a `PER_CONTAINER` placement strategy for the map.

Returns:
the `SessionHandle` for the partition that owns the cache entry.

See Also:
`MapSet.getPlacementStrategy()`, [Session.setSessionHandle\(SessionHandle\)](#)

getReason

[ContinuousQueryNotificationEvent.Reason](#) `getReason()`

Retrieve the reason for this notification event

Returns:

the Reason enum

isKeysOnly

boolean **isKeysOnly**()

If true, this notification only includes the key of the matching cache entry. If false, this notification also includes the value

Returns:

whether or not this notification includes the cache value.

See Also:

[getValue\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help
--------------------------	-------------------------	-------------------------	----------------------------	----------------------------	-----------------------	----------------------

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Enum ContinuousQueryNotificationEvent.Reason

[java.lang.Object](#)

[?](#) [java.lang.Enum](#)<[ContinuousQueryNotificationEvent.Reason](#)>

[?](#) [com.ibm.websphere.objectgrid.continuousquery.ContinuousQueryNotificationEvent.Reason](#)

All Implemented Interfaces:

[Serializable](#), [Comparable](#)<[ContinuousQueryNotificationEvent.Reason](#)>

Enclosing interface:

[ContinuousQueryNotificationEvent](#)<[KeyType](#), [ValueType](#)>

```
public static enum ContinuousQueryNotificationEvent.Reason
extends Enum<ContinuousQueryNotificationEvent.Reason>
```

Indicates the reason a notification event has occurred for this cache entry.

Enum Constant Summary

ADDED

The cache entry was inserted into the grid and matches the query, or an entry which did not match the query was updated such that it now matches the query.

CLEAR

A cache entry which matched the query was removed from the grid due to a map clear operation.

REMOVED

The cache entry, which previously matched the query, was removed from the grid or updated such that it no longer matches.

UPDATED

The cache entry, which previously matched the query, was updated and still matches the query.

Method Summary

s
t
a
t
i
c
C
o
n
t
i
n
u
o
u
s

Q
u
e
r
y
N
o
t
i
f
i
c
a
t
i
o
n
D
e
f
i
n
e
D
e
s
c
r
i
b
e

[valueOf](#)([String](#) name)

Returns the enum constant of this type with the specified name.

s
t
a
t
i
c
C
o
n
s
t
a
n
t
D
e
f
i
n
e
D
e
s
c
r
i
b
e
D
e
f
i
n
e
D
e
s
c
r
i
b
e

[values](#)()

Returns an array containing the constants of this enum type, in the order they are declared.

[
]

Methods inherited from class java.lang.Enum

[clone](#), [compareTo](#), [equals](#), [finalize](#), [getDeclaringClass](#), [hashCode](#), [name](#), [ordinal](#), [toString](#), [valueOf](#)

Methods inherited from class java.lang.Object

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Enum Constant Detail

ADDED

public static final [ContinuousQueryNotificationEvent.Reason](#) **ADDED**

The cache entry was inserted into the grid and matches the query, or an entry which did not match the query was updated such that it now matches the query.

REMOVED

public static final [ContinuousQueryNotificationEvent.Reason](#) **REMOVED**

The cache entry, which previously matched the query, was removed from the grid or updated such that it no longer matches.

UPDATED

public static final [ContinuousQueryNotificationEvent.Reason](#) **UPDATED**

The cache entry, which previously matched the query, was updated and still matches the query. Note: The result of `getValue()` for REMOVE events will always be null, even if the query was defined to include values.

See Also:

[ContinuousQueryManager.defineContinuousQuery\(java.lang.String, com.ibm.websphere.objectgrid.continuousquery.ContinuousQueryFilter, boolean, boolean, boolean, java.util.Collection>, boolean, com.ibm.websphere.objectgrid.OutputFormat, java.util.List\)](#)

CLEAR

public static final [ContinuousQueryNotificationEvent.Reason](#) **CLEAR**

A cache entry which matched the query was removed from the grid due to a map clear operation.

Method Detail

values

public static [ContinuousQueryNotificationEvent.Reason](#)[] **values()**

Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows:

```
for (ContinuousQueryNotificationEvent.Reason c : ContinuousQueryNotificationEvent.Reason.values())
    System.out.println(c);
```


Returns:

an array containing the constants of this enum type, in the order they are declared

valueOf

```
public static ContinuousQueryNotificationEvent.Reason valueOf(String name)
```

Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

Parameters:

name - the name of the enum constant to be returned.

Returns:

the enum constant with the specified name

Throws:

[IllegalArgumentException](#) - if this enum type has no constant with the specified name

[NullPointerException](#) - if the argument is null

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS NEXT CLASS			FRAMES NO FRAMES All					
SUMMARY: NESTED ENUM			DETAIL: ENUM					
CONSTANTS FIELD METHOD			CONSTANTS FIELD METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Class ContinuousQueryManagerFactory

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.continuousquery.ContinuousQueryManagerFactory

```
public final class ContinuousQueryManagerFactory  
extends Object
```

A factory class for obtaining the ContinuousQueryManager.

Since:

8.6, XC10 2.5

Constructor Summary

[ContinuousQueryManagerFactory\(\)](#)

Method Summary

s
t
a
t
i
c

c
o
n
s
t
r
u
c
t
o
r

m
e
t
h
o
d

s
u
m
m
a
r
y

[getManager\(ObjectGrid objGrid\)](#)

Factory that returns the single instance of a ContinuousQueryManager for the given object grid.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ContinuousQueryManagerFactory

```
public ContinuousQueryManagerFactory()
```

Method Detail

getManager

```
public static ContinuousQueryManager getManager(ObjectGrid objGrid)
```

Factory that returns the single instance of a ContinuousQueryManager for the given object grid.

Parameters:

objGrid - - the ObjectGrid this Continuous Query Manager will define continuous queries for.

Returns:

the ContinuousQueryManager instance.

Overview	Package	Classes	Serialized	Deprecated	Index	Help
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All Classes		

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.continuousquery

Interface ContinuousQueryManager

public interface **ContinuousQueryManager**

The management interface used to define new continuous queries, retrieve an existing query, or destroy a query. An instance can be retrieved via

[ContinuousQueryManagerFactory.getManager\(com.ibm.websphere.objectgrid.ObjectGrid\)](#)

Since:

8.6, XC10 2.5

Method Summary

<KeyType,ValueType> ContinuousQueryTopic <KeyType,ValueType>	defineContinuousQuery (String mapName, ContinuousQueryFilter filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated) Define a new continuous query.
<KeyType,ValueType> ContinuousQueryTopic <KeyType,ValueType>	defineContinuousQuery (String mapName, ContinuousQueryFilter filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated, Collection <ContinuousQueryListener <KeyType,ValueType> continuousQueryListeners, boolean noCache, com.ibm.websphere.objectgrid.OutputFormat format, List <Integer > partitionSubset) Define a new continuous query.
<KeyType,ValueType> ContinuousQueryTopic <KeyType,ValueType>	defineContinuousQuery (String mapName, ContinuousQueryFilter filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated, List <Integer > partitionSubset) Define a new continuous query.
List <ContinuousQueryTopic <?,?>	getDefinedContinuousQueries () Returns a list view of the currently defined Continuous Query Topics.
boolean	removeContinuousQuery (ContinuousQueryTopic <?,?> topic) Remove a defined continuous query.

Method Detail

defineContinuousQuery

[<KeyType,ValueType> ContinuousQueryTopic](#)[<KeyType,ValueType>](#) **defineContinuousQuery**([String](#) mapName, [ContinuousQueryFilter](#) filter, boolean keysOnly, boolean returnInitialResultSet, boolean notifyOnUpdated)

Updated,

```
Collection<ContinuousQueryListener<KeyType,ValueType>> continuousQueryListeners,  
boolean noCache,  
com.ibm.websphere.objectgrid.OutputFormat format,  
List<Integer> partitionSubset)  
throws ContinuousQueryIncompatibleDuplicateException,  
UndefinedMapException
```

Define a new continuous query. This query will run in the grid container processes. When cache entries that match the filter criteria are inserted or deleted, the client's [ContinuousQueryCache](#) will be updated and any [ContinuousQueryListener](#) implementations will be invoked. This will also occur if a cache entry is updated such that it now matches, or no longer matches, the filter criteria. Optionally, notifications can be sent for update operations to cache entries that previously matched the criteria, and still match the criteria after the update.

Parameters:

`mapName` - - the name of the map the continuous query will be defined on.
`filter` - - the filter that will determine the contents of the continuous query. Custom filter logic can be implemented by extending [AbstractCQFilter](#).
`keysOnly` - - true if the query should only return the keys of items that satisfy the query; false to also send the values as well as the keys. If false, the [ContinuousQueryCache](#) and [ContinuousQueryNotificationEvent](#) objects will include the value associated with cache entries that match the query.
`returnInitialResultSet` - - true to run the filter on all existing matches in the map at the time of the query definition and return the items to the cache. If false, the client will only be notified of matches which occur after the query is defined. Returning the initial result set will have a high computational requirement for large maps.
`notifyOnUpdated` - - true causes this topic to get called every time an existing item that satisfies this query gets updated and continues to satisfy this query. This will drive [ContinuousQueryListener](#) call backs as well as updated values in the query cache if `keysOnly` is false.
`continuousQueryListeners` - - used to register listeners before the query is defined. These listeners will be guaranteed to be invoked for any entries in the initial result set, as well as changes to the result set which might occur before [ContinuousQueryTopic.addListener\(ContinuousQueryListener\)](#) can be invoked. Can be null.
`noCache` - - true to indicate that no caching of keys or values should occur on the client. Continuous Query listeners will still be notified.
`format` - - if `OutputFormat.RAW` is specified, keys and values returned from the [ContinuousQueryCache](#) and [ContinuousQueryListener](#) will be in `SerializedEntry` format when a `DataSerializer` is defined on the map. - if `OutputFormat.NATIVE` is specified, or when no `DataSerializer` is defined, keys and values will be returned as `Objects`.
`partitionSubset` - - Used to indicate that this continuous query should only be defined on the specified subset of partitions. Passing null or an empty list indicates this continuous query should be defined on all existing and future partitions containing this map. An `IllegalArgumentException` may be thrown if an invalid partition ID is passed.

Returns:

The [ContinuousQueryTopic](#) for this continuous query. If the call to this method would result in a topic which is identical to an existing topic being created, the existing instance is returned.

Throws:

[ContinuousQueryIncompatibleDuplicateException](#)
[UndefinedMapException](#)
[IllegalArgumentException](#)

defineContinuousQuery

```

<KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType> defineContinuousQuery(String mapName,
ContinuousQueryF
ilter filter,
,
initialResultSet,
Updated,
List<Integer> pa
rtitionSubset)
throws ContinuousQueryIncompatibleDu
plicateException,
UndefinedMapException

```

Define a new continuous query. This query will run in the grid container processes. When cache entries that match the filter criteria are inserted or deleted, the client's [ContinuousQueryCache](#) will be updated and any [ContinuousQueryListener](#) implementations will be invoked. This will also occur if a cache entry is updated such that it now matches, or no longer matches, the filter criteria. Optionally, notifications can be sent for update operations to cache entries that previously matched the criteria, and still match the criteria after the update.

Parameters:

mapName - - the name of the map the continuous query will be defined on.
filter - - the filter that will determine the contents of the continuous query. Custom filter logic can be implemented by extending [AbstractCQFilter](#).
keysOnly - - true if the query should only return the keys of items that satisfy the query; false to also send the values as well as the keys. If false, the [ContinuousQueryCache](#) and [ContinuousQueryNotificationEvent](#) objects will include the value associated with cache entries that match the query.
returnInitialResultSet - - true to run the filter on all existing matches in the map at the time of the query definition and return the items to the cache. If false, the client will only be notified of matches which occur after the query is defined. Returning the initial result set will have a high computational requirement for large maps.
notifyOnUpdated - - true causes this topic to get called every time an existing item that satisfies this query gets updated and continues to satisfy this query. This will drive [ContinuousQueryListener](#) call backs as well as updated values in the query cache if keysOnly is false.
partitionSubset - - Used to indicate that this continuous query should only be defined on the specified subset of partitions. Passing null or an empty list indicates this continuous query should be defined on all existing and future partitions containing this map. An [IllegalArgumentException](#) may be thrown if an invalid partition ID is passed.

Returns:

The [ContinuousQueryTopic](#) for this continuous query. If the call to this method would result in a topic which is identical to an existing topic being created, the existing instance is returned.

Throws:

[ContinuousQueryIncompatibleDuplicateException](#)
[UndefinedMapException](#)
[IllegalArgumentException](#)

defineContinuousQuery

```

<KeyType,ValueType> ContinuousQueryTopic<KeyType,ValueType> defineContinuousQuery(String mapName,
ContinuousQueryF
ilter filter,
,
initialResultSet,
Updated,
List<Integer> pa
rtitionSubset)
throws ContinuousQueryIncompatibleDu
plicateException,
UndefinedMapException

```


Updated)

throws [ContinuousQueryIncompatibleDuplicateException](#),

[uplicateException](#),

[UndefinedMapException](#)

Define a new continuous query. This query will run in the grid container processes. When cache entries that match the filter criteria are inserted or deleted, the client's [ContinuousQueryCache](#) will be updated and any [ContinuousQueryListener](#) implementations will be invoked. This will also occur if a cache entry is updated such that it now matches, or no longer matches, the filter criteria. Optionally, notifications can be sent for update operations to cache entries that previously matched the criteria, and still match the criteria after the update. This query will be applied to all partitions.

Parameters:

`mapName` - - the name of the map the continuous query will be defined on.
`filter` - - the filter that will determine the contents of the continuous query. Custom filter logic can be implemented by extending [AbstractCQFilter](#).
`keysOnly` - - true if the query should only return the keys of items that satisfy the query; false to also send the values as well as the keys. If false, the [ContinuousQueryCache](#) and [ContinuousQueryNotificationEvent](#) objects will include the value associated with cache entries that match the query.
`returnInitialResultSet` - - true to run the filter on all existing matches in the map at the time of the query definition and return the items to the cache. If false, the client will only be notified of matches which occur after the query is defined. Returning the initial result set will have a high computational requirement for large maps.
`notifyOnUpdated` - - true causes this topic to get called every time an existing item that satisfies this query gets updated and continues to satisfy this query. This will drive [ContinuousQueryListener](#) call backs as well as updated values in the query cache if `keysOnly` is false.

Returns:

The [ContinuousQueryTopic](#) for this continuous query. If the call to this method would result in a topic which is identical to an existing topic being created, the existing instance is returned.

Throws:

[ContinuousQueryIncompatibleDuplicateException](#)
[UndefinedMapException](#)
[IllegalArgumentException](#)

removeContinuousQuery

boolean `removeContinuousQuery`([ContinuousQueryTopic](#)<?,?> topic)

Remove a defined continuous query. Does the opposite of [defineContinuousQuery\(String, ContinuousQueryFilter, boolean, boolean, boolean\)](#). If the number of calls to this method for a given topic equals the number of calls to `defineContinuousQuery` that return the specified topic, the topic will cease receiving updates from the server, all listeners of this topic will be removed and the reference to this topic will be invalid.

Parameters:

`topic` - - the topic to remove

Returns:

true if the topic was removed, false otherwise

Throws:

[IllegalArgumentException](#)

getDefinedContinuousQueries

[List](#)<[ContinuousQueryTopic](#)<?,?>> `getDefinedContinuousQueries`()

Returns a list view of the currently defined Continuous Query Topics.

Returns:
a list of Continuous Query Topics

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Interface

ContinuousQueryListener<KeyType,ValueType>

Type Parameters:

KeyType - Type of the key object for the map being queried

ValueType - Type of the value object for the map being queried

```
public interface ContinuousQueryListener<KeyType,ValueType>
```

This interface should be implemented to get a callback for a continuous query. When a Continuous Query listener is added to a Continuous Query Topic, that listener will be called after every change to the current set of results matching the continuous query.

Since:

8.6, XC10 2.5

Method Summary

v o i d	cacheUpdated (ContinuousQueryNotificationEvent < KeyType , ValueType > event) This method will be called by every ContinuousQueryTopic that this listener is added to.
------------------	--

Method Detail

cacheUpdated

```
void cacheUpdated(ContinuousQueryNotificationEvent<KeyType,ValueType> event)
```

This method will be called by every ContinuousQueryTopic that this listener is added to. The caller will capture any [Throwable](#)s thrown from calls to this method and create an FFDC entry. This [ContinuousQueryListener](#) will not be called again for the same [ContinuousQueryNotificationEvent](#). Additionally, the performance of this [ContinuousQueryListener](#) may make an impact on the speed with which other [ContinuousQueryListeners](#) are called.

Parameters:

event - - [ContinuousQueryNotificationEvent](#) containing information about the change in the query result set

See Also:

[ContinuousQueryTopic.addListener\(ContinuousQueryListener\)](#),
[ContinuousQueryNotificationEvent](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.continuousquery

Interface

ContinuousQueryFilter<KeyType,ValueType,AttributeType,MatchType>

Type Parameters:

- KeyType - Type of the key object for the map being queried
- ValueType - Type of the value object for the map being queried
- AttributeType - Type of the attribute referenced by the attribute path
- MatchType - Type of the object being compared to

All Superinterfaces:

[Serializable](#)

All Known Implementing Classes:

[AbstractCQFilter](#), [AndFilter](#), [BinaryLogicalFilter](#), [CompareFilter](#), [EQFilter](#), [FalseFilter](#),
[GTEFilter](#), [GTFilter](#), [IsNotNullFilter](#), [IsNullFilter](#), [LTEFilter](#), [LTFilter](#), [MatchFilter](#),
[NEQFilter](#), [NotFilter](#), [NotMatchFilter](#), [OrFilter](#), [TrueFilter](#)

```
public interface ContinuousQueryFilter<KeyType,ValueType,AttributeType,MatchType>
extends Serializable
```

An interface which provides an abstraction to check if an object matches the criteria defined in a filter. Filters are invoked within the eXtreme Scale containers hosting partitions for the map on which the query is defined. All implementations must extend [AbstractCQFilter](#).

Since:

8.6, XC10 2.5

Field Summary	
s t a t i c S t r i n g	<p>POJO_ADDRESSABLEKEYNAME</p> <p>The name used to identify the key objects in POJO maps.</p>
s t a t i c S t r i n g	<p>POJO_PATHSEPARATOR</p> <p>The path separator to use when identifying attribute paths for POJO maps.</p>

Method Summary

`boolean filter(FilterContent<KeyType,ValueType> content)`
Checks if the supplied object passes the filter.

Field Detail

POJO_ADDRESSABLEKEYNAME

static final [String](#) POJO_ADDRESSABLEKEYNAME

The name used to identify the key objects in POJO maps. Not applicable to maps using XDF or a custom MapSerializedPlugin.

See Also:

[CompareFilter.CompareFilter\(String, Object\)](#), [Constant Field Values](#)

POJO_PATHSEPARATOR

static final [String](#) POJO_PATHSEPARATOR

The path separator to use when identifying attribute paths for POJO maps. Not applicable to maps using XDF or a custom MapSerializedPlugin.

See Also:

[CompareFilter.CompareFilter\(String, Object\)](#), [Constant Field Values](#)

Method Detail

filter

boolean **filter**([FilterContent](#)<KeyType,ValueType> content)
throws [ContinuousQueryException](#)

Checks if the supplied object passes the filter.

Parameters:

content - A representation of the cache entry to be checked

Returns:

true if the object matches the filtering criteria, false otherwise

Throws:

[ContinuousQueryException](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.continuousquery

Interface ContinuousQueryCache<KeyType, ValueType>

Type Parameters:

KeyType - Type of the key object for the map being queried

ValueType - Type of the value object for the map being queried

```
public interface ContinuousQueryCache<KeyType, ValueType>
```

A Continuous Query Cache contains the keys and optionally the values that match a defined continuous query. The contents of this cache arrive asynchronously from the grid for which the query is defined. If the query is defined such that only the keys are stored in the continuous query cache then all operations that return a value will return null.

Since:

8.6, XC10 2.5

Method Summary

b
o
o
l
e
a
n

[containsKey](#)([KeyType](#) key)

Returns true if the cache contains the given key.

b
o
o
l
e
a
n

[containsValue](#)([ValueType](#) value)

Returns true if the cache contains the given value.

V
a
l
u
e
T
y
p
e

[get](#)([KeyType](#) key)

Returns the value for the given key.

L
i
s
t
<
c
o
m
.
i
b
m
.

w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
S
e
s
s
i
o
n
H
a
n
d
l
e
>

[getSessionHandles](#)([KeyType](#) key)

Returns SessionHandle objects that can be used to retrieve all query matches associated with this key.

b
o
o
l
e
a
n

[includesValues](#)()

Returns true if this cache includes the values associated with the keys that have matched the query.

S
e
<
K
e
y
T
y
p
e
>

[keySnapshot](#)()

Returns a Set view of a snapshot of the keys contained in the cache.

i
n
t

[size](#)()

Return the number of keys in the cache.

Method Detail

keySnapshot

[Set](#)<[KeyType](#)> [keySnapshot](#)()

Returns a Set view of a snapshot of the keys contained in the cache. The objects in the set are references to the keys in the query cache, therefore they should not be modified.

Returns:

a set view of the keys in the cache.

size

int **size**()

Return the number of keys in the cache.

Returns:

the number of keys in the cache.

get

[ValueType](#) **get**([KeyType](#) key)

Returns the value for the given key. The object returned is a reference to the value in the query cache, therefore it should not be modified.

Parameters:

key - - the key whose associated value is to be returned

Returns:

the value in the cache for the specified key or null if either the cache does not contain this key or if the cache is configured to not contain values.

getSessionHandles

[List](#)<[com.ibm.websphere.objectgrid.SessionHandle](#)> **getSessionHandles**([KeyType](#) key)

Returns `SessionHandle` objects that can be used to retrieve all query matches associated with this key. This method can only be invoked when using a `PER_CONTAINER` placement strategy for the map.

Parameters:

key - - the key for which `SessionHandles` will be returned

Returns:

a List of `SessionHandles`, one for each instance of this key in the grid.

See Also:

`MapSet.getPlacementStrategy()`

containsKey

boolean **containsKey**([KeyType](#) key)

Returns true if the cache contains the given key.

Parameters:

key - - key to check for

Returns:

true if the specified key is in the cache at the time of the request.

containsValue

boolean **containsValue**([ValueType](#) value)

Returns true if the cache contains the given value.

Parameters:

value - - value to check for

Returns:

true if the specified object is in the cache at the time of the request.

includesValues

boolean **includesValues()**

Returns true if this cache includes the values associated with the keys that have matched the query. If true, [containsValue\(Object\)](#) and [get\(Object\)](#) will return usable values, otherwise they will return null.

Returns:

true if this query cache contains values in addition to keys.

See Also:

[ContinuousQueryManager.defineContinuousQuery\(java.lang.String, com.ibm.websphere.objectgrid.continuousquery.ContinuousQueryFilter, boolean, boolean, boolean, java.util.Collection>, boolean, com.ibm.websphere.objectgrid.OutputFormat, java.util.List\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help
--------------------------	-------------------------	-------------------------	----------------------------	----------------------------	-----------------------	----------------------

PREV CLASS [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#) | [OD](#) | [DETAIL](#): FIELD | CONSTR | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

Package **com.ibm.websphere.objectgrid.management**

This package contains the interfaces for all ObjectGrid MBeans.

See:

[Description](#)

Interface Summary	
AgentManagerMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific Agent on a server process.
CatalogServiceManagementMBean	This MBean interface allows user to manipulate the behaviors of heartbeat and leader manager and other catalog service specific actions.
ContainerMBean	This MBean interface allows a client process to perform operations on and get status from an ObjectGrid container running in a dynamic environment.
DynamicServerMBean	This MBean interface allows a client process to access different attributes about a specific server process in a dynamic environment.
HashIndexMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific HashIndex on a server process.
MapMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific map on a server process.
ObjectGridMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific ObjectGrid on a server process.
PlacementMediationServiceMBean	This MBean interface allows a client process to perform operations on and get status from the PlacementMediationService running in a dynamic environment.
PlacementServiceMBean	This MBean interface allows a client process to perform operations on and get status from the PlacementService running in a dynamic environment.
QueryManagerMBean	This MBean interface allows a client process to perform operations on and get status from an ObjectGrid Query Manager running in a dynamic environment.
QuorumManagerMBean	Each catalog service has a QuorumManager and an associated MBean.
ServerMBean	This MBean interface allows a client process to access different attributes about a specific server process.
SessionMBean	This MBean interface allows a client process to access different attributes and statistical data about a specific session.
ShardMBean	This MBean interface allows a client process to perform operations on and get status from a shard running in a dynamic environment.
ThreadPoolMBean	This MBean interface allows user to access the thread pool properties.

Package com.ibm.websphere.objectgrid.management Description

This package contains the interfaces for all ObjectGrid MBeans.

Overview

Each MBean interface has several methods to administer and monitor ObjectGrid services and components.

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.management

Interface ThreadPoolMBean

public interface **ThreadPoolMBean**

This MBean interface allows user to access the thread pool properties. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ThreadPool
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

7.0.0.0 FIX2, XC10

Method Summary

i n t	getActiveThreadCount() Retrieves the approximate number of active threads in the pool.
i n t	getMaximumSize() Retrieves the maximum number of threads in the pool.
i n t	getMinimumSize() Retrieves the minimum number of threads in the pool.
S t r i n g	getName() Retrieves the name of the ThreadPool.
V o i d	setMaximumSize(int size) Sets the maximum thread pool size.
V o i d	setMinimumSize(int size) Sets the minimum thread pool size.

Method Detail

setMaximumSize

void **setMaximumSize**(int size)

Sets the maximum thread pool size.

Parameters:

size - the maximum number of threads.

getMaximumSize

int **getMaximumSize**()

Retrieves the maximum number of threads in the pool.

Returns:

the maximum number of threads in the pool

setMinimumSize

void **setMinimumSize**(int size)

Sets the minimum thread pool size.

Parameters:

size - the minimum number of threads.

getMinimumSize

int **getMinimumSize**()

Retrieves the minimum number of threads in the pool.

Returns:

the minimum number of threads in the pool

getActiveThreadCount

int **getActiveThreadCount**()

Retrieves the approximate number of active threads in the pool.

Returns:

the number of active threads in the pool in use

getName

[String](#) **getName**()

Retrieves the name of the ThreadPool.

Returns:

the name of the ThreadPool

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management
Interface ShardMBean

public interface **ShardMBean**

This MBean interface allows a client process to perform operations on and get status from a shard running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=Shard,name=<objectgrid>,objectgrid=<objectgrid>,mapset=<mapset>
,partition=<partition id>,container=<container>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary	
s t a t i c S t r i n g	<p>ROLE_SWAP_REQUESTED_WITH_SAME_TYPE Indicates that this shard is the same type of shard as the requested swap type.</p>
s t a t i c S t r i n g	<p>ROLE_SWAP_SUCCESSFUL Indicates that the role swap was executed successfully.</p>
s t a t i c S t r i n g	<p>ROLE_SWAP_TIMEOUT Indicates that this shard has timed out waiting to inherit its requested role</p>
s t	

a t t r i b u t e	<p><u>TYPE_INACTIVE</u> Indicates the shard type is the inactive role.</p>
s t a t i c	<p><u>TYPE_PRIMARY</u> Indicates the shard type is the primary role.</p>
s t a t i c	<p><u>TYPE_REPLICA_ASYNCHRONOUS</u> Indicates the shard type is the asynchronous replica role.</p>
s t a t i c	<p><u>TYPE_REPLICA_SYNCHRONOUS</u> Indicates the shard type is the synchronous replica role.</p>

Method Summary

l o n g	<p><u>getActiveRequestCount()</u> Retrieves the number of requests currently being processed by this shard.</p>
S t r i n g	<p><u>getContainerName()</u> Retrieves the name of the container that is hosting this shard.</p>
S t r i n g	<p><u>getDomainName()</u> Retrieve the name of the catalog server grouping administering this shard.</p>
l o n g	<p><u>getForwardedRequestCount()</u> Retrieves the number of requests that this shard has forwarded since its inception.</p>

S t r i n g	<p>getMapSetName() Retrieve the name of the MapSet in which the shard resides.</p>
S t r i n g	<p>getObjectGridName() Retrieve the name of the ObjectGrid in which the shard resides.</p>
S t r i n g	<p>getPartitionName() Retrieve the name of the partition in which the shard resides.</p>
l o n g	<p>getProcessedRequestCount() Retrieves the number of requests that this shard has processed since its inception.</p>
S t r i n g	<p>getState() Retrieve the state of the shard.</p>
l o n g	<p>getTotalRequestCount() Retrieves the number of requests that this shard has processed or forwarded since its inception.</p>
S t r i n g	<p>getType() Retrieve the type of the shard.</p>
S t r i n g	<p>swapWithPrimary() Causes this shard to swap roles with the primary shard for the partition.</p>

Field Detail

ROLE_SWAP_SUCCESSFUL

static final [String](#) ROLE_SWAP_SUCCESSFUL

Indicates that the role swap was executed successfully.

Since:

7.1.0.0 FIX1

See Also:

[swapWithPrimary\(\)](#), [Constant Field Values](#)

ROLE_SWAP_REQUESTED_WITH_SAME_TYPE

static final [String](#) ROLE_SWAP_REQUESTED_WITH_SAME_TYPE

Indicates that this shard is the same type of shard as the requested swap type. No swap will be executed.

Since:

7.1.0.0 FIX1

See Also:

[swapWithPrimary\(\)](#), [Constant Field Values](#)

ROLE_SWAP_TIMEOUT

static final [String](#) ROLE_SWAP_TIMEOUT

Indicates that this shard has timed out waiting to inherit its requested role

Since:

7.1.0.0 FIX1

See Also:

[swapWithPrimary\(\)](#), [Constant Field Values](#)

TYPE_PRIMARY

static final [String](#) TYPE_PRIMARY

Indicates the shard type is the primary role. This means that this is the shard that handles all updates and coordinates state transitions with the replicas.

See Also:

[Constant Field Values](#)

TYPE_REPLICA_SYNCHRONOUS

static final [String](#) TYPE_REPLICA_SYNCHRONOUS

Indicates the shard type is the synchronous replica role. This means that this shard is receiving state updates from another shard that is acting as primary.

See Also:

[Constant Field Values](#)

TYPE_REPLICA_ASYNCHRONOUS

static final [String](#) TYPE_REPLICA_ASYNCHRONOUS

Indicates the shard type is the asynchronous replica role. This means that this shard is receiving state updates from another shard that is acting as primary.

See Also:

[Constant Field Values](#)

TYPE_INACTIVE

static final [String](#) TYPE_INACTIVE

Indicates the shard type is the inactive role. This means that this shard is not actively enrolled in the partition.

See Also:

[Constant Field Values](#)

Method Detail

getObjectGridName

[String](#) getObjectGridName()

Retrieve the name of the ObjectGrid in which the shard resides.

Returns:

The ObjectGrid name.

getMapSetName

[String](#) getMapSetName()

Retrieve the name of the MapSet in which the shard resides.

Returns:

The MapSet name.

getPartitionName

[String](#) getPartitionName()

Retrieve the name of the partition in which the shard resides.

Returns:

The partition name.

getType

[String](#) getType()

Retrieve the type of the shard.

Returns:

The shard type.

getDomainName

[String](#) getDomainName()

Retrieve the name of the catalog server grouping administering this shard.

Returns:

The domain name.

getState

[String](#) getState()

Retrieve the state of the shard.

Returns:

The shard state.

getTotalRequestCount

long `getTotalRequestCount()`

Retrieves the number of requests that this shard has processed or forwarded since its inception.

Returns:

A count of the total number of requests.

getActiveRequestCount

long `getActiveRequestCount()`

Retrieves the number of requests currently being processed by this shard.

Returns:

A count of the active requests.

getForwardedRequestCount

long `getForwardedRequestCount()`

Retrieves the number of requests that this shard has forwarded since its inception.

Returns:

A count of the total number of forwarded requests.

getProcessedRequestCount

long `getProcessedRequestCount()`

Retrieves the number of requests that this shard has processed since its inception.

Returns:

A count of the total number of processed requests.

getContainerName

[String](#) `getContainerName()`

Retrieves the name of the container that is hosting this shard.

Returns:

The name of the container.

Since:

WAS XD 6.1.0.3

swapWithPrimary

[String](#) `swapWithPrimary()`

Causes this shard to swap roles with the primary shard for the partition. This shard becomes the primary while the shard that was previously the primary inherits this shard's former role.

If the role swap is not complete within 10 seconds, this operation will timeout.

Returns:

String the contains the return code of the operation

Since:

7.1.0.0 FIX1

See Also:

[ROLE_SWAP_SUCCESSFUL](#), [ROLE_SWAP_REQUESTED_WITH_SAME_TYPE](#), [ROLE_SWAP_TIMEOUT](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.management
Interface SessionMBean

public interface **SessionMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific session. The object name pattern for this MBean is:

com.ibm.websphere.objectgrid:type=Session,name=<id>,host=<host>,ogServerName=<server>

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.0.1, XC10

Method Summary	
l o n g	getAccessedSessionsCount() Gets accessed sessions count
l o n g	getAccessToNonExistentSessionCount() Gets access to non-existent session count
l o n g	getActiveSessionsCount() Gets active sessions count
l o n g	getAffinityBreaksCount() Gets affinity breaks count
l o n g	getCacheDiscardsCount() Gets cache discards count
l o n g	getCreatedSessionsCount() Gets the map count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getInvalidatedByTimeoutCount() Gets invalidated by timeout count
l	

o n g	<p>getInvalidatedSessionsCount() Gets invalidated sessions count</p>
l o n g	<p>getMemoryCount() Gets memory count</p>
S t r i n g	<p>getSessionID() Gets the ID of the session instance associated with this MBean.</p>
S t r i n g	<p>getSessionStatsModule() Gets a string representation of the SessionStatsModule attributes loaded up by the retrieveStatsModule() method.</p>
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . s t a t s . S e s s i o n S t a t s M o d u l e	<p>retrieveStatsModule() Gets the SessionStatsModule used to retrieve statistics associated with the session for this MBean.</p>

Method Detail

retrieveStatsModule

`com.ibm.websphere.objectgrid.stats.SessionStatsModule` **retrieveStatsModule()**

Gets the `SessionStatsModule` used to retrieve statistics associated with the session for this MBean.

Returns:

an `SessionStatsModule` for statistics associated with this session

See Also:

`SessionStatsModule`

getSessionID

[String](#) **getSessionID()**

Gets the ID of the session instance associated with this MBean.

Returns:

the ID of the session instance associated with this MBean.

getSessionStatsModule

[String](#) **getSessionStatsModule()**

Gets a string representation of the `SessionStatsModule` attributes loaded up by the `retrieveStatsModule()` method.

Returns:

String form of `SessionStatsModule`

See Also:

[retrieveStatsModule\(\)](#), `SessionStatsModule`

getCreatedSessionsCount

`long` **getCreatedSessionsCount()**

Gets the map count attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the number of entries in the map

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getNumEntries(boolean)`

getInvalidatedSessionsCount

`long` **getInvalidatedSessionsCount()**

Gets invalidated sessions count

Returns:

the count of invalidated Sessions

getActiveSessionsCount

long **getActiveSessionsCount()**

Gets active sessions count

Returns:
the count of active Sessions

getMemoryCount

long **getMemoryCount()**

Gets memory count

Returns:
memory count

getCacheDiscardsCount

long **getCacheDiscardsCount()**

Gets cache discards count

Returns:
cache discards count

getAffinityBreaksCount

long **getAffinityBreaksCount()**

Gets affinity breaks count

Returns:
affinity breaks count

getInvalidatedByTimeoutCount

long **getInvalidatedByTimeoutCount()**

Gets invalidated by timeout count

Returns:
count

getAccessToNonExistentSessionCount

long **getAccessToNonExistentSessionCount()**

Gets access to non-existent session count

Returns:
count

getAccessedSessionsCount

long **getAccessedSessionsCount()**

Gets accessed sessions count

Returns:
count

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

IBM WebSphere® DataPower® XC10
Appliance

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

Release 2.5 Client API Specification

com.ibm.websphere.objectgrid.management

Interface ServerMBean

All Known Subinterfaces:

[DynamicServerMBean](#)

public interface **ServerMBean**

This MBean interface allows a client process to access different attributes about a specific server process.

Since:

WAS XD 6.0.1, XC10

Method Summary

S
t
r
i
n
g

[getServerName\(\)](#)

Gets the name of the server associated with this MBean.

v
o
i
d

[modifyServerTraceSpec\(String spec\)](#)

Deprecated. This is deprecated in version 7.1. See [DynamicServerMBean.setTraceSpec\(String\)](#)

b
o
o
l
e
a
n

[stopServer\(\)](#)

Stops the server associated with this MBean.

Method Detail

getServerName

[String](#) [getServerName\(\)](#)

Gets the name of the server associated with this MBean.

Returns:

the server name

stopServer

boolean [stopServer\(\)](#)

Stops the server associated with this MBean.

Returns:

true if server was stopped, false if not

modifyServerTraceSpec

void **modifyServerTraceSpec**([String](#) spec)

Deprecated. *This is deprecated in version 7.1. See [DynamicServerMBean.setTraceSpec\(String\)](#)*

Modifies the trace spec for the server associated with this MBean.

Parameters:

spec - new trace specification

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

com.ibm.websphere.objectgrid.management

Interface QuorumManagerMBean

public interface **QuorumManagerMBean**

Each catalog service has a QuorumManager and an associated MBean. The QuorumManager monitors and manages the quorum state of the catalog service grid. When quorum is enabled, the QuorumManager for each catalog service process detects when all catalog services in the grid have quorum or not. This MBean allows querying the current quorum state and allows administrators to force quorum when there is a network failure.

com.ibm.websphere.objectgrid:type=QuorumManager

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

The following notifications are available:

com.ibm.websphere.objectgrid.quorum.lost

Description: Catalog service quorum has been lost.

Message: A translated string identifying the number of active catalog servers and the number in quorum.

com.ibm.websphere.objectgrid.quorum.changed

Description: The catalog service has quorum, but the number of catalog servers required for quorum has changed.

Message: A translated string identifying the number of active catalog servers and the number in quorum.

Since:
7.0, XC10

Field Summary

s
t
a
t
i
c
S
t
r
i
n
g

[QUORUM_CHANGED_NOTIFICATION](#)

s
t
a
t

`String`
[QUORUM_LOST_NOTIFICATION](#)

Method Summary

`String[]`
[getActiveCatalogServerNames\(\)](#)
Retrieves the names of the known active catalog service processes.

`int`
[getActiveCatalogServers\(\)](#)
Retrieve the known number of active catalog service processes.

`int`
[getQuorumCatalogServers\(\)](#)
Retrieve the number of catalog service processes required for quorum.

`void`
[overrideQuorum\(\)](#)
This operation forces surviving catalog service grid processes to reestablish a quorum.

Field Detail

QUORUM_LOST_NOTIFICATION

static final `String` QUORUM_LOST_NOTIFICATION

See Also:

[Constant Field Values](#)

QUORUM_CHANGED_NOTIFICATION

static final `String` QUORUM_CHANGED_NOTIFICATION

See Also:

[Constant Field Values](#)

Method Detail

overrideQuorum

`void` **overrideQuorum()**
throws [Exception](#)

This operation forces surviving catalog service grid processes to reestablish a quorum.

If a portion the catalog service grid fails or is divided due to a network failure, the grid will lose quorum. Once the administrator identifies the failure and the viable portion of the grid, this operation can be invoked on any of the surviving catalog service processes to reestablish a quorum. Reestablishing a quorum will allow the catalog service to

continue to react to failures and topology changes.

Throws:

[Exception](#)

getActiveCatalogServers

int `getActiveCatalogServers()`

Retrieve the known number of active catalog service processes.

Returns:

the known number of active catalog service processes.

getQuorumCatalogServers

int `getQuorumCatalogServers()`

Retrieve the number of catalog service processes required for quorum.

Returns:

the number of catalog service processes required for quorum.

getActiveCatalogServerNames

[String](#)[] `getActiveCatalogServerNames()`

Retrieves the names of the known active catalog service processes.

Returns:

the names of the known active catalog service processes.

Since:

7.1

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management

Interface QueryManagerMBean

public interface **QueryManagerMBean**

This MBean interface allows a client process to perform operations on and get status from an ObjectGrid Query Manager running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=QueryManager,name=<grid name>,mapset=<mapset name>,partition=<partition number>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1.0.5, XC10

Method Summary	
d o u b l e	<p>getPlanCreationTime(String query) Gets the query's plan creation time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryExecutionCount(String query) Gets the query's execution count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryExecutionTime(String query) Gets the query's execution time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryFailureCount(String query) Gets the query's failure count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getQueryResultCount(String query) Gets the query's result count attribute loaded up by the retrieveStatsModule() method.</p>

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
Q
u
e
r
y
S
t
a
t
s
M
o
d
u
l
e

[retrieveStatsModule\(String query\)](#)

Gets the QueryStatsModule used to retrieve statistics associated with the specified query String

Method Detail

getPlanCreationTime

double `getPlanCreationTime(String query)`

Gets the query's plan creation time attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the plan creation time for this query in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getPlanCreationTime(boolean copy)`

getQueryExecutionTime

double `getQueryExecutionTime(String query)`

Gets the query's execution time attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the execution time for this query in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryExecutionTime(boolean copy)`

getQueryExecutionCount

double `getQueryExecutionCount(String query)`

Gets the query's execution count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the execution count for this query

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryExecutionCount(boolean copy)`

getQueryResultCount

double `getQueryResultCount(String query)`

Gets the query's result count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the result count for this query

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryResultCount(boolean copy)`

getQueryFailureCount

double `getQueryFailureCount(String query)`

Gets the query's failure count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the failure count for this query

See Also:

[retrieveStatsModule\(String\)](#), `QueryStatsModule.getQueryFailureCount(boolean copy)`

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.QueryStatsModule `retrieveStatsModule(String query)`

Gets the `QueryStatsModule` used to retrieve statistics associated with the specified query String

Returns:

an `QueryStatsModule` for statistics associated with the specified query String

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management

Interface PlacementServiceMBean

All Superinterfaces:

com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean

```
public interface PlacementServiceMBean
extends com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean
```

This MBean interface allows a client process to perform operations on and get status from the PlacementService running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=PlacementService
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary	
s t a t i c i n t	<p>ALL</p> <p>Constant representing a all shard types</p>
s t a t i c i n t	<p>ASYNCHRONOUS_REPLICA</p> <p>Constant representing an asynchronous replica shard type.</p>
s t a t i c i n t	<p>PRIMARY</p> <p>Constant representing a primary shard type.</p>
s t a	

t
i
c

i
n
t

[SYNCHRONOUS_REPLICA](#)

Constant representing a synchronous replica shard type.

Fields inherited from interface

com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean

HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE, HEARTBEAT_FREQUENCY_LEVEL_RELAXED,
HEARTBEAT_FREQUENCY_LEVEL_TYPICAL

Method Summary

S
t
r
i
n
g

[balanceShardTypes](#)([String](#) objectGridName, [String](#) mapSetName)

The placement service will examine the distribution of primaries and replicas for a given mapSet and attempt (if zone rules and other balancing constraints allow) to achieve a consistent primary to replica ratio across the set of containers.

S
t
r
i
n
g

[balanceStatus](#)([String](#) objectGridName, [String](#) mapSetName)

Check the balance status (suspended or resumed) for a specified MapSet.

S
t
r
i
n
g

[collectContainerStatus](#)([String](#) objectGridName, [String](#) mapSetName)

Retrieves the container status for all containers in the domain.

b
o
l
e
a
n

[enableForPlacement](#)([String](#) containerName)

Re-enables a disabled container for placement.

S
t
r
i
n
g

[getCoreGroups](#)()

Gets the coregroup status.

T
a
b
u
l
a
r
D
a
t
a

[getDisabledForPlacement](#)()

Retrieves TabularData of containers which have been disabled because of the failure of shard placement operations.

b
o
l
e
a
n

[getInitialized](#)()

Returns whether this instance of the Placement Service finished initialization.

S t r i n g	<p>getObjectGridNames() Gets the names of all ObjectGrids and their mapsets in the domain.</p>
S t r i n g	<p>listCoreGroupMembers(String coreGroupName) List the coregroup members for a given coregroup.</p>
S t r i n g	<p>listObjectGridPlacement(String objectGridName, String mapSetName) List the placement of shards for each container in the domain.</p>
S t r i n g	<p>listObjectGridPlacementStatus(String objectGridName, String mapSetName) List the current placement status.</p>
S t r i n g	<p>listPartition(String objectGridName, String mapSetName, String partitionId) List the partition placement status in the domain.</p>
S t r i n g	<p>listShards(String objectGridName, String mapSetName, String containerName, int mask) List the shard placement status.</p>
S t r i n g	<p>listVerifiedRoutingTable(String objectGridName) Deprecated.</p>
S t r i n g	<p>replaceLostShards(String objectGridName, String mapSetName) Lost shards are placed onto the UNREPAIRED container when autoReplaceLostShards is disabled.</p>
S t r i n g	<p>resumeBalancing(String objectGridName, String mapSetName) Execute balancing operation at next opportunity and allow execution of future balancing attempts for the map set specified.</p>
L i s t	<p>retrieveAllServersJMXAddresses() Retrieves a List of JMX addresses for all servers that have registered with the placement service.</p>
S t r	<p>retrieveMapSetName(String gridName, String mapName)</p>

i n q	Retrieves the name of the MapSet in which the specified map is defined.
L i s t	retrieveServerJMXAddress (String hostName, String serverName) Retrieves a List of JMX address strings for a specific host and server name that has registered with the placement service.
T a b u l a r D a t a	retrieveServerJMXAddressesWithInfo (String hostName, String serverName) Retrieves TabularData of JMX address strings for a specific host and server name that has registered with the placement service.
S t r i n g	suspendBalancing (String objectGridName, String mapSetName) Prevent future balancing attempts for a specific map set.
S t r i n g	tearDownServers (String [] servers) Each of the container servers that are passed into this method will be stopped.
S t r i n g	triggerPlacement (String objectGridName, String mapSetName) Placement normally occurs implicitly after an event such as an ObjectGrid container starting or stopping.

Methods inherited from interface com.ibm.websphere.objectgrid.management.CoreGroupServiceMBean
getHeartBeatFrequencyLevel, setHeartBeatFrequencyLevel

Field Detail

PRIMARY

static final int PRIMARY

Constant representing a primary shard type.

See Also:

[Constant Field Values](#)

SYNCHRONOUS_REPLICA

static final int SYNCHRONOUS_REPLICA

Constant representing a synchronous replica shard type.

See Also:

[Constant Field Values](#)

ASYNCHRONOUS_REPLICA

static final int ASYNCHRONOUS_REPLICA

Constant representing an asynchronous replica shard type.

See Also:

[Constant Field Values](#)

ALL

static final int ALL

Constant representing a all shard types

See Also:

[Constant Field Values](#)

Method Detail

retrieveServerJMXAddress

[List](#) retrieveServerJMXAddress([String](#) hostName,
[String](#) serverName)

Retrieves a List of JMX address strings for a specific host and server name that has registered with the placement service.

Parameters:

hostName - The name of the host to retrieve the JMX addresses.

serverName - The name of the server to retrieve the JMX addresses.

Returns:

the List of all JMX address strings for the specified host and server name.

retrieveServerJMXAddressesWithInfo

[TabularData](#) retrieveServerJMXAddressesWithInfo([String](#) hostName,
[String](#) serverName)

Retrieves TabularData of JMX address strings for a specific host and server name that has registered with the placement service. Null host or null server names can be used to retrieve several results. Using a null host and null server name will retrieve all of the servers. The TabularData contains CompositeData with the following items, where each CompositeData represents a server:

Item Name	Type	Description
------------------	-------------	--------------------

JMXServiceURL	String	JMX Service URL
---------------	--------	-----------------

HostName	String	Host name
----------	--------	-----------

ServerName	String	Server Name
------------	--------	-------------

Parameters:

hostName - The name of the host for which to retrieve JMX addresses. Use null or an empty string to retrieve JMX addresses for any host.

serverName - The name of the server for which to retrieve JMX addresses. Use null or an empty string to retrieve JMX addresses for any server.

Returns:

the TabularData of all JMX address strings for the specified host and server name.

Since:

8.5

retrieveAllServersJMXAddresses

[List](#) retrieveAllServersJMXAddresses()

Retrieves a List of JMX addresses for all servers that have registered with the placement service.

Returns:

the List of all servers' JMX addresses

collectContainerStatus

[String](#) collectContainerStatus([String](#) objectGridName,
[String](#) mapSetName)

Retrieves the container status for all containers in the domain.

The results are returned in the following format:

```
<container name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>">
  <shard type="<type>" partitionName="<partition>"/>
</container>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get container status.

mapSetName - The name of the mapset for which to get the container status.

Returns:

The String status object for all containers in XML form.

listObjectGridPlacement

[String](#) listObjectGridPlacement([String](#) objectGridName,
[String](#) mapSetName)

List the placement of shards for each container in the domain.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset">
  <container name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>">
    <shard type="<type>" partitionName="<partition>" reserved="<true>"/>
  /container>
</objectGrid>
```

NOTE: The default value for the "reserved" attribute is false.

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.

mapSetName - The name of the mapset for which to get the placement status.

Returns:

The placement status in XML form.

listObjectGridPlacementStatus

```
String listObjectGridPlacementStatus(String objectGridName,  
                                     String mapSetName)
```

List the current placement status.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <configuration>  
    <attribute name="<placementStrategy>" value="<strategy>"/>  
    <attribute name="<numInitialContainers>" value="<num>"/>  
    <attribute name="<minSyncReplicas>" value="<min>"/>  
    <attribute name="<developmentMode>" value="<mode>"/>  
  </configuration>  
  <runtime>  
    <attribute name="<numContainers>" value="<num>"/>  
    <attribute name="<numMachines>" value="<num>"/>  
    <attribute name="<numOutstandingWorkItems>" value="<num>"/>  
    <attribute name="<numActiveZones>" value="<num>"/>  
  </runtime>  
</objectGrid>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.

mapSetName - The name of the mapset for which to get the placement status.

Returns:

The placement status in XML form.

getCoreGroups

```
String getCoreGroups()
```

Gets the coregroup status.

The results are returned in the following format:

```
<coreGroup name="<coregroup>">  
  <coreGroupLeader hostName="<host>" serverName="<server>"/>  
  <coreGroupMember hostName="<host>" serverName="<server>"/>  
</coreGroup>
```

Returns:

the coregroup status in XML form.

listCoreGroupMembers

```
String listCoreGroupMembers(String coreGroupName)
```

List the coregroup members for a given coregroup.

The results are returned in the following format:

```
<coreGroup name="<coregroup>">  
  <coreGroupMember hostName="<host>" serverName="<server>"/>  
</coreGroup>
```

Parameters:

coreGroupName - The name of the coregroup for which to get the members.

Returns:

The coregroup members in XML form.

listPartition

```
String listPartition(String objectGridName,  
                    String mapSetName,  
                    String partitionId)
```

List the partition placement status in the domain. The results are returned in the following format:

```
<partition name="<partition>">  
  <shard type="<type>" containerName="<container>" hostName="<host>" serverName="<server>"/>  
</partition>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.
mapSetName - The name of the mapset for which to get the placement status.
partitionId - The name of the partition for which to get the placement status.

Returns:

The partition placement status in the XML form.

listShards

```
String listShards(String objectGridName,  
                String mapSetName,  
                String containerName,  
                int mask)
```

List the shard placement status.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <container name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>">  
    <shard type="<type>" partitionName="<partition>"/>  
  </container>  
</objectGrid>
```

Parameters:

objectGridName - The name of the ObjectGrid for which to get placement status.
mapSetName - The name of the mapset for which to get the placement status.
containerName - The name of the container for which to get the placement status. If empty string (""), get shard placement for all containers.
mask - The Integer mask to determine for which shard types to get status.

Returns:

The shard placement status in XML form.

See Also:

[ALL](#), [PRIMARY](#), [SYNCHRONOUS_REPLICA](#), [ASYNCHRONOUS_REPLICA](#)

getObjectGridNames

```
String getObjectGridNames()
```

Gets the names of all ObjectGrids and their mapsets in the domain.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>"/>
```

Returns:

the names of all ObjectGrids and their mapsets in the domain in XML form.

replaceLostShards

`String replaceLostShards(String objectGridName,
String mapSetName)`

Lost shards are placed onto the UNREPAIRED container when `autoReplaceLostShards` is disabled. Shards on the UNREPAIRED will not be placed until this method is called.

Calling this method will move shards off the UNREPAIRED container onto the UNASSIGNED container.

Balance and placement operations will be queued up for the MapSet specified. These operations will execute when all outstanding placement work from previous events has completed.

The string returned is an XML representation of the shards that moved as a result of the call to this method.

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <shard type="<type>" partitionName="<partition>">  
    <currentContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />  
    <previousContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />  
  </shard>  
</objectGrid>
```

The returned XML will look as follows when no shards have been moved:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">  
  <!-- No shards were moved -->  
</objectGrid>
```

Parameters:

`objectGridName` - replace lost shards for this ObjectGrid

`mapSetName` - replace lost shards for this MapSet

Returns:

An XML String containing shards that have moved

Since:

WAS XD 6.1.0.5

triggerPlacement

`String triggerPlacement(String objectGridName,
String mapSetName)`

Placement normally occurs implicitly after an event such as an ObjectGrid container starting or stopping.

Calling this method will trigger a placement operation for the ObjectGrid and MapSet specified.

Under normal circumstances, the `numInitialContainers` attribute (in the deployment policy) must be met in order for placement to occur. However, when this method is called, the `numInitialContainers` value is ignored.

The string returned is an XML representation of the shards that moved as a result of the call to this method.


```

<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <shard type="<type>" partitionName="<partition>">
    <currentContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
    <previousContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
  </shard>
</objectGrid>

```

The returned XML will look as follows when no shards have been moved:

```

<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <!-- No shards were moved -->
</objectGrid>

```

Parameters:

objectGridName - trigger placement for this ObjectGrid
mapSetName - trigger placement for this MapSet

Returns:

An XML String containing shards that have moved

Since:

WAS XD 6.1.0.5

See Also:

ObjectGridDeployment.addMapSet(com.ibm.websphere.objectgrid.deployment.MapSet),
MapSet.setNumInitialContainers(int)

tearDownServers

[String](#) tearDownServers([String](#)[] servers)

Each of the container servers that are passed into this method will be stopped. If the server cannot be reached, all of the server's artifacts will be removed.

Use this method if servers are found to be in a corrupt state or bindings need to be cleared from the catalog server.

The string returned is an XML representation of the results of the attempt to tear down each of the servers. If the command is successful, the XML will look as follows:

```

<domain name="<domain>">
  <server name="<server>" tearDownSuccessful="true" />
  <server name="<server>" tearDownSuccessful="true" />
</domain>

```

If the command is not successful, the string will look as follows (where the exception element is only present if an exception is part of the failure):

```

<domain name="<domain>">
  <server name="<server>" tearDownSuccessful="false" reason="<String>">
    <exception type="<String>" message="<String>" stack="<String>" />
  </server>
</domain>

```

Parameters:

servers - String array of servers to tear down.

Returns:

An XML String containing the results of tear down attempts.

Since:

WAS XD 6.1.0.5 FIX2

listVerifiedRoutingTable

[@Deprecated](#)

[String](#) listVerifiedRoutingTable([String](#) objectGridName)

Deprecated.

This method is deprecated. The `com.ibm.websphere.objectgrid.client.RouteTableValidation` utility replaces this method.

Calling this method will return an XML string of the current known routing table. The Placement service will contact each shard and return state on whether it was able to verify that's shard's existence. All shards will be included in the XML doc, whether they were reachable or not. The user can use the `reachable` attribute below to filter valid or invalid shards.

```
<objectGrid name="<objectgrid>" name="<name>">
  <primary zone="<zone>"> partition="<partition>"> state="<reachable>"> ipaddress="<ipaddress>">
  <replica zone="<zone>"> partition="<partition>"> state="<reachable>"> ipaddress="<ipaddress>">
  </primary>
</objectGrid>
```

Parameters:

objectGridName - retrieve routing table for this ObjectGrid

Returns:

An XML String containing a pre-verified routing table

Since:

WAS XD 6.1.0.5 FIX2

retrieveMapSetName

[String](#) retrieveMapSetName([String](#) gridName,
[String](#) mapName)

Retrieves the name of the MapSet in which the specified map is defined.

Parameters:

gridName - the name of the ObjectGrid

mapName - the name of the map

Returns:

the name of the MapSet in which the specified map is defined.

Since:

7.0

getInitialized

boolean `getInitialized()`

Returns whether this instance of the Placement Service finished initialization.

Returns:

true if initialized, false if not initialized.

Since:

8.5

balanceShardTypes

[String](#) `balanceShardTypes`([String](#) objectGridName,
[String](#) mapSetName)

The placement service will examine the distribution of primaries and replicas for a given mapSet and attempt (if zone rules and other balancing constraints allow) to achieve a consistent primary to replica ratio across the set of containers.

If the number of primaries or the number of replicas do not divide evenly across the containers, some tolerance must be allowed for the ratio to differ from container to container. However, the difference in the number of primaries from one container to the next will not be greater than 1. Similarly, the difference in the number of replicas from one container to the next will not be greater than 1.

Null arguments are not allowed as input to this method.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <shard type="<type>" partitionName="<partition>">
    <currentContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
    <previousContainer name="<container>" zoneName="<zone>" hostName="<host>" serverName="<server>" />
  </shard>
</objectGrid>
```

If no shards were moved or a problem was encountered attempting to execute this method, no shard elements will appear in the XML output. A detail element will appear instead. The message attribute will have further information.

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <detail message="<message>" />
</objectGrid>
```

Parameters:

objectGridName - the grid
mapSetName - the map set within the grid

Returns:

An XML String containing the results of the attempt to redistribute shards for better primary/replica balance

Since:

7.1.1

suspendBalancing

[String](#) `suspendBalancing`([String](#) objectGridName,
[String](#) mapSetName)

Prevent future balancing attempts for a specific map set. Balancing work that is in progress will be allowed to complete.

Other placement activities are allowed to execute while balancing is suspended.

- shard promotion due to container loss
- shard role swap
- shard reservation
- triggerPlacement
- replaceLostShards

Balancing will remain suspended until it is resumed by calling [resumeBalancing\(String, String\)](#).

Null arguments are not allowed as input to this method.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>" />
</objectGrid>
```

Additionally, an optional detail element may be contained within the suspendBalancing element. The detail element will include additional data regarding execution of this method. The XML result will be in the following format when a detail element is included:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>">
    <detail message="<message>" />
  </suspendBalancing/>
</objectGrid>
```

Parameters:

objectGridName - suspend balancing for the map set specified within this ObjectGrid
mapSetName - suspend balancing for this map set

Returns:

An XML String containing the results of the attempt to suspend balancing

Since:

7.1.0.3

See Also:

[resumeBalancing\(String, String\)](#)

resumeBalancing

```
String resumeBalancing(String objectGridName,  
                        String mapSetName)
```

Execute balancing operation at next opportunity and allow execution of future balancing attempts for the map set specified. Balancing is executed in reaction to key placement events. Such events include containers starting and containers stopping.

By default, balancing work is executed unless [suspendBalancing\(String, String\)](#) has been called for the map set.

Null arguments are not allowed as input to this method.

The results are returned in the following format:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>" />
</objectGrid>
```

Additionally, an optional detail element may be contained within the suspendBalancing element. The detail element will include additional data regarding execution of this method. The XML result will be in the following format when a detail element is included:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <suspendBalancing currentValue="<currentValue>" previousValue="<previousValue>">
    <detail message="<message>" />
  </suspendBalancing/>
</objectGrid>
```

Parameters:

objectGridName - resume balancing for the map set specified within this ObjectGrid
mapSetName - resume balancing for this map set

Returns:

An XML String containing the results of the attempt to resume balancing

Since:

7.1.0.3

See Also:

[suspendBalancing\(String, String\)](#)

balanceStatus

[String](#) **balanceStatus**([String](#) objectGridName,
[String](#) mapSetName)

Check the balance status (suspended or resumed) for a specified MapSet.

Null arguments are not allowed as input to this method.

The string returned is an XML representation of the balance status. The XML will look as follows:

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <balanceStatus suspended="<suspended>" />
</objectGrid>
```

Additionally, an optional detail element may be contained within the balanceStatus element. When balancing has been pre-suspended, the message attribute of the detail element will contain the following message.

```
<objectGrid name="<objectgrid>" mapSetName="<mapset>">
  <balanceStatus suspended="true" >
    <detail message="Balancing has been pre-suspended for this mapSet." />
  </balanceStatus>
</objectGrid>
```

Parameters:

objectGridName - check balance status for the map set specified within this ObjectGrid
mapSetName - check balance status for this map set

Returns:

An XML String containing the balance status

Since:

7.1.1

See Also:

[suspendBalancing\(String, String\)](#), [resumeBalancing\(String, String\)](#)

enableForPlacement

boolean **enableForPlacement**([String](#) containerName)

Re-enables a disabled container for placement. A container may become disabled because of a failure to place a shard into the container.

Use the [getDisabledForPlacement\(\)](#) attribute to determine which containers are disabled.

Parameters:

containerName - The name of the container to re-enable.

Returns:

Answers true if the container's status was changed from disabled to enabled, false if

the container was already enabled for placement.

Since:

8.6, XC10 2.5

getDisabledForPlacement

[TabularData](#) `getDisabledForPlacement()`

Retrieves TabularData of containers which have been disabled because of the failure of shard placement operations. The TabularData contains CompositeData with the following items, where each CompositeData represents a container:

Item Name Type Description

Container String A container that has been disabled for placement

Returns:

A TabularData of the names of disabled containers.

Since:

8.6, XC10 2.5

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
ew	ge	ss	ed	ted			
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All		
			Classes				

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid.management

Interface PlacementMediationServiceMBean

public interface PlacementMediationServiceMBean

This MBean interface allows a client process to perform operations on and get status from the PlacementMediationService running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=PlacementMediationService
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

7.1, XC10

Method Summary

C o m p o s i t i o n s	<p>dismissLink(String foreignDomain) Dismiss a previously established link with the foreign domain specified.</p>
C o m p o s i t i o n s	<p>establishLink(String foreignDomain, String endPoints) Establish a link between this domain and the foreign domain specified.</p>
T a b l e s	<p>getLinkedDomains() Retrieve the foreign domains that have an active link with the local domain.</p>

[getLinkedDomainsWithGrids\(\)](#)

Retrieve the foreign domains that have an active link with the local domain and the map sets eligible for linking.

Method Detail

establishLink

[CompositeData](#) **establishLink**([String](#) foreignDomain, [String](#) endPoints)

Establish a link between this domain and the foreign domain specified. This is functionally equivalent to providing the foreign domain and its end points in the server properties file at server startup time.

Domains that are linked will share placement with each other. When compatible map sets are detected within linked domains, a multi-primary topology will be achieved. Data written to a primary in either domain will be asynchronously replicated to the other domain.

The result is a [CompositeData](#) that includes the following items:

Item Name Type Description

Result	String	The result of the attempt.
StatusBefore	String	The status of the link before the attempt was made to establish the link.
StatusAfter	String	The status of the link after the attempt was made to establish the link.

Parameters:

- foreignDomain - the name of the foreign domain
- endPoints - end points of the foreign domain

Returns:

[CompositeData](#) representing the status of the attempt to link with the foreign domain

See Also:

- [CatalogServerProperties.setForeignDomains\(String\)](#),
- [CatalogServerProperties.setDomainEndPoints\(String, String\)](#)

dismissLink

[CompositeData](#) **dismissLink**([String](#) foreignDomain)

Dismiss a previously established link with the foreign domain specified. Any map sets that were participating in a multi-primary topology will be disconnected from each other. Data will no longer be replicated from between domains.

The result is a [CompositeData](#) that includes the following items:

<u>Item Name</u>	<u>Ty</u>	<u>Description</u>
-------------------------	------------------	---------------------------

Result	String	The result of the attempt. Can be one of: SUCCESS, FAILURE, NOP
Status Before	String	The status of the link before the attempt was made to dismiss the link. Can be one of: LINKED, ESTABLISHING_LINK, UNLINKED, DISMISSING_LINK
Status After	String	The status of the link after the attempt was made to dismiss the link. Can be one of: LINKED, ESTABLISHING_LINK, UNLINKED, DISMISSING_LINK

Parameters:

foreignDomain - the name of the foreign domain

Returns:

CompositeData representing the status of the attempt to dismiss the link with the foreign domain

getLinkedDomains

[TabularData](#) `getLinkedDomains()`

Retrieve the foreign domains that have an active link with the local domain.

The result is a TabularData where each row is a CompositeData that includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

Domain	String	The name of the foreign domain linked to the local domain.
--------	--------	--

Returns:

TabularData representing the foreign domains linked to this domain

getLinkedDomainsWithGrids

[TabularData](#) `getLinkedDomainsWithGrids()`

Retrieve the foreign domains that have an active link with the local domain and the map sets eligible for linking.

The result is a TabularData where each row is a CompositeData that includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

Domain	String	The name of the foreign domain linked to the local domain.
ObjectGrid	String	The name of the ObjectGrid that is compatible with the foreign domain.
MapSet	String	The name of the map set that is compatible with the foreign domain.

Returns:

TabularData representing the foreign domains and eligible map sets linked to this domain

Since:

8.6, XC10 2.5

com.ibm.websphere.objectgrid.management

Interface ObjectGridMBean

public interface **ObjectGridMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific ObjectGrid on a server process. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ObjectGrid,name=<objectgrid>,mapset=<mapset>,partition=<partition id>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.0.1, XC10

Method Summary	
S t r i n g	<p>getContainerName() Gets the name of the container name containing the replication group member for the ObjectGrid associated with this MBean.</p>
l o n g	<p>getCurrentRevision() Retrieves the current revision number of this ObjectGrid shard.</p>
S t r i n g	<p>getDomainName() Retrieves the domain name of this ObjectGrid shard.</p>
I n t e r f a c e	<p>getKnownRevisions() An ObjectGrid shard may exist over several different lifetimes.</p>
S t r i n g	<p>getLifetimeId() Retrieves the lifetime id for this ObjectGrid shard.</p>

n g	
S t r i n g	getObjectGridName() Gets the name of the ObjectGrid associated with this MBean.
l o n g	getOGCount() Gets the ObjectGrid count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGMaxTranTime() Gets the maximum transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
d o u b l e	getOGMeanTranTime() Gets the mean transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGMinTranTime() Gets the minimum transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
S t r i n g	getOGStatsModule() Gets a string representation of the OGStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGTotalTranTime() Gets the total transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getOGTransPerSecond() Transactions per second attribute loaded up by the retrieveStatsModule call.
T a b l e	getPrimaryShardLinks() Get the shard's list of foreign or domestic linked primaries.
S t r i n g	getServerName() Gets the name of the server containing the replication group member for the ObjectGrid associated with this MBean.
c o m .	

i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
O
G
S
t
a
t
s
M
o
d
u
l
e

[retrieveStatsModule\(\)](#)

Gets the OGStatsModule used to retrieve statistics associated with the ObjectGrid for this MBean.

Method Detail

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.OGStatsModule **retrieveStatsModule()**

Gets the OGStatsModule used to retrieve statistics associated with the ObjectGrid for this MBean.

Returns:

an OGStatsModule for statistics associated with this ObjectGrid

See Also:

OGStatsModule

getObjectGridName

[String](#) getObjectGridName()

Gets the name of the ObjectGrid associated with this MBean.

Returns:

name of the ObjectGrid

getServerName

[String](#) getServerName()

Gets the name of the server containing the replication group member for the ObjectGrid associated with this MBean.

Returns:

the name of server containing the replication group member for the ObjectGrid associated with this MBean.

getContainerName

[String](#) getContainerName()

Gets the name of the container name containing the replication group member for the ObjectGrid associated with this MBean.

Returns:

the name of container containing the replication group member for the ObjectGrid associated with this MBean.

Since:

8.5

getOGStatsModule

[String](#) getOGStatsModule()

Gets a string representation of the OGStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

String form of OGStatsModule

See Also:

[retrieveStatsModule\(\)](#), OGStatsModule

getOGCount

long getOGCount()

Gets the ObjectGrid count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

the number of transactions

See Also:

[retrieveStatsModule\(\)](#), OGStatsModule.getTransactionTime(String, boolean)

getOGMaxTranTime

long getOGMaxTranTime()

Gets the maximum transaction time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

the maximum transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), OGStatsModule.getTransactionTime(String, boolean)

getOGMinTranTime

long **getOGMinTranTime()**

Gets the minimum transaction time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the minimum transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getOGMeanTranTime

double **getOGMeanTranTime()**

Gets the mean transaction time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the mean transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getOGTotalTranTime

long **getOGTotalTranTime()**

Gets the total transaction time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the total transaction time for the ObjectGrid in milliseconds

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getOGTransPerSecond

long **getOGTransPerSecond()**

Transactions per second attribute loaded up by the `retrieveStatsModule` call. `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

the transactions per second for the ObjectGrid

See Also:

[retrieveStatsModule\(\)](#), `OGStatsModule.getTransactionTime(String, boolean)`

getCurrentRevision

long **getCurrentRevision()**

Retrieves the current revision number of this ObjectGrid shard.

Returns:

the current revision number of this ObjectGrid shard.

Since:

7.1

getDomainName

[String](#) **getDomainName()**

Retrieves the domain name of this ObjectGrid shard.

Returns:

the name of the domain name of this ObjectGrid shard.

Since:

7.1

getLifetimeId

[String](#) `getLifetimeId()`

Retrieves the lifetime id for this ObjectGrid shard.

Returns:

the lifetime id for this ObjectGrid shard.

Since:

7.1

getKnownRevisions

[TabularData](#) `getKnownRevisions()`

An ObjectGrid shard may exist over several different lifetimes. As such, each shard instance will have a unique lifetime id and revision number associated with it. This method returns a TabularData object representing the known history of revision numbers for each lifetime. Each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Domain	String	The domain name of this ObjectGrid shard.	7.1
Server	String	The name of the server owning the lifetime id.	8.5
LifetimeId	String	The lifetime id of this ObjectGrid shard.	7.1
Revision	Long	The revision of this ObjectGrid shard.	7.1

Returns:

TabularData representing the known lifetimes and revisions of this shard.

Throws:

[OpenDataException](#)

Since:

7.1

See Also:

[TabularData](#)

getPrimaryShardLinks

[TabularData](#) `getPrimaryShardLinks()`

Get the shard's list of foreign or domestic linked primaries.

This method returns a TabularData object representing the current state of each primary shard link.

Each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
RemoteDomain	String	The catalog service domain name of the remote primary shard..
RemoteContainer	String	The container name of the remote primary shard.
Status	String	The status of the link. Valid states include: online and recovery.

Returns:

TabularData representing the linked primaries

Since:

7.1.1

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

Release 2.5 Client API Specification

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

com.ibm.websphere.objectgrid.management

Interface MapMBean

public interface **MapMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific map on a server process. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ObjectMap,name=<map>,partition=<partition id>,objectgrid=<objectgrid>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.0.1, XC10

Method Summary	
S t r i n g	<p>getContainerName() Gets the name of the container containing the replication group member for the map associated with this MBean.</p>
d o u b l e	<p>getMapBatchUpdateMaxTime() Gets the maximum batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>
d o u b l e	<p>getMapBatchUpdateMeanTime() Gets the mean batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>
d o u b l e	<p>getMapBatchUpdateMinTime() Gets the minimum batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>
d o u b l e	<p>getMapBatchUpdateTotalTime() Gets the total batch update time attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.</p>

l o n g	getMapCountStatistic() Gets the map count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getMapGetCountStatistic() Gets the get count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getMapHitCountStatistic() Gets the hit count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
d o u b l e	getMapHitRateStatistic() Gets the hit rate attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
S t r i n g	getMapName() Gets the name of the map associated with this MBean.
S t r i n g	getMapStatsModule() Gets a string representation of the MapStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.
l o n g	getMapUsedBytes() Gets the used bytes attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.
S t r i n g	getObjectGridName() Gets the name of the ObjectGrid containing the map associated with this MBean.
i n t	getPartitionId() Retrieves the partition identifier for this map instance.
S t r i n g	getServerName() Gets the name of the server containing the replication group member for the map associated with this MBean.
I a b l e	retrieveEntries(String regex) Operation to iterate through all of the entries in this map, convert the key to string form, then match the string against the regular expression if passed, finally return the matching entries.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
M
a
p
S
t
a
t
s
M
o
d
u
l
e

[retrieveStatsModule\(\)](#)

Gets the MapStatsModule used to retrieve statistics associated with the map for this MBean.

Method Detail

retrieveStatsModule

`com.ibm.websphere.objectgrid.stats.MapStatsModule` **retrieveStatsModule()**

Gets the MapStatsModule used to retrieve statistics associated with the map for this MBean.

Returns:

A MapStatsModule for statistics associated with this map.

See Also:

MapStatsModule

getMapName

[String](#) **getMapName()**

Gets the name of the map associated with this MBean.

Returns:

The name of the map.

getObjectGridName

[String](#) getObjectGridName()

Gets the name of the ObjectGrid containing the map associated with this MBean.

Returns:

The name of the ObjectGrid for the map associated with this MBean.

getServerName

[String](#) getServerName()

Gets the name of the server containing the replication group member for the map associated with this MBean.

Returns:

The name of server containing the replication group member for the map associated with this MBean.

getMapStatsModule

[String](#) getMapStatsModule()

Gets a string representation of the MapStatsModule attributes loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

The String form of MapStatsModule

See Also:

[retrieveStatsModule\(\)](#), MapStatsModule

getMapCountStatistic

long getMapCountStatistic()

Gets the map count attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

The number of entries in the map.

See Also:

[retrieveStatsModule\(\)](#), MapStatsModule.getNumEntries(boolean)

getMapHitRateStatistic

double getMapHitRateStatistic()

Gets the hit rate attribute loaded up by the retrieveStatsModule() or refreshStatsModule() method.

Returns:

The hit rate for the map.

See Also:

[retrieveStatsModule\(\)](#), MapStatsModule.getHitRate(boolean)

getMapGetCountStatistic

long **getMapGetCountStatistic()**

Gets the get count attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The get count for the map.

Since:

7.1

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getHitRate(boolean)`

getMapUsedBytes

long **getMapUsedBytes()**

Gets the used bytes attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

The used bytes statistics are accurate only when you are using simple objects or the `COPY_TO_BYTES` copy mode.

Returns:

The number of bytes in use by the map.

Since:

7.1

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getUsedBytes(boolean)`

getMapHitCountStatistic

long **getMapHitCountStatistic()**

Gets the hit count attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The hit count for the map.

Since:

7.1

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getHitRate(boolean)`

getMapBatchUpdateMeanTime

double **getMapBatchUpdateMeanTime()**

Gets the mean batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The mean batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getMapBatchUpdateMaxTime

double **getMapBatchUpdateMaxTime()**

Gets the maximum batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The maximum batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getMapBatchUpdateMinTime

double **getMapBatchUpdateMinTime()**

Gets the minimum batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The minimum batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getMapBatchUpdateTotalTime

double **getMapBatchUpdateTotalTime()**

Gets the total batch update time attribute loaded up by the `retrieveStatsModule()` or `refreshStatsModule()` method.

Returns:

The total batch update time for the map in milliseconds.

See Also:

[retrieveStatsModule\(\)](#), `MapStatsModule.getBatchUpdateTime(boolean)`

getPartitionId

int **getPartitionId()**

Retrieves the partition identifier for this map instance.

Returns:

The partition identifier.

Since:

WAS XD 6.1.0.4

getContainerName

[String](#) **getContainerName()**

Gets the name of the container containing the replication group member for the map associated with this MBean.

Returns:

The name of container containing the replication group member for the map associated with this MBean.

Since:

8.5

retrieveEntries

[TabularData](#) **retrieveEntries([String](#) regex)**

Operation to iterate through all of the entries in this map, convert the key to string form, then match the string against the regular expression if passed, finally return the matching entries. This method could potentially return a very large data structure so care should be taken to ensure the regular expression will reduce the number of keys appropriately.

Each `CompositeData` (row in the `TabularData`) contains the following items:

Item Name	Type	Description
KeyName	String	The domain name of this ObjectGrid shard.
LifetimeIndex	Short	The lifetime index for revisioning.
Revision	Long	The revision number of the last update.

Parameters:

regex - the regular expression to apply to the String form of the key. It should be used in narrowing the entries returned. If null, all entries are returned.

Returns:

A table of entries containing the user readable (String) form of the key and some meta information about the entry.

Since:

7.1.1

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [ALL Classes](#)

Release 2.5 Client API Specification

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.management
Interface HashIndexMBean

public interface **HashIndexMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific HashIndex on a server process. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=HashIndex,name=<index-name>,partition=<partition id>,objectgrid=<objectgrid>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:
WAS XD 6.1.0.5, XC10

Method Summary	
d o u b l e	getBatchUpdateCount() Gets the index's batchupdate count attribute loaded up by the retrieveStatsModule() method.
d o u b l e	getFindCollisionCount() Gets the index's collision count attribute loaded up by the retrieveStatsModule() method.
d o u b l e	getFindCount() Gets the index's find count attribute loaded up by the retrieveStatsModule() method.
d o u b l e	getFindDurationTime() Gets the index's find duration time attribute loaded up by the retrieveStatsModule() method.
d o u b l e	getFindFailureCount() Gets the index's failure count attribute loaded up by the retrieveStatsModule() method.

d o u b l e	<p>getFindResultCount() Gets the index's result count attribute loaded up by the retrieveStatsModule() method.</p>
S t r i n g	<p>getParentMapName() Gets the index's parent map name</p>
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . s t a t s . H a s h I n d e x S t a t s M o d u l e	<p>retrieveStatsModule() Gets the HashIndexStatsModule used to retrieve statistics associated with this index</p>

Method Detail

getParentMapName

[String](#) getParentMapName()

Gets the index's parent map name

Returns:

the name of the map which this index belongs to

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.HashIndexStatsModule **retrieveStatsModule()**

Gets the HashIndexStatsModule used to retrieve statistics associated with this index

Returns:

an HashIndexStatsModule for statistics associated with this index

See Also:

HashIndexStatsModule

getFindCount

double **getFindCount()**

Gets the index's find count attribute loaded up by the retrieveStatsModule() method.

Returns:

the find operation's invocation count for this index

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindCount(boolean copy)

getFindDurationTime

double **getFindDurationTime()**

Gets the index's find duration time attribute loaded up by the retrieveStatsModule() method.

Returns:

the find call's duration time for this index in milliseconds

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindDurationTime(boolean copy)

getFindResultCount

double **getFindResultCount()**

Gets the index's result count attribute loaded up by the retrieveStatsModule() method.

Returns:

the result count for this index and find operation

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindResultCount(boolean copy)

getFindFailureCount

double **getFindFailureCount()**

Gets the index's failure count attribute loaded up by the retrieveStatsModule() method.

Returns:

the failure count for this index and find operation

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindFailureCount(boolean copy)

getFindCollisionCount

double `getFindCollisionCount()`

Gets the index's collision count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the collision count for this index and find operation

See Also:

[retrieveStatsModule\(\)](#), HashIndexStatsModule.getFindCollisionCount(boolean copy)

getBatchUpdateCount

double `getBatchUpdateCount()`

Gets the index's batchupdate count attribute loaded up by the `retrieveStatsModule()` method.

Returns:

the doBatchUpdate method's invocation count for this index

See Also:

[retrieveStatsModule\(\)](#), MapIndexPlugin.doBatchUpdate(TxID txid, LogSequence sequence), HashIndexStatsModule.getBatchUpdateCount(boolean copy)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [IndexHelp](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management

Interface DynamicServerMBean

All Superinterfaces:

[ServerMBean](#)

```
public interface DynamicServerMBean
extends ServerMBean
```

This MBean interface allows a client process to access different attributes about a specific server process in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ObjectGridServer,name=<server>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

The following notifications are available:

`og.server.container.coregroup.membership.change`

Description All core group membership changes detected by the server's core group manager.

UserData: The number of members in the core group.

Message: The name of the core group.

Since: 6.1 FIX 3

com.ibm.websphere.objectgrid.log

Description: All log messages detected by the log notification filter. See the [setLogNotificationFilter\(String\)](#) attribute.

UserData A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The log record severity level.	8.6
LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The originating log record class name.	8.6
SourceMethodName	String	The originating log record method name.	8.6
SequenceNumber	Long	The log record sequence number	8.6
ThreadID	Integer	The originating log record thread id.	8.6
ThrownMessage	String	The thrown exception message or empty string, if	8.6

g not available.

Message: The log message.

Since: 8.6

com.ibm.websphere.objectgrid.ffdc

Description: All first-failure data captured by the grid server.

UserData A CompositeData with the following items:
:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
SourceID	String	The source code generating the exception.	8.6
ProbeID	String	The source code location associated with the exception captured.	8.6
ExceptionName	String	The exception captured.	8.6
Count	Integer	The current occurrence count for the specified exception.	8.6
DateOfFirstOccurrence	String	The exception's first occurrence time stamp.	8.6
Label	String	The label associated with the captured exception occurrence.	8.6

Message: A notification was generated on the server for a new exception.

Since: 8.6

Since:

WAS XD 6.1 FIX3, XC10

Field Summary	
s t a t i c S t r i n g	COUNT
s t a t i c S t r i n g	DATE_OF_FIRST_OCCURRENCE
s t a t i c S	EXCEPTION_NAME

t
r
i
n
g

s
t
a
t
i
c
S
t
r
i
n
g

[FFDC_INCIDENT_DATA](#)

s
t
a
t
i
c
S
t
r
i
n
g

[FFDC_NOTIFICATION](#)

s
t
a
t
i
c
S
t
r
i
n
g

[LABEL](#)

s
t
a
t
i
c
S
t
r
i
n
g

[LEVEL_NAME](#)

s
t
a
t
i
c
S
t
r
i
n
g

[LOG_MESSAGE_NOTIFICATION](#)

The constant identifying the notification type for log messages.

s
t
a
t
i
c
S
t
r
i
n
g

[LOG_RECORD_DATA](#)

integer

statistic
String

LOGGER_NAME

statistic
String

PROBE_ID

statistic
String

SEQUENCE_NUMBER

statistic
String

SERVER_COREGROUP_MEMBERSHIP_CHANGE

A constant identifying the notification type for all core group membership changes detected by this server.

statistic
String

SOURCE_CLASS_NAME

statistic
String

SOURCE_ID

g	
s t a t i c S t r i n g	SOURCE_METHOD_NAME
s t a t i c S t r i n g	THREAD_ID
s t a t i c S t r i n g	THROWN_MESSAGE

Method Summary

v o i d	checkFFDCNotification() Triggers a simulated exception to be captured as a first-failure data capture (FFDC) event (and subsequently broadcasted as a JMX notification), as a means to test and verify the monitoring being enabled on the server.
v o i d	checkLoggingNotification() Generates a set of log records with various severity levels, providing a simple means to test and verify when the JMX notification monitoring being enabled on the grid server.
i n t	getAvailableProcessors() Returns the number of available processors for the JVM hosting this server.
C o m p o s i t e D a t a	getEnvironmentInfo() Retrieve the environment information for the server (host name, WebSphere eXtreme Scale version, and additional information).
l o	getFreeMemory() Returns the available memory in bytes for the JVM hosting this server.

n g	
S t r i n g	getHostName() Returns the host name for this process.
S t r i n g	getLogNotificationFilter() Retrieves current regular expression filter being applied while screening log record messages before being broadcasted as JMX notifications by the grid server as a JMX Notification of type:
l o n g	getMaxMemory() Returns the maximum memory in bytes for the JVM hosting this server.
T a b u l a r D a t a	getPrimaryRevisions() Provides revisions for all primary shards in the container in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:
T a b u l a r D a t a	getRevisions() Provides revisions for all shards in the server in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:
b o o l e a n	getSafeToShutdown() Returns true if a replica exists for each primary hosted on this server.
S t r i n g	getStatsSpec() Retrieve the current statistics specification for the server.
l o n g	getTotalMemory() Returns the total memory in bytes for the JVM hosting this server.
S t r i n g	getTraceSpec() Retrieve the current trace specification for the server.

S t r i n g	getZoneName() Returns the zone name for this process
v o i d	setLogNotificationFilter(String regexFilter) Sets the regular expression filter to be applied in screening log record messages before being broadcasted by the grid server as a JMX Notification of type:
v o i d	setStatsSpec(String statsSpec) Set the statistics specification for the server.
v o i d	setTraceSpec(String traceSpec) Set the trace specification for the server.

Methods inherited from interface
com.ibm.websphere.objectgrid.management.[ServerMBean](#)
[getServerName](#), [modifyServerTraceSpec](#), [stopServer](#)

Field Detail

SERVER_COREGROUP_MEMBERSHIP_CHANGE

static final [String](#) SERVER_COREGROUP_MEMBERSHIP_CHANGE

A constant identifying the notification type for all core group membership changes detected by this server.

Since:

6.1 FIX 3

See Also:

[Constant Field Values](#)

LOG_MESSAGE_NOTIFICATION

static final [String](#) LOG_MESSAGE_NOTIFICATION

The constant identifying the notification type for log messages.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

FFDC_NOTIFICATION

static final [String](#) FFDC_NOTIFICATION

See Also:

[Constant Field Values](#)

LOG_RECORD_DATA

static final [String](#) LOG_RECORD_DATA

See Also:

[Constant Field Values](#)

THROWN_MESSAGE

static final [String](#) THROWN_MESSAGE

See Also:

[Constant Field Values](#)

THREAD_ID

static final [String](#) THREAD_ID

See Also:

[Constant Field Values](#)

SEQUENCE_NUMBER

static final [String](#) SEQUENCE_NUMBER

See Also:

[Constant Field Values](#)

SOURCE_METHOD_NAME

static final [String](#) SOURCE_METHOD_NAME

See Also:

[Constant Field Values](#)

SOURCE_CLASS_NAME

static final [String](#) SOURCE_CLASS_NAME

See Also:

[Constant Field Values](#)

LOGGER_NAME

static final [String](#) LOGGER_NAME

See Also:

[Constant Field Values](#)

LEVEL_NAME

static final [String](#) LEVEL_NAME

See Also:

[Constant Field Values](#)

FFDC_INCIDENT_DATA

static final [String](#) FFDC_INCIDENT_DATA

See Also:

[Constant Field Values](#)

SOURCE_ID

static final [String](#) SOURCE_ID

See Also:

[Constant Field Values](#)

PROBE_ID

static final [String](#) PROBE_ID

See Also:

[Constant Field Values](#)

LABEL

static final [String](#) LABEL

See Also:

[Constant Field Values](#)

DATE_OF_FIRST_OCCURRENCE

static final [String](#) DATE_OF_FIRST_OCCURRENCE

See Also:

[Constant Field Values](#)

COUNT

static final [String](#) COUNT

See Also:

[Constant Field Values](#)

EXCEPTION_NAME

static final [String](#) EXCEPTION_NAME

See Also:

[Constant Field Values](#)

Method Detail

getAvailableProcessors

int `getAvailableProcessors()`

Returns the number of available processors for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.availableProcessors\(\)](#)

getFreeMemory

long `getFreeMemory()`

Returns the available memory in bytes for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.freeMemory\(\)](#)

getMaxMemory

long `getMaxMemory()`

Returns the maximum memory in bytes for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.maxMemory\(\)](#)

getTotalMemory

long `getTotalMemory()`

Returns the total memory in bytes for the JVM hosting this server.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[Runtime.totalMemory\(\)](#)

getHostName

[String](#) `getHostName()`

Returns the host name for this process.

Returns:

The answer from the Runtime call on the JVM hosting this server.

See Also:

[InetAddress.getHostName\(\)](#)

getZoneName

[String](#) `getZoneName()`

Returns the zone name for this process

Returns:

the zone name that was included in the properties used to start the server or DefaultZone if no zone name was used

getSafeToShutdown

boolean `getSafeToShutdown()`

Returns true if a replica exists for each primary hosted on this server. Returns false if the server has the only copy of data.

Returns:

If server is safe to shutdown.

getStatsSpec

[String](#) `getStatsSpec()`

Retrieve the current statistics specification for the server.

Returns:

a string representation of the statistics specification.

Since:

7.1

See Also:

StatsSpec

setStatsSpec

void `setStatsSpec(String statsSpec)`

Set the statistics specification for the server.

Parameters:

statsSpec - the statistics specification string.

Since:

7.1

See Also:

StatsSpec

getTraceSpec

[String](#) `getTraceSpec()`

Retrieve the current trace specification for the server.

Returns:

the trace specification string.

Since:

7.1

setTraceSpec

void `setTraceSpec(String traceSpec)`

Set the trace specification for the server.

Parameters:

traceSpec - the statistics specification string.

Since:

7.1

See Also:

[ObjectGridManager.setTraceSpecification\(String\)](#)

getEnvironmentInfo

[CompositeData](#) `getEnvironmentInfo()`

Retrieve the environment information for the server (host name, WebSphere eXtreme Scale version, and additional information). The CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
JMXServicePort	String	JMX Service Port
WASServerName	String	WebSphere Application Server Full Server Name
JVMVersion	String	JAVA Version
JMXConnectorPort	String	JMX Connector Port
IPAddress	String	IP Address
JavaVM	String	JVM Version
WASInstallRoot	String	WebSphere Application Server Product Directory
OSGiFrameworkVersion	String	OSGi Version
ClientPort	String	Client Port
HostName	String	Host name
Timestamp	String	Time Stamp from Server
WASBaseVersion	String	IBM WebSphere Application Server Version
OSName	String	Operating System
XSInstallRoot	String	WebSphere eXtreme Scale Product Directory
OSArch	String	OS Architecture
PeerPort	String	Peer Port
ServerType	String	Server Type
HAManagerPort	String	HAManager Port
JVMInstallPath	String	JAVA Directory
XC10Model	String	Machine Type and Model
JavaRuntimeInfo	String	JVM Runtime Version
xioContainerTCPNonSecure	String	XIO TCP/IP Port
JMXServiceURL	String	JMX Service Port
listenerPort	String	Listener Port (ORB)
ServerName	String	Server Name
WASXDVersion	String	IBM WebSphere Application Server - XD Version
WASExpressVersion	String	IBM WebSphere Application Server - ND Version
JavaBitMode	String	JAVA Bit Mode
XSVersion	String	WebSphere eXtreme Scale Version
JVMVendor	String	JAVA Vendor
OSVersion	String	Operating System Version
xioContainerTCPSecure	String	XIO TCP/IP SSL Port
PID	String	Process ID
WASNDVersion	String	IBM WebSphere Application Server - ND Version
ORB Version	String	ORB Version

Returns:

CompositeData containing the environment information

Since:
8.5, XC10 2.5

setLogNotificationFilter

void **setLogNotificationFilter**([String](#) regexFilter)

Sets the regular expression filter to be applied in screening log record messages before being broadcasted by the grid server as a JMX Notification of type:

com.ibm.websphere.objectgrid.log

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The logging level.	8.6
LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The class name.	8.6
SourceMethodName	String	The method name.	8.6
SequenceNumber	Long	The sequence number	8.6
ThreadID	Integer	The thread id.	8.6
ThrownMessage	String	The thrown exception message or empty string, if not available.	8.6

Message: The log message.

Since: 8.6

Parameters:

regexFilter - The regular expression filter to applied in screening log record messages.

Since:
8.6, XC10 2.5

getLogNotificationFilter

[String](#) **getLogNotificationFilter**()

Retrieves current regular expression filter being applied while screening log record messages before being broadcasted as JMX notifications by the grid server as a JMX Notification of type:

com.ibm.websphere.objectgrid.log

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The logging level.	8.6

LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The class name.	8.6
SourceMethodName	String	The method name.	8.6
SequenceNumber	Long	The sequence number	8.6
ThreadID	Integer	The thread id.	8.6
ThrownMessage	String	The thrown exception message or empty string, if not available.	8.6

Message: The log message.

Since: 8.6

Returns:

Current regular expression filter being applied in screening log record messages.

Since:

8.6, XC10 2.5

checkFFDCNotification

void **checkFFDCNotification()**

Triggers a simulated exception to be captured as a first-failure data capture (FFDC) event (and subsequently broadcasted as a JMX notification), as a means to test and verify the monitoring being enabled on the server. The JMX notification generated and broadcasted has the following type and content:

`com.ibm.websphere.objectgrid.ffdc`

Description: All first-failure data captured by the grid server.

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
SourceID	String	The source code generating the exception.	8.6
ProbeID	String	The source code location associated with the exception captured.	8.6
ExceptionName	String	The exception captured.	8.6
Count	Integer	The current occurrence count for the specified exception.	8.6
DateOfFirstOccurrence	String	The exception's first occurrence time stamp.	8.6
Label	String	The label associated with the captured exception occurrence.	8.6

Message: A simulated first-failure data capture (FFDC) exception notification was generated by the server.

Since: 8.6

Since:

8.6, XC10 2.5

checkLoggingNotification

void **checkLoggingNotification()**

Generates a set of log records with various severity levels, providing a simple means to test and verify when the JMX notification monitoring being enabled on the grid server. Each of the JMX notifications generated and broadcasted has the following type and content:

com.ibm.websphere.objectgrid.log

UserData: A CompositeData with the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LevelName	String	The logging level.	8.6
LoggerName	String	The name of the logger.	8.6
SourceClassName	String	The class name.	8.6
SourceMethodName	String	The method name.	8.6
SequenceNumber	Long	The sequence number	8.6
ThreadID	Integer	The thread id.	8.6
ThrownMessage	String	A simulated log info/warning/error notification was generated by the server	8.6

Message: A simulated log info/warning/error notification was generated by the server.

Since: 8.6

Since:
8.6, XC10 2.5

getRevisions

[TabularData](#) **getRevisions()**

Provides revisions for all shards in the server in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Grid	String	The object grid name.	8.6
MapSet	String	The name of the mapSet.	8.6
PartitionNumber	Integer	The number of a given partition.	8.6
ContainerName	String	Name of the container.	8.6
ShardType	String	Type of the shard.	8.6
RevisionState	TabularData	The lifetimeId / revisionNumber info for a given partition.	8.6

The RevisionState TabularData's CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LifetimeId	Long	The LifetimeId for the shard	8.6
Revision	Long	The revision number	8.6

Returns:

A TabularData object with the revision information.

Since:

8.6, XC10 2.5

getPrimaryRevisions

[TabularData](#) `getPrimaryRevisions()`

Provides revisions for all primary shards in the container in the form of a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Grid	String	The object grid name.	8.6
MapSet	String	The name of the mapSet.	8.6
PartitionNumber	Integer	The number of a given partition.	8.6
ContainerName	String	Name of the container.	8.6
ShardType	String	Type of the shard.	8.6
RevisionState	TabularData	The lifetimeId / revisionNumber info for a given partition.	8.6

The RevisionState TabularData's CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
LifetimeId	Long	The LifetimeId for the shard	8.6
Revision	Long	The revision number	8.6

Returns:

A TabularData object with the revision information.

Since:

8.6, XC10 2.5

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL : FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid.management

Interface ContainerMBean

public interface **ContainerMBean**

This MBean interface allows a client process to perform operations on and get status from an ObjectGrid container running in a dynamic environment. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=ObjectGridContainer,name=<server>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary	
s t a t i c S t r i n g	<p>INVALID_PARTITION INVALID_PARTITION indicates that no partition was found for the requested shard.</p>
s t a t i c S t r i n g	<p>MAPSET_UNSUPPORTED_ON_CONTAINER MAPSET_UNSUPPORTED_ON_CONTAINER indicates that an attempt was made to reserve a shard from a map set that is not supported on this container.</p>
s t a t i c S t r i n g	<p>QUIESCE_COMPLETE QUIESCE_COMPLETE is the MBean notification type for a completed quiesce.</p>
s	

statistic

RELEASE_SUCCESSFUL

RELEASE_SUCCESSFUL indicates that the attempt to release the shard was successful.

statistic

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy.

statistic

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT indicates that the attempt to reserve the shard was processed successfully.

statistic

RESERVATION_SUCCESSFUL

RESERVATION_SUCCESSFUL indicates that the attempt to reserve the shard was successful.

statistic

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy.

statistic

SHARD_ALREADY_RESERVED

SHARD_ALREADY_RESERVED indicates that the shard is already reserved elsewhere and cannot be reserved on the specified container.

stat

SHARD_NOT_RESERVED_ON_CONTAINER
 SHARD_NOT_RESERVED_ON_CONTAINER indicates that the attempt to release the shard from the requesting container failed because the specified shard was not found to be reserved by the requesting container.

Method Summary

i n t	<p>getActivatedShardCount() Retrieve the total number of shards that have been activated for the life of this ObjectGrid container.</p>
i n t	<p>getActiveShardCount() Retrieve the number of active shards hosted in this ObjectGrid container.</p>
S t r i n g	<p>getContainerName() Retrieve the name of the container.</p>
i n t	<p>getDeactivatedShardCount() Retrieve the total number of shards that have been deactivated for the life of this ObjectGrid container.</p>
S t r i n g	<p>getDomainName() Retrieve the name of the catalog server grouping administering this container.</p>
S t r i n g	<p>getStatus() Retrieve the status information for the shards in this container.</p>
S t r i n g	<p>getZoneName() Retrieve the name of the zone grouping that this container belongs to.</p>
i n t	<p>quiesceContainer(Boolean inQuiesce) Prepare the container for a potential shutdown by moving replica shards, verifying that primaries have required sync replicas and preventing the placement of new shards.</p>
S t r i n g	<p>release(String objectGridName, String mapSetName, String partitionName) Release a shard that has been previously reserved by this container.</p>
S t r i n g	<p>reserve(String objectGridName, String mapSetName, String partitionName, String shardType) Reserve a specific shard on this container.</p>

S t r i n g	retrieveStatus (String objectGridName, String mapSetName) Retrieve the status information for the shards in this container, filtered by ObjectGrid and/or mapset.
v o i d	teardown () Tears down and stops the container in a way to allow partitions to be moved to new locations.
v o i d	terminate () Terminates a container without coordinating partition movement, partitions will failover.

Field Detail

QUIESCE_COMPLETE

static final [String](#) QUIESCE_COMPLETE

QUIESCE_COMPLETE is the MBean notification type for a completed quiesce.

See Also:

[quiesceContainer\(Boolean\)](#), [Constant Field Values](#)

RESERVATION_SUCCESSFUL

static final [String](#) RESERVATION_SUCCESSFUL

RESERVATION_SUCCESSFUL indicates that the attempt to reserve the shard was successful. The shard is reserved by the requesting container.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT

static final [String](#) RESERVATION_PRIOR_TO_INITIAL_PLACEMENT

RESERVATION_PRIOR_TO_INITIAL_PLACEMENT indicates that the attempt to reserve the shard was processed successfully. However, since initial placement has not yet occurred, the reserved shard is not immediately moved to the requesting container. The shard will be placed on the container when initial placement is triggered.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

SHARD_ALREADY_RESERVED

static final [String](#) SHARD_ALREADY_RESERVED

SHARD_ALREADY_RESERVED indicates that the shard is already reserved elsewhere

and cannot be reserved on the specified container. The shard must be released from the owning container before it can be reserved again.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

INVALID_PARTITION

static final [String](#) INVALID_PARTITION

INVALID_PARTITION indicates that no partition was found for the requested shard.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER

static final [String](#) RESERVE_UNSUPPORTED_WITH_PER_CONTAINER

RESERVE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy. Shard reservation is not supported with this placement strategy.

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

RELEASE_SUCCESSFUL

static final [String](#) RELEASE_SUCCESSFUL

RELEASE_SUCCESSFUL indicates that the attempt to release the shard was successful. The shard is no longer reserved by this container. The shard is free to migrate, but it is not forced to migrate.

Since:

7.0.0.0 FIX1

See Also:

[release\(String, String, String\)](#), [Constant Field Values](#)

SHARD_NOT_RESERVED_ON_CONTAINER

static final [String](#) SHARD_NOT_RESERVED_ON_CONTAINER

SHARD_NOT_RESERVED_ON_CONTAINER indicates that the attempt to release the shard from the requesting container failed because the specified shard was not found to be reserved by the requesting container. Only the container owning the reservation may release a shard.

Since:

7.0.0.0 FIX1

See Also:

[release\(String, String, String\)](#), [Constant Field Values](#)

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER

static final [String](#) RELEASE_UNSUPPORTED_WITH_PER_CONTAINER

RELEASE_UNSUPPORTED_WITH_PER_CONTAINER indicates that the shard is part of a map set using the PER_CONTAINER placement strategy. Shard release is not supported with this placement strategy.

Since:

7.0.0.0 FIX1

See Also:

[release\(String, String, String\)](#), [Constant Field Values](#)

MAPSET_UNSUPPORTED_ON_CONTAINER

static final [String](#) MAPSET_UNSUPPORTED_ON_CONTAINER

MAPSET_UNSUPPORTED_ON_CONTAINER indicates that an attempt was made to reserve a shard from a map set that is not supported on this container. Only map sets that were included in the deployment policy at container initialization are supported to run on this container.

Since:

7.1

See Also:

[reserve\(String, String, String, String\)](#), [Constant Field Values](#)

Method Detail

teardown

void `teardown()`

Tears down and stops the container in a way to allow partitions to be moved to new locations.

terminate

void `terminate()`

Terminates a container without coordinating partition movement, partitions will failover.

getActiveShardCount

int `getActiveShardCount()`

Retrieve the number of active shards hosted in this ObjectGrid container.

Returns:

The current number of active shards.

getActivatedShardCount

int `getActivatedShardCount()`

Retrieve the total number of shards that have been activated for the life of this

ObjectGrid container.

Returns:

The number of activated shards.

getDeactivatedShardCount

int `getDeactivatedShardCount()`

Retrieve the total number of shards that have been deactivated for the life of this ObjectGrid container.

Returns:

The number of deactivated shards

getDomainName

[String](#) `getDomainName()`

Retrieve the name of the catalog server grouping administering this container.

Returns:

The domain name.

getZoneName

[String](#) `getZoneName()`

Retrieve the name of the zone grouping that this container belongs to.

Returns:

The name of the zone.

quiesceContainer

int `quiesceContainer(Boolean inQuiesce)`

Prepare the container for a potential shutdown by moving replica shards, verifying that primaries have required sync replicas and preventing the placement of new shards.

Parameters:

`inQuiesce` - Initiate quiesce mode (true) or cancel quiesce mode (false)

Returns:

The number of replicas moved off of the ObjectGrid container

See Also:

[QUIESCE_COMPLETE](#)

getStatus

[String](#) `getStatus()`

Retrieve the status information for the shards in this container.

Returns:

The status information for the shards in this container.

retrieveStatus

```
String retrieveStatus(String objectGridName,  
                    String mapSetName)
```

Retrieve the status information for the shards in this container, filtered by ObjectGrid and/or mapset. For example, calling `retrieveStatus` with "og1" and "ms1" as parameters will return the partition status for those partitions in ObjectGrid og1 and mapset ms1. Passing in an empty string ("") objectGridName or mapSetName will return all of the partitions, since the empty string acts as a wildcard. Passing in the empty string for both parameters will return the same status as calling `getStatus()`.

Parameters:

objectGridName - The name of the ObjectGrid for which the status is requested.
mapSetName - The name of the mapset within the ObjectGrid for which the status is requested.

Returns:

The status information for the shards in this container.

reserve

```
String reserve(String objectGridName,  
             String mapSetName,  
             String partitionName,  
             String shardType)
```

Reserve a specific shard on this container. Calling this method will cause the requested shard to move to this container. The shard can be moved to this container only if it is not reserved elsewhere. Calling this method prior to initial placement will pre-reserve the shard so that it will be placed onto this container when initial placement occurs. If non-reserved shard for the same partition is on this container prior to reservation, the non-reserved shard will be moved off the container upon reservation. A reserved shard will not be moved off of this container until it is released or the container is stopped.

Parameters:

objectGridName - the ObjectGrid containing the shard
mapSetName - the map set containing the shard
partitionName - the partition containing the shard
shardType - the type of shard. Currently, only primary shards can be reserved:
[ShardMBean.TYPE_PRIMARY](#)

Returns:

the return code indicating the result of the reserve request

Throws:

[IllegalArgumentException](#) - if any of the arguments are null or the empty String. Also thrown if shardType is not [ShardMBean.TYPE_PRIMARY](#)

Since:

7.0.0.0 FIX1

See Also:

[ShardMBean.TYPE_PRIMARY](#), [release\(String, String, String\)](#), [RESERVATION_SUCCESSFUL](#), [RESERVATION_PRIOR_TO_INITIAL_PLACEMENT](#), [SHARD_ALREADY_RESERVED](#), [INVALID_PARTITION](#), [RESERVE_UNSUPPORTED_WITH_PER_CONTAINER](#)

release

```
String release(String objectGridName,  
             String mapSetName,  
             String partitionName)
```

Release a shard that has been previously reserved by this container. This container can only release shards that it has reserved. Releasing the shard does not guarantee the

shard will be moved. The shard may remain on this container. However, it will not be explicitly bound to this container. Releasing a shard allows the shard to move freely to other containers or to be reserved by another container.

Parameters:

objectGridName - the ObjectGrid containing the shard
mapSetName - the map set containing the shard
partitionName - the partition containing the shard

Returns:

the return code indicating the result of the release request

Throws:

[IllegalArgumentExcpion](#) - if any of the arguments are null or the empty String

Since:

7.0.0.0 FIX1

See Also:

[reserve\(String, String, String, String\)](#), [RELEASE_SUCCESSFUL](#),
[SHARD_NOT_RESERVED_ON_CONTAINER](#), [RELEASE_UNSUPPORTED_WITH_PER_CONTAINER](#)

getContainerName

[String](#) getContainerName()

Retrieve the name of the container. The container name is based on the server name and includes a suffix which uniquely identifies the container within the server.

Returns:

the name of the container

Since:

7.1

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Class	Next Class	Frames	No Frames	All			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid.management

Interface CatalogServiceManagementMBean

public interface **CatalogServiceManagementMBean**

This MBean interface allows user to manipulate the behaviors of heartbeat and leader manager and other catalog service specific actions. The object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=CatalogService
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

7.1, XC10

Field Summary	
s t a t i c S t r i n g	<p>COREGROUP_MEMBERSHIP_CHANGE Constant representing a core group membership change notification.</p>
s t a t i c i n t	<p>HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE Constant representing a heartbeat frequency level at an aggressive rate.</p>
s t a t i c i n t	<p>HEARTBEAT_FREQUENCY_LEVEL_RELAXED Constant representing a heartbeat frequency level at relaxed rate.</p>
s t a t i c	<p>HEARTBEAT_FREQUENCY_LEVEL_TYPICAL Constant representing a heartbeat frequency level at a typical rate.</p>

i n t	
s t a t i c S t r i n g	<u>MANAGEMENT_CONCENTRATOR_STATUS_STARTED</u>
s t a t i c S t r i n g	<u>MANAGEMENT_CONCENTRATOR_STATUS_STOPPED</u>
s t a t i c S t r i n g	<u>ORB</u> Constant representing the ORB or Object Request Broker transport
s t a t i c S t r i n g	<u>SERVER_EVENT_STARTED</u> Constant representing an eXtreme Scale server start notification.
s t a t i c S t r i n g	<u>SERVER_EVENT_STOPPED</u> Constant representing an eXtreme Scale server stop notification.
s t a t i c S t r i n g	<u>XIO</u> Constant representing the eXtremeIO transport

Method Summary

I
a
b
u
l
e
r
D
e
t
a
i
l**[getContainerReplicationState\(\)](#)**

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross containers within one domain.

I
a
b
u
l
e
r
D
e
t
a
i
l**[getDomainReplicationState\(\)](#)**

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross domains.

i
n
t**[getHeartBeatFrequencyLevel\(\)](#)**

Retrieves the heartbeat frequency level.

S
t
r
i
n
g**[getManagementConcentratorStatus\(\)](#)**

Returns the String representation of the management concentrator status.

i
n
t**[getNumberOfServers\(\)](#)**

Retrieves the number eXtreme Scale servers that are currently registered with the catalog service.

C
o
m
p
o
s
i
t
e
D
e
t
a
i
l**[getServers\(\)](#)**

Retrieves a CompositeData of each eXtreme Scale server that is currently registered with the catalog service.

S
t
r
i
n
g**[getTransport\(\)](#)**

Returns the transport used by the catalog service domain.

b
o
o
l
e
a
n**[isPrimary\(\)](#)**

Provides indication if the catalog is primary.

v o i d	LogMessage (String level, String message) Provides support for logging user messages from processes outside the catalog and/or container servers.
v o i d	startManagementConcentrator () Starts the Management Concentrator.
v o i d	stopManagementConcentrator () Stops the Management Concentrator.

Field Detail

MANAGEMENT_CONCENTRATOR_STATUS_STARTED

static final [String](#) MANAGEMENT_CONCENTRATOR_STATUS_STARTED

See Also:

[Constant Field Values](#)

MANAGEMENT_CONCENTRATOR_STATUS_STOPPED

static final [String](#) MANAGEMENT_CONCENTRATOR_STATUS_STOPPED

See Also:

[Constant Field Values](#)

COREGROUP_MEMBERSHIP_CHANGE

static final [String](#) COREGROUP_MEMBERSHIP_CHANGE

Constant representing a core group membership change notification. The user data associated with this notification is a CompositeData.

The CompositeData includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

MemberName	String	The name of the server that is included in the core group.
------------	--------	--

See Also:

[Constant Field Values](#)

SERVER_EVENT_STARTED

static final [String](#) SERVER_EVENT_STARTED

Constant representing an eXtreme Scale server start notification.

The UserData argument of the Notification includes a TabularData that includes information for each of the servers. Each CompositeData (row in the TabularData) contains the following items:

<u>Description</u>

<u>Item Name</u>	<u>Type</u>	
HAPort	String	The port number of the high availability manager.
Host	String	The host/ip address of the server.
JMXServiceURL	String	The JMX service url used to access the server.
ServerName	String	The name of the server.
ZoneName	String	The name of the zone that the server belongs.

See Also:

[Constant Field Values](#)

SERVER_EVENT_STOPPED

```
static final String SERVER_EVENT_STOPPED
```

Constant representing an eXtreme Scale server stop notification.

The UserData argument of the Notification includes a TabularData instance where each CompositeData contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

ServerName	String	The name of the server.
------------	--------	-------------------------

See Also:

[Constant Field Values](#)

HEARTBEAT_FREQUENCY_LEVEL_TYPICAL

```
static final int HEARTBEAT_FREQUENCY_LEVEL_TYPICAL
```

Constant representing a heartbeat frequency level at a typical rate.

A typical heartbeat frequency allows reasonable failover detection and resource utilization. This value is the default.

See Also:

[Constant Field Values](#)

HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE

```
static final int HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE
```

Constant representing a heartbeat frequency level at an aggressive rate.

An increased heartbeat frequency allows failures to be detected more quickly, but can also use additional CPU and network resources. This level is more sensitive to missing heartbeats when the server is stressed.

See Also:

[Constant Field Values](#)

HEARTBEAT_FREQUENCY_LEVEL_RELAXED

```
static final int HEARTBEAT_FREQUENCY_LEVEL_RELAXED
```

Constant representing a heartbeat frequency level at relaxed rate.

A decreased heartbeat frequency increases the time to detect failures, but also decreases

CPU and network utilization.

See Also:

[Constant Field Values](#)

ORB

static final [String](#) ORB

Constant representing the ORB or Object Request Broker transport

See Also:

[Constant Field Values](#)

XIO

static final [String](#) XIO

Constant representing the eXtremeIO transport

See Also:

[Constant Field Values](#)

Method Detail

getHeartBeatFrequencyLevel

int `getHeartBeatFrequencyLevel()`

Retrieves the heartbeat frequency level.

Valid values include:

- [HEARTBEAT_FREQUENCY_LEVEL_TYPICAL](#)
- [HEARTBEAT_FREQUENCY_LEVEL_RELAXED](#)
- [HEARTBEAT_FREQUENCY_LEVEL_AGGRESSIVE](#)

Returns:

the heartbeat frequency level: -1, 0 or 1 as defined by the constants that begin with name HEARTBEAT_FREQUENCY_LEVEL.

getServers

[CompositeData](#) `getServers()`

Retrieves a CompositeData of each eXtreme Scale server that is currently registered with the catalog service.

The CompositeData includes the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>
-------------------------	--------------------	---------------------------

serverName	String	The name of the server that is registered with the catalog service.
------------	--------	---

Returns:

the CompositeData representing the currently registered eXtreme Scale servers.

getNumberOfServers

int **getNumberOfServers()**

Retrieves the number eXtreme Scale servers that are currently registered with the catalog service.

Returns:

the number of registered eXtreme Scale servers.

logMessage

void **logMessage**([String](#) level,
[String](#) message)

Provides support for logging user messages from processes outside the catalog and/or container servers. Example: XC10 surfaced SNMP trap messages can be flowed from the SNMP agent which throws traps in the console server (sMash) process, not typically an XS catalog/container server.

Parameters:

level - name describing the severity of the event which is compatible with `java.util.logging.Level.parse(String name)` where name may be either level name (ex. "SEVERE") or an integer value (ex. "1000") - @see [Level.parse\(String\)](#)
message - for the end user (already sNLS rendered)

Since:

8.6, XC10 2.5

isPrimary

boolean **isPrimary()**

Provides indication if the catalog is primary.

Returns:

true for primary catalog.

Since:

8.6, XC10 2.5

getContainerReplicationState

[TabularData](#) **getContainerReplicationState()**

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross containers within one domain. For a given container, outstanding out-bound revisions need to be replicated from primary shards located in this container into replicas located in other containers. In similar way, outstanding in-bound revisions need to be replicated from primary shards located in other containers into corresponding replicas located in this container. This operation can be used to check the differences in data revisions between containers within one domain.

The result is a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Container	String	The container ID.	8.6
OutboundRevisions	Long	Number of out-bound revisions for container.	8.6
InboundRevisions	Long	Number of in-bound revisions for container.	8.6

Returns:

A TabularData object with the container replication state information.

Since:

8.6, XC10 2.5

getDomainReplicationState

[TabularData](#) `getDomainReplicationState()`

Provides the numbers of outstanding in-bound and out-bound revisions which need to be replicated cross domains. Outstanding out-bound revisions need to be replicated from local primary shards into corresponding remote primary shards. Outstanding in-bound revisions need to be replicated from remote primary shards into corresponding local primary shards. This operation can be used to check the differences in data revisions between different domains linked by MMR replication.

The result is a TabularData object, where each CompositeData (row in the TabularData) contains the following items:

<u>Item Name</u>	<u>Type</u>	<u>Description</u>	<u>Since</u>
Domain	String	The domain name.	8.6
Container	String	The container ID.	8.6
OutboundRevisions	Long	Number of out-bound revisions for container.	8.6
InboundRevisions	Long	Number of in-bound revisions for container.	8.6

Returns:

A TabularData object with the domain replication state information.

Since:

8.6, XC10 2.5

getTransport

[String](#) `getTransport()`

Returns the transport used by the catalog service domain.

Returns:

String containing the transport type

Since:

8.6, XC10 2.5

See Also:

[ORB](#), [XIO](#)

startManagementConcentrator

`void startManagementConcentrator()`

Starts the Management Concentrator. The catalog server will now start listening for log messages.

Since:

8.6.0.2, XC10 2.5

stopManagementConcentrator

`void stopManagementConcentrator()`

Stops the Management Concentrator. The catalog server will no longer listen for log

messages. Stopping the Management Concentrator will also stop the Message Center in the web monitoring console.

Since:

8.6.0.2, XC10 2.5

getManagementConcentratorStatus

[String](#) getManagementConcentratorStatus()

Returns the String representation of the management concentrator status. Status will be either CatalogServiceManagementMBean.MANAGEMENT_CONCENTRATOR_STATUS_STARTED or CatalogServiceManagementMBean.MANAGEMENT_CONCENTRATOR_STATUS_STOPPED.

Returns:

String containing the status

Since:

8.6.0.2, XC10 2.5

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.management

Interface AgentManagerMBean

public interface **AgentManagerMBean**

This MBean interface allows a client process to access different attributes and statistical data about a specific Agent on a server process. The Agent Manager MBean is scoped at the map level and therefore can access statistical data for every agent run against the specified map. In a dynamic ObjectGrid environment, the object name pattern for this MBean is:

```
com.ibm.websphere.objectgrid:type=AgentManager,name=Agent-<map>,partition=<partition id>,objectgrid=<objectgrid>,host=<host>,ogServerName=<server>
```

If ObjectGrid is running in a WebSphere Application Server process, more key=value pairs may be added to the object name.

Since:

WAS XD 6.1.0.5, XC10

Method Summary	
d o u b l e	<p>getAgentInflationTime(<i>String</i> agentClassName) Gets the specified agent's inflation time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getAgentSerializationTime(<i>String</i> agentClassName) Gets the specified agent's serialization time attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getFailureCount(<i>String</i> agentClassName) Gets the specified agent's failure count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getInvocationCount(<i>String</i> agentClassName) Gets the specified agent's invocation count attribute loaded up by the retrieveStatsModule() method.</p>
d o u b l e	<p>getPartitionCount(<i>String</i> agentClassName) Gets the specified agent's partition count attribute loaded up by the retrieveStatsModule() method.</p>

d
o
u
b
l
e

[getReduceTime](#)(String agentClassName)

Gets the specified agent's total reduce time attribute loaded up by the retrieveStatsModule() method.

d
o
u
b
l
e

[getResultInflationTime](#)(String agentClassName)

Gets the specified agent's result inflation time attribute loaded up by the retrieveStatsModule() method.

d
o
u
b
l
e

[getResultSerializationTime](#)(String agentClassName)

Gets the specified agent's result serialization time attribute loaded up by the retrieveStatsModule() method.

d
o
u
b
l
e

[getTotalDurationTime](#)(String agentClassName)

Gets the specified agent's total run time attribute loaded up by the retrieveStatsModule() method.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
s
t
a
t
s
.
A
g
e
n
t
S
t
a
t
s
M
o

[retrieveStatsModule](#)(String agentClassName)

Gets the AgentStatsModule used to retrieve statistics associated with the specified agent class

Method Detail

getReduceTime

double **getReduceTime**([String](#) agentClassName)

Gets the specified agent's total reduce time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the reduce time for this agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `AgentStatsModule.getReduceTime(boolean copy)`

getTotalDurationTime

double **getTotalDurationTime**([String](#) agentClassName)

Gets the specified agent's total run time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the total run time for this agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `AgentStatsModule.getTotalDurationTime(boolean copy)`

getAgentSerializationTime

double **getAgentSerializationTime**([String](#) agentClassName)

Gets the specified agent's serialization time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to serialize the agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), `AgentStatsModule.getAgentSerializationTime(boolean copy)`

getAgentInflationTime

double **getAgentInflationTime**([String](#) agentClassName)

Gets the specified agent's inflation time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to inflate the agent in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getAgentInflationTime(boolean copy)

getResultInflationTime

double **getResultInflationTime**([String](#) agentClassName)

Gets the specified agent's result inflation time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to inflate the agent results for a given partition in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getResultInflationTime(boolean copy)

getResultSerializationTime

double **getResultSerializationTime**([String](#) agentClassName)

Gets the specified agent's result serialization time attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the time it takes to serialize the agent results for a given partition in milliseconds

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getResultSerializationTime(boolean copy)

getPartitionCount

double **getPartitionCount**([String](#) agentClassName)

Gets the specified agent's partition count attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the number of partitions this agent is sent to

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getPartitionCount(boolean copy)

getFailureCount

double **getFailureCount**([String](#) agentClassName)

Gets the specified agent's failure count attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the failure count for the specified agent

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getFailureCount(boolean copy)

getInvocationCount

double **getInvocationCount**([String](#) agentClassName)

Gets the specified agent's invocation count attribute loaded up by the `retrieveStatsModule()` method.

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

the invocation count for the specified agent

See Also:

[retrieveStatsModule\(String\)](#), AgentStatsModule.getInvocationCount(boolean copy)

retrieveStatsModule

com.ibm.websphere.objectgrid.stats.AgentStatsModule **retrieveStatsModule**([String](#) agentClassName)

Gets the AgentStatsModule used to retrieve statistics associated with the specified agent class

Parameters:

agentClassName - The fully qualified class name of the agent

Returns:

an AgentStatsModule for statistics associated with the specified agent class

See Also:

AgentStatsModule

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

PREV CLASS [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package com.ibm.websphere.objectgrid.plugins

These are the interfaces for adding plugins to the Grid core framework.

See:

[Description](#)

Interface Summary	
BeanFactory	Implement this interface to allow bean factories like Spring or Google guice to be integrated.
CacheEntry	This interface represents a cache entry in an ObjectGrid map.
EvictionEventCallback	An instance of EvictionEventCallback is passed into the Evictor at initialization time.
Evictor	Data contained in a BackingMap are evicted when the map is full.
EvictorData	This interface is optionally used by an implementator of the Evictor interface.
LogElement	LogElements are the individual entries within a LogSequence.
LogSequence	LogSequence is the ordered list of changes performed against a given map for a given transaction.
LogSequenceFilter	This interface can be used to filter a LogSequence.
MapEventListener	This callback interface is implemented by the application when it wants to receive events about a Map such as the eviction of a map entry.
ObjectGridEventGroup	This is a set of single method interfaces for fine grained events delivered for an ObjectGrid.
ObjectGridEventGroup.ShardEvents	These events are fired when a shard is made a primary shard and when the shard is demoted from a primary.
ObjectGridEventGroup.ShardLifecycle	These events are fired when an ObjectGrid shard is initialized and destroyed.
ObjectGridEventGroup.TransactionEvents	These events are called every single transaction.
ObjectGridEventListener	This interface is used to create an implementation of an event listener for an ObjectGrid.
ReplicationMapListener	Deprecated. <i>The client replicated map function is deprecated in version 8.6.</i>
Transaction	Calling methods on a Session will send corresponding events to the

Callback	TransactionCallback.
TransactionCallback.BeforeCommit	The BeforeCommit optional mix-in interface for the TransactionCallback plugin interface allows plug-ins to be notified at the beginning of a Session.commit() .
TransactionCallback.BeforeCommit.TransactionContext	The TransactionContext identifies various information that's available to the beforeCommit() method.
ValueProxyInfo	This interface can be used by value objects to inform a BackingMap or Loader which attribute(s) have been dirtied.

Class Summary	
EvictorData.SpecialEvictorData	Special value class used for representing the key not being found in the BackingMap.
LogElement.Type	The Type class is used to represent a LogElement type.

Exception Summary	
CacheEntryException	This exception indicates an error occurred during a cache entry operation.
LoaderException	This exception is the base exception for any exceptions encountered by a Loader.
TransactionCallbackException	This exception is thrown when a method call to TransactionCallback fails.

Package **com.ibm.websphere.objectgrid.plugins** Description

These are the interfaces for adding plugins to the Grid core framework.

Overview

These plugins can be added into ObjectGrid in several ways such as xml configuration, programmatically adding, or using annotation.

Annotation based callbacks

ObjectGrid when running on Java 5 will start to use an annotated method callback system. This means that objects can be registered as callbacks or listeners. The methods on the object must be annotated as to be invoked for a certain event. Unannotated methods are not invoked. The name of the method is unimportant. The method arguments and return type must be the same as expected for the callback method.

Why?

Usually, callbacks are specified using an interface. This works well but results in a possible performance loss as all methods on the interface will be invoked by the ObjectGrid even

though the application is only interested in a single event. This wastes precious resources. Another issue is when we need to add a new event. Adding a new method to an existing interface breaks back wards compatibility. We can make a new interface extending the old one with the new methods but this is also undesirable as soon there are many interfaces in the hierarchy as new events are added. The annotation system allows the application to only mark methods to be called avoiding the first problem and if new event types are added they have no impact on existing callback objects. Newer applications can add a method and annotate it with the new event annotation to receive the event.

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV PACKAGE](#) [NEXT PACKAGE](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ValueProxyInfo

```
public interface ValueProxyInfo
```

This interface can be used by value objects to inform a BackingMap or Loader which attribute(s) have been dirtied. This mechanism allows the BackingMap and Loader to interrogate the set of changed attributes in the value object instead of just assuming the whole value object has been updated. For this to be useful, the application must only use the getter and setter methods defined for the value object's interface.

Since:

WAS XD 6.0, XC10

See Also:

[CopyMode.COPY_ON_WRITE](#), [BackingMap.setCopyMode\(CopyMode, Class\)](#), [Loader.batchUpdate\(TxID, LogSequence\)](#)

Method Summary

V o i d	ibmClearDirtyAttributes() Clears the list of dirty attributes.
L i s t	ibmGetDirtyAttributes() Returns a list of dirty attributes based on the value interface set on the map.
O b j e c t	ibmGetRealValue() Returns the real value object this proxy represents.

Method Detail

ibmGetDirtyAttributes

[List](#) [ibmGetDirtyAttributes\(\)](#)

Returns a list of dirty attributes based on the value interface set on the map.

The attribute name is always starts with an upper case letter. For example, if the setter for the attribute is setPrice then 'Price' is the string returned here. The runtime uses substring(3) of the setter method name as the attribute name.

Returns:

List of attribute names (Strings)

ibmGetRealValue

[Object](#) `ibmGetRealValue()`

Returns the real value object this proxy represents.

Needed internally by the BackingMap to return a separate proxy for each transaction.

Returns:

actual value object.

ibmClearDirtyAttributes

`void ibmClearDirtyAttributes()`

Clears the list of dirty attributes.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Class TransactionCallbackException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[com.ibm.websphere.objectgrid.ClientServerTransactionCallbackException](#),
[ReplicationVotedToRollbackTransactionException](#)

```
public class TransactionCallbackException
extends ObjectGridException
```

This exception is thrown when a method call to TransactionCallback fails.

Since:

WAS XD 6.0, XC10

See Also:

[TransactionCallback](#), [Serialized Form](#)

Constructor Summary

[TransactionCallbackException](#)()

Constructs a new TransactionCallbackException with null as its detail message.

[TransactionCallbackException](#)([String](#) message)

Constructs a new TransactionCallbackException with the specified detail message.

[TransactionCallbackException](#)([String](#) message, [Throwable](#) cause)

Constructs a new TransactionCallbackException with the specified detail message and cause.

[TransactionCallbackException](#)([Throwable](#) cause)

Constructs a new TransactionCallbackException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#),
[printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionCallbackException

```
public TransactionCallbackException()
```

Constructs a new `TransactionCallbackException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionCallbackException

```
public TransactionCallbackException(String message)
```

Constructs a new `TransactionCallbackException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TransactionCallbackException

```
public TransactionCallbackException(String message,  
                                   Throwable cause)
```

Constructs a new `TransactionCallbackException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `TransactionCallbackException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

TransactionCallbackException

```
public TransactionCallbackException(Throwable cause)
```

Constructs a new `TransactionCallbackException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains

the class and detail message of cause). This constructor is useful for TransactionCallbackExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Interface TransactionCallback

All Known Subinterfaces:

[TransactionCallback.BeforeCommit](#)

All Known Implementing Classes:

[WebSphereTransactionCallback](#)

```
public interface TransactionCallback
```

Calling methods on a `Session` will send corresponding events to the `TransactionCallback`. An `ObjectGrid` can have zero or one `TransactionCallback`. `BackingMaps` defined on an `ObjectGrid` with a `TransactionCallback` should have corresponding `Loaders`.

A `TransactionCallback` works with `Loaders` and place transaction specific objects in slots on the `TxID` object that `Loaders` can obtain. Examples are database connections, prepared statement caches, etc. The `TransactionCallback` should reserve slots in the `TxID` by calling `ObjectGrid.reserveSlot(String)` using the name `TxID.SLOT_NAME`. The `TransactionCallback` can then put an object at that index in the `TxID`. A `Loader` can retrieve the index used by the `TransactionCallback` by calling an internal method on the `TransactionCallback`'s implementation. A reference to the configured `TransactionCallback` can be found using the `TxID.getSession().getObjectGrid().getTransactionCallback()` code sequence.

A `TransactionCallback` implementation that also implements the `ObjectGridLifecycleListener` interface will be automatically added as an `EventListener` on the [ObjectGrid](#) when the callback is set on the object grid.

A `TransactionCallback` may implement the `ObjectGridPlugin` interface in order to receive enhanced `ObjectGrid` plug-in lifecycle method calls. The plug-in is also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Since:

WAS XD 6.0, XC10

See Also:

`Loader`, [ObjectGrid.addEventListener\(EventListener\)](#), [ObjectGrid.getTransactionCallback\(\)](#), [ObjectGrid.reserveSlot\(String\)](#), [ObjectGrid.setTransactionCallback\(TransactionCallback\)](#), [Session.getObjectGrid\(\)](#), [TxID.putSlot\(int, Object\)](#), [TxID.getSlot\(int\)](#), [TxID.getSession\(\)](#)

Nested Class Summary

s
t
a
t
i
c

i
n
t
e
r

[TransactionCallback.BeforeCommit](#)

The `BeforeCommit` optional mix-in interface for the `TransactionCallback` plug-in interface allows plug-ins to be notified at the beginning of a [Session.commit\(\)](#).

Method Summary

void [begin](#)([TxID](#) id)

Invoked when starting a Session transaction.

void [commit](#)([TxID](#) id)

Invoked when committing a Session transaction.

void [initialize](#)([ObjectGrid](#) objectGrid)

Invoked when an ObjectGrid is initialized.

boolean [isExternalTransactionActive](#)([Session](#) session)

Called when an application attempts to use a Session with no transaction active.

void [rollback](#)([TxID](#) id)

Invoked when rolling back a Session transaction.

Method Detail

initialize

```
void initialize(ObjectGrid objectGrid)
    throws TransactionCallbackException
```

Invoked when an ObjectGrid is initialized.

This method is called so this object can do any implementation specific initialization.

Parameters:

objectGrid - A reference to the ObjectGrid.

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[ObjectGrid.reserveSlot\(String\)](#)

begin

```
void begin(TxID id)
    throws TransactionCallbackException
```

Invoked when starting a Session transaction.

A TransactionCallback can communicate the begin processing (along with the TxID) to the appropriate BackingMap and/or Loader. The Loader may use this signal to start a

corresponding transaction on the underlying connection to a database.

Parameters:

id - transaction identifier (TxID)

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#), [TxID](#)

commit

```
void commit(TxID id)
    throws TransactionCallbackException
```

Invoked when committing a Session transaction.

This method should be used to commit any underlying transaction and return any underlying connection back to the pool. The TxID is provided to determine which transaction is being committed

Parameters:

id - transaction identifier (TxID)

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[begin\(TxID\)](#), [Session.commit\(\)](#), [TxID](#)

rollback

```
void rollback(TxID id)
    throws TransactionCallbackException
```

Invoked when rolling back a Session transaction.

This method should be used to roll back any underlying transaction and return any underlying connection back to the pool. The TxID is provided to determine which transaction is being committed

Parameters:

id - transaction identifier (TxID)

Throws:

[TransactionCallbackException](#) - if an error occurs during processing

See Also:

[begin\(TxID\)](#), [Session.rollback\(\)](#), [TxID](#)

isExternalTransactionActive

```
boolean isExternalTransactionActive(Session session)
```

Called when an application attempts to use a Session with no transaction active.

The callback could return true in which case an auto `Session.begin()` is executed. If false is returned, an application exception is thrown indicating no transaction is active. This event is usually used when integrating with a J2EE environment such as WebSphere Application Server.

Parameters:

session - the session which the application is using

Returns:

true if an auto begin should be done, false if this is not the case

See Also:

[Session](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface TransactionCallback.BeforeCommit

All Superinterfaces:

[TransactionCallback](#)

Enclosing interface:

[TransactionCallback](#)

```
public static interface TransactionCallback.BeforeCommit  
extends TransactionCallback
```

The BeforeCommit optional mix-in interface for the TransactionCallback plug-in interface allows plug-ins to be notified at the beginning of a [Session.commit\(\)](#). Implementations can use the beforeCommit() method to validate changed data in the transaction and modify the data.

Since:

7.1.1

Nested Class Summary

s
t
a
t
i
c

i
n
t
e
r
f
a
c
e

[TransactionCallback.BeforeCommit.TransactionContext](#)

The TransactionContext identifies various information that's available to the beforeCommit() method.

Nested classes/interfaces inherited from interface

com.ibm.websphere.objectgrid.plugins.[TransactionCallback](#)

[TransactionCallback.BeforeCommit](#)

Method Summary

v
o
i
d

[beforeCommit](#)([TransactionCallback.BeforeCommit.TransactionContext](#) ctx)

Invoked at the beginning of a [Session.commit\(\)](#).

Methods inherited from interface

com.ibm.websphere.objectgrid.plugins.[TransactionCallback](#)

[begin](#), [commit](#), [initialize](#), [isExternalTransactionActive](#), [rollback](#)

Method Detail

beforeCommit

void **beforeCommit**([TransactionCallback.BeforeCommit.TransactionContext](#) ctx)
throws [TransactionCallbackException](#)

Invoked at the beginning of a `Session.commit()`.

Use the `TransactionContext.getLogSequences()` method to retrieve the changes made by this transaction. Use the `TransactionContext.getTxId().getSession()` methods to access the `Session`. The `Session` can be used to access `ObjectMaps` and modify data in the current transaction.

Parameters:

ctx - the context of the transaction.

Throws:

[TransactionCallbackException](#) - if an error occurs during processing. Any exception will roll back the transaction and will be included in the `TransactionException` thrown to the caller.

See Also:

[Session.commit\(\)](#), [TxID](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Interface

TransactionCallback.BeforeCommit.TransactionContext

Enclosing interface:

[TransactionCallback.BeforeCommit](#)

```
public static interface TransactionCallback.BeforeCommit.TransactionContext
```

The TransactionContext identifies various information that's available to the beforeCommit() method.

Since:

7.1.1

Method Summary

C	
o	
j	
r	
e	
c	
i	
t	
i	
j	
o	
g	
<	
T	getLogSequences() The LogSequences that reflect the pending changes to the map.
o	
j	
r	
e	
c	
i	
t	
i	
j	
o	
g	
>	
T	getTxID() Retrieve the TxID for the transaction.
X	
I	
D	

Method Detail

getTxID

[TxID](#) [getTxID\(\)](#)

Retrieve the TxID for the transaction.

Returns:
the TxID for the transaction.

getLogSequences

[Collection<LogSequence>](#) `getLogSequences()`

The LogSequences that reflect the pending changes to the map.

Returns:
the LogSequences.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ReplicationMapListener

Deprecated. *The client replicated map function is deprecated in version 8.6. Use the [ContinuousQueryManager](#) function.*

```
public interface ReplicationMapListener
```

This interface is used to create an implementation of an event listener for client-side maps that are in replication mode. Registered listeners receive notification callbacks for replication start and exit events and data changes.

Listener instances can be registered with a map using the [ClientReplicableMap.enableClientReplication\(com.ibm.websphere.objectgrid.ClientReplicableMap.Mode, int\[\], ReplicationMapListener\)](#) method.

Since:

WAS XD 6.1, XC10

Method Summary

v o i d	onData (LogSequence logSequence) Deprecated. This method is invoked when a data change is replicated to the client replication map.
v o i d	replicationExits () Deprecated. This method is invoked when replication has been disabled for this map.
v o i d	replicationStarts () Deprecated. This method is invoked when a snapshot mode replica has been synchronized with the client or a continuous replica has started replicating.

Method Detail

replicationStarts

```
void replicationStarts()
```

Deprecated.

This method is invoked when a snapshot mode replica has been synchronized with the client or a continuous replica has started replicating.

See also:

[ClientReplicableMap.enableClientReplication\(com.ibm.websphere.objectgrid.ClientReplicableMap.Mode, int\[\], ReplicationMapListener\)](#)

onData

void **onData**([LogSequence](#) logSequence)

Deprecated.

This method is invoked when a data change is replicated to the client replication map.

Parameters:

logSequence - the log sequence containing all of the data changes.

replicationExits

void **replicationExits**()

Deprecated.

This method is invoked when replication has been disabled for this map.

See Also:

[ClientReplicableMap.disableClientReplication\(\)](#)

Overvi	Packa	Cla	TreeSerializ	Depreca	IndexHelp	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
ew	ge	ss	ed	ted		
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All	
			Classes			

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventListener

All Superinterfaces:

com.ibm.websphere.objectgrid.plugins.EventListener

```
public interface ObjectGridEventListener  
extends com.ibm.websphere.objectgrid.plugins.EventListener
```

This interface is used to create an implementation of an event listener for an ObjectGrid. Instances of ObjectGridEventListeners are set on the ObjectGrid interface. Any significant events are communicated to the application using the methods outlined below. When using Java 5, this callback also supports new callback annotation mechanism.

Since:

WAS XD 6.0, XC10

See Also:

[ObjectGrid.addEventListener\(EventListener\)](#), [ObjectGrid.removeEventListener\(EventListener\)](#), [EventListener](#)

Method Summary

v o i d	destroy() Called when the ObjectGrid associated with this listener is destroyed.
v o i d	initialize(Session session) Invoked when an ObjectGrid is initialized.
v o i d	transactionBegin(String txid, boolean isWriteThroughEnabled) Signals the beginning of a Session transaction.
v o i d	transactionEnd(String txid, boolean isWriteThroughEnabled, boolean committed, Collection changes) Signals the ending of a Session transaction.

Method Detail

initialize

```
void initialize(Session session)
```

Invoked when an ObjectGrid is initialized.

A usable Session instance is passed to this listener to provide all of the necessary access

to the various ObjectGrid objects.

Parameters:

session - a Session instance that this listener is associated with.

See Also:

[ObjectGrid.initialize\(\)](#)

transactionBegin

```
void transactionBegin(String txid,  
                    boolean isWriteThroughEnabled)
```

Signals the beginning of a Session transaction.

A stringified version of the TxID is provided for correlating with the end of the transaction, if so desired. The type of transaction is also provided by the isWriteThroughEnabled boolean parameter.

Parameters:

txid - Stringified version of the TxID

isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the Session.beginNoWriteThrough(). method. false is passed if beginNoWriteThrough() was used.

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#)

transactionEnd

```
void transactionEnd(String txid,  
                  boolean isWriteThroughEnabled,  
                  boolean committed,  
                  Collection changes)
```

Signals the ending of a Session transaction.

A string version of the TxID is provided for correlating with the begin of the transaction, if so desired. Map changes are also reported with the collection of LogSequences passed to this method. Typical uses of this event are for customers doing custom peer invalidation or peer commit push. This event listener gives them the changes. Calls to this method are made after commit and are sequenced so that they are delivered one by one, not in parallel. The event order is the commit and rollback order.

For an ObjectGridEventListener receiving changes in an [ObjectMap](#) that is configured to use a OutputFormat.RAW for the keys or values, the keys and values objects in the LogSequences will be SerializedKey or SerializedValue objects respectively. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key or value object.

To override the map's output format configuration, use the PluginOutputFormat annotation in the implementation class.

Parameters:

txid - string version of the TxID

isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the Session.beginNoWriteThrough(). method. false is passed if beginNoWriteThrough() was used.

committed - a boolean flag indicating whether the transaction was committed (true) or rolled back (false)

changes - a Collection of LogSequences representing the changes that were committed or rolled back.

See Also:

[LogSequence.isRollback\(\)](#), [Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#),
[Session.commit\(\)](#), [Session.rollback\(\)](#)

destroy

void **destroy**()

Called when the ObjectGrid associated with this listener is destroyed.

This method is the opposite of the initialize method. When it is called, the listener can free up any resources it uses.

See Also:

[ObjectGrid.destroy\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All Classes			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup

public interface **ObjectGridEventGroup**

This is a set of single method interfaces for fine grained events delivered for an ObjectGrid. Classes implementing these interfaces AND ObjectGridEventListener can receive these events. If an ObjectGridEventListener implements ANY of these interfaces that only the specific methods on the interfaces implemented will be called.

Since:

WAS XD 6.1, XC10

See Also:

[ObjectGridEventListener](#)

Nested Class Summary

s
t
a
t
i
c

i
n
t
e
r
f
a
c
e

s
t
a
t
i
c

i
n
t
e
r
f
a
c
e

s
t
a
t
i
c

i
n
t

[ObjectGridEventGroup.ShardEvents](#)

These events are fired when a shard is made a primary shard and when the shard is demoted from a primary.

[ObjectGridEventGroup.ShardLifecycle](#)

These events are fired when an ObjectGrid shard is initialized and destroyed.

[ObjectGridEventGroup.TransactionEvents](#)

These events are called every single transaction.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
OD

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup.ShardLifecycle

Enclosing interface:

[ObjectGridEventGroup](#)

```
public static interface ObjectGridEventGroup.ShardLifecycle
```

These events are fired when an ObjectGrid shard is initialized and destroyed. A shard can be activated/deactivated multiple times within these two events.

Method Summary

v o i d	destroy() Called when the ObjectGrid associated with this listener is destroyed.
------------------	---

v o i d	initialize(Session session) Invoked when an ObjectGrid is initialized.
------------------	---

Method Detail

initialize

```
void initialize(Session session)
```

Invoked when an ObjectGrid is initialized.

A usable Session instance is passed to this listener to provide all of the necessary access to the various ObjectGrid objects.

Parameters:

session - a Session instance that this listener is associated with.

See Also:

[ObjectGrid.initialize\(\)](#)

destroy

```
void destroy()
```

Called when the ObjectGrid associated with this listener is destroyed.

This method is the opposite of the initialize method. When it is called, the listener can free up any resources it uses.

See Also:

[ObjectGrid.destroy\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup.ShardEvents

Enclosing interface:

[ObjectGridEventGroup](#)

```
public static interface ObjectGridEventGroup.ShardEvents
```

These events are fired when a shard is made a primary shard and when the shard is demoted from a primary.

Method Summary

v o i d	shardActivated (ObjectGrid grid) This is called when a shard is promoted to a primary.
v o i d	shardDeactivate (ObjectGrid grid) This is called when a primary shard is demoted to a replica.

Method Detail

shardActivated

```
void shardActivated(ObjectGrid grid)
```

This is called when a shard is promoted to a primary.

Parameters:

grid - This is a local reference to the shard containing the primary data.

shardDeactivate

```
void shardDeactivate(ObjectGrid grid)
```

This is called when a primary shard is demoted to a replica. This can happen is the balancer decides the primary is better placed in a different container. Replication is still active until this method returns to the caller. If any application controlled transactions are in flight then they should be stopped before returning. Once this method returns then any remaining transactions will fail.

Parameters:

grid - A reference to the shard.

com.ibm.websphere.objectgrid.plugins

Interface ObjectGridEventGroup.TransactionEvents

Enclosing interface:

[ObjectGridEventGroup](#)

```
public static interface ObjectGridEventGroup.TransactionEvents
```

These events are called every single transaction. These are primarily used when transaction level listening is required. This is usually for pushing changes or invalidation events to peer caches for simple scenarios.

Method Summary

v o i d	transactionBegin (String txid, boolean isWriteThroughEnabled) Signals the beginning of a Session transaction.
------------------	---

v o i d	transactionEnd (String txid, boolean isWriteThroughEnabled, boolean committed, Collection changes) Signals the ending of a Session transaction.
------------------	---

Method Detail

transactionBegin

```
void transactionBegin(String txid,  
                     boolean isWriteThroughEnabled)
```

Signals the beginning of a Session transaction.

A stringified version of the TxID is provided for correlating with the end of the transaction, if so desired. The type of transaction is also provided by the isWriteThroughEnabled boolean parameter.

Parameters:

txid - Stringified version of the TxID

isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the `Session.beginNoWriteThrough()` method. false is passed if `beginNoWriteThrough()` was used.

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#)

transactionEnd

```
void transactionEnd(String txid,  
                  boolean isWriteThroughEnabled,
```

boolean committed,
[Collection](#) changes)

Signals the ending of a Session transaction.

A string version of the TxID is provided for correlating with the begin of the transaction, if so desired. Map changes are also reported with the collection of LogSequences passed to this method. Typical uses of this event are for customers doing custom peer invalidation or peer commit push. This event listener gives them the changes. Calls to this method are made after commit and are sequenced so that they are delivered one by one, not in parallel. The event order is the commit and rollback order.

Parameters:

- txid - string version of the TxID
- isWriteThroughEnabled - boolean flag indicating whether the Session transaction was started using the `Session.beginNoWriteThrough()` method. false is passed if `beginNoWriteThrough()` was used.
- committed - a boolean flag indicating whether the transaction was committed (true) or rolled back (false)
- changes - a Collection of LogSequences representing the changes that were committed or rolled back.

See Also:

[LogSequence.isRollback\(\)](#), [Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#), [Session.commit\(\)](#), [Session.rollback\(\)](#)

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All			Classes			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD								
OD								

com.ibm.websphere.objectgrid.plugins

Interface MapEventListener

All Superinterfaces:

com.ibm.websphere.objectgrid.plugins.EventListener

```
public interface MapEventListener  
extends com.ibm.websphere.objectgrid.plugins.EventListener
```

This callback interface is implemented by the application when it wants to receive events about a Map such as the eviction of a map entry.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.addMapEventListener\(EventListener\)](#),
[BackingMap.removeMapEventListener\(EventListener\)](#), EventListener

Method Summary

v o i d	entryEvicted (Object key, Object value) Invoked when the specified entry is evicted from the map.
------------------	---

v o i d	preloadCompleted (Throwable t) Invoked when the preloading of this map has completed.
------------------	---

Method Detail

entryEvicted

```
void entryEvicted(Object key,  
                 Object value)
```

Invoked when the specified entry is evicted from the map.

The eviction could have occurred either by an Evictor's processing or by invoking one of the invalidate methods on the ObjectMap.

For a MapEventListener in an [ObjectMap](#) that is configured to use OutputFormat.RAW for the keys and values, the keys and values objects passed will be SerializedKey or SerializedValue objects respectively. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key or value object.

To override the map's output format configuration, use the PluginOutputFormat annotation in the implementation class.

Parameters:

key - The key for the map entry that was evicted.
value - The value that was in in the map entry evicted. The value object should not be modified.

See Also:

[Evictor](#), [EvictionEventCallback](#), [ObjectMap.invalidate\(Object, boolean\)](#)

preloadCompleted

void `preloadCompleted`([Throwable](#) t)

Invoked when the preloading of this map has completed.

This method is useful to determine when a preload operation finishes if asynchronous preloading is enabled. In addition if any error occurred during synchronous or asynchronous preload, it is reported with the invocation of this method.

Parameters:

t - A Throwable object that indicates if preload completed without any Throwable occurring during the preload of the map. A null reference indicates preload completed without any Throwable objects occurring during the preload of the map.

See Also:

`Loader.preloadMap(Session, BackingMap)`, [BackingMap.setPreloadMode\(boolean\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Interface LogSequenceFilter

public interface **LogSequenceFilter**

This interface can be used to filter a LogSequence. As an operation, such as serialization, needs to know whether a given LogElement should be included or not, this callback object will be used for the boolean check. If the given LogElement should be used in the operation, then "true" should be returned. If the given LogElement should not be used, then "false" should be returned. This interface is primarily used by the `serialize` method of the `LogSequenceTransformer` class.

Since:

WAS XD 6.0, XC10

Method Summary

b o o l e a n	<code>accept</code> (LogElement logElement) Returns true if the given LogElement should be used; false if the given LogElement should not be used.
---------------------------------	--

Method Detail

accept

boolean `accept`([LogElement](#) logElement)

Returns true if the given LogElement should be used; false if the given LogElement should not be used.

Parameters:

logElement - the LogElement to be filtered

Returns:

true if the given LogElement should be used in the operation; false otherwise.

com.ibm.websphere.objectgrid.plugins
Interface LogSequence

All Superinterfaces:
[Serializable](#)

public interface **LogSequence**
 extends [Serializable](#)

LogSequence is the ordered list of changes performed against a given map for a given transaction. These changes are recorded as LogElement objects.

Since:
 WAS XD 6.0, XC10

Method Summary	
I t e r a t o r	<p>getAllChanges() Returns an iterator for processing all of the changes for a LogSequence.</p>
I t e r a t o r	<p>getChangesByKeys(Collection keys) Returns an iterator for processing the LogElements that have the requested keys.</p>
I t e r a t o r	<p>getChangesByTypes(Collection types) Returns an iterator for processing the LogElements that are of the requested type.</p>
S t r i n g	<p>getMapName() Returns the name of the map that these changes apply to.</p>
S t r i n g	<p>getObjectGridName() Returns the name of the ObjectGrid that houses the map that these changes apply to.</p>

I t e r a t o r	<p>getPendingChanges() Returns an iterator for processing all of the "pending" changes for a LogSequence (for example, pending inserts, updates, and deletes).</p>
b o o l e a n	<p>isDirty() Returns whether this LogSequence has any LogElements that would "dirty" a Map.</p>
b o o l e a n	<p>isRollback() Returns whether or not this LogSequence was generated to rollback a transaction.</p>
i n t	<p>size() Returns the total number of LogElements within this LogSequence.</p>

Method Detail

size

int **size()**

Returns the total number of LogElements within this LogSequence.

Returns:

total number of LogElements

getPendingChanges

[Iterator](#) **getPendingChanges()**

Returns an iterator for processing all of the "pending" changes for a LogSequence (for example, pending inserts, updates, and deletes).

This method is normally used by a Loader. A pending change is one that has not been written out to a loader yet using a `flush()` operation. Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Returns:

an Iterator for processing the pending LogElement changes

See Also:

[ObjectMap.flush\(\)](#), [Session.flush\(\)](#)

getAllChanges

[Iterator](#) **getAllChanges()**

Returns an iterator for processing all of the changes for a LogSequence.

This method would normally be used by an Evictor and other plugins that want to know all of the changes introduced by this LogSequence. Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Returns:

an Iterator for processing all of the LogElement changes

getChangesByTypes

[Iterator](#) `getChangesByTypes(Collection types)`

Returns an iterator for processing the LogElements that are of the requested type.

Each member of the input Collection should be one of the defined LogElement Types (INSERT, UPDATE, DELETE, FETCH, TOUCH, or EVICT). Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Parameters:

types - A Collection of LogElement Types (INSERT, UPDATE, etc)

Returns:

Iterator for processing all LogElements that support the input Type(s)

Throws:

[IllegalArgumentException](#) - if types is null

See Also:

[LogElement.DELETE](#), [LogElement.EVICT](#), [LogElement.FETCH](#), [LogElement.INSERT](#), [LogElement.TOUCH](#), [LogElement.UPDATE](#), [LogElement.CLEAR](#)

getChangesByKeys

[Iterator](#) `getChangesByKeys(Collection keys)`

Returns an iterator for processing the LogElements that have the requested keys.

Note, the returned iterator's `remove()` is not allowed to be called and will throw an exception.

Parameters:

keys - a collection of key objects

Returns:

an Iterator for processing all LogElements that match the input key(s)

getMapName

[String](#) `getMapName()`

Returns the name of the map that these changes apply to.

The caller can use the return value of this method as input to the `Session.getMap(String)` method.

Returns:

The name of the map that these changes apply to

See Also:

[Session.getMap\(String\)](#)

getObjectGridName

[String](#) `getObjectGridName()`

Returns the name of the ObjectGrid that houses the map that these changes apply to.

Returns:

The name of the ObjectGrid that this LogSequence is associated with

Since:

WAS XD 6.0.1

isDirty

boolean **isDirty**()

Returns whether this LogSequence has any LogElements that would "dirty" a Map.

That is, if it contains any LogElements of any type other than Fetch/Get, it is considered "dirty".

Returns:

true if the LogSequence would modify a Map, if applied; false if the LogSequence would not modify a Map, if applied

isRollback

boolean **isRollback**()

Returns whether or not this LogSequence was generated to rollback a transaction.

Note, depending on when this LogSequence is used, the transaction itself might already be rolled back.

Returns:

true iff this LogSequence was generated to rollback a transaction.

Since:

WAS XD 6.0.1

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins
Interface LogElement

All Superinterfaces:
[Serializable](#)

```
public interface LogElement
extends Serializable
```

LogElements are the individual entries within a LogSequence. A LogElement has attributes such as operation type (delete, insert, update, etc.), current value, last access time, versioned value, etc. A LogElement is created during a transaction to record in-flight operations. For a LogElement on an [ObjectMap](#) that is configured to use OutputFormat.RAW for the keys or values, the keys or values objects in the LogElement will be SerializedKey or SerializedValue objects respectively. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key or value object. To override the map's output format configuration, use the PluginOutputFormat annotation in the caller of the LogElement.

Since:
 WAS XD 6.0, XC10

See Also:
[LogSequence](#)

Nested Class Summary

s t a t i c c l a s s	<p>LogElement.Type</p> <p>The Type class is used to represent a LogElement type.</p>
---	--

Field Summary

s t a t i c L o g E l e m e n t	<p>CLEAR</p> <p>The Type that represents the CLEAR operation.</p>
--	---

t
I
V
P
E

s
t
a
t
i
c
i
n
t

[CODE_CLEAR](#)

The type code constant for CLEAR.

s
t
a
t
i
c
i
n
t

[CODE_DELETE](#)

The type code constant for DELETE.

s
t
a
t
i
c
i
n
t

[CODE_EVICT](#)

The type code constant for EVICT.

s
t
a
t
i
c
i
n
t

[CODE_FETCH](#)

The type code constant for FETCH.

s
t
a
t
i
c
i
n
t

[CODE_INSERT](#)

The type code constant for INSERT.

s
t
a
t
i
c
i
n
t

[CODE_LOCK](#)

The type code constant for LOCK.

s
t
a
t
i
c
i

[CODE_TOUCH](#)

The type code constant for TOUCH.

n
t

s
t
a
t
i
c
i
n
t

CODE_UNDO_NOT_NEEDED

The code constant for UNDO_NOT_NEEDED.

s
t
a
t
i
c
i
n
t

CODE_UPDATE

The type code constant for UPDATE.

s
t
a
t
i
c
i
n
t

CODE_UPSERT

The type code constant for UPSERT.

s
t
a
t
i
c
L
O
Q
U
E
L
E
M
E
N
T
T
Y
P
E

DELETE

The Type that represents the DELETE operation.

s
t
a
t
i
c
L
O
Q
U
E
L
E
M
E
N
T
T
Y
P
E

EVICT

The Type that represents the EVICT operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T
T
I
P
E

FETCH

The Type that represents the FETCH operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T
T
I
P
E

INSERT

The Type that represents the INSERT operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T
T
I
P
E

LOCK

The Type that represents the LOCK operation.

s
t
a
t
i
c
L
O
G
E
L
E
M
E
N
T

TOUCH

The Type that represents the TOUCH operation.

e
n
t
i
t
y
p
e

s
t
a
t
i
c
L
o
g
E
l
e
m
e
n
t
i
t
y
p
e

UNDO_NOT_NEEDED

The Type that represents an UNDO action is NOT required for this LogElement.

s
t
a
t
i
c
L
o
g
E
l
e
m
e
n
t
i
t
y
p
e

UPDATE

The Type that represents the UPDATE operation.

s
t
a
t
i
c
L
o
g
E
l
e
m
e
n
t
i
t
y
p
e

UPSERT

The Type that represents the UPSERT operation.

Method Summary

O
b
j
e
c
t

[getAfterImage\(\)](#)

Gets the "after image" value object.

O
b
j
e
c
t

[getBeforeImage\(\)](#)

Gets the "before image" of the value object.

C
a
c
h
e
E
n
t
r
y

[getCacheEntry\(\)](#)

Returns the CacheEntry for this key.

O
b
j
e
c
t

[getCurrentValue\(\)](#)

Gets the value for this LogElement.

O
b
j
e
c
t

[getKey\(\)](#)

Returns the key for this LogElement.

L
o
g

[getLastAccessTime\(\)](#)

Returns the last access time associated with this LogElement.

L
o
g
E
l
e
m
e
n
t
.
T
y
p
e

[getType\(\)](#)

Gets the type of this LogElement.

L
o
g
E
l
e
m
e
n
t
.
T
y
p
e

[getUndoType\(\)](#)

Returns what operation must be performed to "undo" a prior change the transaction made to the map entry.

e	
Object	<p>getVersionedValue() Gets the versioned object at the time the object was first associated with the transaction.</p>
boolean	<p>isCascaded() Answers true if this LogElement is a result of a cascade operation.</p>
boolean	<p>isPending() Answers true if this change has NOT been applied to the loader.</p>
void	<p>setVersionedValue(Object v) Used to update the versioned object after an update of map entry occurs.</p>

Field Detail

CODE_INSERT

static final int **CODE_INSERT**

The type code constant for INSERT.

See Also:

[INSERT](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_UPDATE

static final int **CODE_UPDATE**

The type code constant for UPDATE.

See Also:

[UPDATE](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_DELETE

static final int **CODE_DELETE**

The type code constant for DELETE.

See Also:

[DELETE](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_EVICT

static final int **CODE_EVICT**

The type code constant for EVICT.

See Also:

[EVICT](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_FETCH

static final int **CODE_FETCH**

The type code constant for FETCH.

See Also:

[FETCH](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_TOUCH

static final int **CODE_TOUCH**

The type code constant for TOUCH.

See Also:

[TOUCH](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_CLEAR

static final int **CODE_CLEAR**

The type code constant for CLEAR.

Since:

WAS XD 6.1.0.3

See Also:

[CLEAR](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_LOCK

static final int **CODE_LOCK**

The type code constant for LOCK.

Since:

8.6, XC10 2.5

See Also:

[LOCK](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_UPSERT

static final int **CODE_UPSERT**

The type code constant for UPSERT.

Since:

8.6, XC10 2.5

See Also:

[UPSERT](#), [LogElement.Type.getCode\(\)](#), [Constant Field Values](#)

CODE_UNDO_NOT_NEEDED

static final int **CODE_UNDO_NOT_NEEDED**

The code constant for UNDO_NOT_NEEDED. Used to indicate no operation is needed to undo the changes for this LogElement since this LogElement was never processed or it was an operation that does not require an undo operation.

See Also:

[Constant Field Values](#)

INSERT

static final [LogElement.Type](#) **INSERT**

The Type that represents the INSERT operation.

UPDATE

static final [LogElement.Type](#) **UPDATE**

The Type that represents the UPDATE operation.

DELETE

static final [LogElement.Type](#) **DELETE**

The Type that represents the DELETE operation.

EVICT

static final [LogElement.Type](#) **EVICT**

The Type that represents the EVICT operation.

FETCH

static final [LogElement.Type](#) **FETCH**

The Type that represents the FETCH operation.

TOUCH

static final [LogElement.Type](#) **TOUCH**

The Type that represents the TOUCH operation.

CLEAR

static final [LogElement.Type](#) **CLEAR**

The Type that represents the CLEAR operation.

Since:

WAS XD 6.1.0.3

LOCK

static final [LogElement.Type](#) LOCK

The Type that represents the LOCK operation.

Since:
8.6, XC10 2.5

UPSERT

static final [LogElement.Type](#) UPSERT

The Type that represents the UPSERT operation.

Since:
8.6, XC10 2.5

UNDO_NOT_NEEDED

static final [LogElement.Type](#) UNDO_NOT_NEEDED

The Type that represents an UNDO action is NOT required for this LogElement.

Method Detail

getType

[LogElement.Type](#) getType()

Gets the type of this LogElement. The type indicates what operation needs to be applied to the map entry.

Returns:
the type of this LogElement.

getCurrentValue

[Object](#) getCurrentValue()

Gets the value for this LogElement.

For a LogElement on an [ObjectMap](#) that is configured to use a ValueSerializerPlugin, the values in the LogSequence will be SerializedValue objects. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original value object.

The original value represents the new value that should be applied to the BackingMap and Loader. This value can be cast to ValueProxyInfo when a value interface is in use in order to determine the set of dirty attributes.

Returns:
the value in case of INSERT, UPDATE, UPSERT, or FETCH, null in the case of DELETE or EVICT.

See Also:
[ValueProxyInfo](#)

getCacheEntry

[CacheEntry](#) `getCacheEntry()`

Returns the CacheEntry for this key. The key, current committed value, etc. can be accessed from the CacheEntry.

Returns:

the entry in the cache that is requested to be updated.

See Also:

[CacheEntry.getCommittedValue\(\)](#), [getKey\(\)](#)

isPending

boolean `isPending()`

Answers true if this change has NOT been applied to the loader.

Changes can previously be applied to a loader using the `ObjectMap.flush()` or `Session.flush()` methods. This method reveals whether the change in this `LogElement` has already been applied to the Loader using one of those methods.

Returns:

true if this change has NOT been applied to the loader.

See Also:

[ObjectMap.flush\(\)](#), [Session.flush\(\)](#)

getVersionedValue

[Object](#) `getVersionedValue()`

Gets the versioned object at the time the object was first associated with the transaction.

For a `LogElement` on an [ObjectMap](#) that is configured to use a `ValueSerializerPlugin`, the versioned object will be returned as an `XsDataInputStream`, read will be `SerializedKey` or `SerializedValue` objects respectively. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original key or value object. For a `LogElement` on an [ObjectMap](#) that is configured to use a `ValueSerializerPlugin` that generates version objects, the version object will be the data stream representing the data.

Returns:

The versioned object.

See Also:

`OptimisticCallback`

setVersionedValue

void `setVersionedValue(Object v)`

Used to update the versioned object after an update of map entry occurs.

The Loader can use this method when it is using an optimistic strategy and uses the `OptimisticCallback.updateVersionedObjectForValue(Object)` method to get an updated version object.

Parameters:

v - The versioned object.

See Also:

`OptimisticCallback.updateVersionedObjectForValue(Object)`

getLastAccessTime

long `getLastAccessTime()`

Returns the last access time associated with this `LogElement`.

Returns:

last access time

getUndoType

[LogElement.Type](#) `getUndoType()`

Returns what operation must be performed to "undo" a prior change the transaction made to the map entry.

Note, an undo type of `UNDO_NOT_NEEDED` is returned if nothing needs to be undone for this `LogElement`.

Returns:

the "undo" type of this `LogElement`. It can be one of: `INSERT`, `UPDATE`, `DELETE` or `UNDO_NOT_NEEDED`

getBeforeImage

[Object](#) `getBeforeImage()`

Gets the "before image" of the value object.

The "before image" is the value object that existed in map entry prior to applying a change to map entry. Note, it is possible for a `null` reference to be returned (e.g. in the case where a new map entry is created).

For a `LogElement` on an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the values, the value will be a `SerializedValue` object. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original value object.

To override the map's output format configuration, use the `PluginOutputFormat` annotation in the caller of the `LogElement`.

Returns:

the value prior to applying the change

getAfterImage

[Object](#) `getAfterImage()`

Gets the "after image" value object.

The "after image" is the value object that existed in map entry after applying a change to the map entry. Note, it is possible for a `null` reference to be returned (e.g. in the case where an existing map entry is removed/evicted).

For a `LogElement` on an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the value, the value will be a `SerializedValue` object. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original value object.

To override the map's output format configuration, use the `PluginOutputFormat` annotation in the caller of the `LogElement`.

Returns:
the value after applying the change

isCascaded

boolean `isCascaded()`

Answers true if this `LogElement` is a result of a cascade operation. This only applies to ObjectGrid EntityManager programming model.

ObjectGrid EntityManager supports cascade operations. For example, when persisting an entity P, if P has a relation to entity C with `CascadeType.PERSIST` enabled, C will also be persisted as a result of the cascade operation. The method `isCascaded()` returns true for the `LogElement` object which represents C, and the method returns false for the `LogElement` object which represents P.

Returns:
true if the `LogElement` object is a result of cascade operation.

Since:
6.1.0.5 FIX1

See Also:
`EntityManager`

getKey

[Object](#) `getKey()`

Returns the key for this `LogElement`.

For a `LogElement` on an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the keys, the value will be a `SerializedKey` object. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original key object.

To override the map's output format configuration, use the `PluginOutputFormat` annotation in the caller of the `LogElement`.

This method can be used instead of `LogElement.getCacheEntry().getKey()`.

Returns:
the key for this `LogElement`.

Since:
7.0

See Also:
[CacheEntry.getKey\(\)](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins
Class LogElement.Type

[java.lang.Object](#)



All Implemented Interfaces:

[Comparable](#)

Enclosing interface:

[LogElement](#)

```

public static class LogElement.Type
    extends Object
    implements Comparable
  
```

The Type class is used to represent a LogElement type.

Since:

WAS XD 6.0

Method Summary	
i n t	compareTo (Object object)
i n t	getCode () Gets the type code for this object.
S t r i n g	toString ()
s t a t i c	valueOf (int typeCode) Return the Type for a given type code.

Methods inherited from class java.lang.[Object](#)[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)**Method Detail****getCode**

```
public int getCode()
```

Gets the type code for this object.

Returns:
the type code

compareTo

```
public int compareTo(Object object)
```

Specified by:
[compareTo](#) in interface [Comparable](#)

See Also:
[Comparable.compareTo\(Object\)](#)

toString

```
public String toString()
```

Overrides:
[toString](#) in class [Object](#)

See Also:
[Object.toString\(\)](#)

valueOf

```
public static LogElement.Type valueOf(int typeCode)
```

Return the Type for a given type code.

Parameters:
typeCode - the typecode

Returns:
the Type

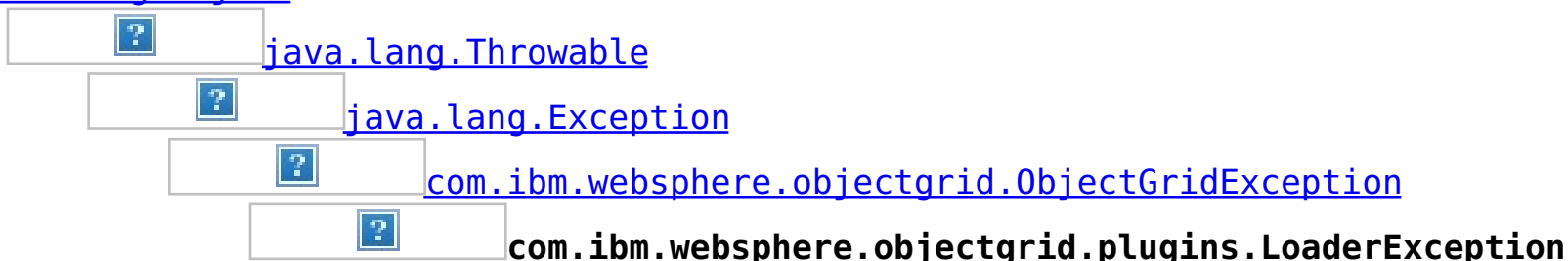
Throws:
[IllegalArgumentException](#) - if the typeCode isn't valid.

Since:
8.6, XC10 2.5

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins
Class LoaderException

[java.lang.Object](#)



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:
[UnavailableServiceException](#)

```
public class LoaderException
extends ObjectGridException
```

This exception is the base exception for any exceptions encountered by a Loader.

Since:
WAS XD 6.0, XC10

See Also:
Loader, [Serialized Form](#)

Constructor Summary

LoaderException ()	Constructs a new LoaderException with null as its detail message.
LoaderException (String message)	Constructs a new LoaderException with the specified detail message.
LoaderException (String message, Throwable cause)	Constructs a new LoaderException with the specified detail message and cause.
LoaderException (Throwable cause)	Constructs a new LoaderException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)
[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)
[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LoaderException

```
public LoaderException()
```

Constructs a new LoaderException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LoaderException

```
public LoaderException(String message)
```

Constructs a new LoaderException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

LoaderException

```
public LoaderException(String message,  
                       Throwable cause)
```

Constructs a new LoaderException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this LoaderException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

LoaderException

```
public LoaderException(Throwable cause)
```

Constructs a new LoaderException with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for LoaderExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for

later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							

[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface EvictorData

All Known Implementing Classes:

[EvictorData.SpecialEvictorData](#)

```
public interface EvictorData
```

This interface is optionally used by an implementator of the `Evictor` interface. Application changes applied to a `BackingMap` are asynchronous from the `Evictor` activity. The `Evictor` is not notified of changes to the `BackingMap` until after application transactions are committed. Consequently, if an `Evictor` decides to evict a map entry, it is possible that the `BackingMap` could evict an entry that was different from the original entry it was tracking. For example, consider that an application could execute a transaction that removes a map entry. Before the `Evictor` is notified of the remove, another transaction inserts a new entry into the `BackingMap` for the same key as the old entry. Consequently, the `Evictor` could evict the newly created entry when it meant to evict the old entry. To help close this small timing window, the `Evictor` can use this interface to associate evictor specific data with a map entry. The `Evictor` can then do the following:

- store the `EvictorData` object for a map entry by using the `EvictionEventCallback.setEvictorData(Object, EvictorData)` method.
- retrieve the `EvictorData` object for a map entry by using the `EvictionEventCallback.getEvictorData(Object)` method.
- Conditionally evict a map entry if and only if the cache entry for a specified key has the exact same `EvictorData` object (the `java ==` operator returns true) associated with it by using the `EvictionEventCallback.evictMapEntries(List)` method.

Since:

WAS XD 6.0.1, XC10

See Also:

[Evictor](#), [EvictionEventCallback](#)

Nested Class Summary

s
t
a
t
i
c
c
l
a
s
s

[EvictorData.SpecialEvictorData](#)

Special value class used for representing the key not being found in the `BackingMap`.

Field Summary

s
t
a
t

i
c
E
V
i
c
t
o
r
D
a
t
a

[KEY_NOT_FOUND](#)

A special value indicating that the key was not found.

Method Summary

O
b
j
e
c
t

[getKey\(\)](#)

Retrieves the key object for this EvictorData instance.

Field Detail

KEY_NOT_FOUND

static final [EvictorData](#) KEY_NOT_FOUND

A special value indicating that the key was not found.

Method Detail

getKey

[Object](#) [getKey\(\)](#)

Retrieves the key object for this EvictorData instance.

Returns:

the same key object that was passed to the `EvictionEventCallback.setEvictorData(Object, EvictorData)` method when this `EvictorData` was associated with the map entry with the given key.

See Also:

[EvictionEventCallback.setEvictorData\(Object, EvictorData\)](#)

[Overvi](#) [Packa](#) [Cla](#) [TreeSerializ](#) [Depreca](#) [Index](#) [Help](#)
[ew](#) [ge](#) [ss](#) [ed](#) [ted](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Class EvictorData.SpecialEvictorData

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.plugins.EvictorData.SpecialEvictorData

All Implemented Interfaces:

[EvictorData](#)

Enclosing interface:

[EvictorData](#)

```
public static final class EvictorData.SpecialEvictorData
extends Object
implements EvictorData
```

Special value class used for representing the key not being found in the BackingMap.

Since:

WAS XD 6.0.1

Nested Class Summary

Nested classes/interfaces inherited from interface
com.ibm.websphere.objectgrid.plugins.[EvictorData](#)

[EvictorData.SpecialEvictorData](#)

Field Summary

Fields inherited from interface com.ibm.websphere.objectgrid.plugins.[EvictorData](#)

[KEY_NOT_FOUND](#)

Constructor Summary

[EvictorData.SpecialEvictorData](#)()

Method Summary

[getKey](#)()

Dummy implementation method since this class will not be called.

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

EvictorData.SpecialEvictorData

```
public EvictorData.SpecialEvictorData()
```

Method Detail

getKey

```
public Object getKey()
```

Dummy implementation method since this class will not be called.

Specified by:

[getKey](#) in interface [EvictorData](#)

Returns:

null

See Also:

[EvictionEventCallback.setEvictorData\(Object, EvictorData\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins

Interface Evictor

All Known Implementing Classes:

[LFUEvictor](#), [LRUEvictor](#)

public interface **Evictor**

Data contained in a BackingMap are evicted when the map is full. This plugin is used by the BackingMap to determine when and what to evict from the map based on some algorithm (LRU, LFU, time based, etc).

An Evictor implementation that also implements the BackingMapLifecycleListener interface will be automatically added as an EventListener on the [BackingMap](#) when the evictor set on the backing map.

An Evictor may also implement the BackingMapPlugin interface in order to receive enhanced BackingMap plug-in lifecycle method calls. The plug-in is then also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.addMapEventListener\(EventListener\)](#), [BackingMap.setEvictor\(Evictor\)](#), [EvictorData](#)

Method Summary	
v o i d	activate() This method is called to activate the Evictor.
v o i d	apply(LogSequence sequence) Called after a transaction has committed to allow the evictor to track object usage in the BackingMap.
v o i d	deactivate() This method is called to deactivate the Evictor.
v o i d	destroy() Called when the BackingMap associated with this evictor is destroyed.
v o i d	initialize(BackingMap map, EvictionEventCallback callback) Called by a BackingMap instance during the evictor initialization time.

Method Detail

initialize

```
void initialize(BackingMap map,  
               EvictionEventCallback callback)
```

Called by a [BackingMap](#) instance during the evictor initialization time.

The [BackingMap](#) calls this method so the Evictor instance can have references to the [BackingMap](#) and [EvictionEventCallback](#) instances. The evictor can signal events to have specific entries evicted using the [EvictionEventCallback](#).

Parameters:

map - the [BackingMap](#) instance
callback - the [EvictionEventCallback](#) instance

See Also:

[BackingMap](#), [EvictionEventCallback](#)

destroy

```
void destroy()
```

Called when the [BackingMap](#) associated with this evictor is destroyed.

This method is the opposite of the `initialize` method. When it is called, the Evictor can free up any resources it uses.

See Also:

[ObjectGrid.destroy\(\)](#)

apply

```
void apply(LogSequence sequence)
```

Called after a transaction has committed to allow the evictor to track object usage in the [BackingMap](#).

This method also reports any entries that have been successfully evicted. Note, this method is not called for transactions that are rolled back. If there is a need to track object usage for rolled back transactions, the evictor must implement the [RollbackEvictor](#) interface as well.

This method is called after a transaction has completed. Consequently, all transaction locks that were acquired by the completed transaction are no longer held. Potentially, multiple threads could call this method concurrently and each thread would be completing its own transaction. Since transaction locks are already released by the completed transaction, this method must provide its own synchronization to ensure it is thread safe. For an Evictor in an [ObjectMap](#) that is configured to use `OutputFormat.RAW` for the keys or values, the keys and values objects in the [LogSequence](#) will be `SerializedKey` or `SerializedValue` objects respectively. If required, you can use the `SerializedEntry.getObject()` method to retrieve (possibly inflating the serialized object) the original key or value object. To override the map's output format configuration, use the `PluginOutputFormat` annotation in the implementation class.

Parameters:

sequence - the [LogSequence](#) of changes committed to the map

See Also:

[RollbackEvictor](#)

activate

void **activate**()

This method is called to activate the Evictor. Until this method is called, the Evictor must not use the EvictionEventCallback interface to evict any map entries. If it does use the EvictionEventcallback interface to evict map entries prior to activate being called, an IllegalStateException is thrown.

deactivate

void **deactivate**()

This method is called to deactivate the Evictor. Once this method is called, the Evictor must quit using the EvictionEventCallback interface to evict any map entries. If it does use the EvictionEventcallback interface after this method is called, an IllegalStateException is thrown.

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.plugins

Interface EvictionEventCallback

public interface **EvictionEventCallback**

An instance of EvictionEventCallback is passed into the Evictor at initialization time. When an eviction method is called, corresponding methods of EvictionEventCallback will be called so the BackingMap can process evictions.

Since:

WAS XD 6.0, XC10

See Also:

[Evictor](#), [EvictorData](#)

Method Summary

v o i d	evictEntries (List keysToEvictList) If an Evictor chooses not to implement the EvictorData interface, this method can be used to evict a map entry.
------------------	---

v o i d	evictMapEntries (List evictorDataList) This method is the preferred method for the Evictor to use when evicting map entries.
------------------	--

E v i c t o r D a t a	getEvictorData (Object key) Gets the evictor data for a specified BackingMap cache entry.
---	---

v o i d	setEvictorData (Object key, EvictorData data) Sets the evictor data for a specified BackingMap cache key.
------------------	---

Method Detail

setEvictorData

void [setEvictorData](#)([Object](#) key,
[EvictorData](#) data)

Sets the evictor data for a specified BackingMap cache key.

This method can be used by an implementor of the `Evictor` interface to keep data that the evictor needs for determining which cache entry to evict.

Parameters:

key - is the key for accessing a `BackingMap` entry.

data - the `EvictorData` object to store as evictor data for a specified key.

Throws:

[IllegalArgumentException](#) - if key is a null reference or there is no `BackingMap` cache entry for this key.

Since:

WAS XD 6.0.1

See Also:

[Evictor](#)

getEvictorData

[EvictorData](#) `getEvictorData(Object key)`

Gets the evictor data for a specified `BackingMap` cache entry.

Parameters:

key - the key for the `BackingMap` entry to set.

Returns:

if the specified key is not found in `BackingMap`, then the special value `EvictorData.KEY_NOT_FOUND` is returned. If the key is found in the `BackingMap`, the same reference that was previously passed to the `setEvictorData(Object, EvictorData)` method of this interface is returned. A null reference is returned if the key is found, but the `setEvictorData` method was not previously called for the specified key.

Throws:

[IllegalArgumentException](#) - if key is a null reference.

Since:

WAS XD 6.0.1

See Also:

[setEvictorData\(Object, EvictorData\)](#), [EvictorData.KEY_NOT_FOUND](#)

evictMapEntries

void `evictMapEntries(List evictorDataList)`
throws [ObjectGridException](#)

This method is the preferred method for the `Evictor` to use when evicting map entries. A list of `EvictorData` objects is passed as an argument to this method. For each `EvictorData` object in the list, the key is obtained from the `EvictorData` object and used to determine which `BackingMap` entry to evict. The `BackingMap` entry is evicted if and only if the cache entry for `BackingMap` entry contains the exact same `EvictorData` object in it. That is, the `java ==` operator is used to ensure it is the exact same `EvictorData` object. If the `==` operator indicates a different object, then the map entry is not evicted. For those map entries that are physically evicted from the map, the `Evictor` will receive notification through its `apply` method.

Parameters:

evictorDataList - a list of `EvictorData` objects to process. The caller must guarantee this parameter is not null or contain any null references.

Throws:

[ObjectGridException](#) - if an error occurs during processing

[ClassCastException](#) - if an object in `evictorDataList` does not implement the `EvictorData` interface.

Since:

WAS XD 6.0.1

See Also:

[Evictor.apply\(LogSequence\)](#), [EvictorData.getKey\(\)](#)

evictEntries

void **evictEntries**([List](#) keysToEvictList)
throws [ObjectGridException](#)

If an Evictor chooses not to implement the EvictorData interface, this method can be used to evict a map entry. However, the Evictor must be prepared to handle the exposure of an application removing and recreating a map entry before the Evictor has an opportunity to call this method.

For this method, a list of map keys is passed. The list is evaluated and an eviction is conducted on the list. When the entries are physically evicted from the map, the Evictor will receive notification through its apply method.

Parameters:

keysToEvictList - List of keys to evict from the map. The caller must guarantee this parameter is not null or contain any null references.

Throws:

[ObjectGridException](#) - if an error occurs during processing

See Also:

[Evictor.apply\(LogSequence\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

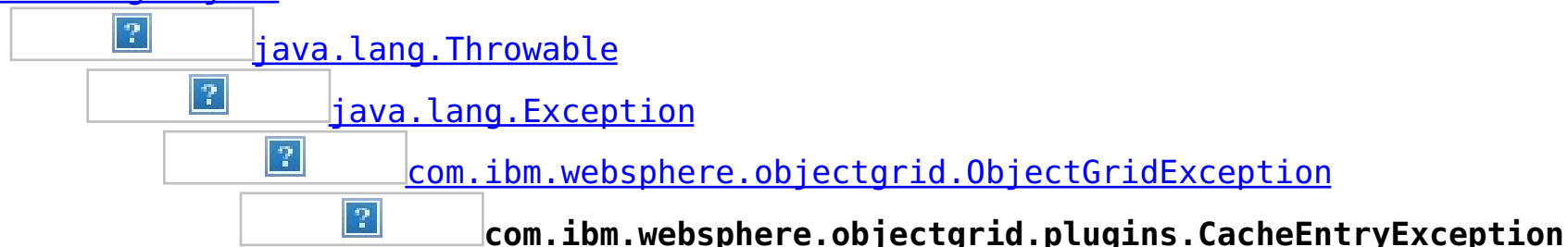
[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins
Class CacheEntryException

[java.lang.Object](#)



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

```
public class CacheEntryException
extends ObjectGridException
```

This exception indicates an error occurred during a cache entry operation.

Since:
WAS XD 6.0, XC10

See Also:
[Serialized Form](#)

Constructor Summary

[CacheEntryException](#)()

Constructs a new CacheEntryException with null as its detail message.

[CacheEntryException](#)(String message)

Constructs a new CacheEntryException with the specified detail message.

[CacheEntryException](#)(String message, Throwable cause)

Constructs a new CacheEntryException with the specified detail message and cause.

[CacheEntryException](#)(Throwable cause)

Constructs a new CacheEntryException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#),
[printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

CacheEntryException

```
public CacheEntryException()
```

Constructs a new `CacheEntryException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

CacheEntryException

```
public CacheEntryException(String message)
```

Constructs a new `CacheEntryException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

CacheEntryException

```
public CacheEntryException(String message,
                           Throwable cause)
```

Constructs a new `CacheEntryException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `CacheEntryException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

CacheEntryException

```
public CacheEntryException(Throwable cause)
```

Constructs a new `CacheEntryException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `CacheEntryExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

com.ibm.websphere.objectgrid.plugins

Interface CacheEntry

All Superinterfaces:

[Serializable](#)

```
public interface CacheEntry
extends Serializable
```

This interface represents a cache entry in an ObjectGrid map.

Since:

WAS XD 6.0, XC10

Method Summary

ObjectGrid	getCommittedValue() Returns the committed value for this entry.
ObjectGrid	getKey() Returns the key for this entry.
Long	getLastAccessTime() Returns the last time this entry was accessed.
Long	getTTL() Returns the time-to-live value for this entry.
boolean	isInBackingMap() Indicates whether this entry is in the BackingMap

Method Detail

isInBackingMap

```
boolean isInBackingMap()
```

Indicates whether this entry is in the BackingMap

Returns:

Returns true if this element is in the BackingMap

getKey

[Object](#) getKey()

Returns the key for this entry.

For a CacheEntry on an [ObjectMap](#) that is configured to use a KeySerializerPlugin, the value will be a SerializedKey object. If required, you can use the SerializedEntry.getObject() method to retrieve (possibly inflating the serialized object) the original key object.

Returns:

the key

getCommittedValue

[Object](#) getCommittedValue()

Returns the committed value for this entry.

The type of the object returned from the getCommittedValue() method depends on various configuration and storage options used by the [ObjectMap](#) that holds the CacheEntry. In the default case, getCommittedValue() returns the Java object of the same type that was put into the map. For an [ObjectMap](#) that is configured to use a ValueSerializerPlugin, the committed value depends on the underlying storage mechanism, typically represented as an array of bytes.

Returns:

the committed value

getTTL

long getTTL()

Returns the time-to-live value for this entry.

Returns:

the time-to-live value

getLastAccessTime

long getLastAccessTime()

Returns the last time this entry was accessed.

Returns:

last access time.

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.plugins
Interface BeanFactory

public interface **BeanFactory**

Implement this interface to allow bean factories like Spring or Google guice to be integrated. This allows ObjectGrid to delegate to an external Bean Factory to instantiate beans needed by ObjectGrid.

Since:
WAS XD 6.1 FIX3, XC10

Method Summary

ObjectGrid	getBean (String name) This returns an instance of the bean with the specified name.
------------	--

Method Detail

getBean

[Object](#) **getBean**([String](#) name)

This returns an instance of the bean with the specified name.

Parameters:

name - The name of the bean instance to return.

Returns:

the bean instance.

Package com.ibm.websphere.objectgrid.security

This package has the class MapPermission and class AdminPermission which represents the permissions for to access the ObjectGrid maps and ObjectGrid administration respectively.

See:

[Description](#)

Class Summary	
SecurityConstants	This class contains the constants used for security configuration.

Exception Summary	
ObjectGridSecurityException	This exception represents a general ObjectGrid security exception.

Package com.ibm.websphere.objectgrid.security Description

This package has the class MapPermission and class AdminPermission which represents the permissions for to access the ObjectGrid maps and ObjectGrid administration respectively.

MapPermission action types.

The ObjectGrid defines 5 permission actions that are used to authorize accesses to the maps. These permissions allow access to maps to be controlled by an administrator. Objects within the ObjectGrid use a simple naming scheme. Each Map is named using the convention of the ObjectGrid name followed by a period followed by the Map name. For example, if the object grid name is "myObjectGrid" and the map name is "myMap", then the map name used in the permission is "myobjectgrid.mymap".

Wildcards can be used on names with some restrictions. A wild card "*" can be used to replace the map name or the object grid name, but not partially. For example, "myObjectGrid.*", ".*myMap", and ".*.*" are valid names, but "myObject*.*" is not valid.

There are five actions with the permission object ObjectMapPermission.

- Read
This action allows get operations to be issued against a Map.
- Write
This action allows put operations to be issued against a Map. It allows existing entries to be updated.
- Remove
This action allows entries to be removed from the Map.
- Insert
This action allows clients to add entries to a Map.

- Invalidate
This action allows clients to invalidate entries from the Map.

	com.ibm.websphere.objectgrid.ObjectMap/ com.ibm.websphere.objectgrid.JavaMap
Read	boolean containsKey(Object)
	boolean equals(Object)
	Object get(Object)
	Object get(Object, Serializable)
	List getAll(List)
	List getAll(List keyList, Serializable)
	List getAllForUpdate(List)
	List getAllForUpdate(List, Serializable)
	Object getForUpdate(Object)
	Object getForUpdate(Object, Serializable)
write	Object put(Object key, Object value)
	void put(Object, Object, Serializable)
	void putAll(Map)
	void putAll(Map, Serializable)
	void update(Object, Object)
	void update(Object, Object, Serializable)
insert	public void insert(Object, Object)
	void insert(Object, Object, Serializable)
	remove Object remove(Object)
	void removeAll(Collection)
invalidate	public void invalidate(Object, boolean)
	void invalidateAll(Collection, boolean)
	int setTimeToLive(int)

An authroizationMechanism setting of the ObjectGrid has two possible values: JAAS and custom. Users can also use API [ObjectGrid.setAuthorizationMechanism\(int\)](#) to set which authorization mechanism the object grid will use.

A value "JAAS" means ObjectGrid will rely on JAAS authorization mechanism to handle the authorization. A JAAS policy file should be configured to associate permissions with a set of credentials and/or groups of credentials. We recommend that groups should be used as then new users can be added to groups without modifying the policy file.

A value "custom" means ObjectGrid will rely on custom authorization mechanism to handle the authorization. Users can set call [ObjectGrid.setObjectGridAuthorization\(com.ibm.websphere.objectgrid.security.plugins.ObjectGridAuth orization ogAuthorization\)](#) to set their custom authorization plug-in. Users can also configure the objectgrid.xml to achieve the same result.

AdminPermission types

An AdminPermission has two types: ADMIN and MONITOR. An AdminPermission with ADMIN name grants permissions to access all the ManagementMBean methods. An AdminPermission with MONITOR name grants permissions to access the ManagementMBean read-only methods. Therefore, ADMIN permission implies MONITOR permission.

The detailed operations granted to users with different permissions are listed in the following

table. These operations correspond to the methods in the ManagementMBean interface:

operations	admin	monitor
startServer	Y	N
stopServer	Y	N
forceStopServer	Y	N
setServerTrace	Y	N
retrieveServerStatus	Y	Y
getMapStats	Y	Y
getOGStats	Y	Y
getReplicationStats	Y	Y

The table can read like this: If the client has admin permission, it can execute "startServer" task; if the client has monitor permission, it cannot execute "startServer" task.

AgentPermission types

An AgentPermission represents permissions to the datagrid agents. The name of the permission is the full name of the ObjectGrid map, and the action is a "," delimited string of agent implementation class names or package names.

The following methods in the class AgentManager requires AgentPermission:

- AgentManager.callMapAgent(MapGridAgent, Collection)
- AgentManager.callMapAgent(MapGridAgent)
- AgentManager.callReduceAgent(ReduceGridAgent, Collection)
- AgentManager.callReduceAgent(ReduceGridAgent, Collection)

ObjectGridPermission types

An ObjectGridPermission represents permissions to an ObjectGrid. The name of the permission is the ObjectGrid name, and the action is either "query" or "dynamicmap".

The detailed methods which require different permissions are listed in the following table:

methods	action
Session.createObjectQuery(String)	query
EntityManager.createQuery(String)	query
Session.getMap(String)	dynamicmap

ServerMapPermission types

An ServerMapPermission represents permissions to an ObjectMap hosted in a server. The name of the permission is the full name of the ObjectGrid map name, and the action is either "replicate" or "dynamicIndex".

The detailed methods which require different ServerMapPermission are listed in the following table:

methods	action
ClientReplicableMap.enableClientReplication(Mode, int[], ReplicationMapListener)	replicate
BackingMap.createDynamicIndex(String, boolean, String, DynamicIndexCallback)	dynamicIndex
BackingMap.removeDynamicIndex(String)	dynamicIndex

SecurityConstants

SecurityConstants class contains constants used for representing the security parameters.

Overview	Package	Class	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV PACKAGE	NEXT PACKAGE	FRAMES	NO FRAMES	All Classes				

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.security
Class SecurityConstants

[java.lang.Object](#)



```
public class SecurityConstants
extends Object
```

This class contains the constants used for security configuration.

Since:
 WAS XD 6.0, XC10

Field Summary

s t a t i c i n t	<p>ACCESS BY CREATOR ONLY COMPLEMENT The access by creator only authorization is enabled to complement the ObjectGrid map authorization.</p>
s t a t i c i n t	<p>ACCESS BY CREATOR ONLY DISABLED The access by creator only authorization is disabled.</p>
s t a t i c i n t	<p>ACCESS BY CREATOR ONLY SUPERSEDE The access by creator only authorization is enabled to supersede the ObjectGrid map authorization.</p>
s t a t i c i n t	<p>AUTHORIZATION MECHANISM_CUSTOM Constant representing custom authorization</p>
s t	

a
t
t
i
c
i
n
t

[AUTHORIZATION_MECHANISM_JAAS](#)

Constant representing JAAS authorization

s
t
a
t
i
c
i
n
t

[CLIENT_CERTIFICATE_AUTHENTICATION_NEVER](#)

Constant indicating client certificate authentication is not supported.

s
t
a
t
i
c
i
n
t

[CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED](#)

Constant indicating client certificate authentication is required.

s
t
a
t
i
c
i
n
t

[CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED](#)

Constant indicating client certificate authentication is supported.

s
t
a
t
i
c
i
n
t

[CREDENTIAL_AUTHENTICATION_NEVER](#)

Constant indicating credential authentication is not supported.

s
t
a
t
i
c
i
n
t

[CREDENTIAL_AUTHENTICATION_REQUIRED](#)

Constant indicating credential authentication is required.

s
t
a
t
i
c
i
n
t

[CREDENTIAL_AUTHENTICATION_SUPPORTED](#)

Constant indicating credential authentication is supported.

s
t
a
t
i
c

[NEVER_STRING](#)

String	String representation for value Never indicating an option is not supported.
s t a t i c String	<u>NEW_SECURE_TOKEN_MANAGER_STRING</u> String representation for "autoSecret" type of the secure token manager.
s t a t i c String	<u>REQUIRED_STRING</u> String representation for value Required indicating an option is required.
s t a t i c String	<u>SECURE_TOKEN_MANAGER_CUSTOM_STRING</u> String representation for "custom" type of the secure token manager.
s t a t i c String	<u>SECURE_TOKEN_MANAGER_DEFAULT_STRING</u> String representation for "default" type of the secure token manager.
s t a t i c String	<u>SECURE_TOKEN_MANAGER_NONE_STRING</u> String representation for "none" type of the secure token manager.
s t a t i c i	<u>SSL_REQUIRED</u> Constant indicating SSL transport is required.

n t	
s t a t i c S t r i n g	<p><u>SSL_REQUIRED_STRING</u> String representation for value SSL-Required indicating SSL transport type is required.</p>
s t a t i c i n t	<p><u>SSL_SUPPORTED</u> Constant indicating SSL transport is supported.</p>
s t a t i c S t r i n g	<p><u>SSL_SUPPORTED_STRING</u> String representation for value SSL-Supported indicating SSL transport type is supported.</p>
s t a t i c S t r i n g	<p><u>SUPPORTED_STRING</u> String representation for value Supported indicating an option is supported.</p>
s t a t i c i n t	<p><u>TCP_IP</u> Constant indicating TCP/IP is the only supported transport.</p>
s t a t i c S t r i n g	<p><u>TCPIP_STRING</u> String representation for value TCP/IP indicating a transport type of TCP/IP is used.</p>

Constructor Summary

[SecurityConstants\(\)](#)

Method Summary

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Field Detail

AUTHORIZATION_MECHANISM_JAAS

```
public static final int AUTHORIZATION_MECHANISM_JAAS
```

Constant representing JAAS authorization

See Also:

[ObjectGrid.setAuthorizationMechanism\(int\)](#), [Constant Field Values](#)

AUTHORIZATION_MECHANISM_CUSTOM

```
public static final int AUTHORIZATION_MECHANISM_CUSTOM
```

Constant representing custom authorization

See Also:

[ObjectGrid.setAuthorizationMechanism\(int\)](#), [Constant Field Values](#)

TCP_IP

```
public static final int TCP_IP
```

Constant indicating TCP/IP is the only supported transport.

If the client's transport type is set to this value, TCP/IP is the only supported transport type. If the server requires SSL, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setTransportType\(int\)](#), [Constant Field Values](#)

SSL_SUPPORTED

```
public static final int SSL_SUPPORTED
```

Constant indicating SSL transport is supported.

If the client's transport type is set to this value, the client supports both TCP/IP and SSL. SSL will be used if both sides side supports SSL. Otherwise, TCP/IP will be used.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setTransportType\(int\)](#), [Constant Field Values](#)

SSL_REQUIRED

```
public static final int SSL_REQUIRED
```

Constant indicating SSL transport is required.

If the client's transport type is set to this value, SSL is the only supported transport type. If the server requires TCP/IP, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setTransportType\(int\)](#), [Constant Field Values](#)

CREDENTIAL_AUTHENTICATION_NEVER

```
public static final int CREDENTIAL_AUTHENTICATION_NEVER
```

Constant indicating credential authentication is not supported.

If the credential authentication type is set to this value, no credential authentication will be enforced. If the server requires credential authentication, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setCredentialAuthenticationType\(int\)](#), [Constant Field Values](#)

CREDENTIAL_AUTHENTICATION_SUPPORTED

```
public static final int CREDENTIAL_AUTHENTICATION_SUPPORTED
```

Constant indicating credential authentication is supported.

If the credential authentication type is set to this value, credential authentication will be enforced if and only if both client and server support credential authentication.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setCredentialAuthenticationType\(int\)](#), [Constant Field Values](#)

CREDENTIAL_AUTHENTICATION_REQUIRED

```
public static final int CREDENTIAL_AUTHENTICATION_REQUIRED
```

Constant indicating credential authentication is required.

If the credential authentication type is set to this value, credential authentication will be enforced. If the server doesn't support credential authentication, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setCredentialAuthenticationType\(int\)](#), [Constant Field Values](#)

CLIENT_CERTIFICATE_AUTHENTICATION_NEVER

```
public static final int CLIENT_CERTIFICATE_AUTHENTICATION_NEVER
```

Constant indicating client certificate authentication is not supported.

If the client certificate authentication type is set to this value, no client certificate authentication will be enforced. If the server doesn't support client certificate authentication, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setClientCertificateAuthentication\(int\)](#), [Constant Field Values](#)

CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED

```
public static final int CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED
```

Constant indicating client certificate authentication is supported.

If the client certificate authentication type is set to this value, client certificate authentication will be enforced when the following conditions are met:

- both client and server supports or requires client certificate authentication;
- the transport protocol to use is SSL;
- no credential authentication will be done.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setClientCertificateAuthentication\(int\)](#), [Constant Field Values](#)

CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED

```
public static final int CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED
```

Constant indicating client certificate authentication is required. If the client certificate authentication type is set to this value, client certificate authentication will be enforced if the following conditions are met:

- both client and server supports or requires client certificate authentication;
- the transport protocol to use is SSL;
- no credential authentication will be done.

If the server doesn't support client certificate authentication and no credential authentication will be done, the client won't be able to connect to the server.

Since:

WAS XD 6.0.1

See Also:

[ClientSecurityConfiguration.setClientCertificateAuthentication\(int\)](#), [Constant Field Values](#)

NEVER_STRING

```
public static final String NEVER_STRING
```

String representation for value Never indicating an option is not supported.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration settings "clientCertificateAuthentication" and "credentialAuthentication".

Since:

WAS XD 6.0.1

See Also:

[CLIENT_CERTIFICATE_AUTHENTICATION_NEVER](#), [CREDENTIAL_AUTHENTICATION_NEVER](#), [Constant Field Values](#)

SUPPORTED_STRING

```
public static final String SUPPORTED_STRING
```

String representation for value Supported indicating an option is supported.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration settings "clientCertificateAuthentication" and "credentialAuthentication".

Since:

WAS XD 6.0.1

See Also:

[CLIENT_CERTIFICATE_AUTHENTICATION_SUPPORTED](#), [CREDENTIAL_AUTHENTICATION_SUPPORTED](#), [Constant Field Values](#)

REQUIRED_STRING

```
public static final String REQUIRED_STRING
```

String representation for value Required indicating an option is required.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration settings "clientCertificateAuthentication" and "credentialAuthentication".

Since:

WAS XD 6.0.1

See Also:

[CLIENT_CERTIFICATE_AUTHENTICATION_REQUIRED](#), [CREDENTIAL_AUTHENTICATION_REQUIRED](#), [Constant Field Values](#)

TCPIP_STRING

```
public static final String TCPIP_STRING
```

String representation for value TCP/IP indicating a transport type of TCP/IP is used.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration setting "transportType".

Since:

WAS XD 6.0.1

See Also:

[TCP_IP](#), [Constant Field Values](#)

SSL_SUPPORTED_STRING

```
public static final String SSL_SUPPORTED_STRING
```

String representation for value SSL-Supported indicating SSL transport type is supported.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration setting "transportType".

Since:

WAS XD 6.0.1

See Also:

[SSL_SUPPORTED](#), [Constant Field Values](#)

SSL_REQUIRED_STRING

```
public static final String SSL_REQUIRED_STRING
```

String representation for value SSL-Required indicating SSL transport type is required.

This value is used as a value to configuration settings in a Properties object or property file for client and server security configurations. It is used for the configuration setting "transportType".

Since:

WAS XD 6.0.1

See Also:

[SSL_REQUIRED](#), [Constant Field Values](#)

SECURE_TOKEN_MANAGER_NONE_STRING

```
public static final String SECURE_TOKEN_MANAGER_NONE_STRING
```

String representation for "none" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType".

Since:

WAS XD 6.0.1

See Also:

[Constant Field Values](#)

SECURE_TOKEN_MANAGER_DEFAULT_STRING

```
public static final String SECURE_TOKEN_MANAGER_DEFAULT_STRING
```

String representation for "default" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType". This value requires users to provide the secure token key store settings.

Since:

WAS XD 6.0.1

See Also:

[Constant Field Values](#)

SECURE_TOKEN_MANAGER_CUSTOM_STRING

```
public static final String SECURE_TOKEN_MANAGER_CUSTOM_STRING
```

String representation for "custom" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType". This value requires users to provide the SecureTokenManager implementation class name using the "customSecureTokenManagerClass" configuration setting.

Since:

WAS XD 6.0.1

See Also:

[Constant Field Values](#)

NEW_SECURE_TOKEN_MANAGER_STRING

```
public static final String NEW_SECURE_TOKEN_MANAGER_STRING
```

String representation for "autoSecret" type of the secure token manager.

This value is used in a property file for server security configurations. It is used for the configuration setting "secureTokenManagerType". This value does not require users to provide other settings.

Since:

8.6, XC10 2.5

See Also:

[Constant Field Values](#)

ACCESS_BY_CREATOR_ONLY_DISABLED

```
public static final int ACCESS_BY_CREATOR_ONLY_DISABLED
```

The access by creator only authorization is disabled.

The access by creator authorization ensures that only the user (represented by the Principals associated with it), who inserts the data entry into the map, can access the data. Here the access means read, update, invalidate, and remove.

Since:

WAS XD 6.1 FIX3

See Also:

[Constant Field Values](#)

ACCESS_BY_CREATOR_ONLY_COMPLEMENT

```
public static final int ACCESS_BY_CREATOR_ONLY_COMPLEMENT
```

The access by creator only authorization is enabled to complement the ObjectGrid map authorization.

The access by creator authorization ensures that only the user (represented by the Principals associated with it), who inserts the data entry into the map, can access the data. Here the access means read, update, invalidate, and remove.

If this constant is used, both map authorization and access by creator only authorization will take effect. Therefore, you can further limit the operations to the data entries. For example, you can restrict the creator from invalidating the data entries.

Since:

WAS XD 6.1 FIX3

See Also:

[Constant Field Values](#)

ACCESS_BY_CREATOR_ONLY_SUPERSEDE

```
public static final int ACCESS_BY_CREATOR_ONLY_SUPERSEDE
```

The access by creator only authorization is enabled to supersede the ObjectGrid map authorization.

The access by creator authorization ensures that only the user (represented by the Principals associated with it), who inserts the data entry into the map, can access the data. Here the access means read, update, invalidate, and remove.

If this constant is used, the access by creator only authorization will supersede the map authorization; no map authorization will be done.

Since:

WAS XD 6.1 FIX3

See Also:

[Constant Field Values](#)

Constructor Detail

SecurityConstants

```
public SecurityConstants()
```

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Class	Next Class	Frames	No Frames	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

com.ibm.websphere.objectgrid.security

Class ObjectGridSecurityException

[java.lang.Object](#)

[java.lang.Throwable](#)

[java.lang.Exception](#)

[com.ibm.websphere.objectgrid.ObjectGridException](#)

[com.ibm.websphere.objectgrid.security.ObjectGridSecurityException](#)

All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[CannotGenerateCredentialException](#), [ExpiredCredentialException](#),
[InvalidCredentialException](#)

```
public class ObjectGridSecurityException  
extends ObjectGridException
```

This exception represents a general ObjectGrid security exception.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridSecurityException](#)()

Constructs a new ObjectGridSecurityException with null as its detail message.

[ObjectGridSecurityException](#)(String message)

Constructs a new ObjectGridSecurityException with the specified detail message.

[ObjectGridSecurityException](#)(String message, [Throwable](#) cause)

Constructs a new ObjectGridSecurityException with the specified detail message and cause.

[ObjectGridSecurityException](#)([Throwable](#) cause)

Constructs a new ObjectGridSecurityException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#),
[printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridSecurityException

```
public ObjectGridSecurityException()
```

Constructs a new `ObjectGridSecurityException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ObjectGridSecurityException

```
public ObjectGridSecurityException(String message)
```

Constructs a new `ObjectGridSecurityException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridSecurityException

```
public ObjectGridSecurityException(String message,
                                   Throwable cause)
```

Constructs a new `ObjectGridSecurityException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ObjectGridSecurityException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ObjectGridSecurityException

```
public ObjectGridSecurityException(Throwable cause)
```

Constructs a new `ObjectGridSecurityException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for

ObjectGridSecurityExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

PREV CLASS [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

Package com.ibm.websphere.objectgrid.spring

This package holds the Spring specific APIs for ObjectGrid.

See:

[Description](#)

Interface Summary

SpringLocalTransactionManager	This interface has the methods for interacting with the ObjectGrid Spring LocalTransactionManager.
---	--

Class Summary

ObjectGridCache	This class is a WebSphere eXtreme Scale implementation of the Spring Framework's Cache interface.
ObjectGridCatalogServiceDomainBean	This class is a Spring bean representing a Catalog Service Domain for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.
ObjectGridClientBean	This class is a Spring bean representing an ObjectGrid client instance for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.
ObjectGridSpringFactory	This class serves as a factory to construct instances of the various Spring specific APIs.

Exception Summary

CannotGetObjectGridSessionException	This can be thrown by getSession if a new Session cannot be obtained for any reason.
ObjectGridTransactionException	The ObjectGrid platform transaction manager can throw this unchecked exception whenever errors occur.

Package com.ibm.websphere.objectgrid.spring Description

This package holds the Spring specific APIs for ObjectGrid.

Local Transaction Support

ObjectGrid has implemented a Spring PlatformTransactionManager. This allows Spring to manage local transactions using a single ObjectGrid session. Spring can then be used to annotate POJOs with container managed transaction semantics much like a J2EE application server does using J2EE CMT. An application should instantiate a SpringLocalTxManager using the appropriate factory method on ObjectGridSpringFactory and then wire a reference to that object in to all POJOs that use Spring CMT. This instance has a getSession method to obtain the correct Session for that POJO. The application must call one of the SpringLocalTxManager#setObjectGridForThread methods before invoking a managed POJO to specify which ObjectGrid instance should be used for any CMT on this thread.

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV PACKAGE](#) [NEXT PACKAGE](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.spring

Interface SpringLocalTxManager

public interface **SpringLocalTxManager**

This interface has the methods for interacting with the ObjectGrid Spring LocalTransactionManager. It also allows the desired partition to use with this thread to be specified.

Since:

WAS XD 6.1 FIX3, XC10

See Also:

[ObjectGridSpringFactory.getLocalPlatformTransactionManager\(\)](#)

Method Summary

S e s s i o n	getSession() This returns a managed session for the ObjectGrid associated with this thread.
V o i d	setObjectGridForThread(ObjectGrid grid) This indicates the ObjectGrid to use on this thread when a session is requested.

Method Detail

setObjectGridForThread

void **setObjectGridForThread**([ObjectGrid](#) grid)

This indicates the ObjectGrid to use on this thread when a session is requested. This replaces any previously associated ObjectGrid, i.e. only a single grid instance can be associated with a thread at a time.

Parameters:

grid - the ObjectGrid to set on this thread.

See Also:

[getSession\(\)](#)

getSession

[Session](#) **getSession()**

This returns a managed session for the ObjectGrid associated with this thread.

Do not call `begin()`, `commit()` or `rollback()` directly on the session. Spring manages the transaction automatically.

Returns:

A managed Session to use with this thread.

Throws:

[CannotGetObjectGridSessionException](#) - thrown when an ObjectGrid session can't be retrieved.

See Also:

[setObjectGridForThread\(ObjectGrid\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH		DETAIL: FIELD CONSTR METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid.spring

Class ObjectGridTransactionException

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public class ObjectGridTransactionException
extends TransactionException
```

The ObjectGrid platform transaction manager can throw this unchecked exception whenever errors occur.

Since:

WAS XD 6.1 FIX3, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridTransactionException](#)([String](#) message)

Constructs a new ObjectGridTransactionException with the specified detail message.

[ObjectGridTransactionException](#)([String](#) message, [Throwable](#) cause)

Constructs a new ObjectGridTransactionException with the specified detail message and cause.

Method Summary

Methods inherited from class org.springframework.core.[NestedRuntimeException](#)

[contains](#), [getMessage](#), [getMostSpecificCause](#), [getRootCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getCause](#), [getLocalizedMessage](#), [getStackTrace](#), [initCause](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridTransactionException

```
public ObjectGridTransactionException(String message,  
                                     Throwable cause)
```

Constructs a new ObjectGridTransactionException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ObjectGridTransactionException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[Throwable.getCause\(\)](#), [NestedRuntimeException.getMessage\(\)](#)

ObjectGridTransactionException

```
public ObjectGridTransactionException(String message)
```

Constructs a new ObjectGridTransactionException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[Throwable.initCause\(Throwable\)](#), [NestedRuntimeException.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.spring

Class ObjectGridSpringFactory

[java.lang.Object](#)



```
public final class ObjectGridSpringFactory
extends Object
```

This class serves as a factory to construct instances of the various Spring specific APIs.

Since:

WAS XD 6.1 FIX3, XC10

Field Summary

s t a t i c	SCOPE_SHARD Scope identifier for shard scope: "shard".
----------------------------	---

Constructor Summary

ObjectGridSpringFactory()

Method Summary

s t a t i c B e a n F a c t o r y	getBeanFactoryForObjectGrid(String objectGridName) This returns the currently registered external bean factory for a named object grid.
s t a	

t
i
c
O
b
j
e
c
t

[getBeanInShardScope](#)([ObjectGrid](#) shard, [String](#) beanName)

This returns an instance of the named Spring bean with the current shard scope using the specified ObjectGrid instance.

s
t
a
t
i
c
S
p
r
i
n
g
L
o
c
a
l
T
r
a
n
s
a
c
t
i
o
n
M
a
n
a
g
e
r

[getLocalPlatformTransactionManager](#)()

This returns an ObjectGrid PlatformLocalTransactionManager.

s
t
a
t
i
c
v
o
i
d

[registerSpringBeanFactoryAdapter](#)([String](#) objectGridName, [Object](#) springBeanFactory)

This returns an adapter for a Spring based bean factory.

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Field Detail

SCOPE_SHARD

public static final [String](#) SCOPE_SHARD

Scope identifier for shard scope: "shard".

See Also:

[getBeanInShardScope\(ObjectGrid, String\)](#), [Constant Field Values](#)

Constructor Detail

ObjectGridSpringFactory

public [ObjectGridSpringFactory](#)()

Method Detail

getLocalPlatformTransactionManager

```
public static SpringLocalTxManager getLocalPlatformTransactionManager()
```

This returns an ObjectGrid PlatformLocalTransactionManager.

Returns:

the PlatformLocalTransactionManager instance.

registerSpringBeanFactoryAdapter

```
public static void registerSpringBeanFactoryAdapter(String objectGridName,  
                                                    Object springBeanFactory)  
    throws ClassCastException
```

This returns an adapter for a Spring based bean factory. We use an Object type here to avoid making ObjectGrid dependent on Spring classes being present. A ClassCastException exception is thrown if the supplied factory isn't a Spring BeanFactory instance.

Parameters:

objectGridName - the name of the ObjectGrid
springBeanFactory - A Spring BeanFactory instance.

Throws:

[ClassCastException](#) - thrown when the Object type is not a BeanFactory instance.

getBeanFactoryForObjectGrid

```
public static BeanFactory getBeanFactoryForObjectGrid(String objectGridName)
```

This returns the currently registered external bean factory for a named object grid. If no factory has been registered then it attempts to construct a Spring BeanFactory using the xml resource on the class path @ "/X_spring.xml" and /META-INF/X_spring.xml where X is the name of the ObjectGrid. If the xml file is on the class path then the ObjectGrid name MUST be a valid resource name.

Parameters:

objectGridName - The name of the ObjectGrid

Returns:

The BeanFactory instance or null if there were none registered

getBeanInShardScope

```
public static Object getBeanInShardScope(ObjectGrid shard,  
                                          String beanName)
```

This returns an instance of the named Spring bean with the current shard scope using the specified ObjectGrid instance. This allows shard scoped beans to be obtained.

Parameters:

shard - The ObjectGrid instance to use to scope Spring beans using "shard" as scope.
beanName - The bean to return

Returns:

The bean instance if it exists

See Also:

[SCOPE_SHARD](#)

com.ibm.websphere.objectgrid.spring
Class ObjectGridClientBean

[java.lang.Object](#)



All Implemented Interfaces:
[InitializingBean](#)

```
public final class ObjectGridClientBean
extends Object
implements InitializingBean
```

This class is a Spring bean representing an ObjectGrid client instance for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.

Users must provide a [ObjectGridCatalogServiceDomainBean](#) to configure this client. The ObjectGrid name is optional when using the provided XML configuration files.

Since:
8.5, XC10

Constructor Summary

[ObjectGridClientBean\(\)](#)

Method Summary

V	afterPropertiesSet() Initializes the client bean.
V	setCatalogServiceDomain(ObjectGridCatalogServiceDomainBean catalogServiceDomain) Sets the ObjectGridCatalogServiceDomainBean used by the client.
V	setObjectGridName(String objectGridName) Sets the name of the ObjectGrid for the client.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridClientBean

```
public ObjectGridClientBean()
```

Method Detail

setObjectGridName

```
public void setObjectGridName(String objectGridName)
```

Sets the name of the ObjectGrid for the client. This is optional when using the provided XML configuration files.

Parameters:

objectGridName - The name of the ObjectGrid to connect to.

setCatalogServiceDomain

```
public void setCatalogServiceDomain(ObjectGridCatalogServiceDomainBean catalogServiceDomain)
```

Sets the [ObjectGridCatalogServiceDomainBean](#) used by the client.

Parameters:

catalogServiceDomain - The [ObjectGridCatalogServiceDomainBean](#) for the client.

afterPropertiesSet

```
public void afterPropertiesSet()  
    throws Exception
```

Initializes the client bean. This method is to be called after all setter methods have been called and will throw an [IllegalArgumentException](#) if [setCatalogServiceDomain\(ObjectGridCatalogServiceDomainBean\)](#) has not been called prior.

Specified by:

[afterPropertiesSet](#) in interface [InitializingBean](#)

Throws:

[Exception](#) - if [setCatalogServiceDomain\(ObjectGridCatalogServiceDomainBean\)](#) has not been called prior or an error occurs retrieving the ObjectGrid.

See Also:

[InitializingBean.afterPropertiesSet\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid.spring

Class ObjectGridCatalogServiceDomainBean

[java.lang.Object](#)



All Implemented Interfaces:

[InitializingBean](#)

```

public final class ObjectGridCatalogServiceDomainBean
extends Object
implements InitializingBean
  
```

This class is a Spring bean representing a Catalog Service Domain for use with the WebSphere eXtreme Scale implementation of Spring's cache abstraction.

Users must provide the catalog service endpoints used to connect to the eXtreme Scale cluster/domain. Users optionally may provide a client override XML and/or a client security configuration.

Since:

8.5, XC10

Constructor Summary

[ObjectGridCatalogServiceDomainBean\(\)](#)

Method Summary

V [afterPropertiesSet\(\)](#)
Initializes the connection to the Catalog Service Domain.

V [setCatalogServiceEndpoints\(String catalogServiceEndpoints\)](#)
Sets the catalog service endpoints used to connect to the cluster/domain.

V [setClientOverrideXml\(Resource clientOverrideXml\)](#)
Sets the location of the client override XML.

V [setClientSecurityConfig\(Resource clientSecurityConfiguration\)](#)
Sets the location of the client security configuration.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridCatalogServiceDomainBean

```
public ObjectGridCatalogServiceDomainBean()
```

Method Detail

setCatalogServiceEndpoints

```
public void setCatalogServiceEndpoints(String catalogServiceEndpoints)
```

Sets the catalog service endpoints used to connect to the cluster/domain.

Parameters:

catalogServiceEndpoints - The catalog service endpoints to connect to the cluster/domain

setClientOverrideXml

```
public void setClientOverrideXml(Resource clientOverrideXml)
```

Sets the location of the client override XML.

Parameters:

clientOverrideXml - The location of the client override XML.

setClientSecurityConfig

```
public void setClientSecurityConfig(Resource clientSecurityConfiguration)
```

Sets the location of the client security configuration.

Parameters:

clientSecurityConfiguration - The location of the client security configuration.

afterPropertiesSet

```
public void afterPropertiesSet()  
    throws Exception
```

Initializes the connection to the Catalog Service Domain. This method is to be called after all setter methods have been called and will throw an [IllegalArgumentException](#) if [setCatalogServiceEndpoints\(String\)](#) have not been called prior.

Specified by:

[afterPropertiesSet](#) in interface [InitializingBean](#)

Throws:

[Exception](#) - if [setCatalogServiceEndpoints\(String\)](#) have not been called prior or an error occurs initializing the connection.

See Also:

[InitializingBean.afterPropertiesSet\(\)](#)

com.ibm.websphere.objectgrid.spring

Class ObjectGridCache

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.spring.ObjectGridCache

All Implemented Interfaces:

[InitializingBean](#), [Cache](#)

```
public final class ObjectGridCache
extends Object
implements Cache, InitializingBean
```

This class is a WebSphere eXtreme Scale implementation of the Spring Framework's [Cache](#) interface.

Users must provide a name and a [ObjectGridClientBean](#) to configure this cache. The ObjectMap name is optional when using the provided XML configuration files.

This implementation allows for the storage of null values and does not support null keys.

The following Spring Inversion of Control (IoC) container configuration snippet creates two caches, named default and books hosted by the catalog service domain with connection endpoints of host1:2809,host2:2809.

```
<bean id="wxsCSDomain" class="com.ibm.websphere.objectgrid.spring.ObjectGridCatalogServiceDomainBean"
  p:catalog-service-endpoints="host1:2809,host2:2809" />

<bean id="wxsGridClient" class="com.ibm.websphere.objectgrid.spring.ObjectGridClientBean"
  p:catalog-service-domain-ref="wxsCSDomain" />

<bean id="cacheManager" class="org.springframework.cache.support.SimpleCacheManager">
  <property name="caches">
    <set>
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="default"
        p:object-grid-client-ref="wxsGridClient" />
      <bean class="com.ibm.websphere.objectgrid.spring.ObjectGridCache"
        p:name="books"
        p:object-grid-client-ref="wxsGridClient" />
    </set>
  </property>
</bean>
```

Since:

8.5, XC10

Nested Class Summary

Nested classes/interfaces inherited from interface

[org.springframework.cache.Cache](#)

[Cache.ValueWrapper](#)

Constructor Summary

[ObjectGridCache\(\)](#)

Method Summary

void [afterPropertiesSet\(\)](#)
Initializes this cache.

void [clear\(\)](#)
Clears all entries from the cache.

void [evict\(Object key\)](#)
Evicts the entry at the given key.

Cache.ValueWrapper [get\(Object key\)](#)
Retrieves the object from the cache at the given key.

String [getName\(\)](#)
Returns the name of the cache.

Object [getNativeCache\(\)](#)
This implementation returns null.

void [put\(Object key, Object value\)](#)
Creates an entry in the cache.

void [setMapName\(String mapName\)](#)
Sets the ObjectMap name.

`void setName(String name)`
Sets the cache name.

`void setObjectClient(ObjectGridClientBean objectGridClient)`
Sets the [ObjectGridClientBean](#) to use.

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridCache

```
public ObjectGridCache()
```

Method Detail

afterPropertiesSet

```
public void afterPropertiesSet()  
           throws Exception
```

Initializes this cache. This method is to be called after all setter methods have been called and will throw an [IllegalArgumentException](#) if [setObjectGridClient\(ObjectGridClientBean\)](#) or [setName\(String\)](#) have not been called prior.

Specified by:

[afterPropertiesSet](#) in interface [InitializingBean](#)

Throws:

[Exception](#) - if [setObjectGridClient\(ObjectGridClientBean\)](#) or [setName\(String\)](#) have not been called prior or an error occurs initializing the cache.

See Also:

[InitializingBean.afterPropertiesSet\(\)](#)

getName

```
public String getName()
```

Returns the name of the cache.

Specified by:

[getName](#) in interface [Cache](#)

Returns:

The name of the cache.

See Also:

[Cache.getName\(\)](#)

setName

```
public void setName(String name)
```

Sets the cache name.

Parameters:

name - The cache name.

See Also:

[getName\(\)](#)

setMapName

```
public void setMapName(String mapName)
```

Sets the ObjectMap name. This is optional when using the provided XML configuration files.

Parameters:

mapName - The name of the ObjectMap

setObjectGridClient

```
public void setObjectGridClient(ObjectGridClientBean objectGridClient)
```

Sets the [ObjectGridClientBean](#) to use.

Parameters:

objectGridClient - The [ObjectGridClientBean](#) to use

getNativeCache

```
public Object getNativeCache()
```

This implementation returns null.

Specified by:

[getNativeCache](#) in interface [Cache](#)

Returns:

This implementation returns null.

See Also:

[Cache.getNativeCache\(\)](#)

get

```
public Cache.ValueWrapper get(Object key)
```

Retrieves the object from the cache at the given key. Returns null if there is no mapping for the key or if the key is null

Specified by:

[get](#) in interface [Cache](#)

Returns:

the object from the cache at the given key or null if there is no mapping or the key is null

See Also:

[Cache.get\(java.lang.Object\)](#)

put

```
public void put(Object key,  
               Object value)
```


Creates an entry in the cache. Overwrites the value if an entry for the given key exists. This method does not create an entry if the given key is null.

Specified by:

[put](#) in interface [Cache](#)

See Also:

[Cache.put\(java.lang.Object, java.lang.Object\)](#)

evict

public void **evict**([Object](#) key)

Evicts the entry at the given key. If the provided key is null this method does not evict any entries.

Specified by:

[evict](#) in interface [Cache](#)

See Also:

[Cache.evict\(java.lang.Object\)](#)

clear

public void **clear**()

Clears all entries from the cache.

Specified by:

[clear](#) in interface [Cache](#)

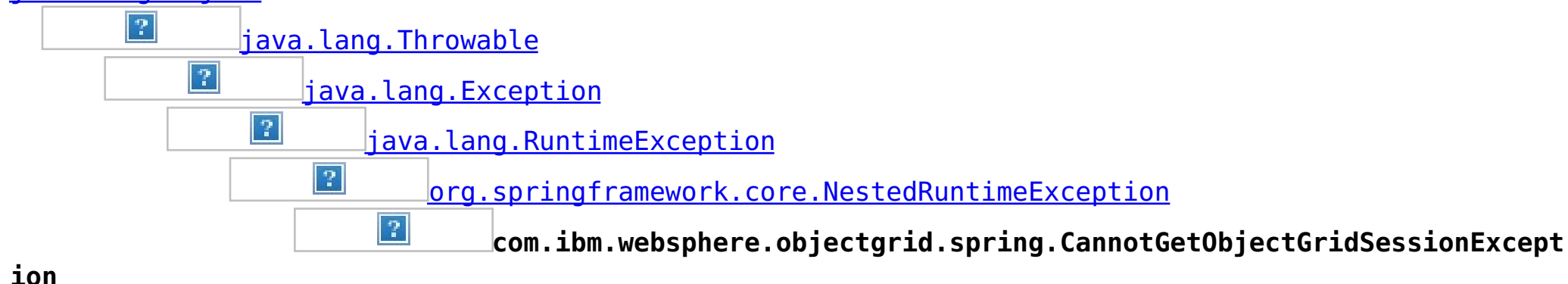
See Also:

[Cache.clear\(\)](#)

com.ibm.websphere.objectgrid.spring

Class CannotGetObjectGridSessionException

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public class CannotGetObjectGridSessionException
extends NestedRuntimeException
```

This can be thrown by getSession if a new Session cannot be obtained for any reason.

Since:

WAS XD 6.1 FIX3, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[CannotGetObjectGridSessionException](#)([String](#) message)

Constructs a new CannotGetObjectGridSessionException with the specified detail message.

[CannotGetObjectGridSessionException](#)([String](#) message, [Throwable](#) cause)

Constructs a new CannotGetObjectGridSessionException with the specified detail message and cause.

Method Summary

Methods inherited from class [org.springframework.core.NestedRuntimeException](#)

[contains](#), [getMessage](#), [getMostSpecificCause](#), [getRootCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getCause](#), [getLocalizedMessage](#), [getStackTrace](#), [initCause](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

CannotGetObjectGridSessionException

```
public CannotGetObjectGridSessionException(String message,  
                                           Throwable cause)
```

Constructs a new `CannotGetObjectGridSessionException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `CannotGetObjectGridSessionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[Throwable.getCause\(\)](#), [NestedRuntimeException.getMessage\(\)](#)

CannotGetObjectGridSessionException

```
public CannotGetObjectGridSessionException(String message)
```

Constructs a new `CannotGetObjectGridSessionException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[Throwable.initCause\(Throwable\)](#), [NestedRuntimeException.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

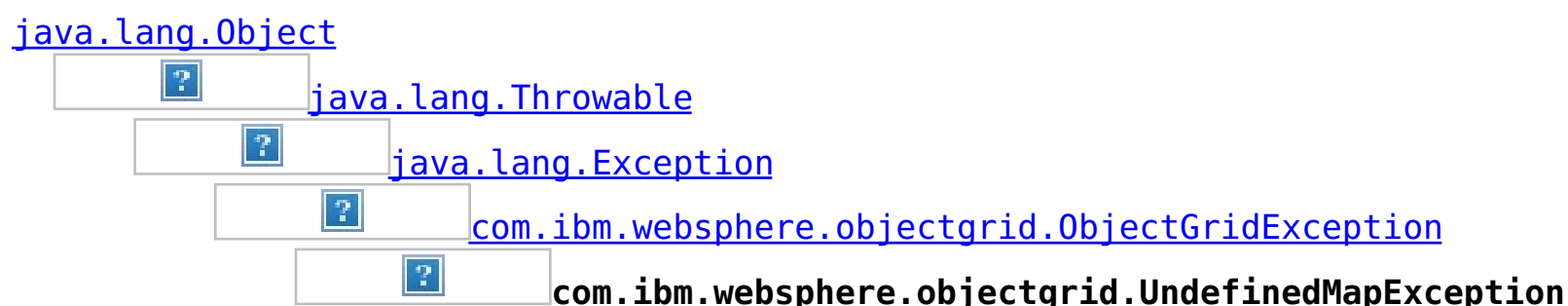
PREV CLASS [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid
Class UndefinedMapException



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

```
public class UndefinedMapException
extends ObjectGridException
```

This exception indicates that the map which an application tries to access is not defined in the ObjectGrid.

Since:
WAS XD 6.0, XC10

See Also:
[Serialized Form](#)

Constructor Summary	
UndefinedMapException ()	Constructs a new UndefinedMapException with null as its detail message.
UndefinedMapException (String message)	Constructs a new UndefinedMapException with the specified detail message.
UndefinedMapException (String message, Throwable cause)	Constructs a new UndefinedMapException with the specified detail message and cause.
UndefinedMapException (Throwable cause)	Constructs a new UndefinedMapException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid. ObjectGridException
getCause , initCause

Methods inherited from class java.lang. Throwable
fillInStackTrace , getLocalizedMessage , getMessage , getStackTrace , printStackTrace , printStackTrace , printStackTrace , setStackTrace , toString

Methods inherited from class java.lang. Object
--

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

UndefinedMapException

```
public UndefinedMapException()
```

Constructs a new UndefinedMapException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

UndefinedMapException

```
public UndefinedMapException(String message)
```

Constructs a new UndefinedMapException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

UndefinedMapException

```
public UndefinedMapException(String message,
                             Throwable cause)
```

Constructs a new UndefinedMapException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this UndefinedMapException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

UndefinedMapException

```
public UndefinedMapException(Throwable cause)
```

Constructs a new UndefinedMapException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for UndefinedMapExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that

the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

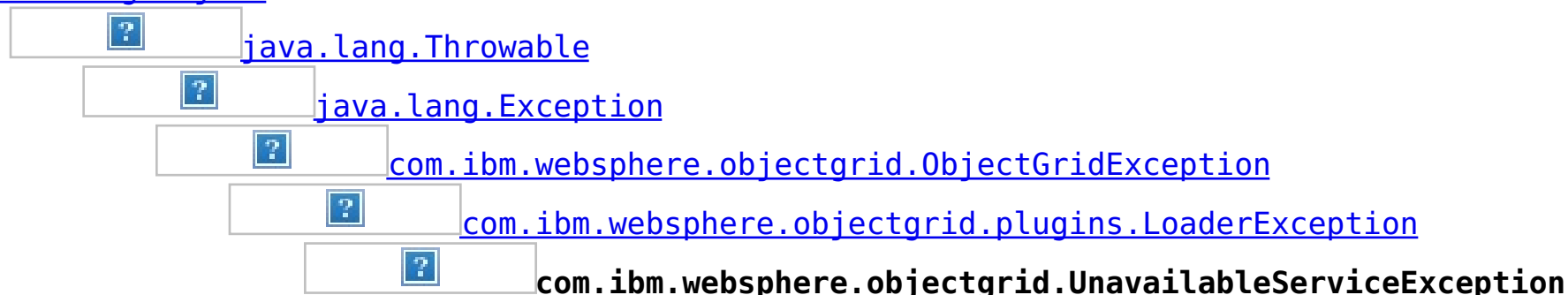
**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class UnavailableServiceException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[TransactionAffinityException](#), [TransactionQuiesceException](#)

```
public class UnavailableServiceException
extends LoaderException
```

This exception is thrown when all servers are dead or when all services are unavailable even though servers are running.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[UnavailableServiceException\(\)](#)

Constructs a new UnavailableServiceException with null as its detail message.

[UnavailableServiceException\(String message\)](#)

Constructs a new UnavailableServiceException with the specified detail message.

[UnavailableServiceException\(String message, Throwable cause\)](#)

Constructs a new UnavailableServiceException with the specified detail message and cause.

[UnavailableServiceException\(Throwable cause\)](#)

Constructs a new UnavailableServiceException with a specified cause.

Method Summary

[getReplicationGroup\(\)](#)

Returns the replication group identifier for this exception.

[setReplicationGroup\(int replicationGroup\)](#)

Sets the replication group identifier for this exception.

Methods inherited from class [com.ibm.websphere.objectgrid.ObjectGridException](#)[getCause](#), [initCause](#)**Methods inherited from class [java.lang.Throwable](#)**[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)**Methods inherited from class [java.lang.Object](#)**[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

UnavailableServiceException

```
public UnavailableServiceException()
```

Constructs a new `UnavailableServiceException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:[ObjectGridException.initCause\(Throwable\)](#)

UnavailableServiceException

```
public UnavailableServiceException(String message)
```

Constructs a new `UnavailableServiceException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

UnavailableServiceException

```
public UnavailableServiceException(String message,  
                                  Throwable cause)
```

Constructs a new `UnavailableServiceException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `UnavailableServiceException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

UnavailableServiceException

public **UnavailableServiceException**([Throwable](#) cause)

Constructs a new UnavailableServiceException with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for UnavailableServiceExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Method Detail

getReplicationGroup

public int **getReplicationGroup**()

Returns the replication group identifier for this exception.

Returns:

the argument that was passed to the `setReplicationGroup(int)` method of this class or 0 if the `setReplicationGroup` method was not previously called for this object.

See Also:

[setReplicationGroup\(int\)](#)

setReplicationGroup

public void **setReplicationGroup**(int replicationGroup)

Sets the replication group identifier for this exception.

Parameters:

replicationGroup - The replication group identifier

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface TxID

All Superinterfaces:

[Serializable](#)

```
public interface TxID
extends Serializable
```

This interface is an opaque identifier for a transaction. Context information can be stored and retrieved in multiple slots on this object. This mechanism allows a TransactionCallback and Loader, for example, to share state information with each other in the context of a specific session transaction.

The TxID.toString() output can be used to determine whether the originating Session transaction is a single partition transaction or a multi-partition transaction. If the String output begins with the keyword Local then this indicates a single partition transaction, for example: Local-40000139-72B2-C037-E000-1C271366B073

If the String output begins with the keyword WXS then this indicates a multi-partition transaction, for example: WXS-40000139-72B2-BD3A-E000-1C271366B073

Since:

WAS XD 6.0, XC10

See Also:

Loader, [ObjectGrid.reserveSlot\(String\)](#), [Session](#), [TransactionCallback](#)

Field Summary

s t a t i c S t r i n g	<p>SLOT_NAME</p> <p>All slots should be reserved using this name.</p>
--	---

Method Summary

b o o l e a n	<p>equals(TxID o)</p> <p>Checks for equality between two TxID objects.</p>
S e	

S S i o n	getSession() Returns the Session that owns this TxID.
O b j e c t	getSlot(int slotNumber) Gets the context information currently associated with this transaction.
i n t	hashCode() Returns the hashcode of the Tx identifier.
V o i d	putSlot(int slotNumber, Object o) Sets some context information to be associated with this transaction.

Field Detail

SLOT_NAME

static final [String](#) SLOT_NAME

All slots should be reserved using this name.

See Also:

[ObjectGrid.reserveSlot\(String\)](#), [Constant Field Values](#)

Method Detail

equals

boolean **equals**([TxID](#) o)

Checks for equality between two TxID objects.

Parameters:

o - Input TxID to check for equality against

Returns:

true, if they are equal; false, if they not equal

hashCode

int **hashCode**()

Returns the hashcode of the Tx identifier.

Overrides:

[hashCode](#) in class [Object](#)

Returns:

hashcode

getSlot

[Object](#) `getSlot(int slotNumber)`

Gets the context information currently associated with this transaction.

Parameters:

slotNumber - the slot number for the context information being requested

Returns:

Object the current context information for the slot number

See Also:

[putSlot\(int, Object\)](#), [ObjectGrid.reserveSlot\(String\)](#)

putSlot

void `putSlot(int slotNumber,`
`Object o)`

Sets some context information to be associated with this transaction.

Parameters:

slotNumber - the slot number

o - Object to be put into the TxID slot

See Also:

[getSlot\(int\)](#), [ObjectGrid.reserveSlot\(String\)](#)

getSession

[Session](#) `getSession()`

Returns the Session that owns this TxID.

Returns:

a Session object to use.

See Also:

[Session](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class TransactionTimeoutException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionTimeoutException
extends ObjectGridRuntimeException
```

This exception is thrown when a transaction exceeds the transaction timeout that was specified on the ObjectGrid or Session.

Since:

WAS XD 6.0.1, XC10

See Also:

[ObjectGrid.setTxTimeout\(int\)](#), [Session.setTransactionTimeout\(int\)](#), [Serialized Form](#)

Constructor Summary

- [TransactionTimeoutException](#)([String](#) message, [String](#) txIdString)
Constructs a new TransactionTimeoutException with the specified detail message.
- [TransactionTimeoutException](#)([String](#) message, [Throwable](#) cause)

Method Summary

- | | |
|----------------------------|---|
| S
t
r
i
n
g | getTxIDString ()
Get the String representation of the TXID for the transaction that timed out. |
| D
a
t
e | whenOccurred ()
Gives the time when this TransactionTimeoutException was created. |

Methods inherited from class

com.ibm.websphere.objectgrid.[ObjectGridRuntimeException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionTimeoutException

```
public TransactionTimeoutException(String message,  
                                   Throwable cause)
```

TransactionTimeoutException

```
public TransactionTimeoutException(String message,  
                                   String txIdString)
```

Constructs a new TransactionTimeoutException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

`txIdString` - the result of `TxID.toString()` for the transaction that timed out.

See Also:

[ObjectGridRuntimeException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

Method Detail

whenOccurred

```
public Date whenOccurred()
```

Gives the time when this TransactionTimeoutException was created.

Returns:

Date object that represents the instant in time when this exception object was created.

getTxIDString

```
public String getTxIDString()
```

Get the String representation of the TxID for the transaction that timed out.

Returns:

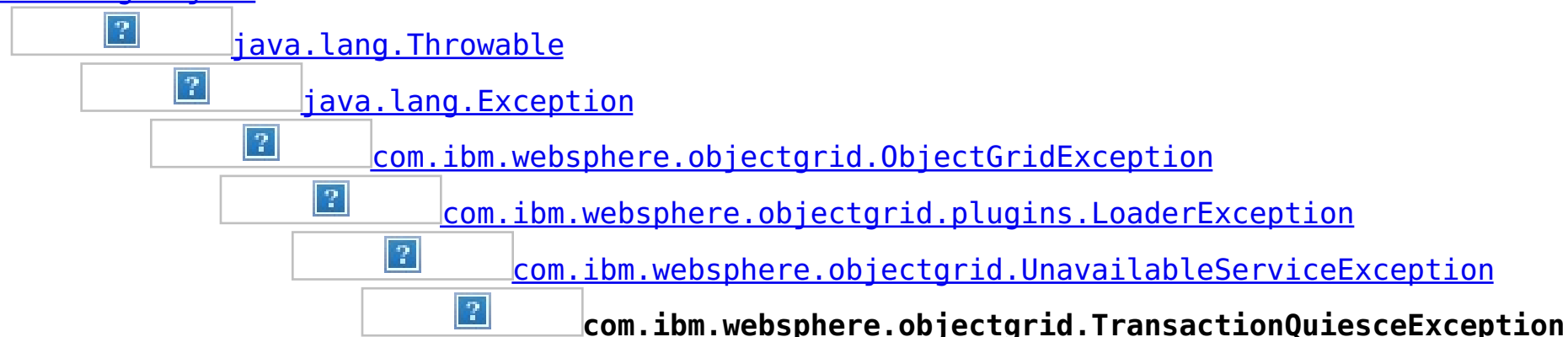
String value of TxID of transaction that timed out.

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class TransactionQuiesceException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionQuiesceException
extends UnavailableServiceException
```

This exception is thrown when partition/shard/mapset/replication group/ replication group member/server/cluster/objectgrid is entered quiesce process for various reasons such as shard movement, partition relocation, system update, server shutdown, and others. Quiesce process ensures data integrity and transaction integrity. This exception is thrown only for new start of a new transaction; it will not impact old transaction requests that are allowed to finish.

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[TransactionQuiesceException](#)()

Constructs a new TransactionQuiesceException with null as its detail message.

[TransactionQuiesceException](#)(String message)

Constructs a new TransactionQuiesceException with the specified detail message.

Method Summary

Methods inherited from class

com.ibm.websphere.objectgrid.[UnavailableServiceException](#)

[getReplicationGroup](#), [setReplicationGroup](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionQuiesceException

```
public TransactionQuiesceException()
```

Constructs a new TransactionQuiesceException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionQuiesceException

```
public TransactionQuiesceException(String message)
```

Constructs a new TransactionQuiesceException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

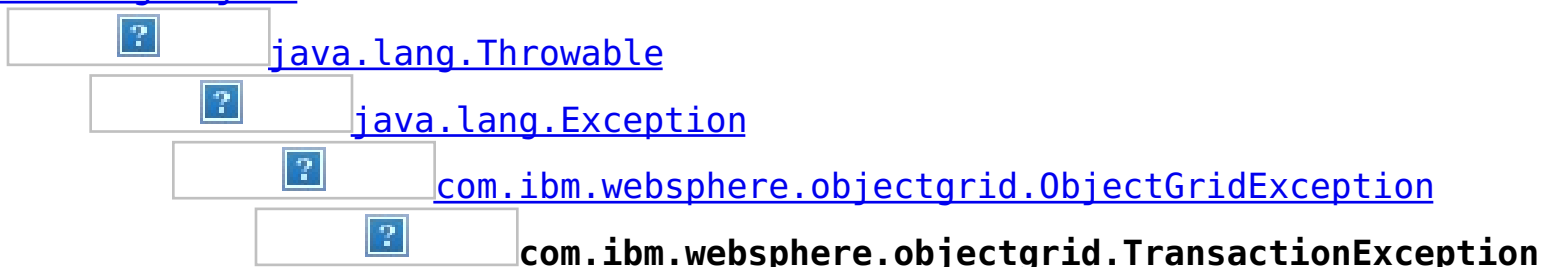
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class TransactionException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[SessionNotReentrantException](#), [TransactionAlreadyActiveException](#)

```
public class TransactionException
extends ObjectGridException
```

A general transaction exception indicating something went wrong with a transaction. The `isTransactionActive()` and `wasTransactionRolledBack()` methods can be used to determine whether transaction is still active or was rolled back as a result of this exception occurring.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Field Summary

protected

[ivTransactionRolledBack](#)

Indicates whether the transaction was rolled back or not.

Constructor Summary

[TransactionException](#)([String](#) message, boolean rolledBack)

Constructs a new TransactionException with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([String](#) message, [Throwable](#) cause, boolean rolledBack)

Constructs a new TransactionException with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([String](#) message, [TransactionException](#) cause, boolean rolledBack)

Constructs a new TransactionException with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([Throwable](#) cause, boolean rolledBack)

Constructs a new TransactionException with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception.

[TransactionException](#)([TransactionException](#) cause, boolean rolledBack)

Constructs a new TransactionException with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception.

Method Summary

[boolean](#) [isTransactionActive](#)()
Returns true if the transaction is active.

[boolean](#) [wasTransactionRolledBack](#)()
Returns true if the transaction was rolled back.

Methods inherited from class [com.ibm.websphere.objectgrid.ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

ivTransactionRolledBack

protected boolean **ivTransactionRolledBack**

Indicates whether the transaction was rolled back or not.

Constructor Detail

TransactionException

```
public TransactionException(String message,  
                             boolean rolledBack)
```

Constructs a new `TransactionException` with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.
`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(Throwable cause,  
                           boolean rolledBack)
```

Constructs a new `TransactionException` with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for as a wrapper for other `Throwable` objects that occur.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.
`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.getCause\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(TransactionException cause,  
                           boolean rolledBack)
```

Constructs a new `TransactionException` with a specified cause and a specified indication of whether the transaction was rolled back as a result of this exception. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for as a wrapper for other `Throwable` objects that occur.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.
`rolledBack` - A value of `true` indicates the transaction was rolled back.

Since:

WAS XD 6.1 IFIX1

See Also:

[ObjectGridException.getCause\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(String message,  
                           Throwable cause,  
                           boolean rolledBack)
```


Constructs a new `TransactionException` with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `TransactionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#), [wasTransactionRolledBack\(\)](#)

TransactionException

```
public TransactionException(String message,  
                           TransactionException cause,  
                           boolean rolledBack)
```

Constructs a new `TransactionException` with the specified detail message, cause, and indication of whether the transaction was rolled back as a result of this exception.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `TransactionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

`rolledBack` - A value of `true` indicates the transaction was rolled back.

Since:

WAS XD 6.1 IFIX1

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#), [wasTransactionRolledBack\(\)](#)

Method Detail

isTransactionActive

```
public boolean isTransactionActive()
```

Returns `true` if the transaction is active. Otherwise, `false` is returned to indicate either the transaction never started or was completed.

Returns:

`true` if the transaction is active

wasTransactionRolledBack

```
public boolean wasTransactionRolledBack()
```

Returns `true` if the transaction was rolled back.

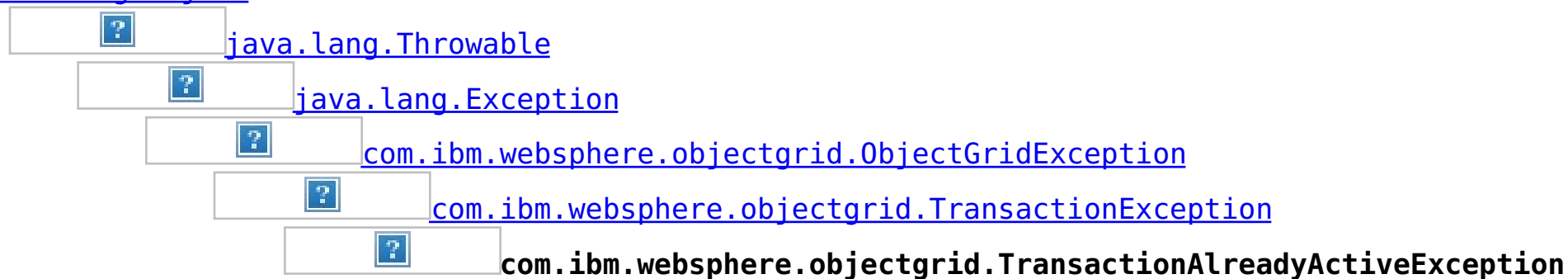
Returns:

`true` if the transaction was rolled back

com.ibm.websphere.objectgrid

Class TransactionAlreadyActiveException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionAlreadyActiveException
extends TransactionException
```

An exception indicating a transaction is already active for the current session. This exception does not cause the current active transaction to be rolled back, so the `isTransactionActive` method will return true.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Field Summary

Fields inherited from class `com.ibm.websphere.objectgrid.TransactionException`

[ivTransactionRolledBack](#)

Constructor Summary

[TransactionAlreadyActiveException](#)()

Constructs a new TransactionAlreadyActiveException with null as its detail message.

[TransactionAlreadyActiveException](#)(String message)

Constructs a new TransactionAlreadyActiveException with the specified detail message.

[TransactionAlreadyActiveException](#)(String message, [Throwable](#) cause)

Constructs a new TransactionAlreadyActiveException with the specified detail message and cause.

[TransactionAlreadyActiveException](#)([Throwable](#) cause)

Constructs a new TransactionAlreadyActiveException with a specified cause.

Method Summary

Methods inherited from class `com.ibm.websphere.objectgrid.TransactionException`

[isTransactionActive](#), [wasTransactionRolledBack](#)

Methods inherited from class `com.ibm.websphere.objectgrid.ObjectGridException`

[getCause](#), [initCause](#)

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException()
```

Constructs a new `TransactionAlreadyActiveException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException(String message)
```

Constructs a new `TransactionAlreadyActiveException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException(Throwable cause)
```

Constructs a new `TransactionAlreadyActiveException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `TransactionAlreadyActiveExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

TransactionAlreadyActiveException

```
public TransactionAlreadyActiveException(String message,  
                                         Throwable cause)
```

Constructs a new TransactionAlreadyActiveException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this TransactionAlreadyActiveException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
Prev Class	Next Class	Frames	No Frames	All				
			Classes					
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD					

com.ibm.websphere.objectgrid

Class TransactionAffinityException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class TransactionAffinityException
extends UnavailableServiceException
```

This exception is thrown for inflight transaction when server fails over. We suggest applications to retry transaction.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[TransactionAffinityException](#)()

Constructs a new TransactionAffinityException with null as its detail message.

[TransactionAffinityException](#)(String message)

Constructs a new TransactionAffinityException with the specified detail message.

[TransactionAffinityException](#)(String message, Throwable cause)

Constructs a new TransactionAffinityException with the specified detail message and cause.

[TransactionAffinityException](#)(Throwable cause)

Constructs a new TransactionAffinityException with a specified cause.

Method Summary

Methods inherited from class

com.ibm.websphere.objectgrid.[UnavailableServiceException](#)

[getReplicationGroup](#), [setReplicationGroup](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TransactionAffinityException

```
public TransactionAffinityException()
```

Constructs a new TransactionAffinityException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

TransactionAffinityException

```
public TransactionAffinityException(String message)
```

Constructs a new TransactionAffinityException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TransactionAffinityException

```
public TransactionAffinityException(String message,  
                                   Throwable cause)
```

Constructs a new TransactionAffinityException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this TransactionAffinityException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

TransactionAffinityException

```
public TransactionAffinityException(Throwable cause)
```

Constructs a new TransactionAffinityException with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for TransactionAffinityExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class TargetNotAvailableException

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public class TargetNotAvailableException
extends RuntimeException
```

A TargetNotAvailableException indicates the ObjectGrid target is not available. This could be due to the fact that ObjectGrid servers are not available or the ObjectGrid placement has not finished.

Since:

WAS XD 6.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[TargetNotAvailableException](#)()

Constructs a new TargetNotAvailableException with null as its detail message.

[TargetNotAvailableException](#)(String message)

Constructs a new TargetNotAvailableException with the specified detail message.

[TargetNotAvailableException](#)(String message, [Throwable](#) cause)

Constructs a new TargetNotAvailableException with the specified detail message and cause.

[TargetNotAvailableException](#)([Throwable](#) cause)

Constructs a new TargetNotAvailableException with a specified cause.

Method Summary

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getCause](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [initCause](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

TargetNotAvailableException

```
public TargetNotAvailableException()
```

Constructs a new TargetNotAvailableException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[Throwable.initCause\(Throwable\)](#)

TargetNotAvailableException

```
public TargetNotAvailableException(String message)
```

Constructs a new TargetNotAvailableException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[Throwable.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

TargetNotAvailableException

```
public TargetNotAvailableException(Throwable cause)
```

Constructs a new TargetNotAvailableException with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for TargetNotAvailableExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[Throwable.getCause\(\)](#)

TargetNotAvailableException

```
public TargetNotAvailableException(String message,  
                                   Throwable cause)
```

Constructs a new TargetNotAvailableException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this TargetNotAvailableException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[Throwable.getCause\(\)](#), [Throwable.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class TTLType

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```

public class TTLType
    extends Object
    implements Serializable
  
```

Every BackingMap in ObjectGrid has a built in timed based evictor that is referred to as "time to live" evictor or TTL evictor. Each BackingMap entry has an expiration time that determines how long the entry is allowed to live in the BackingMap. When the expiration time is reached, the TTL evictor causes the expired entry to be evicted from the BackingMap. This class is used to define the TTLType value constants that determine how the the expiration time is computed for a map entry.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.setTtlEvictorType\(TTLType\)](#), [Serialized Form](#)

Field Summary

s t a t i c	<p>CREATION_TIME</p> <p>A TTLType.CREATION_TIME indicates an entry expiration time is the sum of the creation time of the entry plus the "time to live" value.</p>
I T I L E V E	<p>LAST_ACCESS_TIME</p> <p>A TTLType.LAST_ACCESS_TIME indicates an entry expiration time is the sum of the last access time of the entry plus the "time to live" value.</p>
s t a	

t
i
c
T
T
L
T
Y
P
E

[LAST_UPDATE_TIME](#)

A `TTLType.LAST_UPDATE_TIME` indicates an entry expiration time is the sum of the last update time of the entry plus the "time to live" value.

s
t
a
t
i
c
T
T
L
T
Y
P
E

[NONE](#)

A `TTLType.NONE` indicates an entry has no expiration time and is allowed to live in the `BackingMap` until the application explicitly removes or invalidates the entry or a user defined evictor evicts it.

Method Summary

b
y
t
e

[getId\(\)](#)

Get the raw value of this `TTLType`.

S
t
r
i
n
g

[toString\(\)](#)

Returns a string representation of the `TTLType`.

s
t
a
t
i
c
T
T
L
T
Y
P
E

[valueOf\(byte id\)](#)

Given the raw value of a `TTLType`, this method returns a `TTLType` object, or null if the raw value does not match an existing type.

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

NONE

```
public static final TTLType NONE
```

A `TTLType.NONE` indicates an entry has no expiration time and is allowed to live in the `BackingMap` until the application explicitly removes or invalidates the entry or a user defined evictor evicts it.

CREATION_TIME

```
public static final TTLType CREATION_TIME
```

A `TTLType.CREATION_TIME` indicates an entry expiration time is the sum of the creation time of the entry plus the "time to live" value. The "time to live" value is set using the `BackingMap.setTimeToLive(int)` method and is the same for every entry and can **not** be changed by the application by using the `ObjectMap.setTimeToLive(int)` method. It can only be set prior to ObjectGrid initialization by use of the `BackingMap.setTimeToLive(int)` method.

See Also:

[BackingMap.setTimeToLive\(int\)](#)

LAST_ACCESS_TIME

```
public static final TTLType LAST_ACCESS_TIME
```

A `TTLType.LAST_ACCESS_TIME` indicates an entry expiration time is the sum of the last access time of the entry plus the "time to live" value. By default, the time to live value is set using the `BackingMap.setTimeToLive(int)` method and the default can be overridden by the application by using the `ObjectMap.setTimeToLive(int)` method.

See Also:

[BackingMap.setTimeToLive\(int\)](#), [ObjectMap.setTimeToLive\(int\)](#)

LAST_UPDATE_TIME

```
public static final TTLType LAST_UPDATE_TIME
```

A `TTLType.LAST_UPDATE_TIME` indicates an entry expiration time is the sum of the last update time of the entry plus the "time to live" value. By default, the time to live value is set using the `BackingMap.setTimeToLive(int)` method and the default can be overridden by the application by using the `ObjectMap.setTimeToLive(int)` method. The difference between this `TTLType` and `LAST_ACCESS_TIME` is that fetch operations do not cause the entry expiration time to be updated.

Since:

7.1

See Also:

[BackingMap.setTimeToLive\(int\)](#), [ObjectMap.setTimeToLive\(int\)](#)

Method Detail

valueOf

```
public static final TTLType valueOf(byte id)
```

Given the raw value of a `TTLType`, this method returns a `TTLType` object, or null if the raw value does not match an existing type. This method is used to deserialize this object.

Parameters:

id - the raw value of a `TTLType`

Returns:

the `TTLType` corresponding to the raw input value

Since:

8.6, XC10 2.5

getId

public byte **getId()**

Get the raw value of this TTLType. This method is used to serialize this object.

Returns:
the raw value of this TTLType.

Since:
8.6, XC10 2.5

toString

public [String](#) **toString()**

Returns a string representation of the TTLType.

Overrides:
[toString](#) in class [Object](#)

Returns:
a string representation of the TTLType.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface StateManager

public interface **StateManager**

The StateManager can be used to retrieve the availability state of an ObjectGrid. Use the StateManagerFactory.getStateManager() method to retrieve a StateManager instance.

Since:

WAS XD 6.1.0.3, XC10

Method Summary

AvailabilityState	getObjectGridState (ObjectGrid objectGrid) Get the AvailabilityState of an ObjectGrid.
ObjectGrid	setObjectGridState (AvailabilityState state, ObjectGrid objectGrid) Set the AvailabilityState for an ObjectGrid.

Method Detail

getObjectGridState

[AvailabilityState](#) getObjectGridState([ObjectGrid](#) objectGrid)

Get the AvailabilityState of an ObjectGrid. A random shard within the ObjectGrid is chosen for reporting availability state.

Parameters:

objectGrid - the availability state of the specified remote ObjectGrid will be retrieved

Returns:

the AvailabilityState of the remote ObjectGrid

Throws:

IllegalArgumentException. - If parameter objectGrid, is either null or it is of type 'LOCAL'. See [ObjectGrid.getObjectGridType\(\)](#).

[TargetNotAvailableException](#) - if there are no active shards for the ObjectGrid specified.

setObjectGridState

```
void setObjectGridState(AvailabilityState state,  
                        ObjectGrid objectGrid)
```

Set the AvailabilityState for an ObjectGrid. Each shard in the ObjectGrid will be transitioned to the state specified. This method does not return until each shard in the ObjectGrid has transitioned to the AvailabilityState specified or if it times-out.

Parameters:

state - the AvailabilityState to transition to.

objectGrid - the ObjectGrid to transaction to the specified AvailabilityState.

Throws:

IllegalArgumentException. -

1. If parameter ObjectGrid. is either null or is of type 'LOCAL'. See [ObjectGrid.getObjectGridType\(\)](#).
2. If parameter AvailabilityState is null.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

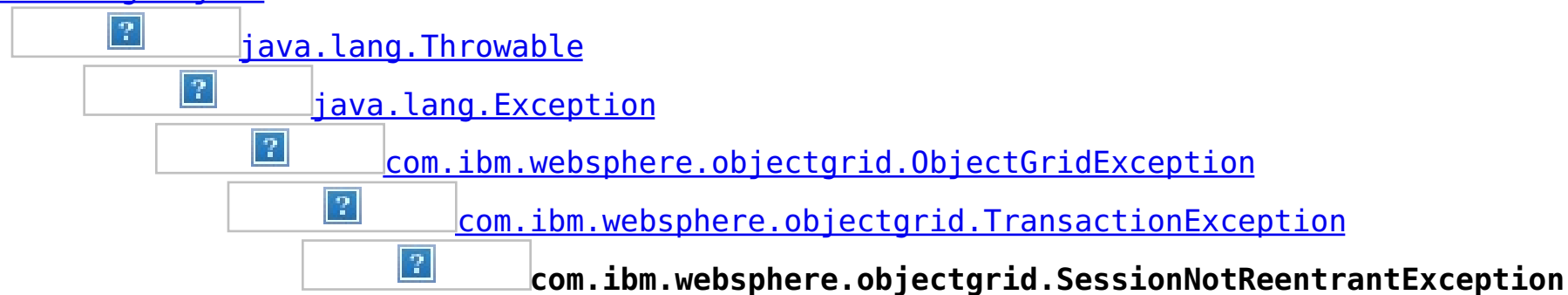
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class SessionNotReentrantException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class SessionNotReentrantException
extends TransactionException
```

A Session object can only be used by a single thread concurrently to perform map operations. If a thread tries to execute a map operation (for example, call a method on ObjectMap interface) while another thread is already executing a map operation for the Session, then this exception is thrown.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Field Summary

Fields inherited from class com.ibm.websphere.objectgrid.[TransactionException](#)

[ivTransactionRolledBack](#)

Constructor Summary

[SessionNotReentrantException](#)([String](#) message, boolean rolledBack)

Constructs a new SessionNotReentrantException with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[TransactionException](#)

[isTransactionActive](#), [wasTransactionRolledBack](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

SessionNotReentrantException

```
public SessionNotReentrantException(String message,  
                                   boolean rolledBack)
```

Constructs a new SessionNotReentrantException with the specified detail message and a special indication of whether the transaction was rolled back as a result of this exception. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

`rolledBack` - A value of `true` indicates the transaction was rolled back.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [TransactionException.wasTransactionRolledBack\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: NESTED | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid
Interface Session

public interface **Session**

This interface represents a session container for ObjectMaps. A thread must get its own Session object to interact with ObjectGrid. You can think of this interface as a session that can only be used by a single thread at a time. A Session itself is shareable across threads so long as only one thread uses it at a time. However, if a J2EE connection/transaction infrastructure is being used, that won't be shareable across threads and will prevent the Session object from being shared across threads. A good analogy for this object is a JDBC connection to a database. For best performance, use the [close\(\)](#) method to close the session once it is no longer required.

Since:

WAS XD 6.0, XC10

See Also:

[ObjectGrid.getSession\(\)](#), [ObjectGrid.getSession\(Subject\)](#),
[ObjectGrid.getSession\(CredentialGenerator\)](#)

Nested Class Summary

s
t
a
t
i
c
c
l
a
s
s

[Session.TxCommitProtocol](#)

The commit protocols that can be used to commit the Session's transaction

Field Summary

s
t
a
t
i
c
l
o
n
g

[DEFAULT_RETRY_TIMEOUT](#)

s
t
a
t
i
c
i
n
t

[TRANSACTION_NO_TIMEOUT](#)

A special value for the timeout parameter of the setTransactionTimeout(int) method.

s t a t i c i n t	<p>TRANSACTION_READ_COMMITTED</p> <p>A transaction isolation level constant indicating that dirty reads are prevented; non-repeatable reads and phantom reads can occur.</p>
s t a t i c i n t	<p>TRANSACTION_READ_UNCOMMITTED</p> <p>A transaction isolation level constant indicating that dirty reads, non-repeatable reads and phantom reads can occur.</p>
s t a t i c i n t	<p>TRANSACTION_REPEATABLE_READ</p> <p>A transaction isolation level constant indicating that dirty reads and non-repeatable reads are prevented; phantom reads can occur.</p>
s t a t i c S t r i n g	<p>TRANSACTION_TYPE_DEFAULT</p> <p>A string indicating the default transaction type</p>

Method Summary

v o i d	<p>begin()</p> <p>Begins a new transaction.</p>
v o i d	<p>beginNoWriteThrough()</p> <p>Starts a new transaction that does not write changes through to a Loader or ObjectGrid server.</p>
v o i d	<p>close()</p> <p>Closes this session, freeing all resources that are held.</p>
v o i d	<p>commit()</p> <p>Commits a transaction.</p>
c o m . i b	

m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
q
u
e
r
y
.
O
b
j
e
c
t
Q
u
e
r
y

[createObjectQuery\(String qlString\)](#)

Creates an instance of an object query for executing a query over the ObjectMaps visible to this session.

v
o
i
d

[flush\(\)](#)

Forces the current changes in the Session to the Loader or ObjectGrid server.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
e
m

[getEntityManager\(\)](#)

Retrieve the EntityManager associated with this Session.

· EntityManager

ObjectMap

[getMap\(String cacheName\)](#)
Returns the ObjectMap for the specified name.

ObjectGrid

[getObjectGrid\(\)](#)
Returns the ObjectGrid that owns this session.

long

[getRequestRetryTimeout\(\)](#)
Retrieves the current request retry timeout for this session.

com.ibm.websphere.objgrid.Session

[getSessionHandle\(\)](#)
Retrieves a handle for this session.

o
n
H
a
n
d
l
e

i
n
t

[getTransactionIsolation\(\)](#)
Retrieves the current transaction isolation level for this session.

i
n
t

[getTransactionTimeout\(\)](#)
Gets current transaction timeout for this Session.

S
t
r
i
n
g

[getTransactionType\(\)](#)
Retrieves the transaction type that is set with the [setTransactionType\(String\)](#) method.

S
e
s
s
i
o
n
.
T
x
C
o
m
m
i
t
p
r
o
t
o
c
o
l

[getTxCommitProtocol\(\)](#)
Retrieve the current commit protocol for this Session.

T
x
I
D

[getTxID\(\)](#)
Gets the TxID transaction identifier, if a transaction is active.

b
o
l
e
a
n

[isCommitting\(\)](#)
Returns whether the current session transaction is performing a session `commit()` operation.

b
o
l
e
a
n

[isFlushing\(\)](#)
Returns whether the current session transaction is performing a session `flush()` operation.

b
o
o

[isMarkedRollbackOnly\(\)](#)
Returns whether or not the current active session transaction is marked as being

l e a n	rollback only as a result of a prior call to the <code>markRollbackOnly(Throwable)</code> method on this Session.
b o o l e a n	<code>isSessionHandleSet()</code> Determines if a SessionHandle is currently set on this Session.
b o o l e a n	<code>isTransactionActive()</code> Determines if a transaction is currently active.
b o o l e a n	<code>isWriteThroughEnabled()</code> Returns whether the current session transaction is writing through to the back end Loader or ObjectGrid server(<code>true</code>), or if the changes are only applying to the BackingMap (<code>false</code>) or client respectively.
v o i d	<code>markRollbackOnly(Throwable t)</code> Marks the current transaction as being rollback only.
v o i d	<code>processLogSequence(LogSequence logSequence)</code> Processes a LogSequence.
v o i d	<code>rollback()</code> Rolls back a transaction.
v o i d	<code>setRequestRetryTimeout(long requestRetryTimeout)</code> Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions.
v o i d	<code>setSessionHandle(com.ibm.websphere.objectgrid.SessionHandle target)</code> Apply a SessionHandle to this session.
v o i d	<code>setTransactionIsolation(int level)</code> Attempts to change the transaction isolation level for this session.
v o i d	<code>setTransactionTimeout(int timeout)</code> Sets the transaction timeout for the next transaction started by this Session object to a specified number of seconds.
v o i d	<code>setTransactionType(String tranType)</code> Sets the transaction type for future transactions.
v o	<code>setTxCommitProtocol(Session.TxCommitProtocol protocol)</code>

i d	Set the commit protocol to be used when committing this Session's transaction.
b o o l l e a n	transactionTimedOut() Determines whether the current session transaction has timed out.

Field Detail

TRANSACTION_TYPE_DEFAULT

static final [String](#) TRANSACTION_TYPE_DEFAULT

A string indicating the default transaction type

See Also:

[Constant Field Values](#)

TRANSACTION_NO_TIMEOUT

static final int TRANSACTION_NO_TIMEOUT

A special value for the timeout parameter of the `setTransactionTimeout(int)` method. This special value is used to indicate that the next transaction started by this Session is allowed unlimited amount of time.

See Also:

[setTransactionTimeout\(int\)](#), [Constant Field Values](#)

DEFAULT_RETRY_TIMEOUT

static final long DEFAULT_RETRY_TIMEOUT

See Also:

[Constant Field Values](#)

TRANSACTION_REPEATABLE_READ

static final int TRANSACTION_REPEATABLE_READ

A transaction isolation level constant indicating that dirty reads and non-repeatable reads are prevented; phantom reads can occur. This level prohibits a transaction from reading an uncommitted cache entry, and it also prohibits the situation where one transaction reads an entry, a second transaction alters the entry, and the first transaction rereads the entry, getting different values the second time (a "non-repeatable read").

Since:

WAS XD 6.1.0.1

See Also:

[setTransactionIsolation\(int\)](#), [Constant Field Values](#)

TRANSACTION_READ_COMMITTED

static final int TRANSACTION_READ_COMMITTED

A transaction isolation level constant indicating that dirty reads are prevented; non-repeatable reads and phantom reads can occur. This level only prohibits a transaction from reading a cache entry with uncommitted changes in it.

Since:

WAS XD 6.1.0.1

See Also:

[setTransactionIsolation\(int\)](#), [Constant Field Values](#)

TRANSACTION_READ_UNCOMMITTED

static final int TRANSACTION_READ_UNCOMMITTED

A transaction isolation level constant indicating that dirty reads, non-repeatable reads and phantom reads can occur. This level allows a cache entry changed by one transaction to be read by another transaction before any changes in that entry have been committed (a "dirty read"). If any of the changes are rolled back, the second transaction will have retrieved an invalid entry.

Since:

WAS XD 6.1.0.1

See Also:

[setTransactionIsolation\(int\)](#), [Constant Field Values](#)

Method Detail

beginNoWriteThrough

```
void beginNoWriteThrough()  
    throws TransactionAlreadyActiveException,  
           TransactionException
```

Starts a new transaction that does not write changes through to a Loader or ObjectGrid server.

Changes made by the session transaction started by this method are only applied to the BackingMap and not given to the Loader. This method can be used to apply changes made in a peer cache to the local BackingMap only. In addition, with a distributed map, this method can be used to start a session transaction which changes will only be applied to the client BackingMap, but not the BackingMap on the server side.

Throws:

[TransactionAlreadyActiveException](#) - if there is already an active transaction
[TransactionException](#) - a TransactionCallbackException occurred or some other error occurred starting a new transaction

getMap

```
ObjectMap getMap(String cacheName)  
    throws UndefinedMapException
```

Returns the ObjectMap for the specified name.

The ObjectMap is used to retrieve and modify values in the BackingMap. Multiple invocations of this method on the same Session object will always return the same object.

This method can also be used to create a BackingMap and its associated ObjectGrid after ObjectGrid initialization. If cacheName does not match the name of a previously created

map, a name comparison will be executed against template maps that have been configured. The ObjectMap and BackingMap will be created if the name matches the regular expression of a template.

Required Client Permission: `ObjectGridPermission.DYNAMIC_MAP` (when creating a new map from a template)

Parameters:

cacheName - name of desired map

Returns:

ObjectMap the transactional interface to modify values in the map

Throws:

[UndefinedMapException](#) - if the map is not defined.

See Also:

[ObjectGrid.defineMap\(String\)](#), [ObjectMap](#)

begin

```
void begin()  
    throws TransactionAlreadyActiveException,  
           TransactionException
```

Begins a new transaction.

Throws:

[TransactionAlreadyActiveException](#) - if this method is invoked with an active transaction
[TransactionException](#) - a TransactionCallbackException occurred or some other error occurred starting a new transaction

commit

```
void commit()  
    throws NoActiveTransactionException,  
           TransactionException
```

Commits a transaction.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction
[TransactionException](#) - if an error occurred during commit processing, see the caused by to determine the root error

See Also:

[markRollbackOnly\(Throwable\)](#)

rollback

```
void rollback()  
    throws NoActiveTransactionException,  
           TransactionException
```

Rolls back a transaction.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction
[TransactionException](#) - if an error occurred during rollback processing, see the caused by to determine the root error

flush

void **flush()**
throws [NoActiveTransactionException](#),
[TransactionException](#)

Forces the current changes in the Session to the Loader or ObjectGrid server. This method does not commit the changes, it just applies the changes.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction
[TransactionException](#) - if an error occurred during flush processing, see the caused by to determine the root error

getObjectGrid

[ObjectGrid](#) getObjectGrid()

Returns the ObjectGrid that owns this session.

Returns:

the owning ObjectGrid instance.

isTransactionActive

boolean isTransactionActive()

Determines if a transaction is currently active.

Returns:

true if a transaction is currently active for this session.

Since:

WAS XD 6.1 FIX3

getTxID

[TxID](#) getTxID()
throws [NoActiveTransactionException](#)

Gets the TxID transaction identifier, if a transaction is active.

Returns:

The current TxID object.

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction

isWriteThroughEnabled

boolean isWriteThroughEnabled()

Returns whether the current session transaction is writing through to the back end Loader or ObjectGrid server(true), or if the changes are only applying to the BackingMap (false) or client respectively.

Returns:

true, if write through is enabled

See Also:

[begin\(\)](#), [beginNoWriteThrough\(\)](#)

setTransactionType


```
void setTransactionType(String tranType)
```

Sets the transaction type for future transactions.

After this method is called, all future transactions will have the same type until another transaction type is set. If no transaction type is set, the default transaction type TRANSACTION_TYPE_DEFAULT will be used.

Transaction types are used mainly for statistical data tracking purpose. Users can predefine types of transactions that will be executed in an application. The idea is to categorize transactions with the same characteristics to one category (type), so one transaction response time statistics can be used to track each transaction type. This approach is useful when your application has different types of transactions. Some types of transactions, such as update transactions, process longer than others transactions, such as read-only transactions. By using the transaction type, different transactions are tracked by different statistics, so the statistics can be more useful.

Parameters:

tranType - the transaction type for future transactions.

See Also:

[TRANSACTION_TYPE_DEFAULT](#)

getTransactionType

```
String getTransactionType()
```

Retrieves the transaction type that is set with the [setTransactionType\(String\)](#) method.

Returns:

the transaction type for the session.

Since:

7.1.1.1

processLogSequence

```
void processLogSequence(LogSequence logSequence)  
    throws NoActiveTransactionException,  
           UndefinedMapException,  
           ObjectGridException
```

Processes a LogSequence.

Each LogElement within the LogSequence will be examined and the appropriate operation (insert, update, invalidate, etc) will be performed against the BackingMap identified by the LogSequence's map name. An ObjectGrid Session must be active before this method is invoked. The caller is responsible for issuing the appropriate commit or rollback invocation to complete the Session. Autocommit processing is not available for this method invocation.

The main use of this method is for processing a LogSequence that was received by a remote JVM. For example, using the Distributed Commit support, the LogSequences associated with a given committed Session are distributed to other listening ObjectGrids in other JVMs. After receiving the LogSequences at the remote JVM, the listener could start a Session using beginNoWriteThrough(), invoke this method, and commit the Session transaction.

Parameters:

logSequence - LogSequence of changes to be applied to an active transaction

Throws:

[NoActiveTransactionException](#) - if this method is invoked with no active transaction

[UndefinedMapException](#) - if the map referenced by the LogSequence cannot be found

[ObjectGridException](#) - if the LogSequence elements cannot be processed

See Also:

[beginNoWriteThrough\(\)](#), [LogSequence](#), [ObjectGridEventListener](#)

isFlushing

boolean **isFlushing()**

Returns whether the current session transaction is performing a session `flush()` operation. It is helpful to know if a session `flush()` is active (`true`), or if only an `ObjectMap.flush()` is in progress (returns `false` in this case).

Returns:

`true`, if the session is executing a session `flush()` call.

Since:

WAS XD 6.0.1

See Also:

[flush\(\)](#), [ObjectMap.flush\(\)](#)

isCommitting

boolean **isCommitting()**

Returns whether the current session transaction is performing a session `commit()` operation. It is helpful to know if a session `commit` is active (`true`), or if an `ObjectMap.flush()` or session `flush()` is in progress (returns `false` in these cases).

Returns:

`true`, if session is executing a session `commit()` call.

Since:

WAS XD 6.0.1

See Also:

[commit\(\)](#), [flush\(\)](#), [ObjectMap.flush\(\)](#)

markRollbackOnly

void **markRollbackOnly**([Throwable](#) t)
throws [NoActiveTransactionException](#)

Marks the current transaction as being rollback only.

Marking a transaction rollback only ensures that even if the `commit()` method is called for this session transaction, the transaction is rolled back. A rollback is typically done when either ObjectGrid itself or the application knows that data corruption could occur if the `commit()` method was allowed to commit the transaction. Once this method is called, the `Throwable` object that is passed to it is chained to the `TransactionException` that is thrown if the `commit` method is ever called. Any subsequent calls to this method for the current active transaction is ignored (e.g. only the first call that passes a non null `Throwable` reference is used). Once the transaction is completed, the rollback only mark is removed so that the next transaction that is started using this session can be committed.

Parameters:

t - the `Throwable` that caused this method to be called.

Throws:

[NoActiveTransactionException](#) - if there is no active transaction for this Session.

Since:

WAS XD 6.0.1

See Also:

[commit\(\)](#), [TransactionException](#)

isMarkedRollbackOnly

boolean `isMarkedRollbackOnly()`

Returns whether or not the current active session transaction is marked as being rollback only as a result of a prior call to the `markRollbackOnly(Throwable)` method on this Session.

Returns:

true if and only if current session transaction is marked rollback only.

Since:

WAS XD 6.0.1

See Also:

[markRollbackOnly\(Throwable\)](#)

setTransactionTimeout

void `setTransactionTimeout(int timeout)`

Sets the transaction timeout for the next transaction started by this Session object to a specified number of seconds.

This method does not affect the transaction timeout of any transactions previously started by this Session. It only affects transactions that are started after this method is called. If this method is never called, the ObjectGrid configured transaction timeout value is used.

Parameters:

`timeout` - is the transaction timeout value in seconds. Use the special value `TRANSACTION_NO_TIMEOUT` if transaction is allowed unlimited amount of time and no transaction timeout should occur.

Since:

WAS XD 6.0.1

See Also:

[TRANSACTION_NO_TIMEOUT](#), [ObjectGrid.setTxTimeout\(int\)](#), [TransactionTimeoutException](#)

getTransactionTimeout

int `getTransactionTimeout()`

Gets current transaction timeout for this Session.

The transaction timeout value returned is the value that was configured for the ObjectGrid using `ObjectGrid.setTxTimeout(int)` or the value passed to `setTransactionTimeout(int)` to override the value configured on ObjectGrid. The return value is in seconds.

Returns:

timeout value in seconds.

Since:

WAS XD 6.0.1

See Also:

[setTransactionTimeout\(int\)](#), [ObjectGrid.setTxTimeout\(int\)](#)

getTransactionIsolation

int `getTransactionIsolation()`

Retrieves the current transaction isolation level for this session.

Returns:

one of the following Session constants: [TRANSACTION_READ_UNCOMMITTED](#), [TRANSACTION_READ_COMMITTED](#) or [TRANSACTION_REPEATABLE_READ](#)

Since:

WAS XD 6.1.0.1

transactionTimedOut

boolean **transactionTimedOut**()

Determines whether the current session transaction has timed out.

Returns:

true if and only if transaction has timed out.

Since:

WAS XD 6.0.1

See Also:

[setTransactionTimeout\(int\)](#)

createObjectQuery

com.ibm.websphere.objectgrid.query.ObjectQuery **createObjectQuery**([String](#) qlString)
throws com.ibm.websphere.objectgrid.query.ObjectQueryException

Creates an instance of an object query for executing a query over the ObjectMaps visible to this session.

When ObjectGrid security is enabled, this method requires an `com.ibm.websphere.objectgrid.security.ObjectGridPermission` with action "query".

Required Client Permission: `ObjectGridPermission.QUERY`

Parameters:

qlString - a query string

Returns:

the new query instance.

Throws:

`com.ibm.websphere.objectgrid.query.ObjectQueryException` - if an error occurs creating the object query.

Since:

WAS XD 6.1

getEntityManager

com.ibm.websphere.objectgrid.em.EntityManager **getEntityManager**()

Retrieve the EntityManager associated with this Session. Each session is associated with a single EntityManager instance. Repeated calls to this method on the same Session instance will result in the same EntityManager instance.

Returns:

this session's EntityManager instance.

Since:

WAS XD 6.1

setTransactionIsolation

void **setTransactionIsolation**(int level)

Attempts to change the transaction isolation level for this session. The constants defined in the Session interface are the possible transaction isolation levels.

This method should normally be invoked prior to beginning a transaction. Invoking after a transaction has started may result in an exception.

Parameters:

level - one of the following Session constants: [TRANSACTION_READ_UNCOMMITTED](#), [TRANSACTION_READ_COMMITTED](#) or [TRANSACTION_REPEATABLE_READ](#)

Since:

WAS XD 6.1.0.1

getSessionHandle

com.ibm.websphere.objectgrid.SessionHandle **getSessionHandle**()

Retrieves a handle for this session.

A SessionHandle contains partition information for the current session and can be re-applied to a new session using the [setSessionHandle\(SessionHandle\)](#) method. A SessionHandle is only applicable for ObjectGrids using per-container partition placement. If [setSessionHandle\(SessionHandle\)](#) is not called before invoking this method, a Session Handle is selected using the properties configured in the [ClientProperties](#). If there are no per-container partition placement mapsets or more than one in the ObjectGrid, an `IllegalStateException` is thrown.

Returns:

the SessionHandle for this session

Throws:

[IllegalStateException](#) - if this method is called in an invalid environment.

Since:

WAS XD 6.1.0.3

setSessionHandle

void **setSessionHandle**(com.ibm.websphere.objectgrid.SessionHandle target)
throws [TargetNotAvailableException](#)

Apply a SessionHandle to this session.

Parameters:

target - the SessionHandle to apply or null to disassociate a SessionHandle from this session.

Throws:

[TargetNotAvailableException](#) - when the target is no longer available.

[IllegalStateException](#) - if the Session has modified some maps already and the SessionHandle has already been set or if this method is called in an invalidate environment.

Since:

WAS XD 6.1.0.3

setRequestRetryTimeout

void **setRequestRetryTimeout**(long requestRetryTimeout)

Set the request retry timeout to indicate how long to retry a request (in milliseconds) when recoverable failures occur, such as fail-over exceptions. A request will timeout

when either the request timeout expires or the transaction timeout expires, whichever expires first.

A value of 0 indicates that all requests should fail immediately and avoid any retry logic. Exceptions that cannot succeed even if tried again such as `DuplicateKeyException` exceptions will be thrown immediately.

A value of -1 indicates that the request retry timeout is not set, meaning that the request duration is governed by the request retry timeout set on the `ClientProperties`. If the `ClientProperties` is also set to -1, then the request retry timeout is governed by the transaction timeout.

Parameters:

`requestRetryTimeout` - the duration in milliseconds retry a client request, 0 if the request should fail immediately or -1 if the request timeout is not set.

Since:

7.0

See Also:

[ClientProperties.setRequestRetryTimeout\(long\)](#), [setTransactionTimeout\(int\)](#)

getRequestRetryTimeout

long `getRequestRetryTimeout()`

Retrieves the current request retry timeout for this session. Returns -1 if it was not set.

Returns:

the duration in milliseconds retry a client request, 0 if the request should fail immediately or -1 if the request timeout is not set.

Since:

7.0

isSessionHandleSet

boolean `isSessionHandleSet()`

Determines if a `SessionHandle` is currently set on this `Session`.

Returns:

true if a `SessionHandle` is currently set on this session.

Since:

7.1

close

void `close()`

Closes this session, freeing all resources that are held. Once closed, this session must be discarded. Use one of the [ObjectGrid.getSession\(\)](#) methods to retrieve a new session. If the session has an active transaction, the transaction will be rolled back and the session resources are not freed.

Throws:

[ObjectGridRuntimeException](#) - thrown if there is a problem releasing resources held by this session.

Since:

7.1.1

setTxCommitProtocol

void `setTxCommitProtocol`([Session.TxCommitProtocol](#) protocol)

Set the commit protocol to be used when committing this Session's transaction. The constants defined in the `TxCommitProtocol` enum are the possible commit protocols.

This method should normally be invoked prior to beginning a transaction. Invoking after a transaction has started will result in an exception.

Parameters:

protocol - one of the following constants `TxCommitProtocol.ONEPHASE` or `TxCommitProtocol.TWOPHASE`

Since:

8.6, XC10 2.5

getTxCommitProtocol

[Session.TxCommitProtocol](#) `getTxCommitProtocol`()

Retrieve the current commit protocol for this Session.

Returns:

one of the following constants `TxCommitProtocol.ONEPHASE` or `TxCommitProtocol.TWOPHASE`

Since:

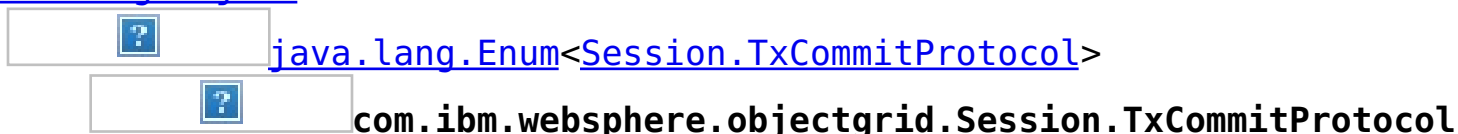
8.6, XC10 2.5

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All Classes			
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid

Enum Session.TxCommitProtocol

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#), [Comparable](#)<[Session.TxCommitProtocol](#)>

Enclosing interface:

[Session](#)

```
public static enum Session.TxCommitProtocol  
extends Enum<Session.TxCommitProtocol>
```

The commit protocols that can be used to commit the Session's transaction

Since:

8.6, XC10 2.5

Enum Constant Summary

[ONEPHASE](#)

A commit protocol constant indicating that the Session transaction can read from multiple partitions but can only write to a single partition.

[TWOPHASE](#)

A commit protocol constant indicating that the Session transaction can read and write from multiple partitions.

Method Summary

s
t
a
t
i
c
S
e
s
s
i
o
n
.
T
x
C
o
m
m
i
t

[valueOf](#)([String](#) name)

Returns the enum constant of this type with the specified name.

P
r
o
t
o
c
o
l

s
t
a
t
i
c
s
e
s
s
i
o
n
.
T
x
C
o
m
m
i
t
P
r
o
t
o
c
o
l
[
]

[values\(\)](#)

Returns an array containing the constants of this enum type, in the order they are declared.

Methods inherited from class [java.lang.Enum](#)

[clone](#), [compareTo](#), [equals](#), [finalize](#), [getDeclaringClass](#), [hashCode](#), [name](#), [ordinal](#), [toString](#), [valueOf](#)

Methods inherited from class [java.lang.Object](#)

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Enum Constant Detail

ONEPHASE

public static final [Session.TxCommitProtocol](#) **ONEPHASE**

A commit protocol constant indicating that the Session transaction can read from multiple partitions but can only write to a single partition. A `TransactionException` is thrown if the transaction writes to multiple partitions. The transaction is committed using the one-phase commit protocol.

TWOPHASE

public static final [Session.TxCommitProtocol](#) **TWOPHASE**

A commit protocol constant indicating that the Session transaction can read and write from multiple partitions. The transaction is committed using the two-phase commit protocol. If the transaction only writes to a single partition then the transaction is

committed using the one-phase commit protocol.

Method Detail

values

```
public static Session.TxCommitProtocol[] values()
```

Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows:

```
for (Session.TxCommitProtocol c : Session.TxCommitProtocol.values())  
    System.out.println(c);
```

Returns:

an array containing the constants of this enum type, in the order they are declared

valueOf

```
public static Session.TxCommitProtocol valueOf(String name)
```

Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

Parameters:

name - the name of the enum constant to be returned.

Returns:

the enum constant with the specified name

Throws:

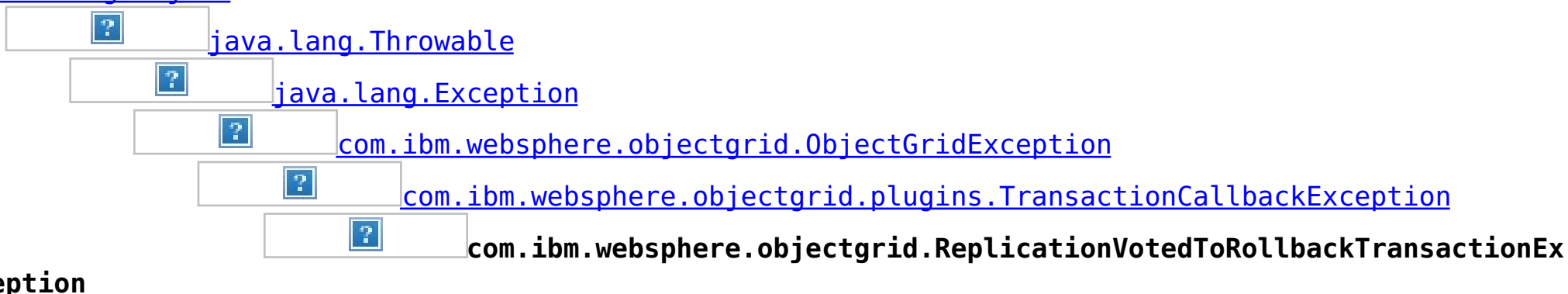
[IllegalArgumentException](#) - if this enum type has no constant with the specified name
[NullPointerException](#) - if the argument is null

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All			
SUMMARY: NESTED ENUM		DETAIL: ENUM						
CONSTANTS FIELD METHOD		CONSTANTS FIELD METHOD						

com.ibm.websphere.objectgrid

Class **ReplicationVotedToRollbackTransactionException**

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class ReplicationVotedToRollbackTransactionException
extends TransactionCallbackException
```

This exception is thrown when a transaction was rolled back because some/all of the replicas failed to apply the transaction when in synchronous replication mode.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ReplicationVotedToRollbackTransactionException](#)()

Constructs a new `ReplicationVotedToRollbackTransactionException` with `null` as its detail message.

[ReplicationVotedToRollbackTransactionException](#)([String](#) message)

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message.

[ReplicationVotedToRollbackTransactionException](#)([String](#) message, [Throwable](#) cause)

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message and cause.

[ReplicationVotedToRollbackTransactionException](#)([Throwable](#) cause)

Constructs a new `ReplicationVotedToRollbackTransactionException` with a specified cause.

Method Summary

Methods inherited from class `com.ibm.websphere.objectgrid.ObjectGridException`

[getCause](#), [initCause](#)

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException()
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException(String message)
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException(String message,
                                                       Throwable cause)
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ReplicationVotedToRollbackTransactionException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ReplicationVotedToRollbackTransactionException

```
public ReplicationVotedToRollbackTransactionException(Throwable cause)
```

Constructs a new `ReplicationVotedToRollbackTransactionException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ReplicationVotedToRollbackTransactionExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

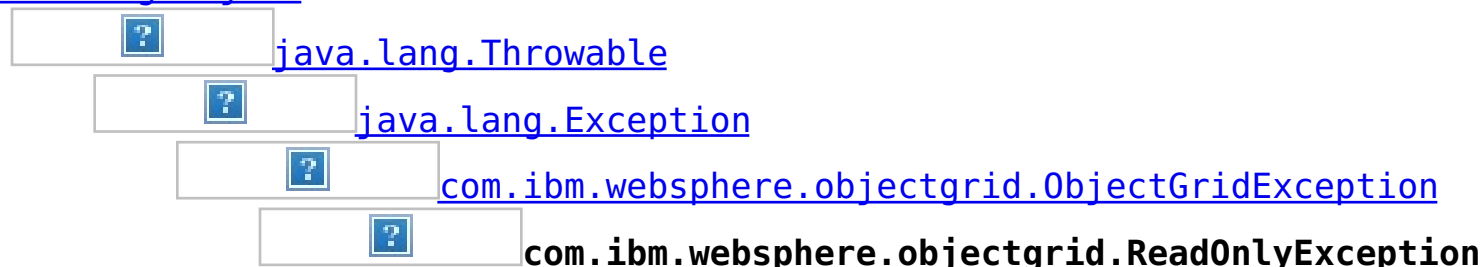
[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid

Class ReadOnlyException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class ReadOnlyException
extends ObjectGridException
```

This exception is thrown when an attempt is made to modifying operations on a read only maps.

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ReadOnlyException](#)()

Constructs a new `ReadOnlyException` with `null` as its detail message.

[ReadOnlyException](#)([String](#) message)

Constructs a new `ReadOnlyException` with the specified detail message.

[ReadOnlyException](#)([String](#) message, [Throwable](#) cause)

Constructs a new `ReadOnlyException` with the specified detail message and cause.

[ReadOnlyException](#)([Throwable](#) cause)

Constructs a new `ReadOnlyException` with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ReadOnlyException

```
public ReadOnlyException()
```

Constructs a new `ReadOnlyException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ReadOnlyException

```
public ReadOnlyException(String message)
```

Constructs a new `ReadOnlyException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ReadOnlyException

```
public ReadOnlyException(String message,
                          Throwable cause)
```

Constructs a new `ReadOnlyException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ReadOnlyException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ReadOnlyException

```
public ReadOnlyException(Throwable cause)
```

Constructs a new `ReadOnlyException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ReadOnlyExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Interface PartitionManager

public interface **PartitionManager**

This interface will be used for calculating the proper partition for a given input key. The set of partitions is determined by the BackingMap configuration.

Since:

WAS XD 6.0.1, XC10

See Also:

[BackingMap.getPartitionManager\(\)](#)

Method Summary

i n t	getNumOfPartitions() Returns the number of configured partitions for this PartitionManager.
i n t	getPartition(Object key) Obtains a 0-based partition number determined by the input parameter's hashCode() method.
L i s t	getPartitionLists(List keyList) This method is very similar to getPartitions(List), except it returns the keys organized by the partition identifiers.
L i s t	getPartitions(List keyList) Obtains the 0-based partition numbers for each of the keys in the input List of keys.
L i s t	partitionLogSequence(LogSequence ls) Partitions a LogSequence based on the partitioning algorithm for the Map.

Method Detail

getPartition

int **getPartition**([Object](#) key)

Obtains a 0-based partition number determined by the input parameter's hashCode() method.

Parameters:

key - Individual key used to determine partition (can not be null)

Returns:

int 0-based partition number

getPartitions

[List](#) `getPartitions(List keyList)`

Obtains the 0-based partition numbers for each of the keys in the input `List` of keys. Each object in the returned list of partition identifiers is an instance of `java.lang.Integer`.

Parameters:

`keyList` - Ordered list of keys

Returns:

List of partition identifiers that corresponds to the input list of keys

getPartitionLists

[List](#) `getPartitionLists(List keyList)`

This method is very similar to `getPartitions(List)`, except it returns the keys organized by the partition identifiers. The return value is a `List` of `Lists`. The outer `List` is an ordered `List` of the partition numbers, with the first entry in the `List` corresponding to partition 0. The inner `Lists` contain the keys from the input parameter that correspond to that partition identifier.

The return value will always contain a `List` object. Either the outer or the inner `Lists` may contain zero elements, but the `List` objects themselves will not be `null`.

Parameters:

`keyList` - Ordered list of keys

Returns:

List of `Lists` that correspond to the 0-based partition numbers, with each inner `List` containing the set of keys that parse to that partition number.

partitionLogSequence

[List](#) `partitionLogSequence(LogSequence ls)`

Partitions a `LogSequence` based on the partitioning algorithm for the `Map`.

Parameters:

`ls` - `LogSequence` that needs to be partitioned

Returns:

List of partitioned `LogSequences`. The first `LogSequence` in the `List` corresponds to the first partition, etc.

getNumOfPartitions

`int` `getNumOfPartitions()`

Returns the number of configured partitions for this `PartitionManager`.

Returns:

the number of configured partitions

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) | DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid

Interface ObjectMap

```
public interface ObjectMap
```

This is a handle to a named Map. Maps should have homogeneous keys and values. An instance of this ObjectMap can only be used by the thread that is currently associated with the Session that was used to get this ObjectMap instance. Both Session and ObjectMap objects are not allowed to be shared by multiple threads concurrently.

The CopyMode setting on the map determines whether or not a copy of the value is returned by get methods. It also determines whether or not a copy of the committed value is made at commit time. The LockStrategy setting for the map determines whether or not a lock is obtained for each map entry accessed by the transaction, the lock mode of the lock obtained, and when the lock is obtained.

Each data access method includes a "Specification details" table that includes the following information:

Required permission: The permission required to use the API.

Pessimistic read lock acquired: The type of lock that is acquired when using pessimistic locking with repeatable read transaction isolation.

Pessimistic read lock held: The type of lock that is held for the duration of the transaction when using pessimistic locking with repeatable read transaction isolation. Locks can be upgraded but not demoted during a transaction.

Transaction: The state of the transaction when invoking the API.

- **Manual** - The transaction is explicitly demarcated using the associated [Session](#) and the specified API is used within the scope of that transaction.
- **Automatic** - The transaction is automatically demarcated. This is also referred to as an auto-commit transaction or API which the associated [Session.isTransactionActive\(\)](#) is false when invoking the API.

Cache tier: Identifies the map cache tiers that are included when fetching or updating cache entries in the call and under what circumstances.
The following tiers are available for client maps:

- Transactional Cache
- Near Cache (if enabled)
- Server Cache
- Loader (if enabled)

The following tiers are available for local maps:

- Transactional Cache
- Local Cache
- Loader (if enabled)

Since:

WAS XD 6.0, XC10

See Also:

[Session.getMap\(String\)](#), [BackingMap.setCopyMode\(CopyMode, Class\)](#),
[BackingMap.setLockStrategy\(LockStrategy\)](#)

Nested Class Summary

s
t
a
t
i
c
c
l
a
s
s

[ObjectMap.PutMode](#)

Identifies the operation mode of the [put\(Object, Object\)](#), [putAll\(Map\)](#), [JavaMap.put\(Object, Object\)](#) and [JavaMap.putAll\(Map\)](#) methods.

Field Summary

s
t
a
t
i
c
l
o
n
g

[QUEUE_TIMEOUT_INFINITE](#)

Used as a parameter on the [getNextKey\(long\)](#) method, specifies the method should block until a key becomes available.

s
t
a
t
i
c
l
o
n
g

[QUEUE_TIMEOUT_NONE](#)

Used as a parameter on the [getNextKey\(long\)](#) method, specifies to return a null value if the map is empty.

s
t
a
t
i
c
i
n
t

[TTL_FOREVER](#)

A constant indicating the time-to-live value is "forever".

s
t
a
t
i
c
i
n
t

[USE_DEFAULT](#)

A constant indicating the time-to-live value or lock timeout value is the default setting.

Method Summary

v

[clear\(\)](#)

o i d	Clear all keys from the Map.
v o i d	clearCopyMode() Resets the CopyMode back to the one in the BackingMap.
b o o l l e a n	containsKey(Object key) Returns true if this map contains a mapping for the specified key.
v o i d	flush() Pushes the current set of changes for the ObjectMap instance to the Loader without committing the changes.
O b j e c t	get(Object key) Retrieves the object from the cache at the given key.
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . d a t a g r i d . A g e n t M a n	getAgentManager() Returns the Agent Manager that allows DataGrid operations to be performed on the objects within this Map.

a
g
e
r

[List](#)

[getAll](#)([List](#) keyList)
Gets a list of entries from the map.

[List](#)

[getAllForUpdate](#)([List](#) keyList)
Same as the `getAll(List)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
p
r
o
j
e
c
t
o
r
.
m
d
.
E
n
t
i
t
y
M
e
t
a
d
a
t
a

[getEntityMetadata](#)()
Retrieve the metadata for the entity associated with this map.

[Object](#)

[getForUpdate](#)([Object](#) key)
Same as `get(Object)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry.

[Object](#)

[getIndex](#)([String](#) name)
Returns a reference to the named index that can be used with this Map.

O
b
j
e
c
t

[getIndex](#)(String name, boolean forUpdate)

Returns a reference to the named index that can be used with this Map.

M
a
p

[getJavaMap](#)()

Returns an implementation of java.util.Map that is backed by this ObjectMap.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

[getKeyOutputFormat](#)()

Retrieves the data format for all data access APIs that return cache keys for the current session.

i
n
t

[getMapType](#)()

Returns the type of the underlying BackingMap.

S
t
r
i
n
g

[getName](#)()

Returns the name of the ObjectMap as defined by the configuration.

O
b
j
e
c
t

[getNextKey](#)(long timeout)

Retrieves a key off the map in first-in-first-out (FIFO) insert order.

O
b

j
e
c
t
M
a
p
.
P
u
t
M
o
d
e

[getPutMode\(\)](#)

Retrieve the current PutMode set for this ObjectMap instance.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

[getValueOutputFormat\(\)](#)

Retrieves the data format for all data access APIs that return cache values for the current session.

v
o
i
d

[insert\(Object key, Object value\)](#)

Performs an explicit insert of a given entry.

v
o
i
d

[invalidate\(Object key, boolean isGlobal\)](#)

Invalidates an entry in the cache based on the key parameter.

v
o
i
d

[invalidateAll\(Collection keyList, boolean isGlobal\)](#)

Invalidate a set of cache entries based on the Collection of keys provided.

b
o

o l e a n	lock (Object key, com.ibm.websphere.objectgrid.LockMode lockMode) Obtains a lock for the given key.
L i s t < B o o l e a n >	lockAll (List keyList, com.ibm.websphere.objectgrid.LockMode lockMode) Obtains locks for the given keys.
O b j e c t	put (Object key, Object value) Puts the Object value into the cache at location represented by key.
v o i d	putAll (Map map) Puts each of the Object value into the cache at location represented by key contained in the Map.
O b j e c t	remove (Object key) Removes the Object value from the cache represented by key.
v o i d	removeAll (Collection keyList) Batch remove from the Map.
v o i d	setCopyMode (CopyMode copyMode, Class valueInterface) Allows the CopyMode for the Map to be overridden on this map on this session only.
v o i d	setKeyOutputFormat (com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data format for all data access APIs that return cache keys for the current session.
v o i d	setLockTimeout (int seconds) Overrides the BackingMap's lock timeout for this ObjectMap.
v o i d	setPutMode (ObjectMap.PutMode putMode) Allows the default operation for the put(Object, Object) and putAll(Map) methods to be changed, allowing the put to use the optimized upsert(Object, Object) and upsertAll(LinkedHashMap) implementations.
i n t	setTimeToLive (int ttl) Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL.
v o i d	setValueOutputFormat (com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data format for all data access APIs that return cache values for the current session.

v o i d	<p>touch(Object key)</p> <p>Updates the last access time in the BackingMap without retrieving the value to the ObjectMap.</p>
v o i d	<p>update(Object key, Object value)</p> <p>Performs an explicit update of a given entry.</p>
v o i d	<p>upsert(Object key, Object value)</p> <p>Puts the Object value into the cache at location represented by key.</p>
v o i d	<p>upsertAll(LinkedHashMap map)</p> <p>Puts each of the Object value into the cache at location represented by key contained in the Map.</p>

Field Detail

TTL_FOREVER

static final int **TTL_FOREVER**

A constant indicating the time-to-live value is "forever".

See Also:

[Constant Field Values](#)

USE_DEFAULT

static final int **USE_DEFAULT**

A constant indicating the time-to-live value or lock timeout value is the default setting.

The default setting is to retain the time-to-live value for any existing map entry and to use the default value from BackingMap setting if a new map entry is being created.

For lock timeout override the default setting is to use the value defined on the BackingMap

See Also:

[setLockTimeout\(int\)](#), [setTimeToLive\(int\)](#), [BackingMap.setTimeToLive\(int\)](#), [BackingMap.getTimeToLive\(\)](#), [BackingMap.setLockTimeout\(int\)](#), [Constant Field Values](#)

QUEUE_TIMEOUT_NONE

static final long **QUEUE_TIMEOUT_NONE**

Used as a parameter on the [getNextKey\(long\)](#) method, specifies to return a null value if the map is empty.

See Also:

[Constant Field Values](#)

QUEUE_TIMEOUT_INFINITE

static final long `QUEUE_TIMEOUT_INFINITE`

Used as a parameter on the [getNextKey\(long\)](#) method, specifies the method should block until a key becomes available.

See Also:

[Constant Field Values](#)

Method Detail

getName

[String](#) `getName()`

Returns the name of the ObjectMap as defined by the configuration.

Returns:

name of ObjectMap

get

[Object](#) `get(Object key)`
throws [ObjectGridException](#)

Retrieves the object from the cache at the given key.

Whether or not a copy of the object is returned is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode. If the key cannot be found in the map, a null value will be returned. A null value is also returned if a value is null and this map allows null values. To distinguish the two, use the `containsKey` method.

The return value is a SerializedValue when using the [CopyMode.COPY_TO_BYTES_RAW](#) CopyMode or `OutputFormat.RAW` OutputFormat with a ValueSerializerPlugin plug-in defined on the [BackingMap](#). The SerializedValue allows access to the value in its serialized form, or its native Java Object form.

The return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

Specification details:

Required permission:	MapPermission.READ
Pessimistic read lock acquired:	LockMode.SHARED
Pessimistic read lock held:	Yes
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found.

Parameters:

key - The entry to fetch

Returns:

the value, null, SerializedValue OR Tuple

Throws:

[IllegalArgumentException](#) - if key is null
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[containsKey\(Object\)](#), [getForUpdate\(Object\)](#), [CopyMode](#)

put

[Object](#) put([Object](#) key,
[Object](#) value)
throws [ObjectGridException](#)

Puts the Object value into the cache at location represented by key.

The value will be pushed down to the BackingMap/Loader at commit time and has two behaviors, which can be altered using the [setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSERTUPDATE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSERT](#) The value is put into the map using the specification of the [upsert\(Object, Object\)](#).

Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See [CopyMode](#) for a description of each possible copy mode.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an [EntityManager](#) API entity is associated with the [BackingMap](#).

Specification details:

Required [MapPermission.WRITE](#)
permission:

Transaction: Automatic or manual

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - The entry to put into the map
value - The value to put into the map using the key

Returns:

If [ObjectMap.PutMode.INSERTUPDATE](#) is set, return the previous value in this transaction.
If [ObjectMap.PutMode.UPSERT](#) is set, the return value is null.

Throws:

[IllegalArgumentException](#) - if key is null, or if the map does not allow null values and value is null
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[CopyMode](#)

getForUpdate

[Object](#) getForUpdate([Object](#) key)
throws [ObjectGridException](#)

Same as `get(Object)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry. See `LockStrategy.PESSIMISTIC` for additional information. Whether or not a copy of the object is returned is determined by the `CopyMode` setting for this map. See `CopyMode` for a description of each possible `CopyMode`. If the key cannot be found in the map, a `null` value will be returned. A `null` value is also returned if the value is `null` and this map allows `null` values. To distinguish the two, use the `containsKey` method.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

The return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

Specification details:

Required permission:	<code>MapPermission.READ</code>
Pessimistic read lock acquired:	<code>LockMode.UPGRADABLE</code> Note: Transaction isolation is ignored.
Pessimistic locks held:	Yes
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found and the appropriate lock is acquired.

Parameters:

key - The entry to fetch

Returns:

the value, `null`, `SerializedValue` OR `Tuple`

Throws:

[IllegalArgumentException](#) - if key is `null`

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[containsKey\(Object\)](#), [get\(Object\)](#), [CopyMode](#), [LockStrategy.PESSIMISTIC](#)

remove

`Object` `remove(Object key)`
throws [ObjectGridException](#)

Removes the Object value from the cache represented by key.

This removal will be pushed down to the `BackingMap/Loader` at commit time. If the key cannot be found in the map, a `null` value will be returned.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

The return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

Specification details:

Required `MapPermission.REMOVE`
permissi

on:

Transact Automatic or manual

ion:

Cache Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the
tier: operation to the Client Cache tier for client maps, or the Server Cache tier for
local or shard maps.

Parameters:

key - The entry to remove

Returns:

the current value at invocation time

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not
have the appropriate permission.

getAll

[List](#) `getAll(List keyList)`
throws [ObjectGridException](#)

Gets a list of entries from the map.

If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

A return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

Specification details:

Required permission:	<code>MapPermission.READ</code>
Pessimistic read lock acquired:	<code>LockMode.SHARED</code>
Pessimistic read lock held:	Yes
Transaction:	Automatic or manual
Cache tier:	For each key, progresses to all tiers until the key is found.

Parameters:

keyList - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not
have the appropriate permission.

See Also:

[get\(Object\)](#)

getAllForUpdate

[List](#) `getAllForUpdate(List keyList)`
throws [ObjectGridException](#)

Same as the `getAll(List)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries. See `LockStrategy.PESSIMISTIC` for additional information. If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

A return value is a `Tuple` when an an `EntityManager` API entity is associated with the `BackingMap`.

Specification details:

Required permission: `MapPermission.READ`
Pessimistic read lock acquired: `LockMode.UPGRADABLE`
Note: Transaction isolation is ignored.
Pessimistic locks held: Yes
Transaction: Automatic or manual
Cache tier: Progresses to all tiers until the keys are found and the appropriate locks are acquired.

Parameters:

`keyList` - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if `keyList` is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[getAll\(List\)](#), [getForUpdate\(Object\)](#), [LockStrategy.PESSIMISTIC](#)

removeAll

`void removeAll(Collection keyList)`
throws [ObjectGridException](#)

Batch remove from the Map. If a key in the list cannot be found, it will be ignored.

Specification details:

Required permission: `MapPermission.REMOVE`
Transaction: Automatic or manual
Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

`keyList` - A list of keys for identifying which entries to remove

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[remove\(Object\)](#)

putAll

```
void putAll(Map map)
    throws ObjectGridException
```

Puts each of the Object value into the cache at location represented by key contained in the Map.

The values will be pushed down to the BackingMap/Loader at commit time and has two behaviors, which can be altered using the [setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSE](#) The values are put into the map using the specification of the [upsertAll\(LinkedHashMap\)](#).

Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See CopyMode for a description of each possible copy mode.

An existing Map object will be passed in to use for obtaining the keys and values to be inserted or updated into the existing Map.

Specification details:

Required MapPermission.WRITE
permission:

Transact Automatic or manual
ion:

Cache Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the
tier: operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

map - The key/values to be put into the map.

Throws:

[IllegalArgumentException](#) - if map is null or contains a null key or if null values are not allowed and map contains a null value.

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[put\(Object, Object\)](#)

invalidate

```
void invalidate(Object key,
    boolean isGlobal)
    throws ObjectGridException
```


Invalidates an entry in the cache based on the key parameter.

If the key's value has changes pending in the ObjectMap, it is the application's responsibility to flush these changes to the Loader before invalidation. If a flush is not performed prior to invoking the invalidate operation, all pending changes for this key will be removed from the ObjectMap. If the key cannot be found in the map, it will be ignored.

The isGlobal parameter is used to indicate which cache level is used to invalidate the entries. If isGlobal is true, when the transaction is committed, the key is removed from the BackingMap also. If a subsequent get operation is performed, the BackingMap will be skipped and the Loader will be used to get the data. If isGlobal is false, the entry is only invalidated in the ObjectMap (transactional cache). If a subsequent get operation is performed, the BackingMap can be used; and, if it's not in the BackingMap, the Loader will be used to get the data.

A typical use of isGlobal being false is when a large number of records are touched in a transaction and the application wants to evict records that are no longer used in the cache.

Specification details:

Requires MapPermission.INVALIDATE

red

permission:

ssion:

Transaction Automatic or manual

action:

n:

Caching Applied to all tiers during commit except the Loader. Use

the tier: [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps. Set the isGlobal parameter to false to limit the operation to the transaction cache tier.

Parameters:

key - Object representing the key to be used for cache entry invalidation

isGlobal - Indicates whether to remove the entry from the BackingMap (true) or just the ObjectMap (false).

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

invalidateAll

```
void invalidateAll(Collection keyList,  
                  boolean isGlobal)  
    throws ObjectGridException
```

Invalidate a set of cache entries based on the Collection of keys provided. If a key in the collection cannot be found, it will be ignored.

Specification details:

Requires MapPermission.INVALIDATE

red

permission:

ssion:

Transaction Automatic or manual

action:

n:

Cache tier: Applied to all tiers during commit except the Loader. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps. Set the `isGlobal` parameter to `false` to limit the operation to the transaction cache tier.

Parameters:

`keyList` - A Collection of keys representing the entries to be invalidated
`isGlobal` - Indicates whether to remove the entry from the BackingMap (true) or just the ObjectMap (false).

Throws:

[IllegalArgumentException](#) - if `keyList` is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[invalidate\(Object, boolean\)](#)

setTimeToLive

```
int setTimeToLive(int ttl)
```

Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this ObjectMap, any previous value set by the BackingMap.setTimeToLive(int) method is overridden for this ObjectMap. If this method is never called on the ObjectMap, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from BackingMap setting if a new map entry is being created. If TTL is never set on the BackingMap, the cache entry can live "forever".

This method can only be used when the TTLType is set to LAST_ACCESS_TIME or LAST_UPDATE_TIME on the BackingMap. If this method is called on the ObjectMap and the TTLType is something other than LAST_ACCESS_TIME or LAST_UPDATE_TIME, an IllegalStateException is thrown.

Required permission: MapPermission.INVALIDATE

Parameters:

`ttl` - is the time-to-live value in seconds. The value must be ≥ -1 . A value of 0 is used to indicate the cache entry can live "forever" and -1 to indicate to use default setting. Use of the constant TTL_FOREVER is recommended when "forever" is desired and the constant USE_DEFAULT is recommended when "use default" setting is desired.

Returns:

previous time-to-live value in seconds. The constant TTL_FOREVER and constant USE_DEFAULT can be used to determine if the previous TTL is one of the special values.

Throws:

[IllegalArgumentException](#) - if seconds argument is < -1 .
[IllegalStateException](#) - if BackingMap.getTtlEvictorType() returns anything other than TTLType.LAST_ACCESS_TIME or TTLType.LAST_UPDATE_TIME.
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[TTL_FOREVER](#), [USE_DEFAULT](#), [BackingMap.setTimeToLive\(int\)](#), [TTLType.LAST_ACCESS_TIME](#), [TTLType.LAST_UPDATE_TIME](#)

update


```
void update(Object key,  
           Object value)  
    throws KeyNotFoundException,  
           ObjectGridException
```

Performs an explicit update of a given entry.

A get operation is not required prior to invoking the update method (unlike the put method). Also, an update invocation will never insert a new record. If the map's LockStrategy is LockStrategy.OPTIMISTIC this method will implicitly get the entry so as to have the version value of the object for when this method was invoked. Whether or not a copy of the object is made when transaction is committed is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode.

If a key cannot be found in the map during commit, a TransactionException will be thrown.

Specification details:

Required MapPermission.WRITE
permission:

Transaction: Automatic or manual

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - Identifies the entry to be updated
value - The updated value for this entry

Throws:

[IllegalArgumentException](#) - if key is null or if the map does not allow null values and value is null.

[KeyNotFoundException](#) - if the key cannot be found in the map

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

See Also:

[insert\(Object, Object\)](#), [put\(Object, Object\)](#), [CopyMode](#), [LockStrategy.OPTIMISTIC](#)

insert

```
void insert(Object key,  
           Object value)  
    throws DuplicateKeyException,  
           ObjectGridException
```

Performs an explicit insert of a given entry.

The key must not exist before executing this method. Also, an insert invocation will never update an existing record. Whether or not a copy of the object is made when a transaction is committed is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode.

If the key is already in the map, a TransactionException will be thrown during commit.

Specification details:

Required MapPermission.INSERT

permissi
on:

Transact Automatic or manual
ion:

Cache Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the
tier: operation to the Client Cache tier for client maps, or the Server Cache tier for
local or shard maps.

Parameters:

key - Identifies the entry to be inserted
value - The value for this entry

Throws:

[IllegalArgumentException](#) - if key is null or if the map does not allow null values and
value is null.

[DuplicateKeyException](#) - if this entries already exists in the map

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not
have the appropriate permission.

See Also:

[put\(Object, Object\)](#), [update\(Object, Object\)](#), [CopyMode](#)

getIndex

[Object](#) [getIndex](#)([String](#) name)
throws [com.ibm.websphere.objectgrid.IndexUndefinedException](#),
[com.ibm.websphere.objectgrid.IndexNotReadyException](#)

Returns a reference to the named index that can be used with this Map. This index cannot be shared between threads and works on the same rules as Session. The returned value should be cast to the right index interface such as [MapIndex](#), [MapRangeIndex](#) or a custom index interface such as a geo spatial index.

Parameters:

name - The index name

Returns:

A reference to the index, it must be cast to the appropriate index interface.

Throws:

[IndexUndefinedException](#) - if the index is not defined on the [BackingMap](#)

[IndexNotReadyException](#) - if the index is a dynamic index and it is not ready

Since:

WAS XD 6.0.1

getIndex

[Object](#) [getIndex](#)([String](#) name,
boolean forUpdate)
throws [com.ibm.websphere.objectgrid.IndexUndefinedException](#),
[com.ibm.websphere.objectgrid.IndexNotReadyException](#)

Returns a reference to the named index that can be used with this Map. This index cannot be shared between threads and works on the same rules as Session. The returned value should be cast to the right index interface such as [MapIndex](#), [MapRangeIndex](#) or a custom index interface such as a geo spatial index.

Parameters:

name - The index name

forUpdate - if true, the returned index will always operate with forUpdate intent.

Returns:

A reference to the index, it must be cast to the appropriate index interface.

Throws:

`IndexUndefinedException` - if the index is not defined on the `BackingMap`
`IndexNotReadyException` - if the index is a dynamic index and it is not ready

Since:

WAS XD 6.1.0.1

flush

void **flush**()
throws [ObjectGridException](#)

Pushes the current set of changes for the `ObjectMap` instance to the `Loader` without committing the changes. The changes are not propagated to the `BackingMap` either. This is useful for re-priming the `Loader`'s data without committing the current transaction and starting over.

Throws:

[ObjectGridException](#) - if an error occurs during processing

See Also:

[Session.flush\(\)](#)

containsKey

boolean **containsKey**([Object](#) key)
throws [ObjectGridException](#)

Returns `true` if this map contains a mapping for the specified key. `ObjectGrid` does not support null keys. If you configured the map to support `null` values, this method can be used to determine whether a key is contained in the map or not.

This API does not hold any locks when using pessimistic locking.

Specification details:

Required permission:	<code>MapPermission.READ</code>
Pessimistic read lock acquired:	<code>LockMode.SHARED</code>
Pessimistic locks held:	None
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found.

Parameters:

key - key whose presence in this map is to be tested.

Returns:

`true` if this map contains a mapping for the specified key.

Throws:

[IllegalArgumentException](#) - if `null` key parameter is passed in
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

getJavaMap

[Map](#) **getJavaMap**()

Returns an implementation of `java.util.Map` that is backed by this `ObjectMap`.

The returned `java.util.Map` implementation can be cast to `com.ibm.websphere.objectgrid.JavaMap` to be able to use the rest of the `ObjectGrid`

programming model, but with `java.util.Map`'s use of `RuntimeExceptions` instead of checked `ObjectGridExceptions`.

Returns:

a `java.util.Map` backed by this `ObjectMap`

See Also:

[Map](#), [JavaMap](#)

touch

```
void touch(Object key)
    throws ObjectGridException
```

Updates the last access time in the `BackingMap` without retrieving the value to the `ObjectMap`.

The last access time is updated during commit. If the key does not exist in the `BackingMap`, a `TransactionException` will be returned during commit processing.

Specification details:

Required None

permission:

Transaction: Automatic or manual

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - key to be touched

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

setCopyMode

```
void setCopyMode(CopyMode copyMode,
    Class valueInterface)
    throws TransactionAlreadyActiveException,
    ObjectGridException
```

Allows the `CopyMode` for the Map to be overridden on this map on this session only.

This method allows an application to use an optimal `CopyMode` `TRANSACTION` by `TRANSACTION` as its needs dictate. The `CopyMode` cannot be changed during a transaction. There must be no active transaction when this method is called.

Parameters:

copyMode - must be one of the final static variables defined in `CopyMode`. See `CopyMode` class for an explanation of each mode and how the valueInterface is used for `CopyMode.COPY_ON_WRITE`.

valueInterface - the value interface `Class` object. Specify null in version 7.1 and later.

Throws:

[IllegalArgumentException](#) - if copyMode is null or `COPY_ON_WRITE` `CopyMode` is specified and the required value interface parameter is null

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

[ObjectGridException](#) - if an error occurs during processing

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\), CopyMode](#)

clearCopyMode

void **clearCopyMode**()
throws [TransactionAlreadyActiveException](#)

Resets the CopyMode back to the one in the BackingMap.

This method is used to reverse a previous setCopyMode method call for this ObjectMap. This method can only be called when no transaction is active on the associated session.

Throws:

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

See Also:

[setCopyMode\(CopyMode, Class\)](#)

getNextKey

[Object](#) **getNextKey**(long timeout)
throws [ObjectGridException](#)

Retrieves a key off the map in first-in-first-out (FIFO) insert order.

The entry is locked by the session such that other calls to getNextKey will not return the same key. The key can be used to remove or manipulate the value although leaving the entry will result in the key remaining at the beginning of the queue. This order is optimized for performance and is not guaranteed especially across partitions or in highly concurrent environments.

The return value is a SerializedKey when OutputFormat.RAW is set for the keys. The default key output format for maps that are associated with a KeySerializerPlugin is OutputFormat.RAW. The SerializedKey allows access to the value in its serialized form, or its native Java Object form.

The return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

Specification details:

Required permission:	MapPermission.READ
Pessimistic read lock acquired:	LockMode.EXCLUSIVE
Pessimistic read lock held:	Yes
Transaction:	Automatic or manual

Parameters:

timeout - The period of time in milliseconds to wait for an entry to become available on the queue.

Returns:

the next key

Throws:

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

WAS XD 6.1

See Also:

getEntityMetadata

com.ibm.websphere.projector.md.EntityMetadata **getEntityMetadata()**

Retrieve the metadata for the entity associated with this map.

Returns:

the EntityMetadata if an entity is associated with this map or null if there is no entity associated with this map.

Since:

WAS XD 6.1

getMapType

int **getMapType()**

Returns the type of the underlying BackingMap.

The return value is equivalent to one of the constants declared on the BackingMap interface, [BackingMap.LOCAL](#), [BackingMap.SERVER](#), or [BackingMap.CLIENT](#).

Returns:

the BackingMap type

Since:

WAS XD 6.1

getAgentManager

com.ibm.websphere.objectgrid.datagrid.AgentManager **getAgentManager()**

Returns the Agent Manager that allows DataGrid operations to be performed on the objects within this Map.

This method should only be called on a client ObjectGrid. If called on a non client ObjectGrid an `IllegalStateException` will be thrown

Returns:

AgentManager

Throws:

[IllegalStateException](#) - if this method is invoked on a non client ObjectGrid

Since:

WAS XD 6.1

setLockTimeout

void **setLockTimeout**(int seconds)

Overrides the BackingMap's lock timeout for this ObjectMap.

Establishes the number of seconds that any given fetch (get, getForUpdate, find, findForUpdate) of a cache entry will wait to get a lock. When the lock strategy is `LockStrategy.NONE`, no lock manager is used by this map. In this case, a call to this method does nothing.

Parameters:

seconds - is the lock timeout in seconds. The value must be ≥ -1 . A value of -1 is

used to indicate to use the default setting. Use of the constant `USE_DEFAULT` is recommended when "use default" setting is desired. A value of 0 indicates that if a lock cannot be retrieved immediately to time out without waiting for any period of time for the lock to be released and made available.

Throws:

[IllegalArgumentOutOfRangeException](#) - if seconds argument is less than -1 (`USE_DEFAULT`)

Since:

WAS XD 6.1

See Also:

[USE_DEFAULT](#), [BackingMap.setLockTimeout\(int\)](#), [BackingMap.setLockStrategy\(LockStrategy\)](#), [LockStrategy.OPTIMISTIC](#), [LockStrategy.PESSIMISTIC](#)

clear

```
void clear()  
    throws ObjectGridException
```

Clear all keys from the Map.

This method is an automatic transaction call. The [Session.isTransactionActive\(\)](#) must answer false prior to invoking this method.

Specification details:

Required permission:	MapPermission.REMOVE
Pessimistic read lock acquired:	Map exclusive lock is acquired.
Transaction:	Hybrid. Each shard is cleared in a separate transaction.
Cache tier:	Applies to all cache tiers except the Loader.

Throws:

[ObjectGridException](#) - if an error occurs during processing
[TransactionAlreadyActiveException](#) - if a transaction is already started.
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

WAS XD 6.1.0.3

lock

```
boolean lock(Object key,  
            com.ibm.websphere.objectgrid.LockMode lockMode)  
    throws ObjectGridException
```

Obtains a lock for the given key. **Specification details:**

Required permission:	MapPermission.READ
Pessimistic read lock acquired:	LockMode.SHARED, LockMode.UPGRADABLE OR LockMode.EXCLUSIVE Note: Transaction isolation is ignored.
Pessimistic locks held:	Yes
Transaction:	Automatic or manual
Cache tier:	Progresses to all tiers until the key is found and the appropriate lock is acquired.

Parameters:

key - the key to lock
lockMode - the lockMode to obtain for the given key

Returns:

true if the entry exists in the grid or Loader (if one is defined)

Throws:

[IllegalArgumentException](#) - if key is null
[IllegalStateException](#) - if this map is not using [LockStrategy.PESSIMISTIC](#) LockStrategy
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:
8.6, XC10 2.5

lockAll

[List<Boolean>](#) **lockAll**([List](#) keyList,
com.ibm.websphere.objectgrid.LockMode lockMode)
throws [ObjectGridException](#)

Obtains locks for the given keys. **Specification details:**

Required permission: MapPermission.READ
Pessimistic read lock acquired: LockMode.SHARED, LockMode.UPGRADABLE OR LockMode.EXCLUSIVE
Note: Transaction isolation is ignored.
Pessimistic locks held: Yes
Transaction: Automatic or manual
Cache tier: Progresses to all tiers until the keys are found and the appropriate locks are acquired.

Parameters:

keyList - the keys to lock
lockMode - the lockMode to obtain for the given keys

Returns:

a List of Booleans indicating whether the entries exists in the grid or Loader (if one is defined)

Throws:

[IllegalStateException](#) - if this map is not using [LockStrategy.PESSIMISTIC](#) LockStrategy
[IllegalArgumentException](#) - if keyList is null or contains a null element.
[ObjectGridException](#) - if an error occurs during processing
[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:
8.6, XC10 2.5

upsert

void **upsert**([Object](#) key,
[Object](#) value)
throws [ObjectGridException](#)

Puts the Object value into the cache at location represented by key.

The value will be pushed down to the BackingMap/Loader at commit time. The semantics of this method are that the Loader will receive a [LogElement.UPSERT](#) operation and the map will either do an insert or an update to cause the map to contain this updated value. Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See CopyMode for a description of each possible copy mode.

Specification details:

Required MapPermission.WRITE
permissi
on:
Transact Automatic or manual

ion:

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

key - The entry to insert or update in the map

value - The value to insert or update in the map using the key

Throws:

[IllegalArgumentException](#) - if key is null, or if the map does not allow null values and value is null

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

8.6, XC10 2.5

See Also:

[CopyMode](#)

upsertAll

```
void upsertAll(LinkedHashMap map)
    throws ObjectGridException
```

Puts each of the Object value into the cache at location represented by key contained in the Map. The values will be pushed down to the BackingMap/Loader at commit time. The semantics of this method are that the Loader will receive a [LogElement.UPSERT](#) operation and the map will either do an insert or an update to cause the map to contain this updated value. Whether or not a copy of the objects is made when transaction is committed is determined by the copy mode setting for this map. See [CopyMode](#) for a description of each possible copy mode.

Specification details:

Required `MapPermission.WRITE`

permissi

on:

Transact Automatic or manual

ion:

Cache tier: Applied to all tiers during commit. Use [Session.beginNoWriteThrough\(\)](#) to limit the operation to the Client Cache tier for client maps, or the Server Cache tier for local or shard maps.

Parameters:

map - The key/values to be inserted or updated in the map. The type is `LinkedHashMap` so that the order can be controlled to avoid deadlocks.

Throws:

[IllegalArgumentException](#) - if map is null or contains a null key or if null values are not allowed and map contains a null value.

[ObjectGridException](#) - if an error occurs during processing

[AccessControlException](#) - if the Subject or Credential specified on the Session does not have the appropriate permission.

Since:

8.6, XC10 2.5

See Also:

[CopyMode](#)

getKeyOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getKeyOutputFormat()**

Retrieves the data format for all data access APIs that return cache keys for the current session.

Returns:

the data output format or null if the default should be used.

Since:

8.6, XC10 2.5

setKeyOutputFormat

void **setKeyOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)
throws [TransactionAlreadyActiveException](#)

Sets the data format for all data access APIs that return cache keys for the current session.

This method supports map configurations with a KeyDataSerializer plug-in defined, or with eXtreme Data Format enabled.

Parameters:

outputFormat - the data output format to use or OutputFormat.UNDEFINED to use the default defined on the parent [BackingMap](#).

Throws:

[IllegalArgumentException](#) - thrown when the data format is not valid for the current configuration.

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

Since:

8.6, XC10 2.5

See Also:

[BackingMap.getKeyOutputFormat\(\)](#)

getValueOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getValueOutputFormat()**

Retrieves the data format for all data access APIs that return cache values for the current session.

Returns:

the data output format or null if the default should be used.

Since:

8.6, XC10 2.5

setValueOutputFormat

void **setValueOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)
throws [TransactionAlreadyActiveException](#)

Sets the data format for all data access APIs that return cache values for the current session.

This method is functionally equivalent to the [setCopyMode\(CopyMode, Class\)](#) with the [CopyMode.COPY_TO_BYTES](#) and [CopyMode.COPY_TO_BYTES_RAW](#) values when used with a ValueDataSerializer or eXtreme Data Format.

This method supports map configurations with a ValueDataSerializer plug-in defined, or with eXtreme Data Format enabled.

Parameters:

outputFormat - the data output format to use or `OutputFormat.UNDEFINED` to use the default defined on the parent [BackingMap](#).

Throws:

[IllegalArgumentException](#) - thrown when the data format is not valid for the current configuration.

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

Since:

8.6, XC10 2.5

setPutMode

```
void setPutMode(ObjectMap.PutMode putMode)
    throws TransactionAlreadyActiveException,
           ObjectGridException
```

Allows the default operation for the [put\(Object, Object\)](#) and [putAll\(Map\)](#) methods to be changed, allowing the put to use the optimized [upsert\(Object, Object\)](#) and [upsertAll\(LinkedHashMap\)](#) implementations.

The PutMode cannot be changed during a transaction. There must be no active transaction when this method is called.

Parameters:

putMode - the mode in which the put methods operate.

Throws:

[TransactionAlreadyActiveException](#) - if a transaction is active and this map has already been used in the transaction.

[ObjectGridException](#) - if an error occurs during processing

Since:

8.6, XC10 2.5

getPutMode

```
ObjectMap.PutMode getPutMode()
```

Retrieve the current PutMode set for this ObjectMap instance.

Returns:

the current PutMode.

Since:

8.6, XC10 2.5

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

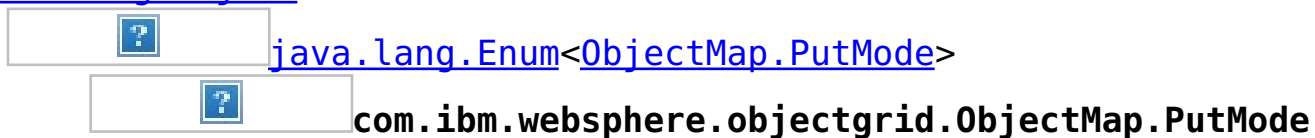
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Enum ObjectMap.PutMode

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#), [Comparable](#)<[ObjectMap.PutMode](#)>

Enclosing interface:

[ObjectMap](#)

```
public static enum ObjectMap.PutMode
extends Enum<ObjectMap.PutMode>
```

Identifies the operation mode of the [ObjectMap.put\(Object, Object\)](#), [ObjectMap.putAll\(Map\)](#), [JavaMap.put\(Object, Object\)](#) and [JavaMap.putAll\(Map\)](#) methods.

Since:

8.6, XC10 2.5

See Also:

[ObjectMap.setPutMode\(PutMode\)](#)

Enum Constant Summary

[INSERTUPDATE](#)

Deprecated. *Deprecated in 8.6. Use the [UPSERT](#) enumeration.*

[UPSERT](#)

The put API behaves like the upsert method.

Method Summary

s
t
a
t
i
c
O
b
j
e
c
t
M
a
p
.
P
u
t
M
o
d
e

[valueOf](#)([String](#) name)

Returns the enum constant of this type with the specified name.

d
e
s
t
a
t
i
c
O
b
j
e
c
t
M
e
t
h
o
d
s
[
]

[values\(\)](#)

Returns an array containing the constants of this enum type, in the order they are declared.

Methods inherited from class [java.lang.Enum](#)

[clone](#), [compareTo](#), [equals](#), [finalize](#), [getDeclaringClass](#), [hashCode](#), [name](#), [ordinal](#), [toString](#), [valueOf](#)

Methods inherited from class [java.lang.Object](#)

[getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Enum Constant Detail

INSERTUPDATE

```
public static final ObjectMap.PutMode INSERTUPDATE
```

Deprecated. *Deprecated in 8.6. Use the [UPSERT](#) enumeration.*
The put API behaves like an insert or update.

UPSERT

```
public static final ObjectMap.PutMode UPSERT
```

The put API behaves like the upsert method.

Method Detail

values

```
public static ObjectMap.PutMode[] values()
```

Returns an array containing the constants of this enum type, in the order they are declared. This method may be used to iterate over the constants as follows:

```
for (ObjectMap.PutMode c : ObjectMap.PutMode.values())  
    System.out.println(c);
```

Returns:

an array containing the constants of this enum type, in the order they are declared

valueOf

```
public static ObjectMap.PutMode valueOf(String name)
```

Returns the enum constant of this type with the specified name. The string must match *exactly* an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

Parameters:

name - the name of the enum constant to be returned.

Returns:

the enum constant with the specified name

Throws:

[IllegalArgumentException](#) - if this enum type has no constant with the specified name

[NullPointerException](#) - if the argument is null

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
SUMMARY: NESTED ENUM		DETAIL: ENUM					
CONSTANTS FIELD METHOD		CONSTANTS FIELD METHOD					

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class ObjectGridRuntimeException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[TransactionTimeoutException](#)

```
public class ObjectGridRuntimeException
extends RuntimeException
implements IObjectGridException
```

This exception is the base class for all runtime exceptions thrown by the cache.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

ObjectGridRuntimeException ()	Constructs a new ObjectGridRuntimeException with null as its detail message.
ObjectGridRuntimeException (String message)	Constructs a new ObjectGridRuntimeException with the specified detail message.
ObjectGridRuntimeException (String message, Throwable cause)	Constructs a new ObjectGridRuntimeException with the specified detail message and cause.
ObjectGridRuntimeException (Throwable cause)	Constructs a new ObjectGridRuntimeException with a specified cause.

Method Summary

I h r o w a b l e	getCause () Returns the cause of this ObjectGridRuntimeException or null if the cause is nonexistent or unknown.
---	---

[initCause](#)([Throwable](#) cause)

Initializes the *cause* of this `ObjectGridRuntimeException` to the specified value.

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridRuntimeException

```
public ObjectGridRuntimeException()
```

Constructs a new `ObjectGridRuntimeException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[initCause\(Throwable\)](#)

ObjectGridRuntimeException

```
public ObjectGridRuntimeException(String message)
```

Constructs a new `ObjectGridRuntimeException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridRuntimeException

```
public ObjectGridRuntimeException(Throwable cause)
```

Constructs a new `ObjectGridRuntimeException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ObjectGridRuntimeException`s that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[getCause\(\)](#)

ObjectGridRuntimeException

```
public ObjectGridRuntimeException(String message,  
                                Throwable cause)
```

Constructs a new ObjectGridRuntimeException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ObjectGridRuntimeException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A `null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[getCause\(\)](#), [Throwable.getMessage\(\)](#)

Method Detail

getCause

```
public Throwable getCause()
```

Returns the cause of this ObjectGridRuntimeException or `null` if the cause is nonexistent or unknown. (The cause is the throwable that caused this ObjectGridRuntimeException to get thrown.)

This implementation returns the cause that was supplied via one of the constructors requiring a `Throwable`, or that was set after creation with the `initCause(Throwable)` method. While it is typically unnecessary to override this method, a subclass can override it to return a cause set by some other means. This is appropriate for a "legacy chained throwable" that predates the addition of chained exceptions to `Throwable`. Note that it is *not* necessary to override any of the `PrintStackTrace` methods, all of which invoke the `getCause` method to determine the cause of an ObjectGridRuntimeException

Specified by:

[getCause](#) in interface [IObjectGridException](#)

Overrides:

[getCause](#) in class [Throwable](#)

Returns:

the cause of this ObjectGridRuntimeException or `null` if the cause is nonexistent or unknown.

See Also:

[ObjectGridRuntimeException\(String, Throwable\)](#), [ObjectGridRuntimeException\(Throwable\)](#), [initCause\(Throwable\)](#)

initCause

```
public Throwable initCause(Throwable cause)
```

Initializes the *cause* of this ObjectGridRuntimeException to the specified value. (The cause is the throwable that caused this ObjectGridRuntimeException to get thrown.)

This method can be called at most once. It is generally called from within the constructor, or immediately after creating the ObjectGridRuntimeException. If this

ObjectGridRuntimeException was created with ObjectGridRuntimeException(Throwable) or ObjectGridRuntimeException(String,Throwable), this method cannot be called even once.

Specified by:

[initCause](#) in interface [IObjectGridException](#)

Overrides:

[initCause](#) in class [Throwable](#)

Parameters:

cause - the cause (which is saved for later retrieval by the `getCause()` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown.)

Returns:

a reference to this ObjectGridRuntimeException instance.

Throws:

[IllegalArgumentException](#) - if cause is this ObjectGridRuntimeException. (An ObjectGridRuntimeException cannot be its own cause.)

[IllegalStateException](#) - if this ObjectGridRuntimeException was created with ObjectGridRuntimeException(Throwable) or ObjectGridRuntimeException(String,Throwable), or this method has already been called on this ObjectGridRuntimeException.

See Also:

[ObjectGridRuntimeException\(String, Throwable\)](#), [ObjectGridRuntimeException\(Throwable\)](#), [getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class ObjectGridManagerFactory

[java.lang.Object](#)

 com.ibm.websphere.objectgrid.ObjectGridManagerFactory

```
public final class ObjectGridManagerFactory  
extends Object
```

This factory class is a high level helper class to get ObjectGridManager instances.

Since:

WAS XD 6.0, XC10

Constructor Summary

[ObjectGridManagerFactory\(\)](#)

Method Summary

s
t
a
t
i
c
O
b
j
e
c
t
G
r
i
d
M
a
n
a
g
e
r

[getObjectGridManager\(\)](#)

Returns the ObjectGridManager singleton.

Methods inherited from class java.lang.Object

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [toString](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridManagerFactory

```
public ObjectGridManagerFactory()
```

Method Detail

getObjectGridManager

```
public static final ObjectGridManager getObjectGridManager()
```

Returns the ObjectGridManager singleton.

Returns:

The ObjectGridManager singleton

See Also:

[ObjectGridManager](#)

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Interface ObjectGridManager

```
public interface ObjectGridManager
```

ObjectGridManager is responsible for creating or retrieving local ObjectGrid instances and connecting to distributed ObjectGrid servers. Use the [ObjectGridManagerFactory](#) to retrieve an ObjectGridManager.

Use the createObjectGrid methods to create a local, in-memory ObjectGrid instance. The createObjectGrid methods give you the choice of caching the created ObjectGrid instance. If you choose to cache the instance, you cannot create an ObjectGrid with the same name unless you remove the previously created ObjectGrid using the removeObjectGrid(String) method. A cached ObjectGrid instance can later be retrieved using the getObjectGrid(String) method.

An example of creating a local in-memory ObjectGrid programmatically:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ObjectGrid grid = ogMgr.createObjectGrid("LocalBookStoreGrid");
grid.defineMap("Orders");
grid.defineMap("Books");
grid.initialize();
...
grid.destroy();
```

An example of creating a local in-memory ObjectGrid using an ObjectGrid descriptor XML file:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
URL objectgridXML = Thread.currentThread().getContextClassLoader().getResource("configs/objectgrid.xml");
ObjectGrid grid = ogMgr.createObjectGrid("LocalBookStoreGrid", objectgridXML);
grid.initialize();
...
ogMgr.destroy();
```

Use the connect methods to connect to a distributed ObjectGrid. The connect methods return a ClientClusterContext that can then be passed to one of the getObjectGrid methods, which will in turn retrieve a client ObjectGrid instance.

An example to connect to a dynamic, distributed ObjectGrid using a catalog server cluster:

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ClientClusterContext ccc = ogMgr.connect("catserver1:2809,catserver2:2809", null, null);
ObjectGrid grid = ogMgr.getObjectGrid(ccc, "BookStoreGrid");
...
ogMgr.disconnect(ccc);
```

An example to connect to an embedded ObjectGrid server (a server running in the current process):

```
ObjectGridManager ogMgr = ObjectGridManagerFactory.getObjectGridManager();
ClientClusterContext ccc = ogMgr.connect((ClientSecurityConfiguration) null, (URL) null);
ObjectGrid grid = ogMgr.getObjectGrid(ccc, "BookStoreGrid");
...
```



```
ogMgr.disconnect(ccc);
```

This interface also allows ObjectGrid trace to be disabled completely for performance improvements especially on a processor with a smaller L2 cache.

Since:

WAS XD 6.0, XC10

Method Summary	
ClientSecurityConfiguration	<p>connect(ClientSecurityConfiguration securityProps, URL overRideObjectGridXml) Use this method to connect to an embedded ObjectGrid server.</p>
String	<p>connect(String catalogServerEndpoints, ClientSecurityConfiguration securityProps, URL overRideObjectGridXml) Connects a client process to a remote ObjectGrid catalog server or catalog server cluster (a dynamic ObjectGrid deployment topology).</p>
ObjectGrid	<p>createObjectGrid() Creates a local, in-memory instance of an ObjectGrid.</p>
String	<p>createObjectGrid(String objectGridName) Creates and caches a local, in-memory ObjectGrid instance with the specified name.</p>

i
d

O
b
j
e
c
t
G
r
i
d

O
b
j
e
c
t
G
r
i
d

O
b
j
e
c
t
G
r
i
d

O
b
j
e
c
t
G
r
i
d

L
i
s
t

L
i
s
t

L
i
s
t

b
o
l
e
a
n

C
l
a

[createObjectGrid](#)([String](#) objectGridName, boolean cacheInstance)
Creates a local, in-memory instance of an ObjectGrid with the specified name.

[createObjectGrid](#)([String](#) objectGridName, [URL](#) xmlFile)
Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

[createObjectGrid](#)([String](#) objectGridName, [URL](#) xmlFile, boolean cacheInstance)
Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

[createObjectGrid](#)([String](#) objectGridName, [URL](#) xmlFile, boolean enableXmlValidation, boolean cacheInstance)
Deprecated. *Deprecated in version 8.6. XML validation is always enabled. Use [instead](#).*

[createObjectGrids](#)([URL](#) xmlFile)
Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

[createObjectGrids](#)([URL](#) xmlFile, boolean cacheInstances)
Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

[createObjectGrids](#)([URL](#) xmlFile, boolean enableXmlValidation, boolean cacheInstances)
Deprecated. *Deprecated in version 8.6. XML validation is always enabled. Use [createObjectGrids\(URL, boolean\)](#) instead.*

[disconnect](#)([ClientClusterContext](#) context)
Disconnects a client process from an ObjectGrid server

getCatalogDomainManager()

Retrieve the CatalogDomainManager for this ObjectGridManager.

getObjectGrid(ClientClusterContext context, String objectGridName)

Returns a client ObjectGrid instance corresponding to an ObjectGrid in an ObjectGrid cluster.

getObjectGrid(ClientClusterContext context, String objectGridName, ObjectGridConfiguration overrideOGConfig)

Returns a client ObjectGrid instance corresponding to an ObjectGrid in an ObjectGrid cluster.

getObjectGrid(String objectGridName)

Returns a cached local, in-memory ObjectGrid instance by name.

getObjectGrids()

Gets a List of the local, in-memory ObjectGrid instances that have been previously cached.

getOverrideObjectGridConfigurations()

Deprecated. This method is replaced in 7.1.1 with the getObjectGrid method that takes a ObjectGridConfigurations.

getTraceSpecification()

Retrieves the trace specification for the current JVM.

putOverrideObjectGridConfigurations(String clusterName, List objectGridConfigList)

Deprecated. This method is replaced in 7.1.1 with the getObjectGrid method that takes a ObjectGridConfigurations.

v o i d	removeObjectGrid (String objectGridName) Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances.
v o i d	removeObjectGrid (String objectGridName, boolean destroy) Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances and optionally destroys its associated resources.
v o i d	setOverrideObjectGridConfigurations (Map overrideMap) Deprecated. <i>This method is replaced in 7.1.1 with the getObjectGrid method that takes a ObjectGridConfigurations.</i>
v o i d	setTraceEnabled (boolean enabledFlag) Enables or disables ObjectGrid trace for the JVM.
v o i d	setTraceFileName (String traceFileName) Sets the trace output to go to a file instead of System.out.
v o i d	setTraceSpecification (String traceSpec) Set the trace specification for the current JVM.

Method Detail

createObjectGrid

[ObjectGrid](#) createObjectGrid()
throws [ObjectGridException](#)

Creates a local, in-memory instance of an ObjectGrid.

The created ObjectGrid returned by this method is assigned a unique name is **not** cached by the ObjectGridManager. Use the ObjectGrid.setName(String) method to change the ObjectGrid name.

Returns:

an instance of ObjectGrid with a unique name assigned

Throws:

[ObjectGridException](#) - if any error occurs during the ObjectGrid creation

See Also:

[ObjectGrid](#), [ObjectGrid.setName\(String\)](#)

createObjectGrid

[ObjectGrid](#) createObjectGrid([String](#) objectGridName,
boolean cacheInstance)
throws [ObjectGridException](#)

Creates a local, in-memory instance of an ObjectGrid with the specified name.

The instance of ObjectGrid created can optionally be cached. If an ObjectGrid with the specified name was previously created and cached, an ObjectGridException will be thrown.

Parameters:

objectGridName - the name of the ObjectGrid to be created. Must not be null.
cacheInstance - true, if the ObjectGrid instance should be cached

Returns:

an ObjectGrid instance with the specified name.

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with this name has already been cached or any error occurs during the ObjectGrid creation.

See Also:

[ObjectGrid](#)

createObjectGrid

[ObjectGrid](#) createObjectGrid([String](#) objectGridName)
throws [ObjectGridException](#)

Creates and caches a local, in-memory ObjectGrid instance with the specified name.

The ObjectGrid instance created by this method will be cached. Invoking this method is equivalent to invoke createObjectGrid(String, true)

Parameters:

objectGridName - the Name of the ObjectGrid instance to be created. Must not be null.

Returns:

an ObjectGrid instance with the specified name

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with this name has already been cached, or any error occurs during the ObjectGrid creation

See Also:

[createObjectGrid\(String, boolean\)](#), [ObjectGrid](#)

createObjectGrid

[@Deprecated](#)

[ObjectGrid](#) createObjectGrid([String](#) objectGridName,
[URL](#) xmlFile,
boolean enableXmlValidation,
boolean cacheInstance)
throws [ObjectGridException](#)

Deprecated. *Deprecated in version 8.6. XML validation is always enabled. Use [instead](#).*

Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

An ObjectGrid instance is created for the ObjectGrid configuration in the XML file corresponding to the specified ObjectGrid name. If the specified ObjectGrid name cannot be found in the XML file, an exception will be thrown.

This returned ObjectGrid instance optionally can be cached.

If the URL is null, it will be simply ignored. In this case, this method behaves the same as the createObjectGrid(String, boolean).

Parameters:

objectGridName - the Name of the ObjectGrid instance to be returned. Must not be null.

xmlFile - a URL to a well formed xml file based on the ObjectGrid schema.

enableXmlValidation - if true the XML is validated

cacheInstance - a boolean value indicating whether the returned ObjectGrid instance

defined in the XML will be cached or not. If true, the instance will be cached.

Returns:

an ObjectGrid instance

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with the same name has been previously cached, an ObjectGrid configuration with the specified name can be found in the xml file, or any other error occurs during ObjectGrid creation.

See Also:

[createObjectGrid\(String, boolean\)](#), [ObjectGrid](#)

createObjectGrid

[ObjectGrid](#) createObjectGrid([String](#) objectGridName,
[URL](#) xmlFile,
boolean cacheInstance)
throws [ObjectGridException](#)

Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

An ObjectGrid instance is created for the ObjectGrid configuration in the XML file corresponding to the specified ObjectGrid name. If the specified ObjectGrid name cannot be found in the XML file, an exception will be thrown.

This returned ObjectGrid instance optionally can be cached.

If the URL is null, it will be simply ignored. In this case, this method behaves the same as the createObjectGrid(String, boolean).

Parameters:

objectGridName - the Name of the ObjectGrid instance to be returned. Must not be null.

xmlFile - a URL to a well formed xml file based on the ObjectGrid schema.

cacheInstance - a boolean value indicating whether the returned ObjectGrid instance defined in the XML will be cached or not. If true, the instance will be cached.

Returns:

an ObjectGrid instance

Throws:

[IllegalArgumentException](#) - if objectGridName is null

[ObjectGridException](#) - if an ObjectGrid with the same name has been previously cached, an ObjectGrid configuration with the specified name can be found in the xml file, or any other error occurs during ObjectGrid creation.

Since:

8.6, XC10 2.5

See Also:

[createObjectGrid\(String, boolean\)](#), [ObjectGrid](#)

createObjectGrids

[List](#) createObjectGrids([URL](#) xmlFile,
boolean enableXmlValidation,
boolean cacheInstances)
throws [ObjectGridException](#)

Deprecated. *Deprecated in version 8.6. XML validation is always enabled. Use [createObjectGrids\(URL, boolean\)](#) instead.*

Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

The returned ObjectGrid instances can be cached. An ObjectGridException will be thrown when attempting to cache a newly created ObjectGrid that has the same name as an ObjectGrid that has already been cached.

Parameters:

xmlFile - the file that defines an ObjectGrid or multiple ObjectGrids

cacheInstances - set to true to cache all ObjectGrid instances created based on the file

Returns:

a list of ObjectGrid instances

Throws:

[ObjectGridException](#) - if attempting to create and cache an ObjectGrid with the same name as an ObjectGrid that has already been cached, or any other error occurs during ObjectGrid creation

See Also:

[ObjectGrid](#)

createObjectGrids

[List](#) createObjectGrids([URL](#) xmlFile,
boolean cacheInstances)
throws [ObjectGridException](#)

Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

The returned ObjectGrid instances can be cached. An ObjectGridException will be thrown when attempting to cache a newly created ObjectGrid that has the same name as an ObjectGrid that has already been cached.

Parameters:

xmlFile - the file that defines an ObjectGrid or multiple ObjectGrids

cacheInstances - set to true to cache all ObjectGrid instances created based on the file

Returns:

a list of ObjectGrid instances

Throws:

[ObjectGridException](#) - if attempting to create and cache an ObjectGrid with the same name as an ObjectGrid that has already been cached, or any other error occurs during ObjectGrid creation

Since:

8.6, XC10 2.5

See Also:

[ObjectGrid](#)

createObjectGrids

[List](#) createObjectGrids([URL](#) xmlFile)
throws [ObjectGridException](#)

Creates a List of a local, in-memory ObjectGrid instances based upon the ObjectGrid configurations in the specified ObjectGrid descriptor XML file.

The XML file will be validated against the schema and each ObjectGrid instance that is created will be cached. An ObjectGridException will be thrown when attempting to cache a newly created ObjectGrid that has the same name as an ObjectGrid that has already been cached. Using this method is equivalent to calling the createObjectGrids(URL, true, true) method.

Parameters:

xmlFile - The XML file to process. ObjectGrid(s) will be created based on the configurations what is in the file.

Returns:

A list of ObjectGrid instances that have been created.

Throws:

[ObjectGridException](#) - if attempting to create and cache an ObjectGrid with the same name as an ObjectGrid that has already been cached, or any other error occurs during ObjectGrid creation

See Also:

[createObjectGrids\(URL, boolean, boolean\)](#), [ObjectGrid](#)

createObjectGrid

```
ObjectGrid createObjectGrid(String objectGridName,  
                             URL xmlFile)  
throws ObjectGridException
```

Creates a local, in-memory ObjectGrid instance based on the specified ObjectGrid name and the ObjectGrid XML configuration file.

If there is no ObjectGrid with this name defined in the XML file, an ObjectGridException will be thrown. The XML file will be validated against the schema and the ObjectGrid instance created will be cached. Using this method is equivalent to calling the createObjectGrid(String, URL, true, true) method.

Parameters:

objectGridName - name of the ObjectGrid to create. This ObjectGrid should be defined in the XML file. Must not be null.
xmlFile - the XML file to process

Returns:

A newly created ObjectGrid

Throws:

[IllegalArgumentException](#) - if objectGridName is null
[ObjectGridException](#) - if an ObjectGrid with the same name has been previously cached, an ObjectGrid configuration with the specified name can be found in the xml file, or any other error occurs during ObjectGrid creation.

See Also:

[createObjectGrid\(String, URL, boolean, boolean\)](#), [ObjectGrid](#)

removeObjectGrid

```
void removeObjectGrid(String objectGridName)  
throws ObjectGridException
```

Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances.

Invoking this method is equivalent to calling removeObjectGrid(String, false)

Parameters:

objectGridName - the name of the ObjectGrid instance to remove from the cache

Throws:

[ObjectGridException](#) - if an ObjectGrid with the objectGridName was not found in the cache

See Also:

[removeObjectGrid\(String, boolean\)](#)

removeObjectGrid

```
void removeObjectGrid(String objectGridName,  
                     boolean destroy)  
throws ObjectGridException
```

Removes a local, in-memory ObjectGrid from the cache of ObjectGrid instances and optionally destroys its associated resources.

Parameters:

objectGridName - the name of the ObjectGrid instance to remove from the cache
destroy - if true, destroy the objectgrid instance and its associated resources

Throws:

[ObjectGridException](#) - if an ObjectGrid with the objectGridName was not found in the cache

See Also:

[ObjectGrid.destroy\(\)](#)

getObjectGrids

[List](#) getObjectGrids()

Gets a List of the local, in-memory ObjectGrid instances that have been previously cached.

This method returns null if no ObjectGrid instances have been cached.

Returns:

a list of ObjectGrid instances that have been previously cached or null if there are no cached ObjectGrid instances

getObjectGrid

[ObjectGrid](#) getObjectGrid([String](#) objectGridName)

Returns a cached local, in-memory ObjectGrid instance by name.

This method returns null if no ObjectGrid with the specified name is currently cached.

Parameters:

objectGridName - the cached objectgrid name.

Returns:

a cached ObjectGrid which currently exists.

setTraceSpecification

void setTraceSpecification([String](#) traceSpec)

Set the trace specification for the current JVM.

This operation is a replace operation, not an append operation. The specification should be of the form:

```
TraceString := <ComponentString>(:<ComponentString>)* ComponentString := <ComponentName>=<type>=<state>(,<type>=<state>)*  
ComponentName := a java String state := [enabled|disabled] type := [all|debug|event|entryExit]
```

For example, ObjectGrid=all=enabled

Parameters:

traceSpec - the new trace specification

getTraceSpecification

[String](#) getTraceSpecification()

Retrieves the trace specification for the current JVM.

Since:

7.1

See Also:

[setTraceSpecification\(String\)](#)

setTraceFileName

void **setTraceFileName**([String](#) traceFileName)

Sets the trace output to go to a file instead of System.out.

The supplied file name can be relative to the working directory or a fully-qualified file name.

Parameters:

traceFileName - Name of trace file

setTraceEnabled

void **setTraceEnabled**(boolean enabledFlag)

Enables or disables ObjectGrid trace for the JVM.

Disabling trace improves the performance when ObjectGrid runs on processors whose L2 caches are not large enough to contain the trace enabled code paths. If this is set to false, ObjectGrid trace is suppressed even if it is enabled using [setTraceSpecification\(String\)](#). By default ObjectGrid trace is enabled.

Parameters:

enabledFlag - true to enable trace

Since:

WAS XD 6.0.1

See Also:

[setTraceSpecification\(String\)](#)

connect

[ClientClusterContext](#) **connect**([ClientSecurityConfiguration](#) securityProps,
[URL](#) overrideObjectGridXml)
throws [ConnectException](#)

Use this method to connect to an embedded ObjectGrid server. An embedded ObjectGrid server is typically started in an application server process such as IBM WebSphere Application Server. This method allows connecting to the in-memory ObjectGrid server instance without specifying connection information.

This method can be used to connect to both dynamic and static ObjectGrid server deployments.

Parameters:

securityProps - client security configuration. The value can be null if not running in secure mode.

overrideObjectGridXml - This parameter can be null. If it is not null, the client side configuration of the ObjectGrid plugins are overridden with the ObjectGrid configuration using this URL. If this parameter is null, client side ObjectGrid plugins can be overridden by providing an overrideMap to [setOverrideObjectGridConfigurations\(Map\)](#) or by calling [putOverrideObjectGridConfigurations\(String, List\)](#).

In cases where this parameter is not null, and overriding configuration objects have been provided to the ObjectGridManager by `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)`, the configuration based on the XML file will be used to override ObjectGrid settings. Overriding objects provided to `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)` will be ignored.

Not all plugins can be overridden. For details please see the ObjectGrid documentation.

Returns:

a `ClientClusterContext` representing the cluster ObjectGrid configuration to which the client is connected.

Throws:

[ConnectException](#) - if any error occurs connecting to the server

Since:

WAS XD 6.0.1

See Also:

[ClientClusterContext](#), [ClientSecurityConfiguration](#)

connect

[ClientClusterContext](#) `connect`([String](#) catalogServerEndpoints,
[ClientSecurityConfiguration](#) securityProps,
[URL](#) overRideObjectGridXml)
throws [ConnectException](#)

Connects a client process to a remote ObjectGrid catalog server or catalog server cluster (a dynamic ObjectGrid deployment topology). The result `ClientClusterContext` can then be used to get any ObjectGrid reference managed by that catalog server cluster.

Parameters:

`catalogServerEndpoints` - A concatenated list of host/port pairs belonging to the catalog servers in the form "host:port<,host:port>". This list can be arbitrarily long and is used for bootstrapping only. The first viable address will be used.

`securityProps` - This parameter may be null if the client does not wish to establish a secure connection with the server.

`overRideObjectGridXml` - This parameter can be null. If it is not null, the client side configuration of the ObjectGrid plugins are overridden with the ObjectGrid configuration using this URL. If this parameter is null, client side ObjectGrid plugins can be overridden by providing an `overrideMap` to `setOverrideObjectGridConfigurations(Map)` or by calling `putOverrideObjectGridConfigurations(String, List)`.

In cases where this parameter is not null, and overriding configuration objects have been provided to the ObjectGridManager by `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)`, the configuration based on the XML file will be used to override ObjectGrid settings. Overriding objects provided to `setOverrideObjectGridConfigurations(Map)` or `putOverrideObjectGridConfigurations(String, List)` will be ignored.

Not all plugins can be overridden. For details please see the ObjectGrid documentation.

Returns:

a `ClientClusterContext` representing the cluster ObjectGrid configuration to which the client is connected.

Throws:

[ConnectException](#) - If there is a problem connecting to the addresses given.

disconnect

boolean **disconnect**([ClientClusterContext](#) context)

Disconnects a client process from an ObjectGrid server

Parameters:

context - the [ClientClusterContext](#) object returned from a previous call to one of the connect methods The caller must guarantee this parameter is not null.

Returns:

true if the disconnect was successful, false if the supplied context was not connected

Throws:

[IllegalArgumentException](#) - if the [ClientClusterContext](#) is null

Since:

WAS XD 6.0.1

See Also:

[ClientClusterContext](#)

getObjectGrid

[ObjectGrid](#) **getObjectGrid**([ClientClusterContext](#) context,
[String](#) objectGridName)

Returns a client [ObjectGrid](#) instance corresponding to an [ObjectGrid](#) in an [ObjectGrid](#) cluster.

This method is equivalent to calling `getObjectGrid(context, objectGridName, null)`

Parameters:

context - the [ClientClusterContext](#) object returned from a previous call to one of the connect methods The caller must guarantee this parameter is not null.

objectGridName - the name of the requested client [ObjectGrid](#)

Returns:

a client [ObjectGrid](#) instance corresponding to a remote [ObjectGrid](#)

Throws:

[IllegalArgumentException](#) - if either provided parameter is null

[ObjectGridRuntimeException](#) - if the [ObjectGrid](#) with the specified name is not hosted in any eXtreme Scale servers managed by the catalog server

Since:

WAS XD 6.0.1

See Also:

[ClientClusterContext](#), [ObjectGrid](#), [getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

getObjectGrid

[ObjectGrid](#) **getObjectGrid**([ClientClusterContext](#) context,
[String](#) objectGridName,
[ObjectGridConfiguration](#) overrideOGConfig)

Returns a client [ObjectGrid](#) instance corresponding to an [ObjectGrid](#) in an [ObjectGrid](#) cluster.

This method replaces the `get/set/putOverrideObjectGridConfigurations` methods. Those methods had thread safety issues. In addition they were global in nature so we end up having configuration override happen for all client connections unless it was managed correctly. If [ClientClusterContext](#) was used previously to get an [ObjectGrid](#) for the given name, the same [ObjectGrid](#) instance is returned even if the `overrideOGConfig` parameter is different.

Parameters:

context - the `ClientClusterContext` object returned from a previous call to one of the connect methods. The caller must guarantee this parameter is not `null`.
objectGridName - the name of the requested client `ObjectGrid`
overrideOGConfig - This parameter can be `null`. If it is not `null`, the client side configuration of the `ObjectGrid` plugins are overridden with the `ObjectGridConfiguration` provided. The provided override configuration takes precedence over any other provided override configuration for the requested `ObjectGrid` name provided by the `connect`, `putOverrideObjectGridConfigurations`, and `putOverrideObjectGridConfigurations` methods.

Not all plugins can be overridden. For details please see the `ObjectGrid` documentation.

Returns:

a client `ObjectGrid` instance corresponding to a remote `ObjectGrid`

Since:

7.1.1

See Also:

[ObjectGridConfiguration](#), [ObjectGridConfigFactory](#)

setOverrideObjectGridConfigurations

```
void setOverrideObjectGridConfigurations(Map overrideMap)
```

Deprecated. *This method is replaced in 7.1.1 with the `getObjectGrid` method that takes a `ObjectGridConfigurations`.*

Override `ObjectGrid` settings on client side `ObjectGrids` by passing in a `Map` where each key corresponds to a cluster name or domain name and each value is a `List` of `ObjectGridConfiguration` objects to be overridden.

Client side configuration of `ObjectGrid` and `BackingMap` plugins are overridden using the `ObjectGridConfiguration` values provided in the `List`. To override a `Plugin`, each `ObjectGridConfiguration` object must have a name that matches the name of the `ObjectGrid` to be overridden. `BackingMapConfiguration` objects must have the same name as a `BackingMap` and be associated with the proper `ObjectGridConfiguration`.

Not all plugins can be overridden. `ObjectGrid` plugins that can be overridden on the client side are `TransactionCallback` and `ObjectGridEventListener`. `BackingMap` plugins that can be overridden on the client side are `Evictor` and `MapEventListener`. Settings for the builtin `Evictor` can also be altered on the `BackingMap`. These settings include `numberOfBuckets`, `timeToLive`, and `ttlEvictorType`.

Parameters:

overrideMap - a `Map` that will be used to override `ObjectGrid` settings on the client side. To override client side settings, each key of the `Map` must be a `String` with a value that corresponds to a cluster name or domain name. Each value of the `overrideMap` must be a `java.util.List`. The `List` elements must be `ObjectGridConfiguration` objects. Each call to a connect method with a `clusterName` that matches a key in the `overrideMap` will result in the client side settings being overridden using the `List` of `ObjectGridConfiguration` objects provided in the key's corresponding value. Pass in `null` to clear an `overrideMap` that was previously set and thereby remove any overriding settings from future connect calls.

Throws:

[IllegalArgumentException](#) - if any keys or values are `null` or if keys or values are of the wrong type

Since:

WAS XD 6.0.1.2

See Also:

[connect\(String, ClientSecurityConfiguration, URL\)](#), [connect\(ClientSecurityConfiguration,](#)

[URL](#)), [ObjectGridConfiguration](#), [BackingMapConfiguration](#),
[getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

putOverrideObjectGridConfigurations

```
void putOverrideObjectGridConfigurations(String clusterName,  
                                         List objectGridConfigList)
```

Deprecated. This method is replaced in 7.1.1 with the *getObjectGrid* method that takes a *ObjectGridConfigurations*.

Put an entry into the Map that is used to override client side ObjectGrid and BackingMap plugins.

Parameters:

clusterName - to be used as a key in the Map used to override client side ObjectGrid plugins. If a connect method is called with a matching clusterName, the client side ObjectGrid and BackingMap plugins can be overridden using the objectGridConfigList. In the dynamic environment, use the domain name to override ObjectGrid settings.

objectGridConfigList - a List of ObjectGridConfiguration objects that will be used to override client side ObjectGrid settings if a connect method is called with a cluster name that matches the clusterName on this method

Throws:

[IllegalArgumentException](#) - if the clusterName or objectGridConfigList is null

See Also:

[getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

getOverrideObjectGridConfigurations

```
Map getOverrideObjectGridConfigurations()
```

Deprecated. This method is replaced in 7.1.1 with the *getObjectGrid* method that takes a *ObjectGridConfigurations*.

Get the Map that is used to override client side ObjectGrid and BackingMap plugins.

Returns:

the Map that was set by the call to setOverrideObjectGridConfigurations. The Map may also have entries that were put there using the putOverrideObjectGridConfigurations method.

See Also:

[getObjectGrid\(ClientClusterContext, String, ObjectGridConfiguration\)](#)

getCatalogDomainManager

```
CatalogDomainManager getCatalogDomainManager()
```

Retrieve the CatalogDomainManager for this ObjectGridManager.

Returns:

the CatalogDomainManager, if available. Returns null if a CatalogDomainManager is not supported in the current runtime environment.

Since:

8.5

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class ObjectGridException

[java.lang.Object](#)

 [java.lang.Throwable](#)

 [java.lang.Exception](#)

 [com.ibm.websphere.objectgrid.ObjectGridException](#)

All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[CacheEntryException](#), [ConnectException](#), [ContinuousQueryException](#),
[DuplicateKeyException](#), [KeyNotFoundException](#), [LoaderException](#), [LockException](#),
[NoActiveTransactionException](#), [ObjectGridConfigurationException](#),
[ObjectGridSecurityException](#), [ReadOnlyException](#), [TransactionCallbackException](#),
[TransactionException](#), [UndefinedMapException](#)

```
public class ObjectGridException  
extends Exception  
implements IObjectGridException
```

Base exception class for all checked exceptions thrown by the ObjectGrid product.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[ObjectGridException](#)()

Constructs a new ObjectGridException with null as its detail message.

[ObjectGridException](#)(String message)

Constructs a new ObjectGridException with the specified detail message.

[ObjectGridException](#)(String message, Throwable cause)

Constructs a new ObjectGridException with the specified detail message and cause.

[ObjectGridException](#)(Throwable cause)

Constructs a new ObjectGridException with a specified cause.

Method Summary

[I](#)
[h](#)
[r](#)
[o](#)
[w](#)
[a](#)
[b](#)

[getCause](#)()

Returns the cause of this ObjectGridException or null if the cause is nonexistent or unknown.

l
e

I
h
r
o
w
a
b
l
e

[initCause\(Throwable cause\)](#)

Initializes the *cause* of this ObjectGridException to the specified value.

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ObjectGridException

```
public ObjectGridException()
```

Constructs a new ObjectGridException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[initCause\(Throwable\)](#)

ObjectGridException

```
public ObjectGridException(String message)
```

Constructs a new ObjectGridException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ObjectGridException

```
public ObjectGridException(Throwable cause)
```

Constructs a new ObjectGridException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for ObjectGridExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:
[getCause\(\)](#)

ObjectGridException

```
public ObjectGridException(String message,  
                           Throwable cause)
```

Constructs a new ObjectGridException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ObjectGridException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[getCause\(\)](#), [Throwable.getMessage\(\)](#)

Method Detail

getCause

```
public Throwable getCause()
```

Returns the cause of this ObjectGridException or `null` if the cause is nonexistent or unknown. (The cause is the throwable that caused this ObjectGridException to get thrown.)

This implementation returns the cause that was supplied via one of the constructors requiring a `Throwable`, or that was set after creation with the `initCause(Throwable)` method. While it is typically unnecessary to override this method, a subclass can override it to return a cause set by some other means. This is appropriate for a "legacy chained throwable" that predates the addition of chained exceptions to `Throwable`. Note that it is *not* necessary to override any of the `PrintStackTrace` methods, all of which invoke the `getCause` method to determine the cause of an ObjectGridException

Specified by:

[getCause](#) in interface [IObjectGridException](#)

Overrides:

[getCause](#) in class [Throwable](#)

Returns:

the cause of this ObjectGridException or `null` if the cause is nonexistent or unknown.

See Also:

[ObjectGridException\(String, Throwable\)](#), [ObjectGridException\(Throwable\)](#),
[initCause\(Throwable\)](#)

initCause

```
public Throwable initCause(Throwable cause)
```

Initializes the *cause* of this ObjectGridException to the specified value. (The cause is the throwable that caused this ObjectGridException to get thrown.)

This method can be called at most once. It is generally called from within the constructor, or immediately after creating the ObjectGridException. If this

ObjectGridException was created with ObjectGridException(Throwable) or ObjectGridException(String,Throwable), this method cannot be called even once.

Specified by:

[initCause](#) in interface [IObjectGridException](#)

Overrides:

[initCause](#) in class [Throwable](#)

Parameters:

cause - the cause (which is saved for later retrieval by the `getCause()` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown.)

Returns:

a reference to this ObjectGridException instance.

Throws:

[IllegalArgumentException](#) - if cause is this ObjectGridException. (An ObjectGridException cannot be its own cause.)

[IllegalStateException](#) - if this ObjectGridException was created with ObjectGridException(Throwable) OR ObjectGridException(String,Throwable), or this method has already been called on this ObjectGridException.

See Also:

[ObjectGridException\(String, Throwable\)](#), [ObjectGridException\(Throwable\)](#), [getCause\(\)](#)

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
ew	ge	ss	ed	ted			
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All		
			Classes				

SUMMARY: NESTED | FIELD | [CONSTR](#) | [METH](#) | [OD](#) DETAIL: FIELD | [CONSTR](#) | [METHOD](#)

com.ibm.websphere.objectgrid
Interface ObjectGrid

public interface **ObjectGrid**

This object is used for creating sessions to the ObjectGrid. It is the central core of the ObjectGrid framework. Besides creating sessions, it is also responsible for defining BackingMaps, setting a TransactionCallback, adding event listeners, and managing the security settings.

Since:
 WAS XD 6.0, XC10

Field Summary	
s t a t i c i n t	<p>CLIENT Indicates the ObjectGrid is a client-side distributed ObjectGrid.</p>
s t a t i c i n t	<p>DEFAULT_TX_TIMEOUT_VALUE The default transaction time out value of 10 minutes if no transaction time out value is set.</p>
s t a t i c i n t	<p>LOCAL Indicates the ObjectGrid is a local ObjectGrid.</p>
s t a t i c i n t	<p>SERVER Indicates the ObjectGrid is a server-side distributed ObjectGrid.</p>

Method Summary

v
o
i
d

[addEventListener](#)(com.ibm.websphere.objectgrid.plugins.EventListener listener)
Adds an EventListener.

v
o
i
d

[addEventListener](#)(ObjectGridEventListener listener)
Deprecated. *This method is deprecated in version 7.1.1, use the [addEventListener\(EventListener\)](#) method.*

B
a
c
k
i
n
g
M
a
p

[createMap](#)(String name)
Creates a BackingMap, but does not associate it with this ObjectGrid.

B
a
c
k
i
n
g
M
a
p

[defineMap](#)(String name)
Defines a BackingMap that will be used by the application.

v
o
i
d

[destroy](#)()
Destroys this instance.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
r
e
v
i
s
i
o
n

[getCollisionArbiter](#)()
Retrieves the CollisionArbiter that this grid is using to resolve revision collisions.

C
o
l
l
i
s
i
o
n
A
r
b
i
t
r

L
i
s
t

[getEventListeners\(\)](#)
Returns the current list of EventListeners.

L
i
s
t

[getListOfMapNames\(\)](#)
Gets the list of map names currently defined for this ObjectGrid instance.

B
a
c
k
i
n
g
M
a
p

[getMap\(String name\)](#)
Returns a BackingMap previously configured by calling the defineMap(String) or setMaps(List) method.

S
t
r
i
n
g

[getName\(\)](#)
Gets the name of this ObjectGrid.

i
n
t

[getObjectGridType\(\)](#)
Returns the type of ObjectGrid.

S
e
s
s
i
o
n

[getSession\(\)](#)
Gets a Session object that can be used by a single thread at a time.

S
e
s
s
i
o
n

[getSession\(CredentialGenerator credGen\)](#)
Get a session using a CredentialGenerator.

S
e
s
s
i
o
n

[getSession\(Subject subject\)](#)
Allows the use of a specific Subject rather than use the SubjectSource configured on the ObjectGrid to get a Session.

[getState\(\)](#)

Retrieve the current life cycle state of this ObjectGrid.

a c t i o n c a l l b a c k	<p>getTransactionCallback() Retrieves the TransactionCallback object.</p>
i n t	<p>getTxIsolation() Retrieves the default transaction isolation level.</p>
i n t	<p>getTxTimeout() Gets transaction timeout setting for this ObjectGrid instance.</p>
v o i d	<p>initialize() Begins the bootstrapping of the ObjectGrid and Session instances.</p>
b o l e a n	<p>isSecurityEnabled() Checks whether security is enabled on this ObjectGrid or not.</p>
v o i d	<p>registerEntities(Class[] entities) Register one or more entities based on the class metadata.</p>
v o i d	<p>registerEntities(URL entityXML) Registers one or more entities from an entity XML file.</p>
v o i d	<p>removeEventListener(com.ibm.websphere.objectgrid.plugins.EventListener listener) Removes an EventListener.</p>
v o i d	<p>removeEventListener(ObjectGridEventListener listener) Deprecated. <i>This method is deprecated in version 7.1.1, use the removeEventListener(EventListener) method.</i></p>
i n t	<p>reserveSlot(String containerName) Allows plugins on this ObjectGrid to reserve slots for use to store context data.</p>
v o i d	<p>setAccessByCreatorOnlyMode(int accessByCreatorOnlyMode) Set the "access by creator only" mode.</p>
v o i d	<p>setAuthorizationMechanism(int authMechanism) Sets the authorization mechanism.</p>
v	<p>setCollisionArbiter(com.ibm.websphere.objectgrid.revision.CollisionArbiter arbiter)</p>

o i d	Sets the CollisionArbiter that is responsible for arbitration of revision conflicts.
v o i d	<p>setEventListeners(List listeners)</p> <p>Deprecated. <i>This method is deprecated in version 7.1.1. Use the addEventListener(EventListener) or removeEventListener(EventListener) methods. Plug-ins that implement the ObjectGridLifecycleListener interface are automatically registered with the grid. Using this method will remove those automatically added listeners.</i></p>
v o i d	<p>setMaps(List mapList)</p> <p>Clears any BackingMaps that have been previously defined on this ObjectGrid and replaces them with the List of BackingMaps provided.</p>
v o i d	<p>setName(String gridName)</p> <p>Sets the name of this ObjectGrid.</p>
v o i d	<p>setObjectGridAuthorization(com.ibm.websphere.objectgrid.security.plugins.ObjectGridAuthorization ogAuthorization)</p> <p>Sets the ObjectGridAuthorization for this ObjectGrid instance.</p>
v o i d	<p>setPermissionCheckPeriod(int period)</p> <p>Sets the permission check period.</p>
v o i d	<p>setQueryConfig(QueryConfig queryConfig)</p> <p>Set the QueryConfig object for this ObjectGrid.</p>
v o i d	<p>setSecurityEnabled()</p> <p>Enables the ObjectGrid security.</p>
v o i d	<p>setSubjectSource(com.ibm.websphere.objectgrid.security.plugins.SubjectSource source)</p> <p>Sets the SubjectSource plugin.</p>
v o i d	<p>setSubjectValidation(com.ibm.websphere.objectgrid.security.plugins.SubjectValidation subjectValidation)</p> <p>Sets the SubjectValidation for this ObjectGrid instance.</p>
v o i d	<p>setTransactionCallback(TransactionCallback callback)</p> <p>Sets the TransactionCallback object.</p>
v o i d	<p>setTxIsolation(int level)</p> <p>Sets the default transaction isolation level for all sessions created by the ObjectGrid.</p>
v o i d	<p>setTxTimeout(int timeout)</p> <p>Sets the transaction timeout value to a specified number of seconds.</p>

Field Detail

DEFAULT_TX_TIMEOUT_VALUE

static final int **DEFAULT_TX_TIMEOUT_VALUE**

The default transaction time out value of 10 minutes if no transaction time out value is set.

Since:

WXS 7.1.0.0 FIX1

See Also:

[Constant Field Values](#)

LOCAL

static final int **LOCAL**

Indicates the ObjectGrid is a local ObjectGrid.

See Also:

[Constant Field Values](#)

SERVER

static final int **SERVER**

Indicates the ObjectGrid is a server-side distributed ObjectGrid.

See Also:

[Constant Field Values](#)

CLIENT

static final int **CLIENT**

Indicates the ObjectGrid is a client-side distributed ObjectGrid.

See Also:

[Constant Field Values](#)

Method Detail

getSession

[Session](#) getSession()

throws [ObjectGridException](#),
[TransactionCallbackException](#)

Gets a Session object that can be used by a single thread at a time.

It is not allowed to share this Session object between threads without placing a critical section around it. While the core framework allows the object to move between threads, the TransactionCallback and Loader may prevent this usage, especially in J2EE environments.

When the ObjectGrid is a local ObjectGrid, and its security is enabled, this method will use the SubjectSource to get a Subject object and then associate the Subject object with this session .

When the ObjectGrid is a distributed ObjectGrid (client server mode), and its security is

enabled, this method will utilize the client server security infrastructure to get a secure session.

If the `initialize()` method has not been invoked prior to the first `getSession` invocation, an implicit initialization will occur. This ensures that all of the configuration is complete before any runtime usage is required.

Returns:

An instance of `Session`

Throws:

[ObjectGridException](#) - if an error occurs during processing

[TransactionCallbackException](#) - if the `TransactionCallback` throws an exception

[IllegalStateException](#) - if this method is called after the `destroy()` method is called.

See Also:

[destroy\(\)](#), [initialize\(\)](#), [Session](#), `SubjectSource`

getSession

[Session](#) `getSession(Subject subject)`
throws [ObjectGridException](#),
[TransactionCallbackException](#),
`com.ibm.websphere.objectgrid.security.plugins.InvalidSubjectException`

Allows the use of a specific `Subject` rather than use the `SubjectSource` configured on the `ObjectGrid` to get a `Session`.

This method should only be used when `ObjectGrid` security is enabled. If the `ObjectGrid` security is disabled, the provided `Subject` object will not be used.

If the `initialize()` method has not been invoked prior to the first `getSession` invocation, an implicit initialization will occur. This ensures that all of the configuration is complete before any runtime usage is required.

Parameters:

`subject` - `Subject` to associate with the returned `Session`

Returns:

An instance of `Session`

Throws:

[ObjectGridException](#) - if an error occurs during processing

[TransactionCallbackException](#) - if the `TransactionCallback` throws an exception

`com.ibm.websphere.objectgrid.security.plugins.InvalidSubjectException` - the subject passed in is invalid based on the `SubjectValidation` mechanism.

[IllegalStateException](#) - if this method is called after the `destroy()` method is called.

See Also:

[destroy\(\)](#), [initialize\(\)](#), [Session](#), `SubjectValidation`

setTransactionCallback

`void setTransactionCallback(TransactionCallback callback)`

Sets the `TransactionCallback` object.

A single cache is a single domain. All `Loaders` defined for `BackingMaps` in an `ObjectGrid` will normally cooperate, thus a corresponding `TransactionCallback` object needs to be set on the `ObjectGrid`.

A `TransactionCallback` that implements the `ObjectGridLifecycleListener` interface is automatically added as if the [addEventListener\(EventListener\)](#) method was called. Any previous callback which implements `ObjectGridLifecycleListener` interface is removed as if the [removeEventListener\(EventListener\)](#) method was called.

A TransactionCallback may implement the ObjectGridPlugin interface in order to receive enhanced ObjectGrid plug-in lifecycle method calls. The plug-in is also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

`callback` - An instance of a TransactionCallback

Throws:

[IllegalArgumentException](#) - if `callback` is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#), [TransactionCallback](#)

getTransactionCallback

[TransactionCallback](#) `getTransactionCallback()`

Retrieves the TransactionCallback object.

The TransactionCallback can be used in conjunction with the TXID to house transaction-specific context data, such as the connection to the database.

Returns:

the argument that was passed to the `setTransactionCallback(TransactionCallback)` method of this interface or a default TransactionCallback object if `setTransactionCallback` was not previously called for this ObjectGrid.

See Also:

[setTransactionCallback\(TransactionCallback\)](#), [TransactionCallback](#)

setCollisionArbiter

`void setCollisionArbiter(com.ibm.websphere.objectgrid.revision.CollisionArbiter arbiter)`

Sets the CollisionArbiter that is responsible for arbitration of revision conflicts.

A CollisionArbiter that implements the ObjectGridLifecycleListener interface is automatically added as if the [addEventListener\(EventListener\)](#) method was called. Any previous arbiter which implements ObjectGridLifecycleListener interface is removed as if the [removeEventListener\(EventListener\)](#) method was called.

A CollisionArbiter may implement the ObjectGridPlugin interface in order to receive enhanced ObjectGrid plug-in lifecycle method calls. The plug-in is also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

`arbiter` - The arbitration logic that will be used to resolve collisions.

Since:

7.1

getCollisionArbiter

`com.ibm.websphere.objectgrid.revision.CollisionArbiter getCollisionArbiter()`

Retrieves the CollisionArbiter that this grid is using to resolve revision collisions.

Returns:

The arbitration logic that is responsible for resolving revision collisions.

Since:

defineMap

[BackingMap](#) `defineMap(String name)`

Defines a BackingMap that will be used by the application.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

name - the name of the map being defined.

Returns:

a BackingMap reference

Throws:

[IllegalArgumentException](#) - if name is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#), [BackingMap](#)

createMap

[BackingMap](#) `createMap(String name)`

Creates a BackingMap, but does not associate it with this ObjectGrid.

This method is to be used in tandem with the `setMaps(List)` method, which will associate BackingMaps with this ObjectGrid. These methods are for use when configuring an ObjectGrid with the Spring Framework.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

name - the name of the map being defined.

Returns:

a BackingMap reference

Throws:

[IllegalArgumentException](#) - if name is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

Since:

WAS XD 6.0.1

See Also:

[initialize\(\)](#), [setMaps\(List\)](#)

setMaps

void `setMaps(List mapList)`

Clears any BackingMaps that have been previously defined on this ObjectGrid and replaces them with the List of BackingMaps provided.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

mapList - a list of BackingMaps to set on this ObjectGrid.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.0.1

See Also:

[createMap\(String\)](#), [initialize\(\)](#)

getListOfMapNames

[List](#) `getListOfMapNames()`

Gets the list of map names currently defined for this ObjectGrid instance.

Note, once the initialize() method is called, the List returned will not change. However, it could change if called prior to initialization. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Returns:

a List of String objects, one String per map that was previously configured by the defineMap(String) or setMaps(List) method. An empty List is returned if no maps are currently defined.

See Also:

[defineMap\(String\)](#), [initialize\(\)](#), [setMaps\(List\)](#)

getMap

[BackingMap](#) `getMap(String name)`

Returns a BackingMap previously configured by calling the defineMap(String) or setMaps(List) method.

Parameters:

name - the same name that was used as an argument to the defineMap(String) or createMap(String) method. A null reference is returned if a map is not associated with this ObjectGrid for the specified map name.

Returns:

the BackingMap instance

See Also:

[createMap\(String\)](#), [defineMap\(String\)](#), [setMaps\(List\)](#)

initialize

void `initialize()`

throws [ObjectGridException](#)

Begins the bootstrapping of the ObjectGrid and Session instances.

After this method has been invoked, the configuration of the ObjectGrid is considered complete and is ready for runtime usage. Any additional configuration method invocations, such as defineMap(String), will result in an exception. This method is considered optional since the first call to one of the getSession methods will perform an implicit initialization.

Throws:

[ObjectGridException](#) - if an error occurs during processing

addEventListener

void `addEventListener(com.ibm.websphere.objectgrid.plugins.EventListener listener)`

Adds an EventListener.

Significant events will be communicated to interested listeners through the ObjectGridEventListener and ObjectGridLifecycleListener callback interface. Multiple event listeners are allowed to be registered, with no implied ordering of event notifications.

Note, this method is allowed to be invoked before and after the initialize() method.

Object grid plug-ins (TransactionCallback, CollisionArbiter) that implement the ObjectGridLifecycleListener are automatically added as lifecycle listeners when added to the ObjectGrid.

Parameters:

listener - An instance of ObjectGridEventListener or ObjectGridLifecycleListener

Throws:

[IllegalArgumentException](#) - if listener is null or not an instance of ObjectGridEventListener, ObjectGridLifecycleListener.

[IllegalStateException](#) - if this method is called during initialization by one of the configured plugins and the ObjectGrid runtime is not in a usable state to initialize the ObjectGridEventListener.

See Also:

[ObjectGridEventListener](#), ObjectGridLifecycleListener, EventListener

addEventListener

void **addEventListener**([ObjectGridEventListener](#) listener)

Deprecated. *This method is deprecated in version 7.1.1, use the [addEventListener\(EventListener\)](#) method.*

Provided for compatibility with old releases, use the [addEventListener\(EventListener\)](#) method.

Parameters:

listener -

removeEventListener

void **removeEventListener**(com.ibm.websphere.objectgrid.plugins.EventListener listener)

Removes an EventListener.

This method removes an ObjectGridEventListener or ObjectGridLifecycleListener that was previously added to this object using the addEventListener(ObjectGridEventListener) or setEventListeners(List) method. If the desired ObjectGridEventListener is not found, no error will be returned.

Note, this method is allowed to be invoked before and after the initialize() method. Object grid plug-ins (TransactionCallback, CollisionArbiter) that implement the ObjectGridLifecycleListener are automatically removed as lifecycle listeners when removed from the ObjectGrid.

Parameters:

listener - An instance of ObjectGridEventListener or ObjectGridLifecycleListener

Throws:

[IllegalArgumentException](#) - if listener is null or not an instance of ObjectGridEventListener, ObjectGridLifecycleListener

See Also:

[addEventListener\(EventListener\)](#), [ObjectGridEventListener](#), EventListener

removeEventListener

void **removeEventListener**([ObjectGridEventListener](#) listener)

Deprecated. *This method is deprecated in version 7.1.1, use the [removeEventListener\(EventListener\)](#) method.*

Provided for compatibility with old releases, use the [removeEventListener\(EventListener\)](#) method.

Parameters:

listener -

setEventListeners

[@Deprecated](#)

void **setEventListeners**([List](#) listeners)

Deprecated. *This method is deprecated in version 7.1.1. Use the [addEventListener\(EventListener\)](#) or [removeEventListener\(EventListener\)](#) methods. Plug-ins that implement the `ObjectGridLifecycleListener` interface are automatically registered with the grid. Using this method will remove those automatically added listeners.*

This overwrites the current list of EventListeners and replaces it with the supplied List of EventListeners

Note, this method is allowed to be invoked before and after the `initialize()` method.

Parameters:

listeners - List of `ObjectGridEventListeners` and `ObjectGridLifecycleListener` instances

Throws:

[ClassCastException](#) - if one of the elements in the provided list is not an instance of `ObjectGridEventListener`

[IllegalArgumentException](#) - if listeners is null, contains a null reference, or contains an instance of a type other than `ObjectGridEventListener` and `ObjectGridLifecycleListener`

[IllegalStateException](#) - if this method is called during initialization by one of the configured plugins and the `ObjectGrid` runtime is not in a usable state to initialize the `ObjectGridEventListener` objects.

See Also:

`EventListener`, [addEventListener\(EventListener\)](#), [removeEventListener\(EventListener\)](#)

getEventListeners

[List](#) `getEventListeners()`

Returns the current list of EventListeners.

Returns:

The current list of EventListeners.

See Also:

[addEventListener\(EventListener\)](#), `EventListener`, [ObjectGridEventListener](#), `ObjectGridLifecycleListener`

getName

[String](#) `getName()`

Gets the name of this `ObjectGrid`.

This method is useful for authorization as all Maps are prefixed with the `ObjectGrid` name.

Returns:

The name of the `ObjectGrid`.

See Also:

[setName\(String\)](#)

setName

void `setName(String gridName)`

Sets the name of this `ObjectGrid`. Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

gridName - The `ObjectGrid` name to use.

Throws:

[IllegalArgumentException](#) - if gridName is null

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

reserveSlot

int `reserveSlot(String containerName)`

Allows plugins on this `ObjectGrid` to reserve slots for use to store context data.

Currently the TxID object is the only object that uses slots for storing context data. TxID slots are used for storing transactional context data.

Once a slot is reserved, the slot assignment is permanent and cannot be given back. Note, this method is allowed to be invoked before and after the initialize() method.

Parameters:

containerName - The name of the Object with the slots. Currently TxID.SLOT_NAME is the only supported value for this argument.

Returns:

The slot index to use.

Throws:

[IllegalArgumentException](#) - if containerName is not TxID.SLOT_NAME.

See Also:

[TxID.SLOT_NAME](#), [TxID.getSlot\(int\)](#), [TxID.putSlot\(int, Object\)](#)

setSubjectValidation

void **setSubjectValidation**(com.ibm.websphere.objectgrid.security.plugins.SubjectValidation subjectValidation)

Sets the SubjectValidation for this ObjectGrid instance.

Passing null to this method removes a previously set SubjectValidation object from an earlier invocation of this method and indicates that this ObjectGrid is not associated with a SubjectValidation object.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provided SubjectValidation object will not be used.

A SubjectValidation plugin can be used to validate the Subject object passed in is a valid Subject. Please refer to SubjectValidation for more details.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:

subjectValidation - the SubjectValidation plugin

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

See Also:

[getSession\(Subject\)](#), [initialize\(\)](#), SubjectValidation

setAuthorizationMechanism

void **setAuthorizationMechanism**(int authMechanism)

Sets the authorization mechanism.

If this method is not invoked, the default authorization mechanism is SecurityConstants.AUTHORIZATION_MECHANISM_JAAS.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provide authorization mechanism will not be used.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:

authMechanism - the authorization mechanism, must be one of the final static variable on the SecurityConstants class.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

See Also:

[initialize\(\)](#), [SecurityConstants.AUTHORIZATION_MECHANISM_CUSTOM](#), [SecurityConstants.AUTHORIZATION_MECHANISM_JAAS](#)

setSecurityEnabled

void **setSecurityEnabled()**

Enables the ObjectGrid security.

Security on the ObjectGrid level refers to ObjectGrid authorizations.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Throws:

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#)

isSecurityEnabled

boolean **isSecurityEnabled()**

Checks whether security is enabled on this ObjectGrid or not.

Security on the ObjectGrid level refers to ObjectGrid authorizations. Security is disabled by default.

Returns:

true if security is enabled on this ObjectGrid; false otherwise.

See Also:

[setSecurityEnabled\(\)](#)

setPermissionCheckPeriod

void **setPermissionCheckPeriod**(int period)

Sets the permission check period.

This method takes a single parameter indicating how often the customer wants to check the permission used to allow a client access. If the parameter is 0 then every single authorized operation call will ask the authorization mechanism, either JAAS authorization or custom authorization to check if the current Subject has permission. This approach may be prohibitively expensive from a performance point of view depending on the authorization implementation, but if it is required then you can do it. Alternatively, if the parameter is > 0 then it indicates the number of seconds to cache a set of permissions before returning to the authorization mechanism to refresh them. This mechanism provides much better performance, but you run the risk that if the back-end permissions are changed during this time, the ObjectGrid will possibly allow or prevent access even though the back-end security provider has been modified.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `initialize()` method. Also, keep in mind that the `getSession` methods implicitly call the `initialize()` method if it has yet to be called by the application.

Parameters:

period - the permission check period in seconds.

Throws:

[IllegalStateException](#) - if this method is called after the `initialize()` method is called.

See Also:

[initialize\(\)](#)

setSubjectSource

void **setSubjectSource**(com.ibm.websphere.objectgrid.security.plugins.SubjectSource source)

Sets the SubjectSource plugin.

Passing null to this method removes a previously set SubjectSource object from an earlier invocation of this method and indicates that this ObjectGrid is not associated with a SubjectSource object.

A SubjectSource plugin can be used to get a Subject object from the environment to represent the ObjectGrid client.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provided SubjectSource object will not be used.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:

source - the SubjectSource plugin

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

See Also:

[initialize\(\)](#), SubjectSource

setTxTimeout

void **setTxTimeout**(int timeout)

Sets the transaction timeout value to a specified number of seconds.

Any transaction that is started by use of a Session returned by one of the getSession methods on this interface must complete within the number of seconds specified by the transaction timeout parameter of this method. The timeout value is the maximum number of seconds the transaction is allowed to execute. If a transaction executes longer than this amount, a TransactionTimeoutException is thrown and the transaction is rolled back even if commit is requested.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

The transaction timeout is used by any transaction started by a Session that is returned by the getSession methods of this interface. Since this method must be called prior to getSession method to avoid IllegalStateException, this method only affects transactions that are started after this method is called. If this method is never called, the transaction is allowed unlimited amount of time to complete.

Parameters:

timeout - is the transaction timeout value in seconds. Use a value of 0 to indicate a transaction is allowed unlimited amount of time so that no TransactionTimeoutException ever occurs.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.0.1

See Also:

[initialize\(\)](#), [Session.TRANSACTION_NO_TIMEOUT](#), [Session.setTransactionTimeout\(int\)](#), [TransactionTimeoutException](#)

getTxTimeout

int **getTxTimeout**()

Gets transaction timeout setting for this ObjectGrid instance.

Returns:

timeout value that was passed to the setTxTimeout(int) method or 0 if setTxTimeout was never called.

Since:

WAS XD 6.0.1

See Also:

[setTxTimeout\(int\)](#)

setTxIsolation

void **setTxIsolation**(int level)

Sets the default transaction isolation level for all sessions created by the ObjectGrid. The constants defined in the Session interface are the possible transaction isolation levels. The default is [Session.TRANSACTION_REPEATABLE_READ](#).

Parameters:

level - one of the following Session constants: [Session.TRANSACTION_READ_UNCOMMITTED](#), [Session.TRANSACTION_READ_COMMITTED](#) or [Session.TRANSACTION_REPEATABLE_READ](#)

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

[IllegalArgumentException](#) - if this method includes an invalid transaction isolation level.

Since:

7.1.1

getTxIsolation

int **getTxIsolation**()

Retrieves the default transaction isolation level.

Returns:

the current transaction isolation level.

Since:

7.1.1

See Also:

[setTxIsolation\(int\)](#)

destroy

void **destroy**()

Destroys this instance.

This method should be invoked when the ObjectGrid is no longer being used. When this method is called, the ObjectGrid can free up any resources it is using. No new Sessions can be created or used after the destroy() has been invoked. Any in-flight Sessions will be allowed to continue, if the resources are still available to complete processing.

getSession

[Session](#) **getSession**([CredentialGenerator](#) credGen)
throws [ObjectGridException](#),
[TransactionCallbackException](#)

Get a session using a CredentialGenerator.

This method can only be called by the ObjectGrid client in an ObjectGrid client server environment. If ObjectGrid is used in a local model, that is, within the same JVM with no client or server existing, getSession(Subject) or the SubjectSource plugin should be used to secure the ObjectGrid.

If the initialize() method has not been invoked prior to the first getSession invocation, an implicit initialization will occur. This ensures that all of the configuration is complete before any runtime usage is required.

Parameters:

credGen - A CredentialGenerator for generating a credential for the session returned.

Returns:

An instance of Session

Throws:

[ObjectGridException](#) - if an error occurs during processing

[TransactionCallbackException](#) - if the TransactionCallback throws an exception
[IllegalStateException](#) - if this method is called after the destroy() method is called.

Since:

WAS XD 6.0.1

See Also:

[destroy\(\)](#), [initialize\(\)](#), [CredentialGenerator](#), [Session](#)

setQueryConfig

void **setQueryConfig**([QueryConfig](#) queryConfig)

Set the QueryConfig object for this ObjectGrid. A QueryConfig object provides query configurations for executing object queries over the maps in this ObjectGrid.

Parameters:

queryConfig - The QueryConfig to associate with this ObjectGrid instance.

Throws:

[IllegalArgumentException](#) - if queryConfig is null.

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.1

See Also:

[QueryConfig](#)

registerEntities

void **registerEntities**([URL](#) entityXML)

Registers one or more entities from an entity XML file.

Entity registration is required prior to ObjectGrid initialization to bind an Entity with a BackingMap and any defined indices.

This method may be called multiple times.

Parameters:

entityXML - the URL of the entity XML that defines the entities.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.1

registerEntities

void **registerEntities**([Class](#)[] entities)

Register one or more entities based on the class metadata.

Entity registration is required prior to ObjectGrid initialization to bind an Entity with a BackingMap and any defined indices.

This method may be called multiple times.

Parameters:

entities - one or more annotated entity classes to register as entities.

Throws:

[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:

WAS XD 6.1

getObjectGridType

int **getObjectGridType**()

Returns the type of ObjectGrid.

The return value is equivalent to one of the constants declared on this interface, [LOCAL](#), [SERVER](#), or [CLIENT](#).

Returns:
the ObjectGrid type

Since:
WAS XD 6.1

setObjectGridAuthorization

void **setObjectGridAuthorization**(com.ibm.websphere.objectgrid.security.plugins.ObjectGridAuthorization ogAuthorization)

Sets the ObjectGridAuthorization for this ObjectGrid instance.

Passing null to this method removes a previously set ObjectGridAuthorization object from an earlier invocation of this method and indicates that this ObjectGrid is not associated with a ObjectGridAuthorization object.

This method should only be used when ObjectGrid security is enabled. If the ObjectGrid security is disabled, the provided ObjectGridAuthorization object will not be used.

A ObjectGridAuthorization plugin can be used to authorize access to the ObjectGrid and maps. Please refer to ObjectGridAuthorization for more details.

Note, to avoid an IllegalStateException, this method must be called prior to the initialize() method. Also, keep in mind that the getSession methods implicitly call the initialize() method if it has yet to be called by the application.

Parameters:
ogAuthorization - the ObjectGridAuthorization plugin

Throws:
[IllegalStateException](#) - if this method is called after the initialize() method is called.

Since:
WAS XD 6.1

See Also:
[initialize\(\)](#), ObjectGridAuthorization

setAccessByCreatorOnlyMode

void **setAccessByCreatorOnlyMode**(int accessByCreatorOnlyMode)

Set the "access by creator only" mode.

Enabling "access by creator only" mode ensures that only the user (represented by the Principals associated with it), who inserts the record into the map, can access (read, update, invalidate, and remove) the record.

The "access by creator only" mode can be disabled, or can complement the ObjectGrid authorization model, or it can supersede the ObjectGrid authorization model. The default value is disabled: [SecurityConstants.ACCESS_BY_CREATOR_ONLY_DISABLED](#).

Parameters:
accessByCreatorOnlyMode - the access by creator mode.

Since:
WAS XD 6.1 FIX3

See Also:
[SecurityConstants.ACCESS_BY_CREATOR_ONLY_DISABLED](#),
[SecurityConstants.ACCESS_BY_CREATOR_ONLY_COMPLEMENT](#),
[SecurityConstants.ACCESS_BY_CREATOR_ONLY_SUPERSEDE](#)

getState

com.ibm.websphere.objectgrid.plugins.ObjectGridLifecycleListener.State **getState**()

Retrieve the current life cycle state of this ObjectGrid.

Returns:

the current state.
Since:
7.1.1

[Overview](#) [Package](#) [Class](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

*IBM WebSphere® DataPower® XC10
Appliance*

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

Release 2.5 Client API Specification

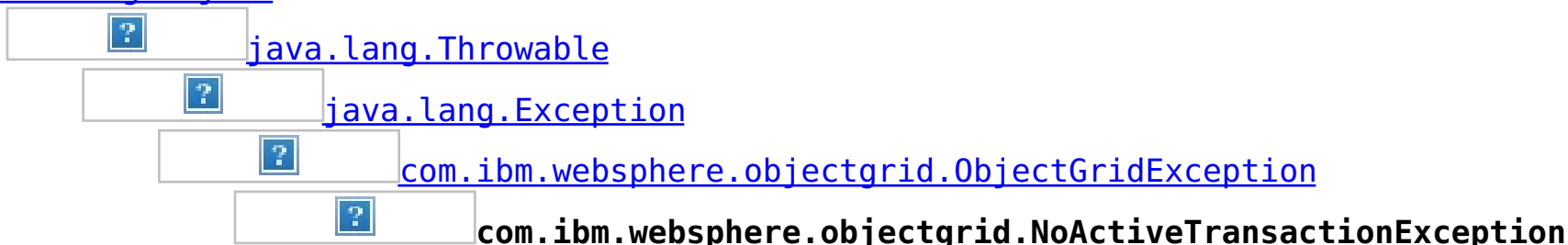
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class NoActiveTransactionException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class NoActiveTransactionException
extends ObjectGridException
```

An exception indicating there is no active transaction.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[NoActiveTransactionException](#)()

Constructs a new NoActiveTransactionException with null as its detail message.

[NoActiveTransactionException](#)(String message)

Constructs a new NoActiveTransactionException with the specified detail message.

[NoActiveTransactionException](#)(String message, Throwable cause)

Constructs a new NoActiveTransactionException with the specified detail message and cause.

[NoActiveTransactionException](#)(Throwable cause)

Constructs a new NoActiveTransactionException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

NoActiveTransactionException

```
public NoActiveTransactionException()
```

Constructs a new NoActiveTransactionException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

NoActiveTransactionException

```
public NoActiveTransactionException(String message)
```

Constructs a new NoActiveTransactionException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

NoActiveTransactionException

```
public NoActiveTransactionException(String message,  
                                     Throwable cause)
```

Constructs a new NoActiveTransactionException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this NoActiveTransactionException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

NoActiveTransactionException

```
public NoActiveTransactionException(Throwable cause)
```

Constructs a new NoActiveTransactionException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for NoActiveTransactionExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

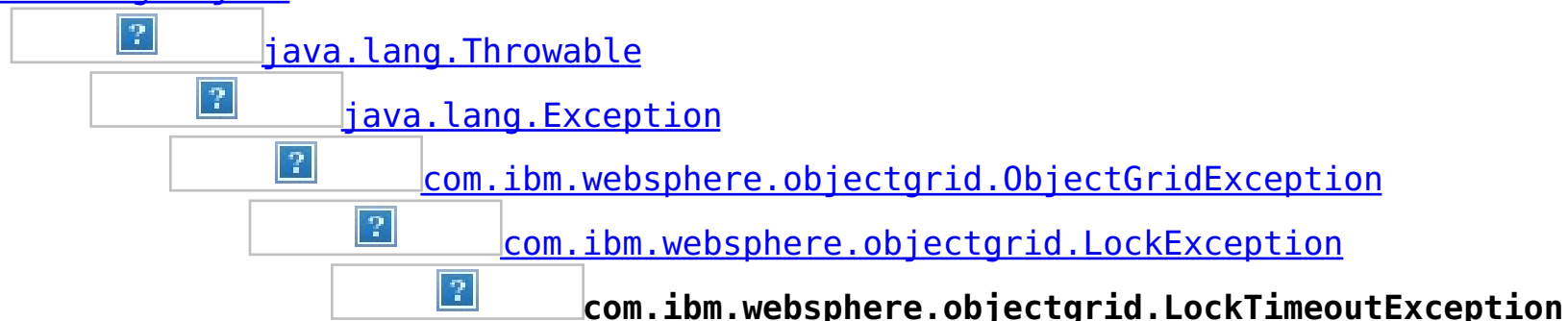
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class LockTimeoutException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

Direct Known Subclasses:

[LockDeadlockException](#), [LockInternalFailureException](#)

```
public class LockTimeoutException
extends LockException
```

This exception is used by the lock manager to indicate that the maximum wait time for a lock has been exceeded. The timeout may or may not be the result of a deadlock. If it is a deadlock, the timeout is used to break the deadlock.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

LockTimeoutException ()	Constructs a new LockTimeoutException with null as its detail message.
LockTimeoutException (String message)	Constructs a new LockTimeoutException with the specified detail message.
LockTimeoutException (String message, Throwable cause)	Constructs a new LockTimeoutException with the specified detail message and cause.
LockTimeoutException (Throwable cause)	Constructs a new LockTimeoutException with a specified cause.

Method Summary

v o i d	forceJavaCore ()
S t	getLockRequestQueueDetails ()

r i n g	Provides detailed information about the state of the lock queue when the lock timeout occurred.
S t r i n g	getMessage() Returns the detail message string of this exception.
v o i d	setLockRequestQueueDetails(String string) Sets the details of the lock requests on the lock request queue at the time the lock timeout occurred.

Methods inherited from class [com.ibm.websphere.objectgrid.ObjectGridException](#)
[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)
[fillInStackTrace](#), [getLocalizedMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)
[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockTimeoutException

```
public LockTimeoutException()
```

Constructs a new LockTimeoutException with null as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockTimeoutException

```
public LockTimeoutException(String message)
```

Constructs a new LockTimeoutException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [getMessage\(\)](#)

LockTimeoutException

```
public LockTimeoutException(String message,
                             Throwable cause)
```

Constructs a new `LockTimeoutException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `LockTimeoutException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [getMessage\(\)](#)

LockTimeoutException

```
public LockTimeoutException(Throwable cause)
```

Constructs a new `LockTimeoutException` with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for `LockTimeoutExceptions` that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Method Detail

getLockRequestQueueDetails

```
public String getLockRequestQueueDetails()
```

Provides detailed information about the state of the lock queue when the lock timeout occurred.

Returns:

the argument that was passed to the `setLockRequestQueueDetails(String)` method of this class or `null` if the `setLockRequestQueueDetails` method was not previously called for this object.

forceJavaCore

```
public void forceJavaCore()
```

setLockRequestQueueDetails

```
public void setLockRequestQueueDetails(String string)
```

Sets the details of the lock requests on the lock request queue at the time the lock timeout occurred.

Parameters:

`string` - the details of lock requests on the lock request queue at the time the lock timeout occurred.

getMessage

public [String](#) getMessage()

Returns the detail message string of this exception. The returned String includes the request queue details as well as the message provided to the constructor.

Overrides:

[getMessage](#) in class [Throwable](#)

Returns:

the detail message string of this object instance

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All				
			Classes					
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD					

[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid
Class LockStrategy

[java.lang.Object](#)



public final class **LockStrategy**
extends [Object](#)

LockStrategy provides an enumerated type idiom for use on the `BackingMap.setLockStrategy(LockStrategy)` method. It determines whether or not a lock manager is needed for a BackingMap and if so, whether to use an optimistic or pessimistic locking strategy.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.setLockStrategy\(LockStrategy\)](#)

Field Summary

s t a t i c L O C K S T R A T E G Y	<p>NONE</p> <p>NONE indicates internal LockManager use is not needed since concurrency control is provided outside of ObjectGrid either by a persistence manager using objectgrid as a side cache, the application, or by a Loader plugin (for example, uses database locks to control concurrency).</p>
s t a t i c L O C K S T R A T E G Y	<p>OPTIMISTIC</p> <p>OPTIMISTIC is typically used for a map that does not have a Loader plugin, the map is read mostly, and locking is neither provided by a persistence manager using ObjectGrid as a side cache or by the application itself.</p>
s	

t
a
t
i
c
L
o
c
k
S
t
r
a
t
e
g
y

PESSIMISTIC

PESSIMISTIC is typically used for a map that does not have a Loader plugin and locking is neither provided by a persistence manager using ObjectGrid as a side cache, by a Loader plugin, or by the application itself.

Method Summary

S
t
r
i
n
g

toString()

Returns a string representation of the LockStrategy.

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

NONE

```
public static final LockStrategy NONE
```

NONE indicates internal LockManager use is not needed since concurrency control is provided outside of ObjectGrid either by a persistence manager using objectgrid as a side cache, the application, or by a Loader plugin (for example, uses database locks to control concurrency).

OPTIMISTIC

```
public static final LockStrategy OPTIMISTIC
```

OPTIMISTIC is typically used for a map that does not have a Loader plugin, the map is read mostly, and locking is neither provided by a persistence manager using ObjectGrid as a side cache or by the application itself. For this strategy, an exclusive lock is obtained on a map entry being inserted, updated, or removed at commit time. The lock ensures version information cannot be changed by another transaction while the transaction being committed is performing an optimistic version check.

PESSIMISTIC

```
public static final LockStrategy PESSIMISTIC
```

PESSIMISTIC is typically used for a map that does not have a Loader plugin and locking is neither provided by a persistence manager using ObjectGrid as a side cache, by a Loader plugin, or by the application itself. It is typically used when optimistic approach fails too often since there are update transactions that frequently collide on the same

map entry (e.g. not a read mostly map or large number of clients accessing a small map).

Method Detail

toString

```
public String toString()
```

Returns a string representation of the LockStrategy.

Overrides:

[toString](#) in class [Object](#)

Returns:

a string representation of the LockStrategy.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class LockInternalFailureException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class LockInternalFailureException
extends LockTimeoutException
```

This exception is used by the lock manager to indicate it detected some internal programming error while processing a lock or unlock request.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[LockInternalFailureException](#)()

Constructs a new LockInternalFailureException with null as its detail message.

[LockInternalFailureException](#)(String message)

Constructs a new LockInternalFailureException with the specified detail message.

[LockInternalFailureException](#)(String message, Throwable cause)

Constructs a new LockInternalFailureException with the specified detail message and cause.

[LockInternalFailureException](#)(Throwable cause)

Constructs a new LockInternalFailureException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[LockTimeoutException](#)

[forceJavaCore](#), [getLockRequestQueueDetails](#), [getMessage](#), [setLockRequestQueueDetails](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class [java.lang.Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockInternalFailureException

```
public LockInternalFailureException()
```

Constructs a new `LockInternalFailureException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockInternalFailureException

```
public LockInternalFailureException(String message)
```

Constructs a new `LockInternalFailureException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [LockTimeoutException.getMessage\(\)](#)

LockInternalFailureException

```
public LockInternalFailureException(String message,  
                                   Throwable cause)
```

Constructs a new `LockInternalFailureException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `LockInternalFailureException`'s detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [LockTimeoutException.getMessage\(\)](#)

LockInternalFailureException

public **LockInternalFailureException**([Throwable](#) cause)

Constructs a new **LockInternalFailureException** with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for **LockInternalFailureExceptions** that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A null value is permitted and indicates that the cause is nonexistent or is unknown.

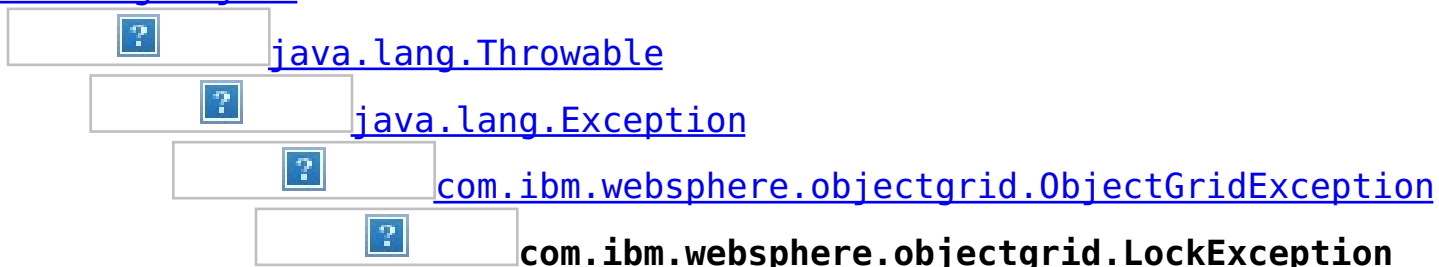
See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

com.ibm.websphere.objectgrid
Class LockException

[java.lang.Object](#)



All Implemented Interfaces:
[ObjectGridException](#), [Serializable](#)

Direct Known Subclasses:
[LockTimeoutException](#)

```
public class LockException
extends ObjectGridException
```

A general locking exception indicating something went wrong with locking operations.

Since:
WAS XD 6.0, XC10

See Also:
[LockTimeoutException](#), [Serialized Form](#)

Constructor Summary

- [LockException](#)()
Constructs a new LockException with null as its detail message.
- [LockException](#)(String message)
Constructs a new LockException with the specified detail message.
- [LockException](#)(String message, Throwable cause)
Constructs a new LockException with the specified detail message and cause.
- [LockException](#)(Throwable cause)
Constructs a new LockException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)
[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)
[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockException

```
public LockException()
```

Constructs a new LockException with null as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockException

```
public LockException(String message)
```

Constructs a new LockException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

LockException

```
public LockException(String message,  
                    Throwable cause)
```

Constructs a new LockException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this LockException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

LockException

```
public LockException(Throwable cause)
```

Constructs a new LockException with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of cause). This constructor is useful for LockExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for

later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All			
		Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							

[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class LockDeadlockException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class LockDeadlockException
extends LockTimeoutException
```

This exception is used by the lock manager to indicate that it detected a deadlock. It prevents the deadlock by throwing this exception. Typically, this deadlock is a result of the following scenario: one transaction owns a weaker lock as a result of getting a map entry, and then, at commit time, the transaction attempts to promote the weaker lock to a stronger lock in order to apply the changes to the data store. For example, two transactions try to promote from shared locks to exclusive locks but each transaction already owns a shared lock.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[LockDeadlockException](#)()

Constructs a new LockDeadlockException with null as its detail message.

[LockDeadlockException](#)(String message)

Constructs a new LockDeadlockException with the specified detail message.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[LockTimeoutException](#)

[forceJavaCore](#), [getLockRequestQueueDetails](#), [getMessage](#), [setLockRequestQueueDetails](#)

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

LockDeadlockException

```
public LockDeadlockException()
```

Constructs a new LockDeadlockException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

LockDeadlockException

```
public LockDeadlockException(String message)
```

Constructs a new LockDeadlockException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [LockTimeoutException.getMessage\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

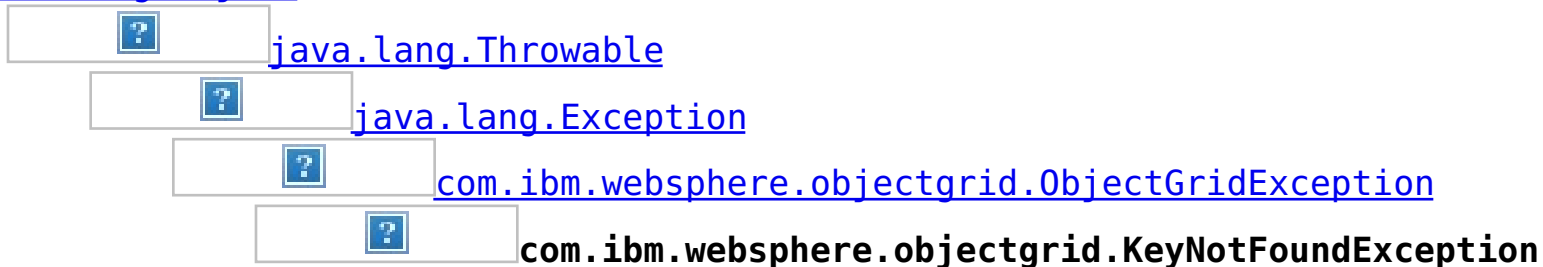
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class KeyNotFoundException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class KeyNotFoundException
extends ObjectGridException
```

Normally, record not found means a null is returned. However, sometimes on the explicit operation methods like update methods, we can figure that the record isn't there and then we throw this exception.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[KeyNotFoundException](#)()

Constructs a new KeyNotFoundException with null as its detail message.

[KeyNotFoundException](#)(String message)

Constructs a new KeyNotFoundException with the specified detail message.

[KeyNotFoundException](#)(String message, Throwable cause)

Constructs a new KeyNotFoundException with the specified detail message and cause.

[KeyNotFoundException](#)(Throwable cause)

Constructs a new KeyNotFoundException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

KeyNotFoundException

```
public KeyNotFoundException()
```

Constructs a new KeyNotFoundException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

KeyNotFoundException

```
public KeyNotFoundException(String message)
```

Constructs a new KeyNotFoundException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

KeyNotFoundException

```
public KeyNotFoundException(String message,  
                             Throwable cause)
```

Constructs a new KeyNotFoundException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this KeyNotFoundException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

KeyNotFoundException

```
public KeyNotFoundException(Throwable cause)
```

Constructs a new KeyNotFoundException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for KeyNotFoundExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that

the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Interface JavaMap

All Superinterfaces:

[Map](#)

```
public interface JavaMap  
extends Map
```

This interface is a handle to a named Map. Maps should have homogeneous keys and values. An instance of this JavaMap can only be used by the thread that is currently associated with the Session that was used to get this JavaMap instance. Both Session and JavaMap objects are not allowed to be shared by multiple threads concurrently.

Users can get an instance of JavaMap from an instance of ObjectMap by calling `ObjectMap.getJavaMap()`. There are two main differences between JavaMap and ObjectMap:

- JavaMap extends `java.util.Map`. Therefore, users can cast an instance of JavaMap to `java.util.Map` if they want.
- The methods in JavaMap are defined to throw Exceptions similar to those defined on the `java.util.Map` interface that is `ObjectGridRuntimeException`, which is a subclass of `java.lang.RuntimeException` is used for error conditions. The methods in ObjectMap are defined to throw `ObjectGridExceptions`, which are checked exceptions.

The only methods that are supported from the `java.util.Map` interface are:

- `containsKey(Object)`
- `get(Object)`
- `put(Object, Object)`
- `putAll(Map)`
- `remove(Object)`
- `clear()`

All other methods on the `java.util.Map` interface will throw `java.lang.UnsupportedOperationException`.

Since:

WAS XD 6.0, XC10

See Also:

[ObjectMap](#), [ObjectMap.getJavaMap\(\)](#), [Map](#), [BackingMap.setCopyMode\(CopyMode, Class\)](#), [BackingMap.setLockStrategy\(LockStrategy\)](#)

Nested Class Summary

Nested classes/interfaces inherited from interface `java.util.Map`

[Map.Entry<K, V>](#)

Method Summary

v o i d	<p>clear() Clear all keys from the Map.</p>
v o i d	<p>clearCopyMode() Resets the CopyMode back to the one in the BackingMap.</p>
b o o l e a n	<p>containsKey(Object key) Returns true if this map contains a mapping for the specified key.</p>
b o o l e a n	<p>containsValue(Object value) This method of the java.util.Map interface is not supported.</p>
S e t	<p>entrySet() This method of the java.util.Map interface is not supported.</p>
v o i d	<p>flush() Pushes the current set of changes for the JavaMap instance to the Loader without committing the changes.</p>
O b j e c t	<p>get(Object key) Retrieves the object from the cache at the given key.</p>
L i s t	<p>getAll(List keyList) Gets a list of entries from the map.</p>
L i s t	<p>getAllForUpdate(List keyList) Same as the getAll(List) method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries.</p>
c o m . i b m . w e b s p h e r e . p r o	

j
e
c
t
o
r
.
m
d
.
E
n
t
i
t
y
M
e
t
a
d
a
t
a

[getEntityMetadata\(\)](#)

Retrieve the metadata for the entity associated with this map.

O
b
j
e
c
t

[getForUpdate\(Object key\)](#)

Same as `get(Object)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry.

O
b
j
e
c
t

[getIndex\(String name\)](#)

Returns a reference to the named index that can be used with this Map.

S
t
r
i
n
g

[getName\(\)](#)

Returns the name of the JavaMap as defined by the configuration.

O
b
j
e
c
t

[getNextKey\(long timeout\)](#)

Retrieves a key off the map in first-in-first-out (FIFO) insert order.

V
o
i
d

[insert\(Object key, Object value\)](#)

Performs an explicit insert of a given entry.

V
o
i
d

[invalidate\(Object key, boolean isGlobal\)](#)

Invalidates an entry in the cache based on the key parameter.

V
o
i
d

[invalidateAll\(Collection keyList, boolean isGlobal\)](#)

Invalidate a set of cache entries based on the Collection of keys provided.

b
o
o
l
e

[isEmpty\(\)](#)

This method of the `java.util.Map` interface is not supported.

a n	
S e t	keySet() This method of the java.util.Map interface is not supported.
O b j e c t	put(Object key, Object value) Puts the Object value into the cache at location represented by key.
v o i d	putAll(Map map) Puts each of the Object values into the cache at location represented by the corresponding key contained in the Map.
O b j e c t	remove(Object key) Removes the Object value from the cache represented by key.
v o i d	removeAll(Collection keyList) Batch remove from the Map.
v o i d	setCopyMode(CopyMode copyMode, Class valueInterface) Allows the CopyMode for the Map to be overridden on this map on this session only.
i n t	setTimeToLive(int ttl) Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL.
i n t	size() This method of the java.util.Map interface is not supported.
v o i d	touch(Object key) Updates the last access time in the BackingMap without retrieving the value to the JavaMap.
v o i d	update(Object key, Object value) Performs an explicit update of a given entry.
C o l l e c t i o n	values() This method of the java.util.Map interface is not supported.

Methods inherited from interface java.util.[Map](#)

[equals](#), [hashCode](#)

Method Detail

getName

[String](#) getName()

Returns the name of the JavaMap as defined by the configuration.

Returns:

name of JavaMap

getForUpdate

[Object](#) getForUpdate([Object](#) key)
throws [ObjectGridRuntimeException](#)

Same as get(Object) method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for this map entry. See [LockStrategy.PESSIMISTIC](#) for additional information. Whether or not a copy of the object is returned is determined by the [CopyMode](#) setting for this map. See [CopyMode](#) for a description of each possible [CopyMode](#). If the key cannot be found in the map, a null value will be returned. A null value is also returned if the value is null and this map allows null values. To distinguish the two, use the [containsKey](#) method.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an an [EntityManager](#) API entity is associated with the [BackingMap](#).

See [ObjectMap.getForUpdate\(Object\)](#) for additional specification details.

Parameters:

key - The entry to fetch

Returns:

the value retrieved for update or null

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[containsKey\(Object\)](#), [get\(Object\)](#), [CopyMode](#), [LockStrategy.PESSIMISTIC](#), [ObjectMap.getForUpdate\(Object\)](#)

getAll

[List](#) getAll([List](#) keyList)
throws [ObjectGridRuntimeException](#)

Gets a list of entries from the map.

If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

A return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

See [ObjectMap.getAll\(List\)](#) for additional specification details.

Parameters:

keyList - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[get\(Object\)](#), [ObjectMap.getAll\(List\)](#)

getAllForUpdate

[List](#) `getAllForUpdate(List keyList)`
throws [ObjectGridRuntimeException](#)

Same as the `getAll(List)` method except that if pessimistic lock strategy is used for this map, an upgradable lock mode is obtained for these map entries. See `LockStrategy.PESSIMISTIC` for additional information. If a key in the list cannot be found, a null value will be set at the appropriate position in the returned list.

The return value is a `SerializedValue` when using the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` or `OutputFormat.RAW` `OutputFormat` with a `ValueSerializerPlugin` plug-in defined on the [BackingMap](#). The `SerializedValue` allows access to the value in its serialized form, or its native Java Object form.

A return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

See [ObjectMap.getAllForUpdate\(List\)](#) for additional specification details.

Parameters:

keyList - A list of keys for identifying which entries to fetch

Returns:

a list of values

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[getAll\(List\)](#), [getForUpdate\(Object\)](#), [LockStrategy.PESSIMISTIC](#),
[ObjectMap.getAllForUpdate\(List\)](#)

removeAll

`void removeAll(Collection keyList)`
throws [ObjectGridRuntimeException](#)

Batch remove from the Map. If a key in the list cannot be found, it will be ignored.

See [ObjectMap.removeAll\(Collection\)](#) for additional specification details.

Parameters:

keyList - A list of keys for identifying which entries to remove

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[remove\(Object\)](#), [ObjectMap.removeAll\(Collection\)](#)

invalidate

```
void invalidate(Object key,  
              boolean isGlobal)  
              throws ObjectGridRuntimeException
```

Invalidates an entry in the cache based on the key parameter.

If the key's value has changes pending in the JavaMap, it is the application's responsibility to flush these changes to the Loader before invalidation. If a flush is not performed prior to invoking the invalidate operation, all pending changes for this key will be removed from the JavaMap. If the key cannot be found in the map, it will be ignored.

The isGlobal parameter is used to indicate which cache level is used to invalidate the entries. If isGlobal is true, when the transaction is committed, the key is removed from the BackingMap also. If a subsequent get operation is performed, the BackingMap will be skipped and the Loader will be used to get the data. If isGlobal is false, the entry is only invalidated in the JavaMap (transactional cache). If a subsequent get operation is performed, the BackingMap can be used; and, if it's not in the BackingMap, the Loader will be used to get the data.

A typical use of isGlobal being false is when a large number of records are touched in a transaction and the application wants to evict records that are no longer used in the cache.

See [ObjectMap.invalidate\(Object, boolean\)](#) for additional specification details.

Parameters:

key - Object representing the key to be used for cache entry invalidation
isGlobal - Indicates whether to remove the entry from the BackingMap (true) or just the JavaMap (false).

Throws:

[IllegalArgumentException](#) - if key is null
[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.invalidate\(Object, boolean\)](#)

invalidateAll

```
void invalidateAll(Collection keyList,  
                 boolean isGlobal)  
                 throws ObjectGridRuntimeException
```

Invalidate a set of cache entries based on the Collection of keys provided. If a key in the collection cannot be found, it will be ignored.

See [ObjectMap.invalidateAll\(Collection, boolean\)](#) for additional specification details.

Parameters:

keyList - A Collection of keys representing the entries to be invalidated
isGlobal - Indicates whether to remove the entry from the BackingMap (true) or just the JavaMap (false).

Throws:

[IllegalArgumentException](#) - if keyList is null or contains a null element.
[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[invalidate\(Object, boolean\)](#), [ObjectMap.invalidateAll\(Collection, boolean\)](#)

setTimeToLive

```
int setTimeToLive(int ttl)
```

Establishes the number of seconds that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this `JavaMap`, any previous value set by the `BackingMap.setTimeToLive(int)` method is overridden for this `JavaMap`. If this method is never called on the `JavaMap`, the TTL value from the `BackingMap` setting is used by default. If TTL is never set on the `BackingMap`, the cache entry can live "forever".

This method can only be used when the `TTLType` is set to `LAST_ACCESS_TIME` on the `BackingMap`. If this method is called on the `JavaMap` and the `TTLType` is something other than `LAST_ACCESS_TIME`, an `IllegalStateException` is thrown.

Parameters:

`ttl` - is the time-to-live value in seconds. The value must be ≥ 0 . A value of 0 is used to indicate the cache entry can live "forever". Use of the constant `ObjectMap.TTL_FOREVER` is recommended when "forever" is desired.

Returns:

previous time-to-live value in seconds. The constant `ObjectMap.TTL_FOREVER` can be used to determine if the previous TTL was set to "forever".

Throws:

[IllegalArgumentException](#) - if seconds argument is < 0 .

[IllegalStateException](#) - if `BackingMap.getTtlEvictorType()` returns anything other than `TTLType.LAST_ACCESS_TIME`.

See Also:

[BackingMap.setTimeToLive\(int\)](#), [ObjectMap.setTimeToLive\(int\)](#), [TTLType.LAST_ACCESS_TIME](#)

update

```
void update(Object key,  
            Object value)  
    throws ObjectGridRuntimeException
```

Performs an explicit update of a given entry.

A get operation is not required prior to invoking the update method (unlike the `put` method). Also, an update invocation will never insert a new record. If a the map's `LockStrategy` is `LockStrategy.OPTIMISTIC` this method will implicitly get the entry so as to have the version value of the object for when this method was invoked. Whether or not a copy of the object is made when transaction is committed is determined by the `CopyMode` setting for this map. See `CopyMode` for a description of each possible `CopyMode`.

If a key cannot be found in the map during commit, a `TransactionException` will be thrown.

See [ObjectMap.update\(Object, Object\)](#) for additional specification details.

Parameters:

`key` - Identifies the entry to be updated
`value` - The updated value for this entry

Throws:

[IllegalArgumentException](#) - if `key` is `null` or if the map does not allow `null` values and `value` is `null`.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[insert\(Object, Object\)](#), [put\(Object, Object\)](#), [CopyMode](#), [LockStrategy.OPTIMISTIC](#),

insert

```
void insert(Object key,  
           Object value)  
    throws ObjectGridRuntimeException
```

Performs an explicit insert of a given entry.

The key must not exist before executing this method. Also, an insert invocation will never update an existing record. Whether or not a copy of the object is made when a transaction is committed is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode.

If the key is already in the map, a TransactionException will be thrown during commit.

See [ObjectMap.insert\(Object, Object\)](#) for additional specification details.

Parameters:

key - Identifies the entry to be inserted
value - The value for this entry

Throws:

[IllegalArgumentException](#) - if key is null or if the map does not allow null values and value is null.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[put\(Object, Object\)](#), [update\(Object, Object\)](#), [CopyMode](#), [ObjectMap.insert\(Object, Object\)](#)

getIndex

```
Object getIndex(String name)  
    throws com.ibm.websphere.objectgrid.IndexUndefinedException,  
           com.ibm.websphere.objectgrid.IndexNotReadyException,  
           UnsupportedOperationException
```

Returns a reference to the named index that can be used with this Map. This index cannot be shared between threads and works on the same rules as Session. The returned value should be cast to the right index interface such as MapIndex, MapRangeIndex or a custom index interface such as a geo spatial index.

Parameters:

name - The index name

Returns:

A reference to the index, it must be cast to the appropriate index interface.

Throws:

[IndexUndefinedException](#) - if the index is not defined on the BackingMap

[IndexNotReadyException](#) - if the index is a dynamic index and it is not ready

[UnsupportedOperationException](#) - if the map is a distributed map

Since:

WAS XD 6.0.1

flush

```
void flush()  
    throws ObjectGridRuntimeException
```

Pushes the current set of changes for the JavaMap instance to the Loader without committing the changes. The changes are not propagated to the BackingMap either. This is useful for re-priming the Loader's data without committing the current transaction and

starting over.

Throws:

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[Session.flush\(\)](#), [ObjectMap.flush\(\)](#)

size

int `size()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[size](#) in interface [Map](#)

Returns:

the number of key-value mappings in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

isEmpty

boolean `isEmpty()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[isEmpty](#) in interface [Map](#)

Returns:

true if this map contains no key-value mappings.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

containsKey

boolean `containsKey(Object key)`

Returns true if this map contains a mapping for the specified key. ObjectGrid does not support null keys. If you configured the map to support null values, this method can be used to determine whether a key is contained in the map or not.

This API does not hold any locks when using pessimistic locking.

See [ObjectMap.containsKey\(Object\)](#) for additional specification details.

Specified by:

[containsKey](#) in interface [Map](#)

Parameters:

key - key whose presence in this map is to be tested.

Returns:

true if this map contains a mapping for the specified key.

Throws:

[IllegalArgumentException](#) - if null key parameter is passed in

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.containsKey\(Object\)](#)

containsValue

boolean **containsValue**([Object](#) value)

This method of the `java.util.Map` interface is not supported.

Specified by:

[containsValue](#) in interface [Map](#)

Parameters:

value - value whose presence in this map is to be tested.

Returns:

true if this map maps one or more keys to the specified value.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

get

[Object](#) **get**([Object](#) key)

Retrieves the object from the cache at the given key.

Whether or not a copy of the object is returned is determined by the CopyMode setting for this map. See CopyMode for a description of each possible CopyMode. If the key cannot be found in the map, a null value will be returned. A null value is also returned if a value is null and this map allows null values. To distinguish the two, use the `containsKey` method.

The return value is a SerializedValue when using the [CopyMode.COPY_TO_BYTES_RAW](#) CopyMode or `OutputFormat.RAW` OutputFormat with a ValueSerializerPlugin plug-in defined on the [BackingMap](#). The SerializedValue allows access to the value in its serialized form, or its native Java Object form.

The return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

See [ObjectMap.get\(Object\)](#) for additional specification details.

Specified by:

[get](#) in interface [Map](#)

Parameters:

key - The entry to fetch

Returns:

the value, null, SerializedValue OR Tuple

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[containsKey\(Object\)](#), [getForUpdate\(Object\)](#), [CopyMode](#), [ObjectMap.get\(Object\)](#)

put

[Object](#) **put**([Object](#) key,
[Object](#) value)

Puts the Object value into the cache at location represented by key.

The values will be pushed down to the BackingMap/Loader at commit time and has two behaviors, which can be altered using the [ObjectMap.setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSERTUPDATE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSER](#) The values are put into the map using the specification of the [ObjectMap.putAll\(LinkedHashMap\)](#).

Whether or not a copy of the object is made when transaction is committed is determined by the copy mode setting for this map. See [CopyMode](#) for a description of each possible copy mode.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an an [EntityManager](#) API entity is associated with the [BackingMap](#).

See [ObjectMap.put\(Object, Object\)](#) for additional specification details.

Specified by:

[put](#) in interface [Map](#)

Parameters:

key - The entry to put into the map

value - The value to put into the map using the key

Returns:

If [ObjectMap.PutMode.INSERTUPDATE](#) is set, return the previous value in this transaction.

If [ObjectMap.PutMode.UPSER](#) is set, the return value is null.

Throws:

[IllegalArgumentException](#) - if key is null, or if the map does not allow null values and value is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[CopyMode](#), [ObjectMap.put\(Object, Object\)](#)

remove

[Object](#) [remove\(Object key\)](#)

Removes the [Object](#) value from the cache represented by key.

This removal will be pushed down to the [BackingMap/Loader](#) at commit time. If the key cannot be found in the map, a null value will be returned.

The return value is a [SerializedValue](#) when using the [CopyMode.COPY_TO_BYTES_RAW](#) [CopyMode](#) or [OutputFormat.RAW](#) [OutputFormat](#) with a [ValueSerializerPlugin](#) plug-in defined on the [BackingMap](#). The [SerializedValue](#) allows access to the value in its serialized form, or its native Java Object form.

The return value is a [Tuple](#) when an an [EntityManager](#) API entity is associated with the [BackingMap](#).

See [ObjectMap.remove\(Object\)](#) for additional specification details.

Specified by:

[remove](#) in interface [Map](#)

Parameters:

key - The entry to remove

Returns:

the current value at invocation time

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.remove\(Object\)](#)

putAll

void **putAll**([Map](#) map)

Puts each of the Object values into the cache at location represented by the corresponding key contained in the Map.

The value will be pushed down to the BackingMap/Loader at commit time and has two behaviors, which can be altered using the [ObjectMap.setPutMode\(PutMode\)](#) property:

[ObjectMap.PutMode.INSE](#) (Deprecated) A put without a preceding get is an insert. For an entry in a map, a put following a get is always an update. However, if the entry is not in the map, a put following a get is an insert.

[ObjectMap.PutMode.UPSE](#) The value is put into the map using the specification of the [ObjectMap.upsert\(Object, Object\)](#).

Whether or not a copy of the objects contained in the map is made when transaction is committed is determined by the copy mode setting for this map. See CopyMode for a description of each possible copy mode.

An existing Map object will be passed in to use for obtaining the keys and values to be inserted or updated into the existing Map.

See [ObjectMap.putAll\(Map\)](#) for additional specification details.

Specified by:

[putAll](#) in interface [Map](#)

Parameters:

map - The key/values to be put into the map.

Throws:

[IllegalArgumentException](#) - if map is null or contains a null key or if null values are not allowed and map contains a null value.

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[put\(Object, Object\)](#), [ObjectMap.putAll\(Map\)](#)

clear

void **clear**()

Clear all keys from the Map.

This method is an auto-commit call, so a session should not be explicitly begun or committed when calling clear on the Map.

Specified by:

[clear](#) in interface [Map](#)

Throws:

[ObjectGridRuntimeException](#) - if an error occurs during processing

Since:

keySet

[Set](#) `keySet()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[keySet](#) in interface [Map](#)

Returns:

a set view of the keys contained in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

values

[Collection](#) `values()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[values](#) in interface [Map](#)

Returns:

a collection view of the values contained in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

entrySet

[Set](#) `entrySet()`

This method of the `java.util.Map` interface is not supported.

Specified by:

[entrySet](#) in interface [Map](#)

Returns:

a set view of the mappings contained in this map.

Throws:

[UnsupportedOperationException](#) - indicating this method is not supported

touch

`void touch(Object key)`

Updates the last access time in the `BackingMap` without retrieving the value to the `JavaMap`.

The last access time is updated during commit. If the key does not exist in the `BackingMap`, a `TransactionException` will be returned during commit processing.

See [ObjectMap.touch\(Object\)](#) for additional specification details.

Parameters:

key - key to be touched

Throws:

[IllegalArgumentException](#) - if key is null

[ObjectGridRuntimeException](#) - if an error occurs during processing

See Also:

[ObjectMap.touch\(Object\)](#)

setCopyMode

```
void setCopyMode(CopyMode copyMode,  
                Class valueInterface)  
    throws ObjectGridRuntimeException
```

Allows the CopyMode for the Map to be overridden on this map on this session only.

This method allows an application to use an optimal CopyMode TRANSACTION by TRANSACTION as its needs dictate. The CopyMode cannot be changed during a transaction. There must be no active transaction when this method is called.

Parameters:

copyMode - must be one of the final static variables defined in CopyMode. See CopyMode class for an explanation of each mode and how the valueInterface is used for

CopyMode.COPY_ON_WRITE .

valueInterface - the value interface Class object. Specify null in version 7.1 and later.

Throws:

[IllegalArgumentException](#) - if copyMode is null or COPY_ON_WRITE CopyMode is specified and the required value interface parameter is null

[ObjectGridRuntimeException](#) - if a transaction is active and this map has already been used in the transaction or an error occurs during processing

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\)](#), [CopyMode](#), [ObjectMap.setCopyMode\(CopyMode, Class\)](#)

clearCopyMode

```
void clearCopyMode()  
    throws ObjectGridRuntimeException
```

Resets the CopyMode back to the one in the BackingMap.

This method is used to reverse a previous setCopyMode method call for this JavaMap. This method can only be called when no transaction is active on the associated session.

Throws:

[ObjectGridRuntimeException](#) - if a transaction is active and this map has already been used in the transaction or an error occurs during processing

See Also:

[setCopyMode\(CopyMode, Class\)](#), [ObjectMap.clearCopyMode\(\)](#)

getNextKey

```
Object getNextKey(long timeout)
```

Retrieves a key off the map in first-in-first-out (FIFO) insert order. The entry is locked by the session such that other calls to getNextKey will not return the same key. The key can be used to remove or manipulate the value although leaving the entry will result in the key remaining at the beginning of the queue. This order is optimized for performance and is not guaranteed especially across partitions or in highly concurrent environments.

The return value is a SerializedKey when OutputFormat.RAW is set for the keys. The default key output format for maps that are associated with a KeySerializerPlugin is OutputFormat.RAW. The SerializedKey allows access to the value in its serialized form, or

its native Java Object form.

The return value is a Tuple when an an EntityManager API entity is associated with the BackingMap.

See [ObjectMap.getNextKey\(long\)](#) for additional specification details.

Parameters:

timeout - The period of time to wait for an entry to become available on the queue.

Returns:

The next available key in the map.

See Also:

[ObjectMap.getNextKey\(long\)](#)

getEntityMetadata

com.ibm.websphere.projector.md.EntityMetadata **getEntityMetadata()**

Retrieve the metadata for the entity associated with this map.

Returns:

the EntityMetadata if an entity is associated with this map or null if there is no entity associated with this map.

Since:

WAS XD 6.1

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
ew	ge	ss	ed	ted			
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All		
			Classes				

SUMMARY: NESTED | FIELD | CONSTR | [METH](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

com.ibm.websphere.objectgrid

Interface IObjectGridException

All Known Implementing Classes:

[CacheEntryException](#), [CannotGenerateCredentialException](#), [ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#), [com.ibm.websphere.objectgrid.ClientServerTransactionCallbackException](#), [ConnectException](#), [ContinuousQueryAttributePathException](#), [ContinuousQueryException](#), [ContinuousQueryGetValueException](#), [ContinuousQueryIncompatibleDuplicateException](#), [DuplicateKeyException](#), [ExpiredCredentialException](#), [InvalidCredentialException](#), [KeyNotFoundException](#), [LoaderException](#), [LockDeadlockException](#), [LockException](#), [LockInternalFailureException](#), [LockTimeoutException](#), [NoActiveTransactionException](#), [ObjectGridConfigurationException](#), [ObjectGridException](#), [ObjectGridRuntimeException](#), [ObjectGridSecurityException](#), [ReadOnlyException](#), [ReplicationVotedToRollbackTransactionException](#), [SessionNotReentrantException](#), [TransactionAffinityException](#), [TransactionAlreadyActiveException](#), [TransactionCallbackException](#), [TransactionException](#), [TransactionQuiesceException](#), [TransactionTimeoutException](#), [UnavailableServiceException](#), [UndefinedMapException](#)

public interface **IObjectGridException**

This interface is used to ensure JDK 1.4 Throwable chaining behavior for all exceptions thrown by ObjectGrid even when an earlier JDK is used (e.g. JDK 1.3.1).

Since:

WAS XD 6.0.1, XC10

Method Summary

Throwable

[getCause\(\)](#)

Provides JDK 1.4 Throwable.getCause() behavior.

Throwable

[initCause\(Throwable cause\)](#)

Provides JDK 1.4 Throwable.initCause() behavior.

Method Detail

getCause

[Throwable](#) `getCause()`

Provides JDK 1.4 `Throwable.getCause()` behavior.

Returns the cause of this throwable or `null` if the cause is nonexistent or unknown. (The cause is the throwable that caused this throwable to get thrown.)

This implementation returns the cause that was supplied via one of the constructors requiring a `Throwable`, or that was set after creation with the `initCause(Throwable)` method. While it is typically unnecessary to override this method, a subclass can override it to return a cause set by some other means. This is appropriate for a "legacy chained throwable" that predates the addition of chained exceptions to `Throwable`. Note that it is *not* necessary to override any of the `PrintStackTrace` methods, all of which invoke the `getCause` method to determine the cause of a throwable.

Returns:

the cause of this throwable or `null` if the cause is nonexistent or unknown.

See Also:

[initCause\(Throwable\)](#)

initCause

[Throwable](#) `initCause(Throwable cause)`
throws [IllegalArgumentException](#),
[IllegalStateException](#)

Provides JDK 1.4 `Throwable.initCause()` behavior.

Initializes the *cause* of this throwable to the specified value. (The cause is the throwable that caused this throwable to get thrown.)

This method can be called at most once. It is generally called from within the constructor, or immediately after creating the throwable. If this throwable was created with `Throwable(Throwable)` or `Throwable(String,Throwable)`, this method cannot be called even once.

Parameters:

cause - the cause (which is saved for later retrieval by the `getCause()` method). (A `null` value is permitted, and indicates that the cause is nonexistent or unknown.)

Returns:

a reference to this `Throwable` instance.

Throws:

[IllegalArgumentException](#) - if cause is this throwable. (A throwable cannot be its own cause.)

[IllegalStateException](#) - if this throwable was created with `Throwable(Throwable)` or `Throwable(String,Throwable)`, or this method has already been called on this throwable.

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

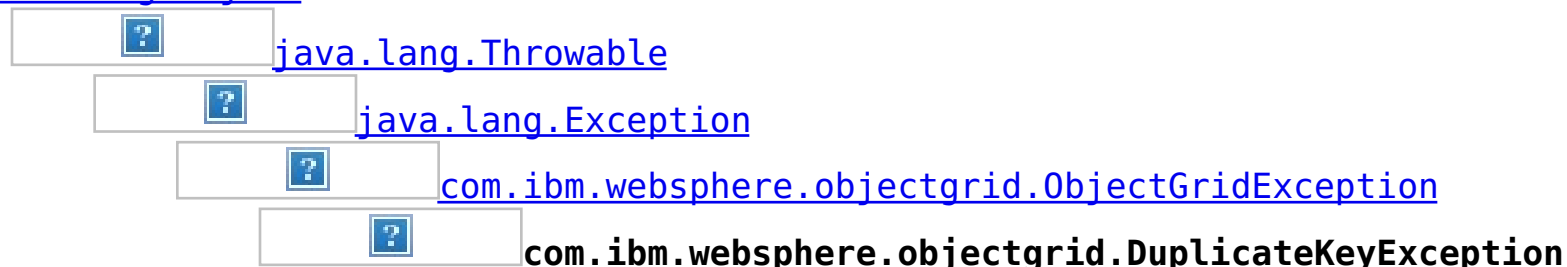
SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#) DETAIL: FIELD | CONSTR | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class DuplicateKeyException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class DuplicateKeyException
extends ObjectGridException
```

A DuplicateKeyException exception is thrown if a key cannot be inserted into a BackingMap because an object with the same key already exists.

Since:

WAS XD 6.0, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[DuplicateKeyException](#)()

Constructs a new DuplicateKeyException with null as its detail message.

[DuplicateKeyException](#)(String message)

Constructs a new DuplicateKeyException with the specified detail message.

[DuplicateKeyException](#)(String message, Throwable cause)

Constructs a new DuplicateKeyException with the specified detail message and cause.

[DuplicateKeyException](#)(Throwable cause)

Constructs a new DuplicateKeyException with a specified cause.

Method Summary

Methods inherited from class com.ibm.websphere.objectgrid.[ObjectGridException](#)

[getCause](#), [initCause](#)

Methods inherited from class java.lang.[Throwable](#)

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

DuplicateKeyException

```
public DuplicateKeyException()
```

Constructs a new DuplicateKeyException with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

DuplicateKeyException

```
public DuplicateKeyException(String message)
```

Constructs a new DuplicateKeyException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

DuplicateKeyException

```
public DuplicateKeyException(String message,
                             Throwable cause)
```

Constructs a new DuplicateKeyException with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this DuplicateKeyException's detail message.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`Null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

DuplicateKeyException

```
public DuplicateKeyException(Throwable cause)
```

Constructs a new DuplicateKeyException with a specified cause. The cause and a detail message of `(cause==null ? null : cause.toString())` is used (which typically contains the class and detail message of `cause`). This constructor is useful for DuplicateKeyExceptions that are little more than wrappers for other throwables.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All
Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid
Class CopyMode

[java.lang.Object](#)



```
public final class CopyMode
extends Object
```

This class is used to define the "copy" mode when the setCopyMode method of the BackingMap interface is used. The application is expected to pass one of the final static variables that are defined in this class to the setCopyMode method.

Since:

WAS XD 6.0, XC10

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\)](#), [ObjectTransformer.copyValue\(Object\)](#)

Field Summary

s t a t i c	<p>COPY_ON_READ The COPY_ON_READ mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when a transaction is committed.</p>
c o p y	<p>COPY_ON_READ_AND_COMMIT The COPY_ON_READ_AND_COMMIT mode is the default mode.</p>
o d e	<p>COPY_ON_WRITE The COPY_ON_WRITE mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when ObjectMap.get is called for the first time by a transaction for a given key.</p>

M
o
d
e

s
t
a
t
i
c
C
o
p
y
M
o
d
e

[COPY_TO_BYTES](#)

The COPY_TO_BYTES mode is similar to the COPY_ON_READ_AND_COMMIT mode in that it ensures that an application never has a reference to the value object that is in the BackingMap.

s
t
a
t
i
c
C
o
p
y
M
o
d
e

[COPY_TO_BYTES_RAW](#)

When set, all ObjectMap APIs that return a SerializedValue rather than the original Java Object, allowing access to the serialized form of the data, preventing inflation of object into Java Object form.

s
t
a
t
i
c
C
o
p
y
M
o
d
e

[NO_COPY](#)

The NO_COPY mode allows an application to promise that it will never modify a value object obtained using an ObjectMap.get method in exchange for performance improvements.

Method Summary

b
o
o
l
e
a
n

[isBytes\(\)](#)

Is the copy mode one of the copy modes that indicate copy to bytes?

S
t
r
i
n
g

[toString\(\)](#)

Returns a string representation of the CopyMode.

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

COPY_ON_READ_AND_COMMIT

```
public static final CopyMode COPY_ON_READ_AND_COMMIT
```

The COPY_ON_READ_AND_COMMIT mode is the default mode. This mode ensures that an application never has a reference to the value object that is in the BackingMap, and instead the application is always working with a copy of the value that is in the BackingMap. The copy ensures the application can never inadvertently corrupt the data that is cached in the BackingMap. When an application transaction calls an `ObjectMap.get` method for a given key, and it is the first access of the `ObjectMap` entry for that key, a copy of the value is returned. When the transaction is committed, any changes the application committed are copied to the BackingMap to ensure that the application does not have reference to the committed value in the BackingMap.

COPY_ON_READ

```
public static final CopyMode COPY_ON_READ
```

The COPY_ON_READ mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when a transaction is committed. To preserve integrity of BackingMap data, the application promises to destroy every reference it has to an entry once the transaction is committed. This mode results in a `ObjectMap.get` method returning a copy of the value rather than a reference to the value to ensure that changes made by the application to the value does not affect the BackingMap value until the transaction is committed. However, when the transaction does commit, a copy of changes is not made. Instead, the reference to the copy that was returned by `ObjectMap.get` is stored in the BackingMap. This is the reason the application must agree to destroy all map entry references once the transaction is committed. If application fails to keep its promise, the application could cause the data cached in BackingMap to become corrupted. If an application is using this mode and it is having problems, then switch to the COPY_ON_READ_AND_COMMIT mode to see if the problem still exists. If the problem goes away, then more than likely the application is failing to destroy all of its references after the transaction has committed.

COPY_ON_WRITE

```
public static final CopyMode COPY_ON_WRITE
```

The COPY_ON_WRITE mode improves performance over the COPY_ON_READ_AND_COMMIT mode by eliminating the copy that occurs when `ObjectMap.get` is called for the first time by a transaction for a given key. Instead, the `ObjectMap.get` method returns a proxy to the value rather than a direct reference to the value object itself. The proxy ensures that a copy of the value is not made unless the application calls a set method on the value interface that is passed on the `BackingMap.setCopyMode(CopyMode, Class)` method. Thus, the proxy provides a "copy on write" implementation. When a transaction commits, the BackingMap examines the proxy to determine if any copy was made as a result of a set method being called. If a copy was made, then the reference to that copy is stored in the BackingMap. The big advantage of this mode is a value is never copied on read or at commit when the transaction never calls a set method to mutate the value.

See Also:

[BackingMap.setCopyMode\(CopyMode, Class\)](#)

NO_COPY

```
public static final CopyMode NO_COPY
```

The NO_COPY mode allows an application to promise that it will never modify a value

object obtained using an `ObjectMap.get` method in exchange for performance improvements. If this mode is used, no copy of the value is ever made. If the application breaks its promise and does modify values, then data in the `BackingMap` will be corrupted. This mode is primarily useful for read only maps where data is never modified by the application. If the application is using this mode and it is having problems, then switch to `COPY_ON_READ_AND_COMMIT` mode to see if the problem still exists. If the problem goes away, then more than likely the application is not keeping its promise and is modifying the value returned by `ObjectMap.get` method (either during transaction or after transaction has committed).

COPY_TO_BYTES

```
public static final CopyMode COPY_TO_BYTES
```

The `COPY_TO_BYTES` mode is similar to the `COPY_ON_READ_AND_COMMIT` mode in that it ensures that an application never has a reference to the value object that is in the `BackingMap`. The value that the application works with is a newly inflated version of the serialized version that is in the `BackingMap`. The copy ensures the application can never inadvertently corrupt the data that is cached in the `BackingMap` since a byte form of the value is what is stored in the `BackingMap` instead of the Object form.

A copy of the value is returned when an application transaction calls an `ObjectMap.get` method for a given key, and it is the first time that the `ObjectMap` entry is accessed for that key. When the transaction is committed, any changes the application committed are copied to bytes in the `BackingMap` to ensure that the application does not have reference to the committed value in the `BackingMap`.

Since:
7.0

COPY_TO_BYTES_RAW

```
public static final CopyMode COPY_TO_BYTES_RAW
```

When set, all `ObjectMap` APIs that return a `SerializedValue` rather than the original Java Object, allowing access to the serialized form of the data, preventing inflation of object into Java Object form.

Since:
7.1.1
See Also:
`ValueDataSerializer`

Method Detail

toString

```
public String toString()
```

Returns a string representation of the `CopyMode`.

Overrides:
[toString](#) in class [Object](#)

Returns:
a string representation of the `CopyMode`.

isBytes

public boolean **isBytes()**

Is the copy mode one of the copy modes that indicate copy to bytes?

Returns:

boolean indicating if copy mode is one of COPY_TO_BYTES or COPY_TO_BYTES_RAW.

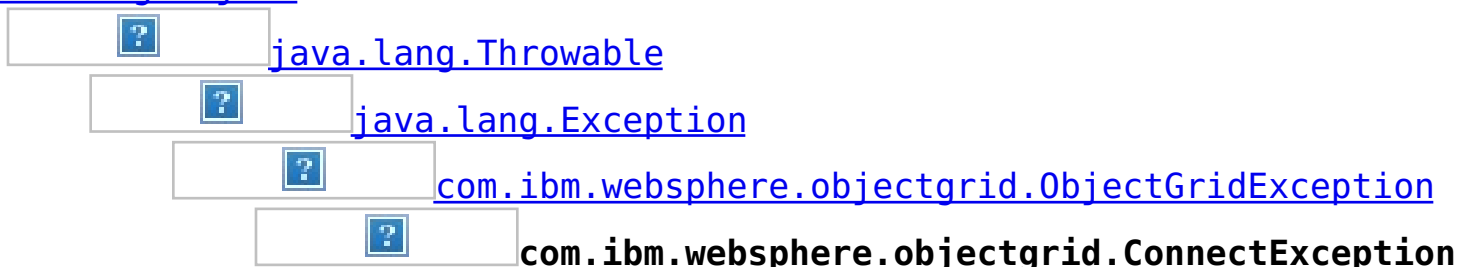
Overview	Package	Classes	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES NO FRAMES All Classes					
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD							
OD							

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class ConnectException

[java.lang.Object](#)



All Implemented Interfaces:

[IObjectGridException](#), [Serializable](#)

```
public class ConnectException
extends ObjectGridException
```

This exception is used to indicate that the client was unable to connect to the server

Since:

WAS XD 6.0.1, XC10

See Also:

[Serialized Form](#)

Field Summary	
s t a t i c i n t	<p>BAD_CONFIGURATION This failure code indicates the provided configuration was corrupt.</p>
s t a t i c i n t	<p>CONNECTION_REFUSED This failure code indicates that the server may not be available.</p>
s t a t i c i n t	<p>FAILED_SECURITY This failure code indicates the a failure to authenticate.</p>
s	

t a t i c i n t	<p>SERVER_DEFINITION_NOT_FOUND This failure code indicates the definition of cluster cannot be accessed.</p>
s t a t i c i n t	<p>UNKNOWN This failure code indicates the reason for the connect failure is unknown.</p>

Constructor Summary

ConnectException ()	Constructs a new <code>ConnectException</code> with <code>null</code> as its detail message.
ConnectException (<code>String</code> message)	Constructs a new <code>ConnectException</code> with the specified detail message.
ConnectException (<code>String</code> message, <code>int</code> failureCode)	Constructs a new <code>ConnectException</code> with the specified detail message.
ConnectException (<code>String</code> message, <code>Throwable</code> cause)	Constructs a new <code>ConnectException</code> with the specified detail message and cause.
ConnectException (<code>String</code> message, <code>Throwable</code> cause, <code>int</code> failureCode)	Constructs a new <code>ConnectException</code> with the specified detail message and cause.
ConnectException (<code>Throwable</code> cause)	Constructs a new <code>ConnectException</code> with a specified cause.

Method Summary

i n t	<p>getFailureCode() Returns the failure code that was set by one of the constructors that accepts a failure code, or <code>UNKNOWN</code> if one of the other constructors was called.</p>
-------------	--

Methods inherited from class `com.ibm.websphere.objectgrid.ObjectGridException`
[getCause](#), [initCause](#)

Methods inherited from class `java.lang.Throwable`
[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`
[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

BAD_CONFIGURATION

```
public static final int BAD_CONFIGURATION
```

This failure code indicates the provided configuration was corrupt.

See Also:

[Constant Field Values](#)

UNKNOWN

```
public static final int UNKNOWN
```

This failure code indicates the reason for the connect failure is unknown.

See Also:

[Constant Field Values](#)

FAILED_SECURITY

```
public static final int FAILED_SECURITY
```

This failure code indicates the a failure to authenticate.

See Also:

[Constant Field Values](#)

CONNECTION_REFUSED

```
public static final int CONNECTION_REFUSED
```

This failure code indicates that the server may not be available.

See Also:

[Constant Field Values](#)

SERVER_DEFINITION_NOT_FOUND

```
public static final int SERVER_DEFINITION_NOT_FOUND
```

This failure code indicates the definition of cluster cannot be accessed.

See Also:

[Constant Field Values](#)

Constructor Detail

ConnectException

```
public ConnectException()
```

Constructs a new `ConnectException` with `null` as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method. In addition the failure code is initialized to `UNKNOWN`.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [UNKNOWN](#)

ConnectException

```
public ConnectException(String message)
```

Constructs a new `ConnectException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method. In addition the failure code is initialized to `UNKNOWN`.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [UNKNOWN](#)

ConnectException

```
public ConnectException(String message,  
                        int failureCode)
```

Constructs a new `ConnectException` with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

`message` - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

`failureCode` - the failure code which should be one of the constants of this exception class.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#), [getFailureCode\(\)](#)

ConnectException

```
public ConnectException(Throwable cause)
```

Constructs a new `ConnectException` with a specified cause. The cause and a detail message of (`cause==null ? null : cause.toString()`) is used (which typically contains the class and detail message of `cause`). This constructor is useful for `ConnectExceptions` that are little more than wrappers for other throwables. The failure code is initialized to `UNKNOWN`.

Parameters:

`cause` - is the exception that caused this exception to be thrown, which is saved for later retrieval by the `getCause()` method. A `null` value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#), [UNKNOWN](#)

ConnectException

```
public ConnectException(String message,  
                        Throwable cause)
```

Constructs a new `ConnectException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated in this `ConnectException`'s detail message. The failure code is initialized to `UNKNOWN`.

Parameters:

`message` - the detail message (which is saved for later retrieval by the `getMessage` method).

`cause` - the cause (which is saved for later retrieval by the `getCause` method). (`null` value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

ConnectException

```
public ConnectException(String message,  
                        Throwable cause,  
                        int failureCode)
```

Constructs a new ConnectException with the specified detail message and cause.

Note that the detail message associated with *cause* is *not* automatically incorporated in this ConnectException's detail message. The failure code is initialized to UNKNOWN.

Parameters:

message - the detail message (which is saved for later retrieval by the `getMessage` method).

cause - the cause (which is saved for later retrieval by the `getCause` method). (A `null` value is permitted, and indicates that the cause is nonexistent or unknown).

failureCode - the failure code which should be one of the constants of this exception class.

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#), [getFailureCode\(\)](#)

Method Detail

getFailureCode

```
public int getFailureCode()
```

Returns the failure code that was set by one of the constructors that accepts a failure code, or UNKNOWN if one of the other constructors was called.

Returns:

the failure code. One of the constants of this exception class.

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

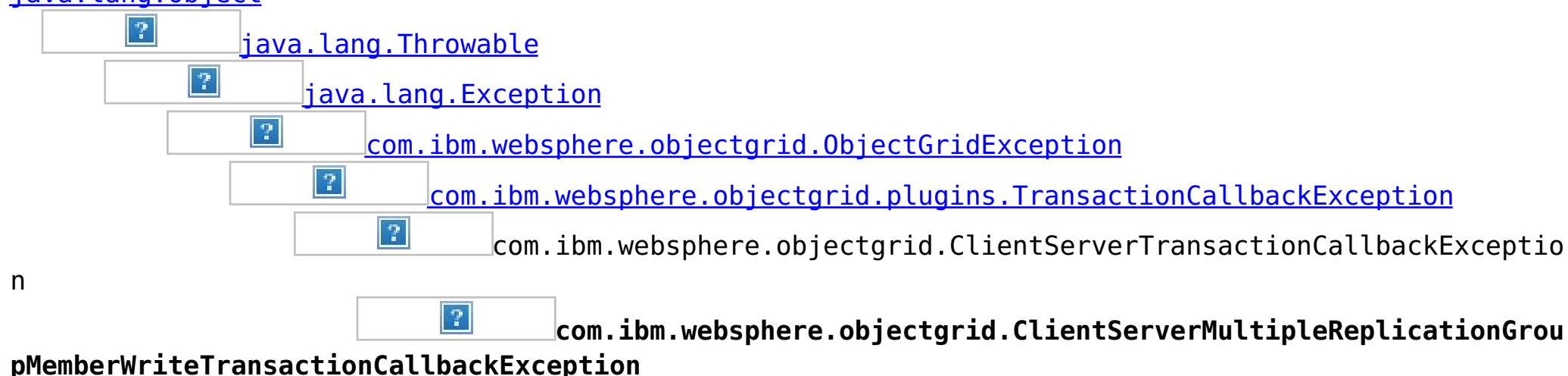
**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Class

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

[java.lang.Object](#)



All Implemented Interfaces:

[ObjectGridException](#), [Serializable](#)

```
public class ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException
extends com.ibm.websphere.objectgrid.ClientServerTransactionCallbackException
```

This exception is thrown when a method call to the Client/Server TransactionCallback detects the user is attempting to perform a write against multiple maps in different Map Sets, Partition Sets or Replication groups. This is not allowed.

Since:

WAS XD 6.0.1, XC10

See Also:

[TransactionCallback](#), [Serialized Form](#)

Constructor Summary

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)()

Constructs a new `ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException` with `null` as its detail message.

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)([String](#) message)

Constructs a new `ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException` with the specified detail message.

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)([String](#) message, [Throwable](#) cause)

Constructs a new `ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException` with the specified detail message and cause.

[ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException](#)([Throwable](#) cause)

Constructs a new

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with a specified cause.

Method Summary

Methods inherited from class `com.ibm.websphere.objectgrid.ObjectGridException`

[getCause](#), [initCause](#)

Methods inherited from class `java.lang.Throwable`

[fillInStackTrace](#), [getLocalizedMessage](#), [getMessage](#), [getStackTrace](#), [printStackTrace](#), [printStackTrace](#), [printStackTrace](#), [setStackTrace](#), [toString](#)

Methods inherited from class `java.lang.Object`

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException()
```

Constructs a new ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with null as its detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#)

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException(String message)
```

Constructs a new ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with the specified detail message. The cause is not initialized, and may subsequently be initialized by a call to the `initCause` method.

Parameters:

message - the detail message. The detail message is saved for later retrieval by the `getMessage` method.

See Also:

[ObjectGridException.initCause\(Throwable\)](#), [Throwable.getMessage\(\)](#)

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException(String message, Throwable cause)
```

Constructs a new

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with the specified detail message and cause.

Note that the detail message associated with cause is *not* automatically incorporated in this ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException's detail message.

Parameters:

message - the detail message (which is saved for later retrieval by the getMessage method).

cause - the cause (which is saved for later retrieval by the getCause method). (A null value is permitted, and indicates that the cause is nonexistent or unknown).

See Also:

[ObjectGridException.getCause\(\)](#), [Throwable.getMessage\(\)](#)

ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException

```
public ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException(Throwable cause)
```

Constructs a new ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackException with a specified cause. The cause and a detail message of (cause==null ? null : cause.toString()) is used (which typically contains the class and detail message of cause). This constructor is useful for ClientServerMultipleReplicationGroupMemberWriteTransactionCallbackExceptions that are little more than wrappers for other throwables.

Parameters:

cause - is the exception that caused this exception to be thrown, which is saved for later retrieval by the getCause() method. A null value is permitted and indicates that the cause is nonexistent or is unknown.

See Also:

[ObjectGridException.getCause\(\)](#)

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help
PREV CLASS	NEXT CLASS		FRAMES	NO FRAMES	All		
			Classes				
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD				
OD							

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface ClientReplicableMap

All Known Subinterfaces:

[BackingMap](#)

Deprecated. *The client replicated map function is deprecated in version 8.6. Use the [ContinuousQueryManager](#) function.*

```
public interface ClientReplicableMap
```

This interface represents a replicable client map. A replicable client map can be a continuous replica or a snapshot replica of the server map.

If the client is a continuous replica of the server map, the data in the server will be replicated to the client continuously in an asynchronous manner.

If the client is a snapshot replica of the server map, a snapshot on the data in the server will be taken and the snapshot will be replicated to the client in an asynchronous manner. A snapshot replication is a one-time replication.

A ReplicationMapListener can be used to listen for the data changes as well as the replication lifecycle events.

Since:

WAS XD 6.1, XC10

See Also:

[ReplicationMapListener](#)

Nested Class Summary

s
t
a
t
i
c
c
l
a
s
s

[ClientReplicableMap.Mode](#)

Deprecated. Client Replication mode

Field Summary

s
t
a
t
i
c
c
l
i
e

n
t
R
e
p
l
i
c
a
b
l
e
M
a
p
.
M
o
d
e

CONTINUOUS_REPLICATION

Deprecated. Full replication mode.

s
t
a
t
i
c
l
i
e
n
t
R
e
p
l
i
c
a
b
l
e
M
a
p
.
M
o
d
e

NONE

Deprecated. No replication mode, aka normal mode.

s
t
a
t
i
c
l
i
e
n
t
R
e
p
l
i
c
a
b
l
e
M
a
p
.
M
o
d
e

SNAPSHOT_REPLICATION

Deprecated. Snapshot replication mode.

Method Summary

[disableClientReplication\(\)](#)

Deprecated. Disables the replication for this client.

[enableClientReplication\(ClientReplicableMap.Mode mode, int\[\] partitions, ReplicationMapListener listener\)](#)

Deprecated. Make the client map a replica of the server side map.

[getReplicationMode\(\)](#)

Deprecated. Returns the current replication mode

Field Detail

NONE

static final [ClientReplicableMap.Mode](#) NONE

Deprecated.

No replication mode, aka normal mode.

CONTINUOUS_REPLICATION

static final [ClientReplicableMap.Mode](#) CONTINUOUS_REPLICATION

Deprecated.

Full replication mode. Data in the server map will be replicated to the client continuously.

SNAPSHOT_REPLICATION

static final [ClientReplicableMap.Mode](#) SNAPSHOT_REPLICATION

Deprecated.

Snapshot replication mode. A snapshot on the data in the server will be taken and the snapshot will be replicated to the client. A snapshot replication is a one-time replication.

Method Detail

enableClientReplication

```
void enableClientReplication(ClientReplicableMap.Mode mode,  
                           int[] partitions,  
                           ReplicationMapListener listener)  
    throws ObjectGridException
```

Deprecated.

Make the client map a replica of the server side map.

When security is enabled, this method requires a `ServerMapPermission` with action "replicate". Refer to `ServerMapPermission` for more permission details.

Required Client Permission: `ServerMapPermission.REPLICATE`

Parameters:

mode - The replication mode.

partitions - The array of partition IDs represent which partitions the data should be replicated from. If the value is null or an empty array, it indicates the data should be replicated from all partitions.

listener - a listener to receive client replication events

Throws:

[IllegalArgumentException](#) - if mode is not [CONTINUOUS_REPLICATION](#) or [SNAPSHOT_REPLICATION](#) or the map isn't currently in the mode specified or is not in [NONE](#) mode

[IllegalStateException](#) - if this method is invoked on a map other than a client map

[ObjectGridException](#) - if an error occurs during processing this request

See Also:

[ReplicationMapListener](#), [CONTINUOUS_REPLICATION](#), [SNAPSHOT_REPLICATION](#), [getReplicationMode\(\)](#), [BackingMap.CLIENT](#)

getReplicationMode

```
ClientReplicableMap.Mode getReplicationMode()
```

Deprecated.

Returns the current replication mode

Returns:

the replication mode

See Also:

[NONE](#), [CONTINUOUS_REPLICATION](#), [SNAPSHOT_REPLICATION](#)

disableClientReplication

```
void disableClientReplication()  
    throws ObjectGridException
```

Deprecated.

Disables the replication for this client. If it is not in a replication mode, this method will be a no-op.

When security is enabled, this method requires a `ServerMapPermission` with action "replicate". Refer to `ServerMapPermission` for more permission details.

Throws:

[IllegalStateException](#) - if this method is invoked on a map other than a client map
[ObjectGridException](#) - if an error occurs during processing this request

See Also:

[BackingMap.CLIENT](#)

Overview	Package	Classes	Serialized	Deprecated	Index	Help	<i>IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification</i>	
PREV CLASS	NEXT CLASS	FRAMES		NO FRAMES	All			
		Classes						
SUMMARY: NESTED FIELD CONSTR METH DETAIL: FIELD CONSTR METHOD								

[OD](#)

© Copyright International Business Machines Corp 2005,2013. All rights reserved.

com.ibm.websphere.objectgrid

Class ClientReplicableMap.Mode

[java.lang.Object](#)



Enclosing interface:

[ClientReplicableMap](#)

```
public static final class ClientReplicableMap.Mode  
extends Object
```

Client Replication mode

Method Summary

S t r i n g	toString()
----------------------------	----------------------------

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Method Detail

toString

```
public String toString()
```

Overrides:

[toString](#) in class [Object](#)

com.ibm.websphere.objectgrid
Interface ClientClusterContext

public interface **ClientClusterContext**

This interface is a context to represent which cluster/domain the client connected to using one of the `ObjectGridManager.connect` methods. An instance of this interface is used to retrieve client `ObjectGrid` instances and for performing admin operations against an `ObjectGrid` cluster/domain or its servers.

Since:

WAS XD 6.0.1, XC10

See Also:

[ObjectGridManager](#)

Method Summary

C l i e n t r o p e r t i e s	<p>getClientProperties(String objectGridName) Retrieve the <code>ClientProperties</code> object for this <code>ClientClusterContext</code> for the specified <code>ObjectGrid</code> name.</p>
S t r i n g	<p>getClusterName() Gets the name of the domain to which the client is connected</p>
v o i d	<p>setClientProperties(String objectGridName, URL url) Sets the <code>ClientProperties</code> properties for the selected <code>ObjectGrid</code> using the specified client properties file.</p>

Method Detail

getClusterName

[String](#) `getClusterName()`

Gets the name of the domain to which the client is connected

Returns:

the name of the domain this context is connected to

getClientProperties

[ClientProperties](#) `getClientProperties(String objectGridName)`

Retrieve the ClientProperties object for this ClientClusterContext for the specified ObjectGrid name. A ClientProperties is scoped to this ClientClusterContext and a single ObjectGrid.

Parameters:

objectGridName - the name of ObjectGrid

Returns:

the ClientProperties instance for this ObjectGrid.

Since:

WAS XD 6.1.0.3

setClientProperties

`void setClientProperties(String objectGridName,
URL url)`

Sets the ClientProperties properties for the selected ObjectGrid using the specified client properties file.

To further adjust the client properties, call the [getClientProperties\(String\)](#) method.

Parameters:

objectGridName - the name of ObjectGrid to apply the ClientProperties to.

url - the URL where the client properties file can be located.

Since:

WAS XD 6.1.0.3

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface CatalogDomainManager

public interface **CatalogDomainManager**

Provides access to catalog domain configuration information for the current environment.

When running in a WebSphere Application Server profile augmented with WebSphere eXtreme Scale, the CatalogDomainManager returns the catalog service domain configuration information that is configured in the administration console.

Since:

8.5, XC10

See Also:

[ObjectGridManager.getCatalogDomainManager\(\)](#)

Method Summary

C
a
t
a
l
o
g
D
o
m
a
i
n
I
n
f
o

[getDefaultDomainInfo\(\)](#)

Retrieve the default, configured CatalogDomainInfo.

C
a
t
a
l
o
g
D
o
m
a
i
n
I
n
f
o

[getDomainInfo\(String domainId\)](#)

Retrieve the specified CatalogDomainInfo for the specified domainId.

C
o
l
l

e
c
t
i
o
n
<
C
e
t
e
r
i
n
a
l
o
g
D
o
m
a
i
n
I
n
f
o
>

[getDomainInfos\(\)](#)

Retrieve all configured CatalogDomainInfo objects.

Method Detail

getDefaultDomainInfo

[CatalogDomainInfo](#) `getDefaultDomainInfo()`

Retrieve the default, configured CatalogDomainInfo.

Returns:

the default CatalogDomainInfo, or null if not available.

getDomainInfo

[CatalogDomainInfo](#) `getDomainInfo(String domainId)`

Retrieve the specified CatalogDomainInfo for the specified domainId.

Parameters:

domainId - the domain identifier.

Returns:

the CatalogDomainInfo if found, or null.

getDomainInfos

[Collection](#)<[CatalogDomainInfo](#)> `getDomainInfos()`

Retrieve all configured CatalogDomainInfo objects.

Returns:

a collection of CatalogDomainInfo

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface CatalogDomainInfo

public interface **CatalogDomainInfo**

Identifies the configuration attributes of a catalog service domain.

Since:

8.5, XC10

See Also:

[CatalogDomainManager](#)

Method Summary

S t r i n g	<p>getClientCatalogServerEndpoints() Retrieve the catalog server endpoints used to connect a client to the remote catalog service domain.</p>
C l i e n t S e c u r i t y C o n f i g u r e n c e	<p>getClientSecurityConfiguration() Retrieve the ClientSecurityConfiguration for the domain.</p>
S t r i n g	<p>getDomainId() Retrieve the identifier of the domain as specified by the catalog service domain.</p>

Method Detail

getDomainId

[String](#) `getDomainId()`

Retrieve the identifier of the domain as specified by the catalog service domain.

Note: This is different than the name of the domain which is specified when starting the catalog services.

Returns:
the identifier of the domain.

getClientCatalogServerEndpoints

[String](#) `getClientCatalogServerEndpoints()`

Retrieve the catalog server endpoints used to connect a client to the remote catalog service domain.

The catalog service endpoints are used with the [ObjectGridManager.connect\(String, com.ibm.websphere.objectgrid.security.config.ClientSecurityConfiguration, java.net.URL\)](#) method to connect to a catalog service domain.

Returns:
the catalog service endpoints.

getClientSecurityConfiguration

[ClientSecurityConfiguration](#) `getClientSecurityConfiguration()`

Retrieve the ClientSecurityConfiguration for the domain.

The ClientSecurityConfiguration are used with the [ObjectGridManager.connect\(String, com.ibm.websphere.objectgrid.security.config.ClientSecurityConfiguration, java.net.URL\)](#) method to connect to a catalog service domain.

Returns:
the ClientSecurityConfiguration or null if security is not configured.

[Overview](#) [Package](#) [Classes](#) [Tree](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.objectgrid

Interface BackingMap

All Superinterfaces:

[ClientReplicableMap](#)

```
public interface BackingMap  
extends ClientReplicableMap
```

This is the public interface to the BackingMap. It is returned when a new Map is defined on the ObjectGrid. It allows the Map to be customized with various plug-ins or by setting properties. The defaults are:

- No external Evictor, but an internal time-based evictor is provided by default
- No Loader
- No EventListeners
- No MapIndexPlugins
- An internal ObjectTransformer
- An internal OptimisticCallback
- Key is not copied
- A value CopyMode Of CopyMode.COPY_ON_READ_AND_COMMIT
- A LockStrategy of LockStrategy.OPTIMISTIC
- A default lock timeout
- null values are supported
- A default number of buckets
- A default number of lock buckets
- Synchronous preload
- Read/write map by default
- A TimeToLive of 0 (indicating unlimited time)
- A TtlEvictor type of TTLType.NONE
- Write-behind updates is disabled
- Time-based database updates are disabled
- Eviction triggers are not set

Since:

WAS XD 6.0, XC10

Nested Class Summary

Nested classes/interfaces inherited from interface
com.ibm.websphere.objectgrid.[ClientReplicableMap](#)

[ClientReplicableMap.Mode](#)

Field Summary

s t a t i	CLIENT
-----------------------	------------------------

c i n t	Constant used to indicate this map is a client to a server map
s t a t i c i n t	DEFAULT_LOCK_TIMEOUT Default lock timeout used if setLockTimeout(int) is not invoked.
s t a t i c i n t	DEFAULT_NUMBER_OF_BUCKETS Deprecated. <i>Deprecated in 8.6. Buckets are no longer required. Use the isNearCacheEnabled() flag to disable the near cache in the ObjectGrid configuration XML file.</i>
s t a t i c i n t	DEFAULT_NUMBER_OF_LOCK_BUCKETS Default number of lock buckets used if setNumberOfLockBuckets(int) is not invoked.
s t a t i c S t r i n g	EVICTIONTRIGGER_MEMORY_USAGE_THRESHOLD The eviction trigger string constant to enable memory based eviction using memory usage threshold provided by the java.lang.management.MemoryPoolMXBean.
s t a t i c i n t	LOCAL Constant used to indicate this map is not a distributed map.
s t a t i c i n t	SERVER Constant used to indicate this map is a server map.

Fields inherited from interface com.ibm.websphere.objectgrid.[ClientReplicableMap](#)
[CONTINUOUS_REPLICATION](#), [NONE](#), [SNAPSHOT_REPLICATION](#)



Method Summary

addMapEventListener(com.ibm.websphere.objectgrid.plugins.EventListener eventListener)
Adds an EventListener to this BackingMap.

addMapEventListener(MapEventListener eventListener)
Deprecated. *This method is deprecated in version 7.1.1, use the [addMapEventListener\(EventListener\)](#) method.*

addMapIndexPlugin(com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index)
Adds an MapIndexPlugin to this Map.

createDynamicIndex(com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index, com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)
Creates a dynamic index on the BackingMap.

createDynamicIndex(String name, boolean isRangeIndex, String attributeName, com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)
Creates a dynamic index on the BackingMap.

getCopyKey()
Gets whether keys are copied for this BackingMap.

getCopyMode()
Gets the CopyMode being used by this BackingMap.

getEntityMetadata()
Retreive the metadata for the entity associated with this backing map.

m
d
.
E
n
t
i
t
y
M
e
t
a
d
a
t
a

S
t
r
i
n
g

[getEvictionTriggers\(\)](#)

Returns the types of additional eviction triggers.

E
v
i
c
t
o
r

[getEvictor\(\)](#)

Gets the Evictor being used by this BackingMap.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
O
u
t
p
u
t
F
o
r
m
a
t

[getKeyOutputFormat\(\)](#)

Retrieves the data format for all data access APIs that return cache keys.

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g
i
n
s
.
L
o
a
d
e
r

[getLoader\(\)](#)

Gets the Loader being used by this BackingMap.

L
o
c
k
S
t
r
a
t
e
g
y

[getLockStrategy\(\)](#)

Gets the LockStrategy object being used by this BackingMap.

i
n
t

[getLockTimeout\(\)](#)

Gets the lock timeout value used by the lock manager for this BackingMap.

L
i
s
t

[getMapEventListeners\(\)](#)

Gets the current list of EventListeners.

L
i
s
t

[getMapIndexPlugins\(\)](#)

Returns the current list of MapIndexPlugin objects for this BackingMap.

S
t
r
i
n
g

[getMapSetName\(\)](#)

Retrieves the name of the MapSet that this BackingMap is currently associated with.

n
g

i
n
t

S
t
r
i
n
g

b
o
o
l
e
a
n

i
n
t

i
n
t

O
b
j
e
c
t
G
r
i
d

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g

[getMapType\(\)](#)
Returns the type of BackingMap.

[getName\(\)](#)
Gets the name of the BackingMap.

[getNullValuesSupported\(\)](#)
Gets whether this BackingMap supports null values or not.

[getNumberOfBuckets\(\)](#)
Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

[getNumberOfLockBuckets\(\)](#)
Gets the number of lock buckets defined for the hash map used by lock manager for this backing map.

[getObjectGrid\(\)](#)
Gets the ObjectGrid that owns this BackingMap.

[getObjectTransformer\(\)](#)
Gets the ObjectTransformer object being used by this BackingMap and/or Loader.

i
n
s
.
O
b
j
e
c
t
T
r
a
n
s
f
o
r
m
e
r

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g
i
n
s
.
O
p
t
i
m
i
s
t
i
c
C
a
l
l
b
a
c
k

[getOptimisticCallback\(\)](#)

Gets the OptimisticCallback being used by this BackingMap and/or Loader or null if the LockStrategy is not optimistic.

i n t	getPartitionId() Gets the partition identifier being used by this BackingMap.
P a r t i t i o n M a n a g e r	getPartitionManager() Allows access to the PartitionManager that is defined for this BackingMap.
b o o l e a n	getPreLoadMode() Returns whether this BackingMap will be asynchronously preloaded or not if a Loader is set.
b o o l e a n	getReadOnly() Retrieves the map type.
c o m . i b m . w e b s p h e r e . o b j e c t g r i d . p l u g i n s .	getSerializerAccessor() Retrieve the SerializerAccessor for this map.

i
o
.
S
e
r
i
a
l
i
z
e
r
A
c
c
e
s
s
o
r

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
p
l
u
g
i
n
s
.
B
a
c
k
i
n
g
M
a
p
L
i
f
e
c
y
c
l

[getState\(\)](#)

Retrieve the current life cycle state of this map.

e
L
i
s
t
e
n
e
r
.
S
t
a
t
e

c
o
m
.
i
b
m
.
w
e
b
s
p
h
e
r
e
.
o
b
j
e
c
t
g
r
i
d
.
T
i
m
e
B
a
s
e
d
D
B
U
p
d
a
t
e
C
o
n
f
i
g

[getTimeBasedDBUpdateConfig\(\)](#)

Get the time-based database update configuration object.

i
n
t

[getTimeToLive\(\)](#)

Gets the number of seconds for an entry to live.

I
I
L

[getTtlEvictorType\(\)](#)

[Type](#)

Gets how expiration time of a BackingMap entry is computed.

com.ibm.websphere.obbj.ctgrid.OutputFormat

[getValueOutputFormat\(\)](#)

Retrieves the data output format for all data access APIs that return cache values.

[String](#)

[getWriteBehind\(\)](#)

Get the write-behind parameter.

boolean

[isNearCacheEnabled\(\)](#)

If true, the client near cache is enabled for supported configurations.

boolean

[isNearCacheInvalidationEnabled\(\)](#)

If true, clients with local caches are automatically invalidated when the data grid map is updated.

boolean

[isNearCacheLastAccessTTLSyncEnabled\(\)](#)

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

a n	
v o i d	removeDynamicIndex (String name) Removes a dynamic index on the BackingMap.
v o i d	removeMapEventListener (com.ibm.websphere.objectgrid.plugins.EventListener eventListener) Removes an EventListener from this BackingMap.
v o i d	removeMapEventListener (MapEventListener eventListener) Provided for compatibility with old releases, use the removeMapEventListener(EventListener) method.
v o i d	setCopyKey (boolean copy) Sets whether or not the key needs to be copied when a map entry is created.
v o i d	setCopyMode (CopyMode mode, Class valueInterface) Sets the CopyMode.
v o i d	setEvictionTriggers (String evictionTriggers) Sets the types of additional eviction triggers, all evictors for the backing map will use the provided set of triggers.
v o i d	setEvictor (Evictor e) Associates an Evictor with this BackingMap.
v o i d	setKeyOutputFormat (com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data output format for all data access APIs that return cache keys.
v o i d	setLoader (com.ibm.websphere.objectgrid.plugins.Loader loader) Associates a Loader with this BackingMap.
v o i d	setLockStrategy (LockStrategy lockStrategy) Sets the LockStrategy.
v o i d	setLockTimeout (int seconds) Sets the lock timeout used by the lock manager for this BackingMap.
v o i d	setMapEventListeners (List eventListenerList) Deprecated. <i>This method is deprecated in version 7.1.1. Use the addMapEventListener(EventListener) or removeMapEventListener(EventListener) methods. Plugins that implement the ObjectGridLifecycleListener interface are automatically registered with the grid. Using this method will remove those automatically added listeners.</i>
v o i d	setMapIndexPlugins (List indexList) Sets the list of MapIndexPlugin objects for this BackingMap.
v	

o i d	<p>setNullValuesSupported(boolean nullValuesSupported) Sets whether this BackingMap supports null values.</p>
v o i d	<p>setNumberOfBuckets(int numBuckets) Deprecated. <i>Deprecated in 8.6. Buckets are no longer required. Use the isNearCacheEnabled() flag to disable the near cache in the ObjectGrid configuration XML file.</i></p>
v o i d	<p>setNumberOfLockBuckets(int numBuckets) Sets the number of lock buckets used by the lock manager for this BackingMap.</p>
v o i d	<p>setObjectTransformer(com.ibm.websphere.objectgrid.plugins.ObjectTransformer t) Sets the ObjectTransformer object for use by this BackingMap and/or Loader.</p>
v o i d	<p>setOptimisticCallback(com.ibm.websphere.objectgrid.plugins.OptimisticCallback checker) Sets the OptimisticCallback.</p>
v o i d	<p>setPreloadMode(boolean async) Sets the preload mode if a Loader is set for this BackingMap.</p>
v o i d	<p>setReadOnly(boolean readOnlyEnabled) Sets the map type of this BackingMap.</p>
v o i d	<p>setTimeBasedDBUpdateConfig(com.ibm.websphere.objectgrid.TimeBasedDBUpdateConfig dbUpdateConfig) Set the time-based database update configuration object.</p>
v o i d	<p>setTimeToLive(int seconds) Sets "time to live" of each map entry in seconds.</p>
v o i d	<p>setTtlEvictorType(TTLType type) Sets how expiration time of a BackingMap entry is computed.</p>
v o i d	<p>setValueOutputFormat(com.ibm.websphere.objectgrid.OutputFormat outputFormat) Sets the data output format for all data access APIs that return cache values.</p>
v o i d	<p>setWriteBehind(String writeBehindParam) Enable write-behind updates for this map.</p>

Methods inherited from interface

com.ibm.websphere.objectgrid.[ClientReplicableMap](#)

[disableClientReplication](#), [enableClientReplication](#), [getReplicationMode](#)

Field Detail

DEFAULT_LOCK_TIMEOUT

static final int **DEFAULT_LOCK_TIMEOUT**

Default lock timeout used if `setLockTimeout(int)` is not invoked.

See Also:

[Constant Field Values](#)

DEFAULT_NUMBER_OF_BUCKETS

static final int **DEFAULT_NUMBER_OF_BUCKETS**

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Default number of lock buckets used if `setNumberOfBuckets(int)` is not invoked.

See Also:

[Constant Field Values](#)

DEFAULT_NUMBER_OF_LOCK_BUCKETS

static final int **DEFAULT_NUMBER_OF_LOCK_BUCKETS**

Default number of lock buckets used if `setNumberOfLockBuckets(int)` is not invoked.

See Also:

[Constant Field Values](#)

EVICTIONTRIGGER_MEMORY_USAGE_THRESHOLD

static final [String](#) **EVICTIONTRIGGER_MEMORY_USAGE_THRESHOLD**

The eviction trigger string constant to enable memory based eviction using memory usage threshold provided by the `java.lang.management.MemoryPoolMXBean`.

Since:

WAS XD 6.1.0.3

See Also:

[setEvictionTriggers\(String\)](#), [Constant Field Values](#)

LOCAL

static final int **LOCAL**

Constant used to indicate this map is not a distributed map.

Since:

WAS XD 6.1

See Also:

[getMapType\(\)](#), [Constant Field Values](#)

SERVER

static final int **SERVER**

Constant used to indicate this map is a server map.

Since:

WAS XD 6.1

See Also:

[getMapType\(\)](#), [Constant Field Values](#)

CLIENT

static final int **CLIENT**

Constant used to indicate this map is a client to a server map

Since:

WAS XD 6.1

See Also:

[getMapType\(\)](#), [Constant Field Values](#)

Method Detail

getName

[String](#) getName()

Gets the name of the BackingMap.

Returns:

value specified when BackingMap was created.

setEvictor

void setEvictor([Evictor](#) e)

Associates an Evictor with this BackingMap.

An Evictor aids with cleaning up the cache based on whatever algorithm is desired (LRU, LFU, etc). Passing null to this method removes a previously set Evictor object from an earlier invocation of this method.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

An Evictor that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous evictor which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

An Evictor may also implement the `BackingMapPlugin` interface in order to receive enhanced `BackingMap` plug-in lifecycle method calls. The plug-in is then also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

e - Evictor instance

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[Evictor](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getEvictor

[Evictor](#) `getEvictor()`

Gets the Evictor being used by this BackingMap.

Returns:

the argument that was passed to the `setEvictor(Evictor)` method of this interface or `null` if `setEvictor` was not previously called for this BackingMap object.

See Also:

[Evictor](#), [setEvictor\(Evictor\)](#)

setObjectTransformer

`void setObjectTransformer(com.ibm.websphere.objectgrid.plugins.ObjectTransformer t)`

Sets the ObjectTransformer object for use by this BackingMap and/or Loader.

An ObjectTransformer aids with the "serialization" of non-Serializable objects. It allows a custom copy function to be installed for more efficient object copy operations.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

An ObjectTransformer that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous transformer which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

t - ObjectTransformer instance

Throws:

[IllegalArgumentException](#) - if the passed in ObjectTransformer is null

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

ObjectTransformer, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getObjectTransformer

`com.ibm.websphere.objectgrid.plugins.ObjectTransformer getObjectTransformer()`

Gets the ObjectTransformer object being used by this BackingMap and/or Loader.

Returns:

the argument that was passed to the `setObjectTransformer(ObjectTransformer)` method of this interface or the default ObjectTransformer object if the `setObjectTransformer` method was not previously called for this object.

See Also:

ObjectTransformer, [setObjectTransformer\(ObjectTransformer\)](#)

setOptimisticCallback

```
void setOptimisticCallback(com.ibm.websphere.objectgrid.plugins.OptimisticCallback checker)
```

Sets the OptimisticCallback.

The OptimisticCallback will be used to check the versions of cache entries during the commit phase. If no OptimisticCallback was previously set, a default OptimisticCallback will be used. For Entities, the default OptimisticCallback will use a version field that was specified in the entity metadata. For POJO objects or Entities that do not have a version field specified, the default OptimisticCallback uses the entire object as the version value. In order for it to work for POJO objects, the application's value object needs to have a useful equals(Object) method. If your application does not require versioning, but is using Optimistic locking, the NoVersioningOptimistCallback should be used.

Note, to avoid an IllegalStateException, this method must be called prior to the ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

An OptimisticCallback that implements the BackingMapLifecycleListener is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous optimistic callback which implements BackingMapLifecycleListener is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

checker - OptimisticCallback instance

Throws:

[IllegalArgumentException](#) - if the passed in OptimisticCallback is null

[IllegalStateException](#) - if this method is called after the ObjectGrid.initialize() method is called.

See Also:

OptimisticCallback, NoVersioningOptimisticCallback, [LockStrategy.OPTIMISTIC](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getOptimisticCallback

```
com.ibm.websphere.objectgrid.plugins.OptimisticCallback getOptimisticCallback()
```

Gets the OptimisticCallback being used by this BackingMap and/or Loader or null if the LockStrategy is not optimistic.

If no OptimisticCallback was previously set, a default OptimisticCallback will be used. For Entities, the default OptimisticCallback will use a version field that was specified in the entity metadata. For POJO objects or Entities that do not have a version field specified, the default OptimisticCallback uses the entire object as the version value. In order for it to work for POJO objects, the application's value object needs to have a useful equals(Object) method. If your application does not require versioning, but is using Optimistic locking, the NoVersioningOptimistCallback should be used.

Returns:

the argument that was passed to the setOptimisticCallback(OptimisticCallback) method of this interface or the default OptimisticCallback object if the setOptimisticCallback method was not previously called for this object. If Optimistic locking is not being used, this method will return null after ObjectGrid.initialize() has been invoked.

See Also:

NoVersioningOptimisticCallback, OptimisticCallback, [LockStrategy.OPTIMISTIC](#), [setOptimisticCallback\(OptimisticCallback\)](#)

setLoader

```
void setLoader(com.ibm.websphere.objectgrid.plugins.Loader loader)
```


Associates a Loader with this BackingMap.

Only one Loader can be associated with a given BackingMap. Passing null to this method removes a previously set Loader object from an earlier invocation of this method and indicates that this BackingMap is not associated with a Loader.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

A loader that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous loader which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

A Loader may also implement the `BackingMapPlugin` interface in order to receive enhanced `BackingMap` plug-in lifecycle method calls. The plug-in is then also required to correctly implement each of the bean methods related to introspection of its state (for example `isInitialized()`, `isDestroyed()`, etc).

Parameters:

loader - Loader instance

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

Loader, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getLoader

`com.ibm.websphere.objectgrid.plugins.Loader getLoader()`

Gets the Loader being used by this BackingMap.

Returns:

the argument that was passed to the `setLoader(Loader)` method of this interface or null if `setLoader` was not previously called for this object.

See Also:

Loader, [setLoader\(Loader\)](#)

setPreloadMode

`void setPreloadMode(boolean async)`

Sets the preload mode if a Loader is set for this BackingMap.

If the parameter is true then the `Loader.preloadMap(Session, BackingMap)` is invoked asynchronously; otherwise it blocks the execution when loading data so the cache is unavailable until preload completes. Preloading occurs during `ObjectGrid` initialization.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

async - If this is true then the cache is loaded asynchronously otherwise it blocks and the cache is unavailable until preload completes.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()`

method is called.

See Also:

`Loader.preloadMap(Session, BackingMap)`

getPreLoadMode

boolean `getPreLoadMode()`

Returns whether this `BackingMap` will be asynchronously preloaded or not if a `Loader` is set.

If `true` is returned then the `Loader.preloadMap(Session, BackingMap)` method is invoked asynchronously; otherwise it blocks the execution when loading data so the cache is unavailable until preload completes. Preloading occurs during `ObjectGrid` initialization.

Returns:

the argument that was passed to the `setPreloadMode(boolean)` method of this interface or `false` if `setPreloadMode` was not previously called for this object.

See Also:

`Loader.preloadMap(Session, BackingMap)`, [setPreloadMode\(boolean\)](#)

addMapIndexPlugin

void `addMapIndexPlugin`(`com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index`)
throws `com.ibm.websphere.objectgrid.IndexAlreadyDefinedException`

Adds an `MapIndexPlugin` to this `Map`. This method assumes the index implementation was constructed with the name of the attribute to index. The name of the index is specified when the index is constructed.

Note, to avoid an `IllegalStateException`, this method must be called prior to `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

A `MapIndexPlugin` that implements the `BackingMapLifecycleListener` is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous index which implements `BackingMapLifecycleListener` is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

`index` - The index implementation.

Throws:

`IndexAlreadyDefinedException` - if this index already exists.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

`MapIndexPlugin`, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getMapIndexPlugins

[List](#) `getMapIndexPlugins()`

Returns the current list of `MapIndexPlugin` objects for this `BackingMap`.

Returns:

The current list of `MapIndexPlugins` for this `BackingMap`. The list is empty if the `addMapIndexPlugin(MapIndexPlugin)` OR `setMapIndexPlugins(List)` method was not previously called for this `BackingMap`.

See Also:

[addMapIndexPlugin\(MapIndexPlugin\)](#), [setMapIndexPlugins\(List\)](#)

setMapIndexPlugins

void **setMapIndexPlugins**([List](#) indexList)

Sets the list of MapIndexPlugin objects for this BackingMap. If the BackingMap already has a List of MapIndexPlugin objects, that list is replaced by the List passed as an argument to the current invocation of this method.

Note, to avoid an `IllegalStateException`, this method must be called prior to `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

indexList - A non-null reference to a List of MapIndexPlugin objects.

Throws:

[IllegalArgumentException](#) - is thrown if indexList is null or the indexList contains either a null reference or an object that is not an instance of MapIndexPlugin.

See Also:

MapIndexPlugin, [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

setCopyMode

void **setCopyMode**([CopyMode](#) mode,
[Class](#) valueInterface)

Sets the CopyMode.

The CopyMode determines whether a get operation of an entry in the BackingMap returns the actual value, a copy of the value, or a proxy for the value. In the case of a proxy, the copy of the value does not occur unless a set method of the application provided value interface is invoked. It also determines that when a transaction is committed, whether a copy of the value object of an entry that was marked as dirty by the transaction is put into the BackingMap at commit time. The CopyMode does not specify if the object is copied when being read or written to a Loader. It is the responsibility of the implementor of a Loader to make copies as appropriate. The default CopyMode is `CopyMode.COPY_ON_READ_AND_COMMIT`.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

mode - must be one of the final static variables defined in CopyMode. See CopyMode class for an explanation of each mode and how the valueInterface is used for `CopyMode.COPY_ON_WRITE`.

valueInterface - the value interface Class object. Specify null in version 7.1 and later.

Throws:

[IllegalArgumentException](#) - if mode is `CopyMode.COPY_ON_WRITE` and valueInterface parameter is null and CGLIB isn't in the classpath.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[CopyMode](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getCopyMode

[CopyMode](#) `getCopyMode()`

Gets the CopyMode being used by this BackingMap.

Returns:

the argument that was passed to the `setCopyMode(CopyMode, Class)` method of this interface or the default CopyMode object if `setCopyMode` was not previously called for this object.

See Also:

[CopyMode](#), [setCopyMode\(CopyMode, Class\)](#)

setLockStrategy

`void setLockStrategy(LockStrategy lockStrategy)`

Sets the LockStrategy.

The locking strategy represented by the LockStrategy object determines if the internal ObjectGrid lock manager is used whenever a map entry is accessed by a transaction. The default strategy is `LockStrategy.OPTIMISTIC`.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`lockStrategy` - must be one of the final static variables defined in `LockStrategy`. See `LockStrategy` class for an explanation of each locking strategy.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[LockStrategy](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getLockStrategy

[LockStrategy](#) `getLockStrategy()`

Gets the LockStrategy object being used by this BackingMap.

Returns:

the argument that was passed to the `setLockStrategy(LockStrategy)` method of this interface or the default LockStrategy object if `setLockStrategy` was not previously called for this object.

See Also:

[LockStrategy](#), [setLockStrategy\(LockStrategy\)](#)

setMapEventListeners

[@Deprecated](#)

`void setMapEventListeners(List eventListenerList)`

Deprecated. *This method is deprecated in version 7.1.1. Use the [addMapEventListener\(EventListener\)](#) or [removeMapEventListener\(EventListener\)](#) methods. Plugins that implement the `ObjectGridLifecycleListener` interface are automatically registered with the grid. Using this method will remove those automatically added listeners.*

Sets the list of EventListener objects.

If this BackingMap already has a List of EventListeners, that list is replaced by the List passed as an argument to the current invocation of this method. This method can be called before and after the ObjectGrid.initialize() method.

Parameters:

eventListenerList - A non-null reference to a List of EventListener objects that are instances of BackingMapLifecycleListener OR MapEventListener

Throws:

[IllegalArgumentException](#) - is thrown if eventListenerList is null, the eventListenerList contains either a null reference or an object that is not an instance of BackingMapLifecycleListener OR MapEventListener

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener, [addMapEventListener\(EventListener\)](#), [removeMapEventListener\(EventListener\)](#)

getMapEventListeners

[List](#) getMapEventListeners()

Gets the current list of EventListeners.

Returns:

the current list of EventListener objects for this BackingMap.

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener

addMapEventListener

void addMapEventListener(com.ibm.websphere.objectgrid.plugins.EventListener eventListener)

Adds an EventListener to this BackingMap.

Note, this method is allowed to be invoked before and after the ObjectGrid.initialize() method. Backing map plug-ins (Loader, Evictor, MapIndexPlugin, ObjectTransformer, OptimisticCallback) that implement the ObjectGridLifecycleListener are automatically added as listeners when added to the BackingMap.

Parameters:

eventListener - A non-null reference to a EventListener to add to the list. The listener must be an instance of BackingMapLifecycleListener OR MapEventListener

Throws:

[IllegalArgumentException](#) - if eventListener is null or not an instance of BackingMapLifecycleListener OR MapEventListener

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener

addMapEventListner

void addMapEventListner([MapEventListener](#) eventListener)

Deprecated. *This method is deprecated in version 7.1.1, use the [addMapEventListener\(EventListener\)](#) method.*

Provided for compatibility with old releases, use the [addMapEventListener\(EventListener\)](#) method.

Parameters:

eventListener - A non-null reference to a EventListener to add to the list. The listener must be an instance of BackingMapLifecycleListener OR MapEventListener

removeMapEventListener

void **removeMapEventListener**(com.ibm.websphere.objectgrid.plugins.EventListener eventListener)

Removes an EventListener from this BackingMap.

Note, this method is allowed to be invoked before and after the ObjectGrid.initialize() method. Backing map plug-ins (Loader, Evictor, MapIndexPlugin, ObjectTransformer, OptimisticCallback) that implement the ObjectGridLifecycleListener are automatically removed as listeners when removed from the ObjectGrid.

Parameters:

eventListener - A non-null reference to an event listener that was previously added by invoking either the addMapEventListener(EventListener) OR setMapEventListeners(List) method of this interface.

Throws:

[IllegalArgumentException](#) - if eventListener is null or not an instance of BackingMapLifecycleListener OR MapEventListener

See Also:

EventListener, [MapEventListener](#), BackingMapLifecycleListener, [addMapEventListener\(EventListener\)](#)

removeMapEventListener

void **removeMapEventListener**([MapEventListener](#) eventListener)

Provided for compatibility with old releases, use the [removeMapEventListener\(EventListener\)](#) method.

Parameters:

eventListener - A non-null reference to an event listener that was previously added by invoking either the addMapEventListener(EventListener) OR setMapEventListeners(List) method of this interface.

getPartitionId

int **getPartitionId**()

Gets the partition identifier being used by this BackingMap.

Returns:

The 0-based index for the partition represented by this BackingMap instance. If there is only a single partition defined for this BackingMap object, a 0 will be returned (default).

Since:

WAS XD 6.0.1

setReadOnly

void **setReadOnly**(boolean readOnlyEnabled)

Sets the map type of this BackingMap.

A map can be a read only map or a read/write map. Passing true as the parameter value will make this map a read only map; passing false as the parameter value will make this map a read/write map.

Note, to avoid an IllegalStateException, this method must be called prior to the

ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

Parameters:

readOnlyEnabled - If set to true, this BackingMap will be a read only map. If false, the map will be a read/write map.

Throws:

[IllegalStateException](#) - if this method is called after the ObjectGrid.initialize() method is called.

getReadOnly

boolean **getReadOnly()**

Retrieves the map type.

Returns:

the argument that was passed to setReadOnly(boolean) method of this interface. True is returned if this a read only map. A return value of false implies that this is a read/write map. If setReadOnly was never called, the default return value is false.

See Also:

[setReadOnly\(boolean\)](#)

getObjectGrid

[ObjectGrid](#) **getObjectGrid()**

Gets the ObjectGrid that owns this BackingMap.

Returns:

the ObjectGrid instance that owns this BackingMap.

See Also:

[ObjectGrid](#)

setNumberOfBuckets

void **setNumberOfBuckets**(int numBuckets)

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Sets the number of buckets used by this BackingMap.

The BackingMap implementation uses a hash map for its implementation. If there are a lot of entries in the BackingMap then more buckets means better performance because the risk of collisions is lower as the number of buckets grows. More buckets also means more concurrency. If number of buckets is 0, no entries will be stored in the map, but the appropriate ObjectGrid and BackingMap plug-ins will still be called.

Once the ObjectGrid is initialized this parameter cannot be changed. Therefore, to avoid an IllegalStateException, this method must be called prior to the ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

Parameters:

numBuckets - The number of buckets to use.

Throws:

[IllegalArgumentException](#) - if numBuckets is less than 0.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getNumberOfBuckets

```
int getNumberOfBuckets()
```

Deprecated. *Deprecated in 8.6. Buckets are no longer required. Use the [isNearCacheEnabled\(\)](#) flag to disable the near cache in the ObjectGrid configuration XML file.*

Gets the number of buckets defined for this BackingMap.

Returns:

the same value passed to the `setNumberOfBuckets(int)` method or `DEFAULT_NUMBER_OF_BUCKETS` if `setNumberOfBuckets` was never called.

See Also:

[setNumberOfBuckets\(int\)](#), [DEFAULT_NUMBER_OF_BUCKETS](#)

setNumberOfLockBuckets

```
void setNumberOfLockBuckets(int numBuckets)
```

Sets the number of lock buckets used by the lock manager for this BackingMap.

When `LockStrategy.OPTIMISTIC` or `LockStrategy.PESSIMISTIC` is used for this BackingMap, a lock manager is created for the BackingMap. The lock manager uses a hash map to keep track of entries that are locked by 1 or more transactions. If there are a lot of entries in the hash map, then more lock buckets means better performance as the risk of collisions is lower as the number of buckets grows. More lock buckets also means more concurrency. When the lock strategy is `LockStrategy.NONE`, no lock manager is used by this BackingMap. In this case, a call to this method does nothing.

Once the `ObjectGrid` is initialized, the number of lock buckets cannot be changed. Therefore, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`numBuckets` - The number of lock buckets to use.

Throws:

[IllegalArgumentException](#) - if numBuckets is less than 1.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[LockStrategy](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getNumberOfLockBuckets

```
int getNumberOfLockBuckets()
```

Gets the number of lock buckets defined for the hash map used by lock manager for this backing map.

Returns:

the same value passed to the `setNumberOfLockBuckets(int)` method or `DEFAULT_NUMBER_OF_LOCK_BUCKETS` if `setNumberOfLockBuckets` was never called.

See Also:

[setNumberOfLockBuckets\(int\)](#), [DEFAULT_NUMBER_OF_LOCK_BUCKETS](#)

setLockTimeout

void **setLockTimeout**(int seconds)

Sets the lock timeout used by the lock manager for this `BackingMap`.

When `LockStrategy.OPTIMISTIC` or `LockStrategy.PESSIMISTIC` is used for this `BackingMap`, a lock manager is created for the `BackingMap`. To prevent deadlocks from occurring, the lock manager has a default timeout value for waiting for a lock to be granted. If this timeout limit is exceeded, a `LockTimeoutException` is thrown. The default value of `DEFAULT_LOCK_TIMEOUT` should be sufficient for most applications, but on a heavily loaded system, a timeout may occur when no deadlock exists. In that case, this method can be used to increase the lock timeout value from the default to whatever is desired to prevent false timeout exceptions from occurring. When the lock strategy is `LockStrategy.NONE`, no lock manager is used by this `BackingMap`. In this case, a call to this method does nothing. A lock timeout value of zero indicates to not wait for the lock if it is not immediately available.

Once the lock manager is initialized, the lock timeout value cannot be changed. Therefore, to avoid an `IllegalStateException`, this method must be called prior to `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application. When an entry is fetched the lock timeout can be changed for a given transaction using `ObjectMap.setLockTimeout(int)`

Parameters:

seconds - is the lock timeout value to use in seconds.

Throws:

[IllegalArgumentException](#) - if seconds is less than 0.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[DEFAULT_LOCK_TIMEOUT](#), [LockStrategy](#), [LockTimeoutException](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#), [ObjectMap.setLockTimeout\(int\)](#)

getLockTimeout

int **getLockTimeout**()

Gets the lock timeout value used by the lock manager for this `BackingMap`.

Returns:

the same value passed to the `setLockTimeout(int)` method or `DEFAULT_LOCK_TIMEOUT` if `setLockTimeout` was never called.

See Also:

[DEFAULT_LOCK_TIMEOUT](#), [setLockTimeout\(int\)](#)

setNullValuesSupported

void **setNullValuesSupported**(boolean nullValuesSupported)

Sets whether this `BackingMap` supports null values.

If null values are supported, users need to be careful when a get operation returns a null

reference. It could be due to the fact that the key is not found in the BackingMap, or that the value in the BackingMap is null. To determine if a key was not found, or the value is null, the `containsKey` method can be used.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`nullValuesSupported` - If set to true, null values are supported; otherwise null values are not supported.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#), [ObjectMap.containsKey\(Object\)](#)

getNullValuesSupported

boolean `getNullValuesSupported()`

Gets whether this BackingMap supports null values or not.

Returns:

the same value passed to the `setNullValuesSupported(boolean)` method or the default value of true if `setNullValuesSupported` was never called.

See Also:

[setNullValuesSupported\(boolean\)](#)

setCopyKey

void `setCopyKey(boolean copy)`

Sets whether or not the key needs to be copied when a map entry is created.

Copying the key object allows the application to use the same key object for each `ObjectMap` operation. The application changes the key object state prior to each `ObjectMap` operation so that it can work with different entries using the same key object. If a separate key object is used for each entry, then there is no reason to copy the key object. This attribute allows an application to make the tradeoff of copying key object versus using more memory as a result of separate key object used by the application for each entry. If this method is not called, then the default of false is used (e.g. the key is NOT copied).

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`copy` - If true is specified, then this BackingMap uses the `ObjectTransformer.copyKey(Object)` method to copy the key object when necessary.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#), `ObjectTransformer.copyKey(Object)`

getCopyKey

boolean `getCopyKey()`

Gets whether keys are copied for this BackingMap.

Returns:

the same value passed to the `setCopyKey(boolean)` method or the default value of false if `setCopyKey` was never called.

See Also:

[setCopyKey\(boolean\)](#)

setTimeToLive

void `setTimeToLive(int seconds)`

Sets "time to live" of each map entry in seconds.

If this method is not called, the lifetime of an entry is forever (or until the application explicitly removes or invalidates the entry, or a user defined Evictor evicts the entry). Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`seconds` - the number of seconds a map entry is allowed to live in map before being evicted.

Throws:

[IllegalArgumentException](#) - if `seconds` is less than 0.

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[setTtlEvictorType\(TTLType\)](#), [ObjectMap.setTimeToLive\(int\)](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getTimeToLive

int `getTimeToLive()`

Gets the number of seconds for an entry to live.

This value returned is in seconds and 0 indicates forever.

Returns:

the same value passed to the `setTimeToLive(int)` method or 0 if `setLockTimeout` was never called.

See Also:

[setTimeToLive\(int\)](#)

setTtlEvictorType

void `setTtlEvictorType(TTLType type)`

Sets how expiration time of a BackingMap entry is computed.

If this method is not called, `TTLType.NONE` is used to indicate the map entry has no expiration time (e.g. is allowed to live until explicitly removed or invalidated by the application, or evicted by a user defined Evictor).

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

type - must be one of the public constants declared in the `TTLType` class.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

See Also:

[TTLType](#), [ObjectGrid.initialize\(\)](#), [ObjectGrid.getSession\(\)](#)

getTtlEvictorType

[TTLType](#) `getTtlEvictorType()`

Gets how expiration time of a `BackingMap` entry is computed.

Returns:

the `TTLType` that was passed to the `setTtlEvictorType(TTLType)` or `TTLType.NONE` if `setTtlEvictorType` was never called.

See Also:

[setTtlEvictorType\(TTLType\)](#), [TTLType](#)

createDynamicIndex

```
void createDynamicIndex(String name,  
                        boolean isRangeIndex,  
                        String attributeName,  
                        com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)  
                        throws com.ibm.websphere.objectgrid.IndexAlreadyDefinedException,  
                               IllegalArgumentException
```

Creates a dynamic index on the `BackingMap`.

Required Client Permission: `ServerMapPermission.REPLICATE`

Parameters:

name - the name of the index. The name can not be null or a zero length string.

isRangeIndex - Indicate whether to create a `MapRangeIndex` or a `MapIndex`. If set to true, the index will be a type of `MapRangeIndex`.

attributeName - The name of the attribute to be indexed. The attributeName can not be null or a zero length string.

dynamicIndexCallback - The callback that will invoke upon dynamic index events. The `dynamicIndexCallback` is optional and can be null.

Throws:

[IllegalArgumentException](#) - if name or attributeName is null or a zero length string.

`IndexAlreadyDefinedException` - if a `MapIndexPlugin` with the specified name already exists.

Since:

WAS XD 6.0.1

See Also:

`MapIndex`, `MapIndexPlugin`, `MapRangeIndex`, [ObjectMap.getIndex\(String\)](#)

createDynamicIndex

```
void createDynamicIndex(com.ibm.websphere.objectgrid.plugins.index.MapIndexPlugin index,
```

```
com.ibm.websphere.objectgrid.plugins.index.DynamicIndexCallback dynamicIndexCallback)
throws com.ibm.websphere.objectgrid.IndexAlreadyDefinedException,
       IllegalArgumentException
```

Creates a dynamic index on the BackingMap.

Required Client Permission: ServerMapPermission.DYNAMIC_INDEX

A MapIndexPlugin that implements the BackingMapLifecycleListener is automatically added as if the [addMapEventListener\(EventListener\)](#) method was called. Any previous index which implements BackingMapLifecycleListener is removed as if the [removeMapEventListener\(EventListener\)](#) method was called.

Parameters:

index - The index implementation. The index can not be null.

dynamicIndexCallback - The callback that will invoke upon dynamic index events. The dynamicIndexCallback is optional and can be null.

Throws:

[IllegalArgumentException](#) - if index is null or index.getName() returns null or a zero length string.

IndexAlreadyDefinedException - if a MapIndexPlugin with the specified name already exists.

Since:

WAS XD 6.0.1

See Also:

MapIndexPlugin, [ObjectMap.getIndex\(String\)](#)

removeDynamicIndex

```
void removeDynamicIndex(String name)
throws com.ibm.websphere.objectgrid.IndexUndefinedException,
       IllegalArgumentException
```

Removes a dynamic index on the BackingMap.

Required Client Permission: ServerMapPermission.DYNAMIC_INDEX

Parameters:

name - the name of the index. The name can not be null.

Throws:

[IllegalArgumentException](#) - if name is null.

IndexUndefinedException - if a MapIndexPlugin with the specified name does not exist.

Since:

WAS XD 6.0.1

See Also:

[createDynamicIndex\(MapIndexPlugin, DynamicIndexCallback\)](#), [createDynamicIndex\(String, boolean, String, DynamicIndexCallback\)](#)

getPartitionManager

```
PartitionManager getPartitionManager()
```

Allows access to the PartitionManager that is defined for this BackingMap. This access may be useful for Loaders during Loader.preloadMap(Session, BackingMap) processing (to properly partition the data to be loaded).

Returns:

PartitionManager associated with this BackingMap.

Since:

WAS XD 6.0.1

See Also:

getEntityMetadata

com.ibm.websphere.projector.md.EntityMetadata **getEntityMetadata()**

Retrieve the metadata for the entity associated with this backing map.

Returns:

the EntityMetadata if an entity is associated with this backing map or null if there is no entity associated with this backing map.

Since:

WAS XD 6.1

setWriteBehind

void **setWriteBehind**([String](#) writeBehindParam)

Enable write-behind updates for this map.

If a map is configured with write-behind loader update, the updates (could be insert type, remove type, or update type) to the backend are not instantly updated to the back end by calling the Loader.batchUpdate(TxID, LogSequence) method. Instead, they are queued in a write-behind queue map and updated to the back end periodically.

A write-behind update is pushed to the backend periodically within a different transaction from the one the update is made to ObjectGrid. When the write-behind update to the backend fails, for example, due to data integrity problem, it is too late to roll back the original ObjectGrid transaction. ObjectGrid will invalidate the entry and create an entry in a failed database update map. The name of this failed database update map is WriteBehindLoaderConstants.WRITE_BEHIND_FAILED_UPDATES_MAP_PREFIX+baseMapName. The key of the entry in this map is an auto-increment Integer, and the value is a [LogElement](#). The logElement can be used to compensate the failure.

Depending on your grid use case and your back end loader configuration, your back end loader or the back end database might benefit from having upsert operations instead of insert and update operations in the LogElements that it receives for a transaction. Use the ConvertToUpsert=true configuration option on the writeBehindParam to have the write behind loader convert insert and update LogElement operations to upsert LogElement operations when they are passed to the back end loader. Not all back end loaders may support the upsert operation, be certain that the back end loader supports upsert operations before using the ConvertToUpsert=true clause in the writeBehindParam. The default value is ConvertToUpsert=false.

Note, to avoid an IllegalStateException, this method must be called prior to the ObjectGrid.initialize() method. Also, keep in mind that the ObjectGrid.getSession() method implicitly calls the ObjectGrid.initialize() method if it has yet to be called by the application.

Parameters:

writeBehindParam - a write-behind parameter consisting of a maximum update time and/or a maximum key update count. The format of the write-behind parameter is "T[time];C[count][;ConvertToUpsert=true]", for example, "T100;C2000". "T100;C2000" means the loader will write to the back end when there are 2000 pending keys to be updated or when 100 seconds have passed since the last update. The default update time is 300 seconds and the default update key count is 1000. You can configure the update time only, the update key count only, or an empty string. The default value(s) will then be used in either of the above three cases. The default value is null to disable write-behind updates.

Throws:

[IllegalArgumentException](#) - if the write behind parameters are unknown or improperly formatted.

[IllegalStateException](#) - if this method is called after the ObjectGrid.initialize() method is called.

Since:

WAS XD 6.1.0.3

See Also:

[WriteBehindLoaderConstants](#)

getWriteBehind

[String](#) `getWriteBehind()`

Get the write-behind parameter. A write-behind parameter consists of a maximum update time and/or a maximum key update count. The format of the write-behind parameter is "T[time];C[count][;ConvertToUpsert=true]".

Returns:

the write-behind parameter. If the write-behind parameter is not set, `null` will be returned.

Since:

WAS XD 6.1.0.3

See Also:

[setWriteBehind\(String\)](#)

setTimeBasedDBUpdateConfig

`void setTimeBasedDBUpdateConfig(com.ibm.websphere.objectgrid.TimeBasedDBUpdateConfig dbUpdateConfig)`

Set the time-based database update configuration object.

When a time-based database update configuration object is set, a thread will be started automatically to update or invalidate the ObjectGrid maps with the latest updates (inserts and updates) from the database.

For a local ObjectGrid map, the thread will be launched in the same JVM. For a distributed ObjectGrid map in an ObjectGrid container, the thread will be automatically launched in partition 0. No database update thread will be started in a client side near cache.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

`dbUpdateConfig` - the time-based database update configuration object or `null`.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

Since:

WAS XD 6.1.0.3

getTimeBasedDBUpdateConfig

`com.ibm.websphere.objectgrid.TimeBasedDBUpdateConfig getTimeBasedDBUpdateConfig()`

Get the time-based database update configuration object.

Returns:

the time-based database update configuration object or `null` if not set.

Since:

WAS XD 6.1.0.3

See Also:

[setTimeBasedDBUpdateConfig\(TimeBasedDBUpdateConfig\)](#)

getMapType

int **getMapType**()

Returns the type of BackingMap.

The return value is equivalent to one of the constants declared on this interface, [LOCAL](#), [SERVER](#), or [CLIENT](#).

Returns:

the map type

Since:

WAS XD 6.1

getEvictionTriggers

[String](#) **getEvictionTriggers**()

Returns the types of additional eviction triggers.

The available eviction trigger strings are as described in the String constants in this interface that begin with the name: EVICTIONTRIGGER.

Returns:

a semicolon separated list of eviction triggers

Since:

WAS XD 6.1.0.3

setEvictionTriggers

void **setEvictionTriggers**([String](#) evictionTriggers)

Sets the types of additional eviction triggers, all evictors for the backing map will use the provided set of triggers.

The available eviction trigger strings are as described in the String constants in this interface that begin with the name: EVICTIONTRIGGER.

Note, to avoid an `IllegalStateException`, this method must be called prior to the `ObjectGrid.initialize()` method. Also, keep in mind that the `ObjectGrid.getSession()` method implicitly calls the `ObjectGrid.initialize()` method if it has yet to be called by the application.

Parameters:

evictionTriggers - a semicolon separated list of eviction triggers

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

[IllegalArgumentException](#) - if the eviction triggers are unknown or improperly formatted.

Since:

WAS XD 6.1.0.3

getMapSetName

[String](#) `getMapSetName()`

Retrieves the name of the MapSet that this BackingMap is currently associated with. A "null" return value indicates it is currently not associated with a MapSet. This method will only return a non null value for a client or server map.

Returns:

name of associated MapSet

Since:

7.1

See Also:

[getMapType\(\)](#)

getSerializerAccessor

`com.ibm.websphere.objectgrid.plugins.io.SerializerAccessor` **getSerializerAccessor()**

Retrieve the SerializerAccessor for this map.

Returns:

the SerializerAccessor

Since:

7.1.1

getState

`com.ibm.websphere.objectgrid.plugins.BackingMapLifecycleListener.State` **getState()**

Retrieve the current life cycle state of this map.

Returns:

the current state.

Since:

7.1.1

isNearCacheInvalidationEnabled

`boolean` **isNearCacheInvalidationEnabled()**

If true, clients with local caches are automatically invalidated when the data grid map is updated.

Returns:

true if client near cache invalidation is enabled.

Since:

8.6, XC10 2.5

isNearCacheLastAccessTTLSyncEnabled

`boolean` **isNearCacheLastAccessTTLSyncEnabled()**

If true, clients automatically send time-to-live access information to the remote data grid when accessed and the [TTLType.LAST_ACCESS_TIME](#) TTL evictor type is configured.

Returns:

true if last-access time-to-live information is sent to the remote data grid.

Since:

8.6, XC10 2.5

isNearCacheEnabled

boolean **isNearCacheEnabled()**

If true, the client near cache is enabled for supported configurations. The client near cache can only be enabled when using optimistic locking or when locking is disabled.

Returns:

true if the client near cache is enabled.

getKeyOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getKeyOutputFormat()**

Retrieves the data format for all data access APIs that return cache keys.

This value does not reflect the data output format that plug-ins will see. See the `PluginOutputFormat` annotation for details on how to influence the data object format that plug-ins receive.

Returns:

the data output format.

Since:

8.6, XC10 2.5

setKeyOutputFormat

void **setKeyOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)

Sets the data output format for all data access APIs that return cache keys.

When set to `OutputFormat.UNDEFINED`, the key output format defaults to `OutputFormat.RAW` when using a custom `KeyDataSerializer` plug-in. The key output format is `OutputFormat.NATIVE` in all other cases.

Parameters:

outputFormat - the data output format to use or `OutputFormat.UNDEFINED` to use the default.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

Since:

8.6, XC10 2.5

getValueOutputFormat

com.ibm.websphere.objectgrid.OutputFormat **getValueOutputFormat()**

Retrieves the data output format for all data access APIs that return cache values.

This value does not reflect the data output format that plug-ins will see. See the `PluginOutputFormat` annotation for details on how to influence the data object format that plug-ins receive.

Returns:

the data output format or `OutputFormat.UNDEFINED` if the default should be used.

Since:

8.6, XC10 2.5

setValueOutputFormat

void **setValueOutputFormat**(com.ibm.websphere.objectgrid.OutputFormat outputFormat)

Sets the data output format for all data access APIs that return cache values.

When set to `OutputFormat.UNDEFINED`, the value output format defaults to `OutputFormat.RAW` when using a custom `ValueDataSerializer` plug-in or when the [CopyMode.COPY_TO_BYTES_RAW](#) `CopyMode` set.

The value output format is `OutputFormat.NATIVE` in all other cases.

Parameters:

`outputFormat` - the data output format to use or `OutputFormat.UNDEFINED` to use the default.

Throws:

[IllegalStateException](#) - if this method is called after the `ObjectGrid.initialize()` method is called.

Since:

8.6, XC10 2.5

Overview	Package	Classes	Tree	Serialized	Deprecated	Index	Help	IBM WebSphere® DataPower® XC10 Appliance Release 2.5 Client API Specification
PREV CLASS	NEXT CLASS	FRAMES	NO FRAMES	All				
			Classes					
SUMMARY: NESTED FIELD CONSTR METH			DETAIL: FIELD CONSTR METHOD					

com.ibm.websphere.objectgrid

Class AvailabilityState

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#)

```
public final class AvailabilityState
extends Object
implements Serializable
```

Each shard in a distributed ObjectGrid has an availability state associated with it. This state refers to the shard's ability to process incoming requests.

Since:

WAS XD 6.1.0.3, XC10

See Also:

[Serialized Form](#)

Field Summary	
s t a t i c A v a i l l i n e	<p>OFFLINE</p> <p>An AvailabilityState.OFFLINE indicates that a shard is offline and unable to process requests.</p>
s t a t i c A v a i l l i n e	<p>ONLINE</p>

b
i
l
l
i
t
y
S
t
a
t
e

An AvailabilityState.ONLINE indicates that a shard is online.

s
t
a
t
i
c
A
v
a
i
l
i
t
y
S
t
a
t
e

PRELOAD

An AvailabilityState.PRELOAD indicates that a shard is in the preload state.

s
t
a
t
i
c
A
v
a
i
l
i
t
y
S
t
a
t
e

QUIESCE

An AvailabilityState QUIESCE indicates that a shard is in quiesce.

s
t
a
t
i
c
A
v
a
i
l
i
t
y

UNKNOWN

An AvailabilityState.UNKNOWN indicates that the availability state of the shard could not be determined.

S
t
a
t
e

Method Summary

getId()
Returns the internal identifier for this state.

toString()

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

OFFLINE

```
public static final AvailabilityState OFFLINE
```

An `AvailabilityState.OFFLINE` indicates that a shard is offline and unable to process requests.

PRELOAD

```
public static final AvailabilityState PRELOAD
```

An `AvailabilityState.PRELOAD` indicates that a shard is in the preload state. When in the preload state, a shard will reject all requests that are not initiated from a client that is preloading data into the ObjectGrid.

ONLINE

```
public static final AvailabilityState ONLINE
```

An `AvailabilityState.ONLINE` indicates that a shard is online. A shard that is online is available for processing requests.

QUIESCE

```
public static final AvailabilityState QUIESCE
```

An `AvailabilityState.QUIESCE` indicates that a shard is in quiesce. Quiesce is a transitional state. Shards that are in the quiesce state are on their way to being offline. A shard in the quiesce state will allow all pending transactions to complete before moving to the `AvailabilityState.OFFLINE`, assuming that all pending transactions complete within 30 seconds after entering the quiesce state.

UNKNOWN

public static final [AvailabilityState](#) UNKNOWN

An AvailabilityState.UNKNOWN indicates that the availability state of the shard could not be determined.

Method Detail

getId

public int **getId**()

Returns the internal identifier for this state.

Returns:

the internal id.

Since:

7.1.1

toString

public [String](#) **toString**()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

PREV CLASS [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#) [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package com.ibm.websphere.xs.ra

These are the eXtreme Scale Resource Adapter APIs that allows integration with a Java EE application.

See:

[Description](#)

Interface Summary	
ObjectGridJ2CConnectionMBean	Allows administration clients control the lifecycle of the eXtreme Scale resource adapter's connection.

Class Summary	
XSConnection	An XSConnection represents an application level connection handle to an eXtreme Scale ObjectGrid.
XSConnectionFactory	The XSConnectionFactory creates connections to eXtreme Scale ObjectGrids.
XSConnectionSpec	The ConnectionSpec for retrieving XSConnections from an XSConnectionFactory
XSManagedConnectionFactory	The XSManagedConnectionFactory create managed connections to the eXtreme Scale ObjectGrids.
XSResourceAdapter	The eXtreme Scale resource adapter

Package com.ibm.websphere.xs.ra Description

These are the eXtreme Scale Resource Adapter APIs that allows integration with a Java EE application.

Introduction

The eXtreme Scale Resource Adapter provides client connection management and local transaction support, allowing Java EE applications to look-up eXtreme Scale client connections and demarcate transactions using the [LocalTransaction](#) interface or the [Session](#) interface.

When used with WebSphere Application Server with last participant support enabled, the eXtreme Scale transaction can be enlisted in a global transaction as the last, single-phase participant.

Programming Tutorial

The following sections show snippets on the usage of the ConnectionFactory. The resource

adapter is is JCA 1.5 compliant. To use the resource adapter the following steps must be followed:

1. Install the resource adapter
2. Configure a J2C ConnectionFactory
3. Configure a javax.resource.cci.ConnectionFactory resource reference in the application.
4. Look-up the [XSConnectionFactory](#)
5. Choose one of the following transaction options:

Use auto-commit, local transactions:

1. Retrieve a [XSConnection](#)
2. Retrieve and use the [Session](#) to interact with the data grid.
3. Close the connection.

Use an ObjectGrid Session to demarcate a local transaction:

1. Retrieve a [XSConnection](#)
2. Retrieve the [Session](#)
3. Use the Session.begin() method to start the transaction.
4. Use the Session to interact with the data grid.
5. Use the Session.commit() or rollback() methods to end the transaction.
6. Close the connection.

Use a javax.resource.cci.LocalTransaction to demarcate a local transaction:

1. Retrieve a [XSConnection](#)
2. Retrieve the javax.resource.cci.LocalTransaction using the XSConnection.getLocalTransaction() method.
3. Use the LocalTransaction.begin() method to start the transaction.
4. Retrieve and use the [Session](#) to interact with the data grid.
5. Use the LocalTransaction.commit() or rollback() methods to end the transaction.
6. Close the connection.

Enlist the connection in a global transaction:

1. Lookup the UserTransaction.
2. Begin the global transaction
3. Retrieve a [XSConnection](#)
4. Retrieve and use the [Session](#)
5. Close the connection.
6. Commit or rollback the global transaction.

Installing the resource adapter

The eXtreme Scale resource adapter is included in the wxsra.rar resource adapter archive with the eXtreme Scale product. See the [WebSphere eXtreme Scale version 8.5 information center](#) (or later) for details on how to install and configure the resource adapter.

Configuring the J2C ConnectionFactory

The eXtreme Scale resource adapter allows configuring one or J2C ManagedConnectionFactory instances. Each ManagedConnectionFactory is managed by the application server and represents a connection to a single catalog service domain. the ManagedConnectionFactory can include the name of the data grid, or the data grid can be provided when the connection is retrieved by the application.

The ManagedConnectionFactory is configured in the application server configuration, bound to a global JNDI name, and provides the following configuration properties:

ConnectionNa

me	(Optional) The name of the eXtreme Scale client connection.
CatalogServiceEndpoints	The catalog service domain end points in the form: <host>:<port>,<host><port>. Required if catalog service domain is not set.
CatalogServiceDomain	The catalog service domain name. Required if catalog service end points are not set.
ObjectGridName	(Optional) The data grid name. If not specified, the client must provide the data grid name when retrieving the connection resource.
ObjectGridURL	(Optional) The URL of the client data grid, override XML file.
ObjectGridResource	(Optional) The resource path of the client data grid, override XML file.
ClientPropertiesURL	(Optional) The URL of the client properties file.
ClientPropertiesResource	(Optional) The resource path of the client properties file.

Obtaining a ConnectionFactory instance.

The Java EE application can use resource injection to inject a ConnectionFactory resource into the application, or it can be looked-up using a resource reference. The Java EE application must first configure resource reference for a javax.resource.cci.ConnectionFactory.

For example:

```
InitialContext ctx = new InitialContext();
XSConnectionFactory cf = (XSConnectionFactory) ctx.lookup("java:comp/env/wxsconnection");
```

Retrieving a Connection

After the [XSConnectionFactory](#) has been looked-up or injected into the application, use one of the getConnection() methods to retrieve a client connection to the data grid. The connection will automatically be established when the first connection is retrieved and will be maintained until the resource adapter is stopped or the connection is reset using the ObjectGridJ2CConnection management bean.

For example:

```
XSConnection con = cf.getConnection("MyGrid");
```

Obtaining an ObjectGrid Session instance.

The XSConnection provides a getSession() method that gives the application direct access to the ObjectGrid Session. The Session is used to interact with the data grid and is valid for the life of the XSConnection. The XSConnection is a handle to a connection and becomes invalid after the application context completes per the Java EE specification.

An eXtreme Scale local transaction can be driven by the Session, javax.resource.cci.LocalTransaction or a global transaction. The transaction methods cannot be mixed.

Closing a connection

After the application has finished using the connection, the connection must be closed. The Java EE container typically will also close the connection automatically at the appropriate times. When the connection is closed the Session and any other objects retrieved directly or indirectly from the connection become invalid.

com.ibm.websphere.xs.ra

Class XSManagedConnectionFactory

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#), [ManagedConnectionFactory](#), [ResourceAdapterAssociation](#)

```

public final class XSManagedConnectionFactory
extends Object
implements ManagedConnectionFactory, ResourceAdapterAssociation
  
```

The XSManagedConnectionFactory create managed connections to the eXtreme Scale ObjectGrids.

Since:

8.5, XC10

See Also:

[Serialized Form](#)

Constructor Summary

XSManagedConnectionFactory()
--

Method Summary

Object	createConnectionFactory() Creates a non-managed XSConnectionFactory instance.
Object	createConnectionFactory(ConnectionManager conMgr) Creates a managed XSConnectionFactory instance.
ManagedConnection	createManagedConnection(Subject subject, ConnectionRequestInfo conReq) Creates a new physical connection to the underlying ObjectGrid

C
t
i
o
n

V
o
i
d

[destroy\(\)](#)

Destroy this MCF and release any other resources.

b
o
o
l
e
a
n

[equals\(Object obj\)](#)

S
t
r
i
n
g

[getCatalogServiceDomain\(\)](#)

Get the eXtreme Scale specific property: CatalogDomain

S
t
r
i
n
g

[getCatalogServiceEndpoints\(\)](#)

Get the eXtreme Scale specific property: CatalogServiceEndpoints

S
t
r
i
n
g

[getClientPropertiesResource\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesResource

S
t
r
i
n
g

[getClientPropertiesURL\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesURL

S
t
r
i
n
g

[getConnectionName\(\)](#)

Get the eXtreme Scale specific property: ConnectionName

P
r
i
n
t
W
r
i
t
e
r

[getLogWriter\(\)](#)

Get the log writer for this XManagedConnectionFactory instance.

S
t
r
i
n
g

[getObjectGridName\(\)](#)

Get the eXtreme Scale specific property: ObjectGridName

S t r i n g	<p>getObjectGridResource() Get the eXtreme Scale specific property: ObjectGridResource</p>
S t r i n g	<p>getObjectGridURL() Get the eXtreme Scale specific property: ObjectGridURL</p>
R e s o u r c e A d a p t e r	<p>getResourceAdapter() Retrieve the associated eXtreme Scale resource adapter instance.</p>
i n t	<p>hashCode()</p>
b o o l e a n	<p>isLocalGrid() Retrieve the eXtreme Scale specific property: LocalGrid</p>
M e n a g e d C o n n e c t i o n	<p>matchManagedConnections(Set connectionSet, Subject subject, ConnectionRequestInfo cxRequestInfo) Returns a matched connection from the candidate set of connections.</p>
V o i d	<p>setCatalogServiceDomain(String catalogServiceDomain) Set the eXtreme Scale specific property: CatalogDomain</p>
V o i d	<p>setCatalogServiceEndpoints(String catalogServiceEndpoints) Set the eXtreme Scale specific property: CatalogServiceEndpoints</p>
V o	<p>setClientPropertiesResource(String clientPropsResource) Get the eXtreme Scale specific property: ClientPropertiesResource</p>

i d	
v o i d	setClientPropertiesURL (String clientPropsURL) Set the eXtreme Scale specific property: ClientPropertiesURL
v o i d	setConnectionName (String connName) Set the eXtreme Scale specific property: ConnectionName
v o i d	setLocalGrid (boolean useLocal) Set the eXtreme Scale specific property: LocalGrid
v o i d	setLogWriter (PrintWriter out) Set the log writer for this ManagedConnectionFactory instance
v o i d	setObjectGridName (String objectGridName) Set the eXtreme Scale specific property: ObjectGrid Name
v o i d	setObjectGridResource (String objectGridResource) Set the eXtreme Scale specific property: ObjectGridResource
v o i d	setObjectGridURL (String objectGridURL) Set the eXtreme Scale specific property: ObjectGridURL
v o i d	setResourceAdapter (ResourceAdapter ra) Associate this object with an eXtreme Scale resource adapter.
S t r i n g	toString ()

Methods inherited from class [java.lang.Object](#)

[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XManagedConnectionFactory

```
public XManagedConnectionFactory()
```

Method Detail

getResourceAdapter

```
public ResourceAdapter getResourceAdapter()
```

Retrieve the associated eXtreme Scale resource adapter instance.

Specified by:

[getResourceAdapter](#) in interface [ResourceAdapterAssociation](#)

Returns:

The associated eXtreme Scale resource adapter

See Also:

[ResourceAdapterAssociation.getResourceAdapter\(\)](#)

setResourceAdapter

```
public void setResourceAdapter(ResourceAdapter ra)
    throws ResourceException
```

Associate this object with an eXtreme Scale resource adapter. Note, this method must be called exactly once. That is, the association must not change during the lifetime of this object.

Specified by:

[setResourceAdapter](#) in interface [ResourceAdapterAssociation](#)

Parameters:

ra - the eXtreme Scale resource adapter

Throws:

[ResourceException](#)

[IllegalStateException](#)

See Also:

[ResourceAdapterAssociation.setResourceAdapter\(ResourceAdapter\)](#)

createConnectionFactory

```
public Object createConnectionFactory()
    throws ResourceException
```

Creates a non-managed XSConnectionFactory instance. The XSConnectionFactory instance gets initialized with a default XSConnectionFactory instance provided by the eXtreme Scale resource adapter.

Specified by:

[createConnectionFactory](#) in interface [ManagedConnectionFactory](#)

Returns:

the new XSConnectionFactory instance

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.createConnectionFactory\(\)](#)

createConnectionFactory

```
public Object createConnectionFactory(ConnectionFactory conMgr)
    throws ResourceException
```

Creates a managed XSConnectionFactory instance. The XSConnectionFactory instance gets initialized with the passed ConnectionManager. In the managed scenario, ConnectionManager is provided by the application server.

Specified by:

[createConnectionFactory](#) in interface [ManagedConnectionFactory](#)

Parameters:

conMgr - - Connection Manager to be associated with the XSConnectionFactory.

Returns:

the new XSConnectionFactory instance

Throws:

[ResourceException](#) - ResourceAdapterInternalException

See Also:

[ManagedConnectionFactory.createConnectionFactory\(ConnectionManager\)](#)

createManagedConnection

```
public ManagedConnection createManagedConnection(Subject subject,  
                                                    ConnectionRequestInfo conReq)  
    throws ResourceException
```

Creates a new physical connection to the underlying ObjectGrid

Specified by:

[createManagedConnection](#) in interface [ManagedConnectionFactory](#)

Parameters:

subject - - Caller's security information. WebSphere eXtreme Scale client uses the credentials specified in a CredentialGenerator as part of the ConnectionSpec or the client properties.

conReq - - XS specific connection properties

Returns:

the new XSManagedConnection instance

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.createManagedConnection\(Subject, ConnectionRequestInfo\)](#)

getLogWriter

```
public PrintWriter getLogWriter()  
    throws ResourceException
```

Get the log writer for this XSManagedConnectionFactory instance.

Specified by:

[getLogWriter](#) in interface [ManagedConnectionFactory](#)

Returns:

PrintWriter

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.getLogWriter\(\)](#)

matchManagedConnections

```
public ManagedConnection matchManagedConnections(Set connectionSet,  
                                                    Subject subject,  
                                                    ConnectionRequestInfo cxRequestInfo)  
    throws ResourceException
```

Returns a matched connection from the candidate set of connections.

Specified by:

[matchManagedConnections](#) in interface [ManagedConnectionFactory](#)

Parameters:

connectionSet - - candidate connection set
subject - - caller's security information
cxRequestInfo - - XS specific connection properties

Returns:

XManagedConnection instance if acceptable match found otherwise null

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.matchManagedConnections\(Set, Subject, ConnectionRequestInfo\)](#)

setLogWriter

```
public void setLogWriter(PrintWriter out)
    throws ResourceException
```

Set the log writer for this ManagedConnectionFactory instance

Specified by:

[setLogWriter](#) in interface [ManagedConnectionFactory](#)

Parameters:

out - - PrintWriter - an out stream for error logging and tracing

Throws:

[ResourceException](#)

See Also:

[ManagedConnectionFactory.setLogWriter\(PrintWriter\)](#)

destroy

```
public void destroy()
```

Destroy this MCF and release any other resources.

getConnectionName

```
public String getConnectionName()
```

Get the eXtreme Scale specific property: ConnectionName

Returns:

The name of the eXtreme Scale client connection

setConnectionName

```
public void setConnectionName(String connName)
```

Set the eXtreme Scale specific property: ConnectionName

Parameters:

connName - - The name of the eXtreme Scale client connection

getObjectGridName

```
public String getObjectGridName()
```

Get the eXtreme Scale specific property: ObjectGridName

Returns:

The data grid name

setObjectGridName

```
public void setObjectGridName(String objectGridName)
```

Set the eXtreme Scale specific property: ObjectGrid Name

Parameters:

objectGridName - - The data grid name

getCatalogServiceEndpoints

```
public String getCatalogServiceEndpoints()
```

Get the eXtreme Scale specific property: CatalogServiceEndpoints

Returns:

The catalog service domain end points

setCatalogServiceEndpoints

```
public void setCatalogServiceEndpoints(String catalogServiceEndpoints)
```

Set the eXtreme Scale specific property: CatalogServiceEndpoints

Parameters:

catalogServiceEndpoints - - The catalog service domain end points

getCatalogServiceDomain

```
public String getCatalogServiceDomain()
```

Get the eXtreme Scale specific property: CatalogDomain

Returns:

The catalog service domain name defined in WebSphere Application Server

setCatalogServiceDomain

```
public void setCatalogServiceDomain(String catalogServiceDomain)
```

Set the eXtreme Scale specific property: CatalogDomain

Parameters:

catalogServiceDomain - - The catalog service domain name defined in WebSphere Application Server

getObjectGridURL

```
public String getObjectGridURL()
```

Get the eXtreme Scale specific property: ObjectGridURL

Returns:

The URL of the client data grid override XML file

setObjectGridURL

public void **setObjectGridURL**([String](#) objectGridURL)

Set the eXtreme Scale specific property: ObjectGridURL

Parameters:

objectGridURL - The URL of the client data grid override XML file

getObjectGridResource

public [String](#) **getObjectGridResource**()

Get the eXtreme Scale specific property: ObjectGridResource

Returns:

The resource path of the client data grid override XML file

setObjectGridResource

public void **setObjectGridResource**([String](#) objectGridResource)

Set the eXtreme Scale specific property: ObjectGridResource

Parameters:

objectGridResource - - The resource path of the client data grid override XML file

getClientPropertiesURL

public [String](#) **getClientPropertiesURL**()

Get the eXtreme Scale specific property: ClientPropertiesURL

Returns:

The URL of the client properties file

setClientPropertiesURL

public void **setClientPropertiesURL**([String](#) clientPropsURL)

Set the eXtreme Scale specific property: ClientPropertiesURL

Parameters:

clientPropsURL - - The URL of the client properties file

getClientPropertiesResource

public [String](#) **getClientPropertiesResource**()

Get the eXtreme Scale specific property: ClientPropertiesResource

Returns:

The resource path of the client properties file

setClientPropertiesResource

```
public void setClientPropertiesResource(String clientPropsResource)
```

Get the eXtreme Scale specific property: ClientPropertiesResource

Parameters:

clientPropsResource - - The resource path of the client properties file

setLocalGrid

```
public void setLocalGrid(boolean useLocal)
```

Set the eXtreme Scale specific property: LocalGrid

When set to true, the application uses the [XSConnectionSpec.setLocalObjectGrid\(com.ibm.websphere.objectgrid.ObjectGrid\)](#) to set the ObjectGrid instance to a local, in-memory grid instance or shard instance. If set to false (the default), the connection will be configured as a client connection to a remote data grid.

Parameters:

useLocal - set to true, to disable normal client ObjectGrid connection management.

Since:

8.6, XC10 2.5

isLocalGrid

```
public boolean isLocalGrid()
```

Retrieve the eXtreme Scale specific property: LocalGrid

Returns:

answers true if the connection is to be used with a local, im-memory ObjectGrid instance. Answers false if the connection represents a normal client ObjectGrid connection.

Since:

8.6, XC10 2.5

hashCode

```
public int hashCode()
```

Specified by:

[hashCode](#) in interface [ManagedConnectionFactory](#)

Overrides:

[hashCode](#) in class [Object](#)

See Also:

[ManagedConnectionFactory.hashCode\(\)](#)

equals

```
public boolean equals(Object obj)
```

Specified by:

[equals](#) in interface [ManagedConnectionFactory](#)

Overrides:

[equals](#) in class [Object](#)

See Also:

[ManagedConnectionFactory.equals\(Object\)](#)

toString

public [String](#) toString()

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [OD](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSConnectionSpec

[java.lang.Object](#)



All Implemented Interfaces:
[Cloneable](#), [ConnectionSpec](#)

```

public final class XSConnectionSpec
extends Object
implements ConnectionSpec, Cloneable
    
```

The ConnectionSpec for retrieving XSConnections from an XSConnectionFactory

Since:
 8.5, XC10

See Also:
[XSConnectionFactory](#)

Constructor Summary	
XSConnectionSpec()	Creates a default XSConnectionSpec instance.

Method Summary	
Object	clone()
boolean	equals(Object obj)
CredentialGenerator	getCredentialGenerator() Returns the CredentialGenerator object for this client security for the connection.

e
r
a
t
o
r

O
b
j
e
c
t
G
r
i
d

[getLocalObjectGrid\(\)](#)

Retrieve the local object grid instance used for the connection, if set.

S
t
r
i
n
g

[getObjectGridName\(\)](#)

Returns the object grid name for the connection.

i
n
t

[hashCode\(\)](#)

b
o
o
l
e
a
n

[isBeginNoWriteThrough\(\)](#)

Returns flag indicating whether the Session transaction should be started using using the [Session.beginNoWriteThrough\(\)](#) method.

b
o
o
l
e
a
n

[isMultiPartitionSupportEnabled\(\)](#)

Answers true if the [Session](#) will have the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

v
o
i
d

[setBeginNoWriteThrough\(boolean beginNoWriteThrough\)](#)

Set the boolean flag indicating whether the Session transaction should be started using using the [Session.beginNoWriteThrough\(\)](#) method.

v
o
i
d

[setCredentialGenerator\(CredentialGenerator credGen\)](#)

Sets the CredentialGenerator object for the client security for the connection.

v
o
i
d

[setLocalObjectGrid\(ObjectGrid instance\)](#)

Sets a local ObjectGrid instance to be used for connections.

v
o
i
d

[setMultiPartitionSupportEnabled\(boolean mptEnabled\)](#)

Set to true to retrieve a connection with a [Session](#) that has the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

v
o
i
d

[setObjectGridName\(String objectGridName\)](#)

Sets the object grid name to be used for the connection.

S
t
r

```
toString()
```

Methods inherited from class [java.lang.Object](#)

[finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSCConnectionSpec

```
public XSCConnectionSpec()
```

Creates a default XSCConnectionSpec instance.

Method Detail

isBeginNoWriteThrough

```
public boolean isBeginNoWriteThrough()
```

Returns flag indicating whether the Session transaction should be started using using the `Session.beginNoWriteThrough()` method.

Returns:

flag indicating whether the Session transaction should be started using using the `Session.beginNoWriteThrough()` method.

setBeginNoWriteThrough

```
public void setBeginNoWriteThrough(boolean beginNoWriteThrough)
```

Set the boolean flag indicating whether the Session transaction should be started using using the `Session.beginNoWriteThrough()` method.

Parameters:

`beginNoWriteThrough` -

See Also:

[Session.begin\(\)](#), [Session.beginNoWriteThrough\(\)](#)

getObjectGridName

```
public String getObjectGridName()
```

Returns the object grid name for the connection.

Returns:

object grid name

setObjectGridName

```
public void setObjectGridName(String objectGridName)
```

Sets the object grid name to be used for the connection.

Parameters:

objectGridName - object grid name to be used for the connection

getLocalObjectGrid

```
public ObjectGrid getLocalObjectGrid()
```

Retrieve the local object grid instance used for the connection, if set.

Returns:

null if the connection retrieves its ObjectGrid instance normally from the physical connection, or non-null if [setLocalObjectGrid\(ObjectGrid\)](#) was specified.

Since:

8.6, XC10 2.5

setLocalObjectGrid

```
public void setLocalObjectGrid(ObjectGrid instance)
```

Sets a local ObjectGrid instance to be used for connections. The local ObjectGrid instance overrides the standard behavior and allows a physical connection to be used with a specific local ObjectGrid instance, such as a shard.

Parameters:

instance - the local ObjectGrid instance.

Since:

8.6, XC10 2.5

getCredentialGenerator

```
public CredentialGenerator getCredentialGenerator()
```

Returns the CredentialGenerator object for this client security for the connection.

Returns:

the argument that was passed to the setCredGen(CredentialGenerator) method of this object or null if setCredGen was not previously called for this object.

setCredentialGenerator

```
public void setCredentialGenerator(CredentialGenerator credGen)
```

Sets the CredentialGenerator object for the client security for the connection.

Parameters:

credGen - a CredentialGenerator object

See Also:

[CredentialGenerator](#)

setMultiPartitionSupportEnabled

```
public void setMultiPartitionSupportEnabled(boolean mptEnabled)
```

Set to true to retrieve a connection with a [Session](#) that has the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

Parameters:

mptEnabled - set to true if the Session should be configured to write to multiple partitions.

Since:
8.6, XC10 2.5

isMultiPartitionSupportEnabled

public boolean **isMultiPartitionSupportEnabled**()

Answers true if the [Session](#) will have the [Session.getTxCommitProtocol\(\)](#) set to [Session.TxCommitProtocol.TWOPHASE](#).

Returns:
true if the Session is capable of writing to multiple partitions.

Since:
8.6, XC10 2.5

clone

public [Object](#) **clone**()

Overrides:
[clone](#) in class [Object](#)

hashCode

public int **hashCode**()

Overrides:
[hashCode](#) in class [Object](#)

equals

public boolean **equals**([Object](#) obj)

Overrides:
[equals](#) in class [Object](#)

toString

public [String](#) **toString**()

Overrides:
[toString](#) in class [Object](#)

[Overvi](#) [Packa](#) [Cla](#) [TreeSerializ](#) [Depreca](#) [IndexHelp](#)
[ew](#) [ge](#) [ss](#) [ed](#) [ted](#)

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All](#)
[Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSResourceAdapter

[java.lang.Object](#)



All Implemented Interfaces:

[ResourceAdapter](#)

```

public final class XSResourceAdapter
extends Object
implements ResourceAdapter
    
```

The eXtreme Scale resource adapter

Since:

8.5, XC10

Constructor Summary

[XSResourceAdapter](#)()

Method Summary

void [endpointActivation](#)([MessageEndpointFactory](#) msgEndpointFactory, [ActivationSpec](#) actSpec)
 This method does nothing as endpoint activation is not supported

void [endpointDeactivation](#)([MessageEndpointFactory](#) msgEndpointFactory, [ActivationSpec](#) spec)
 This method does nothing as endpoint deactivation is not supported

X
A
R
e
s
o
u
r
c
e
[
] [getXAResources](#)([ActivationSpec](#)[] spec)
 This method returns null as the XA protocol is not supported

void [start](#)([BootstrapContext](#) ctx)
 This method initializes the eXtreme Scale resource adapter instance

void [stop](#)()

i d	This method performs the shutdown of the eXtreme Scale resource adapter instance
S t r i n g	toString()

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSResourceAdapter

```
public XSResourceAdapter()
```

Method Detail

endpointActivation

```
public void endpointActivation(MessageEndpointFactory msgEndpointFactory,  
                             ActivationSpec actSpec)  
    throws ResourceException
```

This method does nothing as endpoint activation is not supported

Specified by:

[endpointActivation](#) in interface [ResourceAdapter](#)

Throws:

[ResourceException](#)

See Also:

[ResourceAdapter.endpointActivation\(MessageEndpointFactory, ActivationSpec\)](#)

endpointDeactivation

```
public void endpointDeactivation(MessageEndpointFactory msgEndpointFactory,  
                               ActivationSpec spec)
```

This method does nothing as endpoint deactivation is not supported

Specified by:

[endpointDeactivation](#) in interface [ResourceAdapter](#)

Throws:

[ResourceException](#)

See Also:

[ResourceAdapter.endpointDeactivation\(MessageEndpointFactory, ActivationSpec\)](#)

getXAResources

```
public XAResource[] getXAResources(ActivationSpec[] spec)  
    throws ResourceException
```

This method returns null as the XA protocol is not supported

Specified by:

[getXAResources](#) in interface [ResourceAdapter](#)

Throws:

[ResourceException](#)

See Also:

[ResourceAdapter.getXAResources\(ActivationSpec\[\]\)](#)

start

```
public void start(BootstrapContext ctx)
    throws ResourceAdapterInternalException
```

This method initializes the eXtreme Scale resource adapter instance

Specified by:

[start](#) in interface [ResourceAdapter](#)

Throws:

[ResourceAdapterInternalException](#)

See Also:

[ResourceAdapter.start\(BootstrapContext\)](#)

stop

```
public void stop()
```

This method performs the shutdown of the eXtreme Scale resource adapter instance

Specified by:

[stop](#) in interface [ResourceAdapter](#)

See Also:

[ResourceAdapter.stop\(\)](#)

toString

```
public String toString()
```

Overrides:

[toString](#) in class [Object](#)

Overview	Package	Classes	TreeSerialized	Deprecated	Index	Help
PREV CLASS	NEXT CLASS	FRAMES		NO FRAMES	All Classes	

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL: FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSConnection

[java.lang.Object](#)



All Implemented Interfaces:

[Connection](#)

```

public final class XSConnection
extends Object
implements Connection
  
```

An XSConnection represents an application level connection handle to an eXtreme Scale ObjectGrid.

Since:

8.5, XC10

Constructor Summary

[XSConnection\(\)](#)

Method Summary

V [close\(\)](#)

Initiates close of the connection handle at the application level.

I [createInteraction\(\)](#)

Creates an Interaction associated with this Connection.

L [getLocalTransaction\(\)](#)

Returns an LocalTransaction instance that enables a component to demarcate local transactions on the Connection.

t
i
o
n

C
o
n
n
e
c
t
i
o
n
M
e
t
a
D
a
t
a

[getMetaData\(\)](#)

Gets the information on the underlying eXtreme Scale instance represented through an active connection.

R
e
s
u
l
t
S
e
t
I
n
f
o

[getResultSetInfo\(\)](#)

Gets the information on the ResultSet functionality supported by a connected eXtreme Scale instance.

S
e
s
s
i
o
n

[getSession\(\)](#)

Retrieve the ObjectGrid Session associated with this connection.

S
t
r
i
n
g

[toString\(\)](#)

Methods inherited from class [java.lang.Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSConnection

```
public XSConnection()
```

Method Detail

getSession

```
public Session getSession()  
    throws ResourceException
```

Retrieve the ObjectGrid Session associated with this connection.

Returns:

The ObjectGrid Session

Throws:

[ResourceException](#)

close

```
public void close()  
    throws ResourceException
```

Initiates close of the connection handle at the application level.

Specified by:

[close](#) in interface [Connection](#)

Throws:

[ResourceException](#)

See Also:

[Connection.close\(\)](#)

createInteraction

```
public Interaction createInteraction()  
    throws ResourceException
```

Creates an Interaction associated with this Connection.

Specified by:

[createInteraction](#) in interface [Connection](#)

Returns:

Interaction instance

Throws:

[ResourceException](#)

[NotSupportedException](#)

See Also:

[Connection.createInteraction\(\)](#)

getLocalTransaction

```
public LocalTransaction getLocalTransaction()  
    throws ResourceException
```

Returns an LocalTransaction instance that enables a component to demarcate local transactions on the Connection.

Specified by:

[getLocalTransaction](#) in interface [Connection](#)

Returns:

LocalTransaction instance

Throws:

[ResourceException](#)

See Also:

[Connection.getLocalTransaction\(\)](#)

getMetaData

```
public ConnectionMetaData getMetaData()  
    throws ResourceException
```

Gets the information on the underlying eXtreme Scale instance represented through an active connection.

Specified by:

[getMetaData](#) in interface [Connection](#)

Returns:

WXS ConnectionMetaData

Throws:

[ResourceException](#)

See Also:

[Connection.getMetaData\(\)](#)

getResultSetInfo

```
public ResultSetInfo getResultSetInfo()  
    throws ResourceException
```

Gets the information on the ResultSet functionality supported by a connected eXtreme Scale instance.

Specified by:

[getResultSetInfo](#) in interface [Connection](#)

Returns:

ResultSetInfo instance

Throws:

[ResourceException](#)

See Also:

[Connection.getResultSetInfo\(\)](#)

toString

```
public String toString()
```

Overrides:

[toString](#) in class [Object](#)

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)
[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#) [All Classes](#)
SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Class XSConnectionFactory

[java.lang.Object](#)



All Implemented Interfaces:

[Serializable](#), [Referenceable](#), [ConnectionFactory](#)

```
public final class XSConnectionFactory
extends Object
implements ConnectionFactory
```

The XSConnectionFactory creates connections to eXtreme Scale ObjectGrids.

Usage example:

```
InitialContext ctx = new InitialContext();
XSConnectionFactory cf = (XSConnectionFactory) ctx.lookup("java:comp/env/wxsconnection");
XSConnection con = cf.getConnection("MyGrid");
Session ogSession = con.getSession();
...

con.close();
```

For additional examples, see the [package documentation](#).

Since:

8.5, XC10

See Also:

[Serialized Form](#)

Constructor Summary

[XSConnectionFactory\(\)](#)

Method Summary

[getCatalogServiceDomain\(\)](#)
Get the eXtreme Scale specific property: CatalogDomain

[getCatalogServiceEndpoints\(\)](#)
Get the eXtreme Scale specific property: CatalogServiceEndpoints

g
S
t
r
i
n
g
S
t
r
i
n
g
C
o
n
n
e
c
t
i
o
n
C
o
n
n
e
c
t
i
o
n
C
o
n
n
e
c
t
i
o
n
S
t
r
i
n
g
R
e
s
o
u
r
c
e
A
d
a
p
t
e
r
M
e
t
a

[getClientPropertiesResource\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesResource

[getClientPropertiesURL\(\)](#)

Get the eXtreme Scale specific property: ClientPropertiesURL

[getConnection\(\)](#)

Retrieve an XSConnection for the ObjectGrid configured on the ConnectionFactory.

[getConnection\(ConnectionSpec spec\)](#)

Retrieve an XSConnection for the ObjectGrid using the specified XSConnectionSpec properties.

[getConnection\(String gridName\)](#)

Retrieve an XSConnection for the specified ObjectGrid name.

[getConnectionName\(\)](#)

Get the eXtreme Scale specific property: ConnectionName

[getMetaData\(\)](#)

Retrieve the metadata information regarding the eXtreme Scale resource adapter

Data	
String	<p>getObjectGridName() Get the eXtreme Scale specific property: ObjectGridName</p>
String	<p>getObjectGridResource() Get the eXtreme Scale specific property: ObjectGridResource</p>
String	<p>getObjectGridURL() Get the eXtreme Scale specific property: ObjectGridURL</p>
RecordFactory	<p>getRecordFactory() Retrieve the RecordFactory instance</p>
Reference	<p>getReference() Gets the Reference instance</p>
boolean	<p>isLocalGrid() Get the eXtreme Scale specific property: LocalGrid</p>
void	<p>setReference(Reference ref) Sets the Reference instance</p>
String	<p>toString()</p>

Methods inherited from class java.lang.[Object](#)

[clone](#), [equals](#), [finalize](#), [getClass](#), [hashCode](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Constructor Detail

XSConnectionFactory

```
public XSConnectionFactory()
```

Method Detail

setReference

```
public void setReference(Reference ref)
```

Sets the Reference instance

Parameters:

ref - - Reference instance

See Also:

[Referenceable.setReference\(Reference\)](#)

getReference

```
public Reference getReference()  
    throws NamingException
```

Gets the Reference instance

Specified by:

[getReference](#) in interface [Referenceable](#)

Returns:

Reference instance

Throws:

[NamingException](#)

See Also:

[Referenceable.getReference\(\)](#)

getConnection

```
public Connection getConnection()  
    throws ResourceException
```

Retrieve an XSConnection for the ObjectGrid configured on the ConnectionFactory.

Specified by:

[getConnection](#) in interface [ConnectionFactory](#)

Returns:

the connection to the eXtreme Scale ObjectGrid

Throws:

[ResourceException](#)

See Also:

[ConnectionFactory.getConnection\(\)](#)

getConnection

```
public Connection getConnection(ConnectionSpec spec)
    throws ResourceException
```

Retrieve an XSConnection for the ObjectGrid using the specified XSConnectionSpec properties.

Specified by:

[getConnection](#) in interface [ConnectionFactory](#)

Parameters:

spec - the [XSConnectionSpec](#) properties used to retrieve the correct connection.

Returns:

the connection to the eXtreme Scale ObjectGrid

Throws:

[ResourceException](#)

See Also:

[ConnectionFactory.getConnection\(ConnectionSpec\)](#), [XSConnectionSpec](#)

getConnection

```
public Connection getConnection(String gridName)
    throws ResourceException
```

Retrieve an XSConnection for the specified ObjectGrid name.

Parameters:

gridName - - The ObjectGrid name

Returns:

the connection to the eXtreme Scale ObjectGrid

Throws:

[ResourceException](#)

getMetaData

```
public ResourceAdapterMetaData getMetaData()
    throws ResourceException
```

Retrieve the metadata information regarding the eXtreme Scale resource adapter

Specified by:

[getMetaData](#) in interface [ConnectionFactory](#)

Returns:

The eXtreme Scale resource adapter metadata

Throws:

[ResourceException](#)

See Also:

[ConnectionFactory.getMetaData\(\)](#)

getRecordFactory

```
public RecordFactory getRecordFactory()
    throws ResourceException
```

Retrieve the RecordFactory instance

Specified by:

[getRecordFactory](#) in interface [ConnectionFactory](#)

Throws:

[NotSupportedException](#)

[ResourceException](#)

See Also:

[ConnectionFactory.getRecordFactory\(\)](#)

toString

public [String](#) toString()

Overrides:

[toString](#) in class [Object](#)

getCatalogServiceDomain

public [String](#) getCatalogServiceDomain()

Get the eXtreme Scale specific property: CatalogDomain

Returns:

The catalog service domain name defined in WebSphere Application Server

getCatalogServiceEndpoints

public [String](#) getCatalogServiceEndpoints()

Get the eXtreme Scale specific property: CatalogServiceEndpoints

Returns:

The catalog service domain end points

getClientPropertiesResource

public [String](#) getClientPropertiesResource()

Get the eXtreme Scale specific property: ClientPropertiesResource

Returns:

The resource path of the client properties file

getClientPropertiesURL

public [String](#) getClientPropertiesURL()

Get the eXtreme Scale specific property: ClientPropertiesURL

Returns:

The URL of the client properties file

getConnectionName

public [String](#) getConnectionName()

Get the eXtreme Scale specific property: ConnectionName

Returns:

The name of the eXtreme Scale client connection

getObjectGridName

public [String](#) getObjectGridName()

Get the eXtreme Scale specific property: ObjectGridName

Returns:

The data grid name

getObjectGridResource

public [String](#) getObjectGridResource()

Get the eXtreme Scale specific property: ObjectGridResource

Returns:

The resource path of the client data grid override XML file

getObjectGridURL

public [String](#) getObjectGridURL()

Get the eXtreme Scale specific property: ObjectGridURL

Returns:

The URL of the client data grid override XML file

isLocalGrid

public boolean isLocalGrid()

Get the eXtreme Scale specific property: LocalGrid

Returns:

Is this managed connection factory used only for access to a local ObjectGrid instance?

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) | [DETAIL](#): [FIELD](#) | [CONSTR](#) | [METHOD](#) | [OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

com.ibm.websphere.xs.ra

Interface ObjectGridJ2CConnectionMBean

public interface **ObjectGridJ2CConnectionMBean**

Allows administration clients control the lifecycle of the eXtreme Scale resource adapter's connection.

The object name pattern for this MBean is:

com.ibm.websphere.objectgrid:type=ObjectGridJ2CConnection,objectGridName=<objectgrid>,domain=<domain name>,connectionName=<connection name>

Note: Additional properties may be included.

Since:

8.5, XC10

Field Summary	
s t a t i c S t r i n g	<p>CONNECTIONNAME_DEFAULT The default connection name prefix when not specified.</p>
s t a t i c S t r i n g	<p>CONNECTIONSTATUS_CONNECTED The connection status indicating that a connection has been established.</p>
s t a t i c S t r i n g	<p>CONNECTIONSTATUS_DISCONNECTED The connection status indicating that client is no longer connected.</p>
s	

t
a
t
i
c
s
t
r
i
n
g

MBEAN_TYPE

The MBean type

Method Summary

S
t
r
i
n
g

getCatalogServiceEndpoints()

The catalog service endpoints of the catalog service.

S
t
r
i
n
g

getConnectionName()

The name of the connection as specified on the J2C ConnectionFactory, or "DEFAULT" if not specified.

S
t
r
i
n
g

getConnectionStatus()

The status of the connection.

S
t
r
i
n
g

getDomainName()

The domain name of the catalog service domain as reported by the catalog service.

S
t
r
i
n
g

getObjectGridName()

The name of the ObjectGrid that is connected.

V
o
i
d

resetObjectGridConnection()

Reset the ObjectGrid connection.

Field Detail

MBEAN_TYPE

static final [String](#) MBEAN_TYPE

The MBean type

See Also:

[Constant Field Values](#)

CONNECTIONSTATUS_CONNECTED

static final [String](#) CONNECTIONSTATUS_CONNECTED

The connection status indicating that a connection has been established.

See Also:

[Constant Field Values](#)

CONNECTIONSTATUS_DISCONNECTED

static final [String](#) CONNECTIONSTATUS_DISCONNECTED

The connection status indicating that client is no longer connected.

See Also:

[Constant Field Values](#)

CONNECTIONNAME_DEFAULT

static final [String](#) CONNECTIONNAME_DEFAULT

The default connection name prefix when not specified.

See Also:

[Constant Field Values](#)

Method Detail

resetObjectGridConnection

void `resetObjectGridConnection()`

Reset the ObjectGrid connection.

This destroys the client connection to the ObjectGrid, including any local cache that may be created.

Subsequent uses of the ManagedConnectionFactory will result in a new ObjectGrid connection.

getObjectGridName

[String](#) `getObjectGridName()`

The name of the ObjectGrid that is connected.

Returns:

the ObjectGrid name

getConnectionName

[String](#) `getConnectionName()`

The name of the connection as specified on the J2C ConnectionFactory, or "DEFAULT" if not specified. If there are multiple connections with the same attributes in the same process, the "DEFAULT" name will have an integer appended.

Returns:
the connection name.

getDomainName

[String](#) getDomainName()

The domain name of the catalog service domain as reported by the catalog service.

Returns:
the catalog service domain name.

getCatalogServiceEndpoints

[String](#) getCatalogServiceEndpoints()

The catalog service endpoints of the catalog service.

Returns:
the catalog service endpoints

getConnectionStatus

[String](#) getConnectionStatus()

The status of the connection. Valid states include "CONNECTED" or "DISCONNECTED".

Other states may be returned in the future.

Returns:
one of the CONNECTIONSTATUS constants defined in this interface.

[Overview](#) [Package](#) [Classes](#) [Serialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

Package com.ibm.websphere.xsa

Class Summary

RestValue	Deprecated. Use the <i>RestValue</i> interface and the <i>RestValueFactory</i> .
---------------------------	---

com.ibm.websphere.xsa

Class RestValue

[java.lang.Object](#)



All Implemented Interfaces:

com.ibm.websphere.xs.rest.RestValue, [Externalizable](#), [Serializable](#)

Deprecated. Use the *RestValue* interface and the *RestValueFactory*.

```

public class RestValue
extends Object
implements com.ibm.websphere.xs.rest.RestValue, Externalizable
  
```

This object is used in the XC10 REST Gateway. It is a data wrapper for data to be accessed via XC10 REST Gateway. With its wrapped data, it will be stored in an object grid's map entry, with a String as the key of the map entry.

Since:

8.6, XC10 1.0.0.4

See Also:

[Serialized Form](#)

Field Summary

p r o t e c t e d	<p>compressed</p> <p>Deprecated. Flag to indicate if the value is compressed or not</p>
p r o t e c t e d	<p>contentType</p> <p>Deprecated. Content type of the contained value</p>
S t r i n g	

g	
p r o t e c t e d	value Deprecated. Contained value
b y t e []	

Constructor Summary

RestValue () Deprecated. Constructor
RestValue (byte[] value, String contentType) Deprecated. Constructor

Method Summary

b o o l e a n	equals (Object obj) Deprecated.
S t r i n g	getContentType () Deprecated. Return content type of the contained value
b y t e []	getValue () Deprecated. Return contained value
i n t	hashCode () Deprecated.
b o o l e a n	isCompressed () Deprecated. Indicate if the contained value is compressed
v o i d	readExternal (ObjectInput in) Deprecated.
v o	setCompressed (boolean compressed)

i d	Deprecated. Set indicator to indicate if the contained value is compressed
v o i d	setContentType(String contentType) Deprecated. Set content type of the contained value
v o i d	setValue(byte[] value) Deprecated. Set contained value
S t r i n g	toString() Deprecated.
v o i d	writeExternal(ObjectOutput out) Deprecated.

Methods inherited from class [java.lang.Object](#)
[clone](#), [finalize](#), [getClass](#), [notify](#), [notifyAll](#), [wait](#), [wait](#), [wait](#)

Field Detail

value

protected byte[] **value**

Deprecated.
Contained value

contentType

protected [String](#) **contentType**

Deprecated.
Content type of the contained value

compressed

protected boolean **compressed**

Deprecated.
Flag to indicate if the value is compressed or not

Constructor Detail

RestValue

public **RestValue()**

Deprecated.

Constructor

Since:

8.6, XC10 1.0.0.4

RestValue

```
public RestValue(byte[] value,  
                 String contentType)
```

Deprecated.

Constructor

Parameters:

value - A value to be wrapped by RestValue class.

contentType - Content type of the value.

Since:

8.6, XC10 1.0.0.4

Method Detail

hashCode

```
public int hashCode()
```

Deprecated.

Overrides:

[hashCode](#) in class [Object](#)

equals

```
public boolean equals(Object obj)
```

Deprecated.

Overrides:

[equals](#) in class [Object](#)

toString

```
public String toString()
```

Deprecated.

Overrides:

[toString](#) in class [Object](#)

getValue

```
public byte[] getValue()
```

Deprecated.

Return contained value

Specified by:

getValue in interface [com.ibm.websphere.xs.rest.RestValue](#)

Returns:

An array of byte that represents the contained value

Since:

8.6, XC10 1.0.0.4

setValue

```
public void setValue(byte[] value)
```

Deprecated.

Set contained value

Parameters:

value - An array of byte that represents the contained value

Since:

8.6, XC10 1.0.0.4

getContentType

```
public String getContentType()
```

Deprecated.

Return content type of the contained value

Specified by:

getContentType in interface `com.ibm.websphere.xs.rest.RestValue`

Returns:

A content type

Since:

8.6, XC10 1.0.0.4

setContentType

```
public void setContentType(String contentType)
```

Deprecated.

Set content type of the contained value

Parameters:

contentType - A content type

Since:

8.6, XC10 1.0.0.4

readExternal

```
public void readExternal(ObjectInput in)  
    throws IOException,  
           ClassNotFoundException
```

Deprecated.**Specified by:**

[readExternal](#) in interface [Externalizable](#)

Throws:

[IOException](#)
[ClassNotFoundException](#)

writeExternal

public void writeExternal([ObjectOutput](#) out)
throws [IOException](#)

Deprecated.

Specified by:

[writeExternal](#) in interface [Externalizable](#)

Throws:

[IOException](#)

isCompressed

public boolean isCompressed()

Deprecated.

Indicate if the contained value is compressed

Returns:

true if the contained value is compressed

Since:

8.6, XC10 1.0.0.4

setCompressed

public void setCompressed(boolean compressed)

Deprecated.

Set indicator to indicate if the contained value is compressed

Parameters:

compressed - A boolean value that indicate if the contained value is compressed

[Overview](#) [Package](#) [Classes](#) [TreeSerialized](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#) [All Classes](#)

SUMMARY: [NESTED](#) | [FIELD](#) | [CONSTR](#) | [METH](#) DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)
[OD](#)

**IBM WebSphere® DataPower®
XC10 Appliance
Release 2.5 Client API
Specification**

IBM WebSphere eXtreme Scale Client for .NET API Specification

Contents

- [Namespaces](#)

Namespaces

Namespace

Description

The IBM.WebSphere.Caching namespace is the parent namespace for the the WebSphere eXtreme Scale C# client data access application programming interfaces.

The primary entry point for interacting with the data grid is the [GridManagerFactory](#), which provides access to the [IGridManager](#) and [IGrid](#) interfaces.

[IBM.WebSphere.Caching](#)

The [IGridManager](#) interface provides methods to connect and disconnect from a catalog service domain. It also provides access to IGrid instances from the catalog service domain connections.

The [IGrid](#) interface allows clients to interact with a named data grid using various map interfaces, such as the [IGridMapPessimisticTx TKey, TValue](#) and [IGridMapPessimisticAutoTx TKey, TValue](#).

For additional information about data access APIs, see the [IBM.WebSphere.Caching.Map](#) namespace documentation.

The IBM.WebSphere.Caching.Map namespace includes the data access application programming interfaces. See the [IBM.WebSphere.Caching](#) namespace documentation for a description on how to access a map.

[IBM.WebSphere.Caching.Map](#)

The eXtreme Scale client supports transactional data access to individual maps using automatic and manual transactions. The following maps are available:

- The [IGridMapPessimisticAutoTx TKey, TValue](#) interface provides automatic transactions.
- The [IGridMapPessimisticTx TKey, TValue](#) interface provides manual transaction demarcation.

See each respective interface for programming examples.

[IBM.WebSphere.Caching.Security](#)

The IBM.WebSphere.Caching.Security namespace includes the application programming interfaces specific to security.

The eXtreme Scale client supports transport security and authentication and authorization using the [ICredentialGenerator](#) interface. Security is configured using a client properties file.

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The IBM.WebSphere.Caching namespace is the parent namespace for the the WebSphere eXtreme Scale C# client data access application programming interfaces.

The primary entry point for interacting with the data grid is the [GridManagerFactory](#), which provides access to the [IGridManager](#) and [IGrid](#) interfaces.

The [IGridManager](#) interface provides methods to connect and disconnect from a catalog service domain. It also provides access to IGrid instances from the catalog service domain connections.

The [IGrid](#) interface allows clients to interact with a named data grid using various map interfaces, such as the [IGridMapPessimisticTx TKey, TValue](#) and [IGridMapPessimisticAutoTx TKey, TValue](#).

For additional information about data access APIs, see the [IBM.WebSphere.Caching.Map](#) namespace documentation.

Classes

Class	Description
AvailabilityException	An AvailabilityException exception occurs when a target is not in the correct state to handle a request that it receives.
ClientServerTransactionCallbackException	A ClientServerTransactionCallbackException exception occurs when a method call to the client/server TransactionCallback encounters a remote request problem.
GridConfigurationException	An GridConfigurationException exception occurs when a configuration problem is found. This exception might occur when the configuration specified in the deployment policy, ObjectGrid descriptor, or security descriptor is not correct.
GridException	An GridException exception is the base exception class for all checked exceptions that occur in the product.
GridManagerFactory	The GridManagerFactory is a factory for IGridManager instances, and is the entrypoint for all interactions with the data grid.
GridServerRuntimeException	A GridServerRuntimeException exception is a generic wrapper for exceptions that occur in the server runtime.
LifecycleFailedException	A LifecycleFailedException exception occurs on unexpected lifecycle states.
MixedTransportException	A MixedTransportException is thrown when server and client have mismatched transport. For example, a server is using ORB, but the client is eXtremeIO
NoActiveTransactionException	A NoActiveTransactionException exception indicates that no active transactions exist.
NotReentrantException	A NotReentrantException occurs when a thread tries to run a map operation, such as calling a method on ObjectMap interface, when another thread is already running a map operation for the Session. A Session object can be used by a single thread only to perform concurrent map operations.
OrderedDictionary TKey, TValue	A generic version of the non-generic OrderedDictionary class.
OrderedDictionary TKey	

[TValue Enumerator](#) The enumerator for iterating through the [OrderedDictionary TKey, TValue](#).

[ReplicationVotedToRollbackTransactionException](#)

A [ReplicationVotedToRollbackTransactionException](#) exception occurs when a transaction was rolled back because some or all of the synchronous replicas did not apply the transaction.

[TransactionAlreadyActiveException](#)

A [TransactionAlreadyActiveException](#) exception occurs to indicate that a transaction is already active for the current Session. This exception does not cause the current active transaction to be rolled back, so the [isTransactionActive](#) method returns true.

[TransactionCallbackException](#)

A [TransactionCallbackException](#) exception occurs when a [TransactionCallback](#) method call fails.

[TransactionException](#)

A [TransactionException](#) exception is a general locking exception that indicates something went wrong with a transaction. Use the [isTransactionActive\(\)](#) and [wasTransactionRolledBack\(\)](#) methods to determine whether transaction is still active or was rolled back as a result of this exception.

[TransactionTimeoutException](#)

A [TransactionTimeoutException](#) exception occurs when a transaction exceeds the transaction timeout value that was specified on the [ObjectGrid](#) or [Session](#).

Interfaces

Interface

Description

[ICatalogDomainInfo](#)

Identifies a catalog service domain to be used for connecting to an eXtreme Scale data grid.

[ICatalogDomainManager](#)

The [ICatalogDomainManager](#) is a factory for [ICatalogDomainInfo](#) objects used to connect to a catalog service domain. Use the [CatalogDomainManager](#) to retrieve an [ICatalogDomainManager](#) instance.

[IClientConnectionContext](#)

The handle to a connection to a catalog service domain. An [IClientConnectionContext](#) is returned from the [Connect\(ICatalogDomainInfo\)](#) method when connecting to a data grid. When finished with the data grid, use the [Disconnect\(IClientConnectionContext\)](#) method to disconnect from the data grid.

[IGrid](#)

An [IGrid](#) is the client's access to a named data grid and corresponds to a configured [ObjectGrid](#) in the catalog service domain. An [IGrid](#) is thread safe and should be cached and reused by the application and is valid until the connection is severed.

[IGridManager](#)

Use an [IGrid](#) to get map instances. Maps are defined on the data grid and have unique names within each configured [ObjectGrid](#). Provides methods for connecting to data grids. Use the [GridManagerFactory](#) to obtain an instance to an [IGridManager](#) instance.

[IGridTransaction](#)

Defines the interface for a transaction.

[IOrderedDictionary TKey, TValue](#)

Specifies a generic version of the non-generic [IOrderedDictionary](#) interface.

[ITransactionable](#)

Implementors of [ITransactionable](#) provide transaction demarcation semantics.

Enumerations

Enum

ation**Description**

<input type="checkbox"/>	AvailabilityState	Each shard in a distributed data has an associated availability state. This state determines if the shard can process incoming requests.
<input type="checkbox"/>	TxnIsolationLevel	Specifies an enumeration that defines the valid transaction isolation level values.

Remarks

Note: The client API in this namespace and child namespaces will evolve in future releases. Therefore, to avoid problems with incompatible bindings in the future, interfaces should not be directly implemented nor extended unless explicitly noted in the interface documentation.

Examples

The following example illustrates how to connect to a data grid, retrieve an entry from a map, and disconnect from the grid:

[Copy to ClipboardPrint](#)

```
// Retrieve the IGridManager instance. IGridManager gm = GridManagerFactory.GetGridManager(); // Id
```

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An AvailabilityException exception occurs when a target is not in the correct state to handle a request that it receives.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching AvailabilityException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AvailabilityException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AvailabilityException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> AvailabilityException	Initializes a new instance of the AvailabilityException class.
<input type="checkbox"/> AvailabilityException(Exception)	Initializes a new instance of the AvailabilityException class with the specified exception cause.
<input type="checkbox"/> AvailabilityException(String)	Initializes a new instance of the AvailabilityException class with the specified error message.
<input type="checkbox"/> AvailabilityException(String, Exception)	Initializes a new instance of the AvailabilityException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)

<input type="checkbox"/>	NewAvailabilityState	Gets or sets the availability state. This state controls if the shard can process incoming requests.
<input type="checkbox"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> AvailabilityException	Initializes a new instance of the AvailabilityException class.
<input type="checkbox"/> AvailabilityException(Exception)	Initializes a new instance of the AvailabilityException class with the specified exception cause.
<input type="checkbox"/> AvailabilityException(String)	Initializes a new instance of the AvailabilityException class with the specified error message.
<input type="checkbox"/> AvailabilityException(String, Exception)	Initializes a new instance of the AvailabilityException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System Exception

Specifies the exception that caused of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AvailabilityException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that caused of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[AvailabilityException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AvailabilityException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AvailabilityException](#) type exposes the following members.
Properties

Name	Description
<input type="text"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="text"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="text"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="text"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text"/> NewAvailabilityState	Gets or sets the availability state. This state controls if the shard can process incoming requests.
<input type="text"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AvailabilityException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AvailabilityException NewAvailabilityState Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the availability state. This state controls if the shard can process incoming requests.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AvailabilityException Class](#)

[AvailabilityException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Each shard in a distributed data has an associated availability state. This state determines if the shard can process incoming requests.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Members

Member Name	Description
-------------	-------------

Description

- | | | |
|---------|---|--|
| OFFLINE | 0 | An AvailabilityState.OFFLINE state indicates that a shard is offline and cannot process requests. |
| PRELOAD | 1 | An AvailabilityState.PRELOAD state indicates that a shard is in the preload state. When a shard is in the preload state, the shard allows requests only from a client that is preloading data into the data grid. All other requests are rejected. |
| ONLINE | 2 | An AvailabilityState.ONLINE state indicates that a shard is online. A shard that is online is available for processing requests. |
| QUIESCE | 3 | An AvailabilityState. QUIESCE state indicates that a shard is in quiesce. Quiesce is a transitional state. Shards that are in the quiesce state are being taken offline. |
| UNKNOWN | 4 | An AvailabilityState.UNKNOWN state indicates that the availability state of the shard could not be determined. |

See Also

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A ClientServerTransactionCallbackException exception occurs when a method call to the client/server TransactionCallback encounters a remote request problem.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

[IBM.WebSphere.Caching TransactionCallbackException](#)

IBM.WebSphere.Caching ClientServerTransactionCallbackException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ClientServerTransactionCallbackException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerTransactionCallbackException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> ClientServerTransactionCallbackException	Initializes a new instance of the ClientServerTransactionCallbackException class.
<input type="checkbox"/> ClientServerTransactionCallbackException(Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified exception cause.
<input type="checkbox"/> ClientServerTransactionCallbackException(String)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message.
<input type="checkbox"/> ClientServerTransactionCallbackException(String, Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> ClientServerTransactionCallbackException	Initializes a new instance of the ClientServerTransactionCallbackException class.
<input type="checkbox"/> ClientServerTransactionCallbackException(Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified exception cause.
<input type="checkbox"/> ClientServerTransactionCallbackException(String)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message.
<input type="checkbox"/> ClientServerTransactionCallbackException(String, Exception)	Initializes a new instance of the ClientServerTransactionCallbackException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerTransactionCallbackException
Constructor (String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerTransactionCallbackException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerTransactionCallbackException Class](#)

[ClientServerTransactionCallbackException Members](#)

[ClientServerTransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerTransactionCallbackException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBase Exception	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObject Data	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> Member wiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)
[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional
information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerTransactionCallbackException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerTransactionCallbackException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An GridConfigurationException exception occurs when a configuration problem is found. This exception might occur when the configuration specified in the deployment policy, ObjectGrid descriptor, or security descriptor is not correct.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching GridConfigurationException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridConfigurationException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification




The [GridConfigurationException](#) type exposes the following members.
Constructors





Name	Description
<input type="checkbox"/> GridConfigurationException	Initializes a new instance of the GridConfigurationException class.
<input type="checkbox"/> GridConfigurationException(Exception)	Initializes a new instance of the GridConfigurationException class with a specified exception cause.
<input type="checkbox"/> GridConfigurationException(String)	Initializes a new instance of the GridConfigurationException class with the specified error message.
<input type="checkbox"/> GridConfigurationException(String, Exception)	Initializes a new instance of the GridConfigurationException class with the specified error message and exception cause.

[Back to Top](#)
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)
Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)

	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridConfigurationException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GridConfigurationException	Initializes a new instance of the GridConfigurationException class.
<input type="checkbox"/> GridConfigurationException(Exception)	Initializes a new instance of the GridConfigurationException class with a specified exception cause.
<input type="checkbox"/> GridConfigurationException(String)	Initializes a new instance of the GridConfigurationException class with the specified error message.
<input type="checkbox"/> GridConfigurationException(String, Exception)	Initializes a new instance of the GridConfigurationException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException Constructor IBM WebSphere™ eXtreme Scale Client for .NET
(Exception) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridConfigurationException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridConfigurationException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridConfigurationException Class](#)

[GridConfigurationException Members](#)

[GridConfigurationException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridConfigurationException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)









See Also

[GridConfigurationException Class](#)[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridConfigurationException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridConfigurationException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An GridException exception is the base exception class for all checked exceptions that occur in the product.

Inheritance Hierarchy

System Object

System Exception

IBM.WebSphere.Caching GridException

[IBM.WebSphere.Caching AvailabilityException](#)

[IBM.WebSphere.Caching GridConfigurationException](#)

[IBM.WebSphere.Caching LifecycleFailedException](#)

[IBM.WebSphere.Caching.Map CacheKeyNotFoundException](#)

[IBM.WebSphere.Caching.Map DuplicateKeyException](#)

[IBM.WebSphere.Caching.Map LoaderException](#)

[IBM.WebSphere.Caching.Map LockException](#)

[IBM.WebSphere.Caching.Map LockStrategyNotSupportedException](#)

[IBM.WebSphere.Caching.Map MultiplePartitionWriteException](#)

[IBM.WebSphere.Caching.Map OptimisticCollisionException](#)

[IBM.WebSphere.Caching.Map ReadOnlyException](#)

[IBM.WebSphere.Caching.Map TargetNotAvailableException](#)

[IBM.WebSphere.Caching.Map UndefinedMapException](#)

[IBM.WebSphere.Caching MixedTransportException](#)

[IBM.WebSphere.Caching NotReentrantException](#)

[IBM.WebSphere.Caching TransactionCallbackException](#)

[IBM.WebSphere.Caching TransactionException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[GridException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [GridException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> GridException	Initializes a new instance of the GridException class.
<input type="checkbox"/> GridException(Exc eption)	Initializes a new instance of the GridException class with a specified exception cause.
<input type="checkbox"/> GridException(Stri ng)	Initializes a new instance of the GridException class with the specified error message.
<input type="checkbox"/> GridException(Stri ng, Exception)	Initializes a new instance of the GridException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBase Exception	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjec tData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> Member wiseClon e	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GridException	Initializes a new instance of the GridException class.
<input type="checkbox"/> GridException(Exc eption)	Initializes a new instance of the GridException class with a specified exception cause.
<input type="checkbox"/> GridException(Stri ng)	Initializes a new instance of the GridException class with the specified error message.
<input type="checkbox"/> GridException(Stri ng, Exception)	Initializes a new instance of the GridException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[GridException Class](#)

[GridException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional
information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridException Class](#)

[GridException Members](#)

[GridException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [GridException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridException Class](#)









[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [GridException](#) type exposes the following members.

Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The GridManagerFactory is a factory for IGridManager instances, and is the entrypoint for all interactions with the data grid.

Inheritance Hierarchy

System Object

IBM.WebSphere.Caching GridManagerFactory

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridManagerFactory Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory
Members

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridManagerFactory](#) type exposes the following members.

Methods

	Name	Description
	GetGrid	Retrieves the IGridManager singleton instance that will be used by the client process to connect with the ObjectGrid.
	Manager	

[Back to Top](#)

See Also

[GridManagerFactory Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory Methods IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridManagerFactory](#) type exposes the following members.

Methods

	Name	Description
	GetGrid	Retrieves the IGridManager singleton instance that will be used by the client process to connect with the ObjectGrid.
	Manager	

[Back to Top](#)

See Also

[GridManagerFactory Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridManagerFactory GetGridManager Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the IGridManager singleton instance that will be used by the client process to connect with the ObjectGrid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

The IGridManager singleton

See Also

[GridManagerFactory Class](#)

[GridManagerFactory Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A GridServerRuntimeException exception is a generic wrapper for exceptions that occur in the server runtime.

Inheritance Hierarchy

System Object

System Exception

IBM.WebSphere.Caching GridServerRuntimeException

[IBM.WebSphere.Caching.Security AccessControlException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridServerRuntimeException](#) type exposes the following members.
Constructors

Name	Description
GridServerRuntimeException(String)	Initializes a new instance of the GridServerRuntimeException class.
GridServerRuntimeException(String, String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message and the Java exception class name.
GridServerRuntimeException(String, Exception, String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message, a reference to the inner exception that is the cause of this exception, and the Java exception class name.

[Back to Top](#)

Methods

Name	Description
Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
ToString	Provides a user-readable representation of this GridServerRuntimeException exception. (Overrides Exception ToString.)

[Back to Top](#)

Properties

Name	Description
Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

<input type="text"/>	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text"/>	JavaException ClassName	Identifies the Java Exception class name that this exception identifies.
<input type="text"/>	Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
GridServerRuntime eException(String)	Initializes a new instance of the GridServerRuntimeException class.
GridServerRuntime eException(String , String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message and the Java exception class name.
GridServerRuntime eException(String , Exception, String)	Initializes a new instance of the GridServerRuntimeException class with the specified error message, a reference to the inner exception that is the cause of this exception, and the Java exception class name.

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridServerRuntimeException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

javaExceptionClassName

Type: System.String

Specifies the Java exception class name.

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[GridServerRuntimeException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException Constructor IBM WebSphere™ eXtreme Scale Client for .NET
(String, String) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridServerRuntimeException](#) class with the specified error message and the Java exception class name.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

javaExceptionClassName

Type: System String

Specifies the Java exception class name.

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[GridServerRuntimeException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException Constructor
(String, Exception, String)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridServerRuntimeException](#) class with the specified error message, a reference to the inner exception that is the cause of this exception, and the Java exception class name.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

innerException

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

javaExceptionClassName

Type: System String

Specifies the Java exception class name.

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[GridServerRuntimeException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridServerRuntimeException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Provides a user-readable representation of this GridServerRuntimeException exception. (Overrides Exception ToString .)

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException ToString Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Provides a user-readable representation of this [GridServerRuntimeException](#) exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

The user-readable representation of this exception.

Implements

[_Exception ToString](#)

[See Also](#)

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridServerRuntimeException](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/> JavaException ClassName	Identifies the Java Exception class name that this exception identifies.
<input type="checkbox"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridServerRuntimeException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridServerRuntimeException JavaException IBM WebSphere™ eXtreme Scale Client for .NET API Specification
ClassName Property

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Identifies the Java Exception class name that this exception identifies.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridServerRuntimeException Class](#)

[GridServerRuntimeException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Identifies a catalog service domain to be used for connecting to an eXtreme Scale data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICatalogDomainInfo Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo
Members

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainInfo](#) type exposes the following members.

Properties

Name	Description
CatalogSe <input type="checkbox"/> rverAddre s ses	Retrieves the configured catalog service addresses. See the CreateCatalogDomainInfo(String) method for more details of the format of the catalog service address string.

[Back to Top](#)

See Also

[ICatalogDomainInfo Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainInfo](#) type exposes the following members.
Properties

Name	Description
CatalogSe <input type="checkbox"/> rverAddre s ses	Retrieves the configured catalog service addresses. See the CreateCatalogDomainInfo(String) method for more details of the format of the catalog service address string.

[Back to Top](#)

See Also

[ICatalogDomainInfo Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainInfo CatalogServerAddresses Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the configured catalog service addresses. See the [CreateCatalogDomainInfo\(String\)](#) method for more details of the format of the catalog service address string.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICatalogDomainInfo Interface](#)

[ICatalogDomainInfo Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainManager
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The ICatalogDomainManager is a factory for ICatalogDomainInfo objects used to connect to a catalog service domain. Use the [CatalogDomainManager](#) to retrieve an ICatalogDomainMananager instance.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[ICatalogDomainManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainManager](#) type exposes the following members.
Methods

Name	Description
------	-------------

Creates an instance of the ICatalogDomainInfo object for the specified catalog service addresses.

The catalog service addresses are expressed in a string of the form:

catalogServerAddresses ::= <catalogServiceEndpoint> [, <catalogServiceEndpoint>]

[CreateCatalogDomainInfo](#)

catalogServiceEndpoint ::= <hostName> : <listenerPort>

hostName ::= The IP address or host name of a catalog service.

listenerPort ::= The listener port that the catalog service is configured to use.

For example:

com.acme.server1:2809,com.acme.server2:2809

[Back to Top](#)

See Also

[ICatalogDomainManager Interface](#)
[IBM.WebSphere.Caching Namespace](#)


IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICatalogDomainManager](#) type exposes the following members.
Methods

Name	Description
 CreateCatalogDomainInfo	<p>Creates an instance of the ICatalogDomainInfo object for the specified catalog service addresses.</p> <p>The catalog service addresses are expressed in a string of the form:</p> <pre>catalogServerAddresses ::= <catalogServiceEndpoint> [, <catalogServiceEndpoint>]</pre> <pre>catalogServiceEndpoint ::= <hostName> : <listenerPort></pre> <p>hostName ::= The IP address or host name of a catalog service.</p> <p>listenerPort ::= The listener port that the catalog service is configured to use.</p> <p>For example:</p> <pre>com.acme.server1:2809,com.acme.server2:2809</pre>

[Back to Top](#)

See Also

[ICatalogDomainManager Interface](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICatalogDomainManager CreateCatalogDo IBM WebSphere™ eXtreme Scale Client for
mainInfo Method .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Creates an instance of the ICatalogDomainInfo object for the specified catalog service addresses.

The catalog service addresses are expressed in a string of the form:

catalogServerAddresses ::= <catalogServiceEndpoint> [,<catalogServiceEndpoint>]

catalogServiceEndpoint ::= <hostName> : <listenerPort>

hostName ::= The IP address or host name of a catalog service.

listenerPort ::= The listener port that the catalog service is configured to use.

For example:

com.acme.server1:2809,com.acme.server2:2809

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

catalogServerAddresses

Type: System String

The catalog service addresses.

Return Value

The ICatalogDomainInfo representing the catalog service domain addresses.

See Also

[ICatalogDomainManager Interface](#)

[ICatalogDomainManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IClientConnectionContext
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The handle to a connection to a catalog service domain. An IClientConnectionContext is returned from the [Connect\(ICatalogDomainInfo\)](#) method when connecting to a data grid.

When finished with the data grid, use the [Disconnect\(IClientConnectionContext\)](#) method to disconnect from the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IClientConnectionContext Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IClientConnectionContext
Members


IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IClientConnectionContext](#) type exposes the following members.

Properties

Name	Description
 DomainName	Retrieves the name of the catalog service domain.

[Back to Top](#)

See Also

[IClientConnectionContext Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.


IClientConnectionContext
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IClientConnectionContext](#) type exposes the following members.
Properties

Name	Description
 DomainName	Retrieves the name of the catalog service domain.

[Back to Top](#)

See Also

[IClientConnectionContext Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IClientConnectionContext DomainName Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the name of the catalog service domain.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IClientConnectionContext Interface](#)

[IClientConnectionContext Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid Interface IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An IGrid is the client's access to a named data grid and corresponds to a configured ObjectGrid in the catalog service domain.

An IGrid is thread safe and should be cached and reused by the application and is valid until the connection is severed.

Use an IGrid to get map instances. Maps are defined on the data grid and have unique names within each configured ObjectGrid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [IGrid](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable .)
<input type="checkbox"/> GetGridMapNames	Returns a list of map names that are associated with this data grid.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String)	Returns a map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String, ICredentialGenerator)	Returns a map instance for the specified map using specified credential object (userid password).
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String)	Returns a grid map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String, ICredentialGenerator)	Returns a grid map instance for the specified map using specified credential object (userid password)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> IsSecurityEnabled	Retreives a value that indicates whether data grid security is enabled.
<input type="checkbox"/> Name	Retrieves the data grid name.
<input type="checkbox"/> TransactionTimeout	Gets or sets the default transaction timeout for the data grid.

[Back to Top](#)

See Also

[IGrid Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [IGrid](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable .)
<input type="checkbox"/> GetGridMapNames	Returns a list of map names that are associated with this data grid.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String)	Returns a map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String, ICredentialGenerator)	Returns a map instance for the specified map using specified credential object (userid password).
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String)	Returns a grid map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String, ICredentialGenerator)	Returns a grid map instance for the specified map using specified credential object (userid password)

[Back to Top](#)

See Also

[IGrid Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapNames
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a list of map names that are associated with this data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Returns a list of map names.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String)	Returns a map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticAutoTx TKey, TValue (String, ICredentialGenerator)	Returns a map instance for the specified map using specified credential object (userid password).

[Back to Top](#)

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticAutoTx TKey, TValue Method (String)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map instance for the specified map.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

mapName

Type: System String

Specifies the map name.

Type Parameters

TKey

Specifies a generic type key.

TValue

Specifies a generic type Value.

Return Value

An IGridMapPessimisticAutoTx<TKey,TValue> instance

Exceptions

Exception	Condition
IBM.WebSphere.Caching.MapLockStrategyNotSupportedException	Occurs when the locking strategy for mapName is not supported by this map interface.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticAutoTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticAutoTx TKey, TValue
Method (String, ICredentialGenerator)

IBM WebSphere™ eXtreme Scale
Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map instance for the specified map using specified credential object (userid password).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

mapName

Type: System String

Specifies the map name.

credentialGenerator

Type: [IBM.WebSphere.Caching.Security ICredentialGenerator](#)

Specifies the Credential Generator object.

Type Parameters

TKey

Specifies a generic type key.

TValue

Specifies a generic type Value.

Return Value

An IGridMapPessimisticAutoTx<TKey,TValue> instance

Exceptions

Exception	Condition
IBM.WebSphere.Caching.Map LockStrategyNotSupportedException	Occurs when the locking strategy for mapName is not supported by this map interface.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticAutoTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticTx
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String)	Returns a grid map instance for the specified map.
<input type="checkbox"/> GetGridMapPessimisticTx TKey, TValue (String, ICredentialGenerator)	Returns a grid map instance for the specified map using specified credential object (userid password)

[Back to Top](#)

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticTx TKey,
TValue Method (String)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a grid map instance for the specified map.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

mapName

Type: System String

mapName is the name of the specified map

Type Parameters

TKey

Generic type key.

TValue

Generic Type Value.

Return Value

An IGridMapPessimisticTx<TKey,TValue> instance

Exceptions

Exception	Condition
-----------	-----------

IBM.WebSphere.Caching.MapLockStrategyNotSupportedException	Occurs when the locking strategy for mapName is not configured for pessimistic locking.
--	---

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid GetGridMapPessimisticTx TKey, TValue
Method (String, ICredentialGenerator)

IBM WebSphere™ eXtreme Scale Client
for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a grid map instance for the specified map using specified credential object (userid password)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

mapName

Type: System String

mapName is the name of the specified map

credentialGenerator

Type: [IBM.WebSphere.Caching.Security ICredentialGenerator](#)

Specifies the Credential Generator object.

Type Parameters

TKey

Generic type key.

TValue

Generic Type Value.

Return Value

An IGridMapPessimisticTx<TKey,TValue> instance

Exceptions

Exception	Condition
IBM.WebSphere.Caching.Map LockStrategyNotSupportedException	Occurs when the locking strategy for mapName is not configured for pessimistic locking.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[GetGridMapPessimisticTx Overload](#)

[IBM.WebSphere.Caching Namespace](#)




IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [IGrid](#) type exposes the following members.

Properties

	Name	Description
	IsSecurityEnabled	Retrieves a value that indicates whether data grid security is enabled.
	Name	Retrieves the data grid name.
	TransactionTimeout	Gets or sets the default transaction timeout for the data grid.

[Back to Top](#)

See Also

[IGrid Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid IsSecurityEnabled
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retreives a value that indicates whether data grid security is enabled.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

true if data grid security is enabled; otherwise, false.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid Name Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the data grid name.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The data grid name.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGrid TransactionTimeout
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the default transaction timeout for the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The default transaction timeout value for the data grid.

See Also

[IGrid Interface](#)

[IGrid Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Interface IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Provides methods for connecting to data grids. Use the GridManagerFactory to obtain an instance to an IGridManager instance.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Examples

The following example illustrates how to retrieve an IGridManager and disconnect:

[Copy to ClipboardPrint](#)

```
// Retrieve the singleton IGridManager instance.
IGridManager gm = GridManagerFactory.GetGridManager();

// Retrieve and cache the grid, and retrieve and update data....
...

// Disconnect from the data grid when all done and
// null out any references.
gm.Disconnect(ccc);
grid = null;
ccc = null;
```

See Also

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [IGridManager](#) type exposes the following members.

Methods

Name	Description
Connect(ICatalogDomainInfo)	<p>Connects a client process to the data grid.</p> <p>Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.</p>
Connect(ICatalogDomainInfo, String)	<p>Connects a client process to the data grid using connection properties that are specified in the the configFilename.</p> <p>Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.</p>
Disconnect	<p>Disconnects a client process from a catalog server. The IClientConnectionContext and associated IGrid instances are invalid once disconnected.</p>
GetGrid	<p>Returns a client data grid instance that corresponds to the data grid that is identified by the specified name. The IGrid is thread safe and should be cached for the life of the application.</p> <p>A grid name cooresponds to an ObjectGrid name in the data grid configuration.</p>

[Back to Top](#)

Properties

Name	Description
CatalogDomainManager	<p>Retrieves the CatalogDomainManager, which is used for creating ICatalogDomainInfo objects.</p>

[Back to Top](#)

See Also

[IGridManager Interface](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [IGridManager](#) type exposes the following members.

Methods

Name	Description
	Connects a client process to the data grid.
<input type="checkbox"/> Connect(ICatalogDomainInfo)	Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.
	Connects a client process to the data grid using connection properties that are specified in the the configFilename.
<input type="checkbox"/> Connect(ICatalogDomainInfo, String)	Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.
<input type="checkbox"/> Disconnect	Disconnects a client process from a catalog server. The IClientConnectionContext and associated IGrid instances are invalid once disconnected.
<input type="checkbox"/> GetGrid	Returns a client data grid instance that corresponds to the data grid that is identified by the specified name. The IGrid is thread safe and should be cached for the life of the application.
	A grid name cooresponds to an ObjectGrid name in the data grid configuration.

[Back to Top](#)

See Also

[IGridManager Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
Connect(ICatalogDomainInfo)	Connects a client process to the data grid. Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.
Connect(ICatalogDomainInfo, String)	Connects a client process to the data grid using connection properties that are specified in the the configFilename. Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the Disconnect(IClientConnectionContext) method should be invoked to disconnect from the data grid.

[Back to Top](#)

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Connect Method
(ICatalogDomainInfo)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Connects a client process to the data grid.

Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the [Disconnect\(IClientConnectionContext\)](#) method should be invoked to disconnect from the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

catalogDomainInfo

Type: [IBM.WebSphere.Caching ICatalogDomainInfo](#)

Specifies the CatalogDomainInfo object that contains target catalog service domain information.

Return Value

Returns an IClientConnectionContext object that represents a handle to the data grid to which the client is connected.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a nullcatalogDomainInfo is specified.
IBM.WebSphere.Caching GridException	Occurs if there is an error connecting to the specified catalog domain.

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[Connect Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Connect Method
(ICatalogDomainInfo, String)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Connects a client process to the data grid using connection properties that are specified in the the configFilename.

Multiple invocations will produce additional connections. The IClientConnectionContext should be cached for the life of the application, and the [Disconnect\(IClientConnectionContext\)](#) method should be invoked to disconnect from the data grid.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

catalogDomainInfo

Type: [IBM.WebSphere.Caching ICatalogDomainInfo](#)

Specifies the CatalogDomainInfo object that contains target catalog service domain information.

configFilename

Type: System String

Specifies the full path and file name of the client configuration file or null to use the default configuration.

Return Value

Returns an IClientConnectionContext object that represents a handle to the data grid to which the client is connected.

Exceptions

Exception	Condition
System ArgumentException	Occurs when a nullcatalogDomainInfo is specified.
IBM.WebSphere.Caching GridException	Occurs if there is an error connecting to the specified catalog domain.
System.IO FileNotFoundException	Occurs if configFilename does not exist.

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[Connect Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager Disconnect
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Disconnects a client process from a catalog server. The IClientConnectionContext and associated IGrid instances are invalid once disconnected.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

context

Type: [IBM.WebSphere.Caching IClientConnectionContext](#)

Specifies the IClientConnectionContext object that was returned from a previous Connect method call.

Return Value

Returns true if the disconnect was successful, or false if the supplied context was not connected.

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager GetGrid
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a client data grid instance that corresponds to the data grid that is identified by the specified name. The IGrid is thread safe and should be cached for the life of the application.

A grid name corresponds to an ObjectGrid name in the data grid configuration.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

context

Type: [IBM.WebSphere.Caching IClientConnectionContext](#)

Specifies the IClientConnectionContext object returned from a Connect method call.

gridName

Type: System String

Specifies the name of the requested ObjectGrid.

Return Value

Returns client data grid instance that corresponds to the specified remote data grid.

Exceptions

Exception	Condition
-----------	-----------

IBM.WebSphere.Caching GridException	Occurs if the specified grid is not found or incompatible with the current configuration.
---	---

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [IGridManager](#) type exposes the following members.

Properties

Name	Description
<input type="text"/> CatalogDomainManager	Retrieves the CatalogDomainManager, which is used for creating ICatalogDomainInfo objects.

[Back to Top](#)

See Also

[IGridManager Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridManager CatalogDomainManager Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the CatalogDomainManager, which is used for creating [ICatalogDomainInfo](#) objects.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[IGridManager Interface](#)

[IGridManager Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Defines the interface for a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.





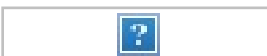
[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridTransaction](#) type exposes the following members.

Methods

	Name	Description
	Begin	Begins a new transaction.
	Commit	Commits a transaction.
	Flush	Apply the current changes to the server without committing the transaction.
	MarkRollbackOnly	Marks the current transaction as being rollback only.
	Rollback	Rolls back a transaction.

[Back to Top](#)

Properties

	Name	Description
<input type="checkbox"/>	Active	Determines if a transaction is currently active.
<input type="checkbox"/>	MarkedRollbackOnly	Determines if a transaction is roll back only.
<input type="checkbox"/>	TransactionIsolationLevel	Gets or sets the transaction isolation level.
<input type="checkbox"/>	TransactionTimeout	Gets or sets the amount of time in which a transaction must complete. To use an unlimited transaction timeout value, set the value to TimeSpan.Zero

[Back to Top](#)

See Also

[IGridTransaction Interface](#)

[IBM.WebSphere.Caching Namespace](#)


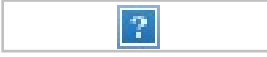


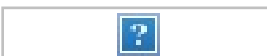
IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridTransaction](#) type exposes the following members.
Methods

	Name	Description
	Begin	Begins a new transaction.
	Commit	Commits a transaction.
	Flush	Apply the current changes to the server without committing the transaction.
	MarkRollbackOnly	Marks the current transaction as being rollback only.
	Rollback	Rolls back a transaction.

[Back to Top](#)

See Also

[IGridTransaction Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Begin
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Begins a new transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Commit
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Commits a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Flush
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Apply the current changes to the server without committing the transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction MarkRollbackOnly IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Marks the current transaction as being rollback only.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

error

Type: System Exception

Specifies the cause of the exception.

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Rollback
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Rolls back a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridTransaction](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Active	Determines if a transaction is currently active.
<input type="checkbox"/> MarkedRoll backOnly	Determines if a transaction is roll back only.
<input type="checkbox"/> Transaction IsolationLev el	Gets or sets the transaction isolation level.
<input type="checkbox"/> Transaction Timeout	Gets or sets the amount of time in which a transaction must complete. To use an unlimited transaction timeout value, set the value to TimeSpan.Zero

[Back to Top](#)

See Also

[IGridTransaction Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction Active
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines if a transaction is currently active.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction MarkedRollbackOnly IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines if a transaction is roll back only.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction TransactionIsolationLevel Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the transaction isolation level.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The transaction isolation level.

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridTransaction TransactionTimeout IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the amount of time in which a transaction must complete. To use an unlimited transaction timeout value, set the value to TimeSpan.Zero

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Field Value

The transaction timeout value.

See Also

[IGridTransaction Interface](#)

[IGridTransaction Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IOrderedDictionary TKey, TValue IBM WebSphere™ eXtreme Scale Client for .NET API
Interface Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies a generic version of the non-generic IOrderedDictionary interface.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Type Parameters

TKey

Specifies the key type.

TValue

Specifies the value type.

See Also

[IOrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)














IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)






IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IOrderedDictionary TKey, TValue](#) type exposes the following members.
Methods

Name	Description
 Add(T)	Adds an item to the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
 Add(TKey, TValue)	Adds an element with the provided key and value to the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
 Clear	Removes all items from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
 Contains	Determines whether the ICollection T contains a specific value. (Inherited from ICollection KeyValuePair TKey , TValue .)
 ContainsKey	Determines whether the IDictionary TKey, TValue contains an element with the specified key. (Inherited from IDictionary TKey , TValue .)
 CopyTo	Copies the elements of the ICollection T to an Array, starting at a particular Array index. (Inherited from ICollection KeyValuePair TKey , TValue .)
 GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable KeyValuePair TKey , TValue .)
 GetEnumerator	Returns an enumerator that iterates through a collection. (Inherited from IEnumerable.)
 Insert	Inserts an item with the specified key and value into the map at the specified index.
 Remove(TKey)	Removes the element with the specified key from the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
 Remove(T)	Removes the first occurrence of a specific object from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
 RemoveAt	Removes the item at the specified index from the IOrderedDictionary object.
 TryGetValue	Gets the value associated with the specified key. (Inherited from IDictionary TKey , TValue .)

[Back to Top](#)

Properties

Name	Description
 Count	Gets the number of elements contained in the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
 IsReadOnly	Gets a value indicating whether the ICollection T is read-only. (Inherited from ICollection KeyValuePair TKey , TValue .)
 Item TKey	Gets or sets the element with the specified key. (Inherited from IDictionary TKey , TValue .)
 Item Int32	Gets or sets the value with the specified index.
 Keys	Gets an ICollection T containing the keys of the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)



Values Gets an ICollection T containing the values in the IDictionary TKey, TValue .
(Inherited from IDictionary [TKey](#), [TValue](#) .)

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IOrderedDictionary TKey, TValue](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Add(T)	Adds an item to the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> Add(TKey, TValue)	Adds an element with the provided key and value to the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> Clear	Removes all items from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> Contains	Determines whether the ICollection T contains a specific value. (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> ContainsKey	Determines whether the IDictionary TKey, TValue contains an element with the specified key. (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> CopyTo	Copies the elements of the ICollection T to an Array, starting at a particular Array index. (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> GetEnumerator	Returns an enumerator that iterates through the collection. (Inherited from IEnumerable KeyValuePair TKey , TValue .)
<input type="checkbox"/> GetEnumerator	Returns an enumerator that iterates through a collection. (Inherited from IEnumerable.)
<input type="checkbox"/> Insert	Inserts an item with the specified key and value into the map at the specified index.
<input type="checkbox"/> Remove(TKey)	Removes the element with the specified key from the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
<input type="checkbox"/> Remove(T)	Removes the first occurrence of a specific object from the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
<input type="checkbox"/> RemoveAt	Removes the item at the specified index from the IOrderedDictionary object.
<input type="checkbox"/> TryGetValue	Gets the value associated with the specified key. (Inherited from IDictionary TKey , TValue .)

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Inserts an item with the specified key and value into the map at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System Int32

Specifies the zero-based index where the item should be inserted

key

Type: [TKey](#)

Specifies the key of the item to insert.

value

Type: [TValue](#)

Specifies the value of the item to insert.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Occurs when index is out of range.
System.ArgumentNullException	Occurs when a null key is specified.
System.ArgumentException	Occurs when key already exists in the IOrderedDictionary object.

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IOrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the item at the specified index from the IOrderedDictionary object.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

The zero-based index of the item to remove.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Occurs when index is less than 0 or greater than Count value.

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IOrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)







IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IOrderedDictionary TKey, TValue](#) type exposes the following members.
Properties

Name	Description
 Count	Gets the number of elements contained in the ICollection T . (Inherited from ICollection KeyValuePair TKey , TValue .)
 IsReadOnly	Gets a value indicating whether the ICollection T is read-only. (Inherited from ICollection KeyValuePair TKey , TValue .)
 Item TKey	Gets or sets the element with the specified key. (Inherited from IDictionary TKey , TValue .)
 Item Int32	Gets or sets the value with the specified index.
 Keys	Gets an ICollection T containing the keys of the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)
 Values	Gets an ICollection T containing the values in the IDictionary TKey, TValue . (Inherited from IDictionary TKey , TValue .)

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)
[IBM.WebSphere.Caching Namespace](#)



IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

	Name	Description
	Item TKey	Gets or sets the element with the specified key. (Inherited from IDictionary TKey , TValue .)
	Item Int32	Gets or sets the value with the specified index.

[Back to Top](#)

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IOrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IOrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET
Property (Int32) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value with the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

Specifies the zero-based index of the value to get or set.

Return Value

The value of the item at the specified index.

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Occurs when index is less than 0 or greater than the Count value.

See Also

[IOrderedDictionary TKey, TValue Interface](#)

[IOrderedDictionary TKey, TValue Members](#)

[Item Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ITransactionable
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Implementors of ITransactionable provide transaction demarcation semantics.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ITransactionable Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ITransactional
Members


IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ITransactionable](#) type exposes the following members.

Properties

Name	Description
 Transaction	The Transaction instance used to configure and demarcate a transaction.

[Back to Top](#)

See Also

[ITransactionable Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.


ITransactional
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ITransactionable](#) type exposes the following members.
Properties

Name	Description
 Transaction	The Transaction instance used to configure and demarcate a transaction.

[Back to Top](#)

See Also

[ITransactionable Interface](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ITransactionable Transaction Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The Transaction instance used to configure and demarcate a transaction.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ITransactionable Interface](#)

[ITransactionable Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LifecycleFailedException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LifecycleFailedException exception occurs on unexpected lifecycle states.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching LifecycleFailedException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LifecycleFailedException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LifecycleFailedException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LifecycleFailedException(String)	Initializes a new instance of the LifecycleFailedException class with the specified error message.
<input type="checkbox"/> LifecycleFailedException(String, Exception)	Initializes a new instance of the LifecycleFailedException class with the specified error message and exception cause.






[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)



StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)



TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[LifecycleFailedException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LifecycleFailedExcepti on(String)	Initializes a new instance of the LifecycleFailedException class with the specified error message.
<input type="checkbox"/> LifecycleFailedExcepti on(String, Exception)	Initializes a new instance of the LifecycleFailedException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LifecycleFailedException Class](#)

[LifecycleFailedException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LifecycleFailedException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LifecycleFailedException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[LifecycleFailedException Class](#)

[LifecycleFailedException Members](#)

[LifecycleFailedException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LifecycleFailedException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LifecycleFailedException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the exception message text.

cause

Type: System.Exception

Specifies the initial cause of the exception that caused this exception to occur.

See Also

[LifecycleFailedException Class](#)

[LifecycleFailedException Members](#)

[LifecycleFailedException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LifecycleFailedException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)





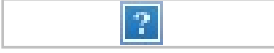


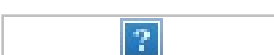
See Also

[LifecycleFailedException Class](#)[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LifecycleFailedException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LifecycleFailedException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A MixedTransportException is thrown when server and client have mismatched transport. For example, a server is using ORB, but the client is eXtremeIO

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching MixedTransportException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MixedTransportException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MixedTransportException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> MixedTransportException	Initializes a new instance of the MixedTransportException class.
<input type="checkbox"/> MixedTransportException(String)	Initializes a new instance of the MixedTransportException class with the specified error message.






[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)



StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)



TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[MixedTransportException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> MixedTransportException	Initializes a new instance of the MixedTransportException class.
<input type="checkbox"/> MixedTransportException(String)	Initializes a new instance of the MixedTransportException class with the specified error message.

[Back to Top](#)

See Also

[MixedTransportException Class](#)

[MixedTransportException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MixedTransportException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MixedTransportException Class](#)

[MixedTransportException Members](#)

[MixedTransportException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MixedTransportException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MixedTransportException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[MixedTransportException Class](#)

[MixedTransportException Members](#)

[MixedTransportException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MixedTransportException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MixedTransportException Class](#)

[IBM.WebSphere.Caching Namespace](#)





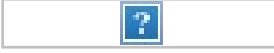



IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MixedTransportException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MixedTransportException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A NoActiveTransactionException exception indicates that no active transactions exist.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

[IBM.WebSphere.Caching TransactionException](#)

IBM.WebSphere.Caching NoActiveTransactionException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[NoActiveTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> NoActiveTransactionException	Initializes a new instance of the NoActiveTransactionException class.
<input type="checkbox"/> NoActiveTransactionException(Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified exception cause.
<input type="checkbox"/> NoActiveTransactionException(String)	Initializes a new instance of the NoActiveTransactionException class with the specified error message.
<input type="checkbox"/> NoActiveTransactionException(String, Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

Fields









Name	Description
ivTransactionRolledBack	Indicates whether the transaction was rolled back or

 [ck](#)

not.
(Inherited from [TransactionException.](#))

[Back to Top](#)

Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> NoActiveTransactionException	Initializes a new instance of the NoActiveTransactionException class.
<input type="checkbox"/> NoActiveTransactionException(Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified exception cause.
<input type="checkbox"/> NoActiveTransactionException(String)	Initializes a new instance of the NoActiveTransactionException class with the specified error message.
<input type="checkbox"/> NoActiveTransactionException(String, Exception)	Initializes a new instance of the NoActiveTransactionException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification
(String, Exception)

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NoActiveTransactionException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NoActiveTransactionException Class](#)

[NoActiveTransactionException Members](#)

[NoActiveTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.


[Send Feedback](#) on this topic to WAS Documentation Team.

NoActiveTransactionException Fields IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Fields

Name	Description
 ivTransactionRolledBack	Indicates whether the transaction was rolled back or not. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NoActiveTransactionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NoActiveTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A NotReentrantException occurs when a thread tries to run a map operation, such as calling a method on ObjectMap interface, when another thread is already running a map operation for the Session. A Session object can be used by a single thread only to perform concurrent map operations.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching NotReentrantException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

See Also

[NotReentrantException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NotReentrantException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> NotReentrantException	Initializes a new instance of the NotReentrantException class.
<input type="checkbox"/> NotReentrantException(Exception)	Initializes a new instance of the NotReentrantException class with a specified exception cause.
<input type="checkbox"/> NotReentrantException(String)	Initializes a new instance of the NotReentrantException class with the specified error message.
<input type="checkbox"/> NotReentrantException(String, Exception)	Initializes a new instance of the NotReentrantException class with the specified error message and exception cause.





[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception.

		(Inherited from Exception.)
<input type="text" value="Source"/>	Source	Gets or sets the name of the application or the object that causes the error.
<input type="text" value="StackTrace"/>	StackTrace	(Inherited from Exception.) Gets a string representation of the frames on the call stack at the time the current exception was thrown.
<input type="text" value="TargetSite"/>	TargetSite	(Inherited from Exception.) Gets the method that throws the current exception.

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> NotReentrantException	Initializes a new instance of the NotReentrantException class.
<input type="checkbox"/> NotReentrantException(Exception)	Initializes a new instance of the NotReentrantException class with a specified exception cause.
<input type="checkbox"/> NotReentrantException(String)	Initializes a new instance of the NotReentrantException class with the specified error message.
<input type="checkbox"/> NotReentrantException(String, Exception)	Initializes a new instance of the NotReentrantException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

NotReentrantException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [NotReentrantException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[NotReentrantException Class](#)

[NotReentrantException Members](#)

[NotReentrantException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NotReentrantException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [NotReentrantException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[NotReentrantException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A generic version of the non-generic OrderedDictionary class.

Inheritance Hierarchy

System Object

IBM.WebSphere.Caching OrderedDictionary TKey, TValue

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Type Parameters

TKey

The key type

TValue

The value type

See Also

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> OrderedDictionary TKey, TValue	Initializes a new instance of the OrderedDictionary TKey, TValue class.
<input type="checkbox"/> OrderedDictionary TKey, TValue (Int32)	Initializes a new instance of the OrderedDictionary TKey, TValue class with the specified capacity.
<input type="checkbox"/> OrderedDictionary TKey, TValue (IOrderedDictionary TKey, TValue)	Initializes a new instance of the OrderedDictionary TKey, TValue class having its contents copied from the specified IOrderedDictionary TKey, TValue .

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Add(KeyValuePair TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Add(TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Clear	Removes all items from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Contains	Determines whether the OrderedDictionary TKey, TValue contains the specified item.
<input type="checkbox"/> ContainsKey	Determines whether the OrderedDictionary TKey, TValue contains the specified key.
<input type="checkbox"/> CopyTo(Array, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified single-dimension array at the specified index.
<input type="checkbox"/> CopyTo(KeyValuePair TKey, TValue, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified array at the specified index.
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetEnumerator	Returns a <code>Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey, TValue>></code> object that iterates through the OrderedDictionary TKey, TValue .
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> Insert(Int32, KeyValuePair TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> Insert(Int32, TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)

<input type="checkbox"/>	Remove(KeyValuePair TKey, TValue)	Removes the specified item from the OrderedDictionary TKey, TValue .
<input type="checkbox"/>	Remove(TKey)	Removes the item with the specified key from the OrderedDictionary TKey, TValue .
<input type="checkbox"/>	RemoveAt	Removes the item at the specified index from the OrderedDictionary TKey, TValue .
<input type="checkbox"/>	ToString	Returns a String that represents the current Object. (Inherited from Object.)
<input type="checkbox"/>	TryGetValue	Get the value associated with the specified key.

[Back to Top](#)
Properties

Name	Description
<input type="checkbox"/> Count	Gets the number of items in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> IsFixedSize	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue has a fixed size.
<input type="checkbox"/> IsReadOnly	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue is read only.
<input type="checkbox"/> IsSynchronized	Gets a value indicating whether access to this Collection is synchronized (thread safe).
<input type="checkbox"/> Item Int32	Gets or sets the value associated with the specified index.
<input type="checkbox"/> Item TKey	Gets or sets the value associated with the specified key.
<input type="checkbox"/> Keys	Gets a ICollection T object containing all the keys in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> SyncRoot	Gets an object that can be used to synchronize this Collection.
<input type="checkbox"/> Values	Gets a ICollection T object containing all the values in the OrderedDictionary TKey, TValue .

[Back to Top](#)
See Also

[OrderedDictionary TKey, TValue Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> OrderedDictionary TKey, TValue	Initializes a new instance of the OrderedDictionary TKey, TValue class.
<input type="checkbox"/> OrderedDictionary TKey, TValue (Int32)	Initializes a new instance of the OrderedDictionary TKey, TValue class with the specified capacity.
<input type="checkbox"/> OrderedDictionary TKey, TValue (IOrderedDictionary TKey, TValue)	Initializes a new instance of the OrderedDictionary TKey, TValue class having its contents copied from the specified IOrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OrderedDictionary TKey, TValue](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[OrderedDictionary TKey, TValue Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue
Constructor (Int32)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OrderedDictionary TKey, TValue](#) class with the specified capacity.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

capacity

Type: System.Int32

The initial number of elements that [OrderedDictionary TKey, TValue](#) can contain.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[OrderedDictionary TKey, TValue Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Constructor
(IOrderedDictionary TKey, TValue)

IBM WebSphere™ eXtreme Scale Client
for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OrderedDictionary TKey, TValue](#) class having its contents copied from the specified [IOrderedDictionary TKey, TValue](#) .

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

orderedDictionary

Type: [IBM.WebSphere.Caching IOrderedDictionary TKey, TValue](#)

The [IOrderedDictionary TKey, TValue](#) whose contents are copied into the new [OrderedDictionary TKey, TValue](#) .

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[OrderedDictionary TKey, TValue Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Add(KeyValuePair TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Add(TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Clear	Removes all items from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Contains	Determines whether the OrderedDictionary TKey, TValue contains the specified item.
<input type="checkbox"/> ContainsKey	Determines whether the OrderedDictionary TKey, TValue contains the specified key.
<input type="checkbox"/> CopyTo(Array, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified single-dimension array at the specified index.
<input type="checkbox"/> CopyTo(KeyValuePair Pair TKey, TValue , Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified array at the specified index.
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetEnumerator	Returns a Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey,TValue>> object that iterates through the OrderedDictionary TKey, TValue .
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> Insert(Int32, KeyValuePair TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> Insert(Int32, TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> Remove(KeyValue Pair TKey, TValue)	Removes the specified item from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Remove(TKey)	Removes the item with the specified key from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> RemoveAt	Removes the item at the specified index from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)
<input type="checkbox"/> TryGetValue	Get the value associated with the specified key.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Add IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> Add(KeyValuePair TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Add(TKey, TValue)	Adds an item with the specified key and value to the OrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Add Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification
(KeyValuePair TKey, TValue)

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds an item with the specified key and value to the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

kvp

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The key-value pair to add.

Implements

ICollection T Add(T)

Exceptions

Exception	Condition
-----------	-----------

System ArgumentNullException	Thrown when the kvp key is null.
------------------------------	----------------------------------

System.Collections.Generic KeyNotFoundException	Thrown when the kvp key is not found.
---	---------------------------------------

Remarks

Use this property to add value in the collection. Values are appended to the end of the collection. A key cannot be null; however a value can be. To update an existing item, use either the [Item Int32](#) or the [Item TKey](#) property.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Add Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Add
Method (TKey, TValue)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds an item with the specified key and value to the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the item to add.

value

Type: [TValue](#)

The value of the item to add.

Implements

IDictionary TKey, TValue Add(TKey, TValue)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentNullException	Thrown when key is null.
------------------------------	--------------------------

Remarks

Use this property to add a new item to the collection. Items are appended to the end of the collection. A key cannot be null; however a value can be. To update an existing item, use either the [Item Int32](#) or the [Item Int32](#) property.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Add Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Clear Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes all items from the [OrderedDictionary TKey, TValue](#) .

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Implements

ICollection T Clear

IDictionary Clear

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Contains Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines whether the [OrderedDictionary TKey, TValue](#) contains the specified item.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

kvp

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The key-value pair to locate in the [OrderedDictionary TKey, TValue](#).

Return Value

true if the [OrderedDictionary TKey, TValue](#) contains the item associated with the specified kvp; otherwise, false.

Implements

ICollection T Contains(T)

Exceptions

Exception	Condition
-----------	-----------

System ArgumentNullException	Thrown when the kvp key is null.
------------------------------	----------------------------------

Remarks

The specified item's key must be contained in the [OrderedDictionary TKey, TValue](#) and it's associated value must be equal to the specified item's value for true to be returned.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue ContainsKey Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Determines whether the [OrderedDictionary TKey, TValue](#) contains the specified key.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key to locate in the [OrderedDictionary TKey, TValue](#).

Return Value

true if the [OrderedDictionary TKey, TValue](#) contains the item associated with the specified key; otherwise, false.

Implements

IDictionary TKey, TValue ContainsKey(TKey)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentNullException	Thrown when key is null.
------------------------------	--------------------------

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> CopyTo(Array, Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified single-dimension array at the specified index.
<input type="checkbox"/> CopyTo(KeyValuePair TKey, TValue , Int32)	Copies the items in the OrderedDictionary TKey, TValue into the specified array at the specified index.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue CopyTo
Method (Array, Int32)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Copies the items in the [OrderedDictionary TKey, TValue](#) into the specified single-dimension array at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

array

Type: System Array

The allocated single-dimension array into which the items from [OrderedDictionary TKey, TValue](#) will be copied.

index

Type: System Int32

The index into array where copying will begin.

Implements

ICollection CopyTo(Array, Int32)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentOutOfRangeException

Thrown when index is out of range.

System.ArgumentException

Thrown when array does not have sufficient size to hold the incoming data.

Remarks

The array must have sufficient storage to contain all items from the [OrderedDictionary TKey, TValue](#); otherwise a ArgumentException will be thrown. index must be equal or greater than 0.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[CopyTo Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue CopyTo Method IBM WebSphere™ eXtreme Scale Client
(KeyValuePair TKey, TValue , Int32) for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Copies the items in the [OrderedDictionary TKey, TValue](#) into the specified array at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

array

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The allocated array into which the items from [OrderedDictionary TKey, TValue](#) will be copied.

index

Type: System Int32

The zero-based index into array where copying will begin.

Implements

ICollection T CopyTo(T , Int32)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentOutOfRangeException	Thrown when index is out of range.
------------------------------------	------------------------------------

System.ArgumentException	Thrown when array does not have sufficient size to hold the incoming data.
--------------------------	--

Remarks

The array must have sufficient storage to contain all items from the [OrderedDictionary TKey, TValue](#); otherwise a ArgumentException will be thrown. index must be equal or greater than 0.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[CopyTo Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue GetEnumerator Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a `Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey,TValue>>` object that iterates through the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Return Value

A `Systems.Collections.Generic.IEnumerator<KeyValuePair<TKey,TValue>>` object that iterates through the [OrderedDictionary TKey, TValue](#).

Implements

IEnumerable T GetEnumerator

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> Insert(Int32, KeyValuePair TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.
<input type="checkbox"/> Insert(Int32, TKey, TValue)	Inserts an item with the specified key and value into the OrderedDictionary TKey, TValue at the specified index.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)
[OrderedDictionary TKey, TValue Members](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Insert Method
(Int32, KeyValuePair TKey, TValue)

IBM WebSphere™ eXtreme Scale Client
for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Inserts an item with the specified key and value into the [OrderedDictionary TKey, TValue](#) at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

index

Type: System.Int32

The zero-based index where the item should be inserted.

kvp

Type: System.Collections.Generic.KeyValuePair [TKey](#), [TValue](#)

The item to insert.

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentNullException	Thrown when the kvp key is null.
System.ArgumentOutOfRangeException	Thrown when index is out of range.

Remarks

The index must be greater than or equal to 0 and the index must be less than the number of elements in the collection; otherwise a `ArgumentOutOfRangeException` is thrown.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Insert Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Insert
Method (Int32, TKey, TValue)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Inserts an item with the specified key and value into the [OrderedDictionary TKey, TValue](#) at the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

The zero-based index where the item should be inserted.

key

Type: [TKey](#)

The key of the item to insert.

value

Type: [TValue](#)

The value of the item to insert.

Implements

[IOOrderedDictionary TKey, TValue Insert\(Int32, TKey, TValue\)](#)

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when the key is null.
System.ArgumentOutOfRangeException	Thrown when index is out of range.

Remarks

The index must be greater than or equal to 0 and the index must be less than the number of elements in the collection; otherwise a `ArgumentOutOfRangeException` is thrown.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Insert Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> Remove(KeyValuePair TKey, TValue)	Removes the specified item from the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Remove(TKey)	Removes the item with the specified key from the OrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Remove Method IBM WebSphere™ eXtreme Scale Client
(KeyValuePair TKey, TValue) for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the specified item from the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

kvp

Type: System.Collections.Generic KeyValuePair [TKey](#), [TValue](#)

The key-value pair to remove.

Return Value

true if the item is successfully removed; otherwise false is returned. false is also returned if the item is not found.

Implements

ICollection T Remove(T)

Exceptions

Exception	Condition
-----------	-----------

System ArgumentNullException	Thrown when the kvp key is null.
------------------------------	----------------------------------

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Remove Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Remove IBM WebSphere™ eXtreme Scale Client for .NET
Method (TKey) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the item with the specified key from the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the item to remove.

Return Value

true if the item is successfully removed; otherwise false. false is also returned if the key is not found.

Implements

IDictionary TKey, TValue Remove(TKey)

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when key is null.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Remove Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the item at the specified index from the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System Int32

The zero-based index of the item to remove.

Implements

[IOrderedDictionary TKey, TValue RemoveAt\(Int32\)](#)

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentOutOfRangeException	Thrown when index is out of range.
------------------------------------	------------------------------------

Remarks

The index must be greater than or equal to 0 and the index must be less than the number of elements in the collection; otherwise a `ArgumentOutOfRangeException` is thrown.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue TryGetValue Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Get the value associated with the specified key.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the value to get.

val

Type: [TValue](#)

The value associated with the specified key.

Return Value

true if the [OrderedDictionary TKey, TValue](#) contained the specified key; otherwise false.

Implements

IDictionary TKey, TValue TryGetValue(TKey, TValue)

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when key is null.

Remarks

If the key is not found, val is initialized to the default value associated with its data type.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Count	Gets the number of items in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> IsFixedSize	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue has a fixed size.
<input type="checkbox"/> IsReadOnly	Gets a value that indicates whether or not the OrderedDictionary TKey, TValue is read only.
<input type="checkbox"/> IsSynchronized	Gets a value indicating whether access to this Collection is synchronized (thread safe).
<input type="checkbox"/> Item Int32	Gets or sets the value associated with the specified index.
<input type="checkbox"/> Item TKey	Gets or sets the value associated with the specified key.
<input type="checkbox"/> Keys	Gets a ICollection T object containing all the keys in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> SyncRoot	Gets an object that can be used to synchronize this Collection.
<input type="checkbox"/> Values	Gets a ICollection T object containing all the values in the OrderedDictionary TKey, TValue .

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Count Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the number of items in the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The number of items in the [OrderedDictionary TKey, TValue](#).

Implements

ICollection T Count

ICollection Count

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue IsFixedSize Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a value that indicates whether or not the [OrderedDictionary TKey, TValue](#) has a fixed size.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Field Value

true if the [OrderedDictionary TKey, TValue](#) has a fixed size; otherwise false.

Implements

IDictionary IsFixedSize

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue IsReadOnly Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a value that indicates whether or not the [OrderedDictionary TKey, TValue](#) is read only.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

true if the [OrderedDictionary TKey, TValue](#) is read only; otherwise false.

Implements

ICollection T IsReadOnly

IDictionary IsReadOnly

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue IsSynchronized Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a value indicating whether access to this Collection is synchronized (thread safe).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

true if this Collection is synchronized.

Implements

ICollection IsSynchronized

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.



[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
 Item Int32	Gets or sets the value associated with the specified index.
 Item TKey	Gets or sets the value associated with the specified key.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET
Property (Int32) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value associated with the specified index.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

index

Type: System.Int32

The zero-based index of the value to get or set.

Field Value

The value of the item at the specified index.

Implements

[IOOrderedDictionary Item Int32](#)

Exceptions

Exception	Condition
System.ArgumentOutOfRangeException	Thrown when index is less than 0 or greater than the number of elements in the collection.
Remarks	

A set operation will add or replace a value in the collection, with new values being appended to the end of the collection. A key must be greater than or equal to 0. Values may be null.
See Also

[OrderedDictionary Class](#)

[OrderedDictionary Members](#)

[Item Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Item IBM WebSphere™ eXtreme Scale Client for .NET
Property (TKey) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value associated with the specified key.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the value to get or set.

Field Value

The value associated with the specified key. If the specified key does not exist during a get operation, a KeyNotFoundException is thrown.

Implements

IDictionary TKey, TValue Item TKey

Exceptions

Exception	Condition
System.ArgumentNullException	Thrown when key is null.
System.Collections.Generic.KeyNotFoundException	Thrown when key is not found during a get operation.
Remarks	

A set operation will add or replace a value in the collection, with new values being appended to the end of the collection. A key cannot be null; however a value can be. The key cannot be an int; if it is an int, then the [Item Int32](#) property is used.

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[Item Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Keys IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a ICollection T object containing all the keys in the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

A ICollection T object containing all the keys in the [OrderedDictionary TKey, TValue](#).

Implements

IDictionary TKey, TValue Keys

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue SyncRoot Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets an object that can be used to synchronize this Collection.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

An object that can be used to synchronize this Collection.

Implements

ICollection SyncRoot

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Values Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets a ICollection T object containing all the values in the [OrderedDictionary TKey, TValue](#) .

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

A ICollection T object containing all the values in the [OrderedDictionary TKey, TValue](#) .

Implements

IDictionary TKey, TValue Values

See Also

[OrderedDictionary TKey, TValue Class](#)

[OrderedDictionary TKey, TValue Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator Class

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The enumerator for interating through the [OrderedDictionary TKey, TValue](#).

Inheritance Hierarchy

System Object

IBM.WebSphere.Caching OrderedDictionary TKey, TValue Enumerator

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A generic version of the non-generic OrderedDictionary class.

The [OrderedDictionary TKey, TValue Enumerator](#) generic type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> MoveNext	Moves the iterator to the next item in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Reset	Moves the iterator to the first item in OrderedDictionary TKey, TValue .
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Current	Gets the item at the iterator's current position.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue Enumerator](#) generic type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> MoveNext	Moves the iterator to the next item in the OrderedDictionary TKey, TValue .
<input type="checkbox"/> Reset	Moves the iterator to the first item in OrderedDictionary TKey, TValue .
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator MoveNext Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Moves the iterator to the next item in the [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

true if the enumerator was successfully able to move the next item in the [OrderedDictionary TKey, TValue](#); otherwise, false is returned to indicate that the enumerator is at the end of the [OrderedDictionary TKey, TValue](#).

Implements

IEnumerator MoveNext

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator Reset Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Moves the iterator to the first item in [OrderedDictionary TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Implements

IEnumerator Reset

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey, TValue Enumerator Properties


IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OrderedDictionary TKey, TValue Enumerator](#) generic type exposes the following members.

Properties

Name	Description
 Current	Gets the item at the iterator's current position.

[Back to Top](#)

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OrderedDictionary TKey,
TValue Enumerator Current Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the item at the iterator's current position.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.ApiImpl (in Client.ApiImpl.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Implements

IEnumerator T Current

See Also

[OrderedDictionary TKey, TValue Enumerator Class](#)

[OrderedDictionary TKey, TValue Enumerator Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A ReplicationVotedToRollbackTransactionException exception occurs when a transaction was rolled back because some or all of the synchronous replicas did not apply the transaction.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.TransactionCallbackException](#)

IBM.WebSphere.Caching.ReplicationVotedToRollbackTransactionException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ReplicationVotedToRollbackTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [ReplicationVotedToRollbackTransactionException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with a specified exception cause.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String, Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

Overload List

Name	Description
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with a specified exception cause.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message.
<input type="checkbox"/> ReplicationVotedToRollbackTransactionException(String, Exception)	Initializes a new instance of the ReplicationVotedToRollbackTransactionException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionException IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionExceptio IBM WebSphere™ eXtreme Scale Client for
n Constructor (Exception) .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

The exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionExcepti IBM WebSphere™ eXtreme Scale Client for
on Constructor (String) .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReplicationVotedToRollbackTransactionException IBM WebSphere™ eXtreme Scale Client
Constructor (String, Exception) for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReplicationVotedToRollbackTransactionException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[ReplicationVotedToRollbackTransactionException Members](#)

[ReplicationVotedToRollbackTransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ReplicationVotedToRollbackTransactionException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBase Exception	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObject tData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> Member wiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional
information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ReplicationVotedToRollbackTransactionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReplicationVotedToRollbackTransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionAlreadyActiveException exception occurs to indicate that a transaction is already active for the current Session. This exception does not cause the current active transaction to be rolled back, so the isTransactionActive method returns true.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

[IBM.WebSphere.Caching TransactionException](#)

IBM.WebSphere.Caching TransactionAlreadyActiveException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionAlreadyActiveException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [TransactionAlreadyActiveException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> TransactionAlreadyActiveException	Initializes a new instance of the TransactionAlreadyActiveException class.
<input type="checkbox"/> TransactionAlreadyActiveException(Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified exception cause.
<input type="checkbox"/> TransactionAlreadyActiveException(String)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message.
<input type="checkbox"/> TransactionAlreadyActiveException(String, Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

Fields

Name	Description
<input type="checkbox"/> ivTransactionRolledBack	Indicates whether the transaction was rolled back or

not.
(Inherited from [TransactionException.](#))

[Back to Top](#)

Properties

Name	Description
<input type="text" value="Data"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="text" value="HelpLink"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="text" value="HResult"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="text" value="InnerException"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text" value="Message"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text" value="Source"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text" value="StackTrace"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text" value="TargetSite"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionAlreadyActiveException	Initializes a new instance of the TransactionAlreadyActiveException class.
<input type="checkbox"/> TransactionAlreadyActiveException(Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified exception cause.
<input type="checkbox"/> TransactionAlreadyActiveException(String)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message.
<input type="checkbox"/> TransactionAlreadyActiveException(String, Exception)	Initializes a new instance of the TransactionAlreadyActiveException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

The error message that explains the reason for the exception.

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException
Constructor (String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAlreadyActiveException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

The error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionAlreadyActiveException Class](#)

[TransactionAlreadyActiveException Members](#)

[TransactionAlreadyActiveException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.


[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAlreadyActiveException IBM WebSphere™ eXtreme Scale Client for .NET API
Fields Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAlreadyActiveException](#) type exposes the following members.
Fields

Name	Description
 ivTransactionRolledBack	Indicates whether the transaction was rolled back or not. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAlreadyActiveException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed. (Inherited from TransactionException .)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back. (Inherited from TransactionException .)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)
[IBM.WebSphere.Caching Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAlreadyActiveException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionAlreadyActiveException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionCallbackException exception occurs when a TransactionCallback method call fails.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching TransactionCallbackException

[IBM.WebSphere.Caching ClientServerTransactionCallbackException](#)

[IBM.WebSphere.Caching ReplicationVotedToRollbackTransactionException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionCallbackException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionCallbackException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> TransactionCallbackException	Initializes a new instance of the TransactionCallbackException class.
<input type="checkbox"/> TransactionCallbackException(Exception)	Initializes a new instance of the TransactionCallbackException class with a specified exception cause.
<input type="checkbox"/> TransactionCallbackException(String)	Initializes a new instance of the TransactionCallbackException class with the specified error message.
<input type="checkbox"/> TransactionCallbackException(String, Exception)	Initializes a new instance of the TransactionCallbackException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc ption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionCallbackException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionCallbackException	Initializes a new instance of the TransactionCallbackException class.
<input type="checkbox"/> TransactionCallbackException(Exception)	Initializes a new instance of the TransactionCallbackException class with a specified exception cause.
<input type="checkbox"/> TransactionCallbackException(String)	Initializes a new instance of the TransactionCallbackException class with the specified error message.
<input type="checkbox"/> TransactionCallbackException(String, Exception)	Initializes a new instance of the TransactionCallbackException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[TransactionCallbackException Class](#)
[TransactionCallbackException Members](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionCallbackException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionCallbackException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionCallbackException Class](#)

[TransactionCallbackException Members](#)

[TransactionCallbackException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionCallbackException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)








See Also

[TransactionCallbackException Class](#)
[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionCallbackException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionCallbackException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionException exception is a general locking exception that indicates something went wrong with a transaction. Use the `isTransactionActive()` and `wasTransactionRolledBack()` methods to determine whether transaction is still active or was rolled back as a result of this exception.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching GridException](#)

IBM.WebSphere.Caching TransactionException

[IBM.WebSphere.Caching NoActiveTransactionException](#)

[IBM.WebSphere.Caching TransactionAlreadyActiveException](#)

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> TransactionException on(Exception, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, Boolean)	Initializes a new instance of the TransactionException class with the specified error message and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(TransactionException, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, TransactionException, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, Exception, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed.
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back.

[Back to Top](#)

Fields

	Name	Description
<input type="checkbox"/>	ivTransactionRolledBack	Indicates whether the transaction was rolled back or not.

[Back to Top](#)
Properties

	Name	Description
<input type="checkbox"/>	Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>	HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/>	HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>	Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)
See Also

[TransactionException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionException on(Exception, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, Boolean)	Initializes a new instance of the TransactionException class with the specified error message and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(TransactionExce ption, Boolean)	Initializes a new instance of the TransactionException class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, TransactionExcepti on, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.
<input type="checkbox"/> TransactionException on(String, Exception, Boolean)	Initializes a new instance of the TransactionException class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

[Back to Top](#)

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor
(Exception, Boolean)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor
(String, Boolean)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with the specified error message and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor
(TransactionException, Boolean)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with a specified exception cause and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: [IBM.WebSphere.Caching TransactionException](#)

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

Remarks

The cause and a detailed message of (cause==null ? null : cause.toString()) is used, which typically contains the class and detailed message of cause. This constructor is useful for as a wrapper for other Throwable objects that occur.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor (String, TransactionException, Boolean)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: [IBM.WebSphere.Caching TransactionException](#)

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

Remarks

The detailed error message that is associated with the cause is not automatically incorporated in this the detailed message for this TransactionException exception.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Constructor (String, Exception, Boolean) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionException](#) class with the specified error message, the cause, and a special indication of whether the transaction was rolled back as a result of this exception.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

rolledBack

Type: System Boolean

Specifies if the transaction was rolled back. A value of true indicates that the transaction was rolled back.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[TransactionException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException Fields IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.
Fields

Name	Description
 ivTransactionRolledBack	Indicates whether the transaction was rolled back or not.

[Back to Top](#)

See Also

[TransactionException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException ivTransactionRoll IBM WebSphere™ eXtreme Scale Client for .NET
edBack Field API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Indicates whether the transaction was rolled back or not.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> isTransactionActive	Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed.
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> wasTransactionRolledBack	Returns true if the transaction was rolled back.

[Back to Top](#)

See Also

[TransactionException Class](#)[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException isTransactionActive Method IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns true if the transaction is active. A value of false indicates that the transaction never started or was completed.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

true if transaction is active, false if transaction never started or was complete.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionException wasTransactionRoll IBM WebSphere™ eXtreme Scale Client for .NET
edBack Method API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns true if the transaction was rolled back.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Returns true if transaction was rolled back, false otherwise.

See Also

[TransactionException Class](#)

[TransactionException Members](#)

[IBM.WebSphere.Caching Namespace](#)








IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionException Class](#)[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionTimeoutException exception occurs when a transaction exceeds the transaction timeout value that was specified on the ObjectGrid or Session.

Inheritance Hierarchy

System Object

System Exception

IBM.WebSphere.Caching TransactionTimeoutException

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionTimeoutException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> TransactionTimeoutException(String, Exception)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and exception cause.
<input type="checkbox"/> TransactionTimeoutException(String, String)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and the transaction that timed out.




[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> getTxIDString	Gets the String representation of the TxID.toString() method for the transaction that timed out.
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> whenOccurred	Returns the time when this TransactionTimeoutException exception was created.

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerExc	Gets the Exception instance that caused the current exception.

ption (Inherited from Exception.)



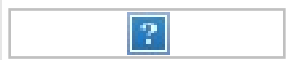
Message Gets a message that describes the current exception.
(Inherited from Exception.)



Source Gets or sets the name of the application or the object that causes the error.
(Inherited from Exception.)



StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)



TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionTimeoutException(String, Exception)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and exception cause.
<input type="checkbox"/> TransactionTimeoutException(String, String)	Initializes a new instance of the TransactionTimeoutException class with the specified error message and the transaction that timed out.

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionTimeoutException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[TransactionTimeoutException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException Constructor IBM WebSphere™ eXtreme Scale Client for .NET
(String, String) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionTimeoutException](#) class with the specified error message and the transaction that timed out.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

txIdString

Type: System.String

Specifies the result of the TxID.ToString() method for the transaction that timed out.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[TransactionTimeoutException Overload](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionTimeoutException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> getTxIDString	Gets the String representation of the TxID.toString() method for the transaction that timed out.
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)
<input type="checkbox"/> whenOccurred	Returns the time when this TransactionTimeoutException exception was created.

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)
[IBM.WebSphere.Caching Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException getTxIDSt IBM WebSphere™ eXtreme Scale Client for .NET
ring Method API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the String representation of the TxID.toString() method for the transaction that timed out.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Return Value

Specifies the result of TxID.toString() method for the transaction that timed out.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionTimeoutException whenOccu IBM WebSphere™ eXtreme Scale Client for .NET
rred Method API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns the time when this [TransactionTimeoutException](#) exception was created.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies a date object that represents the exact time when this exception object was created.

See Also

[TransactionTimeoutException Class](#)

[TransactionTimeoutException Members](#)

[IBM.WebSphere.Caching Namespace](#)








IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionTimeoutException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionTimeoutException Class](#)
[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TxnIsolationLevel Enumeration IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies an enumeration that defines the valid transaction isolation level values.

Namespace: [IBM.WebSphere.Caching](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Members

Member name	Value	Description
NotSet	-1	Specifies that the transaction isolation level value is not set.
ReadUncommitted	1	Specifies that dirty reads, non-repeatable, reads and phantom reads can occur.
ReadCommitted	2	Specifies that dirty reads are prevented, but non-repeatable reads and phantom reads can occur.
RepeatableRead	4	Specifies that dirty reads and non-repeatable reads are prevented, but phantom reads can occur.

See Also

[IBM.WebSphere.Caching Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The IBM.WebSphere.Caching.Map namespace includes the data access application programming interfaces. See the [IBM.WebSphere.Caching](#) namespace documentation for a description on how to access a map.

The eXtreme Scale client supports transactional data access to individual maps using automatic and manual transactions. The following maps are available:

- The [IGridMapPessimisticAutoTx TKey, TValue](#) interface provides automatic transactions.
- The [IGridMapPessimisticTx TKey, TValue](#) interface provides manual transaction demarcation.

See each respective interface for programming examples.

Classes

Class	Description
CacheKeyNotFoundException	A CacheKeyNotFoundException exception occurs if a key cannot be found in the cache.
ClassAlias	Specifies the class alias that you can use to correlate different class names that have the same class alias. The class alias and class fields or types are used to identify a unique class type ID during the object class serialization and de-serialization.
ClassAliasAttribute	The [ClassAlias] annotation can be specified for a user defined class.
ClientServerLoaderException	A ClientServerLoaderException exception is a base exception for any client/server operation exceptions.
DuplicateKeyException	A DuplicateKeyException exception occurs if a key cannot be inserted into the backing map because an object with the same key already exists.
FieldAlias	Specifies the field alias that you can use to correlate different class field names that have the same class field alias. The class alias and field alias or types are used to identify a unique class type ID during the object class serialization and de-serialization.
FieldAliasAttribute	The [FieldAlias] annotation can be specified for user defined class fields.
LoaderException	A LoaderException exception is the base exception that results for any exceptions that are encountered by a Loader.

[on](#)
[Loc](#)
[kDe](#)
[adlo](#)
[ckE](#)
[xce](#)
[ptio](#)
[n](#)

A LockStrategyNotSupportedException exception occurs when the lock manager detects a deadlock. This exception occurs to prevent the deadlock.

[Loc](#)
[kEx](#)
[cept](#)
[ion](#)

A LockException exception indicates errors with locking operations.

[Loc](#)
[kStr](#)
[ateg](#)
[yNo](#)

[tSu](#)
[ppo](#)
[rted](#)
[Exc](#)
[epti](#)
[on](#)

A LockStrategyNotSupportedException exception occurs if a map is configured with an unsupported lock strategy.

[Loc](#)
[kTi](#)
[meo](#)
[utE](#)
[xce](#)
[ptio](#)
[n](#)

A LockTimeoutException exception occurs when the lock manager detects that the lock wait time exceeded the maximum wait time. The timeout might be the result of a deadlock. If a deadlock is causing the timeout, the timeout is used to break the deadlock.

[Mul](#)
[tipl](#)
[ePa](#)
[rtiti](#)
[on](#)
[Writ](#)
[eEx](#)
[cept](#)
[ion](#)

A MultiplePartitionWriteException exception is a base exception for client/server operations when a user attempts to write to multiple remote partitions on remote servers in the same transaction.

An OptimisticCollisionException occurs when an optimistic locking strategy is used and more than one update transaction collides on the same map entry of an ObjectGrid instance. The first transaction to commit updates the version object for the map entry. Other transactions that read this same map entry before committing have the previous version object. When the other transactions try to commit, the version object that is read does not match the version that was last committed. Therefore, other transactions are prevented from updating a map entry with stale data.

[Opti](#)
[mist](#)
[icCo](#)
[llisi](#)
[onE](#)
[xce](#)
[ptio](#)
[n](#)

The default OptimisticCallback plug-in is used by the run time if an implementation is not provided by the application. If a well-constructed equals(Object) method is not on your value object, this exception occurs because the entire value object is used as the version object.

Because this exception indicates that the map entry contains stale data, stale map entries or entries as identified by the key parameter that is passed to the OptimisticCollisionException(String, String, String, Object) method are invalidated. If this exception is thrown by a Loader plug-in and a null reference is used as the key parameter by the loader, the run time assumes that the loader does not know which entry caused the exception. In this scenario, the LogSequence object is passed to the Loader.batchUpdate(TxID, LogSequence) method to determine which map entries to invalidate. Each LogElement entry in the LogSequence object that is type update or

delete is invalidated.

Specifies one or more attributes to use to calculate the partition hash code.

[PartitionKeyAttribute](#)

The PartitionKey attribute can be specified for the class using a path syntax to identify a single attribute. Multiple attributes can be specified using additional PartitionKey annotations on each field, using the [Order](#) attribute to specify the order in which the hash codes will be calculated.

The PartitionKey attribute is not inheritable.

[ReadOnlyException](#)

A ReadOnlyException exception occurs when a modify operation is attempted on a read-only map.

[TargetNotAvailableException](#)

A TargetNotAvailableException occurs when a remote target was not found or was not reachable.

[TransactionAffinityException](#)

A TransactionAffinityException exception occurs during server failover for inflight transactions. Applications can try the transaction again.

[UnavailableServiceException](#)

An UnavailableServiceException exception occurs when all servers are not running, or when all services are not available even though servers are running.

[UndefinedMapException](#)

An UndefinedMapException exception occurs to indicate that the map that an application tried to access is not defined in the ObjectGrid.

Interfaces

Interface	Description
IGridMap	The top level interface for all maps. Use the interface to retrieve an appropriate IGridMap instance.
IGridMapKey , TValue	Different IGridMap implementations are returned which allow additional operations for specific configurations and usage patterns.
IGridMapPessimisticAutoTx	This is a handle to a map using automatic transaction demarcation.
TKey , TValue	An instance of this IGridMapPessimisticAutoTx can only be used by the thread at a time. Use the Dispose method when finished with the map, to improve performance.
	This is a handle to a map using the pessimistic locking strategy and

manual transaction demarcation. All data access operations must occur within an active transaction.

[IGridMapPessimisticTx TKey, TValue](#)

Use the [Transaction](#) property to access the [IGridTransaction](#) instance that is associated with this map instance, and use the [Begin](#) method to begin a transaction. All keys in a single transaction must resolve to the same partition.

An instance of this `IGridMapPessimisticTx` is not thread-safe, and can only be used by the thread at a time. Use the `Dispose` method when finished with the map, to improve performance.

[IPartitionManager TKey, TValue](#)

An `IPartitionManager` provides properties and methods for determining how partitions are calculated. Retrieve an `IPartitionManager` from an [IGridMap TKey, TValue](#).

Enumerations

**E
n
u
m
e
r
a
t
i
o
n**

Description

[LockMode](#)

Specifies the strength of a lock to acquire.

[TTLType](#)

Every grid map has an optional, built in timed based evictor that is referred to as the "time to live" evictor or TTL evictor. Each grid map entry has an expiration time that determines how long the entry is allowed to live in the grid map. When the expiration time is reached, the TTL evictor causes the expired entry to be evicted from the grid map. This enum defines the `TTLType` value constants that determine how the the expiration time is computed for a map entry.

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A CacheKeyNotFoundException exception occurs if a key cannot be found in the cache.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map CacheKeyNotFoundException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[CacheKeyNotFoundException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CacheKeyNotFoundException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> CacheKeyNotFoundException	Initializes a new instance of the CacheKeyNotFoundException class.
<input type="checkbox"/> CacheKeyNotFoundException(Exception)	Initializes a new instance of the CacheKeyNotFoundException class with a specified exception cause.
<input type="checkbox"/> CacheKeyNotFoundException(String)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message.
<input type="checkbox"/> CacheKeyNotFoundException(String, Exception)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[CacheKeyNotFoundExcption Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> CacheKeyNotFoundException	Initializes a new instance of the CacheKeyNotFoundException class.
<input type="checkbox"/> CacheKeyNotFoundException(Exception)	Initializes a new instance of the CacheKeyNotFoundException class with a specified exception cause.
<input type="checkbox"/> CacheKeyNotFoundException(String)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message.
<input type="checkbox"/> CacheKeyNotFoundException(String, Exception)	Initializes a new instance of the CacheKeyNotFoundException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CacheKeyNotFoundException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CacheKeyNotFoundException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

The error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[CacheKeyNotFoundException Class](#)

[CacheKeyNotFoundException Members](#)

[CacheKeyNotFoundException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CacheKeyNotFoundException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)









See Also

[CacheKeyNotFoundException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CacheKeyNotFoundException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[CacheKeyNotFoundException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

Specifies the class alias that you can use to correlate different class names that have the same class alias. The class alias and class fields or types are used to identify a unique class type ID during the object class serialization and de-serialization.

The [ClassAlias] annotation can be specified for a user defined class.

Inheritance Hierarchy

System Object

System Attribute

IBM.WebSphere.Caching.Map ClassAliasAttribute

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

The syntax for the class could look like :

```
[ClassAlias("ACME_Employee")]
```

In the above example, ACME_Employee is the class alias for this user defined class.

If [ClassAlias] annotation is not defined, the name of this class is set as the ClassAlias.

[Copy to ClipboardPrint](#)

```
[ClassAlias("ACME_Employee")]
class Employee1 {
    [FieldAlias("Employee ID")]
    int empId = -1;

    [FieldAlias("Department No.")]
    int deptId = -1;

    [FieldAlias("Year Salary")]
    float salary = 0;

    [FieldAlias("SEX")]
    String sex = "M";

    int age = -1;
    String homeAddress = "";
}
```

When a ClassAlias and/or FieldAlias are specified in a user defined class, the ClassAlias and/or FieldAlias will be used to create or correlate with an object that are stored or will be stored in the grid. If two user defined classes (in a separate .NET application environment) have the different class name, but they were marked as the same ClassAlias, and all fields and field types are matched between these 2 classes, they will be correlated with the same class type ID even though they have the different class name. This way will allow the same class metadata to be reused between these 2 classes when running serialization and de-serialization in the different .NET application runtime, as well as to shared with Java when

the Alias for the class defined in Java and fields are also matched.

See Also

[ClassAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)









IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClassAliasAttribute](#) type exposes the following members.
Constructors

Name	Description
 ClassAliasAttribute	The class alias.



[Back to Top](#)

Methods

Name	Description
 Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
 Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
 GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
 GetType	Gets the Type of the current instance. (Inherited from Object.)
 IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
 Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
 MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
 ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
 Alias	The class alias.
 Typed	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[ClassAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClassAliasAttribute
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The class alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

value

Type: System String

The class alias value.

See Also

[ClassAliasAttribute Class](#)

[ClassAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClassAliasAttribute](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
<input type="checkbox"/> Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[ClassAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)



IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClassAliasAttribute](#) type exposes the following members.
Properties

	Name	Description
	Alias	The class alias.
	TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[ClassAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClassAliasAttribute Alias
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The class alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The class alias.

See Also

[ClassAliasAttribute Class](#)

[ClassAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A ClientServerLoaderException exception is a base exception for any client/server operation exceptions.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map LoaderException](#)

IBM.WebSphere.Caching.Map ClientServerLoaderException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[ClientServerLoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [ClientServerLoaderException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> ClientServerLoaderException	Initializes a new instance of the ClientServerLoaderException class.
<input type="checkbox"/> ClientServerLoaderException(Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified exception cause.
<input type="checkbox"/> ClientServerLoaderException(String)	Initializes a new instance of the ClientServerLoaderException class with the specified error message.
<input type="checkbox"/> ClientServerLoaderException(String, Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerLoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> ClientServerLoaderException	Initializes a new instance of the ClientServerLoaderException class.
<input type="checkbox"/> ClientServerLoaderException(Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified exception cause.
<input type="checkbox"/> ClientServerLoaderException(String)	Initializes a new instance of the ClientServerLoaderException class with the specified error message.
<input type="checkbox"/> ClientServerLoaderException(String, Exception)	Initializes a new instance of the ClientServerLoaderException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ClientServerLoaderException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ClientServerLoaderException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ClientServerLoaderException Class](#)

[ClientServerLoaderException Members](#)

[ClientServerLoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerLoaderException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)









See Also

[ClientServerLoaderException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ClientServerLoaderException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ClientServerLoaderException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A DuplicateKeyException exception occurs if a key cannot be inserted into the backing map because an object with the same key already exists.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map DuplicateKeyException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[DuplicateKeyException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [DuplicateKeyException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> DuplicateKeyException	Initializes a new instance of the DuplicateKeyException class.
<input type="checkbox"/> DuplicateKeyException(Exception)	Initializes a new instance of the DuplicateKeyException class with a specified exception cause.
<input type="checkbox"/> DuplicateKeyException(String)	Initializes a new instance of the DuplicateKeyException class with the specified error message.
<input type="checkbox"/> DuplicateKeyException(String, Exception)	Initializes a new instance of the DuplicateKeyException class with the specified error message and exception cause.





[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception.

(Inherited from Exception.)

Gets or sets the name of the application or the object that causes the error.

(Inherited from Exception.)

Gets a string representation of the frames on the call stack at the time the current exception was thrown.

(Inherited from Exception.)

Gets the method that throws the current exception.

(Inherited from Exception.)

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> DuplicateKeyException	Initializes a new instance of the DuplicateKeyException class.
<input type="checkbox"/> DuplicateKeyException (Exception)	Initializes a new instance of the DuplicateKeyException class with a specified exception cause.
<input type="checkbox"/> DuplicateKeyException (String)	Initializes a new instance of the DuplicateKeyException class with the specified error message.
<input type="checkbox"/> DuplicateKeyException (String, Exception)	Initializes a new instance of the DuplicateKeyException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

The error message that explains the reason for the exception.

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

DuplicateKeyException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [DuplicateKeyException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies an error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[DuplicateKeyException Class](#)

[DuplicateKeyException Members](#)

[DuplicateKeyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [DuplicateKeyException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [DuplicateKeyException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[DuplicateKeyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

Specifies the field alias that you can use to correlate different class field names that have the same class field alias. The class alias and field alias or types are used to identify a unique class type ID during the object class serialization and de-serialization.

The [FieldAlias] annotation can be specified for user defined class fields.

Inheritance Hierarchy

System Object

System Attribute

IBM.WebSphere.Caching.Map FieldAliasAttribute

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

The syntax for the field alias could look as follows :

```
[FieldAlias("Employee ID")]
```

In the above example, "Employee ID" is a field alias for the empId field in this user defined class.

If [FieldAlias] annotation is not defined, the name of this field is set as the FieldAlias.

[Copy to ClipboardPrint](#)

```
[ClassAlias("ACME_Employee")]
class Employee1 {
    [FieldAlias("Employee ID")]
    int empId = -1;

    [FieldAlias("Department No.")]
    int deptId = -1;

    [FieldAlias("Year Salary")]
    float salary = 0;

    [FieldAlias("SEX")]
    String sex = "M";

    int age = -1;
    String homeAddress = "";
}
```

When a ClassAlias and/or FieldAlias are specified in a user defined class, the ClassAlias and/or FieldAlias will be used to create or correlate with an object that are stored or will be stored in the grid. If two user defined classes (in a separate .NET application environment) have the different class name, but they were marked as the same ClassAlias, and all fields and field types are matched between these 2 classes, they will be correlated with the same class type ID even though they have the different class name. This way will allow the same class metadata to be reused between these 2 classes when running serialization and de-

serialization in the different .NET application runtime, as well as to shared with Java when the Alias for the class defined in Java and fields are also matched.

See Also

[FieldAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)


IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)









The [FieldAliasAttribute](#) type exposes the following members.

Constructors

Name	Description
 FieldAliasAttribute	The field alias.



[Back to Top](#)

Methods

Name	Description
 Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
 Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
 GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
 GetType	Gets the Type of the current instance. (Inherited from Object.)
 IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
 Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
 MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
 ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
 Alias	The field alias.
 TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[FieldAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

FieldAliasAttribute
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The field alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

value

Type: System String

The field alias value.

See Also

[FieldAliasAttribute Class](#)

[FieldAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [FieldAliasAttribute](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
<input type="checkbox"/> Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[FieldAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)



IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [FieldAliasAttribute](#) type exposes the following members.
Properties

	Name	Description
	Alias	The field alias.
	TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[FieldAliasAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

FieldAliasAttribute Alias
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The field alias.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The field alias.

See Also

[FieldAliasAttribute Class](#)

[FieldAliasAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The top level interface for all maps. Use the interface to retrieve an appropriate IGridMap instance.

Different IGridMap implementations are returned which allow additional operations for specific configurations and usage patterns.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Type Parameters

TKey

Generic type key.

TValue

Generic type value.

See Also

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

[IBM.WebSphere.Caching IGrid](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue
Members




IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMap TKey, TValue](#) type exposes the following members.

Properties

	Name	Description
	Grid	Retrieves the IGrid instance associated with this map.
	Name	Retrieves the map name.
	PartitionManager	Retrieves the IPartitionManager associated with this map.

[Back to Top](#)

See Also

[IGridMap TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)




IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMap TKey, TValue](#) type exposes the following members.
Properties

	Name	Description
	Grid	Retrieves the IGrid instance associated with this map.
	Name	Retrieves the map name.
	PartitionManager	Retrieves the IPartitionManager associated with this map.

[Back to Top](#)

See Also

[IGridMap TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue Grid IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the IGrid instance associated with this map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The IGrid instance.

See Also

[IGridMap TKey, TValue Interface](#)

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey, TValue Name IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the map name.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The map name.

See Also

[IGridMap TKey, TValue Interface](#)

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMap TKey,
TValue PartitionManager Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the IPartitionManager associated with this map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The IPartitionManager associated with this map.

See Also

[IGridMap TKey, TValue Interface](#)

[IGridMap TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Interface

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

This is a handle to a map using automatic transaction demarcation.

An instance of this IGridMapPessimisticAutoTx can only be used by the thread at a time. Use the Dispose method when finished with the map, to improve performance.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Type Parameters

TKey

Generic type key.

TValue

Generic type value.

Remarks

Each data access method includes a "Specification details" table that includes the following information:

Required permission: The permission required to use the API.

Pessimistic read lock acquired: The type of lock that is acquired when using pessimistic locking with repeatable read transaction isolation.

Cache tier: Identifies the map cache tiers that are included when fetching or updating cache entries in the call and under what circumstances. The following tiers are available for IGridMapPessimisticAutoTx maps:

- Server Cache (data grid)
- Loader (if enabled)

Examples

This sample demonstrates how to put a new cache entry into the data grid:

[Copy to ClipboardPrint](#)

```
// Assume we have already connected to the Grid...
```

```
IGrid grid = ...
```

```
// Retrieve a new map instance.
```

```
IGridMapPessimisticAutoTx<long, string> map = grid.GetGridMapPessimisticAutoTx<long, string>("MyPessimisticMap");
```

```
try
```

```
{
```

```
    // Put the entry in the cache.
```

```
    map.Put(123, "Value to cache");
```

```
}  
catch(GridException)  
{  
    // Handle any consequences of failed put.  
}  
  
// Dispose the map (optional, but it improves performance)  
map.Dispose();
```

See Also

[IGridMapPessimisticAutoTx TKey, TValue Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticAutoTx TKey, TValue](#) type exposes the following members.
Methods

Name	Description
Add	Adds the key-value pair to the data grid.
Add	The key must not exist before executing this method.
Add	A DuplicateKeyException is thrown when the duplicate key is discovered.
Add	Adds multiple key-value pairs to the data grid.
Add	The keys must not exist before executing this method.
Add	A DuplicateKeyException is thrown when the duplicate key is discovered, which
All	may be during a flush or commit operation, in which the exception will be an inner exception.
Contains	Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a nullkey cannot be specified.
Contains	Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
Key	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
Dispose	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
Get	If the map supports null values, use the ContainsKey(TKey) to test for a key that may have a null value.
Get	Retrieves the values associated with the list of keys that are specified in the keyList. If the value is not found, a null is returned.
All	If the map supports null values, use the ContainsKeyAll(IList TKey) to test for multiple keys that may have a null value.
Invalidate	Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).
e	If the key cannot be found in the map, it will be ignored.
Invalidate	Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).
All	If a key cannot be found in the map, it will be ignored.
Put	Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.
Put	Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.
Put	Puts multiple key-value pairs to the data grid, replacing or adding a new entries to each data grid tier as needed.
All	Note: This method has the same specification as the ObjectMap.upsertAll method

in the eXtreme Scale Java client.

[Remove](#) Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

[RemoveAll](#) Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

[Replace](#) Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) is thrown during commit.

[ReplaceAll](#) Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[ResetDefaults](#) Resets the configurable settings for the map back to configured values.

[Touch](#) Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TouchAll](#) Updates the last access time for the data grid entries specified in the keyList without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TryAdd](#) A variant of [Add\(TKey, TValue\)](#) that does not throw exceptions.

[TryAddAll](#) Adds the key-value pairs to the data grid.

[TryInvalidate](#) Removes the entry associated with the specified key from the cache, leaving the data behind unchanged.

[TryInvalidateAll](#) Removes the entries associated with the keys that are specified in the keyList from the cache, leaving the data behind unchanged.

[TryPut](#) Adds the key-value pair to the grid. If an entry for the key exists in the data grid, the value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryPutAll](#) Adds each key-value pair to the data grid. If an entry for a key exists in the data grid, its value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryRemove](#) Removes the entry that is associated with the specified key from the data grid. If the key has no matching entry in the map, no action is taken.

[TryRemoveAll](#) Removes the list of entries associated with the keys that are specified in the keyList. If a key in the keyList has no matching entry in the map, no action is taken for that key.

[TryReplace](#) Replaces the value of a key-value pair in the data grid. If an entry for the key exists in the data grid, its value is replaced with the specified value.

[TryReplaceAll](#) Replaces each key-value pair that is specified in the dictionary object in the data grid. If an entry for the key exists in the data grid, its value is replaced with the corresponding value specified.

[TryTouch](#) Updates the last access time for the data grid entry that matches the key.

[TryTouchAll](#) Updates the last access time for the data grid entries that match each key in the keyList.

[Back to Top](#)

Properties

**N
a
m
e**

Description

[Grid](#) Retrieves the IGrid instance associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

[Item](#) Gets or sets the value in the map using the specified key.

[Lock](#)

[Timeout](#) Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

[Name](#)

[Map](#) Retrieves the map name.
(Inherited from [IGridMap TKey, TValue](#).)

[Partition](#)

[Manager](#) Retrieves the IPartitionManager associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

[TimeToLive](#)

Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".

[Li](#) This property can only be used when the [TtlEvictorType](#) property is set to [v](#) LastAccessTime or LastUpdateTime on the map configuration. If this method is called [e](#) on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException is thrown.

To disable the TTL timeout, use a value of TimeSpan.Zero.

To revert the TTL value to the configured default, use a value of TimeSpan.MinValue.

[T](#)
[tl](#)
[E](#)
[vi](#)
[ct](#)
=[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

Gets the [TtlType](#) of the map's TTL evictor.

[Back to Top](#)

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticAutoTx TKey, TValue](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Add	Adds the key-value pair to the data grid.
<input type="checkbox"/> Add	The key must not exist before executing this method.
<input type="checkbox"/> Add	A DuplicateKeyException is thrown when the duplicate key is discovered.
<input type="checkbox"/> Add	Adds multiple key-value pairs to the data grid.
<input type="checkbox"/> Add	The keys must not exist before executing this method.
<input type="checkbox"/> Add	A DuplicateKeyException is thrown when the duplicate key is discovered, which may be during a flush or commit operation, in which the exception will be an inner exception.
<input type="checkbox"/> ContainsKey	Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a nullkey cannot be specified.
<input type="checkbox"/> ContainsKeyAll	Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
<input type="checkbox"/> Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
<input type="checkbox"/> Get	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
<input type="checkbox"/> Get	If the map supports null values, use the ContainsKey(TKey) to test for a key that may have a null value.
<input type="checkbox"/> Get	Retrieves the values associated with the list of keys that are specified in the keyList. If the value is not found, a null is returned.
<input type="checkbox"/> GetAll	If the map supports null values, use the ContainsKeyAll(IList TKey) to test for multiple keys that may have a null value.
<input type="checkbox"/> Invalidate	Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).
<input type="checkbox"/> Invalidate	If the key cannot be found in the map, it will be ignored.
<input type="checkbox"/> InvalidateAll	Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).
<input type="checkbox"/> InvalidateAll	If a key cannot be found in the map, it will be ignored.
<input type="checkbox"/> Put	Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.
<input type="checkbox"/> Put	Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.
<input type="checkbox"/> PutAll	Puts multiple key-value pairs to the data grid, replacing or adding a new entries to each data grid tier as needed.
<input type="checkbox"/> PutAll	Note: This method has the same specification as the ObjectMap.upsertAll method

in the eXtreme Scale Java client.

[Remove](#) Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

[RemoveAll](#) Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

[Replace](#) Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) is thrown during commit.

[ReplaceAll](#) Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[ResetDefaults](#) Resets the configurable settings for the map back to configured values.

[Touch](#) Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TouchAll](#) Updates the last access time for the data grid entries specified in the keyList without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

[TryAdd](#) A variant of [Add\(TKey, TValue\)](#) that does not throw exceptions.

[TryAddAll](#) Adds the key-value pairs to the data grid.

[TryInvalidate](#) Removes the entry associated with the specified key from the cache, leaving the data behind unchanged.

[TryInvalidateAll](#) Removes the entries associated with the keys that are specified in the keyList from the cache, leaving the data behind unchanged.

[TryPut](#) Adds the key-value pair to the grid. If an entry for the key exists in the data grid, the value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryPutAll](#) Adds each key-value pair to the data grid. If an entry for a key exists in the data grid, its value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

[TryRemove](#) Removes the entry that is associated with the specified key from the data grid. If the key has no matching entry in the map, no action is taken.

[TryRemoveAll](#) Removes the list of entries associated with the keys that are specified in the keyList. If a key in the keyList has no matching entry in the map, no action is taken for that key.

[TryReplace](#) Replaces the value of a key-value pair in the data grid. If an entry for the key exists in the data grid, its value is replaced with the specified value.

- [TryReplaceAll](#) Replaces each key-value pair that is specified in the dictionary object in the data grid. If an entry for the key exists in the data grid, its value is replaced with the corresponding value specified.
- [TryTouch](#) Updates the last access time for the data grid entry that matches the key.
- [TryTouchAll](#) Updates the last access time for the data grid entries that match each key in the keyList.

[Back to Top](#)

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey, TValue Add Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pair to the data grid.

The key must not exist before executing this method.

A DuplicateKeyException is thrown when the duplicate key is discovered.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be inserted in the data grid.

value

Type: [TValue](#)

Specifies the value to be inserted in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when the key exists.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds multiple key-value pairs to the data grid.

The keys must not exist before executing this method.

A DuplicateKeyException is thrown when the duplicate key is discovered, which may be during a flush or commit operation, in which the exception will be an inner exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic.IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to be inserted into the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when a dictionary key in entries already exists in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ContainsKey Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a nullkey cannot be specified.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to test in the map.

Return Value

Returns true if the key is found, false otherwise.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ContainsKeyAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies a list of keys to test in the map.

Return Value

Specifies a list of boolean values. If the key is found in the keyList, true is listed. Otherwise, false is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when keyList is null, or when a null key is specified in keyList.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Get Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.

If the map supports null values, use the [ContainsKey\(TKey\)](#) to test for a key that may have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to query.

Return Value

The value that is associated with the specified key if it exists; otherwise null is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified, or when the data type of the value that was obtained from the data grid does not match the declared data type.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue GetAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the values associated with the list of keys that are specified in the keyList. If the value is not found, a null is returned.

If the map supports null values, use the [ContainsKeyAll\(IList TKey \)](#) to test for multiple keys that may have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to query.

Return Value

A list of values that are associated with the supplied keys. If the value associated with a particular key is not in the data grid, null is returned in the list at the position that is associated with the key.

Exceptions

Exception	Condition
-----------	-----------

System ArgumentException	Occurs when the keyList is null, or when a null key is specified in keyList.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing,
---	--

IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.
---	--

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Invalidate Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue InvalidateAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Put Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be put in the data grid.

value

Type: [TValue](#)

Specifies the value to be put in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey, TValue PutAll Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts multiple key-value pairs to the data grid, replacing or adding a new entries to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsertAll method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to be put into the data grid.

Return Value

Exceptions

Exception	Condition
-----------	-----------

System ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
---	--

IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.
---	--

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Remove Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue RemoveAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, it will be ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Replace Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) is thrown during commit.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be updated.

value

Type: [TValue](#)

Specifies the value to be updated in the data grid and Loader.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ReplaceAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to replace in the data grid.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within entries.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Map.CacheKeyNotFoundException	Occurs when the key is not found in any of the cache tiers or Loader. This may be deferred until commit time.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue ResetToDefaults Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Resets the configurable settings for the map back to configured values.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Remarks

This method only resets configuration parameters that can be overridden by the client.
See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Touch Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to have last access time updated.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.Map.CacheKeyNotFoundException	Occurs if the key does not exist in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TouchAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entries specified in the keyList without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundException](#) is thrown.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the keys to have last access time updated.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList.
IBM.WebSphere.Caching.Map CacheKeyNotFoundException	Occurs if the key does not exist in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryAdd Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A variant of [Add\(TKey, TValue\)](#) that does not throw exceptions.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be inserted in the data grid.

value

Type: [TValue](#)

Specifies the value to be inserted in the data grid.

Return Value

Returns true if successful, false if unsuccessful.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryAddAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pairs to the data grid.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object of key-value pairs to be inserted into the data grid.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If entries is null or a null key is specified in a dictionary entry or a dictionary key in entries exists, false is returned. If all keys in the dictionary entries do not resolve to the same partition or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryInvalidate Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry associated with the specified key from the cache, leaving the data behind unchanged.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key for which you want to remove the value.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If key is null or the caller does not have authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryInvalidateAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries associated with the keys that are specified in the keyList from the cache, leaving the data behind unchanged.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

The list of keys whose values are to be removed.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If keyList is null or a null key is specified in a keyList entry, false is returned. If all keys in the keyList do not resolve to the same partition or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryPut Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pair to the grid. If an entry for the key exists in the data grid, the value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

specifies the key to be inserted into the data grid.

value

Type: [TValue](#)

specifies the value to be inserted into the data grid.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If the key is null, or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryPutAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds each key-value pair to the data grid. If an entry for a key exists in the data grid, its value is updated with the specified value. If an entry for a key does not exist in the data grid, the key-value pair is added to the data grid.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object that contains the key-value pairs to be added to the data grid.

Return Value

Returns true if successful, false if unsuccessful.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryRemove Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid. If the key has no matching entry in the map, no action is taken.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key for which you want to remove the entry.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If key is null or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryRemoveAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the list of entries associated with the keys that are specified in the keyList. If a key in the keyList has no matching entry in the map, no action is taken for that key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

The list of keys whose entries are to be removed.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If keyList is null or a null key is specified in a keyList entry, false is returned. If all keys in the keyList do not resolve to the same partition or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryReplace Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces the value of a key-value pair in the data grid. If an entry for the key exists in the data grid, its value is replaced with the specified value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key for the entry to be replaced.

value

Type: [TValue](#)

Specifies the new value.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If key is null or the key does not match an existing entry in the data grid, false is returned. If the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryReplaceAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces each key-value pair that is specified in the dictionary object in the data grid. If an entry for the key exists in the data grid, its value is replaced with the corresponding value specified.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

entries

Type: System.Collections.Generic IDictionary [TKey](#), [TValue](#)

Specifies a dictionary object that contains key-value pairs to be replaced in the data grid.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If entries is null, a null key is specified in a dictionary entry, or the key does not match an existing entry in the data grid, false is returned. If all keys in the dictionary entries do not resolve to the same partition, or the caller has insufficient authority to run this operation, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryTouch Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entry that matches the key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to have last access time updated.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If the key is null or the key does not match an existing entry in the data grid, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TryTouchAll Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entries that match each key in the keyList.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies a list of keys to have last access time updated.

Return Value

Returns true if successful, false if unsuccessful.

Remarks

If an entry is null or any key in the keyList is null, false is returned. If a key in the keyList does not match an existing entry in the data grid or all keys in the keyList do not resolve to the same partition, false is returned.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticAutoTx TKey, TValue](#) type exposes the following members.
Properties

Name	Description
Grid	Retrieves the IGrid instance associated with this map. (Inherited from IGridMap TKey, TValue .)
Item	Gets or sets the value in the map using the specified key.
LockTimeOut	Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.
Name	Retrieves the map name. (Inherited from IGridMap TKey, TValue .)
PartitionManager	Retrieves the IPartitionManager associated with this map. (Inherited from IGridMap TKey, TValue .)
TimeToLive	Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".
UseTtlEvictor	This property can only be used when the TtlEvictorType property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException is thrown.

To disable the TTL timeout, use a value of `TimeSpan.Zero`.

To revert the TTL value to the configured default, use a value of `TimeSpan.MinValue`.

[T](#)
[tl](#)
[E](#)
[vi](#)
[ct](#)
[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

Gets the [TtlType](#) of the map's TTL evictor.

[Back to Top](#)

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue Item Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the value in the map using the specified key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

The key of the value to set or get.

Field Value

The value associated with key to get or set.

Remarks

This indexer retrieves or puts the key and value into the map using the [Get\(TKey\)](#) and [Put\(TKey, TValue\)](#) methods respectively.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

[IGridMapPessimisticAutoTx TKey, TValue Get\(TKey\)](#)

[IGridMapPessimisticAutoTx TKey, TValue Put\(TKey, TValue\)](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey, TValue LockTimeout Property

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The maximum time to wait when acquiring a lock on an item in the grid map.

Exceptions

Exception	Condition
System.ArgumentException	Occurs if the value is not \geq Zero.

Remarks

To prevent deadlocks from occurring, the grid map has a default timeout value of 15 seconds; however, on a heavily loaded system, lock timeouts can occur without an actual deadlock. Use this property to increase the value from the default to prevent false lock timeout exceptions from occurring.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".

This property can only be used when the [TtlEvictorType](#) property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException is thrown.

To disable the TTL timeout, use a value of TimeSpan.Zero.

To revert the TTL value to the configured default, use a value of TimeSpan.MinValue.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Exceptions

Exception	Condition
System.ArgumentException	Occurs if the TimeSpan is not ≥ 0 or TimeSpan.MinValue or if the TtlEvictorType property is not LastAccessTime or LastUpdateTime.
Remarks	

Required Permission: MapPermission.INVALIDATE

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticAutoTx TKey,
TValue TtlEvictorType Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the [TtlType](#) of the map's TTL evictor.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The [TtlType](#) of the map's TTL evictor.

See Also

[IGridMapPessimisticAutoTx TKey, TValue Interface](#)

[IGridMapPessimisticAutoTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

This is a handle to a map using the pessimistic locking strategy and manual transaction demarcation. All data access operations must occur within an active transaction.

Use the [Transaction](#) property to access the [IGridTransaction](#) instance that is associated with this map instance, and use the [Begin](#) method to begin a transaction. All keys in a single transaction must resolve to the same partition.

An instance of this IGridMapPessimisticTx is not thread-safe, and can only be used by the thread at a time. Use the `Dispose` method when finished with the map, to improve performance.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Type Parameters

TKey

Generic type key.

TValue

Generic type value.

Remarks

Each data access method includes a "Specification details" table that includes the following information:

Required permission: The permission required to use the API.

Pessimistic read lock acquired: The type of lock that is acquired when you are using pessimistic locking with repeatable read transaction isolation.

Pessimistic read lock held: The type of lock that is held for the duration of the transaction with repeatable read transaction isolation. Locks can be upgraded but not demoted during a transaction.

Identifies the map cache tiers that are included when fetching or updating cache entries in the call and under what circumstances. The following tiers are available for IGridMapPessimisticTx maps:

Cache tier:

- Transactional Cache
- Server Cache (data grid)
- Loader (if enabled)

Examples

This sample demonstrates how to put a new cache entry into the data grid:

[Copy to ClipboardPrint](#)

```

// Assume we have already connected to the Grid...
IGrid grid = ...

// Retrieve a new map instance.
IGridMapPessimisticTx<long, string> map = grid.GetGridMapPessimisticTx<long, string>("MyPessimisticMap");

// Start a transaction.
map.Transaction.Begin();

try
{
    // Lock the entry in the data grid with an Upgradable lock.
    map.Lock(123, LockMode.Upgradable);

    // Put the entry in the transactional cache.
    map.Put(123, "Value to cache");

    // Commit the transaction to the data grid.
    map.Transaction.Commit();
}
catch(GridException)
{
    // Clean-up the transaction if the lock could not be
    // acquired, or the commit failed.
    if(map.Transaction.Active)
    {
        try
        {
            map.Transaction.Rollback();
        }
        catch(Exception)
        {
            // Optionally log this exception, or ignore the exception.
        }
    }

    // Dispose the map (optional, but it improves performance)
    map.Dispose();

    // Rethrow the real exception.
    throw;
}

```

See Also

[IGridMapPessimisticTx TKey, TValue Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticTx TKey, TValue](#) type exposes the following members.
Methods

Name	Description
	Adds the key-value pair to the data grid.
Add	The key must not exist before you run this method.
Add	A DuplicateKeyException exception results when the duplicate key is discovered. Duplicate keys might be discovered with a flush or commit operation. In this scenario, the exception is an inner exception.
	Adds multiple key-value pairs to the data grid.
Add	The keys must not exist before executing this method.
All	A DuplicateKeyException exception results when the duplicate key is discovered. This discovery might happen during a flush or commit operation. In this scenario, the exception is an inner exception.
Contains	Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a null key cannot be specified.
Key	This API does not hold any locks. Use the Lock(TKey, LockMode) to test for a key and retain a lock.
Contains	Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
Key	
All	This API does not hold any locks. Use the LockAll(IList TKey, LockMode) to test for a key and retain a lock.
Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
Get	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
Get	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that may have a null value.
All	Retrieves the values that are associated with the list of keys that are specified in the keyList. If the value is not found, a null value is returned.
Get	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for multiple keys that might have a null value.
All	Locks the specified key and retrieves the associated value. If the value is not found, a null is returned.
Get	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that might have a null value.
And	Locks the specified keys and retrieves the associated values. If a value is not found, a null is returned in the list.
All	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for a key that might have a null value.

invalidate	Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store). If the key cannot be found in the map, the operation is ignored.
invalidateAll	Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store). If a key cannot be found in the map, the operation is ignored.
lock	Locks the specified key and tests to see if the key was previously present in the data grid or Loader.
lockAll	Locks the specified keys and tests to see if each was previously present in the data grid or Loader. Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.
put	Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client. Puts multiple key-value pairs to the data grid, replacing or adding new entries to each data grid tier as needed.
putAll	Note: This method has the same specification as the ObjectMap.upsertAll method in the eXtreme Scale Java client.
remove	Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store). If the key cannot be found in the map, the operation is ignored.
removeAll	Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store). If a key cannot be found in the map, the operation is ignored.
replace	Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store). If the key cannot be found in the data grid a CacheKeyNotFoundException exception results during the commit operation.
replaceAll	Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store). If a key cannot be found in the map a CacheKeyNotFoundException exception results.
resetDefaults	Resets the configurable settings for the map back to configured values.
touch	Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value. If the key cannot be found in the map a CacheKeyNotFoundException exception results during the commit operation.
touchAll	Updates the last access time for the data grid entries that are specified in the keyList list without locking the entries or fetching the values. If a key cannot be found in the map a CacheKeyNotFoundException exception results during the commit operation.

[Back to Top](#)
Properties

N
a

Description

m
e
G
r
i
d
I
t
e
m
L
o
c
k
T
i
m
e
o
u
t
N
a
m
e
P
a
r
t
i
t
i
o
n
M
a
n
a
g
e
r
T
i
m
e
T
o
L
i
v
e
T
r
a
n
s
a
c

Retrieves the IGrid instance associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

An indexer that retrieves or puts the key and value into the map with the [Get\(TKey\)](#) and [Put\(TKey, TValue\)](#) methods.

Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

Retrieves the map name.
(Inherited from [IGridMap TKey, TValue](#).)

Retrieves the IPartitionManager associated with this map.
(Inherited from [IGridMap TKey, TValue](#).)

Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".

This property can be used only when the [TtlEvictorType](#) property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException exception results.

To disable the TTL timeout, use a value of TimeSpan.Zero.

To revert the TTL value to the configured default, use a value of TimeSpan.MinValue.

The Transaction instance used to configure and demarcate a transaction.
(Inherited from [ITransactionable](#).)

[io](#)
[n](#)
[T](#)
[tl](#)
[E](#)
[vi](#)
[ct](#)
[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

Retrieves the time to live type for the evictor on the map.

[Back to Top](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticTx TKey, TValue](#) type exposes the following members.
Methods

Name	Description
	Adds the key-value pair to the data grid.
Add	The key must not exist before you run this method.
Add	A DuplicateKeyException exception results when the duplicate key is discovered. Duplicate keys might be discovered with a flush or commit operation. In this scenario, the exception is an inner exception.
	Adds multiple key-value pairs to the data grid.
Add	The keys must not exist before executing this method.
All	A DuplicateKeyException exception results when the duplicate key is discovered. This discovery might happen during a flush or commit operation. In this scenario, the exception is an inner exception.
Contains	Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a null key cannot be specified.
Key	This API does not hold any locks. Use the Lock(TKey, LockMode) to test for a key and retain a lock.
Contains	Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.
All	This API does not hold any locks. Use the LockAll(IList TKey, LockMode) to test for a key and retain a lock.
Dispose	Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources. (Inherited from IDisposable.)
Get	Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.
Get	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that may have a null value.
All	Retrieves the values that are associated with the list of keys that are specified in the keyList. If the value is not found, a null value is returned.
All	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for multiple keys that might have a null value.
Get	Locks the specified key and retrieves the associated value. If the value is not found, a null is returned.
And	If the map supports null values, use the Lock(TKey, LockMode) or the ContainsKey(TKey) to test for a key that might have a null value.
Lock	Locks the specified keys and retrieves the associated values. If a value is not found, a null is returned in the list.
All	If the map supports null values, use the LockAll(IList TKey, LockMode) or the ContainsKeyAll(IList TKey) to test for a key that might have a null value.

[invalidate](#) Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

[invalidateAll](#) Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

[lock](#) Locks the specified key and tests to see if the key was previously present in the data grid or Loader.

[lockAll](#) Locks the specified keys and tests to see if each was previously present in the data grid or Loader.

Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.

[put](#) Note: This method has the same specification as the `ObjectMap.upsert` method in the eXtreme Scale Java client.

Puts multiple key-value pairs to the data grid, replacing or adding new entries to each data grid tier as needed.

[putAll](#) Note: This method has the same specification as the `ObjectMap.upsertAll` method in the eXtreme Scale Java client.

[remove](#) Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

[removeAll](#) Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

[replace](#) Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) exception results during the commit operation.

[replaceAll](#) Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) exception results.

[resetDefaults](#) Resets the configurable settings for the map back to configured values.

[touch](#) Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) exception results during the commit operation.

[touchAll](#) Updates the last access time for the data grid entries that are specified in the keyList list without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundException](#) exception results during the commit operation.

[Back to Top](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Add Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds the key-value pair to the data grid.

The key must not exist before you run this method.

A DuplicateKeyException exception results when the duplicate key is discovered. Duplicate keys might be discovered with a flush or commit operation. In this scenario, the exception is an inner exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be inserted in the data grid.

value

Type: [TValue](#)

Specifies the value to be inserted in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a nullkey is specified.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when the key exists.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Adds multiple key-value pairs to the data grid.

The keys must not exist before executing this method.

A DuplicateKeyException exception results when the duplicate key is discovered. This discovery might happen during a flush or commit operation. In this scenario, the exception is an inner exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: [IBM.WebSphere.Caching.IOrderedDictionary](#) [TKey](#), [TValue](#)

Specifies a [IOrderedDictionary](#) [TKey](#), [TValue](#) object of key-value pairs to be inserted into the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
IBM.WebSphere.Caching.Map.DuplicateKeyException	Occurs when a dictionary key in entries already exists in the map.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INSERT

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue ContainsKey Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for an entry that is associated with the specified the key. If an entry is found, true is returned. If an entry is not found, false is returned. Data grids do not support null key values, so a null key cannot be specified.

This API does not hold any locks. Use the [Lock\(TKey, LockMode\)](#) to test for a key and retain a lock.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key to test in the map.

Return Value

Returns true if the key is found, false otherwise.

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: No

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey, TValue ContainsKeyAll Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Looks in the map for the entries that are associated with the specified keys in the keyList. If an entry is located, true is returned. If an entry is not located, false is returned. Data grids do not support null key values, so a null key cannot be specified in a keyList.

This API does not hold any locks. Use the [LockAll\(IList TKey, LockMode\)](#) to test for a key and retain a lock.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies a list of keys to test in the map.

Return Value

Specifies a list of boolean values. If the key is found in the keyList, true is listed. Otherwise, false is returned.

Exceptions

Exception	Condition
-----------	-----------

System.ArgumentException	Occurs when keyList is null, or when a null key is specified in keyList.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
---	--

IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.
--	--

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: No

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Get Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the value that is associated with the specified key. If the value is not found, a null is returned.

If the map supports null values, use the [Lock\(TKey, LockMode\)](#) or the [ContainsKey\(TKey\)](#) to test for a key that may have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to query.

Return Value

The value that is associated with the specified key if it exists; otherwise null is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified, or when the data type of the value that was obtained from the data grid does not match the declared data type.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: Yes

Cache tier: Progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue GetAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the values that are associated with the list of keys that are specified in the keyList. If the value is not found, a null value is returned.

If the map supports null values, use the [LockAll\(IList TKey , LockMode\)](#) or the [ContainsKeyAll\(IList TKey \)](#) to test for multiple keys that might have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to query.

Return Value

A list of values that are associated with the supplied keys. If the value associated with a particular key is not in the data grid, null is returned in the list at the position that is associated with the key.

Exceptions

Exception	Condition
-----------	-----------

System ArgumentException	Occurs when the keyList is null, or when a null key is specified in keyList.
--------------------------	--

IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing,
---	--

IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.
---	--

Remarks

Specification details:

Required client permission: MapPermission.READ

Pessimistic locks acquired: LockMode.Shared

Pessimistic locks held: Yes

Cache tier: For each key, progresses to all tiers until the key is found.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey, TValue GetAndLock Method

IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified key and retrieves the associated value. If the value is not found, a null is returned.

If the map supports null values, use the [Lock\(TKey, LockMode\)](#) or the [ContainsKey\(TKey\)](#) to test for a key that might have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)

Specifies the type of lock to acquire.

Return Value

Specifies the value that is associated with the specified key if it exists. Otherwise, null is returned.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:

MapPermission.READ

Pessimistic locks acquired:

LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive

Pessimistic locks held: Yes

Cache tier:

Progresses to all tiers until the key is found and the appropriate lock is acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified keys and retrieves the associated values. If a value is not found, a null is returned in the list.

If the map supports null values, use the [LockAll\(IList TKey , LockMode\)](#) or the [ContainsKeyAll\(IList TKey \)](#) to test for a key that might have a null value.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)
Specifies the type of lock to acquire.

Return Value

A list of values that are associated with the supplied keys. If the value associated with a particular key is not in the data grid, null is returned in the list at the position that is associated with the key.

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:	MapPermission.READ
Pessimistic locks acquired:	LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive
Pessimistic locks held:	Yes
Cache tier:	Progresses to all tiers until the keys are found and the appropriate locks are acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)
[IGridMapPessimisticTx TKey, TValue Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Invalidate Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid, without affecting the Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue InvalidateAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid, without affecting the Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be invalidated from the data grid.

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.INVALIDATE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Lock Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified key and tests to see if the key was previously present in the data grid or Loader.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

Specifies the key to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)

Specifies the type of lock to acquire.

Return Value

Returns true if the key is found in the data grid or Loader (back-end persistent store).

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:

MapPermission.READ

Pessimistic locks acquired:

LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive

Pessimistic locks held: Yes

Cache tier:

Progresses to all tiers until the key is found and the appropriate lock is acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue LockAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Locks the specified keys and tests to see if each was previously present in the data grid or Loader.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)
Specifies the list of keys to lock.

lockMode

Type: [IBM.WebSphere.Caching.Map LockMode](#)
Specifies the type of lock to acquire.

Return Value

A list of bool that are associated with the supplied keys, where true indicates that the key was found in the data grid or Loader (back-end persistent store).

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission:	MapPermission.READ
Pessimistic locks acquired:	LockMode.Shared, LockMode.Upgradable or LockMode.Exclusive
Pessimistic locks held:	Yes
Cache tier:	Progresses to all tiers until the key is found and the appropriate lock is acquired.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional

information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Put Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts the specified key and value into the data grid, replacing or adding a new entry to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsert method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be put in the data grid.

value

Type: [TValue](#)

Specifies the value to be put in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue PutAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Puts multiple key-value pairs to the data grid, replacing or adding new entries to each data grid tier as needed.

Note: This method has the same specification as the ObjectMap.upsertAll method in the eXtreme Scale Java client.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

entries

Type: [IBM.WebSphere.Caching.IOrderedDictionary](#) TKey, TValue

Specifies a [IOrderedDictionary](#) TKey, TValue object of key-value pairs to be put into the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null entries is specified, or when a null key is specified in entries.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Remove Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entry that is associated with the specified key from the data grid and Loader (back-end persistent store).

If the key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue RemoveAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Removes the entries that are associated with the specified keyList from the data grid and Loader (back-end persistent store).

If a key cannot be found in the map, the operation is ignored.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the list of keys to be removed from the data grid and Loader

Return Value

Exceptions

Exception	Condition
System ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList..
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Security AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.REMOVE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Replace Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces the existing entry that is associated with the specified key with the specified value from the data grid and Loader (back-end persistent store).

If the key cannot be found in the data grid a [CacheKeyNotFoundException](#) exception results during the commit operation.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to be updated.

value

Type: [TValue](#)

Specifies the value to be updated in the data grid and Loader.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue ReplaceAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Replaces multiple key-value pairs into the data grid and Loader (back-end persistent store).

If a key cannot be found in the map a [CacheKeyNotFoundException](#) exception results.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

entries

Type: [IBM.WebSphere.Caching IOrderedDictionary TKey, TValue](#)

Specifies a [IOrderedDictionary TKey, TValue](#) object of key-value pairs to replace in the data grid.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within entries.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.Map.CacheKeyNotFoundException	Occurs when the key is not found in any of the cache tiers or Loader. This exception might be deferred until commit time.
IBM.WebSphere.Caching.Security.AccessControlException	Occurs when the caller has insufficient authority to perform this operation.

Remarks

Specification details:

Required client permission: MapPermission.WRITE

Cache tier: Applied to all tiers during commit.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue ResetToDefaults Method

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Resets the configurable settings for the map back to configured values.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Remarks

This method resets configuration parameters that can be overridden by the client only.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Touch Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entry that matches the key without locking the entry or fetching the value.

If the key cannot be found in the map a [CacheKeyNotFoundException](#) exception results during the commit operation.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key to have last access time updated.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null key is specified.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue TouchAll Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Updates the last access time for the data grid entries that are specified in the keyList list without locking the entries or fetching the values.

If a key cannot be found in the map a [CacheKeyNotFoundExcepcion](#) exception results during the commit operation.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies the keys to have last access time updated.

Return Value

Exceptions

Exception	Condition
System.ArgumentException	Occurs when a null keyList is specified, or when a null key is specified within keyList.
IBM.WebSphere.Caching.GridException	Occurs when an error occurs during processing.
IBM.WebSphere.Caching.SecurityAccessControlException	Occurs when the caller has insufficient authority to perform this operation.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IGridMapPessimisticTx TKey, TValue](#) type exposes the following members.
Properties

Name	Description
Grid	Retrieves the IGrid instance associated with this map. (Inherited from IGridMap TKey, TValue .)
Item	An indexer that retrieves or puts the key and value into the map with the Get(TKey) and Put(TKey, TValue) methods.
LockTimeout	Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.
Name	Retrieves the map name. (Inherited from IGridMap TKey, TValue .)
PartitionManager	Retrieves the IPartitionManager associated with this map. (Inherited from IGridMap TKey, TValue .)
TimeToLive	Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".
TtlEvictorType	This property can be used only when the TtlEvictorType property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException exception results.

To disable the TTL timeout, use a value of `TimeSpan.Zero`.

To revert the TTL value to the configured default, use a value of `TimeSpan.MinValue`.

[T](#)
[r](#)
[a](#)
[n](#)
[s](#)
[a](#)
[c](#)
[t](#)
[i](#)
[o](#)
[n](#)
[T](#)
[t](#)
[l](#)
[E](#)
[v](#)
[i](#)
[c](#)
[t](#)
[o](#)
[r](#)
[T](#)
[y](#)
[p](#)
[e](#)

The Transaction instance used to configure and demarcate a transaction.
(Inherited from [ITransactionable](#).)

Retrieves the time to live type for the evictor on the map.

[Back to Top](#)
See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue Item Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An indexer that retrieves or puts the key and value into the map with the [Get\(TKey\)](#) and [Put\(TKey, TValue\)](#) methods.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

key

Type: [TKey](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue LockTimeout Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the maximum time to wait when acquiring a lock on an item in the grid map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Field Value

The maximum time to wait when acquiring a lock on an item in the grid map.

Exceptions

Exception	Condition
System.ArgumentException	Occurs if the value is not \geq Zero.

Remarks

To prevent deadlocks from occurring, the grid map has a default timeout value of 15 seconds; however, on a heavily loaded system, lock timeouts can occur without an actual deadlock. Use this property to increase the value from the default to prevent false lock timeout exceptions from occurring.

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Establishes the interval of time that any given cache entry can live for, which is referred to as "time to live" or TTL. Setting a new TTL value affects cache entries that are accessed after this method call occurs. It does not affect any cache entry that was created or accessed prior to this method call. By calling this method on this IGridMapPessimisticTx, any previous value set by the IGridMapPessimisticTx.TimeToLive property is overridden for this map instance. If this method is never called on the map, the default setting is used. The default setting is to retain the time-to-live value for any existing map entry and to use the default value from map configuration setting if a new map entry is being created. If TTL is never set on the map configuration, the cache entry can live "forever".

This property can be used only when the [TtlEvictorType](#) property is set to LastAccessTime or LastUpdateTime on the map configuration. If this method is called on the IGridMapPessimisticTx and the TtlEvictorType is something other than LastAccessTime or LastUpdateTime, an ArgumentException exception results.

To disable the TTL timeout, use a value of TimeSpan.Zero.

To revert the TTL value to the configured default, use a value of TimeSpan.MinValue.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Exceptions

Exception	Condition
System.ArgumentException	Occurs if the TimeSpan is not ≥ 0 or TimeSpan.MinValue or if the TtlEvictorType property is not LastAccessTime or LastUpdateTime.
Remarks	

Required Permission: MapPermission.INVALIDATE

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IGridMapPessimisticTx TKey,
TValue TtlEvictorType Property

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the time to live type for the evictor on the map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IGridMapPessimisticTx TKey, TValue Interface](#)

[IGridMapPessimisticTx TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey, TValue
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An IPartitionManager provides properties and methods for determining how partitions are calculated. Retrieve an IPartitionManager from an [IGridMap TKey, TValue](#).

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Type Parameters

TKey

Specifies a generic type key.

TValue

Specifies a generic type value.

Remarks

The partition id is calculated as follows:

- The key class is examined for the [PartitionKeyAttribute](#) attribute. If present, the GetHashCode of the referenced attributes is used to calculate the partition.
- Otherwise, the key's GetHashCode method is used to calculate the partition id.

See Also

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)




IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)


IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IPartitionManager TKey, TValue](#) type exposes the following members.
Methods

Name	Description
 GetPartition	Returns the partition number for the specified map element key.
 GetPartitionAll(IList TKey)	Returns a map of partition numbers to map keys from a collection of map element keys.
 GetPartitionAll(IOrderedDictionary TKey, TValue)	Returns a map of partition numbers to map keys from a collection of map element keys.

[Back to Top](#)

Properties

Name	Description
 NumPartitions	Retrieves the number of partitions that are defined for the map.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IPartitionManager TKey, TValue](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> GetPartition	Returns the partition number for the specified map element key.
<input type="checkbox"/> GetPartitionAll(IList TKey)	Returns a map of partition numbers to map keys from a collection of map element keys.
<input type="checkbox"/> GetPartitionAll(IOrderedDictionary TKey, TValue)	Returns a map of partition numbers to map keys from a collection of map element keys.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey,
TValue GetPartition Method

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns the partition number for the specified map element key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: [TKey](#)

Specifies the key for which to return a partition number.

Return Value

Returns the partition number for the specified map element key.

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GetPartitionAll(IList TKey)	Returns a map of partition numbers to map keys from a collection of map element keys.
<input type="checkbox"/> GetPartitionAll(IOrderedDictionary TKey, TValue)	Returns a map of partition numbers to map keys from a collection of map element keys.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey,
TValue GetPartitionAll Method (IList TKey)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map of partition numbers to map keys from a collection of map element keys.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

keyList

Type: System.Collections.Generic IList [TKey](#)

Specifies a list of keys for which you want to return partition numbers.

Return Value

Returns a collection of partition numbers and associated keys.

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[GetPartitionAll Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey, TValue GetPartitionAll
Method (IOrderedDictionary TKey, TValue)

IBM WebSphere™ eXtreme Scale
Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns a map of partition numbers to map keys from a collection of map element keys.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

entries

Type: [IBM.WebSphere.Caching.IOrderedDictionary TKey, TValue](#)

Specifies a dictionary object of key-value pairs for which you want to return partition numbers.

Return Value

Specifies a collection of partition numbers and associated key-value pairs.

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[GetPartitionAll Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.


IPartitionManager TKey, TValue
Properties

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [IPartitionManager TKey, TValue](#) type exposes the following members.
Properties

Name	Description
 NumPartitions	Retrieves the number of partitions that are defined for the map.

[Back to Top](#)

See Also

[IPartitionManager TKey, TValue Interface](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

IPartitionManager TKey,
TValue NumPartitions Property

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Retrieves the number of partitions that are defined for the map.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[IPartitionManager TKey, TValue Interface](#)

[IPartitionManager TKey, TValue Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LoaderException exception is the base exception that results for any exceptions that are encountered by a Loader.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map LoaderException

[IBM.WebSphere.Caching.Map ClientServerLoaderException](#)

[IBM.WebSphere.Caching.Map UnavailableServiceException](#)

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LoaderException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LoaderException	Initializes a new instance of the LoaderException class.
<input type="checkbox"/> LoaderException(Ex ception)	Initializes a new instance of the LoaderException class with a specified exception cause.
<input type="checkbox"/> LoaderException(St ring)	Initializes a new instance of the LoaderException class with the specified error message.
<input type="checkbox"/> LoaderException(St ring, Exception)	Initializes a new instance of the LoaderException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBase Exception	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjec tData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> Member wiseClon e	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LoaderException	Initializes a new instance of the LoaderException class.
<input type="checkbox"/> LoaderException(Exception)	Initializes a new instance of the LoaderException class with a specified exception cause.
<input type="checkbox"/> LoaderException(String)	Initializes a new instance of the LoaderException class with the specified error message.
<input type="checkbox"/> LoaderException(String, Exception)	Initializes a new instance of the LoaderException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LoaderException Constructor (String, IBM WebSphere™ eXtreme Scale Client for .NET API Exception) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LoaderException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LoaderException Class](#)

[LoaderException Members](#)

[LoaderException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LoaderException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LoaderException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LoaderException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockStrategyNotSupportedException exception occurs when the lock manager detects a deadlock. This exception occurs to prevent the deadlock.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map.LockException](#)

[IBM.WebSphere.Caching.Map.LockTimeoutException](#)

IBM.WebSphere.Caching.Map.LockDeadlockException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Remarks

Typically, a deadlock is the result of the following scenario: One transaction gets a map entry, resulting in a weaker lock than an existing lock on the same entry. At commit time, the transaction attempts to promote the weaker lock to a stronger lock to apply the changes to the data store. For example, two transactions try to promote from shared locks to exclusive locks, but each transaction already owns a shared lock.

See Also

[LockDeadlockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockDeadlockException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LockDeadlockException	Initializes a new instance of the LockDeadlockException class.
<input type="checkbox"/> LockDeadlockException(Exception)	Initializes a new instance of the LockDeadlockException class with the specified exception cause.
<input type="checkbox"/> LockDeadlockException(String)	Initializes a new instance of the LockDeadlockException class with the specified error message.
<input type="checkbox"/> LockDeadlockException(String, Exception)	Initializes a new instance of the LockDeadlockException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor. (Inherited from LockTimeoutException .)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception.

(Inherited from Exception.)

InnerException Gets the Exception instance that caused the current exception.
(Inherited from Exception.)

[LockRequestQueueDetails](#) Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.
(Inherited from [LockTimeoutException](#).)

Message Gets a message that describes the current exception.
(Inherited from Exception.)

Source Gets or sets the name of the application or the object that causes the error.
(Inherited from Exception.)

StackTrace Gets a string representation of the frames on the call stack at the time the current exception was thrown.
(Inherited from Exception.)

TargetSite Gets the method that throws the current exception.
(Inherited from Exception.)

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LockDeadlockExcepti on	Initializes a new instance of the LockDeadlockException class.
<input type="checkbox"/> LockDeadlockExcepti on(Exception)	Initializes a new instance of the LockDeadlockException class with the specified exception cause.
<input type="checkbox"/> LockDeadlockExcepti on(String)	Initializes a new instance of the LockDeadlockException class with the specified error message.
<input type="checkbox"/> LockDeadlockExcepti on(String, Exception)	Initializes a new instance of the LockDeadlockException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockDeadlockException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockDeadlockException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockDeadlockException Class](#)

[LockDeadlockException Members](#)

[LockDeadlockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockDeadlockException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor. (Inherited from LockTimeoutException .)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockDeadlockException](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/> LockRequestQueueDetails	Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred. (Inherited from LockTimeoutException .)
<input type="checkbox"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockDeadlockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockException exception indicates errors with locking operations.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map LockException

[IBM.WebSphere.Caching.Map LockTimeoutException](#)

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [LockException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LockException	Initializes a new instance of the LockException class.
<input type="checkbox"/> LockException(Exception)	Initializes a new instance of the LockException class with a specified exception cause.
<input type="checkbox"/> LockException(String)	Initializes a new instance of the LockException class with the specified error message.
<input type="checkbox"/> LockException(String, Exception)	Initializes a new instance of the LockException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LockException	Initializes a new instance of the LockException class.
<input type="checkbox"/> LockException(Exception)	Initializes a new instance of the LockException class with a specified exception cause.
<input type="checkbox"/> LockException(String)	Initializes a new instance of the LockException class with the specified error message.
<input type="checkbox"/> LockException(String, Exception)	Initializes a new instance of the LockException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LockException Class](#)

[LockException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

The error message that explains the reason for the exception.

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockException Class](#)

[LockException Members](#)

[LockException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

The [LockException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [LockException](#) type exposes the following members.

Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies the strength of a lock to acquire.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Members

Member name	Value	Description
Shared	0	Shared lock that can be obtained if no one is currently holding an exclusive lock.
Upgradable	1	Upgradable lock that can be obtained if no one is currently holding an upgradable or exclusive lock.
Exclusive	2	Exclusive lock that can only be obtained if no locks are held.

See Also

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedExceptio IBM WebSphere™ eXtreme Scale Client for .NET API
n Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockStrategyNotSupportedException exception occurs if a map is configured with an unsupported lock strategy.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map LockStrategyNotSupportedException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockStrategyNotSupportedException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockStrategyNotSupportedException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> LockStrategyNotSupportedException	Initializes a new instance of the LockStrategyNotSupportedException class.
<input type="checkbox"/> LockStrategyNotSupportedException(Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with a specified exception cause.
<input type="checkbox"/> LockStrategyNotSupportedException(String)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message.
<input type="checkbox"/> LockStrategyNotSupportedException(String, Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockStrategyNotSupportedException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

Overload List

Name	Description
<input type="checkbox"/> LockStrategyNotSupportedException	Initializes a new instance of the LockStrategyNotSupportedException class.
<input type="checkbox"/> LockStrategyNotSupportedException(Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with a specified exception cause.
<input type="checkbox"/> LockStrategyNotSupportedException(String)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message.
<input type="checkbox"/> LockStrategyNotSupportedException(String, Exception)	Initializes a new instance of the LockStrategyNotSupportedException class with the specified error message and exception cause.

[Back to Top](#)

See Also

-
- [LockStrategyNotSupportedException Class](#)
 - [LockStrategyNotSupportedException Members](#)
 - [IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockStrategyNotSupportedException
Constructor (String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockStrategyNotSupportedException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockStrategyNotSupportedException Class](#)

[LockStrategyNotSupportedException Members](#)

[LockStrategyNotSupportedException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockStrategyNotSupportedException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockStrategyNotSupportedException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockStrategyNotSupportedException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockStrategyNotSupportedException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A LockTimeoutException exception occurs when the lock manager detects that the lock wait time exceeded the maximum wait time. The timeout might be the result of a deadlock. If a deadlock is causing the timeout, the timeout is used to break the deadlock.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map.LockException](#)

IBM.WebSphere.Caching.Map.LockTimeoutException

[IBM.WebSphere.Caching.Map.LockDeadlockException](#)

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockTimeoutException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> LockTimeoutException	Initializes a new instance of the LockTimeoutException class.
<input type="checkbox"/> LockTimeoutException(Exception)	Initializes a new instance of the LockTimeoutException class with a specified exception cause.
<input type="checkbox"/> LockTimeoutException(String)	Initializes a new instance of the LockTimeoutException class with the specified error message.
<input type="checkbox"/> LockTimeoutException(String, Exception)	Initializes a new instance of the LockTimeoutException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

<input type="checkbox"/>	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>	LockRequestQueueDetails	Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.
<input type="checkbox"/>	Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> LockTimeoutException	Initializes a new instance of the LockTimeoutException class.
<input type="checkbox"/> LockTimeoutException (Exception)	Initializes a new instance of the LockTimeoutException class with a specified exception cause.
<input type="checkbox"/> LockTimeoutException (String)	Initializes a new instance of the LockTimeoutException class with the specified error message.
<input type="checkbox"/> LockTimeoutException (String, Exception)	Initializes a new instance of the LockTimeoutException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [LockTimeoutException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[LockTimeoutException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockTimeoutException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getMessage	Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException getMessage IBM WebSphere™ eXtreme Scale Client for .NET API
Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Returns the detail message string of this exception. The returned String value includes the request queue details and the message that was provided to the constructor.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Return Value

Specifies the detailed message string of this [LockTimeoutException](#) instance.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [LockTimeoutException](#) type exposes the following members.
Properties

Name	Description
<input type="text"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="text"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="text"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="text"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="text"/> LockRequestQueueDetails	Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.
<input type="text"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[LockTimeoutException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

LockTimeoutException LockRequestQueue IBM WebSphere™ eXtreme Scale Client for .NET
Details Property API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets detailed information about the state of the lock on the lock request queue at the time the lock timeout occurred.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

Gets the value that was set in the LockRequestQueueDetails property. If the LockRequestQueueDetails property was not previously set for this exception, the return value is null.

Sets the details of the lock requests on the lock request queue at the time the lock timeout occurred.

See Also

[LockTimeoutException Class](#)

[LockTimeoutException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A MultiplePartitionWriteException exception is a base exception for client/server operations when a user attempts to write to multiple remote partitions on remote servers in the same transaction.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map MultiplePartitionWriteException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MultiplePartitionWriteException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MultiplePartitionWriteException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> MultiplePartitionWriteException	Initializes a new instance of the MultiplePartitionWriteException class.
<input type="checkbox"/> MultiplePartitionWriteException(Exception)	Initializes a new instance of the MultiplePartitionWriteException class with a specified exception cause.
<input type="checkbox"/> MultiplePartitionWriteException(String)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message.
<input type="checkbox"/> MultiplePartitionWriteException(String, Exception)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc ption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> MultiplePartitionWriteException	Initializes a new instance of the MultiplePartitionWriteException class.
<input type="checkbox"/> MultiplePartitionWriteException(Exception)	Initializes a new instance of the MultiplePartitionWriteException class with a specified exception cause.
<input type="checkbox"/> MultiplePartitionWriteException(String)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message.
<input type="checkbox"/> MultiplePartitionWriteException(String, Exception)	Initializes a new instance of the MultiplePartitionWriteException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)
[MultiplePartitionWriteException Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

MultiplePartitionWriteException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [MultiplePartitionWriteException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[MultiplePartitionWriteException Class](#)

[MultiplePartitionWriteException Members](#)

[MultiplePartitionWriteException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MultiplePartitionWriteException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)








IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [MultiplePartitionWriteException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[MultiplePartitionWriteException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An OptimisticCollisionException occurs when an optimistic locking strategy is used and more than one update transaction collides on the same map entry of an ObjectGrid instance. The first transaction to commit updates the version object for the map entry. Other transactions that read this same map entry before committing have the previous version object. When the other transactions try to commit, the version object that is read does not match the version that was last committed. Therefore, other transactions are prevented from updating a map entry with stale data.

The default OptimisticCallback plug-in is used by the run time if an implementation is not provided by the application. If a well-constructed equals(Object) method is not on your value object, this exception occurs because the entire value object is used as the version object.

Because this exception indicates that the map entry contains stale data, stale map entries or entries as identified by the key parameter that is passed to the OptimisticCollisionException(String, String, String, Object) method are invalidated. If this exception is thrown by a Loader plug-in and a null reference is used as the key parameter by the loader, the run time assumes that the loader does not know which entry caused the exception. In this scenario, the LogSequence object is passed to the Loader.batchUpdate(TxID, LogSequence) method to determine which map entries to invalidate. Each LogElement entry in the LogSequence object that is type update or delete is invalidated.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map OptimisticCollisionException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification



The [OptimisticCollisionException](#) type exposes the following members.
Constructors






Name	Description
<input type="checkbox"/> OptimisticCollisionException	Initializes a new instance of the OptimisticCollisionException class with the specified error message, data grid name, map name, and keys that caused the exception.

[Back to Top](#)
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> getGridName	Gets the name of the ObjectGrid instance in which the optimistic collision occurred.
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getKey	Gets key object or array of key objects that caused the OptimisticCollisionException to occur.
<input type="checkbox"/> getMapName	Gets the map name in which the OptimisticCollisionException exception occurred.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> setKey	Set the key that caused this exception to occur.
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)
Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc ption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[OptimisticCollisionException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [OptimisticCollisionException](#) class with the specified error message, data grid name, map name, and keys that caused the exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

gridName

Type: System String

Specifies the name of the data grid to which the map belongs.

mapName

Type: System String

Specifies the name of the map.

key

Type: System Object

Specifies the key or array of keys that caused the optimistic collision exception to occur.

Remarks

If more than a single key caused the exception, use an array object for this parameter. Each array element identifies a single map entry that caused the exception to occur. Using array elements is useful when a Loader uses the batch update support of a Java Database Connectivity (JDBC) driver. Pass a null reference if you are unable to determine which key or set of keys caused this exception to occur.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OptimisticCollisionException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> getGridName	Gets the name of the ObjectGrid instance in which the optimistic collision occurred.
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> getKey	Gets key object or array of key objects that caused the OptimisticCollisionException to occur.
<input type="checkbox"/> getMapName	Gets the map name in which the OptimisticCollisionException exception occurred.
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> setKey	Set the key that caused this exception to occur.
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[OptimisticCollisionException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException GetGridName IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the name of the ObjectGrid instance in which the optimistic collision occurred.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies the name of the ObjectGrid instance.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException getKey IBM WebSphere™ eXtreme Scale Client for .NET API Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets key object or array of key objects that caused the [OptimisticCollisionException](#) to occur.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies the key object or array of key objects that caused the exception to occur.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException getMapName IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets the map name in which the [OptimisticCollisionException](#) exception occurred.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Specifies the map name where the exception occurred.

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

OptimisticCollisionException setKey IBM WebSphere™ eXtreme Scale Client for .NET API
Method Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Set the key that caused this exception to occur.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

key

Type: System Object

the key or array of key objects that caused the exception

See Also

[OptimisticCollisionException Class](#)

[OptimisticCollisionException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [OptimisticCollisionException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[OptimisticCollisionException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies one or more attributes to use to calculate the partition hash code.

The PartitionKey attribute can be specified for the class using a path syntax to identify a single attribute. Multiple attributes can be specified using additional PartitionKey annotations on each field, using the [Order](#) attribute to specify the order in which the hash codes will be calculated.

The PartitionKey attribute is not inheritable.

Inheritance Hierarchy

System Object

System Attribute

IBM.WebSphere.Caching.Map PartitionKeyAttribute

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Remarks

The following examples illustrate how to identify a top-level, primitive field:

Simple class attribute example:

[Copy to ClipboardPrint](#)

```
[PartitionKey("deptId")] class Employee { int empId; int deptId; }
```

Simple field attribute example:

[Copy to ClipboardPrint](#)

```
class Employee { int empId; [PartitionKey] int deptId; }
```

Simple, multiple field attribute example:

[Copy to ClipboardPrint](#)

```
class Employee { int empId; [PartitionKey(order=0)] int deptId; [PartitionKey(order=1)] String coun
```

The following examples illustrate how to address an attribute inside an embedded class. In this case, the path separator is defined in the data grid as a "." character (the default for eXtreme Data Format):

Embedded class annotation example:

[Copy to ClipboardPrint](#)

```
[PartitionKey("deptKey.id")] class Employee { DepartmentKey deptKey; } class DepartmentKey { int id
```

Embedded field annotation example:

[Copy to Clipboard](#)[Print](#)

```
class Employee { [PartitionKey("id")] DepartmentKey deptKey; } class DepartmentKey { int id; }
```

See Also

[PartitionKeyAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [PartitionKeyAttribute](#) type exposes the following members.

Constructors

Name	Description
PartitionKeyAttribute	An attribute-less constructor for use when defined on a field, where the field name is the attribute.
PartitionKeyAttribute(String)	Specifies a single attribute to use to calculate the partition hash code.

[Back to Top](#)

Methods

Name	Description
Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
GetType	Gets the Type of the current instance. (Inherited from Object.)
IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

Properties

Name	Description
AttributeName	Identifies the path to the attribute that should be included as part of the partition key.
Order	Order of the fields that contribute to the partition calculation. The order values must be unique for all PartitionKey attributes defined on a key class.
TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> PartitionKeyAttribute	An attribute-less constructor for use when defined on a field, where the field name is the attribute.
<input type="checkbox"/> PartitionKeyAttribute(String)	Specifies a single attribute to use to calculate the partition hash code.

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An attribute-less constructor for use when defined on a field, where the field name is the attribute.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[PartitionKeyAttribute Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies a single attribute to use to calculate the partition hash code.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

attributeName

Type: System.String

The attribute name.

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[PartitionKeyAttribute Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [PartitionKeyAttribute](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Returns a value that indicates whether this instance is equal to a specified object. (Inherited from Attribute.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetHashCode	Returns the hash code for this instance. (Inherited from Attribute.)
<input type="checkbox"/> GetType	Gets the Type of the current instance. (Inherited from Object.)
<input type="checkbox"/> IsDefaultAttribute	When overridden in a derived class, indicates whether the value of this instance is the default value for the derived class. (Inherited from Attribute.)
<input type="checkbox"/> Match	When overridden in a derived class, returns a value that indicates whether this instance equals a specified object. (Inherited from Attribute.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Returns a String that represents the current Object. (Inherited from Object.)

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [PartitionKeyAttribute](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> AttributeName	Identifies the path to the attribute that should be included as part of the partition key.
<input type="checkbox"/> Order	Order of the fields that contribute to the partition calculation. The order values must be unique for all PartitionKey attributes defined on a key class.
<input type="checkbox"/> TypeId	When implemented in a derived class, gets a unique identifier for this Attribute. (Inherited from Attribute.)

[Back to Top](#)

See Also

[PartitionKeyAttribute Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute AttributeName IBM WebSphere™ eXtreme Scale Client for .NET API
Property Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Identifies the path to the attribute that should be included as part of the partition key.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

The attribute path.

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

PartitionKeyAttribute Order Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Order of the fields that contribute to the partition calculation. The order values must be unique for all PartitionKey attributes defined on a key class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Field Value

The field order.

See Also

[PartitionKeyAttribute Class](#)

[PartitionKeyAttribute Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReadOnlyException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A ReadOnlyException exception occurs when a modify operation is attempted on a read-only map.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map ReadOnlyException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ReadOnlyException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ReadOnlyException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> ReadOnlyException	Initializes a new instance of the ReadOnlyException > class.
<input type="checkbox"/> ReadOnlyException(Exception)	Initializes a new instance of the ReadOnlyException class with a specified exception cause.
<input type="checkbox"/> ReadOnlyException(String)	Initializes a new instance of the ReadOnlyException class with the specified error message.
<input type="checkbox"/> ReadOnlyException(String, Exception)	Initializes a new instance of the ReadOnlyException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)

	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReadOnlyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> ReadOnlyException	Initializes a new instance of the ReadOnlyException > class.
<input type="checkbox"/> ReadOnlyException(Exception)	Initializes a new instance of the ReadOnlyException class with a specified exception cause.
<input type="checkbox"/> ReadOnlyException(String)	Initializes a new instance of the ReadOnlyException class with the specified error message.
<input type="checkbox"/> ReadOnlyException(String, Exception)	Initializes a new instance of the ReadOnlyException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ReadOnlyException Class](#)

[ReadOnlyException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReadOnlyException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReadOnlyException](#)> class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ReadOnlyException Class](#)

[ReadOnlyException Members](#)

[ReadOnlyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReadOnlyException Constructor
(Exception)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReadOnlyException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ReadOnlyException Class](#)

[ReadOnlyException Members](#)

[ReadOnlyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReadOnlyException Constructor (String) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReadOnlyException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[ReadOnlyException Class](#)

[ReadOnlyException Members](#)

[ReadOnlyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ReadOnlyException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ReadOnlyException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ReadOnlyException Class](#)

[ReadOnlyException Members](#)

[ReadOnlyException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ReadOnlyException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReadOnlyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ReadOnlyException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ReadOnlyException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TargetNotAvailableException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TargetNotAvailableException occurs when a remote target was not found or was not reachable.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map TargetNotAvailableException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TargetNotAvailableException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TargetNotAvailableException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> TargetNotAvailableException(String)	Initializes a new instance of the TargetNotAvailableException class with the specified error message.
<input type="checkbox"/> TargetNotAvailableException(String, Exception)	Initializes a new instance of the TargetNotAvailableException class with the specified error message and exception cause.





[Back to Top](#)




Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Gets or sets the name of the application or the object that causes the

	Source	error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TargetNotAvailableException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TargetNotAvailableException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TargetNotAvailableException(String)	Initializes a new instance of the TargetNotAvailableException class with the specified error message.
<input type="checkbox"/> TargetNotAvailableException(String, Exception)	Initializes a new instance of the TargetNotAvailableException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[TargetNotAvailableException Class](#)

[TargetNotAvailableException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TargetNotAvailableException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TargetNotAvailableException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

The error message that explains the reason for the exception.

See Also

[TargetNotAvailableException Class](#)

[TargetNotAvailableException Members](#)

[TargetNotAvailableException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TargetNotAvailableException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TargetNotAvailableException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TargetNotAvailableException Class](#)

[TargetNotAvailableException Members](#)

[TargetNotAvailableException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TargetNotAvailableException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)





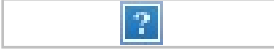


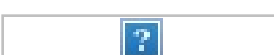
See Also

[TargetNotAvailableException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TargetNotAvailableException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TargetNotAvailableException Class](#)[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAffinityException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A TransactionAffinityException exception occurs during server failover for inflight transactions. Applications can try the transaction again.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map.LoaderException](#)

[IBM.WebSphere.Caching.Map.UnavailableServiceException](#)

IBM.WebSphere.Caching.Map.TransactionAffinityException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionAffinityException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAffinityException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> TransactionAffinityException	Initializes a new instance of the TransactionAffinityException class.
<input type="checkbox"/> TransactionAffinityException(Exception)	Initializes a new instance of the TransactionAffinityException class with a specified exception cause.
<input type="checkbox"/> TransactionAffinityException(String)	Initializes a new instance of the TransactionAffinityException class with the specified error message.
<input type="checkbox"/> TransactionAffinityException(String, Exception)	Initializes a new instance of the TransactionAffinityException class with the specified error message and exception cause.



[Back to Top](#)







Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Replication Group	Gets or sets the replication group ID for this exception. (Inherited from UnavailableServiceException .)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionAffinityException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> TransactionAffinityException	Initializes a new instance of the TransactionAffinityException class.
<input type="checkbox"/> TransactionAffinityException(Exception)	Initializes a new instance of the TransactionAffinityException class with a specified exception cause.
<input type="checkbox"/> TransactionAffinityException(String)	Initializes a new instance of the TransactionAffinityException class with the specified error message.
<input type="checkbox"/> TransactionAffinityException(String, Exception)	Initializes a new instance of the TransactionAffinityException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[TransactionAffinityException Class](#)
[TransactionAffinityException Members](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAffinityException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAffinityException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[TransactionAffinityException Class](#)

[TransactionAffinityException Members](#)

[TransactionAffinityException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAffinityException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAffinityException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionAffinityException Class](#)

[TransactionAffinityException Members](#)

[TransactionAffinityException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAffinityException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAffinityException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[TransactionAffinityException Class](#)

[TransactionAffinityException Members](#)

[TransactionAffinityException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

TransactionAffinityException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [TransactionAffinityException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[TransactionAffinityException Class](#)

[TransactionAffinityException Members](#)

[TransactionAffinityException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAffinityException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)










See Also

[TransactionAffinityException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [TransactionAffinityException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Replication Group	Gets or sets the replication group ID for this exception. (Inherited from UnavailableServiceException .)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[TransactionAffinityException Class](#)
[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Every grid map has an optional, built in timed based evictor that is referred to as the "time to live" evictor or TTL evictor. Each grid map entry has an expiration time that determines how long the entry is allowed to live in the grid map. When the expiration time is reached, the TTL evictor causes the expired entry to be evicted from the grid map. This enum defines the TTLType value constants that determine how the the expiration time is computed for a map entry.

Namespace: [IBM.WebSphere.Caching.Map](#)**Assembly:** Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Members

Member name	Value	Description
None	0	Indicates an entry has no expiration time and is allowed to live in the BackingMap until the application explicitly removes or invalidates the entry or a user defined evictor evicts it.
CreationTime	1	Indicates an entry expiration time is the sum of the creation time of the entry plus the "time to live" value.
LastAccessTime	2	Indicates an entry expiration time is the sum of the last access time of the entry, the time the entry was updated or read, plus the "time to live" value.
LastUpdateTime	3	Indicates an entry expiration time is the sum of the last update time of the entry plus the "time to live" value

See Also

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UnavailableServiceException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An UnavailableServiceException exception occurs when all servers are not running, or when all services are not available even though servers are running.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

[IBM.WebSphere.Caching.Map.LoaderException](#)

IBM.WebSphere.Caching.Map.UnavailableServiceException

[IBM.WebSphere.Caching.Map.TransactionAffinityException](#)

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)
[Print](#)

See Also

[UnavailableServiceException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [UnavailableServiceException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> UnavailableServiceException	Initializes a new instance of the UnavailableServiceException class.
<input type="checkbox"/> UnavailableServiceException(Exception)	Initializes a new instance of the UnavailableServiceException class with a specified exception cause.
<input type="checkbox"/> UnavailableServiceException(String)	Initializes a new instance of the UnavailableServiceException class with the specified error message.
<input type="checkbox"/> UnavailableServiceException(String, Exception)	Initializes a new instance of the UnavailableServiceException class with the specified error message and exception cause.



[Back to Top](#)







Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Replication Group	Gets or sets the replication group ID for this exception.
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[UnavailableServiceException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> UnavailableServiceException	Initializes a new instance of the UnavailableServiceException class.
<input type="checkbox"/> UnavailableServiceException(Exception)	Initializes a new instance of the UnavailableServiceException class with a specified exception cause.
<input type="checkbox"/> UnavailableServiceException(String)	Initializes a new instance of the UnavailableServiceException class with the specified error message.
<input type="checkbox"/> UnavailableServiceException(String, Exception)	Initializes a new instance of the UnavailableServiceException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[UnavailableServiceException Class](#)

[UnavailableServiceException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UnavailableServiceException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UnavailableServiceException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[UnavailableServiceException Class](#)

[UnavailableServiceException Members](#)

[UnavailableServiceException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UnavailableServiceException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UnavailableServiceException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[UnavailableServiceException Class](#)

[UnavailableServiceException Members](#)

[UnavailableServiceException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UnavailableServiceException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UnavailableServiceException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[UnavailableServiceException Class](#)

[UnavailableServiceException Members](#)

[UnavailableServiceException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UnavailableServiceException Constructor (String, Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UnavailableServiceException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[UnavailableServiceException Class](#)

[UnavailableServiceException Members](#)

[UnavailableServiceException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [UnavailableServiceException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[UnavailableServiceException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)










IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [UnavailableServiceException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Replication Group	Gets or sets the replication group ID for this exception.
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[UnavailableServiceException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UnavailableServiceException Replication Group Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Gets or sets the replication group ID for this exception.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Remarks

Gets the value that was set in the ReplicationGroup ID property, or null if the ReplicationGroup property was not previously set for this exception.

Sets the the ReplicationGroup ID for this exception.

See Also

[UnavailableServiceException Class](#)

[UnavailableServiceException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UndefinedMapException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An UndefinedMapException exception occurs to indicate that the map that an application tried to access is not defined in the ObjectGrid.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridException](#)

IBM.WebSphere.Caching.Map UndefinedMapException

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[UndefinedMapException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [UndefinedMapException](#) type exposes the following members.
Constructors

Name	Description
<input type="checkbox"/> UndefinedMapException	Initializes a new instance of the UndefinedMapException class.
<input type="checkbox"/> UndefinedMapException(Exception)	Initializes a new instance of the UndefinedMapException class with a specified exception cause.
<input type="checkbox"/> UndefinedMapException(String)	Initializes a new instance of the UndefinedMapException class with the specified error message.
<input type="checkbox"/> UndefinedMapException(String, Exception)	Initializes a new instance of the UndefinedMapException class with the specified error message and exception cause.





[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception.

		(Inherited from Exception.)
<input type="text" value="Source"/>	Source	Gets or sets the name of the application or the object that causes the error.
<input type="text" value="StackTrace"/>	StackTrace	(Inherited from Exception.) Gets a string representation of the frames on the call stack at the time the current exception was thrown.
<input type="text" value="TargetSite"/>	TargetSite	(Inherited from Exception.) Gets the method that throws the current exception.

[Back to Top](#)

See Also

[UndefinedMapException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> UndefinedMapExcepti on	Initializes a new instance of the UndefinedMapException class.
<input type="checkbox"/> UndefinedMapExcepti on(Exception)	Initializes a new instance of the UndefinedMapException class with a specified exception cause.
<input type="checkbox"/> UndefinedMapExcepti on(String)	Initializes a new instance of the UndefinedMapException class with the specified error message.
<input type="checkbox"/> UndefinedMapExcepti on(String, Exception)	Initializes a new instance of the UndefinedMapException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[UndefinedMapException Class](#)

[UndefinedMapException Members](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UndefinedMapException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UndefinedMapException](#) class.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[UndefinedMapException Class](#)

[UndefinedMapException Members](#)

[UndefinedMapException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UndefinedMapException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UndefinedMapException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[UndefinedMapException Class](#)

[UndefinedMapException Members](#)

[UndefinedMapException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UndefinedMapException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UndefinedMapException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[UndefinedMapException Class](#)

[UndefinedMapException Members](#)

[UndefinedMapException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

UndefinedMapException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [UndefinedMapException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Map](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[UndefinedMapException Class](#)

[UndefinedMapException Members](#)

[UndefinedMapException Overload](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [UndefinedMapException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[UndefinedMapException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [UndefinedMapException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[UndefinedMapException Class](#)

[IBM.WebSphere.Caching.Map Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The IBM.WebSphere.Caching.Security namespace includes the application programming interfaces specific to security.

The eXtreme Scale client supports transport security and authentication and authorization using the [ICredentialGenerator](#) interface. Security is configured using a client properties file.

Classes

Class	Description
<input type="checkbox"/> AccessControlException	This Java exception is thrown when an access control check indicates that access should not be granted.
<input type="checkbox"/> CannotGenerateCredentialException	A CannotGenerateCredentialException exception occurs if a credential cannot be generated.
<input type="checkbox"/> ExpiredCredentialException	An ExpiredCredentialException exception indicates that the credential that was used for authentication is expired.
<input type="checkbox"/> GridSecurityException	An GridSecurityException exception occurs for general security exceptions.

Interfaces

Interface	Description
<input type="checkbox"/> ICredential	Specifies the interface definition of the Credential class. Application developers implement this interface to create a custom ICredential implementation. An instance is created with a custom ICredentialGenerator .
<input type="checkbox"/> ICredentialGenerator	Specifies the interface definition of the CredentialGenerator class. Application developers implement this interface to generate custom ICredential objects to use for authentication and authorization.

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AccessControlException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

This Java exception is thrown when an access control check indicates that access should not be granted.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.GridServerRuntimeException](#)

IBM.WebSphere.Caching.Security AccessControlException

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AccessControlException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AccessControlException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> AccessControlException	Initializes a new instance of the AccessControlException class.
<input type="checkbox"/> AccessControlException(Exception)	Initializes a new instance of the AccessControlException class with a specified exception cause.
<input type="checkbox"/> AccessControlException(String)	Initializes a new instance of the AccessControlException class with the specified error message.
<input type="checkbox"/> AccessControlException(String, Exception)	Initializes a new instance of the AccessControlException class with the specified error message and exception cause.

[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Provides a user-readable representation of this GridServerRuntimeException exception. (Inherited from GridServerRuntimeException .)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)

<input type="text"/>	JavaException ClassName	Identifies the Java Exception class name that this exception identifies. (Inherited from GridServerRuntimeException .)
<input type="text"/>	Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="text"/>	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="text"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="text"/>	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AccessControlException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> AccessControlExcepti on	Initializes a new instance of the AccessControlException class.
<input type="checkbox"/> AccessControlExcepti on(Exception)	Initializes a new instance of the AccessControlException class with a specified exception cause.
<input type="checkbox"/> AccessControlExcepti on(String)	Initializes a new instance of the AccessControlException class with the specified error message.
<input type="checkbox"/> AccessControlExcepti on(String, Exception)	Initializes a new instance of the AccessControlException > class with the specified error message and exception cause.

[Back to Top](#)

See Also

[AccessControlException Class](#)

[AccessControlException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AccessControlException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AccessControlException](#) class.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[AccessControlException Class](#)

[AccessControlException Members](#)

[AccessControlException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AccessControlException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AccessControlException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[AccessControlException Class](#)

[AccessControlException Members](#)

[AccessControlException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AccessControlException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AccessControlException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

See Also

[AccessControlException Class](#)

[AccessControlException Members](#)

[AccessControlException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

AccessControlException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [AccessControlException](#)> class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

innerException

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[AccessControlException Class](#)

[AccessControlException Members](#)

[AccessControlException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AccessControlException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Provides a user-readable representation of this GridServerRuntimeException exception. (Inherited from GridServerRuntimeException .)

[Back to Top](#)

See Also

[AccessControlException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [AccessControlException](#) type exposes the following members.
Properties

Name	Description
<input type="checkbox"/> Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/> HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/> InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/> JavaException ClassName	Identifies the Java Exception class name that this exception identifies. (Inherited from GridServerRuntimeException .)
<input type="checkbox"/> Message	Gets a message that describes the current exception. (Inherited from Exception.)
<input type="checkbox"/> Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
<input type="checkbox"/> StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
<input type="checkbox"/> TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[AccessControlException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CannotGenerateCredentialException IBM WebSphere™ eXtreme Scale Client for .NET API
n Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

A CannotGenerateCredentialException exception occurs if a credential cannot be generated.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.Security GridSecurityException](#)

IBM.WebSphere.Caching.Security CannotGenerateCredentialException

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[CannotGenerateCredentialException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CannotGenerateCredentialException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> CannotGenerateCredentialException	Initializes a new instance of the CannotGenerateCredentialException class.
<input type="checkbox"/> CannotGenerateCredentialException(Exception)	Initializes a new instance of the CannotGenerateCredentialException class with the specified exception cause.
<input type="checkbox"/> CannotGenerateCredentialException(String)	Initializes a new instance of the CannotGenerateCredentialException class with the specified error message.
<input type="checkbox"/> CannotGenerateCredentialException(String, Exception)	Initializes a new instance of the CannotGenerateCredentialException class with the specified error message and exception cause.



[Back to Top](#)






Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)

	InnerExc eption	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTra ce	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSit e	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[CannotGenerateCredentialException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> CannotGenerateCredentialException	Initializes a new instance of the CannotGenerateCredentialException class.
<input type="checkbox"/> CannotGenerateCredentialException(Exception)	Initializes a new instance of the CannotGenerateCredentialException class with the specified exception cause.
<input type="checkbox"/> CannotGenerateCredentialException(String)	Initializes a new instance of the CannotGenerateCredentialException class with the specified error message.
<input type="checkbox"/> CannotGenerateCredentialException(String, Exception)	Initializes a new instance of the CannotGenerateCredentialException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[CannotGenerateCredentialException Class](#)

[CannotGenerateCredentialException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CannotGenerateCredentialException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CannotGenerateCredentialException](#) class.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[CannotGenerateCredentialException Class](#)

[CannotGenerateCredentialException Members](#)

[CannotGenerateCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CannotGenerateCredentialException
Constructor (Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CannotGenerateCredentialException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[CannotGenerateCredentialException Class](#)

[CannotGenerateCredentialException Members](#)

[CannotGenerateCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CannotGenerateCredentialException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CannotGenerateCredentialException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[CannotGenerateCredentialException Class](#)

[CannotGenerateCredentialException Members](#)

[CannotGenerateCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

CannotGenerateCredentialException
Constructor (String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [CannotGenerateCredentialException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[CannotGenerateCredentialException Class](#)

[CannotGenerateCredentialException Members](#)

[CannotGenerateCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CannotGenerateCredentialException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[CannotGenerateCredentialException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [CannotGenerateCredentialException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[CannotGenerateCredentialException Class](#)
[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ExpiredCredentialException Class IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An ExpiredCredentialException exception indicates that the credential that was used for authentication is expired.

Inheritance Hierarchy

System Object

System Exception

[IBM.WebSphere.Caching.Security.GridSecurityException](#)

IBM.WebSphere.Caching.Security ExpiredCredentialException

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ExpiredCredentialException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification




The [ExpiredCredentialException](#) type exposes the following members.
Constructors





Name	Description
<input type="checkbox"/> ExpiredCredentialException	Initializes a new instance of the ExpiredCredentialException class.
<input type="checkbox"/> ExpiredCredentialException(Exception)	Initializes a new instance of the ExpiredCredentialException class with the specified exception cause.
<input type="checkbox"/> ExpiredCredentialException(String)	Initializes a new instance of the ExpiredCredentialException class with the specified error message.
<input type="checkbox"/> ExpiredCredentialException(String, Exception)	Initializes a new instance of the ExpiredCredentialException class with the specified error message and exception cause.

[Back to Top](#)
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)
Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)

	Message	Gets a message that describes the current exception. (Inherited from Exception.)
	Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
	TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ExpiredCredentialException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> ExpiredCredentialException	Initializes a new instance of the ExpiredCredentialException class.
<input type="checkbox"/> ExpiredCredentialException(Exception)	Initializes a new instance of the ExpiredCredentialException class with the specified exception cause.
<input type="checkbox"/> ExpiredCredentialException(String)	Initializes a new instance of the ExpiredCredentialException class with the specified error message.
<input type="checkbox"/> ExpiredCredentialException(String, Exception)	Initializes a new instance of the ExpiredCredentialException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[ExpiredCredentialException Class](#)

[ExpiredCredentialException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ExpiredCredentialException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ExpiredCredentialException](#) class.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ExpiredCredentialException Class](#)

[ExpiredCredentialException Members](#)

[ExpiredCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ExpiredCredentialException Constructor IBM WebSphere™ eXtreme Scale Client for .NET
(Exception) API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ExpiredCredentialException](#) class with the specified exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause of the exception is nonexistent or unknown.

See Also

[ExpiredCredentialException Class](#)

[ExpiredCredentialException Members](#)

[ExpiredCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ExpiredCredentialException
Constructor (String)

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ExpiredCredentialException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[ExpiredCredentialException Class](#)

[ExpiredCredentialException Members](#)

[ExpiredCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ExpiredCredentialException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for
.NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [ExpiredCredentialException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[ExpiredCredentialException Class](#)

[ExpiredCredentialException Members](#)

[ExpiredCredentialException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ExpiredCredentialException](#) type exposes the following members.
Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ExpiredCredentialException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ExpiredCredentialException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[ExpiredCredentialException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridSecurityException IBM WebSphere™ eXtreme Scale Client for .NET API
Class Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

An GridSecurityException exception occurs for general security exceptions.

Inheritance Hierarchy

System Object

System Exception

IBM.WebSphere.Caching.Security GridSecurityException

[IBM.WebSphere.Caching.Security CannotGenerateCredentialException](#)

[IBM.WebSphere.Caching.Security ExpiredCredentialException](#)

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridSecurityException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridSecurityException](#) type exposes the following members.

Constructors

Name	Description
<input type="checkbox"/> GridSecurityException	Initializes a new instance of the GridSecurityException class.
<input type="checkbox"/> GridSecurityException(Exception)	Initializes a new instance of the GridSecurityException class with a specified exception cause.
<input type="checkbox"/> GridSecurityException(String)	Initializes a new instance of the GridSecurityException class with the specified error message.
<input type="checkbox"/> GridSecurityException(String, Exception)	Initializes a new instance of the GridSecurityException class with the specified error message and exception cause.





[Back to Top](#)

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

Properties

Name	Description
<input type="checkbox"/>  Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
<input type="checkbox"/>  HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
<input type="checkbox"/> HRESULT	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
<input type="checkbox"/>  InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
<input type="checkbox"/>  Message	Gets a message that describes the current exception.

		(Inherited from Exception.)
<input style="width: 50px; height: 15px;" type="text" value="?"/>	Source	Gets or sets the name of the application or the object that causes the error.
		(Inherited from Exception.)
<input style="width: 50px; height: 15px;" type="text" value="?"/>	StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown.
		(Inherited from Exception.)
<input style="width: 50px; height: 15px;" type="text" value="?"/>	TargetSite	Gets the method that throws the current exception.
		(Inherited from Exception.)

[Back to Top](#)

See Also

[GridSecurityException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Overload List

Name	Description
<input type="checkbox"/> GridSecurityException	Initializes a new instance of the GridSecurityException class.
<input type="checkbox"/> GridSecurityException (Exception)	Initializes a new instance of the GridSecurityException class with a specified exception cause.
<input type="checkbox"/> GridSecurityException (String)	Initializes a new instance of the GridSecurityException class with the specified error message.
<input type="checkbox"/> GridSecurityException (String, Exception)	Initializes a new instance of the GridSecurityException class with the specified error message and exception cause.

[Back to Top](#)

See Also

[GridSecurityException Class](#)

[GridSecurityException Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridSecurityException
Constructor

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridSecurityException](#) class.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[GridSecurityException Class](#)

[GridSecurityException Members](#)

[GridSecurityException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridSecurityException Constructor (Exception) IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridSecurityException](#) class with a specified exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Parameters

cause

Type: System.Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridSecurityException Class](#)

[GridSecurityException Members](#)

[GridSecurityException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridSecurityException Constructor IBM WebSphere™ eXtreme Scale Client for .NET API
(String) Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridSecurityException](#) class with the specified error message.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System.String

Specifies the error message that explains the reason for the exception.

See Also

[GridSecurityException Class](#)

[GridSecurityException Members](#)

[GridSecurityException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

GridSecurityException Constructor
(String, Exception)

IBM WebSphere™ eXtreme Scale Client for .NET
API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Initializes a new instance of the [GridSecurityException](#) class with the specified error message and exception cause.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

message

Type: System String

Specifies the error message that explains the reason for the exception.

cause

Type: System Exception

Specifies the exception that is the cause of the current exception. A null value is permitted and indicates that the cause is nonexistent or unknown.

See Also

[GridSecurityException Class](#)

[GridSecurityException Members](#)

[GridSecurityException Overload](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridSecurityException](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Determines whether the specified Object is equal to the current Object. (Inherited from Object.)
<input type="checkbox"/> Finalize	Allows an Object to attempt to free resources and perform other cleanup operations before the Object is reclaimed by garbage collection. (Inherited from Object.)
<input type="checkbox"/> GetBaseException	When overridden in a derived class, returns the Exception that is the root cause of one or more subsequent exceptions. (Inherited from Exception.)
<input type="checkbox"/> GetHashCode	Serves as a hash function for a particular type. (Inherited from Object.)
<input type="checkbox"/> GetObjectData	When overridden in a derived class, sets the SerializationInfo with information about the exception. (Inherited from Exception.)
<input type="checkbox"/> GetType	Gets the runtime type of the current instance. (Inherited from Exception.)
<input type="checkbox"/> MemberwiseClone	Creates a shallow copy of the current Object. (Inherited from Object.)
<input type="checkbox"/> ToString	Creates and returns a string representation of the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridSecurityException Class](#)

[IBM.WebSphere.Caching.Security Namespace](#)









IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [GridSecurityException](#) type exposes the following members.
Properties

Name	Description
 Data	Gets a collection of key/value pairs that provide additional user-defined information about the exception. (Inherited from Exception.)
 HelpLink	Gets or sets a link to the help file associated with this exception. (Inherited from Exception.)
 HResult	Gets or sets HRESULT, a coded numerical value that is assigned to a specific exception. (Inherited from Exception.)
 InnerException	Gets the Exception instance that caused the current exception. (Inherited from Exception.)
 Message	Gets a message that describes the current exception. (Inherited from Exception.)
 Source	Gets or sets the name of the application or the object that causes the error. (Inherited from Exception.)
 StackTrace	Gets a string representation of the frames on the call stack at the time the current exception was thrown. (Inherited from Exception.)
 TargetSite	Gets the method that throws the current exception. (Inherited from Exception.)

[Back to Top](#)

See Also

[GridSecurityException Class](#)[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICredential Interface IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies the interface definition of the Credential class.

Application developers implement this interface to create a custom ICredential implementation. An instance is created with a custom [ICredentialGenerator](#).

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICredential Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [ICredential](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Checks if two credential objects are equal. Two credential objects are considered equal if they represent the same identity and security information.
<input type="checkbox"/> GetHashCode	Specifies a hash code representation of the credential object.

[Back to Top](#)

See Also

[ICredential Interface](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

The [ICredential](#) type exposes the following members.

Methods

Name	Description
<input type="checkbox"/> Equals	Checks if two credential objects are equal. Two credential objects are considered equal if they represent the same identity and security information.
<input type="checkbox"/> GetHashCode	Specifies a hash code representation of the credential object.

[Back to Top](#)

See Also

[ICredential Interface](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICredential Equals
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Checks if two credential objects are equal. Two credential objects are considered equal if they represent the same identity and security information.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to ClipboardPrint](#)

Parameters

credential

Type: [IBM.WebSphere.Caching.Security ICredential](#)

Specifies the object that is being tested for equality with the selected object.

Return Value

Returns true if both credential objects are equivalent.

See Also

[ICredential Interface](#)

[ICredential Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICredential GetHashCode
Method

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies a hash code representation of the credential object.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

Return Value

Returns the hash code representation of the credential object.

See Also

[ICredential Interface](#)

[ICredential Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICredentialGenerator
Interface

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies the interface definition of the CredentialGenerator class.

Application developers implement this interface to generate custom [ICredential](#) objects to use for authentication and authorization.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICredentialGenerator Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICredentialGenerator](#) type exposes the following members.

Properties

Name	Description
Credential	Specifies the Credential that represents the client. A <code>CannotGenerateCredentialException</code> exception occurs if generation of the credential for the client fails.
Properties	Specifies the user-defined properties for the CredentialGenerator factory. These properties are used to influence the contents of the credential that is generated by the CredentialGenerator factory. This property value is set to the <code>credentialGeneratorProps</code> property value in the client security configuration property file.

[Back to Top](#)

See Also

[ICredentialGenerator Interface](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

The [ICredentialGenerator](#) type exposes the following members.
Properties

Na me	Description
Cr ed ent ial	Specifies the Credential that represents the client. A CannotGenerateCredentialException exception occurs if generation of the credential for the client fails.
Pr op ert ies	Specifies the user-defined properties for the CredentialGenerator factory. These properties are used to influence the contents of the credential that is generated by the CredentialGenerator factory. This property value is set to the credentialGeneratorProps property value in the client security configuration property file.

[Back to Top](#)

See Also

[ICredentialGenerator Interface](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional
information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICredentialGenerator Credential Property IBM WebSphere™ eXtreme Scale Client for .NET API Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies the Credential that represents the client. A CannotGenerateCredentialException exception occurs if generation of the credential for the client fails.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICredentialGenerator Interface](#)

[ICredentialGenerator Members](#)

[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ICredentialGenerator Properties
Property

IBM WebSphere™ eXtreme Scale Client for .NET API
Specification

[Send Feedback](#)

IBM WebSphere™ eXtreme Scale Client for .NET Release 8.6.0.0 API Specification

Specifies the user-defined properties for the CredentialGenerator factory.

These properties are used to influence the contents of the credential that is generated by the CredentialGenerator factory. This property value is set to the credentialGeneratorProps property value in the client security configuration property file.

Namespace: [IBM.WebSphere.Caching.Security](#)

Assembly: Client.Api (in Client.Api.dll) Version: 8.6.0.0

Syntax

VB

[C#](#)

C++

F#

JScript

[Copy to Clipboard](#)[Print](#)

See Also

[ICredentialGenerator Interface](#)

[ICredentialGenerator Members](#)


[IBM.WebSphere.Caching.Security Namespace](#)

IBM WebSphere™ eXtreme Scale Client for .NET API Specification. For additional information see the [WebSphere™ eXtreme Scale Library](#). © Copyright IBM Corporation 2012.

[Send Feedback](#) on this topic to WAS Documentation Team.

ユーザー・インターフェース設定

この参照情報では、WebSphere® Application Server 管理コンソールなどのページで表示および構成できる設定について説明します。

目次を開いてこの参照情報の場所を見つけるには、インフォメーション・センターのポータル上にある「目次に表示」() ボタンをクリックします。例えば、インフォメーション・センターの検索や、Google などのインターネット検索エンジンからこのページを見つけた場合は、このボタンをクリックすると、インフォメーション・センターのナビゲーション・ツリーにこのページの場所が表示されます。このナビゲーション項目の下にある一段下がった目次をブラウズできます。

[eXtreme Scale セッション管理設定](#)

セッション・パーシスタンスのために WebSphere eXtreme Scale または WebSphere DataPower® XC10 Appliance を使用するように、WebSphere Application Server アプリケーションを構成できます。

[カタログ・サービス・ドメイン・コレクション](#)

このページを使用すると、カタログ・サービス・ドメインを管理できます。カタログ・サービス・ドメインは、断片の配置を管理し、データ・グリッド内のコンテナ・サーバーのヘルスをモニターするカタログ・サーバーのグループを定義します。

[カタログ・サービス・ドメイン設定](#)

このページを使用すると、特定のカタログ・サービス・ドメインの設定を管理できます。カタログ・サービス・ドメインは、断片の配置を管理し、データ・グリッド内のコンテナ・サーバーのヘルスをモニターするカタログ・サーバーのグループを定義します。デプロイメント・マネージャーと同じセルにあるカタログ・サービス・ドメインを定義できます。WebSphere eXtreme Scale 構成が異なるセルにある場合、またはデータ・グリッドが Java™ SE プロセスから構成される場合は、リモート・カタログ・サービス・ドメインも定義できます。

[クライアント・セキュリティー・プロパティー](#)

このページを使用して、カタログ・サービス・ドメインのクライアント・セキュリティーを構成します。この設定は、カタログ・サービス・ドメイン内のすべてのサーバーに適用されます。これらのプロパティーは、`com.ibm.websphere.xs.sessionFilterProps` カスタム・プロパティーを使用して `splicer.properties` ファイルを指定するか、アプリケーション EAR ファイルを接合してオーバーライドできます。

[カタログ・サービス・ドメインのカスタム・プロパティー](#)

カスタム・プロパティーを定義すると、カタログ・サービス・ドメインの構成をさらに編集できます。

親トピック: [参照](#)

eXtreme Scale セッション管理設定

セッション・パーシスタンスのために WebSphere® eXtreme Scale または WebSphere DataPower® XC10 Appliance を使用するよう、WebSphere Application Server アプリケーションを構成できます。

これらの設定は、エンタープライズ・アプリケーション・インストール・ウィザード、あるいはアプリケーションまたはサーバー詳細ページで編集することができます。

- バージョン 7.0: 「アプリケーション」 > 「新規アプリケーション」 > 「新規エンタープライズ・アプリケーション」とクリックして、アプリケーション作成のための詳細パスを選択します。
- バージョン 7.0: 「アプリケーション」 > 「アプリケーション・タイプ」 > 「WebSphere エンタープライズ・アプリケーション」 > 「*application_name*」 > 「Web モジュール・プロパティ」 > 「セッション管理」 > 「セッション管理設定」
- バージョン 7.0: 「サーバー」 > 「サーバー・タイプ」 > 「WebSphere Application Server」 > 「*server_name*」 > 「コンテナ設定」 > 「セッション管理設定」

親トピック: [ユーザー・インターフェース設定](#)

セッション管理使用可能

セッション管理を使用可能にし、セッション・パーシスタンスのために WebSphere eXtreme Scale の組み込みデータ・グリッドまたはリモート・データ・グリッド、あるいは WebSphere DataPower XC10 アプライアンスを使用できるようにします。

セッション・パーシスタンスの管理

セッション・パーシスタンスがどのように管理されるかを指定します。以下のオプションのいずれかを選択できます。

- WebSphere DataPower XC10 アプライアンス
- リモート eXtreme Scale データ・グリッド
- 組み込み eXtreme Scale データ・グリッド

構成する残りの設定は、選択したセッション・パーシスタンス・メカニズムによります。

WebSphere DataPower XC10 アプライアンス固有の設定

以下の設定は、セッション・パーシスタンスのための WebSphere DataPower XC10 アプライアンスの構成に固有の設定です。

WebSphere DataPower XC10 Appliance の IP またはホスト名

セッションのパーシストのために使用されるアプライアンスの IP またはホスト名を指定します。

IBM® WebSphere DataPower XC10 アプライアンス管理資格情報

DataPower XC10 アプライアンスのユーザー・インターフェースにログインするために使用するユーザー名とパスワードを指定します。「テスト接続...」をクリックして、アプライアンスへの接続をテストします。

セッション・パーシスタンス設定

セッションがパーシストされるデータ・グリッドを指定します。以下のオプションのいずれかを選択できます。

- **IBM WebSphere DataPower XC10** アプライアンスの新規のデータ・グリッドにおけるセッション・パーシスト。ここで「データ・グリッド名」を指定できます。
- **IBM WebSphere DataPower XC10** アプライアンスの既存のデータ・グリッドにおけるセッション・パーシスト。ここで「既存のデータ・グリッド名」を入力または参照できます。

リモート eXtreme Scale データ・グリッド構成

以下の設定は、セッション・パーシスタンスのためのリモート eXtreme Scale グリッドの構成に固有の設定です。

リモート・セッション・データ・グリッドを管理するカタログ・サービス・ドメイン

セッションの管理に使用するカタログ・サービス・ドメインを指定します。

カタログ・サービス・ドメインが表示されていない場合や、新規カタログ・サービス・ドメインを作成したい場合は、「システム管理」 > 「WebSphere eXtreme Scale」 > 「カタログ・サービス・ドメイン」の順にクリックします。

セッション情報が保管されるリモート・データ・グリッド

カタログ・サービス・ドメイン内でセッション情報が保管されるデータ・グリッドの名前を指定します。カタログ・サービスを選択すると、アクティブ・リモート・グリッドのリストが取り込まれます。リモート・データ・グリッドは、eXtreme Scale 構成に事前に存在している必要があります。

組み込み eXtreme Scale データ・グリッド構成

以下の設定は、組み込み eXtreme Scale の構成に固有の設定です。組み込み eXtreme Scale シナリオでは、eXtreme Scale プロセスは、WebSphere Application Server プロセスによってホスティングされます。

eXtreme Scale 組み込みデータ・グリッド構成

- デフォルトの **ObjectGrid** 構成を使用
- カスタム **ObjectGrid** 構成ファイルを指定

構成にコピーする **objectgrid.xml** ファイルの絶対パス

使用する構成の **objectgrid.xml** ファイルの絶対パスを指定します。

構成にコピーされる **objectgriddeployment.xml** ファイルの絶対パス

使用する構成の **objectgriddeployment.xml** ファイルの絶対パスを指定します。

カタログ・サービス・ドメイン・コレクション

このページを使用すると、カタログ・サービス・ドメインを管理できます。カタログ・サービス・ドメインは、断片の配置を管理し、データ・グリッド内のコンテナ・サーバーのヘルスをモニターするカタログ・サーバーのグループを定義します。

この管理コンソール・ページを表示するには、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」の順にクリックします。新規カタログ・サービス・ドメインを作成するには、「新規」をクリックします。カタログ・サービス・ドメインを削除するには、削除したいカタログ・サービス・ドメインを選択して、「削除」をクリックします。

親トピック: [ユーザー・インターフェース設定](#)

テスト接続

「テスト接続」ボタンをクリックすると、定義されたカタログ・サービス・ドメイン・エンドポイントのすべてが1つずつ照会され、使用可能なエンドポイントがあった場合、カタログ・サービス・ドメインへの接続が成功したことを示すメッセージが返されます。このボタンを使用すると、接続情報およびセキュリティー情報が正しく構成されているかをテストすることができます。


デフォルトの設定

デフォルトとして使用するカタログ・サービス・ドメインを定義します。1つのカタログ・サービス・ドメインをデフォルトとして選択し、「デフォルトの設定」を選択します。1つのカタログ・サービス・ドメインのみをデフォルトとして選択できます。

名前

カタログ・サービス・ドメインの名前を指定します。

デフォルト

リスト内のどのカタログ・サービス・ドメインがデフォルトであるかを指定します。デフォルトのカタログ・サービス・ドメインは、次のアイコン  で示されます。

カタログ・サービス・ドメイン設定

このページを使用すると、特定のカタログ・サービス・ドメインの設定を管理できます。カタログ・サービス・ドメインは、断片の配置を管理し、データ・グリッド内のコンテナ・サーバーのヘルスをモニターするカタログ・サーバーのグループを定義します。デプロイメント・マネージャーと同じセルにあるカタログ・サービス・ドメインを定義できます。WebSphere® eXtreme Scale 構成が異なるセルにある場合、またはデータ・グリッドが Java™ SE プロセスから構成される場合は、リモート・カタログ・サービス・ドメインも定義できます。

この管理コンソール・ページを表示するには、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」 > 「*catalog_service_domain_name*」の順にクリックします。

親トピック: [ユーザー・インターフェース設定](#)

テスト接続

「テスト接続」ボタンをクリックすると、定義されたカタログ・サービス・ドメイン・エンドポイントのすべてが1つずつ照会され、使用可能なエンドポイントがあった場合、カタログ・サービス・ドメインへの接続が成功したことを示すメッセージが返されます。このボタンを使用すると、接続情報およびセキュリティー情報が正しく構成されているかをテストすることができます。

名前

カタログ・サービス・ドメインの名前を指定します。

別のカタログ・サービス・ドメインが明示的に指定されない限り、このカタログ・サービス・ドメインをデフォルトとして使用可能にする

このチェック・ボックスを選択すると、選択されたカタログ・サービス・ドメインがそのセルのデフォルトのカタログ・サービス・ドメインになります。WebSphere eXtreme Scale プロファイルで拡張されているセル内の各サーバー・プロファイルは、選択したカタログ・サービス・ドメインに属しています。

異なるカタログ・サーバーのセットを指すようにデフォルトのドメインを変更すると、すべてのコンテナおよびクライアントが、再始動後に新規ドメインを参照します。

IBM eXtremeIO (XIO) 通信を有効にする

カタログ・サービス・ドメインが XIO 通信を使用するかどうかを指定します。このオプションを選択しなかった場合は、オブジェクト・リクエスト・ブローカー (ORB) が使用されます。

注: WebSphere Application Server 管理コンソールからリモート・サーバーの XIO 通信を有効にすることはできません。**startXsServer** スクリプトでリモート・サーバーを始動する際に、リモート・サーバーの XIO を有効にしてください。

カタログ・サーバー

このカタログ・サービス・ドメインに属するカタログ・サーバーのリストを指定します。

リストにカタログ・サーバーを追加するには、「新規」をクリックします。このカタログ・サーバーは、eXtreme Scale 構成に事前に存在する必要があります。エンドポイントを選択して、「編集」または「削除」をクリックすると、リストでサーバーを編集または削除することもできます。各カタログ・サーバー・エンドポイントについて以下のプロパティを定義します。

カタログ・サーバー・エンドポイント

カタログ・サービスが実行している既存のアプリケーション・サーバーまたはリモート・サーバーの名前を指定します。1つのカタログ・サービス・ドメインに、既存のアプリケーション・サーバーとリモート・サーバーのエンドポイントの混合を含めることはできません。

- **既存のアプリケーション・サーバー:** セル内のアプリケーション・サーバー、ノード・エージェント、またはデプロイメント・マネージャーのパスを指定します。選択されたサーバーでカタログ・サービスが自動的に開始します。既存のアプリケーション・サーバーのリストから選択します。カタログ・サービス・ドメイン内で定義したすべてのアプリケーション・サーバーは、同じコア・グループになければなりません。
- **リモート・サーバー:** リモート・カタログ・サーバーのホスト名を指定します。

WebSphere DataPower® XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのホスト名を指定します。

注: WebSphere Application Server 管理コンソールからリモート・サーバーの XIO 通信を有効にすることはできません。**startXsServer** スクリプトでリモート・サーバーを始動する際に、リモート・サーバーの XIO を有効にしてください。

クライアント・ポート

カタログ・サービス・ドメイン内のカタログ・サーバー間の通信に使用するポートを指定します。WebSphere Application Server プロセスで実行中のカタログ・サーバーの場合、この値は必須です。別のプロセスによって使用されていないどのポートにも値を設定できます。


リスナー・ポート

クライアントとの通信に使用するポートを指定します。この値はリモート・エンドポイントに必要で、カタログ・サービスが開始されたときに使用された値と一致している必要があります。リスナー・ポートは、カタログ・サービスとの通信のために、クライアントおよびコンテナによって使用されます。

WebSphere DataPower XC10 アプライアンスのリモート・エンドポイントの場合: アプライアンスのリモート・エンドポイントに、値 2809 を使用します。

状況

表 1. カタログ・サーバー・エンドポイント状況

アイコン	定義
	不明
	始動済み
	停止済み

クライアント・セキュリティー・プロパティー

このページを使用して、カタログ・サービス・ドメインのクライアント・セキュリティーを構成します。この設定は、カタログ・サービス・ドメイン内のすべてのサーバーに適用されます。これらのプロパティーは、`com.ibm.websphere.xs.sessionFilterProps` カスタム・プロパティーを使用して `splicer.properties` ファイルを指定するか、アプリケーション EAR ファイルを接合してオーバーライドできます。

この管理コンソール・ページを表示するには、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」 > 「`catalog_service_domain_name`」 > 「クライアント・セキュリティー・プロパティー」の順にクリックします。

親トピック: [ユーザー・インターフェース設定](#)

クライアント・セキュリティーの使用可能化

カタログ・サーバーのクライアント・セキュリティーが使用可能なことを指定します。選択したカタログ・サーバーに関連付けられたサーバー・プロパティー・ファイルには、一致する **securityEnabled** 設定がなければなりません。これらの設定が一致しない場合、例外が発生します。

資格情報の認証

資格情報の認証を実施するか、またはサポートするかを示します。

常になし

クライアント資格情報の認証は実施されません。

必須

資格情報の認証は必ず実施されます。サーバーが資格情報の認証をサポートしない場合、クライアントはサーバーに接続できません。

サポートされる

資格情報の認証は、クライアントとサーバーの両方が資格情報の認証をサポートする場合のみ実施されます。

認証の再試行回数

資格情報の有効期限が切れている場合に認証を再試行できる回数を指定します。

認証の再試行を望まない場合は、値を 0 に設定します。

資格情報生成プログラムクラス

クライアントが `CredentialGenerator` オブジェクトから資格情報を取得するよう、`com.ibm.websphere.objectgrid.security.plugins.CredentialGenerator` 実装クラスを指定します。

2つの事前定義資格情報生成プログラム・クラスから選択することも、カスタム資格情報生成プログラムを指定することもできます。カスタム資格情報生成プログラムを選択した場合は、資格情報生成プログラム・クラス名を指定する必要があります。

- `com.ibm.websphere.objectgrid.security.plugins.builtins.UserPasswordCredentialGenerator`
- `com.ibm.websphere.objectgrid.security.plugins.builtins.WSTokenCredentialGenerator`
- カスタム資格情報生成プログラム

サブジェクト・タイプ

J2EE 呼び出し元を使用するか、J2EE `runAs` サブジェクト・タイプを使用するかを指定します。この値は、`WSTokenCredentialGenerator` 資格情報生成プログラムを選択するときに指定する必要があります。

- **runAs:** サブジェクトには、J2EE `run as ID` および J2EE `run as` 資格情報のプリンシパルが含まれています。
- **呼び出し元:** このサブジェクトには、J2EE 呼び出し元および J2EE 呼び出し元資格情報のプリンシパルが含まれています。

ユーザー ID

`UserPasswordCredentialGenerator` 資格情報生成プログラム実装を使用する場合、ユーザー ID を指定します。

パスワード

`UserPasswordCredentialGenerator` 資格情報生成プログラムの実装を使用する場合、パスワードを指定します。

資格情報生成プログラム・プロパティー

カスタム CredentialGenerator 実装クラスの プロパティを指定します。このプロパティは、setProperties(String) メソッドを使用してオブジェクトに設定されます。資格情報生成プログラムのプロパティ値は、「資格情報生成プログラム・クラス」フィールドに値が指定されている場合のみ使用されます。

カタログ・サービス・ドメインのカスタム・プロパティ

カスタム・プロパティを定義すると、カタログ・サービス・ドメインの構成をさらに編集できます。

この管理コンソール・ページを表示するには、「システム管理」 > 「**WebSphere eXtreme Scale**」 > 「カタログ・サービス・ドメイン」 > 「カスタム・プロパティ」の順にクリックします。新規カスタム・プロパティを作成するには、「**新規**」をクリックします。

親トピック: [ユーザー・インターフェース設定](#)

名前

カタログ・サービス・ドメインのカスタム・プロパティの名前を指定します。

値

カタログ・サービス・ドメインのカスタム・プロパティの値を指定します。

- [IBM WebSphere DataPower XC10 アプライアンス バージョン 2.5 資料](#)
- [IBM WebSphere DataPower XC10 アプライアンス の概要](#)
 - [新機能](#)
 - [リリース情報](#)
 - [アプライアンスのトポロジー](#)
 - [マルチマスター複製を実装する集合のリンクのトポロジー](#)
 - [エンタープライズ・データ・グリッドの概要](#)
 - [トランザクション処理の概要](#)
 - [トランザクション](#)
 - [ロック・ストラテジー](#)
 - [ロック・タイプ](#)
 - [デッドロック](#)
 - [データ・アクセスおよびトランザクション](#)
 - [トランザクション分離](#)
 - [2 フェーズ・コミットとエラー・リカバリー](#)
 - [特記事項](#)
 - [プライバシー・ポリシーに関する考慮事項](#)
- [シナリオ](#)
 - [エンタープライズ・データ・グリッドの構成](#)
 - [エンタープライズ・データ・グリッドの概要](#)
 - [IBM eXtremeIO \(XIO\) の構成](#)
 - [エンタープライズ・データ・グリッド・アプリケーションの開発](#)
 - [クラスの進化](#)
 - [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)
 - [ClassAlias および FieldAlias 注釈](#)
 - [PartitionKey 注釈を使用したキーから区画へのマップ](#)
 - [Java および C# の等価データ型](#)
 - [WebSphere DataPower XC10 アプライアンスの保護](#)
 - [データ・グリッドへのセキュアなアクセスの構成](#)
 - [データ・グリッドのセキュリティーを使用可能に設定する](#)
 - [クライアント・セキュリティーの構成](#)
 - [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)
 - [WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)
 - [データ・グリッド内におけるセッション管理用のカタログ・サービス・ドメインの作成](#)
 - [以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンスの構成](#)
- [チュートリアル: 単純データ・グリッド・アプリケーションの入門](#)
 - [レッスン 1.1: データ・グリッドの構成](#)
 - [モジュール 2: クライアント・アプリケーションの作成](#)
 - [レッスン 2.1: Java クライアント・アプリケーションの作成](#)
 - [レッスン 2.2: .NET クライアント・アプリケーションの作成](#)
 - [レッスン 2.3: エンタープライズ・データ・グリッド・アプリケーションの作成](#)
 - [モジュール 3: データ・グリッド内でのサンプル・アプリケーションの実行](#)
 - [レッスン 3.1: 入門用サンプル・クライアント・アプリケーションの実行](#)
 - [レッスン 3.2: .NET サンプル・クライアント・アプリケーションの実行](#)
 - [レッスン 4: 環境のモニター](#)
- [DataPower XC10 アプライアンス環境の計画](#)
 - [システム要件](#)
 - [ディレクトリー規則](#)
 - [ネットワーク・ポート](#)
 - [IBM WebSphere DataPower XC10 アプライアンス インストールの前提条件](#)
 - [アプライアンスの仕様およびフィーチャー](#)
 - [Type 7199-92x](#)
 - [Type 7199-92x 正面図](#)
 - [Type 7199-92x 背面図](#)
 - [イーサネット・ネットワーク構成](#)
 - [製品のインターオペラビリティ](#)
 - [Microsoft .NET に関する考慮事項](#)

- [WebSphere DataPower XC10 アプライアンス のインストール](#)
 - [クイック・スタート: アプライアンス・ハードウェアのインストール](#)
 - [IBM WebSphere DataPower XC10 アプライアンス の初期設定および構成](#)
 - [xcadminパスワード](#)
 - [WebSphere eXtreme Scale クライアント のインストール](#)
 - [IBM Installation Manager および WebSphere eXtreme Scale クライアント 製品オファリングのインストール](#)
 - [WebSphere eXtreme Scale クライアント 製品のオファリング ID](#)
 - [WebSphere Application Server でのインストール](#)
 - [GUI の使用による IBM Installation Manager のインストール](#)
 - [GUI の使用による WebSphere eXtreme Scale クライアント のインストール](#)
 - [コマンド行の使用による IBM Installation Manager のインストール](#)
 - [コマンド行の使用による WebSphere eXtreme Scale クライアント のインストール](#)
 - [応答ファイルの使用による IBM Installation Manager のインストール](#)
 - [応答ファイルの使用による WebSphere eXtreme Scale クライアント のインストール](#)
 - [鍵リングの作成](#)
 - [IBM Installation Manager の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)
 - [GUI の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)
 - [コマンド行の使用による WebSphere eXtreme Scale クライアント のアンインストール](#)
 - [応答ファイルを使用した WebSphere eXtreme Scale クライアントのアンインストール](#)
 - [WebSphere eXtreme Scale Client for .NET のインストールの概要](#)
 - [WebSphere eXtreme Scale Client for .NET のインストール](#)
 - [サイレント・モードでの WebSphere eXtreme Scale クライアント for .NET のインストール](#)
 - [WebSphere eXtreme Scale クライアント for .NET のアンインストール](#)
 - [インストール・プログラムによらない WebSphere eXtreme Scale Client for .NET のインストール](#)
 - [Liberty プロファイルのインストール](#)
 - [データ・キャッシングと Liberty プロファイル](#)
 - [JAR ファイルの実行による Liberty プロファイルのアプリケーションのサービス環境のインストール](#)
 - [製品インストールのトラブルシューティング](#)
- [WebSphere DataPower XC10 アプライアンス の更新](#)
 - [ファームウェアの更新](#)
 - [WebSphere Application Server 上の WebSphere eXtreme Scale クライアント の更新](#)
 - [IBM Installation Manager の使用によるフィックスパックのインストール](#)
 - [GUI の使用によるフィックスパックのインストール](#)
 - [コマンド行の使用によるフィックスパックのインストール](#)
 - [応答ファイルの使用によるフィックスパックのインストール](#)
 - [IBM Installation Manager の使用によるフィックスパックのアンインストール](#)
 - [GUI の使用によるフィックスパックのアンインストール](#)
 - [コマンド行の使用によるフィックスパックのアンインストール](#)
 - [応答ファイルを使用したフィックスパックのアンインストール](#)
 - [WebSphere eXtreme Scale Client for .NET のアップグレード](#)
 - [WebSphere eXtreme Scale Client for .NET のフィックスパックのサイドバイサイド・インストールの作成](#)
- [アプライアンスの構成](#)
 - [セキュリティ](#)
 - [IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)
 - [WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)
 - [IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティの構成](#)
 - [LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)
 - [ユーザーおよびグループの管理](#)

- [ユーザーの作成](#)
- [ユーザーの管理](#)
- [ユーザーの除去](#)
- [新規ユーザー・アカウントの登録](#)
- [ユーザー・グループの作成](#)
- [ユーザー・グループの管理](#)
- [ユーザー・グループの除去](#)
- [ユーザー許可](#)
- [IBM WebSphere DataPower XC10 アプライアンス のイーサネット・インターフェースの管理](#)
 - [集約インターフェースの追加](#)
 - [集約インターフェースの編集](#)
 - [集約インターフェースの削除](#)
- [構成のインポートとエクスポート](#)
 - [インポートされた構成の管理と構成のエクスポート](#)
- [ドメイン・ネーム・システム \(DNS\) サーバーの管理](#)
- [ホスト名に対する IP アドレスのマッピング](#)
- [日時設定の管理](#)
- [IBM WebSphere DataPower XC10 アプライアンス の E メール送達の管理](#)
- [ユーザー・インターフェースからのアプライアンスのシャットダウンまたは再始動](#)
- [集合およびゾーンの構成](#)
 - [集合の作成](#)
 - [ゾーンの作成および変更](#)
 - [集合間のマルチマスター複製の構成](#)
 - [IBM eXtremeIO \(XIO\) の構成](#)
 - [カタログ・サービス・ドメインのトランスポート・タイプの表示](#)
- [データ・グリッド の構成](#)
 - [単純データ・グリッドの作成](#)
 - [動的マップの構成](#)
 - [ロック・ストラテジーの構成](#)
 - [存続時間 \(TTL\) Evictor の構成](#)
 - [Evictor](#)
 - [ニア・キャッシュの構成](#)
 - [ニア・キャッシュ無効化の構成](#)
 - [クライアントとサーバー間の通信用ホスト名の管理](#)
 - [セッション・データ・グリッドの作成](#)
 - [データ・グリッド \(data grid\)への WebSphere Application Server HTTP セッション・パーシスタンスの構成](#)
 - [WebSphere Application Server での HTTP セッション・マネージャーの構成](#)
 - [WebSphere Portal での HTTP セッション・マネージャーの構成](#)
 - [各種アプリケーション・サーバー用の HTTP セッション・マネージャーの構成](#)
 - [WebSphere eXtreme Scale セッション管理を使用するための WebSphere Application Server メモリー間複製セッションまたはデータベース・セッションのマイグレーション](#)
 - [WebSphere Application Server 管理コンソールの前の構成設定のメモ](#)
 - [データ・グリッド内におけるセッション管理用のカタログ・サービス・ドメインの作成](#)
 - [以前の構成設定を使用するための WebSphere DataPower XC10 アプライアンスの構成](#)
 - [ASP.NET セッション状態ストア・プロバイダーの構成](#)
 - [ASP.NET セッション状態ストア・プロバイダーで使用するデータ・グリッドの作成](#)
 - [ASP.NET セッション状態ストア・プロバイダーの構成](#)
 - [HTTP セッション・サイズを表示](#)
 - [サーブレット・コンテキスト初期化パラメーター](#)
 - [splicer.properties ファイル](#)
 - [動的キャッシュの作成データ・グリッド](#)
 - [動的キャッシュ・インスタンスの構成](#)
 - [デフォルト動的キャッシュ・インスタンス \(baseCache\) の構成](#)
 - [オブジェクトまたはサーブレットの動的キャッシュ・インスタンスの構成](#)
 - [カスタム・プロパティを使用した動的キャッシュ・インスタンスのカスタマイズ](#)

- [キャッシュ・インスタンス・プロパティ・ファイル](#)
 - [動的キャッシュ・カスタム・プロパティ](#)
 - [動的キャッシュ・プロバイダーの概要](#)
 - [WebSphere Application Server でのカタログ・サービス・ドメインの作成](#)
 - [カタログ・サービス・ドメイン管理用タスク](#)
 - [動的キャッシュのニア・キャッシュの構成](#)
- [データ・グリッドの最大容量の構成](#)
- [データ・グリッドのセキュリティーを使用可能に設定する](#)
- [データ・グリッドのクリア](#)
- [データ・グリッドの除去](#)
- [Spring キャッシュ・プロバイダーの構成](#)
- [クライアントの構成](#)
 - [Java クライアントの構成](#)
 - [Java クライアントのオーバーライド](#)
 - [XML 構成を使用した Java クライアントの構成](#)
 - [Java クライアントのプログラマチック構成](#)
 - [要求タイムアウト値および再試行タイムアウト値の構成](#)
 - [データ・グリッドをセッション管理に使用するための Liberty プロファイルの構成](#)
 - [SSL を有効にしてアクセスするデータ・グリッド用の Liberty プロファイルの構成](#)
 - [クライアントと一緒に実行する Liberty プロファイルの構成](#)
 - [Liberty プロファイルでの webApp フィーチャーの使用可能化](#)
 - [Liberty プロファイル xsWebApp フィーチャー・プロパティ](#)
 - [Liberty プロファイル xsDynacacheApp フィーチャー・プロパティ](#)
 - [REST ゲートウェイのデプロイ](#)
 - [WebSphere eXtreme Scale Client for .NET の構成](#)
 - [WebSphere eXtreme Scale Client for .NET オーバーライド](#)
 - [WebSphere eXtreme Scale Client for .NET のプログラマチックな構成](#)
 - [WebSphere eXtreme Scale Client for .NET の動的構成を使用可能に設定](#)
- [データ・グリッドの管理](#)
 - [データの照会、表示、および無効化](#)
 - [xscmd ユーティリティーによる管理](#)
 - [xscmd ユーティリティーのためのセキュリティー・プロファイルの構成](#)
 - [HTTP コマンド・インターフェースでの管理](#)
- [単純データ・グリッドにアクセスするアプリケーションの開発](#)
 - [Java アプリケーションの開発](#)
 - [Java API 資料へのアクセス](#)
 - [例: 単純データ・グリッド \(data grid\) ・アプリケーション](#)
 - [Java API を使用した動的マップの作成](#)
 - [トランザクションのためのプログラミング](#)
 - [トランザクション内のデータとの対話](#)
 - [単一のトランザクションで複数の区画を更新するアプリケーションの開発](#)
 - [2 フェーズ・コミットとエラー・リカバリー](#)
 - [スタンドアロン環境で WebSphere eXtreme Scale のマルチ区画トランザクションに書き込むアプリケーションの開発](#)
 - [トランザクションを使用する eXtreme Scale クライアント・コンポーネントの開発](#)
 - [ロックの使用](#)
 - [Java アプリケーションにおけるロックの構成と実装](#)
 - [ロックの Java 例外処理](#)
 - [例: flush メソッドのロック順序](#)
 - [Java のトランザクション分離の例](#)
 - [データの索引付けのためのプラグイン](#)
 - [索引によるデータへのアクセス \(索引 API\)](#)
 - [DynamicIndexCallback インターフェース](#)
 - [セッションを使用したグリッド内データへのアクセス](#)
 - [HashIndex プラグインの構成](#)
 - [HashIndex プラグイン属性](#)
 - [複合索引の使用](#)
 - [グローバル索引の使用](#)
 - [クライアントへの連続照会を使用したマップ更新の通知](#)

- [REST ゲートウェイを使用したデータ・グリッド・アプリケーションの開発](#)
 - [URI フォーマット](#)
 - [データ・フォーマット](#)
 - [REST 操作](#)
 - [例: データ・グリッドのマッピング項目の挿入と取得](#)
 - [例: Java クライアントから REST マッピングへのデータの挿入およびデータへのアクセス](#)
 - [例: データ・グリッドのマッピング項目のクリア](#)
 - [例: 動的マッピングの作成](#)
 - [例: TTL の有効期限](#)
 - [REST セキュリティー構成](#)
 - [REST HTTP セッションと Cookie](#)
- [.NET アプリケーションの開発](#)
 - [.NET 開発環境の設定](#)
 - [WebSphere eXtreme Scale Client for .NET API 資料へのアクセス](#)
 - [.NET API を使用した動的マッピングの作成](#)
 - [Java と .NET クラスを相関付けるための ClassAlias および FieldAlias 注釈の定義](#)
 - [ClassAlias および FieldAlias 注釈](#)
 - [PartitionKey 注釈を使用したキーから区画へのマッピング](#)
 - [トランザクションのためのプログラミング](#)
 - [トランザクション内のデータとの対話](#)
 - [.NET アプリケーションにおけるロックの構成と実装](#)
 - [ロックの .NET 例外処理](#)
 - [WebSphere eXtreme Scale Client for .NET 用のデータ・グリッド・セキュリティーの構成](#)
 - [WebSphere eXtreme Scale Client for .NET 用の TLS の構成](#)
 - [WebSphere eXtreme Scale Client for .NET のためのクライアント認証のプログラミング](#)
 - [例: .NET アプリケーション用のユーザー・パスワード資格情報の実装](#)
 - [例: .NET アプリケーション用のユーザー資格情報生成プログラムの実装](#)
 - [WebSphere eXtreme Scale Client for .NET のためのカスタム資格情報のプログラミング](#)
- [モニター](#)
 - [アプライアンスの始動状況のモニター](#)
 - [ユーザー・インターフェースでのデータ・グリッドのモニター](#)
 - [タスクによるアクティビティーのモニター](#)
 - [xscmd ユーティリティーによるモニター](#)
 - [CSV ファイルによるモニター](#)
 - [CSV ファイルの統計定義](#)
 - [xsadmin ユーティリティーによるモニター](#)
 - [xsadmin ユーティリティー・リファレンス](#)
 - [xsadmin ツールから xscmd ツールへのマイグレーション](#)
 - [環境のヘルスのモニター](#)
 - [ヘルス・モニターの概要](#)
 - [メッセージ・センターでのヘルス・イベント通知の表示](#)
 - [xscmd ユーティリティーを使用したヘルス通知の表示](#)
 - [SNMP によるモニター](#)
 - [アプライアンスでの SNMP モニターの使用可能化](#)
 - [SNMP コミュニティーの構成](#)
 - [SNMP トラップ・サブスクリバラーの構成](#)
 - [SNMP トラップ・サブスクリプションの管理](#)
 - [SNMP トラップ・リファレンス](#)
 - [リモート・ロギングの構成](#)
- [セキュリティー](#)
 - [IBM WebSphere DataPower XC10 アプライアンス セキュリティー概要](#)
 - [IBM WebSphere DataPower XC10 アプライアンスのユーザー・インターフェース・セキュリティーの構成](#)
 - [ユーザーおよびグループの管理](#)
 - [ユーザーの作成](#)
 - [ユーザーの管理](#)
 - [ユーザーの除去](#)
 - [新規ユーザー・アカウントの登録](#)

- [ユーザー・グループの作成](#)
 - [ユーザー・グループの管理](#)
 - [ユーザー・グループの除去](#)
 - [ユーザー許可](#)
- [データ・グリッドのセキュリティーを使用可能に設定する](#)
- [WebSphere Application Server 用の Transport Layer Security \(TLS\) の構成](#)
- [クライアント認証を使用するためのデータ・グリッド・アプリケーションの構成](#)
- [データ・グリッド・アプリケーション用の TLS の構成](#)
- [LDAP ディレクトリーを使用したユーザー認証のためのアプライアンスの構成](#)
- [REST セキュリティー構成](#)
- [トラブルシューティング](#)
 - [WebSphere DataPower XC10 アプライアンスのトラブルシューティングおよびサポート](#)
 - [問題のトラブルシューティングのための手法](#)
 - [知識ベースの検索](#)
 - [フィックスの入手](#)
 - [Fix Central からのフィックスの入手](#)
 - [IBM サポートへの連絡](#)
 - [IBM サポート用の診断情報の収集](#)
 - [IBM との情報の交換](#)
 - [サポート更新情報のサブスクライブ](#)
 - [WebSphere DataPower XC10 アプライアンス ログ・ファイルの操作](#)
 - [syslog を使用して WebSphere DataPower XC10 アプライアンス ログ・レコードをリモートの UNIX システムに送信する](#)
 - [監査データのダウンロード](#)
 - [ログおよびトレース・データの分析](#)
 - [ログ分析の概要](#)
 - [ログ・ファイルの保管ポリシー](#)
 - [ログ分析の実行](#)
 - [ログ分析用カスタム・スキャナーの作成](#)
 - [ログ分析のトラブルシューティング](#)
 - [ハードウェア温度のモニター](#)
 - [シリアル接続を使用したイーサネット・インターフェースの状況詳細のモニター](#)
 - [アプライアンスからのアウトバウンド接続の検査](#)
 - [コマンド行インターフェースを使用したアプライアンスでの操作の実行](#)
 - [デッドロックのトラブルシューティング](#)
 - [マルチ区画トランザクションのロック・タイムアウト例外のトラブルシューティング](#)
 - [ロック・タイムアウト例外の解決](#)
 - [キャッシュ統合のトラブルシューティング](#)
 - [インストールのトラブルシューティング](#)
 - [管理のトラブルシューティング](#)
 - [リリース情報](#)
- [参照](#)
 - [xscmd ユーティリティー・リファレンス](#)
 - [HTTP コマンド・インターフェースのリファレンス](#)
 - [クライアント・プロパティー・ファイル](#)
 - [正規表現の構文](#)
 - [動的マップの構成オプション](#)
 - [ユーザー・インターフェース設定](#)
 - [eXtreme Scale セッション管理設定](#)
 - [カタログ・サービス・ドメイン・コレクション](#)
 - [カタログ・サービス・ドメイン設定](#)
 - [クライアント・セキュリティー・プロパティー](#)
 - [カタログ・サービス・ドメインのカスタム・プロパティー](#)
- [サイト・マップ](#)