

**IBM solidDB
IBM solidDB Universal Cache
バージョン 6.5**

InfoSphere CDC レプリ ケーション・ユーザー・ガイド



注記

本書および本書で紹介する製品をご使用になる前に、117 ページの『特記事項』に記載されている情報をお読みください。

本書は、バージョン 6 リリース 5 の IBM solidDB (製品番号 5724-V17) および IBM solidDB Universal Cache (製品番号 5724-W91)、および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原典： SC23-9874-01
IBM solidDB
IBM solidDB Universal Cache
Version 6.5
Replication with InfoSphere CDC User
Guide

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

第1刷 2010.10

© Solid Information Technology Ltd. 1993, 2010

目次

図	v
表	vii
変更の要約	ix
本書について	xi
IBM solidDB 製品によるレプリケーションの概要	xi
規則	xii
書体の規則	xii
構文表記法の規則	xiii
1 InfoSphere CDC レプリケーションの概要	1
InfoSphere CDC レプリケーションのアーキテクチャの概要	1
構成	3
solidDB 高可用性 (HotStandby) での InfoSphere CDC レプリケーションの使用	4
機能	4
制限事項	6
データベースの制限事項	6
InfoSphere CDC の制限事項	7
セキュリティと認証	8
2 InfoSphere CDC レプリケーション・コンポーネントのインストール	9
3 InfoSphere CDC レプリケーションの構成	11
solidDBfor InfoSphere CDC レプリケーションの構成	11
InfoSphere CDC インスタンスの作成	13
レプリケーション・サブスクリプションのセットアップ	13
4 InfoSphere CDC 用の solidDB の固有の設定およびタスク	17
solidDB ソース表のドロップおよび再作成	17
外部キーの使用の有効化 (参照整合性)	17
データを同期化しないミラーリングの開始	18
InfoSphere CDC レプリケーションでの Unicode データベースおよび部分的 Unicode データベースの使用	18
高速リフレッシュの使用可能化	19
InfoSphere CDC レプリケーションでの共有メモリー・アクセス (SMA) の使用	20
5 データ・エージング	23
6 SQL パススルー	25

7 ツールおよびユーティリティ	27
Perl 自動化フレームワーク	28
インスタンスおよびサブスクリプション管理ツール	29
ucdeploy – 構成およびセットアップ・サンプル	29
ucpassthrough – SQL パススルー・セットアップ・サンプル	30
uchsbmonitor – HSB サブスクリプション・モニター・サンプル	30
データ・エージングとリフレッシュのための SQL ストアード・プロシージャ	31
エージング・ストアード・プロシージャの使用	31
リフレッシュ・ストアード・プロシージャの使用	34
例: 双方向サブスクリプションのためのデータ・エージングの自動化	37
8 InfoSphere CDC レプリケーションの障害シナリオ	41
スタンドアロン solidDB サーバーの障害	41
InfoSphere CDC インスタンスの障害	41
HA モード (HotStandby) の solidDB サーバーの障害	42
1 次 solidDB サーバーと InfoSphere CDC for solidDB インスタンス間の通信リンクの障害	43
9 トラブルシューティング	45
10 InfoSphere CDC for solidDB (エンド・ユーザー向け資料)	49
このセクションについて	49
InfoSphere CDC for IBM solidDB について	49
システム要件	49
必須のデータベース、ユーザー・アカウント、およびスキーマ	50
1 バイト文字およびマルチバイト文字のサポート	51
InfoSphere CDC のインストール	53
InfoSphere CDC の対話式インストール	53
InfoSphere CDC のサイレント・インストール	54
InfoSphere CDC の構成 (Windows)	54
InfoSphere CDC インスタンスの構成 (Windows)	55
InfoSphere CDC の構成 (UNIX および Linux)	58
InfoSphere CDC インスタンスの構成 (UNIX および Linux)	58
InfoSphere CDC の開始と停止	62
InfoSphere CDC の開始	62
InfoSphere CDC の停止	62
Management Console での SQL ステートメントの使用可能化	63
InfoSphere CDC がサポートするデータ型	64
サポートされているデータ型	64
サポートされているマッピング	65

InfoSphere CDC メタデータ表	66	ストアード・プロシージャ・ユーザー出口のデ ータベース接続	93
InfoSphere CDC のコマンド	67	ストアード・プロシージャ・ユーザー出口での データのリトリブ	93
InfoSphere CDC コマンドの使用	67	ストアード・プロシージャ・ユーザー出口の例	99
TSINSTANCE 環境変数の設定	68	InfoSphere CDC のサンプル・ユーザー出口 . . .	101
レプリケーション・コマンドの制御	68	競合解決監査表	103
データベース・トランザクション・ログ・コマン ド	71	Management Console 管理ガイドの付録	107
レプリケーション・コマンドに関する表の管理 . . .	75	ユーザー出口の構成	107
レプリケーション・コマンドのモニター	81	InfoSphere CDC for solidDB のシステム・パラメ ーター	109
構成コマンドのエクスポートとインポート	85		
その他のコマンド	86		
InfoSphere CDC のユーザー出口	92	特記事項.	117
表レベルおよび行レベルの操作のためのストア ード・プロシージャ・ユーザー出口	93		
ストアード・プロシージャ・ユーザー出口の定 義	93		



1. InfoSphere CDC レプリケーションのアーキテク
チャー 2

表

1. 書体の規則	xii	8. AUX_AUTOMATIC_DELETES_BREAK 表の定義	33
2. 構文表記法の規則	xiii	9. エージング・プロシージャを作成および実行するスクリプト	33
3. InfoSphere CDC レプリケーションの資料	9	10. TS_REFRESH 表の定義	35
4. dmconfigurets を使用した SMA 接続の使用可能化	21	11. ハードウェア要件	50
5. InfoSphere CDC for solidDB パッケージ - ツールおよびユーティリティー	27	12. ポート要件	50
6. solidDB パッケージ - ストアード・プロシージャ	28	13. 文字データ型列およびワイド文字データ型列のデフォルト・エンコード設定 (部分的 Unicode) および Unicode エンコード設定	52
7. AUX_AUTOMATIC_DELETES 表の定義	31		

変更の要約

改訂 01 での変更点

- 追加された新しいセクションは、以下のとおりです。
 - InfoSphere CDC レプリケーションでの共有メモリー・アクセス (SMA) の使用
 - InfoSphere CDC レプリケーションでの Unicode データベースおよび部分的 Unicode データベースの使用
 - 高速リフレッシュの使用可能化

本書について

InfoSphere™ CDC レプリケーションを使用して、ペアになっている IBM® solidDB® サーバー・インスタンス間でデータを複製できます。例えば、地理的に離れた場所での冗長度を実現すること、つまり、2 つの solidDB インスタンスをアクティブ/アクティブ・セットアップで構成して、同じデータの 2 つのコピーを両側で同じように処理することができます。

本書では、InfoSphere CDC レプリケーション・ソリューションの概要を示し、InfoSphere CDC レプリケーション・システムのインストールと構成について説明します。また、障害とトラブルシューティングのシナリオに対処するためのガイドラインも記載します。本書の『InfoSphere CDC for solidDB (エンド・ユーザー向け資料)』には、InfoSphere CDC for solidDB コンポーネントのインストールおよび構成方法に関する詳細な説明があります。このセクションは、InfoSphere CDC レプリケーション・システムの構成時に必要になります。その他のデータ・サーバーに関して、「InfoSphere Change Data Capture のエンド・ユーザー向け資料」ユーザー・マニュアルに対応する情報を提供します。

本書は、読者が、DBMS に関して一般的な知識を持っていること、また SQL と solidDB に精通していることを前提としています。

IBM solidDB 製品によるレプリケーションの概要

IBM solidDB 製品ファミリーでは、拡張レプリケーション、InfoSphere CDC レプリケーション、および solidDB 高可用性 (HotStandby) の 3 種類のテクノロジーを使用してデータ・レプリケーションを実装できます。

- **拡張レプリケーション** は、組み込みの SQL 拡張機能を使用しており、N+M トポロジーによる不定期またはイベント・ベースの非同期レプリケーションを対象としています。
- **InfoSphere CDC レプリケーション** は、トランザクション・ログの読み取りを使用しており、不定期または連続的なレプリケーション・フローを対象としています。また、異種混合環境をサポートします。
- **HotStandby レプリケーション** は、1+1 トポロジーを使用する、非常に高速なフェイルオーバーおよびリカバリーの機能を必要とする高可用性システムを対象としています。

規則

書体の規則

solidDB の資料では、以下の書体の規則を使用します。

表 1. 書体の規則

フォーマット	用途
データベース表	このフォントは、すべての通常テキストに使用します。
NOT NULL	このフォントの大文字は、SQL キーワードおよびマクロ名を示しています。
solid.ini	これらのフォントは、ファイル名とパス式を表しています。
SET SYNC MASTER YES; COMMIT WORK;	このフォントは、プログラム・コードとプログラム出力に使用します。SQL ステートメントの例にも、このフォントを使用します。
run.sh	このフォントは、サンプル・コマンド行に使用します。
TRIG_COUNT()	このフォントは、関数名に使用します。
java.sql.Connection	このフォントは、インターフェース名に使用します。
LockHashSize	このフォントは、パラメーター名、関数引数、および Windows® レジストリー項目に使用します。
<i>argument</i>	このように強調されたワードは、ユーザーまたはアプリケーションが指定すべき情報を示しています。
管理者ガイド	このスタイルは、他の資料、または同じ資料内の他の章の参照に使用します。新しい用語や強調事項もこのように記述します。
ファイル・パス表示	特に明記していない場合、ファイル・パスは UNIX® フォーマットで示します。スラッシュ (/) 文字は、インストール・ルート・ディレクトリーを表します。
オペレーティング・システム	資料にオペレーティング・システムによる違いがある場合は、最初に UNIX フォーマットで記載します。UNIX フォーマットに続いて、小括弧内に Microsoft® Windows フォーマットで記載します。その他のオペレーティング・システムについては、別途記載します。異なるオペレーティング・システムに対して、別の章を設ける場合があります。

構文表記法の規則

solidDB の資料では、以下の構文表記法の規則を使用します。

表 2. 構文表記法の規則

フォーマット	用途
<code>INSERT INTO table_name</code>	構文の記述には、このフォントを使用します。置き換え可能セクションには、このフォントを使用します。
<code>solid.ini</code>	このフォントは、ファイル名とパス式を表しています。
[]	大括弧は、オプション項目を示します。太字テキストの場合には、大括弧は構文に組み込む必要があります。
	垂直バーは、構文行で、互いに排他的な選択項目を分離します。
{ }	中括弧は、構文行で互いに排他的な選択項目を区切ります。太字テキストの場合には、中括弧は構文に組み込む必要があります。
...	省略符号は、引数が複数回繰り返し可能なことを示します。
· · ·	3 つのドットの列は、直前のコード行が継続することを示します。

1 InfoSphere CDC レプリケーションの概要

IBM InfoSphere Change Data Capture (InfoSphere CDC) テクノロジーを使用すると、ペアになっている solidDB サーバー間で非同期レプリケーションを実装できます。InfoSphere CDC テクノロジーは、非同期プッシュ・モデルに基づいています。データ変更のリアルタイム伝搬のために、ソース側からターゲット側に向けて単一方向サブスクリプションを作成できます。また、ミラーリングされたソースとターゲットの定義で 2 つのサブスクリプションを設定することにより、双方向機能を実現できます。

InfoSphere CDC レプリケーション という用語は、InfoSphere CDC テクノロジーを使用する solidDB 間のレプリケーション・モデルを意味しています。solidDB Universal Cache でも、同じテクノロジーを使用して、solidDB フロントエンド・キャッシュをセットアップします。これにより、バックエンド・データベース・サーバーに格納されている、パフォーマンスが重要な意味を持つデータへのアクセスを高速化します。

InfoSphere CDC レプリケーションのアーキテクチャーの概要

以下の図は、標準的構成の InfoSphere CDC レプリケーション・セットアップのアーキテクチャーとキー・コンポーネントを示したものです。

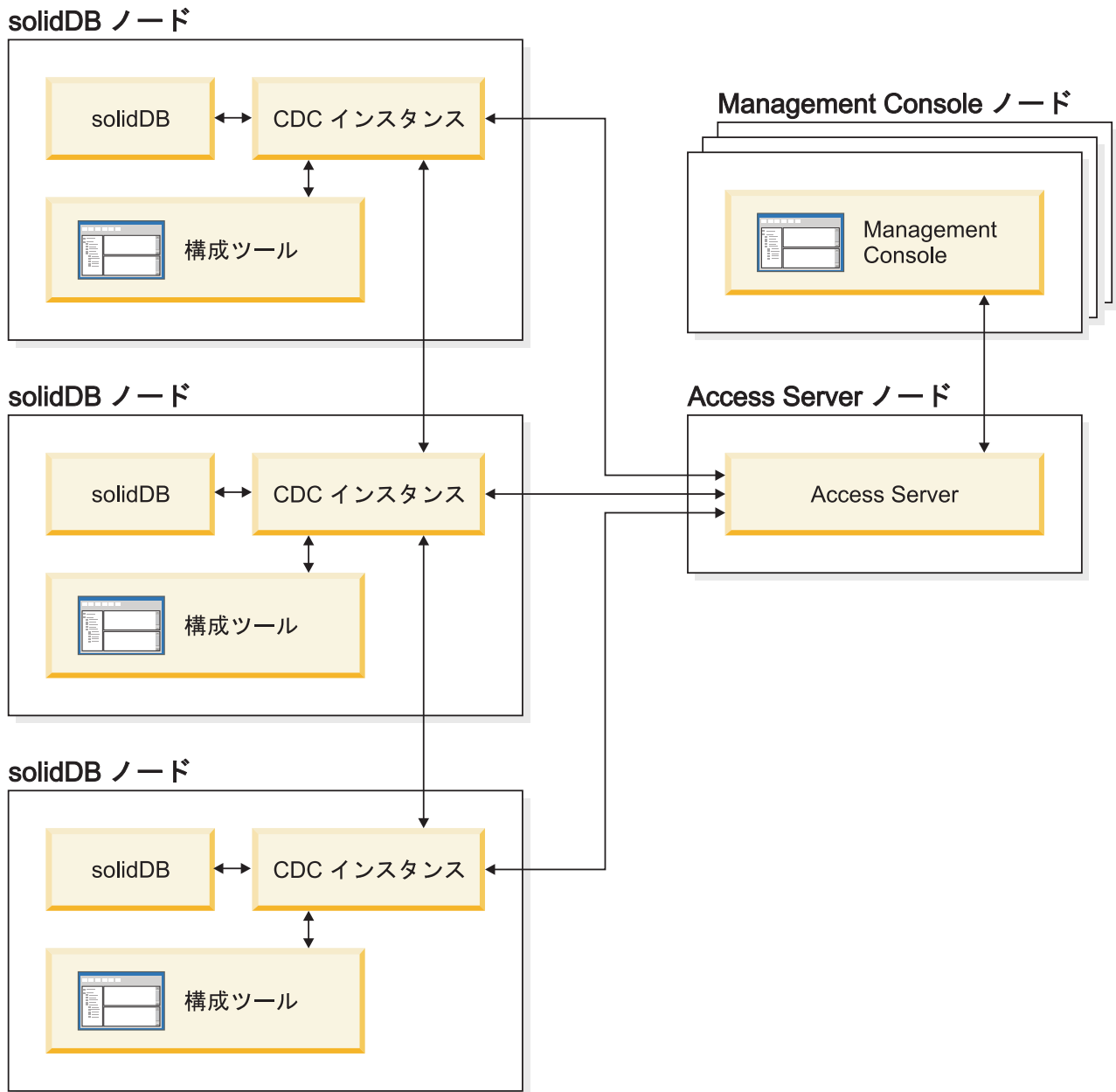


図 1. InfoSphere CDC レプリケーションのアーキテクチャー

コンポーネントの役割と機能を以下に説明します。

solidDB: データの複製先または複製元のデータベース。

InfoSphere CDC インスタンス: solidDB 用の InfoSphere CDC エンジンのランタイム・インスタンス。これらの InfoSphere CDC インスタンスは、solidDB サーバーが稼働しているノードであれば、同じノードにも別のノードにもインストールできます。

構成ツール: InfoSphere CDC インスタンスの構成と作成に使用する GUI ベースまたはコンソール・ベースのツール。インスタンスの構成中に、InfoSphere CDC システムの他の部分との通信のためのポート番号、およびデータベースに接続するためのログイン情報を入力します。

Access Server: Management Console ユーザーが InfoSphere CDC インスタンスにアクセスし、構成するためのプロセス (一般にサービスまたはデーモンとして稼働)。さまざまなユーザーが、さまざまなインスタンスにアクセスできます。Access Server のインストール中に、Management Console で使用するポート番号および管理者ログイン情報の入力を要求されます。

Management Console: レプリケーションの構成とモニターに使用できる GUI を備えた対話式アプリケーション。各種のサーバー上でのレプリケーションの管理、レプリケーション・パラメーターの指定、クライアント・ワークステーションからのリフレッシュ操作とミラーリング操作の開始が行えます。

アクセス・マネージャー・パースペクティブでデータ・ストアを作成することにより、Management Console の使用を開始します。データ・ストアは、データベースとそれに関連する InfoSphere CDC インスタンスの論理エンティティです。定義されたデータ・ストアを使用することにより、1 つのデータ・ストアから別のデータ・ストアへのデータ・レプリケーションを実現するサブスクリプションをセットアップできます。

データ・ストアには、ソース、ターゲット、およびデュアルという 3 つのタイプがあります。デュアル・データ・ストアは、ソースおよびターゲットの両方としてサブスクリプションに関係させることができます。通常、デュアル・データ・ストアは、solidDB Universal Cache で使用され、対称マッピング・ペアを双方向レプリケーションに定義します。レプリカがフロントエンド・データ・サーバーで読み取り専用の場合は、単一のアップロード・サブスクリプションを定義します。

レプリケーションのセットアップ完了後、ソースとターゲットのサーバー間のアクティブなデータ・レプリケーション・アクティビティに影響を及ぼすことなく、クライアント・ワークステーション上で Management Console を閉じることができます。Management Console にはイベント・ログとモニターも含まれています。イベント・ログでは、生成された InfoSphere CDC イベント・メッセージを確認できます。モニターは、レプリケーション操作と待ち時間を継続的にモニターするために必要なサポートを提供します。レプリケーション構成のコンポーネントを示す図は、グラフィカル・オブジェクトの直接操作によって構成されます。Management Console のモニターは、データの移動を継続的に分析する必要がある、時間制限の厳しい作業環境で使用するためのものです。

構成

InfoSphere CDC テクノロジーを使用すると、いろいろなタイプの solidDB 間のレプリケーション・モデルを実装できます。

デプロイメント・トポロジーの例については、2 ページの図 1 を参照してください。

また、その他の実装も可能です。例えば、InfoSphere CDC テクノロジーを使用して、地理的に離れた場所での冗長度、2 つ以上のサイトを持つ構成、複数のアクティブ・データ・サーバーのシナリオ、および複数の 2 次データ・サーバーを持つトポロジーを実装することができます。

InfoSphere CDC レプリケーションは、solidDB HotStandby (HSB) でも使用できます。

レプリケーション・モデルの設計時には、以下のガイドラインを使用できます。

- InfoSphere CDC for solidDB および solidDB データベースは、同じ場所に配置する必要はありません。
- solidDB HSB ペアの使用時には、対応する InfoSphere CDC インスタンスを別のノードに配置する必要があります。
- システム全体の可用性を検討する際には、41 ページの『8 章 InfoSphere CDC レプリケーションの障害シナリオ』の説明に従って、InfoSphere CDC 障害の管理方法を計画してください。
- Management Console は、常に別のノードで実行する必要があります。

solidDB 高可用性 (HotStandby) での InfoSphere CDC レプリケーションの使用

InfoSphere CDC レプリケーションは、solidDB 高可用性 (HA または HotStandby) と一緒に使用できます。

solidDB HA (HotStandby) で InfoSphere CDC レプリケーションの使用を開始するには、以下の点を考慮する必要があります。

- 新しい InfoSphere CDC for solidDB インスタンスを作成する場合は、1 次および 2 次の solidDB サーバーのホスト・アドレスとポート番号を定義する必要があります。
- solidDB HotStandby セットアップの構成方法については、「*IBM solidDB 高可用性ユーザー・ガイド*」を参照してください。
- 環境内での障害シナリオにどのように対処するかを計画し、実装しておいてください。詳しくは、41 ページの『8 章 InfoSphere CDC レプリケーションの障害シナリオ』のセクションおよび「*IBM solidDB 高可用性ユーザー・ガイド*」を参照してください。

機能

InfoSphere CDC テクノロジーをデプロイする構成では、以下の solidDB 機能が使用可能です。

- 共有メモリー・アクセス (SMA) およびリンク・ライブラリー・アクセス (LLA)

アプリケーションを作成して、共有メモリー・アクセス・ドライバー・ライブラリーまたはリンク・ライブラリー・アクセス・ライブラリーにリンクすると、インメモリー・データベースのパフォーマンスと反応性をフル活用することができます。

詳しくは、*IBM solidDB 共有メモリー・アクセスおよびリンク・ライブラリー・アクセス・ユーザー・ガイド*を参照してください。

- SQL パススルー

solidDB Universal Cache で、SQL パススルーを使用すると、アプリケーションからフロントエンドとバックエンドの両方のデータ・サーバーのデータに単一接続

でアクセスできます。例えば、SQL パススルーを有効にして、solidDB フロントエンドでは実行できない SQL ステートメントをバックエンドに渡すように設定することができます。

- **アプリケーション主導型データ・エージング**

solidDB Universal Cache では、データ・エージング機能により、アプリケーションで solidDB フロントエンド・データベースから古くなったデータを削除する一方、バックエンドでそのデータを保持することができます。データ・エージングを使用すると、solidDB フロントエンドのメイン・メモリーの使用量を制御することができます。つまり、solidDB フロントエンドのデータの一部が廃止された場合、および、アプリケーションで使用されなくなった場合に、そのデータを削除できます。廃止データを削除すると、フロントエンドのメイン・メモリー・リソースが解放されます。

- **高速リフレッシュ**

高速リフレッシュ機能により、バックエンド・データ・サーバーから solidDB フロントエンドへの大容量のデータの複製にかかる時間が削減されます。大部分のパフォーマンス向上は、簡単なセットアップで実現されます。例えば、表あたり行あたりのデータ量、コード・ページ変換、および列マッピングなどの要因が、高速リフレッシュ機能のパフォーマンスに影響を及ぼします。

高速リフレッシュは、以下の InfoSphere CDC 機能をサポートしません。

- 競合検出
- 要約
- 行の統合
- アダプティブ・アプライ
- ユーザー出口

- **solidDB の高可用性サポート**

solidDB 高可用性機能 (solidDB サーバーでのフェイルオーバーを含む) はサポートされます。詳しくは、「*IBM solidDB 高可用性ユーザー・ガイド*」を参照してください。

- **スロットル**

レプリケーションによって、solidDB サーバーでの持続的な負荷に対応できない場合は、処理が減速 (低速化) します。アプリケーションの観点からいうと、これは応答時間が長くなることを意味しています。レプリケーション・トラフィックがバッファーに入れられ、負荷のバーストに対応できるようにします。対応するインメモリー・バッファーのサイズは、構成パラメーター **LogReader.MaxSpace** を使用して制御します。

- **オフライン操作とログのオーバーフロー**

レプリケーションが停止、あるいはレプリケーションに失敗しても、solidDB サーバーは負荷の処理を続行し、後で転送するためにデータを蓄積することができます。蓄積するデータの限度は、構成パラメーター **LogReader.MaxLogSize** を使用して設定します。蓄積したデータの量が **LogReader.MaxLogSize** パラメーター

の値を超えると、ログのオーバーフローが発生し、それ以降はレプリケーション・キャッチアップが実行できない状態になります。その場合は、サブスクリプションのリフレッシュが必要です。

- **ログ・リーダーの診断**

solidDB のキャプチャー・プロセス (solidDB をソースとして) およびアプライ・プロセス (solidDB をターゲットとして) の操作をモニターするために、いくつかのランタイム・カウンターが使用可能です。セクション『データベースの制限事項』の LOBS 関連の制限事項も参照してください。

- **データ型のサポート**

レプリケーションでは、solidDB のすべてのデータ型がサポートされています。詳しくは、64 ページの『サポートされているデータ型』のセクションを参照してください。

制限事項

データベースの制限事項

ソースおよびターゲットのデータ・サーバーとして、solidDB には以下の制限事項が適用されます。

- **参照整合性** (solidDB をソースおよびターゲットとして)

連続ミラーリングの場合、ソースとターゲットの両方に対して参照整合性制約 (外部キー) を使用できます。必須要件は、参照整合性関連がサブスクリプション内に限定されていること、つまり外部キーがサブスクリプション外の表を指さないことです。このルールに違反すると、ミラーリング中にターゲットで参照整合性エラーが発生し、レプリケーション・サブスクリプションが終了することがあります。

参照整合性は、表の自動作成ではサポートされていません。予備手段については、『7 ページの『InfoSphere CDC の制限事項』』のセクションを参照してください。

- **データ型のサポート**

- ディスク・ベース表の LOB はサポートされない (solidDB をソースとして)

使用可能なサイズ制限内のインメモリー表で維持しているすべての LOB は miniLOB と呼ばれ、許可されます。サイズ制限は、行サイズとブロック・サイズによって異なります。1 行に 1 つの LOB と想定すると、サイズ制限はブロック・サイズに近くなります。ブロック・サイズが 32 KB に設定されている場合は、miniLOB の実際のサイズ制限は約 30 KB になります。

solidDB のディスク・ベース表の大きなサイズ (最大 2 GB) の LOB (maxiLOB) は、ソースではサポートされていません。ログ・リーダー・パーティションの一部であるディスク・ベース表に maxiLOB を書き込もうとすると失敗し、アプリケーションにエラーが返されるという方法で、この制限事項は実施されます。

- LOB サポートの制限 (solidDB をターゲットとして)

LOB がインメモリ表に書き込まれ、miniLOB サイズ制限を超えた場合は、エラーが返され、サブスクリプションのレプリケーションが終了します。

- **TRUNCATE** (solidDB をソースとして)

サブスクリプションの一部となっている表には、TRUNCATE TABLE ステートメントは許可されません。このルールに違反すると、アプリケーションにエラーが返されます。

- **主キー制約** (solidDB をソースとして)

主キーは推奨されていますが、必須ではありません。表に主キーが定義されていない場合、主キーが定義されている場合と比較して、挿入と更新の実行効率が低下します。主キーの更新は、以下のように制限されます。

- 単一の列に主キーが定義されている場合、主キーの複数行の更新は許可されません。
- 複数列の主キーの場合、主キーの一部にのみ影響する場合に限って、複数行の更新が許可されます。

上記のルールのいずれかに違反すると、エラーが発生し、サブスクリプションのレプリケーション (ミラーリング) が終了します。

- **トランジエント表とテンポラリー表** (solidDB をソースとして)

solidDB をソースとして使用すると、トランジエント表およびテンポラリー表をサブスクリプションの一部とすることはできません。

- **UNIQUE 列の複数の NULL** (solidDB をターゲットとして)

solidDB では、UNIQUE として定義された列は、NULL インスタンスを 1 つだけ含むことができます。さらに NULL の挿入の伝搬を試行すると、UNIQUE 制約違反が発生し、サブスクリプションのレプリケーション (ミラーリング) が終了します。

- **複数の solidDB サーバーを使用したデータとワークロードのパーティション化**

複数の solidDB サーバーを使用して、データとワークロードをパーティション化できます。例えば、フロントエンドの複数の solidDB サーバー上でバックエンド・データを分散 (パーティション化) することができます。ただし、各 solidDB サーバーは自律型であり、他の solidDB サーバー (パーティション) のデータにアクセスすることなく、保持しているパーティション内でのみアプリケーション要求を処理します。

参照整合性制約も適用されるため、パーティションには、パーティションの外側との参照を行う表を収めることはできません。

InfoSphere CDC の制限事項

他のデータ・サーバー用に InfoSphere CDC コンポーネントで使用可能な以下の機能は、InfoSphere CDC for solidDB ではサポートされません。

- **リフレッシュ用の高速ロード**

InfoSphere CDC for solidDB は、リフレッシュ用の高速ロード機能をサポートしていません。

- **ターゲット表の自動作成**

ミラーリングされる表が参照整合性制約に関連付けられているときは、新しいサブスクリプションの定義時に、ターゲット表の作成を自動的に行うオプション（「**Create new target tables**」）を使用することはできません。代わりに、オプション「**Map to existing tables**」を使用します。このルールに違反すると、サブスクリプションは作成されません。

この制限は、他の DBMS 製品も含め、すべての構成に適用されます。

- **行フィルター**

行フィルター（水平パーティショニング）が完全に機能するのは、主キーがソース表で定義されている場合だけです。

- **solidDB がソース・データ・ストアである場合の、表のドロップと再作成**

solidDB がソース・データ・ストアであるサブスクリプションの表をドロップして再作成する必要がある場合は、表マッピングを再構成する必要があります。詳しくは、『17 ページの『solidDB ソース表のドロップおよび再作成』』を参照してください。

セキュリティと認証

レプリケーションに関するデータベースは、通常の利用者名/パスワードのメカニズムによって保護されます。

- InfoSphere CDC インスタンスによるデータベース・アクセスに使用するユーザー名には、データとメタデータのアクセスと変更に関して、必要なすべての資格情報を与えておく必要があります。データベース管理者のユーザー名を使用することを推奨します。
- InfoSphere CDC システムは、InfoSphere CDC インスタンスとそれぞれのパーティシスタント・リポジトリに、ユーザー名とパスワードを格納します。格納する認証データにはスクランブルがかけられます。つまり、弱い暗号化方式で暗号化されます。
- InfoSphere CDC インスタンスとデータベース間のトラフィックは、どのような場合においても暗号化されます。暗号化は、製品固有の JDBC ドライバーで提供されるか、使用可能です。
- InfoSphere CDC インスタンス間のトラフィックは暗号化されません。

2 InfoSphere CDC レプリケーション・コンポーネントのインストール

InfoSphere CDC レプリケーションをデプロイするには、solidDB サーバーおよび関連する InfoSphere CDC コンポーネントをインストールする必要があります。以下で、インストールの概要と、詳細なインストールの説明の参照箇所を示します。

1. IBM solidDB をインストールします。

詳しくは、「スタートアップ・ガイド」の『solidDB のインストール』のセクションを参照してください。

2. InfoSphere CDC for IBM solidDB をインストールします。

詳しくは、53 ページの『InfoSphere CDC のインストール』のセクションを参照してください。

注: インストールの終わりに、インストーラーが、新しい InfoSphere CDC インスタンスを作成するために、構成ツールの開始を可能にします。11 ページの『solidDBfor InfoSphere CDC レプリケーションの構成』の説明に従って solidDB 構成手順を完了していない場合には、構成ツールの開始を選択しないでください。

3. InfoSphere CDC Access Server をインストールします。

詳しくは、「InfoSphere Change Data Capture Access Server および Management Console のインストレーション・ガイド」を参照してください。

4. InfoSphere CDC Management Console をインストールします。

詳しくは、「InfoSphere Change Data Capture Access Server および Management Console のインストレーション・ガイド」を参照してください。

5. 以下の表にリストする資料にアクセスできることを確認してください。

表 3. InfoSphere CDC レプリケーションの資料

コンポーネント	資料の場所	主なドキュメント
InfoSphere CDC for solidDB	セクション『InfoSphere CDC for solidDB (エンド・ユーザー向け資料)』を参照してください。	「IBM solidDB InfoSphere CDC レプリケーション・ユーザー・ガイド」のセクション『InfoSphere CDC for solidDB (エンド・ユーザー向け資料)』
InfoSphere CDC Management Console	InfoSphere Change Data Capture Documentation パッケージ	「InfoSphere Change Data Capture Management Console 管理ガイド」
InfoSphere CDC Access Server	独自の資料はありません。Management Console の資料を参照してください。	独自の資料はありません。Management Console の資料を参照してください。

3 InfoSphere CDC レプリケーションの構成

InfoSphere CDC レプリケーション・システムを構成するには、レプリケーションで使用するデータベースを含む solidDB サーバーを最初に構成し、次に solidDB サーバーに対応する InfoSphere CDC インスタンスを作成し、最後にデータベース間のレプリケーション・サブスクリプションをセットアップします。

solidDBfor InfoSphere CDC レプリケーションの構成

InfoSphere CDC テクノロジーで solidDB を使用するには、InfoSphere CDC for solidDB が solidDB データベースに接続して、データを複製できるように、構成の設定を変更する必要があります。

始める前に

このセクションでは、読者が solidDB の管理に習熟しており、例えば、「*IBM solidDB 管理者ガイド*」の『*solidDB の管理*』、および『*solidDB の構成*』のセクションを読了していることを想定しています。

手順

1. データベース環境をセットアップするために、作業ディレクトリー、solidDB データベース、およびユーザー・アカウントをセットアップします。

手順については、「*IBM solidDB 管理者ガイド*」の『データベースの新規作成』を参照してください。

ヒント:

solidDB をインストールした後、solidDB インストール・ディレクトリーに以下のディレクトリーがあります。

```
<installation directory>
  bin¥
  ..
  eval_kit¥
    standalone¥
    cdc¥
  ..
  samples
  ..
```

solidDB インストール・ディレクトリー内の eval_kit/cdc ディレクトリーを作業ディレクトリーとして使用できます。ここには、solidDB を solidDB Universal Cache または InfoSphere CDC レプリケーションと一緒に使用するためのサンプルの solid.ini ファイルが入っています。

2. **solid.ini** 構成ファイルの **LogReader** セクションにある構成パラメーターを変更することにより、ログ・リーダーを構成します。
 - a. **LogReaderEnabled** 構成パラメーターを「yes」に設定します。

```
[LogReader]
LogReaderEnabled=yes
```

これは、InfoSphere CDC レプリケーションで solidDB をソース・データベースとして使用するために必要です。ファクトリー値は「no」です。

- b. **MaxLogSize** 構成パラメーターで、トランザクション・ログの保存スペースのサイズを設定します。

```
[LogReader]
MaxLogSize=<MB>
```

MaxLogSize パラメーターは、キャッチアップの実行に使用可能なログ・ファイルの量 (サイズ) を設定します。ログ・ファイルの最大サイズは、使用可能なディスク・スペースおよびキャッチアップが必要になるまでのダウン時間によって異なります。ファクトリー値は 10240 (10 GB) です。

ログ・リーダーが有効な場合、指定されたログ・ファイル保存スペースは、常にいっぱいまで使用されます。バックアップが実行されていない場合、またはパラメーター **CheckpointDeleteLog** が「Yes」に設定されていない場合には、ログ・ファイルはより大きなスペースを占有することがあります。

- c. **MaxSpace** 構成パラメーターを指定して、ログ・レコードのインメモリー・バッファー・サイズを設定します。

```
[LogReader]
MaxSpace=<ログ・レコード数>
```

MaxSpace パラメーターは、スロットルで使用されるインメモリー・ログ・リーダー・バッファーのサイズを (ログ・レコードの数で) 設定します。ログ・レコードの最大数は、予想される負荷バースト・サイズによって異なります。ファクトリー値は、100000 ログ・レコードです。

ログ・レコードのサイズは、(バイナリーの) 行サイズに追加メタデータ・オーバーヘッドの数バイトを加算したものになります。バッファーがいっぱいになると、スループット・スロットルが適用され、ログ・リーダー・バッファーに空きができるまで、操作がブロックされます。

3. 必要に応じて、パフォーマンスとデータベース・セットアップに関連するその他の構成パラメーターを変更します。

- **Logging.DurabilityLevel**

デフォルトでは、solidDB サーバーの持続性レベルはリラックス (**Logging.DurabilityLevel=1**) に設定されています。この場合、solidDB サーバーに予期しない障害が発生すると、最新のトランザクションが失われる状態になる可能性があります。

これを防止するために、以下の solid.ini ファイルの設定で、持続性レベルをストリクトに設定します。

```
[Logging]
DurabilityLevel=3
```

注: ストリクト持続性設定では、リラックス持続性と比較して、パフォーマンスで不利な条件が生じます。solidDB HA (HotStandby) 構成が 2-Safe レプリケーション・プロトコル (デフォルト) で適用されている場合、リラックス持続性はデータ損失のリスクなしに使用できます。

- **General.DefaultStoreIsMemory**

デフォルトでは、solidDB 表のストレージ・タイプはインメモリー表 (**General.DefaultStoreIsMemory=yes**) に設定されています。

- **Sql.IsolationLevel**

デフォルトでは、solidDB 分離レベルは、READ COMMITTED (**Sql.IsolationLevel=1**) に設定されています。

InfoSphere CDC インスタンスの作成

InfoSphere CDC インスタンスは、InfoSphere CDC 構成ツールで作成されます。

始める前に

- solidDB データベースが稼働していることを確認します。
- データベースに対する十分なアクセス権を取得しておきます。

手順

InfoSphere CDC for solidDB の新しいインスタンスを作成します。

詳しくは、54 ページの『InfoSphere CDC の構成 (Windows)』または 58 ページの『InfoSphere CDC の構成 (UNIX および Linux)』のセクションを参照してください。

注: ご使用の構成が solidDB 高可用性をデプロイする場合には、1 次および 2 次 solidDB サーバーに関してホスト・アドレスとポート番号を定義する必要があります。

レプリケーション・サブスクリプションのセットアップ

レプリケーション・サブスクリプションは、Management Console で作成されます。このセクションでは、作成処理の概要と、「*InfoSphere Change Data Capture Management Console 管理ガイド*」での詳しい説明の参照個所を示します。

始める前に

- 複製する表が solidDB サーバーに存在することを確認します。また、表に外部キーが含まれない場合には、レプリケーション中に表を作成することもできます。
- solidDB データベースが稼働していることを確認します。
- solidDB サーバー用の InfoSphere CDC インスタンスが稼働していることを確認します。
- データベースに対する十分なアクセス権を取得しておきます。
- ビジネス・ルールに従って、望ましいレプリケーション原則を定義しておきます。

手順

1. **Management Console** にログインし、**Access Server** に接続します。

詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『*Management Console* へのログイン (*Access Server* への接続)』のセクションを参照してください。

ヒント: Management Console のアクセス・マネージャー・パースペクティブで作業するには、データ・ストアとユーザー・アカウントの管理特権を持つシステム管理者でなければなりません。システム管理者のアカウントは、Management Console のインストール時に作成済みです。

2. データ・ストアをセットアップします。

- a. 新しいデータ・ストアを追加します。詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『新規データ・ストアを追加する』のセクションを参照してください。
- b. 接続パラメーターを設定します。詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『データ・ストアの接続パラメーターを設定する』のセクションを参照してください。

3. ユーザーにデータ・ストアを割り当てます。

すべてのデータ・ストアを同じユーザーに割り当てる必要があります。

- a. 必要に応じて、新しいユーザーを追加します。新しいユーザーを追加するには、ユーザー・アカウントの管理特権を持つシステム管理者でなければなりません。詳しくは、「*Management Console 管理ガイド*」の『ユーザー・アカウントの管理』のセクションを参照してください。
 - b. ユーザーにデータ・ストアを割り当てます。詳しくは、「*Management Console 管理ガイド*」の『データ・ストアへのユーザーの割り当て』のセクションを参照してください。
 - c. 変更を有効にするには、Access Server から切断して、再接続します。
 - 1) 「File」 > 「Access Server」 > 「Disconnect」をクリックします。
 - 2) 「File」 > 「Access Server」 > 「Connect」をクリックします。
4. オプションで、データ・ストアへの接続用の接続設定を行います。デフォルト設定をそのまま使用することもできます。

詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『接続の設定』のセクションを参照してください。

5. レプリケーション用のデータ・ストアをセットアップし、そのデータ・ストアに接続します。詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『レプリケーション用データ・ストアのセットアップ』のセクションを参照してください。

6. オプションで、データ・ストア上でシステム・パラメーターを設定します。

詳しくは、「*Management Console 管理ガイド*」の『ソース・データ・ストアおよびターゲット・データ・ストアでのシステム・パラメーターの設定』のセクションを参照してください。

7. レプリケーション・ニーズに従って、サブスクリプションをセットアップします。

例えば、ノード A とノード B の双方向レプリケーションの場合には、以下の手順を実行します。

- a. ノード A がソースで、ノード B がターゲットのサブスクリプションを追加します。

- b. ノード B がソースで、ノード A がターゲットのサブスクリプションを追加します。

詳しくは、「*InfoSphere Change Data Capture Management Console* 管理ガイド」の『サブスクリプションのセットアップ』のセクションを参照してください。

8. すべてのサブスクリプションで、レプリケーション用の表をマップします。

詳しくは、「*InfoSphere Change Data Capture Management Console* 管理ガイド」の『表のマッピング』のセクションを参照してください。

9. 表マッピングごとに、ビジネス・ルールに従って競合検出と解決を設定します。

詳しくは、「*InfoSphere Change Data Capture Management Console* 管理ガイド」の『競合検出および解決の設定』のセクションを参照してください。

10. すべてのサブスクリプションでレプリケーションを開始します。

詳しくは、「*InfoSphere Change Data Capture Management Console* 管理ガイド」の『サブスクリプションでのレプリケーションの開始と終了』のセクションを参照してください。

4 InfoSphere CDC 用の solidDB の固有の設定およびタスク

このセクションでは、InfoSphere CDC テクノロジーの solidDB での使用に固有の説明を行います。InfoSphere CDC インスタンスおよびレプリケーション・サブスクリプションの管理に関する一般的な説明は、「*InfoSphere Change Data Capture Management Console 管理ガイド*」に記載されています。

solidDB ソース表のドロップおよび再作成

solidDB がソース・データ・ストアであるサブスクリプションの表をドロップして再作成する必要がある場合は、表マッピングを再構成する必要があります。

手順

1. solidDB がソース・データ・ストアであるサブスクリプションで、レプリケーションを停止します。
2. ソース表を再マップします。
3. サブスクリプションのレプリケーション (ミラーリング) を再始動します。

表をマップしてサブスクリプションを開始および停止する方法については、「*InfoSphere Change Data Capture Management Console 管理ガイド*」を参照してください。

外部キーの使用の有効化 (参照整合性)

サブスクリプションに外部キーを持つ表が含まれている場合、InfoSphere CDC for solidDB のシステム・パラメーター `refresh_with_referential_integrity` を「true」に設定する必要があります。

このタスクについて

以下を使用して、システム・パラメーターを設定できます。

- `dmset -I <INSTANCE_NAME> <parameter_name>=<parameter_value>` コマンド
- または、Management Console を使用して以下を行います。
 1. Management Console の「**Configuration**」パースペクティブで、データ・ストアを選択します。
 2. データ・ストアを右クリックし、「**Properties**」 → 「**System Parameters**」を選択します。

アクティブ・レプリケーション時にシステム・パラメーターを変更した場合は、変更を有効にするために、レプリケーションを停止して再始動する必要があります。

手順

InfoSphere CDC for solidDB のシステム・パラメーター `refresh_with_referential_integrity` を「true」に設定します。例えば、以下のようにします。

データを同期化しないミラーリングの開始

サブスクリプションに対するミラーリングを開始すると、レプリケーション方式が「Mirror」で、状況が「Refresh」のすべての表は、最初にサブスクリプションでリフレッシュされます。これにより、ソース表とターゲット表が同期します。ミラーリングを開始するキャプチャー・ポイントを手動で設定することにより、それができます。」→「ミラーリングを開始するキャプチャー・ポイントを手動で設定します。これは、例えば、既にフロントエンドとバックエンドのデータベースが同期していることが分かっている場合などに便利です。サブスクリプションに多数のデータが含まれている場合は、リフレッシュせずにミラーリングを開始すると、時間の節約になります。

手順

1. ソース表を含んでいるサブスクリプションに対するアクティブ・レプリケーションが終了していることを確認します。
2. Management Console または `dmmarktablecapturepoint` コマンドを使用して、表キャプチャー・ポイントにマークを付けます。
 - Management Console を使用して表キャプチャー・ポイントにマークを付ける方法については、「InfoSphere Change Data Capture Management Console 管理ガイド」の『ソース表での表キャプチャー・ポイントのマーキング』のセクションを参照してください。
 - `dmmarktablecapturepoint` コマンドの使用方法については、77 ページの『`dmmarktablecapturepoint`: ソース表に表キャプチャー・ポイントのマークを付ける』のセクションを参照してください。

InfoSphere CDC レプリケーションでの Unicode データベースおよび部分的 Unicode データベースの使用

solidDB のデータベース・モード (Unicode または部分的 Unicode) に応じて、文字データ型 (CHAR、VARCHAR など) の列のエンコードを指定する必要が生じることがあります。

このタスクについて

- solidDB のデータベース・モードが *Unicode* (`General.InternalCharEncoding=UTF8`) の場合、solidDB 文字データ型 (CHAR、VARCHAR など) の列のエンコードを UTF-8 に設定します。
- solidDB データベース・モードが *部分的 Unicode* (`General.InternalCharEncoding=Raw`) の場合、solidDB 文字データ型 (CHAR、VARCHAR など) の列のエンコードを、アプリケーション環境で使用されているエンコードに設定します。

重要: デフォルトで、文字データ型列のエンコードは、ISO Latin1 に設定されます。アプリケーションで Latin1 エンコードが使用されている場合、エンコードを明示的に設定する必要はありません。

手順

1. Management Console で、「**Configuration**」 → 「**Subscriptions**」とクリックします。
2. サブスクリプションを選択します。
3. 「**Table Mappings**」ビューをクリックし、表マッピングを選択します。
4. 「**Edit Mapping Details**」を右クリックし、選択します。
5. 「**Translation**」タブをクリックします。
6. 文字データ型 (CHAR、VARCHAR など) のソース列を選択します。これにより、「**Encoding Conversion**」領域が有効になります。
7. 「**Source**」リストから文字エンコードを選択します。
 - Unicode データベース: UTF-8
 - 部分的 Unicode データベース: アプリケーションのエンコード
8. 「**Target**」リストから、変換する文字エンコードを選択します。例えば、バックエンド・データ・サーバーに、UCS-2 ビッグ・エンディアン・フォームの文字データ型が保管されている場合があります。
9. 「**Apply**」をクリックします。
10. solidDB がソース・データ・ストアまたはターゲット・データ・ストアであるすべてのサブスクリプションに対して、上記の手順を繰り返します。

タスクの結果

サブスクリプションでレプリケーションを開始すると、InfoSphere CDC はソース列の文字エンコードを指定したエンコードに変換し、マップされたターゲット列に新規エンコードでデータを入力します。

高速リフレッシュの使用可能化

高速リフレッシュ機能により、バックエンド・データ・サーバーから solidDB フロントエンドへの大容量のデータの複製にかかる時間が削減されます。高速リフレッシュを使用可能にするには、InfoSphere CDC for solidDB のシステム・パラメーター **solid_fast_refresh_on** を「true」に設定します。パフォーマンスをさらに向上させるには、**solid_fast_refresh_apply_pipes** システム・パラメーターを、システム内のプロセッサ (コア) の数に設定します。

始める前に

高速リフレッシュ機能は、solidDB がターゲット・データ・ストアであるサブスクリプションにおいてのみ使用できます。

大部分のパフォーマンス向上は、簡単なセットアップで実現されます。例えば、表あたり行あたりのデータ量、コード・ページ変換、および列マッピングなどの要因が、高速リフレッシュ機能のパフォーマンスに影響を及ぼします。

このタスクについて

以下を使用して、システム・パラメーターを設定できます。

- `dmset -I <INSTANCE_NAME> <parameter_name>=<parameter_value>` コマンド

- または、Management Console を使用して以下を行います。
 1. Management Console の「**Configuration**」パースペクティブで、データ・ストアを選択します。
 2. データ・ストアを右クリックし、「**Properties**」 → 「**System Parameters**」を選択します。

アクティブ・レプリケーション時にシステム・パラメーターを変更した場合は、変更を有効にするために、レプリケーションを停止して再始動する必要があります。

手順

1. InfoSphere CDC for solidDB システム・パラメーター **solid_fast_refresh_on** を「true」に設定します (デフォルトは「false」です)。

例えば、以下のように指定します。

```
dmset -I solidDB_1 solid_fast_refresh_on=true
```

2. InfoSphere CDC for solidDB システム・パラメーター **solid_fast_refresh_apply_pipes** を、システム内のプロセッサ (コア) の数 (デフォルトは「2」) に設定します。

例えば、以下のようにします。

```
dmset -I solidDB_1 solid_fast_refresh_apply_pipes=4
```

InfoSphere CDC レプリケーションでの共有メモリー・アクセス (SMA) の使用

InfoSphere CDC レプリケーションで SMA を使用するには、SMA サーバーを始動し、solidDB インスタンス用の InfoSphere CDC レプリケーションと SMA サーバー間のローカル SMA 接続を有効にする必要があります。

始める前に

SMA 接続の場合、SMA を使用する solidDB サーバーと InfoSphere CDC for solidDB は同じノードに配置する必要があります。

手順

1. SMA ドライバー・ライブラリーの場所が、LD_LIBRARY_PATH or LIBPATH (Linux® および UNIX の場合) または PATH (Windows の場合) 環境変数に含まれていることを確認します。

詳しくは、「*IBM solidDB 共有メモリー・アクセスおよびリンク・ライブラリー・アクセス・ユーザー・ガイド*」の『Java を使用する場合の SMA 用の環境の構成』を参照してください。

2. ファイル・タイプ拡張子を使用せずに、SMA ドライバー・ライブラリーのシンボリック・リンク (ssolidisma65) を <solidDBインストール・ディレクトリー>/bin ディレクトリーに作成します。

例えば、Linux オペレーティング・システムの場合、以下のコマンドを使用します。

```
ln -s ssolidisma65.so ssolidisma65
```

3. コマンド・プロンプトにコマンド `solidsma` を入力して、SMA を始動します。
4. InfoSphere CDC for solidDB インスタンスを構成して、solidDB サーバーに接続するときに SMA 接続が使用されるようにします。

InfoSphere CDC for solidDB 構成ツール (`dmconfigurets`) を使用して、SMA 接続を有効にします。

表 4. `dmconfigurets` を使用した SMA 接続の使用可能化

オペレーティング・システム	<code>dmconfigurets</code> を使用して SMA 接続を使用可能にする方法
Linux および UNIX	<ol style="list-style-type: none"> 1. 「Single server」構成タイプを選択します。 2. 「Enable SMA」オプションで、<code>y</code> を入力し、Enter キーを押します。
Windows	「 New instance 」または「 Edit instance 」ダイアログの「 Server 」領域で、「 Enable SMA 」チェック・ボックスを選択します。

「**Enable SMA**」を選択すると、solidDB 固有の接続プロパティ `solid_shared_memory=yes` が接続ストリングに追加されます。

5 データ・エージング

データ・エージング機能を使用すると、アプリケーションによって、InfoSphere CDC レプリケーション構成内のある solidDB データベースからは不要なデータを削除する一方、別のデータベースではそのデータを保持することができます。

データ・エージング機能は、「*IBM solidDB Universal Cache ユーザー・ガイド*」の『データ・エージング』で説明されています。この説明は、solidDB と solidDB の間の InfoSphere CDC レプリケーション構成にも当てはまりますが、フロントエンドとバックエンドの両方のデータが solidDB データベースである場合を除きます。

6 SQL パススルー

SQL パススルーを使用すると、アプリケーションから、InfoSphere CDC レプリケーション構成にあるすべての solidDB データベースのデータに、単一接続でアクセスできます。例えば、SQL パススルーを有効にして、ある solidDB サーバーで実行できない SQL ステートメントを別のサーバーに渡すことができます。

SQL パススルー機能は、「*IBM solidDB Universal Cache ユーザー・ガイド*」のセクション『SQL パススルー』で説明されています。この説明は、solidDB と solidDB の間の InfoSphere CDC レプリケーション構成にも当てはまりますが、フロントエンドとバックエンドの両方のデータが solidDB データベースである場合を除きます。

7 ツールおよびユーティリティー

solidDB サーバーおよび InfoSphere CDC for solidDB インストール・パッケージには、InfoSphere CDC レプリケーション・テクノロジーのセットアップおよび使用において、共通タスクの自動化およびスクリプト記述を行うためのツールおよびユーティリティーが含まれています。例えば、これらのツールおよびユーティリティーを使用して、InfoSphere CDC インスタンスおよびサブスクリプションのスクリプトを作成したり、データ・エージングおよびリフレッシュを自動化したりすることができます。

ツールはサンプルのアプリケーション、スクリプト、およびストアード・プロシージャのセットとして提供され、オペレーティング・システムのコマンド・プロンプトから使用できます。

InfoSphere CDC for solidDB パッケージ - ツールおよびユーティリティー

表 5. InfoSphere CDC for solidDB パッケージ - ツールおよびユーティリティー

ツールまたはユーティリティー	用途	場所
インスタンスおよびサブスクリプション管理ツール <ul style="list-style-type: none"> • dminstancemanager • dmsubscriptionmanager 	InfoSphere CDC インスタンスおよびサブスクリプションの管理を容易にするコマンド行ユーティリティー	<InfoSphere CDCfor solidDB インストール・ディレクトリー>%samples%ucutils
ucautomation 自動化フレームワーク (Perl)	データ・ストア、サブスクリプション、およびマッピングの作成など、インストールおよび構成タスクを自動化するための Perl ベースのサンプル・スクリプトおよびライブラリー・モジュール	<InfoSphere CDCfor solidDB インストール・ディレクトリー>%samples%ucautomation
ucdeploy サンプル・アプリケーション	スクリプトを使用して InfoSphere CDC レプリケーションをセットアップする方法を示すサンプル・アプリケーション。	<InfoSphere CDCfor solidDB インストール・ディレクトリー>%samples%ucdeploy
uchsbmonitor サンプル・スクリプト	高可用性 (HotStandby) 構成など、サブスクリプションのセットをモニターおよび再開するためのサンプル・スクリプト	<InfoSphere CDCfor solidDB インストール・ディレクトリー>%samples%uchsbmonitor
ucpassthrough サンプル・アプリケーション	SQL パススルー機能を使用して InfoSphere CDC レプリケーションをセットアップする方法を示すサンプル・アプリケーション	<InfoSphere CDCfor solidDB インストール・ディレクトリー>%samples%ucpassthrough

solidDB パッケージ - ストアード・プロシージャ

表 6. solidDB パッケージ - ストアード・プロシージャ

ストアード・プロシージャ	用途	場所
データ・エージング • create_automatic_aging.sql • start_automatic_aging.sql • stop_automatic_aging.sql	ユーザー定義のエージング・ルールに基づいて、solidDB データベース内のデータのエージングを自動化する	<solidDBインストール・ディレクトリー>%procedures
リフレッシュ • create_refresh_package.sql	InfoSphere CDC インスタンスとの対話なしに、アプリケーションからプログラムでリフレッシュを開始できるようにする	<solidDBインストール・ディレクトリー>%procedures

Perl 自動化フレームワーク

Perl 自動化フレームワークによって、Linux および Windows 環境でインストール、構成、およびサブスクリプション処理タスクの自動化を行うための、Perl ベースのさまざまなサンプル・スクリプトおよびライブラリー・モジュールを使用することができます。

Perl 自動化フレームワークは、<InfoSphere CDC for solidDB installation directory>%samples%ucautomation ディレクトリーにあります。

- **include** – Perl モジュール
- **perldoc** – 各ライブラリー・モジュールにある Perl 文書 (POD)
- **samples** – 2 つの solidDB インスタンス間、および solidDB と、DB2[®] for Linux、UNIX、および Windows インスタンス間のサブスクリプションを作成および制御するためのサンプル・スクリプト

このフレームワークを使用して、以下の自動化を行うことができます。

- solidDB Universal Cache のコンポーネントのインストール、または InfoSphere CDC テクノロジーを使用した solidDB と solidDB の間のレプリケーション
- さまざまなタイプ (solidDB、Informix[®]、DB2 for Linux、UNIX、および Windows) のデータベースの作成、およびそれらに対する SQL ステートメントの実行
- サポートされるすべてのデータベース・タイプに応じた、InfoSphere CDC インスタンスの作成/開始
- データ・ストア、サブスクリプション、およびマッピングの作成と、ミラーリングの開始
- 作成済みの任意のコンポーネントの削除による環境のクリーンアップ

自動化フレームワークを使用するには、以下が必要です。

- Linux または Windows 環境
- Perl の稼働中のインストール済み環境 (<http://www.perl.com> で入手可能)

環境のセットアップ方法、および自動化フレームワークの使用方法について詳しくは、<InfoSphere CDC for solidDB インストール・ディレクトリー>%samples%ucautomation ディレクトリーにある readme ファイルを参照してください。

インスタンスおよびサブスクリプション管理ツール

dminstancemanager および **dmsubscriptionmanager** ツールを使用して、InfoSphere CDC インスタンスおよびサブスクリプションの作成、除去、および変更のスクリプトを記述することができます。

これらのツールは、<InfoSphere CDCfor solidDB installation directory>%samples%ucutils ディレクトリーにあります。

- **dminstancemanager** – InfoSphere CDC インスタンスの作成、除去、変更、および状況の照会を行います
- **dmsubscriptionmanager** – サブスクリプションの作成および除去、または既存のサブスクリプションへの表マッピングの追加を行います
- **ucenv** – ユーティリティーを使用するための環境を構成します

dminstancemanager および **dmsubscriptionmanager** ツールは、InfoSphere CDC for solidDB とだけでなく、どの InfoSphere CDC エンジンとも使用することができます。**ucenv** スクリプトは、**dminstancemanager** が使用する InfoSphere CDC エンジンの定義に使用されます。

dminstancemanager および **dmsubscriptionmanager** ユーティリティーを使用するための環境のセットアップ方法、およびそれらの使用方法については、<InfoSphere CDC for solidDBインストール・ディレクトリー>%samples%ucutils ディレクトリーにある readme.txt を参照してください。

ucdeploy および **ucpassthrough** のサンプル・アプリケーションも参照してください。これらは、<InfoSphere CDCfor solidDB installation directory>%samples ディレクトリーにあります。これらのサンプルは、**dminstancemanager** および **dmsubscriptionmanager** ツールを使用して、InfoSphere CDC インスタンスとレプリケーション・サブスクリプションを作成します。

ucdeploy – 構成およびセットアップ・サンプル

ucdeploy サンプルは、2 つの solidDB データベース (フロントエンドおよびバックエンド)、対応する InfoSphere CDC インスタンスとデータ・ストア、およびそれらの間のサブスクリプションを作成します。次に、サンプルはサブスクリプションでのミラーリングを開始し、フロントエンド・データベースからバックエンド・データベースにデータが複製される方法が示されます。

ucdeploy サンプルは **dminstancemanager** ユーティリティーを使用して、フロントエンドおよびバックエンドのインスタンスと、**dmsubscriptionmanager** ユーティリティーを作成し、サブスクリプションを作成します。このサンプルは、標準 InfoSphere CDC dm コマンドも使用して、サブスクリプションでのミラーリングの開始などを行います。

ucdeploy サンプルは、<InfoSphere CDC for solidDB installation directory>%samples%ucdeploy ディレクトリーにあります。

サンプルの使用方法について詳しくは、上記のディレクトリーにある `readme.txt` を参照してください。

ucpassthrough – SQL パススルー・セットアップ・サンプル

ucpassthrough サンプルは、2 つの solidDB データベース (フロントエンドおよびバックエンド)、対応する InfoSphere CDC インスタンスとデータ・ストア、およびそれらの間のサブスクリプションを作成します。次に、このサンプルは SQL パススルー機能を使用して、バックエンド・データベースにデータを挿入します。

ucpassthrough サンプルは **dminstancemanager** ユーティリティーを使用して、フロントエンドおよびバックエンドのインスタンスと、**dmsubscriptionmanager** ユーティリティーを作成し、サブスクリプションを作成します。このサンプルは、標準 InfoSphere CDC `dm` コマンドも使用して、インスタンスの開始などを行います。solidDB SQL ステートメントは、バックエンド・データベースにデータを挿入し、それを読み取るパススルー・ステートメントに使用されます。

ucpassthrough サンプル・スクリプトは、<InfoSphere CDC for solidDB installation directory>%samples%ucpassthrough ディレクトリーにあります。サンプルの使用方法について詳しくは、上記のディレクトリーにある `readme.txt` を参照してください。

uchsbmonitor – HSB サブスクリプション・モニター・サンプル

uchsbmonitor サンプルは Perl スクリプトであり、高可用性セットアップでサブスクリプションをモニターし、フェイルオーバーまたは切り替えイベントによってサブスクリプションが停止した場合に、サブスクリプション上でミラーリングを再開します。

例えば、ターゲット・データ・ストアである solidDB 1 次サーバーに障害が起こると、サブスクリプション上でのレプリケーションは終了します。リカバリーするには、サブスクリプション上のレプリケーションを再開する必要があります。

uchsbmonitor サンプル・スクリプト `hsbmonitor.pl` は、<InfoSphere CDC for solidDB installation directory>%samples%uchsbmonitor ディレクトリーにあります。

このスクリプトを実行する構文は、以下のとおりです。

```
perl hsbmonitor.pl -s src -t tgt <subscription_name>
```

ここで、

- `src` – ソース・インスタンスの名前
- `tgt` – ターゲット・インスタンスの名前
- `<subscription_name>` – モニター対象のサブスクリプションの名前

データ・エージングとリフレッシュのための SQL ストアード・プロシージャ

solidDB パッケージに含まれているストアード・プロシージャによって、データ・エージングとリフレッシュを自動的に行うことができます。

エージング・プロシージャは、ユーザー定義のエージング・ルールに基づいて、solidDB データベースの行を削除します。エージング・プロシージャを solidDB 始動時にアクティブ化して、バックグラウンドで自動データ・エージングを実行することができます。

リフレッシュ・ストアード・プロシージャを使用すると、InfoSphere CDC インスタンスと対話しなくても、アプリケーションからプログラムでリフレッシュを開始できます。

エージング・ストアード・プロシージャの使用

エージング・プロシージャ `SQL_START_AUTOMATIC_AGING` は、solidDB SQL ストアード・プロシージャであり、solidDB データベースでユーザー定義の `DELETE` ステートメントを実行します。ユーザーは、表 `AUX_AUTOMATIC_DELETES` に維持される `DELETE` ステートメントの形式でエージング・ルールを定義します。`AUX_AUTOMATIC_DELETES` 表は、このプロシージャによって自動的に作成されます。

エージング・ルール

通常の SQL ステートメントを使用して、`AUX_AUTOMATIC_DELETES` のエージング・ルールを作成および変更します。ルールは、実行時に削除、追加、または変更することができます。

表 7. `AUX_AUTOMATIC_DELETES` 表の定義

列	データ型	説明
id	INTEGER PRIMARY KEY	エージング・ルール用の ID
statement	LONG VARCHAR NOT NULL	この値には、完全な <code>DELETE</code> ステートメントを指定する必要があります。それ以外のステートメントを指定すると、プロシージャはすべて失敗します。 1 行につき 1 つのステートメントのみを指定できます。
exec_period	INTEGER NOT NULL	エージング間隔を秒単位で定義します。

表 7. AUX_AUTOMATIC_DELETES 表の定義 (続き)

列	データ型	説明
next_exec_date	TIMESTAMP	次にルールを実行する時間を定義します。 プロシージャは、exec_period の値を現在の実行時間に加算して、値を計算します。 ユーザーがルールの作成時にこの値を指定する場合、最初の削除操作は指定した時間に行われます。 この値を指定しない場合は、このステートメントが次の使用可能な機会に行われます。

いずれのタイプの DELETE ステートメントも、エージング・ルールとして使用することができます。AUX_AUTOMATIC_DELETES 表の各行は、単一のルールに対応します。表に複数のルール挿入して、それぞれのルールを独自の頻度で実行することもできます。

ルールの表記は、アプリケーション設計に応じて異なります。以下に 2 つの例を示します。

- **例 1: エージング・ルールは、エージング状態に関する情報を収める列に基づきます。**

「table_1」という名前の表で、エージングを行う行を列「state」の値「DONE」によって識別できる場合、ルール・ステートメントは、以下のようになります。

```
DELETE FROM table_1 WHERE state='DONE';
```

- **例 2: エージング・ルールは日付に基づきます。**

「table_2」という名前の表で、日付が現在の日付よりも古いすべての行に対してエージングを行うことができる場合、ルール・ステートメントは、以下のようになります。

```
DELETE FROM table_2 WHERE DATE<CURDATE();
```

プロシージャのライフ・サイクル

このプロシージャには、パラメーターがありません。これは内部ループで実行され、各反復においてルールを読み取り、適用可能なルールを実行してから、exec_period の値 (秒単位) を現在の実行時間に加算することで、次のルールの実行時間を計算して更新します。デフォルトでは、プロシージャは各反復間に 1 秒スリープします。スリープ間隔は、プロシージャのコードを編集することで、変更することができます。

このプロシージャは、通常、バックグラウンド・ジョブとして実行します。終了メカニズムは、プロシージャが作成する AUX_AUTOMATIC_DELETES_BREAK という表に基づきます。内部ループの各反復で、プロシージャは AUX_AUTOMATIC_DELETES_BREAK 表に行が存在するかどうかを検査します。表に少なくとも 1 つの行が存在する場合、プロシージャは終了します。次の開始時に、プロシージャは AUX_AUTOMATIC_DELETES_BREAK 表からすべての行を削除します。

表 8. AUX_AUTOMATIC_DELETES_BREAK 表の定義

列	データ型	説明
break	INTEGER	行が存在する場合、エージング・プロシージャを終了します。

エージング・プロシージャを作成および実行するスクリプト

solidDB パッケージには、ストアード・プロシージャを作成および実行するための SQL スクリプトが含まれています。このスクリプトは、solidDB のインストール・ディレクトリ下の procedures ディレクトリに格納されています。

表 9. エージング・プロシージャを作成および実行するスクリプト

スクリプト	使用法
create_automatic_aging.sql	ストアード・プロシージャを作成します。
start_automatic_aging.sql	ストアード・プロシージャを呼び出します。
stop_automatic_aging.sql	ストアード・プロシージャを停止します。

エージング・プロシージャの作成

エージング・プロシージャを作成するには、以下の手順を実行します。

1. フロントエンドでエージングを行う表に関連する、バックエンドからフロントエンドへのサブスクリプションが存在する場合、そのサブスクリプションを削除または停止します。

あるいは、データベースを設計するときに、InfoSphere CDC 行フィルターを使用して、エージングを行ったデータの再帰を回避できるようにします。例については、『37 ページの『例: 双方向サブスクリプションのためのデータ・エージングの自動化』』を参照してください。

2. スクリプト create_automatic_aging.sql を実行して、プロシージャを作成します。

例えば、solsql を使用して、以下のスクリプトを実行できます。

```
solsql -f "C:\solidDB\procedures\create_automatic_aging.sql" "tcp 2315" dba dba
```

エージング・プロシージャの開始および実行

プロシージャを作成したら、プロシージャを開始してエージング・ルールを定義する必要があります。エージング・ルールは、実行時に変更することもできます。

1. エージング・プロシージャを開始します。
 - スクリプト start_automatic_aging.sql を実行します。

これにより、エージング・プロシージャがバックグラウンドで開始されます。

または

- -x executeandnoexit コマンド行オプションを使用して、solidDB の始動時に start_automatic_aging.sql スクリプトを組み込みます。

```
solid -x executeandnoexit:start_automatic_aging.sql
```

2. AUX_AUTOMATIC_DELETES 表にデータを設定することで、エージング・ルールを定義します。

例えば、5 秒ごとに「state」列の値に基づいて表「table_1」のデータのエージングを行うには、以下のコマンドを発行します。

```
INSERT INTO aux_automatic_deletes (id, statement, exec_period) values
(1, 'DELETE FROM table_1 WHERE state='DONE'', 5);
COMMIT WORK;
```

エージング・プロシージャの停止

エージング・プロシージャは、以下の方法で停止できます。

- スクリプト stop_automatic_aging.sql を実行します。
- 以下のコマンドを発行することで、AUX_AUTOMATIC_DELETES_BREAK 表に行を追加します。

```
INSERT INTO aux_automatic_deletes_break (1);
COMMIT WORK;
```

- ADMIN COMMAND 'backgroundjob' コマンドを使用して、プロシージャを制御します。

リフレッシュ・ストアード・プロシージャの使用

リフレッシュ・プロシージャ *TS_REFRESH_CDC_SUBSCRIPTION* は、solidDB SQL ストアード・プロシージャであり、サブスクリプション名に基づいて、サブスクリプションのリフレッシュを開始します。

solidDB 接続からリフレッシュを開始できるようにするには、リフレッシュ・プロシージャを開始する前に、InfoSphere CDC for solidDB のコマンド *dmsetaccessserverparams* を使用して、Access Server のログイン・データを設定する必要があります。

このプロシージャが呼び出されると、サブスクリプションの存在とリフレッシュ状況を検査します。

- リフレッシュを開始できる場合、プロシージャ呼び出しは、リフレッシュが完了するまでブロックされます。リフレッシュするデータのサイズによっては、呼び出しが長時間ブロックされる場合があります。

呼び出しが戻らない場合は、通常のタイムアウトが適用されます。

- リフレッシュを開始できない場合、エラーが返されます。

リフレッシュの状態は、*TS_REFRESH* という表に維持されます。この表は、インスタンスが作成されるときに、その InfoSphere CDC for solidDB によって自動的に作成されます。プロシージャを開始すると、状況は「1」（リフレッシュの進行中）に変わります。リフレッシュが完了すると、InfoSphere CDC for solidDB は状態を「2」（リフレッシュが完了しました）に更新します。リフレッシュが失敗した場合、InfoSphere CDC for solidDB は、表のエラーを報告します。

表 10. TS_REFRESH 表の定義

列	データ型	説明
subscription_name	VARCHAR (20) PRIMARY KEY	サブスクリプション名
state	INTEGER NOT NULL	リフレッシュの状態 <ul style="list-style-type: none"> • -1 — エラー • 0 — リフレッシュが要求されました • 1 — リフレッシュの進行中 • 2 — リフレッシュが完了しました
error_description	VARCHAR(255)	エラーの説明 <ul style="list-style-type: none"> • Access Server のパラメーターのロードに関する問題 • Access Server のユーザー名が設定されていません • Access Server のパスワードが設定されていません • Access Server のホスト・アドレスが設定されていません • Access Server のポート番号が設定されていません • Access Server への接続の確立に関するエラー • Access Server への接続エラー • Access Server への接続が存在しません • パブリッシャーの取得に失敗しました • 一致するサブスクリプションの検出に失敗しました • サブスクリプションが存在しません • リフレッシュのポーリング・エラー
inserts_performed	BIG INT	リフレッシュ時にコミットされた挿入の行数 コミットごとの挿入の数は、InfoSphere CDC システム・パラメーター 112 ページの『refresh_commit_after_max_operations』で設定された値によって異なります。 デフォルト値は 0 です。

リフレッシュ・プロシージャの作成

solidDB パッケージには、ストアード・プロシージャを作成するための SQL スクリプトが含まれています。このスクリプトは、solidDB のインストール・ディレクトリ下の procedures ディレクトリに格納されています。

スクリプト	使用法
create_refresh_package.sql	ストアード・プロシージャを作成します。

リフレッシュ・プロシージャを作成するには、以下の手順を実行します。

1. サブスクリプションと、フロントエンドおよびバックエンドのデータ・サーバーが作成されており、InfoSphere CDC コンポーネントが正常に稼働していることを確認します。

2. スクリプト `create_refresh_package.sql` を実行して、リフレッシュ・プロシージャを作成します。

以下の例で示すように `solsql` を使用して、スクリプトを実行できます。

```
solsql -f "C:\%solidDB%\procedures\create_refresh_package.sql" "tcp 2315" dba dba
```

3. InfoSphere CDC for solidDB のコマンド `dmsetaccessserverpararms` を使用して、Access Server 用のログイン・データを定義します。

`dmsetaccessserverpararms` コマンドの構文は、以下のとおりです。

```
dmsetaccessserverpararms [-u <username>] [-p <password>] [-H <hostname>] [-P <port>]
```

例えば、以下のように指定します。

```
dmsetaccessserverpararms -u dba -p dba - H 192.167.3.3 -P 10101
```

リフレッシュ・プロシージャの実行

リフレッシュ・プロシージャを実行するには、以下の手順を実行します。

1. フロントエンドからバックエンドへのサブスクリプションで、進行中のミラーリングが存在することを確認します。
2. 以下の構文を使用して、リフレッシュ・プロシージャを呼び出します。

```
CALL ts_refresh_cdc_subscription ('subscription_name');
```

例えば、以下のように指定します。

```
CALL ts_refresh_cdc_subscription ('current_invoices');
```

リフレッシュの状況のモニター

リフレッシュの進行状況は、リフレッシュの状態およびリフレッシュされた行の数 (`inserts_performed`) について `TS_REFRESH` 表を表示することによって確認できます。

例えば、以下のように指定します。

```
SELECT * from TS_REFRESH;
```

SUBSCRIPTION_NAME	STATE	ERROR_DESCRIPTION	INSERTS_PERFORMED
current_invoices	1		2000

1 rows fetched.

ストアド・プロシージャの停止

プロシージャ呼び出しは、リフレッシュが正常に実行されるまでブロックされます。プロシージャを停止する場合は、ADMIN COMMAND 'throwout' を使用して強制終了します。

通常のタイムアウトとして、以下が適用されます。

- 照会タイムアウトが設定されている場合、呼び出しは照会タイムアウトでタイムアウトになります。デフォルトでは、タイムアウトはありません。

例えば、以下のように指定します。

- ODBC では、ODBC ステートメントの属性 `SQL_ATTR_QUERY_TIMEOUT` を使用して照会タイムアウトを設定します (秒単位)。
- JDBC では、ステートメントのメソッド `setQueryTimeout()` を使用して照会タイムアウトを設定します (秒単位)。
- 接続タイムアウトが設定されている場合は、タイムアウトの満了時に接続が失われます。

タイムアウト動作の詳細については、「*IBM solidDB プログラマー・ガイド*」の付録『タイムアウト制御』を参照してください。

例: 双方向サブスクリプションのためのデータ・エージングの自動化

この例では、エージング・プロシージャーを InfoSphere CDC 行フィルターとともに使用して、双方向サブスクリプション・セットアップでデータ・エージングを自動化する方法について説明します。

セットアップに双方向サブスクリプションが含まれる場合は、アプリケーションとサブスクリプションを設計する際に、フロントエンドから削除される (エージングが行われる) 行が、バックエンドからフロントエンドへのリフレッシュまたはミラーリングの使用時にフロントエンドに返されないようにする必要があります。

可能な方法の 1 つとして、エージング・プロシージャーを使用して、フロントエンドのデータを削除すると同時に、InfoSphere CDC 行フィルターをセットアップして、エージングが行われた行のフロントエンドへのレプリケーションを回避します。

この例では、アプリケーションは、エージング対象にできるデータを制御するために、そのデータのエージング状況に関する情報を維持し、削除される行にはフラグを立てます。データの実際の削除処理は、エージング・プロシージャーを使用して実行されます。次に、削除フラグが立てられた行がバックエンドからフロントエンドに複製されないように、InfoSphere CDC 行フィルターがセットアップされます。

双方向サブスクリプションによるデータ・エージングの設定例

注: この例では、表に新しい列を追加できることを前提とします。これは必須ではありません。データベース設計によっては、既存の列を使用して、エージングが行われた行を特定することができます。

1. **データ・エージングをサポートするために、環境をセットアップします。**
 - 値「0」(エージングが行われていない) または「1」(エージングが行われた) を収める列「aged」を追加します。
 - アプリケーションの設計では、エージングが行われる行の「aged」列に値「1」が設定されるようにします。
2. **solidDB でエージング・プロシージャーを作成して開始します。**

詳しくは、『31 ページの『エージング・ストアード・プロシージャーの使用』』を参照してください。

3. **フロントエンドからバックエンド、およびその逆方向にサブスクリプションをセットアップします。**

4. バックエンドからフロントエンドへのサブスクリプションでは、行フィルターをセットアップします。

「aged」列で値が 1 より小さい行のみを複製するという行フィルター・ルールを作成します。

フィルターの設定方法については、「*InfoSphere Change Data Capture Management Console 管理ガイド*」のセクション『行と列のフィルタリング』を参照してください。

5. **AUX_AUTOMATIC_DELETES** に **DELETE** ステートメントを追加して、エージング・ルールを作成します。

例えば、table_1 で削除フラグが立てられたすべての行を削除するというルールを作成する場合は、以下の INSERT ステートメントを実行します。

```
INSERT INTO aux_automatic_deletes (id, statement, exec_period) values
(1, 'DELETE FROM table_1 WHERE aged=1', 10);
```

結果

アプリケーションが実行中の場合、データベースの所定の行にエージング・フラグ（「aged」=1）が立てられます。それらの行は、変更されたエージング状態とともにバックエンドに複製されます。エージング・プロシージャが実行中の場合、フラグが立てられた行がフロントエンド表から削除されます。これらの行は、行フィルター（「aged」< 1）によって回避されるため、バックエンドからフロントエンドに複製されません。

バックエンドからフロントエンドへのデータの複製

バックエンドからフロントエンドへのすべてのレプリケーション形式が許可されます。つまり、連続ミラーリングも、InfoSphere CDC ツールから開始するリフレッシュも、また「リフレッシュ・プロシージャ」を使用してアプリケーションから開始するリフレッシュも実行できます。

ただし、InfoSphere CDC レプリケーションは実際には非同期であるため（フロントエンドでの変更が即座にバックエンドで有効になりません。また、その逆も同様です）、この例では、以下の制限事項が適用されます。

- エージングを行う行がバックエンドで変更中の場合、バックエンドからフロントエンドへのミラーリングは、エージング・アクティビティーが完了するまで許可されません。
- リフレッシュ・プロシージャを使用する場合は、リフレッシュを実行する前に「aging」列の値がバックエンドに複製されていることを確認します。レプリケーションが完了する前にリフレッシュを実行すると、フロントエンドでエージングを行ったデータがバックエンドに返される場合があります。

リフレッシュの進行状況は、TS_REFRESH 表内のリフレッシュされた行の数を表示することによって確認できます。

例えば、以下のように指定します。

```
SELECT * from TS_REFRESH;

SUBSCRIPTION_NAME  STATE  ERROR_DESCRIPTION  INSERTS_PERFORMED
```

```
-----  
current_invoices      1      2000  
1 rows fetched.
```

8 InfoSphere CDC レプリケーションの障害シナリオ

以下のセクションでは、さまざまな障害シナリオの概要を示し、必要なリカバリー手順があればそれについても説明します。

ヒント: リカバリー手順に手動操作が含まれている場合、その操作は、スクリプトを使用するか、または InfoSphere CDC テクノロジーで使用可能なコマンドを使用して自動化できる場合がよくあります。

スタンドアロン solidDB サーバーの障害

スタンドアロン solidDB サーバーに障害が発生すると、サブスクリプションのレプリケーションも終了します。リカバリーするには、以下の手順を実行します。

手順

1. solidDB サーバーを手動で再始動し、データベースをリカバリーします。

詳しくは、「*IBM solidDB 管理者ガイド*」の『*solidDB の管理*』セクションを参照してください。

2. InfoSphere CDC インスタンスを再開します。

詳しくは、62 ページの『InfoSphere CDC の開始と停止』のセクションを参照してください。

3. サブスクリプションのレプリケーションを再開します。

詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『サブスクリプションでのレプリケーションの開始と終了』のセクションを参照してください。

タスクの結果

再開すると、サブスクリプションのレプリケーションが再開し、データベースが再同期されます。レプリケーションが正常に続行します。

InfoSphere CDC インスタンスの障害

InfoSphere CDC インスタンスに障害が発生すると、サブスクリプションのレプリケーションも終了します。リカバリーするには、以下の手順を実行します。

1. InfoSphere CDC インスタンスを再開します。

詳しくは、62 ページの『InfoSphere CDC の開始と停止』のセクションを参照してください。

2. サブスクリプションのレプリケーションを再開します。

詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『サブスクリプションでのレプリケーションの開始と終了』のセクションを参照してください。

結果

再開すると、サブスクリプションのレプリケーションが再開し、データベースが再同期されます。レプリケーションが正常に続行します。

この障害が発生すると、solidDB サーバーは **LogReader.MaxLogSize** パラメーターで指定された限度に達するまで、トランザクションの処理を続行します。

HA モード (HotStandby) の solidDB サーバーの障害

以下のセクションでは、solidDB HotStandby 構成での障害シナリオを説明します。

1 次 solidDB サーバーの障害

1 次 solidDB サーバーに障害が発生した場合は、高可用性コントローラー (HAC) などの高可用性マネージャーが、標準的なプロシージャとして 2 次 solidDB サーバーへのフェイルオーバーを実行します。2-Safe プロトコルが使用されている場合、データベースとログの状態が完全に保持されます。一般に、アプリケーションが認識するフェイルオーバー時間は、1 秒未満です。

- HA セットアップ内の solidDB サーバーからのみもう 1 つのサーバーにデータが複製されるよう InfoSphere CDC レプリケーション・システムが構成されている場合、InfoSphere CDC インスタンスは新しい 1 次側に自動的に再接続し、レプリケーションが続行されます。
- もう 1 つのサーバーから HA セットアップ内の solidDB サーバーにデータが複製されるよう InfoSphere CDC レプリケーション・システムが構成されている場合、サブスクリプションのレプリケーションは終了します。Management Console または InfoSphere CDC の dmstartmirror コマンドを使用して、サブスクリプションを再開する必要があります。

詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『サブスクリプションでのレプリケーションの開始と終了』のセクションを参照してください。

上記のシナリオ中、InfoSphere CDC インスタンスは常に稼働状態になります。

ヒント: HA (HotStandby) 機能と高可用性コントローラー (HAC) について詳しくは、「*IBM solidDB 高可用性ユーザー・ガイド*」を参照してください。

2 次 solidDB サーバーの障害

2 次フロントエンドの障害の場合、手操作による介入は不要です。

2 次フロントエンドに障害が発生した場合は、2 次フロントエンド・ノードが、インストール固有の通常の方法でリカバリーされます (例えば、自動的にリポートするなど)。HAC が残りのリカバリーを自動的に行います。障害は、アプリケーションや InfoSphere CDC には認識されません。

1 次 solidDB サーバーと InfoSphere CDC for solidDB インスタンス間の通信リンクの障害

1 次 solidDB サーバーと InfoSphere CDC for solidDB インスタンス間の通信リンクに障害が発生すると、サブスクリプションのレプリケーションも終了します。ただし、リンクだけの障害の可能性は低いと考えられます。

リカバリーするには、以下の手順を実行します。

1. InfoSphere CDC インスタンスを再開します。

詳しくは、62 ページの『InfoSphere CDC の開始と停止』のセクションを参照してください。

2. サブスクリプションのレプリケーションを再開します。

詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『サブスクリプションでのレプリケーションの開始と終了』のセクションを参照してください。

結果

再開すると、サブスクリプションのレプリケーションが再開し、データベースが再同期されます。レプリケーションが正常に続行します。

この障害が発生すると、solidDB サーバーは **LogReader.MaxLogSize** パラメーターで指定された限度に達するまで、トランザクションの処理を続行します。

9 トラブルシューティング

このセクションでは、InfoSphere CDC レプリケーションを構成または使用する際の共通の問題を防止またはトラブルシューティングする方法について説明し、ガイドラインを示します。

初期接続が成功しない

InfoSphere CDC レプリケーションのコンポーネントは、9 ページの『2 章 InfoSphere CDC レプリケーション・コンポーネントのインストール』および 11 ページの『3 章 InfoSphere CDC レプリケーションの構成』に説明した順序でインストールおよび構成する必要があります。以下の手順を検討し、インストールおよび構成の手順に必ず従ってください。

インストール順序

- solidDB サーバー
- InfoSphere CDC for solidDB
- Access Server
- Management Console

重要: InfoSphere CDC for solidDB インストールの終わりに、新しい InfoSphere CDC インスタンスを作成するために、構成ツールの開始を促すプロンプトをインストーラーが出します。11 ページの『solidDBfor InfoSphere CDC レプリケーションの構成』の説明に従って対応する solidDB サーバーを構成していない場合には、構成ツールの開始を選択しないでください。

構成順序

- solidDB サーバー
- InfoSphere CDC インスタンス
- Access Server と Management Console

レプリケーションで使用するコンポーネント間の従属関係

データベース間のレプリケーションをセットアップするには、互いに依存する各種エンティティとコンポーネントを定義し、作成する必要があります。これらのエンティティとコンポーネントは、以下の順序で作成し、逆の順序で変更または削除する必要があります。詳しくは、「*InfoSphere Change Data Capture Management Console* 管理ガイド」を参照してください。

1. データベース
2. InfoSphere CDC インスタンス
3. データ・ストア
4. サブスクリプション
5. 表マッピング

レプリケーション・サブスクリプションの変更

レプリケーション・サブスクリプションの変更が必要な場合には、まずサブスクリプションのレプリケーションを終了する必要があります。詳しくは、「*InfoSphere Change Data Capture Management Console 管理ガイド*」の『サブスクリプションでのレプリケーションの終了』のセクションを参照してください。

切り替えの後に **hsb netcopy** を実行するとサブスクリプションが失敗する

solidDB 高可用性 (HotStandby) 構成では、ソース・データ・ストアとして solidDB を使用するサブスクリプションは、**hsb netcopy** の直後に切り替えを行うと、失敗する場合があります。

これは、例えば以下のような場合に起こる可能性があります。

1. 障害または保守のために中断が生じた後、1 次サーバー (ノード 1) および 2 次サーバー (ノード 2) が ADMIN COMMAND 'hsb netcopy' を使用して同期される場合。
2. 1 次サーバー (ノード 1) に対するレプリケーションがいくつかのトランザクションについて続行される場合。
3. 1 次サーバー (ノード 1) に障害が生じ、切り替えによって 2 次サーバー (ノード 2) が新規 1 次サーバーに変更される場合。
4. サブスクリプションが失敗し、新規 1 次サーバー (ノード 2) に対するレプリケーションを再開できない場合。

原因

コマンド ADMIN COMMAND 'hsb netcopy' は、どのログ・ファイルもコピーしません。この結果、InfoSphere CDC レプリケーションは本質的に非同期であるため、InfoSphere CDC for solidDB は、**netcopy** が行われた時点まですべてのトランザクションの処理を行っていない可能性があります。これは、切り替え後に InfoSphere CDC for solidDB が使用しようとするログ位置はおそらく有効でなく、つまり **netcopy** 前のノード 1 上の最後のトランザクションに対するログ・エントリが新規 1 次サーバー (ノード 2) におそらく存在しないことを意味します。

回避策

InfoSphere CDC for solidDB が、切り替えの後、新規 1 次サーバー (ノード 2) の有効なログ・エントリに確実にアクセスできるようにするには、以下のようになります。

- **netcopy** を実行する前に、1 次サーバー (ノード 1) から 2 次サーバー (ノード 2) にログ・ファイルをコピーします。これによって、InfoSphere CDC for solidDB は、**netcopy** が行われる前に実行されたトランザクションのログ位置に確実にアクセスできるようになります。

または

- **netcopy** の直後に切り替えを行わないでください。つまり、バックエンド・データベースにいくつかのトランザクションが複製されるまで待ってから、切り替え

を行ってください。これによって、1 次サーバー (ノード 1) と 2 次サーバー (ノード 2) のログ位置が確実に同期されます。

または

- (例えば、ノード 1 に障害が生じたため) 切り替えが既に行われた場合は、以下の手順を実行します。
 1. 古い 1 次サーバー (ノード 1) をリカバリーします。
 2. 切り替えを行って、以前の 1 次サーバー (ノード 1) を 1 次サーバーに戻します。
 3. サブスクリプションのレプリケーションを再開します。

別の切り替えを (ノード 2 を新規 1 次サーバーにするために) 行うのは、いくつかのトランザクションが複製されるのを待ってからにしてください。これによって、1 次サーバー (ノード 1) と 2 次サーバー (ノード 2) のログ位置が確実に同期されます。

10 InfoSphere CDC for solidDB (エンド・ユーザー向け資料)

このセクションについて

このセクションでは、solidDB Universal Cache と InfoSphere CDC レプリケーションで使用される IBM InfoSphere Change Data Capture for IBM solidDB コンポーネントのインストールおよび構成手順について詳しく説明します。また、このセクションでは、InfoSphere CDC に固有のコマンドおよびその他の参照情報を記載しています。

このセクションは、「*IBM InfoSphere Change Data Capture* のエンド・ユーザー向け資料」という資料に対応しています。この資料は、その他のデータ・サーバーの InfoSphere CDC コンポーネントとともに配信されます。

solidDB Universal Cache または InfoSphere CDC レプリケーションの設定では、システム・レベルのインストールおよび構成の手順に従い、必要に応じてこのセクションを参照してください。

このセクションでは、InfoSphere CDC という用語は、InfoSphere CDC for solidDB を意味しています。

InfoSphere CDC for IBM solidDB について

IBM InfoSphere Change Data Capture (InfoSphere CDC) は、サポート対象のデータベースにデータを複製したり、そのデータベースからデータを複製したりできるレプリケーション・ソリューションです。また、Management Console で構成した表マッピングの詳細に基づいて、サポート対象のデータベースから複製したデータを受け取ることができます。

InfoSphere CDC では、処理オーバーヘッドとネットワーク・トラフィックを減らすのに使用できる複製データベースを維持できます。レプリケーションは、連続的に実施することも、また最終的な変更に応じて定期的にも実施することもできます。ソース・サーバーからデータが転送されると、ターゲット環境で、そのデータの再マップやトランスフォームを行うことができます。

システム要件

ソフトウェア要件

サポートされるプラットフォームおよびオペレーティング・システムについては、<http://www-01.ibm.com/software/data/soliddb/>の solidDB の Web ページを参照してください。

ハードウェア要件

表 11. ハードウェア要件

RAM	ディスク・スペース
500 MB	220 MB インストール・ファイル、データ・キュー、およびログ・ファイル用

ポート要件

InfoSphere CDC では、Management Console と他のサーバーを実行するクライアント・ワークステーションとの通信用のポート・セットを割り振る必要があります。インターネットへのアクセスは必要ありませんが、これらのポートは、ファイアウォール経由でアクセス可能でなければなりません。

表 12. ポート要件

プロトコル	デフォルトのポート	用途
TCP	11101	以下からの接続を受け入れます。 <ul style="list-style-type: none">• Management Console• レプリケーションのソースとして使用する InfoSphere CDC の他のインストール• コマンド行ユーティリティー Management Console のインストール方法の詳細については、「 <i>Management Console のインストール・ガイド</i> 」を参照してください。

必須のデータベース、ユーザー・アカウント、およびスキーマ solidDB データベースの作成

InfoSphere CDC の構成時に、InfoSphere CDC を接続してデータを複製したい solidDB サーバーのホスト名およびポート番号の入力を促すプロンプトが出されます。InfoSphere CDC をインストールする前に、この solidDB データベースが存在することを確認し、それにアクセスできるデータベース・ユーザーの作成とセットアップが完了していることを確認してください。

SYS_ADMIN_ROLE 特権を持つ solidDB アカウントのセットアップ

Sybase ユーザーを作成およびセットアップし、このユーザーに DBA 特権を割り当てます。InfoSphere CDC を solidDB データベースに接続するには、solidDB ユーザー・アカウントを作成し、そのユーザーに SYS_ADMIN_ROLE 特権を割り当てる必要があります。InfoSphere CDC の構成時に、InfoSphere CDC の接続先にしたい solidDB サーバーのホスト名およびポート番号のほか、このデータベースにアクセスできる solidDB ユーザーのユーザー名およびパスワードの入力を促すプロンプトが出されます。

solidDB スキーマの作成

InfoSphere CDC データベース・メタデータ表用に、スキーマを作成するか、または既存のスキーマを選択します。InfoSphere CDC の構成時には、このスキーマを指定する必要があります。

InfoSphere CDC 用の Windows ユーザー・アカウントのセットアップ

InfoSphere CDC を Windows システムにインストールする場合は、InfoSphere CDC のインストール、構成、またはアップグレードに使用する Windows アカウントを新規にセットアップするか、既存のアカウントから選択する必要があります。

InfoSphere CDC 用の Linux または UNIX ユーザー・アカウントのセットアップ

Linux または UNIX システム上に InfoSphere CDC をインストールする場合は、InfoSphere CDC のインストール、構成、またはアップグレードに使用する Linux または UNIX アカウントを新規にセットアップするか、既存のアカウントから選択する必要があります。InfoSphere CDC は任意のディレクトリーにインストールできますが、そのディレクトリーは、その Linux または UNIX アカウントに所有されている必要があります。

1 バイト文字およびマルチバイト文字のサポート

InfoSphere CDC は、1 バイトとマルチバイトの両方の文字セットのレプリケーションをサポートします。

1 バイト文字のサポート

InfoSphere CDC は、1 バイト文字サポート (SBCS) によるコード・ページ変換を透過的に行います。つまり、各システムで使用されているコード・ページをユーザーが認識する必要はありません。InfoSphere CDC は、ユーザーの構成パラメーターを調べることで、この変換を自動的に実行することができます。

マルチバイト文字のサポート

InfoSphere CDC は、1 バイトで表すことができない日本語や中国語などのマルチバイト文字セット (MBCS) のレプリケーションをサポートします。最も一般的な MBCS 実装は、2 バイト文字セット (DBCS) です。

MBCS の仕様では、特定の変換を構成したときに、データはターゲット・システム上のマップ対象列にそのまま適用されるように指示されます。これは、データベースに (データの実際の文字セットに関係なく) 1 バイト文字セットが構成されている場合に可能ですが、このことは文字セットがマルチバイトである場合には保証できません。

InfoSphere CDC は、マッピングを考慮し、構成セットに従ってデータを適用します。列に文字セットを正しく挿入できるという確証はありません。ユーザーは、データベース上の文字セットを把握し、それらのデータの文字セット変換を選択する際に、適切な値を選択する必要があります。Management Console にエンコード変換を設定すると、InfoSphere CDC はデータを、そのデータが受信されたときと完全

に同じフォームで、ターゲット・データベースに適用します。

solidDB データベース上でのマルチバイト文字サポートに対する影響

solidDB 文字データ型のエンコードは、データベース・モードが *Unicode* であるか部分的 *Unicode* であるかによって決まります。

Unicode モード (`General.InternalCharencoding = utf8`)

- 文字データ型 (CHAR、VARCHAR など) は、UTF-8 で保管されます。
- ワイド文字データ型 (WCHAR、WVARCHAR など) は、UTF-16 で保管されます。

部分的 Unicode モード (`General.InternalCharencoding = raw`)

- 文字データ型は特定のエンコードを使用しません。代わりに、データはバイト・ストリングで保管されます (ユーザーのアプリケーションがこれを認識し、必要に応じて変換を処理することが想定されています)。
- ワイド文字データ型は、UTF-16 で保管されます。

InfoSphere CDC for solidDB の新しいインスタンスが作成されると、デフォルトの solidDB データベース・モードである部分的 *Unicode* に従って、デフォルトのエンコードが設定されます。デフォルトで、文字データ型列のエンコードは、常に *ISOLatin1* に設定されます。

- 使用するデータベース・モードが *Unicode* の場合、文字データ型 (CHAR、VARCHAR など) の列のエンコードを UTF-8 に設定する必要があります。
- データベース・モードが部分的 *Unicode* であり、アプリケーション・エンコードが *ISOLatin1* に設定されていない場合、文字データ型 (CHAR、VARCHAR など) の列のエンコードを、アプリケーション環境で使用されているエンコードに設定する必要があります。

表 13. 文字データ型列およびワイド文字データ型列のデフォルト・エンコード設定 (部分的 *Unicode*) および *Unicode* エンコード設定

列の型	デフォルト・エンコード (部分的 <i>Unicode</i>)	<i>Unicode</i> データベース に必要なエンコード
文字データ型 (CHAR、VARCHAR など)	ISOLatin1	UTF-8
ワイド文字データ型 (WCHAR、WVARCHAR など)	UTF-16BE	UTF-16BE

ユーザー出口およびマルチバイト文字セット

InfoSphere CDC の Java™ クラス・ユーザー出口は、マルチバイト文字セット (MBCS) をサポートします。マルチバイト文字セットは、Java ストリング (UTF-16) に変換されます。

InfoSphere CDC のインストール

このセクションでは、InfoSphere CDC のインストールの段階的な手順を説明します。

InfoSphere CDC の対話式インストール

InfoSphere CDC を、Windows サーバーや、UNIX サーバーまたは Linux サーバーにインストールできます。

InfoSphere CDC をインストールするには (Windows)

手順

1. インストール・ファイルをダブルクリックします。InfoSphere CDC インストール・ウィザードが開きます。
2. 「Next」をクリックします。
3. ライセンス条項に同意する場合には、「**I accept the terms in the license agreement**」を選択し、「Next」をクリックします。
4. InfoSphere CDC をインストールするフォルダーを選択し、「Next」をクリックします。
5. 以前の InfoSphere CDC がインストールされている場合には、インストールのアップグレードを促すプロンプトが出されます。「OK」をクリックしてインストール済み環境をアップグレードします。
6. 製品アイコンの位置を選択し、「Next」をクリックします。
7. インストール・サマリーを検討し、「Install」をクリックします。
8. インストール後に、オプションとして「**Launch Configuration Tool**」を選択し、構成ツールを起動します。構成ツールでは、InfoSphere CDC のインスタンスを追加できます。
9. 「Done」をクリックして、インストールを終了します。

InfoSphere CDC をインストールするには (UNIX および Linux)

このタスクについて

注: X Window システムがインストールされている場合、インストール・プログラムは、グラフィック環境で構成ツールを起動します。インスタンスの開始と停止を行う必要がない点を除いて、構成プロセスは Windows に類似しています。

手順

1. InfoSphere CDC 用にセットアップしたアカウントでログオンします。
2. ご使用の Linux プラットフォーム用の InfoSphere CDC インストール・ファイルをコピーします。
3. インストール・プログラムを実行可能にします。
4. インストール・ファイルの名前を入力して、インストール・プログラムを実行します。
5. 「Introduction」画面で Enter キーを押して、使用許諾契約書を表示します。画面の指示に従って、使用許諾契約書をナビゲートします。
6. 使用許諾契約書を受け入れるには、1 を入力します。

7. インストール・ディレクトリーの絶対パスを入力するか、または Enter キーを押してデフォルトを受け入れます。

注: 指定するディレクトリーは、インストールに使用するアカウントが所有するディレクトリーである必要があります。インストール・プログラムがそのディレクトリーを作成できない場合、別のディレクトリーの指定を促すプロンプトが出されます。

8. インストール・サマリーを検討します。Enter キーを押してインストールを開始します。
9. インストールが完了すると、InfoSphere CDC から InfoSphere CDC 用に構成ツールを起動するオプションが表示されます。
10. 1 を入力して構成ツールを起動します。

InfoSphere CDC のサイレント・インストール

サイレント・インストールでは、各種のパラメーターを含むコマンドを指定することにより、InfoSphere CDC を自動的にインストールできます。このタイプのインストール方式は、スクリプトにサイレント・インストール・コマンドを組み込むことにより、InfoSphere CDC の大規模なデプロイメントで使用できます。

InfoSphere CDC のサイレント・インストールを行うには (UNIX および Linux)

手順

1. InfoSphere CDC 用にセットアップしたアカウントでログオンします。
2. InfoSphere CDC インストール・ファイルをコピーします。
3. インストール・プログラムを実行可能にします。
4. 以下のコマンドを実行して、InfoSphere CDC をインストールし、応答ファイルを生成します。

```
<setup.bin> -r <response-file>
```

5. 別のシステムで、以下のコマンドを実行してサイレント・インストールを行います。

```
<setup.bin> -isilent -f <response-file>
```

ここで、

- <response-file> は、インストール・ファイルの絶対パスです。

InfoSphere CDC の構成 (Windows)

InfoSphere CDC のインストール後に、インストール・プログラムが、構成ツールを起動します。構成ツールでは、ご使用の環境用に InfoSphere CDC を構成できます。レプリケーションを開始するには、InfoSphere CDC を構成する必要があります。

InfoSphere CDC インスタンスの構成 (Windows)

InfoSphere CDC のインスタンスを追加、編集、および削除できます。InfoSphere CDC 構成ツールを使用して、インスタンスに対する作業を行います。

InfoSphere CDC の新しいインスタンスを追加するには (Windows) 始める前に

インストール後に InfoSphere CDC の最初のインスタンスを構成する場合は、この手順のステップ 3 に進みます。

手順

1. コマンド・プロンプトで、以下のコマンドを指定されたディレクトリーで実行し、構成ツールを起動します。

```
¥<InfoSphere CDC Installation Directory>¥bin¥dmconfigurets
```
2. ウェルカム・メッセージで、「OK」をクリックして続行します。
3. 「IBM InfoSphere CDC New Instance」ダイアログ・ボックスの「Instance」領域で、以下のオプションを構成できます。

オプション	説明
Name	InfoSphere CDC インスタンスの名前を入力します。この名前はユニークである必要があります。
Server Port	InfoSphere CDC が、Management Console およびその他のサーバーを実行しているクライアント・ワークステーションとの通信に使用するポート番号を入力します。 注: このポート番号は、同じサーバーにインストールされている別のアプリケーションで使用することはできません。このポート番号は、Management Console のアクセス・マネージャー・パースペクティブでデータ・ストアのアクセス・パラメーターを指定するときに使用します。InfoSphere CDC は、デフォルト TCP/IP ポート番号の 11101 を表示します。詳しくは、Management Console の資料を参照してください。 注: 同じノードに複数のインスタンスをインストールする場合、各インスタンスのポート番号はユニークである必要があります。
Auto-Discovery Port	このボックスを選択して、Access Server から送信されるオートディスカバリー・ブロードキャストで使用する UDP ポート番号を入力します。オートディスカバリーについて詳しくは、Management Console の資料を参照してください。

オプション	説明
Maximum Memory Allowed	InfoSphere CDC に割り振る RAM の最大量を入力します。構成する各インスタンスに少なくとも 64 MB を割り振る必要があります。デフォルトでは、32 ビットのインスタンスには 512 MB の RAM が割り振られ、64 ビットのインスタンスには 1024 MB の RAM が割り振られます。
Bit-Version	以下のいずれかのオプションを選択して、データベースのビット・バージョンを選択します。 <ul style="list-style-type: none"> • 32 ビット • 64 ビット InfoSphere CDC を 32 ビット・サーバーにインストールしている場合、これらのオプションは使用できません。

4. 「**Windows Service**」領域で、InfoSphere CDC サービスの開始に使用するアカウントを指定できます。以下のオプションのいずれかを選択します。

オプション	説明
Local System account	ローカル・システム管理者のアカウントで InfoSphere CDC サービスを開始します。
This account	指定したユーザー・アカウントで InfoSphere CDC サービスを開始します。 アカウントは、<domain>%<user name> のフォーマットで指定する必要があります。<domain> は環境のドメイン名で、<user name> は指定したドメインの有効なログイン・ユーザー名です。コンピューターがドメインの一部でない場合は、<computer name>%<user name> と指定できます。 「 Password 」ボックスおよび「 Confirm Password 」ボックスに、選択した Windows ユーザー・アカウントに現在関連付けられているパスワードを入力します。InfoSphere CDC のインストール後に Windows ユーザー・アカウントのパスワードを変更した場合は、「Windows Services」ダイアログを使用して、各 InfoSphere CDC サービスに現在設定されているパスワードを変更する必要があります。

5. 「**Database**」領域で、レプリケーション用の表を含むデータベースへのアクセスを構成できます。この手順を完了するには、システム管理者特権が必要です。これで、Management Console のアクセス・マネージャー・パースペクティブでデ

ータ・ストアを追加し、ユーザーにこのデータベースへのアクセス権限を提供できるようにします。詳しくは、Management Console の資料を参照してください。

オプション	説明
User name	指定したデータベースのユーザー名を入力します。
Password	指定したデータベースのパスワードを入力します。
Metadata Schema	InfoSphere CDC メタデータ表に使用するデータベースのスキーマを選択します。 デフォルトとして、上記で入力したユーザー名が使用されます。インストールされている別の InfoSphere CDC インスタンスがそのデータベースで使用しているスキーマを除いて、任意のスキーマを指定できます。このスキーマは、インストールの前提条件の一部として、セットアップまたは決定する必要があります。 注: メタデータ・スキーマには、必ず大文字を使用してください。デフォルトでは、solidDB 内のすべてのスキーマ名 (カタログ名) は、大文字です。
Advanced	「Advanced」ボタンを使用して、solidDB JDBC ドライバーの構成パラメーターを変更できます。JDBC ドライバー・パラメーターの詳細については、「IBM solidDB プログラマー・ガイド」を参照してください。

6. 「Server」領域で、データの複製先または複製元とし、レプリケーション用のすべての表を含む solidDB サーバーを構成できます。単一サーバーまたは HA 構成 (HotStandby) を構成できます。

オプション	説明
Single server	指定した solidDB サーバーのホスト名とポート番号を入力します。
Enable SMA	このチェック・ボックスは、solidDB で共有メモリー・アクセス (SMA) を使用する場合に選択します。
HA Configuration (HotStandby)	指定した 1 次および 2 次 solidDB サーバーのホスト名とポート番号を入力します。

7. 「OK」をクリックして、InfoSphere CDC インスタンスの構成設定を保存します。
8. 「Apply」をクリックして、構成ツールの変更を保存します。

次のタスク

構成を完了した後、InfoSphere CDC を開始できます。

InfoSphere CDC のインスタンスを編集するには (Windows) 手順

1. InfoSphere CDC が開始している場合は、`dmshutdown` コマンドを使用して停止します。
2. コマンド・プロンプトで、以下のコマンドを指定されたディレクトリーで実行し、構成ツールを起動します。

```
¥<InfoSphere CDC Installation Directory>%bin%dmconfigurets
```
3. 変更するインスタンスが開始している場合は、「**Instances**」領域でそのインスタンスを選択し、「**Stop**」をクリックします。
4. 「**Instances**」領域でインスタンスを選択し、「**Edit**」をクリックします。

「InfoSphere CDC Edit Instance」ダイアログが開きます。
5. インスタンスの追加の際に指定した値を、このダイアログ・ボックスで変更できます。
6. 「**Apply**」をクリックして変更を保存し、「**Close**」をクリックします。

構成ツールがインスタンスを変更します。
7. 「**Instances**」領域で変更したインスタンスを選択し、「**Start**」をクリックしてインスタンスを開始します。

InfoSphere CDC のインスタンスを削除するには (Windows) 手順

1. InfoSphere CDC が開始している場合は、`dmshutdown` コマンドを使用して停止します。
2. コマンド・プロンプトで、以下のコマンドを指定されたディレクトリーで実行し、構成ツールを起動します。

```
¥<InfoSphere CDC Installation Directory>%bin%dmconfigurets
```
3. 削除するインスタンスが開始している場合は、「**Instances**」領域でそのインスタンスを選択し、「**Stop**」をクリックします。
4. 「**Instances**」領域でインスタンスを選択し、「**Delete**」をクリックします。
5. 「**Yes**」をクリックして、インスタンスを永続的に削除します。

InfoSphere CDC の構成 (UNIX および Linux)

InfoSphere CDC のインストール後に、インストール・プログラムが、構成ツールを起動します。構成ツールでは、ご使用の環境用に InfoSphere CDC を構成できます。レプリケーションを開始するには、InfoSphere CDC を構成する必要があります。

InfoSphere CDC インスタンスの構成 (UNIX および Linux)

InfoSphere CDC のインスタンスを追加、編集、および削除できます。InfoSphere CDC 構成ツールを使用して、インスタンスに対する作業を行います。

InfoSphere CDC の新しいインスタンスを追加するには (UNIX および Linux)

始める前に

インストール後に InfoSphere CDC の最初のインスタンスを構成する場合は、この手順のステップ 4 に進みます。

手順

1. コマンド・プロンプトで、以下のコマンドを指定されたディレクトリーで実行し、構成ツールを起動します。

```
<InfoSphere CDC Installation Directory>/bin/dmconfigurets
```
2. ウェルカム・メッセージで、**Enter** キーを押して続行します。
3. InfoSphere CDC の新しいインスタンスを追加するために、2 を入力して **Enter** キーを押します。
4. InfoSphere CDC インスタンスの名前を入力し、**Enter** キーを押します。インスタンス名はユニークである必要があります。
5. InfoSphere CDC が、Management Console およびその他のサーバーを実行しているクライアント・ワークステーションとの通信に使用するポート番号を入力します。InfoSphere CDC は、デフォルト・ポート番号の 11101 を表示します。**Enter** キーを押します。

注: このポート番号は、同じサーバーにインストールされている別のアプリケーションで使用することはできません。このポート番号は、Management Console のアクセス・マネージャー・パースペクティブでデータ・ストアのアクセス・パラメーターを指定するときに使用します。詳しくは、Management Console の資料を参照してください。

注: 同じノードに複数のインスタンスをインストールする場合、各インスタンスのポート番号はユニークである必要があります。

6. オートディスカバリー UDP ポート番号を入力するか、**Enter** キーを押して DISABLE のデフォルト値を使用します。このポート番号は、Access Server から送信されるオートディスカバリー・ブロードキャストで使用されます。オートディスカバリーについて詳しくは、Management Console の資料を参照してください。
7. InfoSphere CDC に割り振る RAM の最大量を入力します。構成する各インスタンスに少なくとも 64 MB を割り振る必要があります。デフォルトでは、32 ビットのインスタンスには 512 MB の RAM が割り振られ、64 ビットのインスタンスには 1024 MB の RAM が割り振られます。
8. 構成する solidDB サーバーの構成タイプを選択します。

オプション	説明
Single server	1 を入力し、 Enter キーを押します。
HA Configuration (HotStandby)	2 を入力し、 Enter キーを押します。

9. 使用する構成タイプに従ってホスト名とポート番号を入力します。

オプション	説明
Single server	<ol style="list-style-type: none"> 指定したサーバーのホスト名を入力し、Enter キーを押します。 指定したサーバーのポート番号を入力し、Enter キーを押します。デフォルトは 1964 です。
HA Configuration (HotStandby)	<ol style="list-style-type: none"> 指定した 1 次サーバーのホスト名を入力し、Enter キーを押します。 指定した 1 次サーバーのポート番号を入力し、Enter キーを押します。デフォルトは 1964 です。 指定した 2 次サーバーのホスト名を入力し、Enter キーを押します。 指定した 2 次サーバーのポート番号を入力し、Enter キーを押します。デフォルトは 1964 です。 <p>注: 1 次側と 2 次側は別のノードに配置されると想定されているため、1 次側と 2 次側のデフォルトのポート番号は同じです。例えば、評価の目的で、1 次サーバーと 2 次サーバーを同じノードに配置する場合には、両方のデフォルトのポート番号を同じにすることはできません。</p>

10. 必要に応じて、solidDB で共有メモリー・アクセス (SMA) を使用できるように選択します。

オプション	説明
Use default settings	n を入力し、 Enter キーを押します。
Enable SMA	y を入力し、 Enter キーを押します。

11. 必要に応じて、詳細パラメーター (JDBC パラメーター) を構成します。

オプション	説明
Use default settings	n を入力し、 Enter キーを押します。
Modify settings	<ol style="list-style-type: none"> y を入力し、Enter キーを押します。 <parameter>=<value>;<parameter>=<value>;... 構文を使用して、パラメーター設定を入力します。 <p>注: HA セットアップでは、パラメーター solid_tf_level は、デフォルトで 'CONNECTION' に設定されています。</p>

- 指定したデータベースのユーザー名を入力し、**Enter** キーを押します。
- 指定したデータベースのパスワードを入力し、**Enter** キーを押します。構成ツールが、データベースでスキーマを検索します。
- 使用するメタデータ・スキーマに対応する番号を入力し、**Enter** キーを押します。

15. データベースへのバルク挿入に使用するディレクトリーのパスを入力します。
Enter キーを押します。solidDB データベースと InfoSphere CDC の両方に、このディレクトリーに対する読み取り権限と書き込み権限が必要です。

注:

- InfoSphere CDC のインスタンスごとに、異なるディレクトリーを使用する必要があります。
 - このディレクトリーには、レプリケーション用のデータベース表が含まれることがあります。このディレクトリーへのユーザー・アクセス権限を決定するときは、このことを考慮してください。
16. 構成ツールによって InfoSphere CDC インスタンスが作成され、インスタンスの開始を促すプロンプトが出されます。y を入力して、インスタンスを開始します。

注: 構成によって既存のインスタンスのメタデータが上書きされようとした場合、構成ツールによってプロンプトが出されます。

InfoSphere CDC のインスタンスを編集するには (UNIX および Linux)

手順

1. InfoSphere CDC が開始している場合は、dmshutdown コマンドを使用して停止します。
2. 以下のコマンドを指定されたディレクトリーで実行し、構成ツールを起動します。

```
/<InfoSphere CDC Installation Directory>/bin/dmconfigurets
```
3. InfoSphere CDC のインストール済みインスタンスをリストするために、1 を入力して **Enter** キーを押します。変更するインスタンスの名前を記録します。
4. InfoSphere CDC のインスタンスを変更するために、3 を入力して **Enter** キーを押します。
5. 変更するインスタンスの名前を入力し、**Enter** キーを押します。

構成ツールを使用すると、インスタンスの追加の際に指定したいいくつかの値を編集できます。

6. 変更後、変更を適用してメインメニューに戻るには、5 を入力して **Enter** キーを押します。変更内容を破棄するには、6 を入力して **Enter** キーを押します。

InfoSphere CDC のインスタンスを削除するには (UNIX および Linux)

手順

1. InfoSphere CDC が開始している場合は、dmshutdown コマンドを使用して停止します。
2. 以下のコマンドを指定されたディレクトリーで実行し、構成ツールを起動します。

```
/<InfoSphere CDC Installation Directory>/bin/dmconfigurets
```
3. InfoSphere CDC のインストール済みインスタンスをリストするために、1 を入力して **Enter** キーを押します。削除するインスタンスの名前を記録します。

4. InfoSphere CDC のインスタンスを削除するために、4 を入力して **Enter** キーを押します。
5. 削除するインスタンスの名前を入力し、**Enter** キーを押します。

InfoSphere CDC の開始と停止

このセクションでは、InfoSphere CDC インスタンスの開始および停止の段階的な手順を説明します。

InfoSphere CDC の開始

サポートされている Windows サーバー上に InfoSphere CDC をインストールすると、初期構成後に手動で開始できます。InfoSphere CDC を開始すると、Windows でサービスが開始されます。サービスは、リブート後に自動的に開始します。

サポートされている Linux サーバー上に InfoSphere CDC をインストールすると、コマンドを実行して開始できます。インストール後に InfoSphere CDC を開始して、Management Console でこのインスタンス用のデータ・ストアを作成できます。

InfoSphere CDC を開始するには (Windows)

手順

1. コマンド・プロンプトで、以下のコマンドを指定されたディレクトリーで実行し、構成ツールを起動します。

```
¥<InfoSphere CDC Installation Directory>¥bin¥dmconfigurets
```

2. 「**Instances**」領域で開始するインスタンスを選択し、「**Start**」をクリックします。

構成ツールが InfoSphere CDC のインスタンスを開始します。

次のタスク

Windows の「サービス」ダイアログを使用して、InfoSphere CDC サービスの開始と停止を行うこともできます。

InfoSphere CDC を開始するには (UNIX および Linux)

手順

InfoSphere CDC を実行しているオペレーティング・システムに応じて、以下のいずれかの開始コマンドを実行します。

- `dmts32 - I <instance_name>`
- `dmts64 - I <instance_name>`

InfoSphere CDC の停止

InfoSphere CDC 構成ツールを使用して構成の設定を変更したい場合、InfoSphere CDC を停止することが必要な場合があります。

Windows では、InfoSphere CDC を停止すると Windows でサービスが停止します。サービスは、リブート後にもう一度自動的に開始します。

UNIX と Linux では、コマンドを実行して InfoSphere CDC を停止できます。保守のため、または InfoSphere CDC のアップグレードのためにサーバーやデータベースをオフラインにする前に、このコマンドを使用してください。

InfoSphere CDC を停止するには (Windows)

手順

1. 以下のコマンドを指定されたディレクトリで実行し、構成ツールを起動します。

```
<InfoSphere CDC Installation Directory>/bin/dmconfigurets
```

2. 「Instances」領域で停止するインスタンスを選択し、「Stop」をクリックします。

構成ツールが InfoSphere CDC のインスタンスを停止します。

次のタスク

Windows の「サービス」ダイアログを使用して、InfoSphere CDC サービスの開始と停止を行うこともできます。

InfoSphere CDC を停止するには (UNIX および Linux)

手順

1. Management Console ですべてのサブスクリプションのレプリケーションを終了します。サブスクリプションのレプリケーションの終了方法については、Management Console の資料を参照してください。
2. InfoSphere CDC の停止方法に応じて、以下のいずれかの停止コマンドを実行します。

オプション	説明
<code>dmshutdown -I <instance_name></code>	このコマンドを使用して、InfoSphere CDC を正常にシャットダウンします。 同じ Linux サーバーに、アクティブな InfoSphere CDC インストールが複数存在しており、すべてをシャットダウンしたい場合は、このコマンドを InfoSphere CDC の各インスタンスのインストール・ディレクトリから実行します。
<code>dmterminate -I <instance_name></code>	このコマンドを使用して、Linux サーバーで実行中のすべてのインスタンスについて、すべての InfoSphere CDC プロセスを強制終了します。dmshutdown コマンドを使用しても InfoSphere CDC のシャットダウンを完了できない場合は、このコマンドを使用してください。

Management Console での SQL ステートメントの使用可能化

InfoSphere CDC では、ターゲット表に対して表レベルのクリア操作またはリフレッシュ操作を適用した後、ユーザーによる SQL ステートメントの実行が可能となり

ます。SQL ステートメントは、Management Console の「**Additional SQL**」ダイアログ・ボックスに指定できます。デフォルトでは、セキュリティ上の理由で、この機能は InfoSphere CDC で使用不可に設定されています。この機能は、InfoSphere CDC をインストールしたデータベースに TS_SQL_EXECAUTH という表を作成することにより使用可能に設定できます。この表の構造は重要ではありません。ただし、この表は、InfoSphere CDC の構成中に、メタデータ表と同じスキーマを使用して作成する必要があります。Management Console での SQL ステートメントの指定について詳しくは、Management Console の資料で『リフレッシュ操作を制御するための SQL の指定』を参照してください。

Management Console で SQL ステートメントを使用可能にするには 手順

1. InfoSphere CDC 用に作成したデータベースをターゲット・サーバーで見つけます。InfoSphere CDC の使用方法に応じて、これは InfoSphere CDC にとっての複製先または複製元となるデータベースです。

注: インストール中に、InfoSphere CDC は、InfoSphere CDC プロセスに必要なメタデータ表をこのデータベース内に配置します。

2. SQL ステートメントの指定を使用可能にする場合は、データベース内に TS_SQL_EXECAUTH という名前の表を作成します。

注: 表の構造は任意ですが、InfoSphere CDC の構成時に指定したスキーマでこの表を作成しなければなりません。

InfoSphere CDC がサポートするデータ型

Management Console でレプリケーション用にソース列とターゲット列をマップする場合、どのデータ型に互換性があるかを認識しておく必要があります。レプリケーション用に表をマップする方法については、Management Console の資料の『表のマッピング』のセクションを参照してください。

サポートされているデータ型

このセクションでは、InfoSphere CDC が複製できるデータ型を示します。

- bigint
- binary
- blob
- char
- clob
- date
- decimal
- double precision
- float
- integer
- long varbinary

- long varchar
- nchar
- nclob
- numeric
- nvarchar
- real
- smallint
- time
- timestamp
- tinyint
- varbinary
- varchar

サポートされているデータ型の詳細については、「*IBM solidDB SQL ガイド*」を参照してください。

サポートされているマッピング

このセクションでは、サポートされているデータ型に対して Management Console でサポートされるマッピングを示します。

パブリッシュされるデータ型	サポートされるマッピング
bigint	任意の数値、バイナリー、または LOB データ型
binary	任意のバイナリーまたは LOB データ型
blob	任意のバイナリーまたは LOB データ型
char	任意の文字、可変長文字、CLOB、バイナリー、またはその他の LOB データ型
clob	任意の文字、可変長文字、CLOB、バイナリー、またはその他の LOB データ型
date	任意のデータ型
decimal	任意の数値、バイナリー、または LOB データ型
double precision	任意の数値、バイナリー、または LOB データ型
float	任意の数値、バイナリー、または LOB データ型
integer	任意の数値、バイナリー、または LOB データ型
long varbinary	任意のバイナリーまたは LOB データ型
long varchar	任意の文字、可変長文字、CLOB、バイナリー、またはその他の LOB データ型
nchar	任意の文字、可変長文字、CLOB、バイナリー、またはその他の LOB データ型

パブリッシュされるデータ型	サポートされるマッピング
nclob	任意の文字、可変長文字、CLOB、バイナリー、またはその他の LOB データ型
nvarchar	任意の文字、可変長文字、CLOB、バイナリー、またはその他の LOB データ型
numeric	任意の数値、バイナリー、または LOB データ型
real	任意の数値、バイナリー、または LOB データ型
smallint	任意の数値、バイナリー、または LOB データ型
tim	任意の時刻データ型
timestamp	任意の日付、時刻、またはタイム・スタンプ・データ型
tinyint	任意の数値、バイナリー、または LOB データ型
varbinary	任意のバイナリーまたは LOB データ型
varchar	任意の文字、可変長文字、CLOB、バイナリー、またはその他の LOB データ型

InfoSphere CDC メタデータ表

InfoSphere CDC は、現行のレプリケーション構成に関するデータを表す表集合を維持しています。いろいろな理由で、これらの表をバックアップしたい場合や、その他の同様な作業を行いたい場合があります。ただし、IBM 担当員から要求された場合を除いて、これらの表の内容を変更しないでください。

以下に、InfoSphere CDC が作成するメタデータ表の名前を示します。

- TS_AUTH

注: Management Console でアクセス・マネージャー のパースペクティブに追加したすべてのユーザーに、TS_AUTH メタデータ表に対する GRANT SELECT 特権を付与するようにしてください。Management Console で アクセス・マネージャー のパースペクティブにユーザーを追加する方法と、アクセス・マネージャー のパースペクティブについて詳しくは、Management Console の資料を参照してください。

- TS_BOOKMARK

- TS_CONFAUD

InfoSphere CDC は、構成プロセスで指定したデータベースとスキーマ内にこれらの表を作成およびインストールします。

InfoSphere CDC がターゲット表に表レベルのリフレッシュ操作またはクリア操作を適用した後で SQL ステートメントを指定したい場合には、TS_SQL_EXECAUTH 表を作成し、維持する必要があります。

- TS_AGED_TABLES

このメタデータ表は、InfoSphere CDC for solidDB に固有のもので、この表には、solidDB フロントエンド内の表のエージング状況に関する情報が含まれています。

InfoSphere CDC のコマンド

このセクションでは、InfoSphere CDC で使用可能なコマンドについて説明します。これらのコマンドを使用して、レプリケーションの制御、レプリケーション用の表の管理、レプリケーションのモニター、およびその他のさまざまな作業を行うことができます。

InfoSphere CDC コマンドの使用

InfoSphere CDC コマンドは、コマンド行プロンプトで実行することも、バッチ・ファイルまたはシェル・スクリプトの一部として実行することもできます。コマンドは、InfoSphere CDC インストール・ディレクトリーの `bin` ディレクトリーにあります。コマンドを実行するには、このディレクトリーにナビゲートします。

注: コマンドに使用可能なフラグと各フラグの簡略説明をリストするには、コマンド・プロンプトでコマンドの名前と `-?` フラグを入力し、**Enter** キーを押します。例えば、`dmterminate -?` のように入力します。

コマンド・フォーマット

コマンドごとに、以下の項目の情報を記載しています。

- **構文:** コマンドの名前を示し、コマンド・パラメーターをリストします。
- **パラメーター:** コマンドの中の各パラメーターについて説明し、指定可能な値を示します。
- **結果:** コマンドが正常に終了した場合に、コマンドから返される値を示します。これらの値は、スクリプトを記述する上で役に立つことがあります。このセクションでは、コマンドの実行結果として画面に表示される情報があればそれについても説明します。
- **例:** コマンド実行の 1 つ以上の例を示します。

パラメーター・フォーマット

コマンド・パラメーターの定義での以下の規則に注意してください。

- 不等号括弧 (`< >`) は**必須**パラメーターを表します。
- 大括弧 (`[]`) は**オプション**・パラメーターを表します。パラメーターを省略した場合には、InfoSphere CDC はデフォルト値を使用します。
- 1 つ以上のパラメーターを区切る垂直バー (`|`) は、リストにあるパラメーターのうち 1 つのみ使用可能であることを表します。大括弧 `[]` に囲まれたパラメーターのリストに 1 つ以上の垂直バーがある場合、選択項目はリスト内のパラメーターに限られますが、パラメーターを何も指定しなくてもよいという意味になります。
- 省略符号 (`...`) は、パラメーターまたはオプションを複数回、繰り返すことができるということを意味します。

- 特に記載がなければ、コマンドはすべてのオペレーティング・システムに適用されます。

TSINSTANCE 環境変数の設定

コマンドを使用する前に、TSINSTANCE 環境変数を InfoSphere CDC インスタンスの名前に設定できます。

TSINSTANCE 環境変数を設定した後は、コマンドの実行時にインスタンス名を指定する必要がありません。

Windows プラットフォーム

コマンド・プロンプトで、以下のコマンドを実行します。

```
SET TSINSTANCE=<instance_name>
```

ここで、

- <instance_name> は、InfoSphere CDC インスタンスの名前です。

Linux プラットフォーム

以下のコマンドを実行します。

```
EXPORT TSINSTANCE=<instance_name>
```

ここで、

- <instance_name> は、InfoSphere CDC インスタンスの名前です。

レプリケーション・コマンドの制御

このセクションでは、InfoSphere CDC でレプリケーションを制御するコマンドについて説明します。

dmendreplication: レプリケーションの終了

このコマンドを使用して、指定したサブスクリプションのリフレッシュまたはミラーリングを終了します。このコマンドは、指定されたサブスクリプションのレプリケーションが正常に完了した後、終了します。

構文

```
dmendreplication -I <instance_name> [-c|-i] <-A|-s <subscription_names> ...> [-L <locale>]
```

パラメーター

-I <instance_name>

レプリケーションを終了する InfoSphere CDC インスタンスを指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

- c InfoSphere CDC がレプリケーションを制御された方式で終了することを指定します。このオプションを指定すると、InfoSphere CDC は、進行中の操作をすべて完了し、保留になっている変更内容をターゲット表に適用します。c と i のオプションをいずれも省略した場合は、InfoSphere CDC はデフォルトで c を指定したものとみなします。

- i InfoSphere CDC がレプリケーションをすぐに終了することを指定します。このオプションは、現行の操作をすべて中断します。InfoSphere CDC は、保留になっている変更内容をターゲット表に適用しません。
- A InfoSphere CDC がすべてのサブスクリプションのレプリケーションを終了することを指定します。
- s <subscription_names>
InfoSphere CDC が指示されたサブスクリプションのレプリケーションを終了することを指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。
- L <locale>
InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmendreplication -I myinstance -c -s Finance
```

InfoSphere CDC は、指定されたインスタンスの **Finance** サブスクリプションのレプリケーションを制御された方式で終了します。

dmrefresh: サブスクリプションのリフレッシュ

このコマンドを使用して、指定したサブスクリプションをリフレッシュします。サブスクリプションをリフレッシュすると、InfoSphere CDC は、ターゲット表とソース表を確実に同期します。一般的には、表のレプリケーション方式を「**Refresh**」に設定している場合に、ターゲット表をリフレッシュします。

ただし、レプリケーション方式が「**Mirror**」に設定されているターゲット表もリフレッシュすることが可能です。ミラーリング対象として構成されている表をリフレッシュする場合は、InfoSphere CDC は、ターゲット表がソース表と同期するようにターゲット表をリフレッシュし、ミラーリングの開始点としてジャーナルにログ位置を設定します。

このコマンドは、指定されたサブスクリプションのリフレッシュが正常に完了した後、終了します。このプログラムを実行中に強制終了すると、InfoSphere CDC は指定されたサブスクリプションのレプリケーションをすぐに終了します。

構文

```
dmrefresh -I <instance_name> [-a|-f] <-A|-s <subscription_names> ...> [-L <locale>]
```

パラメーター

-I <instance_name>

1 つ以上のサブスクリプションをリフレッシュする InfoSphere CDC インスタンスを指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

- a InfoSphere CDC がサブスクリプション内のすべてのターゲット表をリフレッシュすることを指定します。
- f リフレッシュのフラグが立っているターゲット表のみを InfoSphere CDC がリフレッシュすることを指定します。-a と -f のオプションをいずれも省略した場合は、InfoSphere CDC はデフォルトで -f を指定したものとみなします。
- A InfoSphere CDC がすべてのサブスクリプションをリフレッシュすることを指定します。
- s <subscription_names>
InfoSphere CDC が指示されたサブスクリプションをリフレッシュすることを指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。
- L <locale>
InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmrefresh -I new_instance -a -s Finance
```

InfoSphere CDC は、**Finance** サブスクリプション内のすべてのターゲット表をリフレッシュします。

dmstartmirror: ミラーリングの開始

このコマンドを使用して、指定したサブスクリプションのミラーリングを開始します。このコマンドは、レプリケーション方式が「**Mirror**」で、状況が「**Refresh**」または「**Active**」の任意の表のミラーリングを開始します。ミラーリングを開始する前に、レプリケーション方式が「**Mirror**」で、状況が「**Refresh**」のすべての表は、最初にサブスクリプションでリフレッシュされます。連続的なミラーリングを開始すると、このコマンドは指定したサブスクリプションのミラーリングを正常に開始した後、終了します。最終的な変更のミラーリングを開始すると、このコマンドは指定したサブスクリプションの最終的な変更のミラーリングを完了した後、終了します。

構文

```
dmstartmirror -I <instance_name> [-c|-n] <-A|-s <subscription_names> ...>
[-L <locale>]
```

パラメーター

- I <instance_name>
ミラーリングを開始する InfoSphere CDC インスタンスを指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

- c InfoSphere CDC が連続的なミラーリングを開始することを指定します。c と n のオプションをいずれも省略した場合は、InfoSphere CDC はデフォルトで c を指定したものとみなします。
- n InfoSphere CDC が最終的な変更のミラーリングを開始することを指定します。
- A InfoSphere CDC がすべてのサブスクリプションのミラーリングを開始することを指定します。
- s <subscription_names>
InfoSphere CDC が指示されたサブスクリプションのミラーリングを開始することを指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。
- L <locale>
InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmstartmirror -I myinstance c -s Finance
```

InfoSphere CDC は、**Finance** サブスクリプションの連続的なミラーリングを開始します。

データベース・トランザクション・ログ・コマンド

このセクションでは、データベース・トランザクションのログやブックマークの管理を支援するコマンドについて説明します。

dmdecodebookmark: ブックマーク詳細情報の表示

このコマンドを使用して、ブックマークの詳細情報を表示します。

構文

```
dmdecodebookmark -I <instance_name> (-b | -f) [-d] [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前。TSINSTANCE 環境変数を InfoSphere CDC インスタンスの名前に設定することができます。これが完了すると、コマンドを実行する際に、インスタンスを指定する必要がなくなります。

-b <bookmark>

16 進エンコード・ストリングのブックマーク。

-f <bookmark_file>

バイナリー・ファイルのブックマーク・ファイル。

[-d] <database_version>

指定したブックマークの生成元が InfoSphere CDC の旧バージョンの場合、ブックマークの生成元のデータベースとバージョン。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmdecodebookmark -f bookmark.txt
```

InfoSphere CDC は bookmark.txt ファイルの情報を表示します。

dmsetbookmark: ブックマークの設定

このコマンドを使用して、サブスクリプションの収集ポイントを変更します。

構文

```
dmsetbookmark -I <instance_name> -s <subscription_name> (-b | -l | -f | -t) [-a] [-L <locale>]
```

パラメーター**-I <instance_name>**

InfoSphere CDC インスタンスの名前。 TSINSTANCE 環境変数を InfoSphere CDC インスタンスの名前に設定することができます。これが完了すると、コマンドを実行する際に、インスタンスを指定する必要がなくなります。

-s <subscription_name>

InfoSphere CDC がブックマークを設定するサブスクリプションの名前。

-b <bookmark>

データベース・ログにおける InfoSphere CDC のミラーリング再開位置を決めるブックマークを指定します。次のミラーリングのときに、InfoSphere CDC は指定の位置で収集します。ブックマークは、dmshowbookmark コマンドから取得される 16 進エンコード・ストリングです。

-l <bookmark>

新しい収集ポイントを示すブックマーク。ブックマークは、dmdecodebookmark コマンドから取得されるストリングです。詳しくは、71 ページの『dmdecodebookmark: ブックマーク詳細情報の表示』を参照してください。

-f <bookmark_file>

データベース・ログにおける InfoSphere CDC のミラーリング再開位置を決めるブックマークを格納しているバイナリー・ファイルを指定します。次のミラーリングのときに、InfoSphere CDC は指定の位置で収集します。ブックマーク・ファイルは、その位置を格納するバイナリー・ファイルです。

-t <datetime>

次回に InfoSphere CDC が指定日時の前後にログに記録されたエントリーで収集を再開することを指定します。

-a 新規の収集ポイント以降、サブスクリプション内のすべての表 (停止している表を除く) をアクティブに設定します。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmsetbookmark -I MYINSTANCE -b 2FC5GJHKLKSJLKL458K9K809IK9  
-s FINANCE
```

InfoSphere CDC は、指定されたインスタンスの **Finance** サブスクリプションにブックマーク位置を設定します。このコマンドは、データベース・ログ内の指示された位置でミラーリングが再開することを指定します。

dmshowbookmark: ブックマーク情報の表示

このコマンドを使用して、ターゲットにコミットされるサブスクリプションの最新のレプリケーション位置を表示します。

構文

```
dmshowbookmark -I <instance_name> -s <subscription_name> [-f] [-v] [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前。TSINSTANCE 環境変数を InfoSphere CDC インスタンスの名前に設定することができます。これが完了すると、コマンドを実行する際に、インスタンスを指定する必要がなくなります。

-s <subscription_name>

ブックマークを表示するサブスクリプションの名前を指定します。

[-f] <file_name>

ブックマーク出力のバイナリー・ファイルの名前を指定します。

[-v]

16 進エンコード・ストリングを含む、ブックマークに関する詳細情報を表示します。表示される情報の量は、ソース・エンジンのタイプとバージョンによって異なります。16 進エンコード・ストリングは常に表示されます。それは、dmdecodebookmark コマンドの表示内容のサブセットです。指定しない場合は、16 進エンコード・ストリングのみが表示されます。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmshowbookmark -I myinstance -s master
```

InfoSphere CDC は、**master** サブスクリプションのブックマーク情報を表示します。

dmshowlogdependency: ログの従属関係の表示

このコマンドを使用して、InfoSphere CDC に使用され、レプリケーションに必要なデータベース・ログに関する情報を表示します。このコマンドを使用して、ログ保存ポリシーをインプリメントします。このコマンドを使用すると、以下の情報を表示することができます。

- 指定されたインスタンスに必要なデータベース・ログのリスト
- 指定されたインスタンスに現在欠落しているデータベース・ログのリスト
- 指定されたインスタンスのデータベース・ログの中にある最も古いオープン・トランザクション
- InfoSphere CDC の指定されたインスタンスが現在、ソースで読み取っているデータベース・ログ
- InfoSphere CDC の指定されたインスタンスが現在、ターゲットに適用しているサブスクリプションのデータベース・ログ

構文

```
dmshowlogdependency -I <instance_name> (-c | -i | -t | -l)  
(-s <subscription_name> | -A) [-v] [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前。 TSINSTANCE 環境変数を InfoSphere CDC インスタンスの名前に設定することができます。これが完了すると、コマンドを実行する際に、インスタンスを指定する必要がなくなります。

- c 指定されたインスタンスに現在欠落しており、レプリケーションに必要なアーカイブ済みデータベース・ログのリストをチェックして表示します。これらのログは、InfoSphere CDC を使用してレプリケーションを開始する前に、リストアップしておく必要があります。ログに欠落がなければ、このコマンドの出力に「OK」が表示されます。出力をスクリプトまたはバッチ・ファイルで解析することができます。
- i 指定されたインスタンスに必要なデータベース・ログの完全なリストを表示します。これらのログは、レプリケーションの開始に必要なログで、その中にはターゲットにまだ適用されていないデータが含まれています。
- t 指定された InfoSphere CDC インスタンスが現在読み取っているサブスクリプションの現行ターゲット・データベース・ログを表示します。これは、ターゲットが確認した現在位置を含むログです。

-I 指定された InfoSphere CDC インスタンスが現在読み取っている現行ソース・データベース・ログを表示します。これは、収集の現在位置を含むログです。

-s <subscription_name>

InfoSphere CDC が現在読み取っているターゲット・データベース・ログを表示するサブスクリプションの名前を指定します。ターゲット・データベース・ログを表示するには、このパラメーターを -t パラメーターと組み合わせて使用します。

-A すべてのサブスクリプションを指定します。

-v 詳細出力を指定します (これを指定しない場合には、出力はスクリプト用にフォーマットされます)。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmshowlogdependency -I MyInstance
```

レプリケーション・コマンドに関する表の管理

このセクションでは、InfoSphere CDC で複製したい表の管理を支援するコマンドについて説明します。

dmdescribe: ソース表の記述

このコマンドを使用して、ソース表マッピングの変更内容をターゲットに送信します。

このコマンドは、指定されたサブスクリプションの記述が正常に完了した後、終了します。

構文

```
dmdescribe -I <instance_name> [-A|-s <subscription_names> ...] [-L <locale>]
```

パラメーター

-I <instance_name>

ソース表マッピングの変更内容をターゲットに送信する InfoSphere CDC インスタンスを指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-A すべてのサブスクリプションに対するソース表マッピングの変更内容を InfoSphere CDC がターゲットに送信することを指定します。

-s <subscription_names>

指示されたサブスクリプションに対するソース表マッピングの変更内容を

InfoSphere CDC がターゲットに送信することを指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmdescribe -I new_instance -s Finance
```

InfoSphere CDC は、指定されたインスタンスのターゲットに、**Finance** サブスクリプションのソース表マッピングの変更内容を送信します。

dmflagforrefresh: リフレッシュ対象にフラグを立てる

このコマンドを使用して、リフレッシュ対象としてソース表にフラグを立てます。リフレッシュ対象として表にフラグを立てると、その先のある時点でリフレッシュしたい表を選択することになります。サブスクリプションのレプリケーション方式として「**Refresh**」を選択した場合に、このプロシージャを使用してください。

構文

```
dmflagforrefresh -I <instance_name> -s <subscription_names>  
<-A|-t <schema>.<table> ...> [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-s <subscription_names>

サブスクリプションの名前を指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-A InfoSphere CDC がサブスクリプション内でリフレッシュ対象としてすべてのソース表にフラグを立てることを指定します。

-t <schema>.<table>

InfoSphere CDC がサブスクリプション内でリフレッシュ対象としてフラグを立てるソース表の名前を指定します。*schema.table* というフォーマットで表の名前を指定する必要があります。表を複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmflagforrefresh -I myinstance -s Finance -A
```

InfoSphere CDC は、指定されたインスタンスの **Finance** サブスクリプション内でリフレッシュ対象としてすべてのソース表にフラグを立てます。

dmmarktablecapturepoint: ソース表に表キャプチャー・ポイントのマークを付ける

このコマンドを使用して、ソース表に表キャプチャー・ポイントのマークを付け、その表をアクティブな状態にします。このコマンドの実行前に表を変更した場合は、その変更内容は複製されません。

変更データのストリーム内の既存の位置をオーバーライドするときは、ソース表に表キャプチャー・ポイントのマークを付けてください。これは、Management Console 以外のアプリケーション (例えばデータベース・プラットフォームのインポート機能またはエクスポート機能など) を使用してソース表とターゲット表を既に同期 (リフレッシュ) しており、ソースとターゲットが相互に同期する時点を知っているときに可能です。InfoSphere CDC は、変更データのストリーム内の現在位置からターゲット表に変更内容をミラーリングします。「Map Tables」ウィザードで表をマップした後に「Mirror (Change Data Capture)」を選択すると、InfoSphere CDC がこの位置を設定します。InfoSphere CDC が設定した位置をオーバーライドする場合は、Management Console で表キャプチャー・ポイントのマークを手動で付けることができます。サブスクリプションのミラーリングを開始する場合、データベースの変更内容をキャプチャーし、ターゲットに複製する時点として設定した位置を InfoSphere CDC が識別します。

構文

```
dmmarktablecapturepoint -I <instance_name> -s <subscription_names>  
<-A|-t <schema>.<table> ...> [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-s <subscription_names>

サブスクリプション名を指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-A InfoSphere CDC がサブスクリプション内のすべてのソース表を対象に、変更データのストリーム内の既存の位置をオーバーライドすることを指定します。

-t <schema>.<table>

InfoSphere CDC が表キャプチャー・ポイントのマークを付けるサブスクリプシ

ョン内のソース表の名前を指定します。 *schema.table* というフォーマットで表の名前を指定する必要があります。表を複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmmarktablecapturepoint -I myinstance -s Finance -A
```

InfoSphere CDC は、**Finance** サブスクリプション内のすべてのソース表を対象に、変更データのストリーム内の既存の位置をオーバーライドします。

```
dmmarktablecapturepoint -I myinstance -s Finance -t myschema.mytable
```

InfoSphere CDC は、**Finance** サブスクリプション内の指定された表をアクティブにします。

dmpark: 表の停止

このコマンドを使用して、ソース表を停止します。ソース表を停止することにより、サブスクリプション内のその表については、変更内容をキャプチャーしないことを InfoSphere CDC に指示します。表を停止した場合、InfoSphere CDC は、それ以降にソース表に変更があっても、その変更内容を複製しなくなるため、ソース表とターゲット表との間に不整合が発生する可能性があります。

注: ソース表を停止にできるようにするため、その表をターゲットにミラーリングしている場合には、サブスクリプションのレプリケーションを終了させる必要があります。詳しくは、68 ページの『dmendreplication: レプリケーションの終了』を参照してください。

構文

```
dmpark -I <instance_name> -s <subscription_names> [-A|-t <schema>.<table> ...> [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-s <subscription_names>

サブスクリプション名を指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-A InfoSphere CDC がサブスクリプション内のすべてのソース表を停止することを指定します。

-t <schema>.<table>

InfoSphere CDC が停止するサブスクリプション内のソース表の名前を指定します。 *schema.table* というフォーマットで表の名前を指定する必要があります。表を複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmpark -I myinstance -s Finance -A
```

InfoSphere CDC は **Finance** サブスクリプションのすべてのソース表を停止します。

dmreadddtable: ソース表の定義の更新

このコマンドを使用して、InfoSphere CDC メタデータの中にあるソース表の定義を更新します。RDBMS を使用してソース表の定義を変更した後に、このコマンドを実行してください。

構文

```
dmreadddtable -I <instance_name> [-A|-t <schema>.<table> ...] [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-A レプリケーションで使用可能なすべてのソース表の定義を InfoSphere CDC が更新することを指定します。

-t <schema>.<table>

InfoSphere CDC が定義を更新するサブスクリプション内のソース表の名前を指定します。 *schema.table* というフォーマットで表の名前を指定する必要があります。表を複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmreadddtable -I new_instance -A
```

InfoSphere CDC は、レプリケーションで使用可能なすべてのソース表の定義を更新します。

dmreassigntable: ターゲット表の定義の更新

このコマンドを使用して、InfoSphere CDC メタデータの中にあるターゲット表の定義を更新します。RDBMS を使用してターゲット表の定義を変更した後に、このコマンドを実行してください。

構文

```
dmreassigntable -I <instance_name> -s <subscription_names>  
<-A|-t <schema>.<table> ...> [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-s <subscription_names>

表を含む InfoSphere CDC サブスクリプションを指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-A InfoSphere CDC がサブスクリプション内のすべてのターゲット表の定義を更新することを指定します。

-t <schema>.<table>

InfoSphere CDC が定義を更新するサブスクリプション内のターゲット表の名前を指定します。schema.table というフォーマットで表の名前を指定する必要があります。表を複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmreassigntable -I new_instance -s Finance -A
```

InfoSphere CDC は **Finance** サブスクリプション内のすべてのターゲット表の定義を更新します。

dmsetreplicationmethod: レプリケーション方式の設定

このコマンドを使用して、サブスクリプション内の表のレプリケーション方式を変更します。このコマンドを実行すると、InfoSphere CDC は「**Active**」な表の状況を「**Refresh**」に変更します。

注: このコマンドを実行する前に、サブスクリプションのレプリケーションを終了する必要があります。

構文

```
dmsetreplicationmethod -I <instance_name> <-r|-m> -s <subscription_names>  
<-A|-t <schema>.<table> ...> [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに `TSINSTANCE` 環境変数を指定することもできます。

-m 表がレプリケーション方式として「**Mirror (Change Data Capture)**」を使用することを指定します。

-r 表がレプリケーション方式として「**Refresh (Snapshot)**」を使用することを指定します。

-s <subscription_names>

サブスクリプションの名前を指定します。

-A サブスクリプション内のすべての表が、指示されたレプリケーション方式を使用することを指定します。

-t <schema>.<table>

指示されたレプリケーション方式を使用するサブスクリプション内のソース表の名前を指定します。`schema.table` というフォーマットで表の名前を指定する必要があります。表を複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmsetreplicationmethod -I myinstance -r -s Finance -A
```

指定された InfoSphere CDC インスタンスで、**Finance** サブスクリプション内のすべての表が、レプリケーション方式として「**Refresh**」を使用します。

```
dmsetreplicationmethod -I new_instance -m -s Finance -t acct.taxcodes
```

指定された InfoSphere CDC インスタンスで、**Finance** サブスクリプション内のソース表 `acct.taxcodes` が、レプリケーション方式として「**Mirror**」を使用します。

レプリケーション・コマンドのモニター

このセクションでは、InfoSphere CDC でのレプリケーションのモニターを支援するコマンドについて説明します。

dmclearevents: イベントのクリア

このコマンドを使用して、Management Console の「**Event Log**」ビューからイベントを削除します。

構文

```
dmclearevents -I <instance_name> [-S|-T-|-B] <-A|-s <subscription_names> ...>
[-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-S InfoSphere CDC がソースからイベントをクリアすることを指定します。

-T InfoSphere CDC がソースとターゲットの両方からイベントをクリアすることを指定します。S、T、および B オプションのいずれも指定しない場合は、InfoSphere CDC はデフォルトで B を指定したものとみなします。

-B InfoSphere CDC がログ位置を設定するサブスクリプション内のソース表の名前を指定します。schema.table というフォーマットで表の名前を指定する必要があります。表を複数指定する場合は、リスト形式で指定してください。

-A InfoSphere CDC がすべてのサブスクリプションのイベントをクリアすることを指定します。

-s <subscription_names>

InfoSphere CDC が指示されたサブスクリプションのイベントをクリアすることを指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmclearevents -I myinstance -S -A
```

InfoSphere CDC は、指定されたインスタンスのサブスクリプションすべてのソースからイベントをクリアします。

```
dmclearevents -I myinstance -T -s Finance Marketing
```

InfoSphere CDC は、指定されたインスタンスの **Finance** サブスクリプションと **Marketing** サブスクリプションのソースとターゲットの両方からイベントをクリアします。

dmgetsubscriptionstatus: サブスクリプションの状況の取得

このコマンドを使用して、サブスクリプションの現在の状態を示す情報をリトリートし、標準出力に結果を送信します。

構文

```
dmgetsubscriptionstatus -I <instance_name> [-p] <-A|-s <subscription_name> ...>
[-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-p InfoSphere CDC が状態情報を標準出力に送信することを指定します。

-A InfoSphere CDC がすべてのサブスクリプションの状態情報をリトリートすることを指定します。

-s <subscription_name>

状態情報をリトリートするサブスクリプションの名前を指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、以下のいずれかを返します。

- 0: 指定されたサブスクリプションの状態が「**Inactive**」になっている場合。
- 1: 指定されたサブスクリプションのいずれかの状態が「**Inactive**」以外になっている場合。
- 負の値: 状況情報のリトリート中にエラーが発生した場合。

例

```
dmgetsubscriptionstatus -I myinstance -p -A
```

InfoSphere CDC は、すべてのサブスクリプションの状態情報をリトリートし、結果を指定されたインスタンスの標準出力に送信します。

dmshoevents: InfoSphere CDC イベントの表示

このコマンドを使用して、InfoSphere CDC イベントを標準出力に表示します。InfoSphere CDC イベントを Management Console の「Event Log」ビューで表示する代わりに、このコマンドを使用することができます。

このコマンドの出力では、最新のイベントがリストの先頭に来るように発生順にイベントが表示されます。

構文

```
dmshoevents -I <instance_name> <-a|-s <subscription> ...>
|-t <source_ID> ...|-s <subscription> ... -t <source_ID> ...> [-h] [-c max_msg]
[-L <locale>]
```

または

```
dmshoevents -I <instance_name> <-a|-s <subscription>|-t
<source_ID> ...> [-h] [-c max_msg] [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

- a** InfoSphere CDC がすべてのサブスクリプションのイベントを表示することを指定します。

-s <subscription>

InfoSphere CDC がイベントを表示するソース・サブスクリプションの名前を指定します。サブスクリプションを複数指定する場合は、リスト形式で指定してください。

-t <source_ID>

InfoSphere CDC がイベントを表示するソース ID を指定します。ソース ID を複数指定する場合は、リスト形式で指定してください。

- h** InfoSphere CDC がイベントのリストの前にヘッダーを表示することを指定します。このオプションは、イベントごとに表示される情報の各項目を識別するのに役立ちます。

-c max_msg

InfoSphere CDC が表示するイベントの最大数を指定します。このパラメーターを省略した場合、またはイベントの総数よりも大きな値を指定した場合、InfoSphere CDC は、指定されたサブスクリプションまたはソース ID、またはその両方のイベントをすべて表示します。

- 最小設定値: 0。イベントは表示されません。
- 最大設定値: 2147483647

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmshoevents -I new_instance -s Finance
```

InfoSphere CDC は、指定されたインスタンスの **Finance** サブスクリプションのイベントをすべて表示します。

```
dmshoevents -I myinstance -a -h
```

InfoSphere CDC は、すべてのサブスクリプションのイベントをすべて表示します。指定されたインスタンスのイベントのリストの前にヘッダーが表示されます。

```
dmshoevents -I newinstance -s Finance -t Atlanta -s Marketing -h -c 20  
dmshoevents -I myinstance -s Finance Marketing -t Atlanta -h -c 20
```

InfoSphere CDC は、**Finance** サブスクリプションと **Marketing** サブスクリプション、および Atlanta ソース ID のイベントのうち最新の 20 件を表示します。指定されたインスタンスのイベントのリストの前にヘッダーが表示されます。

出力例

```
EVENTTIME|EVENTSOURCE|ORIGINATOR|EVENTID|SEVERITY|EVENTPROGRAM|EVENTTEXT
```

```
2006-04-21 17:23:08.817|T|ATLANTA|95|Information|class com.datamirror.ts.target.  
publication.c|Transformation Server Communications ending.
```

```
2006-04-21 17:23:08.614|T|ATLANTA|1538|Information|class com.datamirror.ts.target.  
publication.c|---Transformation Server for ATLANTA terminating normally.
```

```
2006-04-21 17:23:08.333|T|ATLANTA|1537|Information|class com.datamirror.ts.target.  
publication.c|Describe conversation with ATLANTA completed successfully.
```

```
2006-04-21 17:23:07.911|T|ATLANTA|1536|Information|class com.datamirror.ts.target.  
publication.c|Describe conversation started by ATLANTA.
```

```
2006-04-21 17:23:07.333|T|ATLANTA|1531|Information|class com.datamirror.ts.target.  
publication.c|Communication with ATLANTA successfully started on Data channel.
```

```
2006-04-21 17:23:06.973|T|ATLANTA|1534|Information|class com.datamirror.ts.engine.a  
|Code page conversation from the source database's code page 1252 to the target  
database's code page Cp1252 for ATLANTA will be performed by the Remote system
```

各レコードのフィールドは、垂直バー（|）で区切られます。これらのフィールドは、出力の先頭行で識別されます。*EVENTSOURCE* フィールドの *S* はソースを表し、*T* はターゲットを表します。

構成コマンドのエクスポートとインポート

このセクションでは、InfoSphere CDC グローバル構成のエクスポートとインポートを行うためのコマンドについて説明します。

dmexportconfiguration: InfoSphere CDC 構成のエクスポート

このコマンドを使用して、InfoSphere CDC のインスタンスのインストール時に設定した構成の詳細をエクスポートします。構成の詳細は、XML 構成ファイルに送信されます。このコマンドで作成した XML ファイルを InfoSphere CDC の別のインスタンスにインポートするために、dmimportconfiguration コマンドを使用することができます。

注: このコマンドは、Management Console で構成されたサブスクリプション固有の設定をエクスポートしません。Management Console 内でサブスクリプション固有の設定を XML ファイルにエクスポートすることができます。詳しくは、Management Console の資料を参照してください。

このコマンドは対話式であり、パスワードの入力を促すプロンプトが出されます。スクリプトの中でこのコマンドを使用することはできません。

構文

```
dmexportconfiguration <path_to_configuration_file> [-L <locale>]
```

パラメーター

<path_to_configuration_file>

エクスポートする XML 構成ファイルの相対パスまたは絶対パス。相対パスは、InfoSphere CDC のインストール・ディレクトリーを基準としたパスです。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmexportconfiguration c:%configurations%configuration.xml
```

InfoSphere CDC は、指定された相対パスに XML ファイルをエクスポートします。

dmimportconfiguration: InfoSphere CDC 構成のインポート

このコマンドを使用して、dmexportconfiguration コマンドで作成した XML ファイルから InfoSphere CDC 構成設定をインポートします。

注: サイレント・インストールと連携してこのコマンドをスクリプトの中で使用することで、複数のシステムに InfoSphere CDC をデプロイできます。

構文

```
dmimportconfiguration <path_to_configuration_file> [-L <locale>]
```

パラメーター**<path_to_configuration_file>**

インポートする XML 構成ファイルの相対パスまたは絶対パス。相対パスは、InfoSphere CDC のインストール・ディレクトリーを基準としたパスです。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmimportconfiguration c:%configurations%configuration.xml
```

InfoSphere CDC は、指定された相対パスから XML 構成ファイルをインポートします。

その他のコマンド

このセクションでは、InfoSphere CDC のバージョンの判別、通信の検査、シャットダウン、InfoSphere CDC の終了 (UNIX サーバーのみ)、システム・パラメーターの設定、メタデータのバックアップを実行する各種コマンドについて説明します。

dmbackupmd: メタデータのバックアップ

このコマンドを使用して、メタデータ・データベースのバックアップ・コピーを作成します。バックアップ・コピーは、InfoSphere CDC インストール・ディレクトリ内の instance/<instance_name>/conf/backup ディレクトリに作成されます。サブスクリプションの構成と表の状況を変更した後に、バックアップしてください。InfoSphere CDC が実行中でも、メタデータをバックアップすることができます。

構文

```
dmbackupmd -I <instance_name> [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

dmconfigurets - Configure InfoSphere CDC

このコマンドを使用して、InfoSphere CDC 構成ツールを起動します。このツールを使用すると、インスタンスを作成して、InfoSphere CDC のインストール済み環境を構成できます。

構文

```
dmconfigurets [-L <locale>]
```

パラメーター

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

dmmdcommander

このコマンドは、内部使用専用です。

dmmdconsole

このコマンドは、内部使用専用です。

dmset: InfoSphere CDC システム・パラメーターの設定

このコマンドを使用して、InfoSphere CDC システム・パラメーターの表示または変更を行います。Management Console でシステム・パラメーターを変更することもできます。詳しくは、Management Console の資料を参照してください。

注: このコマンドを使用すれば、どのシステム・パラメーターでも設定することができます。ただし、表示されるシステム・パラメーターは、デフォルト以外の値に設定されているもののみです。

構文

```
dmset -I <instance_name> [<parameter_name>[=<parameter_value>]] [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

<parameter_name>

InfoSphere CDC システム・パラメーターの名前を指定します。

<parameter_value>

システム・パラメーターに割り当てる値を指定します。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmset -I myinstance
```

デフォルト以外の値に設定されているシステム・パラメーターをすべて表示します。

```
dmset -I myinstance global_unicode_as_char=false
```

global_unicode_as_char システム・パラメーターを false に設定します。

```
dmset -I myinstance global_unicode_as_char
```

指定されたパラメーターの現行値を表示します。

```
dmset -I myinstance stop_replication=
```

stop_replication システム・パラメーターを削除します。

dmsetaccessserverparams - Access Server のパラメーターの設定

このコマンドは、Access Server へのアクセス・データおよびログイン・データの定義に使用します。このコマンドは、リフレッシュ・ストアード・プロシージャーを使用する場合に必要です。

構文

```
dmsetaccessserverparams [-u <username>] [-p <password>] [-H <hostname>] [-P <port>]
```

パラメーター

- **-u <username>**—アクセス・マネージャー・ユーザー
- **-p <password>**—アクセス・マネージャー・ユーザーのパスワード
- **-H <hostname>**—Access Server 稼働時のワークステーションのホスト名 (システム名) または全 IP アドレス
- **-P <port>**—Access Server への接続に使用するユニークな TCP/IP 番号。Access Server のインストール時および Management Console へのログオン時に、このポート番号を指定します。デフォルト値は 10101 です。

dmsetaccessserverparams の初回使用時には、すべてのパラメーターを指定します。このパラメーター値を指定しない場合、以下のデフォルト値が設定されます。

- ユーザー - Admin
- パスワード - "" (ブランク)
- ホスト - localhost
- ポート - 10101

全部または一部のパラメーターの値の設定が完了したら、このコマンドをもう一度発行することで、これらの値を変更することができます。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmsetaccessserverparams -u dba -p dba -H localhost -P 10101
```

ユーザー「dba」およびパスワード「dba」のアクセス・データとログイン・データを設定しています。

```
dmsetaccessserverparams -H newmachine
```

Access Server を稼働するワークステーションのホスト名を変更しています。

dmshowversion: InfoSphere CDC バージョンの表示

このコマンドを使用して、InfoSphere CDC のバージョンとビルド番号を表示します。IBM 担当員へのお問い合わせの際は、事前にこのコマンドを実行して、実行している InfoSphere CDC のバージョンとビルド番号をご提供ください。

構文

```
dmshowversion [-L <locale>]
```

パラメーター

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

dmshutdown: InfoSphere CDC のシャットダウン

このコマンドを使用して、InfoSphere CDC を正常にシャットダウンします。保守のためにサーバーやデータベースをオフラインにする前、または InfoSphere CDC を最新バージョンにアップグレードする前に、このコマンドを使用することができます。

このコマンドを実行する前に、Management Console ですべてのサブスクリプションのレプリケーションを終了して、確実にシャットダウンを完了するようにしてください。詳しくは、Management Console の資料を参照してください。

このコマンドが InfoSphere CDC のシャットダウンを完了できない場合には、dmterminate コマンドを使用して、強制的にシャットダウンを完了してください。

構文

```
dmshutdown -I <instance_name> [-L <locale>]
```

パラメーター

-I <instance_name>

InfoSphere CDC インスタンスの名前を指定します。あるいは、この値の代わりに TSINSTANCE 環境変数を指定することもできます。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

dmterminate: InfoSphere CDC プロセスの強制終了

注: このコマンドは Windows ではサポートされていません。

このコマンドを使用して、dmshutdown コマンドではシャットダウンを完了できない UNIX サーバーまたは Linux サーバー上で実行中のインスタンスについて、すべて

の InfoSphere CDC プロセスを強制終了します。InfoSphere CDC は、このコマンドの実行に使用した UNIX アカウントで開始したプロセスだけを終了します。

保守のためにサーバーやデータベースをオフラインにする前、または InfoSphere CDC を最新バージョンにアップグレードする前に、このコマンドを使用することができます。

InfoSphere CDC を正常にシャットダウンするには、dmshutdown コマンドを使用してください。dmshutdown が InfoSphere CDC のシャットダウンを完了できない場合は、dmterminate を使用して、dmshutdown の実行後も残っているアクティブな InfoSphere CDC プロセスを強制終了してください。

構文

```
dmterminate [-L <locale>]
```

パラメーター

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

dmmts32: InfoSphere CDC の開始

このコマンドを使用して、InfoSphere CDC の 32 ビット・バージョンを開始します。

構文

```
dmmts32 -I <instance_name> [-L <locale>]
```

パラメーター

-I <instance_name>

開始する InfoSphere CDC インスタンスを指定します。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmmts32 -I -I myinstance
```

InfoSphere CDC が指定されたインスタンスに対して開始します。

dmts64: InfoSphere CDC の開始

このコマンドを使用して、InfoSphere CDC の 64 ビット・バージョンを開始します。

構文

```
dmts64 -I <instance_name> [-L <locale>]
```

パラメーター

-I <instance_name>

開始する InfoSphere CDC インスタンスを指定します。

-L <locale>

InfoSphere CDC インスタンスに使用するロケールの名前。デフォルトはマシンのロケールです。

結果

このコマンドは、成功した場合には 0 の値を返し、失敗した場合にはゼロ以外の値を返します。

例

```
dmts64 -I myinstance
```

InfoSphere CDC が指定されたインスタンスに対して開始します。

InfoSphere CDC のユーザー出口

ユーザー出口を使用して、指定された表でデータベース・イベントが発生する前または発生した後に InfoSphere CDC で実行可能な、一連のアクションを定義することができます。ユーザー出口を使用すれば、ビジネス要件に合わせて環境をカスタマイズすることができます。

Java クラスまたはストアード・プロシージャのユーザー出口をコンパイルした後、Management Console でユーザー出口を構成することができます。ユーザー出口の構成について詳しくは、Management Console の資料の『ユーザー出口の構成』を参照してください。

InfoSphere CDC と一緒にインストールされる Javadoc (API) 情報には、InfoSphere CDC で使用可能な Java クラスのユーザー出口に関する詳細なクラス仕様とインターフェース仕様が記載されています。インターフェースごとに、サポートされる呼び出し可能なメソッドが識別されます。

ユーザー出口に関する Javadoc (API) の資料は、<system drive>:\%<installation directory>%docs%api ディレクトリにあります。ご使用のブラウザでヘルプを開くには、index.html をクリックします。

サンプルのユーザー出口が InfoSphere CDC と共に提供されています。これらのサンプルを環境に合わせて拡張または変更することができます。

表レベルおよび行レベルの操作のためのストアード・プロシージャ ー・ユーザー出口

ストアード・プロシージャは、データベース内に物理的に格納されたプログラム (すなわちプロシージャ) です。ストアード・プロシージャの利点は、ユーザー要求に応じて実行される場合に、データベース・エンジンによって直接実行されるということです。データベース・エンジンは通常、独立したデータベース・サーバー上で動作し、一般的にデータベース要求の処理が高速です。

ユーザー出口プログラムを作成してコンパイルした後、Management Console の「User Exits」タブで、どのユーザー出口ポイント (行レベル操作の前後、または表レベル操作の前後) でユーザー出口を実行するか指定できます。

ストアード・プロシージャ・ユーザー出口の定義

InfoSphere CDC でストアード・プロシージャを定義する場合、以下の点を検討してください。

- 多重定義されたストアード・プロシージャはサポートされません。
- ストアード・プロシージャには少なくとも 2 つのパラメーターが存在し、以下の順序で先頭の 2 つに定義する必要があります。
 - result。整数出力パラメーターで、イベント・ログに任意のエラー・コードを返すのに使用します。
 - returnMsg。文字出力パラメーターで、ログに記録するエラー・メッセージを返すのに使用します。

ストアード・プロシージャ・ユーザー出口のデータベース接続

ストアード・プロシージャ・ユーザー出口プログラムと InfoSphere CDC は、データベースに接続するデフォルト方式と同じ共有接続を使用します。この設定により、デフォルトで、InfoSphere CDC が表に対して行った変更が、ストアード・プロシージャ・ユーザー出口プログラムから確認できるようになります。

ストアード・プロシージャ・ユーザー出口でのデータのリトリート

ストアード・プロシージャにシステム・パラメーターを渡すことにより、ソース表からデータをリトリートできます。以下のタイプのデータをリトリートできます。

- システム値のリトリート (s\$)。s\$ 接頭部をストアード・プロシージャに渡すと、ストアード・プロシージャでソース・データベースのシステム値が使用可能になります。例えば、s\$entry は、InfoSphere CDC がユーザー出口を実行したエントリー・ポイントを識別します。
- ジャーナル管理フィールドのリトリート (j\$)。j\$ 接頭部をストアード・プロシージャに渡すと、ストアード・プロシージャでソース・データベースのジャーナル管理フィールドが使用可能になります。例えば、j\$USER は、ソース表で更新を行った人のユーザー ID を識別します。これは、ソース表で行われた表レベルまたは行レベルの操作を、ストアード・プロシージャを使用して監査する場合に役に立ちます。

- **データ値のリトリート**。ストアード・プロシージャーに渡す接頭部に応じて、ソース・データベースからデータをリトリートして、ストアード・プロシージャーで使用可能にすることができます。例えば、b\$ を使用して、ソース列の更新前イメージをリトリートすることができます。

これらの各値は、ユーザーが作成したストアード・プロシージャー・ユーザー出口に対する入力パラメーターとして使用できます。データのリトリートに使用するフォーマットは、使用している製品によって多少異なります。

- InfoSphere CDC では、フォーマットは `<x>${value}` です。

ここで、`<x>` は接頭部を表しており、`<value>` はリトリートされる値の名前を表しています。

s\$ 接頭部を使用したシステム値のリトリート

この接頭部は、システム値のリトリートに使用されます。以下の表にこれらの値を示し、簡単に説明します。

接頭部と値	データ型	説明
s\$entry	NUMBER	<p>ストアード・プロシージャが実行されたエントリー・ポイントを示します。以下のエントリー・ポイントからストアード・プロシージャを実行できます。</p> <ul style="list-style-type: none"> • 1: InfoSphere CDC が表クリア (切り捨て) 操作の前にストアード・プロシージャを実行したことを示します。 • 2: InfoSphere CDC が表クリア (切り捨て) 操作の後にストアード・プロシージャを実行したことを示します。 • 3: InfoSphere CDC が行挿入操作の前にストアード・プロシージャを実行したことを示します。 • 4: InfoSphere CDC が行挿入操作の後にストアード・プロシージャを実行したことを示します。 • 5: InfoSphere CDC が行更新操作の前にストアード・プロシージャを実行したことを示します。 • 6: InfoSphere CDC が行更新操作の後にストアード・プロシージャを実行したことを示します。 • 7: InfoSphere CDC が行削除操作の前にストアード・プロシージャを実行したことを示します。 • 8: InfoSphere CDC が行削除操作の後にストアード・プロシージャを実行したことを示します。 • 9: InfoSphere CDC が表リフレッシュ操作の前にストアード・プロシージャを実行したことを示します。 • 10: InfoSphere CDC が表リフレッシュ操作の後にストアード・プロシージャを実行したことを示します。

接頭部と値	データ型	説明
s\$srcSysId	VARCHAR	ソース・データの場所をユニークに識別します。
s\$srcTabId	VARCHAR	複製データをターゲットに送信するソース・データベース内のソース表の名前を表します。
s\$tgtTabId	VARCHAR	ソースから複製データを受信するターゲット・データベース内のターゲット表の名前を表します。

js\$ 接頭部を使用したジャーナル管理フィールドのリトリブ

この接頭部は、ソース・システム上で行われた操作に関する情報のリトリブに使用されます。InfoSphere CDC で **jb\$** を使用して、同じ情報をリトリブできます。

以下に、使用可能な値をリストします。

接頭部と値	データ型	説明
j\$CCID	VARCHAR	挿入、更新、または削除の操作を含むトランザクションを識別します。
j\$CODE	VARCHAR	ジャーナル・エントリーまたはログ・エントリーのタイプを識別します。リフレッシュ操作では「U」、ミラーリングでは「R」が使用されます。
j\$CTRR または j\$CNTRRN	VARCHAR	ジャーナル・エントリーまたはログ・エントリーを記録したソース表の相対レコード番号を識別します。 注: CTRR または CNTRRN には、リフレッシュを構成する挿入エントリーに対してストアード・プロシージャを実行する場合に、意味のある情報が含まれます。
j\$ENTT または j\$ENTTYP	VARCHAR	ソース・システム上で操作タイプを識別するジャーナル・コードまたはログ・コードを生成します。
j\$JRN または j\$JOURNAL	VARCHAR	InfoSphere CDC が挿入、更新、または削除の操作を読み取るジャーナルまたはログの名前。

接頭部と値	データ型	説明
j\$JOB	VARCHAR	ソース・システム上で挿入、更新、または削除を行ったジョブの名前を識別します。
j\$MBR または j\$MEMBER	VARCHAR	ソース表の名前またはその別名を識別します。
j\$NBR または j\$JOBNO	VARCHAR	挿入、更新、または削除の操作を行っているソース表上のプログラムのプロセス ID を識別します。
j\$PGM または j\$PROGRAM	VARCHAR	挿入、更新、または削除の操作を行ったソース・システム上のプログラムの名前を識別します。
j\$SEQN または j\$SEQNO	VARCHAR	ジャーナルまたはログ内の挿入、更新、または削除の操作のシーケンス番号を識別します。
j\$SYNM または j\$SYSTEM	VARCHAR	ソース・システムのホスト名を識別します。
j\$USER	VARCHAR	ソース上で挿入、更新、または削除の操作を行ったデータベース・ユーザーの名前を識別します。
j\$USPF	VARCHAR	ソース上で挿入、更新、または削除の操作を行ったオペレーティング・システム・ユーザーの名前を識別します。
j\$TSTP または j\$TIMESTAMP	VARCHAR	ソース上で挿入、更新、または削除の操作、またはリフレッシュを行った日時を識別します。マイクロ秒の精度をサポートする環境では、このジャーナル管理フィールドの日時フォーマットは YYYY-MM-DD-HH:MM:SS.UUUUUU です。その他の場合、InfoSphere CDC は、マイクロ秒の要素 UUUUUU をゼロに設定するか、またはまったく組み込みません。

b\$, a\$, k\$, および d\$ 接頭部を使用したデータ値のリトリブ

データのリトリブには、4 つの接頭部が使用されます。

接頭部	モード	説明
b\$<source column name>	入力	<p>ソース列内のデータの更新前イメージのリトリブに使用します。更新前イメージは、いずれのトランスフォーマーションも適用される前の、ソース表列からのオリジナル・イメージです。</p> <p>例えば、ソース表に対して、以下の UPDATE を行ったとします。</p> <pre>UPDATE source_table set MYCOLUMN = 2 where MYCOLUMN = 1;</pre> <p>これで、この SQL ステートメントを実行する前に MYCOLUMN が 1 であったすべての行が 2 に設定されます。</p> <p>ストアード・プロシージャを定義し、そのストアード・プロシージャで MYCOLUMN の更新前イメージをリトリブする場合には、以下のように指定します。</p> <pre>b\$MYCOLUMN;</pre> <p>これで、値 1 が返されます。</p>

接頭部	モード	説明
a\$<source column name>	入力	<p>ソース列内のデータの更新後イメージのリトリブに使用します。更新後イメージは、ソース表列からの、変換されたデータです。例えば、派生式によって変換されたデータです。</p> <p>例えば、ソース表に対して、以下の UPDATE を行ったとします。</p> <pre>UPDATE source_table set MYCOLUMN = 2 where MYCOLUMN = 1;</pre> <p>これで、この SQL ステートメントを実行する前に MYCOLUMN が 1 であったすべての行が 2 に設定されます。</p> <p>ストアード・プロシージャを定義し、そのストアード・プロシージャで MYCOLUMN の更新後イメージをリトリブする場合には、以下のように指定します。</p> <pre>a\$MYCOLUMN;</pre> <p>これで、値 2 が返されます。</p>
k\$<target key column name>	入力	<p>変更が必要な行を検索するために、ターゲット表へのアクセスに使用します。</p> <p>注: キー列は、監査には使用できません。</p>
d\$<target column name>	入出力	<p>トランスフォーメーション後のデータ値のリトリブに使用します。このデータ値は、ターゲット・データベース内の表の更新に使用されます。ストアード・プロシージャでは、これらの値のみを変更できます。</p>

ストアード・プロシージャ・ユーザー出口の例

以下のコード・スニペットは、ストアード・プロシージャ・ユーザー出口の例です。

コード	コメント
<pre> create or replace procedure PROD.AUDIT_STPROC (result OUT INT, returnMsg OUT CHAR, s\$entry IN NUMBER, s\$srcSysId IN CHAR, s\$srcTabId IN CHAR, s\$tgtTabId IN CHAR, j\$ENTT IN CHAR, a\$IDNO IN NUMBER, a\$PRICE IN NUMBER, a\$DESC IN CHAR, a\$LONGDESC IN CHAR, a\$TRANSDATE IN DATE, d\$IDNO IN NUMBER, d\$PRICE IN NUMBER, d\$DESC IN CHAR, d\$LONGDESC IN CHAR, d\$TRANSDATE IN DATE) </pre>	<p>宣言してストアード・プロシージャに渡すパラメーターは、有効なデータ型でなければなりません。</p> <p>以下のパラメーターは必須のもので、ストアード・プロシージャ内で宣言しておく必要があります。</p> <p>result: ストアード・プロシージャが成功したことを示す「0」の値か、エラーを示す整数を返します。</p> <p>returnMsg: イベント・ログにエラー・メッセージを返します。</p> <p>このストアード・プロシージャでは、以下のパラメーターが宣言されています。</p> <ul style="list-style-type: none"> • s\$entry: ストアード・プロシージャが呼び出されたエントリー・ポイントをリトリートします。この例では、InfoSphere CDC が各エントリー・ポイントでユーザー出口を呼び出します。 • s\$srcSysId: ソース・データの場所をリトリートします。 • s\$srcSysId: ソース・データの場所をリトリートします。 • s\$srcTabId: ソース表の名前をリトリートします。 • s\$srcTabId: ソース表の名前をリトリートします。 • s\$tgtTabId: ターゲット表の名前をリトリートします。 • s\$tgtTabId: ターゲット表の名前をリトリートします。 • j\$ENTT: ソース表で行われた操作のタイプを示すジャーナル・コードをリトリートします。 • j\$ENTT: ソース表で行われた操作のタイプを示すジャーナル・コードをリトリートします。 • a\$: IDNO、PRICE、DESC、LONGDESC、および TRANSDATE のソース列の更新後イメージをリトリートします。 • a\$: IDNO、PRICE、DESC、LONGDESC、および TRANSDATE のソース列の更新後イメージをリトリートします。 • d\$: IDNO、PRICE、DESC、LONGDESC、および TRANSDATE のターゲット列のトランスフォーム済みデータをリトリートします。 • d\$: IDNO、PRICE、DESC、LONGDESC、および TRANSDATE のターゲット列のトランスフォーム済みデータをリトリートします。
<pre> IS ENTRYPOINT VARCHAR(50); BEGIN CASE s\$entry WHEN 16 THEN ENTRYPOINT := 'User Exit program called Before Insert'; WHEN 1048576 THEN ENTRYPOINT := 'User Exit program called After Insert'; WHEN 64 THEN ENTRYPOINT := 'User Exit program called Before Update'; WHEN 4194304 THEN ENTRYPOINT := 'User Exit program called After Update'; END CASE; </pre>	<p>このストアード・プロシージャ・ユーザー出口は、これらのエントリー・ポイントから呼び出すことができます。</p>

コード	コメント
<pre>insert into PROD.AUDIT_TABLE1 values (s\$entry, s\$srcSysId, s\$srcTabId, s\$tgtTabId, j\$ENTT, a\$IDNO, a\$PRICE, a\$DESC, a\$LONGDESC, a\$TRANSDATE, d\$IDNO, d\$PRICE, d\$DESC, d\$LONGDESC, d\$TRANSDATE, ENTRYPOINT);</pre>	<p>このストアード・プロシージャ・ユーザー出口は、これらの値を、<i>PROD.AUDIT_TABLE1</i> に挿入します。</p>
<pre>result := 1; returnMsg := 'OK!'; END AUDIT_STPROC;</pre>	<p>このストアード・プロシージャ・ユーザー出口は成功しました。</p> <p>注: ストアード・プロシージャが '0' を返した場合は、メッセージがイベント・ログに生成されます。</p>

InfoSphere CDC のサンプル・ユーザー出口

InfoSphere CDC には、ユーザーが自分の環境に合わせて拡張または変更できるサンプル・ユーザー出口が用意されています。サンプルは、InfoSphere CDC インストール・ディレクトリーの `samples` ディレクトリーに置かれている `samples.jar` に入っています。Java ファイルには、以下のサンプルが含まれています。

- **ArchiveLogPathUserExitSample.java** — Oracle アーカイブ・ログ・ファイルへの絶対パス (ファイル名と拡張子を含む) を返します。このサンプルは、`com.datamirror.ts.target.publication.userexit.sample` にあります。
- **DEUserExitSample.java** — `%USERFUNC` 列関数を使用する式の中で使用されます。これはユーザーが提供した (式の中の) パラメーターの合計を計算し、その合計に 1 を加えた値を返します。このサンプルは、`com.datamirror.ts.derivedexpressionmanager` にあります。
- **SPUserExitSample.java** — ソースから着信するイメージを使用して、ストアード・プロシージャを呼び出します。このサンプルは、`com.datamirror.ts.target.publication.userexit.sample` にあります。
- **UserExitSample.java** — レプリケーション・イベントにサブスクライブして、発生したイベントの詳細をリトリブします。このサンプルは、`com.datamirror.ts.target.publication.userexit.sample` にあります。
- **UserExitSample1.java** — ターゲット上の表に挿入された新しい行を記録し、それらの行をテキスト・ファイルに保管します。ユーザーは、テキスト・ファイル名をパラメーターで指定します。このサンプルは、`com.datamirror.ts.target.publication.userexit.sample` にあります。
- **PopWindow**。このサンプルは、ダイアログ・ボックスを開いて、通知情報を表示します。X-Windows などの GUI エミュレーターが存在する場合、このサンプルは、UNIX および Linux のインストールで動作します。

以下の点に注意してください。

- サンプル・ユーザー出口を変更せずに実行するには、Management Console で、コンパイルしたユーザー出口への絶対パスを指定する必要があります。例えば、`com.datamirror.ts.target.publication.userexit.sample.UserExitSample` です。
- コンパイルしたサンプル・ユーザー出口は `ts.jar` ファイルに入っており、このファイルは、InfoSphere CDC インストール・ディレクトリーの `lib` ディレクト

リーにあります。ts.jar ファイル内のコンパイルされたユーザー出口には、*.class 拡張子が付くことに注意してください。

- サンプル・ユーザー出口を変更したい場合には、ソース・コードに変更を行った後、そのユーザー出口をコンパイルする必要があります。
- ユーザー出口クラスは、ユーザーのクラスパスにも含まれている必要があります。

Management Console で Java クラスまたはストアード・プロシージャ・ユーザー出口を指定する方法については、Management Console の資料を参照してください。

サンプル・ユーザー出口をコンパイルするには (Windows)

手順

1. InfoSphere CDC を停止します。
2. samples.jar ファイルを、InfoSphere CDC インストール・フォルダー内の lib フォルダーに unzip します。jar ファイルの unzip 時には、フォルダー構造を必ず維持してください。

jar ファイルの unzip 後、以下のようなフォルダー構造になります。

```
<InfoSphere CDC installation folder>%lib%com%datamirror%ts%target%publication%userexit%sample
```

3. サンプル・ユーザー出口を変更します。
4. 変更済みのユーザー出口をコンパイルします。例えば、UserExitSample.java をコンパイルする場合は、コマンド・ウィンドウを開き、lib フォルダーにナビゲートして以下のコマンドを実行します。

```
javac -classpath ts.jar;. com%datamirror%ts%target%publication%userexit%sample%UserExitSample.java
```

このコマンドの実行が成功すると、画面には何も出力されません。

注: このコマンドを実行するには、システムに Java JDK が必要です。

5. コマンドの実行に成功したら、以下のディレクトリーにナビゲートして、UserExitSample.class ファイルを作成したことを確認してください。

```
<InfoSphere CDC installation directory>%lib%com%datamirror%ts%target%publication%userexit%sample
```

6. InfoSphere CDC を開始します。
7. ユーザー出口を構成するための最終ステップとして、Management Console で UserExitSample の絶対パスを指定します。例えば、以下のように指定します。
com.datamirror.ts.target.publication.userexit.sample.UserExitSample

注: .class 拡張子は指定しないでください。

次のタスク

Management Console での Java クラス・ユーザー出口の指定方法については、Management Console の資料を参照してください。

注: 実稼働環境でサンプル・ユーザー出口を使用する場合は、デプロイ前にサンプルをテストする必要があります。IBM は、変更またはカスタマイズされたユーザー出口クラスによって生じた不利な結果に対して責任を負いません。

サンプル・ユーザー出口をコンパイルするには (UNIX および Linux) 手順

1. InfoSphere CDC を停止します。
2. samples.jar ファイルを、InfoSphere CDC インストール・ディレクトリー内の lib ディレクトリーに unzip します。jar ファイルの unzip 時は、ディレクトリー構造を必ず維持してください。

jar ファイルの unzip 後、以下のようなディレクトリー構造になります。

```
<InfoSphere CDC installation directory>/lib/com/datamirror/ts/target  
/publication/userexit/sample
```

3. サンプル・ユーザー出口を変更します。
4. 変更済みのユーザー出口をコンパイルします。例えば、UserExitSample.java をコンパイルする場合は、コマンド・ウィンドウを開き、lib ディレクトリーにナビゲートして以下のコマンドを実行します。

```
javac -classpath ts.jar:. com/datamirror/ts/target/publication/userexit/sample  
/UserExitSample.java
```

このコマンドの実行が成功すると、画面には何も出力されません。

注: このコマンドを実行するには、システムに Java JDK が必要です。

5. コマンドの実行に成功したら、以下のディレクトリーにナビゲートして、UserExitSample.class ファイルを作成したことを確認してください。

```
<InfoSphere CDC installation directory>/lib/com/datamirror/ts/target  
/publication/userexit/sample
```

6. InfoSphere CDC を開始します。
7. ユーザー出口を構成するための最終ステップとして、Management Console で UserExitSample の絶対パスを指定します。例えば、以下のように指定します。
com.datamirror.ts.target.publication.userexit.sample.UserExitSample

注: .class 拡張子は指定しないでください。

次のタスク

Management Console での Java クラス・ユーザー出口の指定方法について詳しくは、Management Console の資料を参照してください。

注: 実稼働環境でサンプル・ユーザー出口を使用する場合は、デプロイ前にサンプルをテストする必要があります。IBM は、変更またはカスタマイズされたユーザー出口クラスによって生じた不利な結果に対して責任を負いません。

競合解決監査表

InfoSphere CDC は、ソース表とターゲット表の競合を解決するときに、解決に関する情報を TS_CONFAUD 表に記録します。InfoSphere CDC は、InfoSphere CDC の構成時に指定したターゲット・メタデータの場所にこの表を作成します。

このセクションでは、以下の内容を説明します。

競合解決監査表の構造

TS_CONFAUD 表を使用して、競合解決がターゲット表に与えた影響を追跡できます。例えば、AFTERIMG 列を照会して、ターゲット表が変更された時点を確認できます。次に、BEFOREIMG 列と AFTERIMG 列の内容を調べて、ターゲット表のデータの基になった、ソース表での変更内容を確認できます。これは、競合解決戦略で問題を識別するのに役立ちます。

競合検出と解決は、Management Console で構成します。詳しくは、Management Console の資料を参照してください。

以下に、TS_CONFAUD 表の構造を示します。

列	説明
CNFTIME	ターゲット上で競合が検出された日時。
SRCTIME	競合データがソース表に適用された時刻。
SRCSYSID	サブスクリプションのソース ID。
SRCSHEMA	ソース表のスキーマ名またはライブラリー名。
SRCNAME	ソース表の名前。
SRCMEMBER	このフィールドはブランクです。
TGTSCHEMA	ターゲット表のスキーマまたはライブラリー。
TGTNAME	ターゲット表の名前。
OPTYPE	競合の原因となったソース上での行レベルの操作。以下のいずれかの値を取ります。 <ul style="list-style-type: none">• 1: ソース表に行が挿入されました。• 2: ソース表で行が更新されました。• 3: ソース表から行が削除されました。
CNFTYPE	検出された競合のタイプ。以下のいずれかの値を取ります。 <ul style="list-style-type: none">• 1: ソース表に行が挿入されました。その行のキーは、既にターゲット表に存在します。• 2: ソース表で行が更新または削除されました。その行のキーは、ターゲット表に存在しません。• 3: ソース表で行が更新または削除されました。ソース表とターゲット表のイメージが一致しません。• 4: 予期しない競合が検出されました。

列	説明
RESMTD	<p>競合解決方式が使用されました。以下のいずれかの値を取ります。</p> <ul style="list-style-type: none"> • 1: ソースが優先 • 2: ターゲットが優先 • 3: 最大値が優先 • 4: 最小値が優先 • 5: ユーザー出口 <p>解決方式が None の場合、この表には行が挿入されません。これらの方式については、InfoSphere CDC の資料を参照してください。</p>
CNFRES	<p>競合が解決されたかどうかを示します。以下のいずれかの値を取ります。</p> <ul style="list-style-type: none"> • Y: 競合が解決されました。 • N: 競合が解決されませんでした。
BEFOREIMG	<p>変更前のソース表内の行の表記。この列のフォーマットについて詳しくは、106 ページの『行イメージ・フォーマット』を参照してください。</p>
BEFORETRNC	<p>BEFOREIMG に格納された更新前イメージが切り捨てられたかどうかを示します。以下のいずれかの値を取ります。</p> <ul style="list-style-type: none"> • Y: 値が切り捨てられました。 • N: 値が切り捨てられませんでした。
AFTERIMG	<p>変更後のソース表内の行の表記。この列のフォーマットについて詳しくは、106 ページの『行イメージ・フォーマット』を参照してください。</p>
AFTERTRNC	<p>AFTERIMG に格納された更新後イメージが切り捨てられたかどうかを示します。以下のいずれかの値を取ります。</p> <ul style="list-style-type: none"> • Y: 値が切り捨てられました。 • N: 値が切り捨てられませんでした。
TGTIMG	<p>レプリケーションが行われる前のターゲット表内の行の表記。この列のフォーマットについて詳しくは、106 ページの『行イメージ・フォーマット』を参照してください。</p>
TGTRNC	<p>TGTIMG に格納されたイメージが切り捨てられたかどうかを示します。以下のいずれかの値を取ります。</p> <ul style="list-style-type: none"> • Y: 値が切り捨てられました。 • N: 値が切り捨てられませんでした。
WINIMG	<p>競合解決が行われた後のターゲット表内の最終行の表記。この列のフォーマットについて詳しくは、106 ページの『行イメージ・フォーマット』を参照してください。</p>

列	説明
WINTRNC	WINIMG に格納されたイメージが切り捨てられたかどうかを示します。以下のいずれかの値を取ります。 <ul style="list-style-type: none"> • Y: 値が切り捨てられました。 • N: 値が切り捨てられませんでした。

行イメージ・フォーマット

監査表内の BEFOREIMG、AFTERIMG、TGTIMG、および WINIMG 列は、ソース表またはターゲット表内の行の表記を示しています。

これらの列内のイメージは、ターゲット・メタデータ・データベース上の VARCHAR データの最大長で制限されます。これらのイメージには、raw、バイナリー、および LOB 列内のデータを除いて、行内のすべての値が含まれます。各列からのデータは、以下のフォーマットで表示されます。

(length:value)

上記のフォーマットで、*value* は列内のデータ、および *length* はデータの表記に使用される文字の数です。イメージでは、数値データは文字ストリングとして表示され、NULL 値は (ヌル) として表示されます。

行イメージは、ソース表および競合解決監査表内の列順序と一致します。これらのイメージは、ターゲット・メタデータ・データベース内の VARCHAR データの最大長より長い場合には切り捨てられる可能性があります。表のキー列は、表内の最初の列でない場合には切り捨てられる可能性があります。

切り捨てられたイメージ

行イメージは、VARCHAR 列の最大長よりも長いと切り捨てられます。監査表に、各イメージ列が切り捨てられたかどうかを示す列があります。例えば、WINTRNC が Y の場合、WINIMG の値は切り捨てられています。切り捨てられた列のフォーマットは、以下のようになります。

(-length:value)

上記のフォーマットで、*value* は切り捨てられた値であり、*length* は切り捨てられたストリング内の文字数です。

監査対象外のデータ型

監査表では、そのイメージ内に以下のデータ型の列は含まれません。

- IMAGE
- NTEXT
- TEXT

ソース表またはターゲット表にこれらのデータ型の行が含まれる場合、イメージは、これらの行を単に見過ごします。バイナリー・データは、16 進数にエンコードされた文字としてイメージ内に現れます。イメージは、サポートされない列からは、どのような情報も格納しません。

Management Console 管理ガイドの付録

このセクションでは、solidDB に固有のユーザー出口およびシステム・パラメータに関する情報を提供します。このセクションは、「InfoSphere Change Data Capture Management Console 管理ガイド」の付録です。

ユーザー出口の構成

ユーザー出口を使用して、指定された表でデータベース・イベントが発生する前または発生した後に InfoSphere CDC で実行可能な、一連のアクションを定義できます。InfoSphere CDC を使用するとき、行レベルの操作または表レベルの操作としてデータベース・イベントを定義します。行レベルの操作には、挿入、更新、削除があります。表レベルの操作には、リフレッシュ、切り捨て操作があります。例えば、InfoSphere CDC が特定のターゲット表に削除操作を複製した後でアラートを送信する、行レベルのユーザー出口プログラムを構成できます。

ユーザー出口は、「Before User Exit」または「After User Exit」としてグループ化できます。

- **Before User Exit:** InfoSphere CDC が行レベルまたは表レベルの操作をターゲット表に複製する前に、実行されます。
- **After User Exit:** InfoSphere CDC が行レベルまたは表レベルの操作をターゲット表に複製した後に、実行されます。

以下のリストで、行レベルまたは表レベルの操作の前または後のユーザー出口プログラムを作成する共通のシナリオを示します。

- InfoSphere CDC が行レベルの操作をターゲット表に複製するタイミングをカスタマイズする。例えば、特定の基準（オリジナルの請求書の日付など）に基づいて挿入、更新、または削除の操作が行われるように、これらの操作のロジックを開発できます。InfoSphere CDC は、オリジナルの請求書の日付（2004 年 1 月、2004 年 2 月、2006 年 11 月など）に基づいて、ユーザー出口を実行し、行レベルの操作（挿入、更新、または削除）を適切なターゲット表に適用できます。
- デフォルトの行レベルまたは表レベルの操作を無効にして、カスタム操作を実行するユーザー出口プログラムを起動することで置き換える。例えば、表レベルの切り捨て操作に応じて、ターゲット表で永久的な削除ではなく一時的な削除を実行するユーザー出口を作成できます。

InfoSphere CDC for solidDB のユーザー出口の構成

InfoSphere CDC for solidDB では、Java クラス・ユーザー出口を構成できます。

Java クラス・ユーザー出口のメソッド名は、事前定義済みです。つまり、ユーザー出口プログラムを有効または無効にすることのみが可能で、InfoSphere CDC for solidDB で提供される UserExitIF インターフェース・クラスをインプリメントするユーザー出口を Java で構成する必要があります。

Java クラスのユーザー出口を構成するには:
手順

1. 「**Configuration**」 → 「**Subscriptions**」をクリックします。
2. サブスクリプションを選択します。
3. 「**Table Mappings**」ビューをクリックし、表マッピングを選択します。

4. 「**Edit Mapping Details**」を右クリックし、選択します。
5. 「**User Exits**」タブをクリックします。
6. 「**User Exit Type**」リストから「**Java Class**」を選択します。
7. 「**Class Name**」ボックスに、**UserExitIF** インターフェースをインプリメントする Java クラス・ユーザー出口の名前を入力します。

例えば、UserExitIF インターフェースをインポート済みであれば、関数でこのインターフェースをインプリメントするユーザー出口プログラム・クラスの定義は、`public class UE1 implements UserExitIF` のようになります。

「**Class Name**」ボックスには、以下を入力する必要があります。

オプション	説明
UE1	スタンドアロン・クラスの場合。
<Java package>.UE1	クラスが Java パッケージに含まれている場合 (com.datamirror.interface.UE1 など)。

ユーザー出口プログラムをコンパイルすることで生成されるファイルは、CLASSPATH 環境変数で参照されるライブラリーまたはフォルダーに置く必要があります。

8. 「**Parameter**」ボックスに、ユーザー出口プログラムで使用可能にするパラメーターを入力します。

初期化プロセスで `getParameter()` メソッドを呼び出して、ユーザー出口プログラム・クラスのパラメーターにアクセスできます。パラメーターの指定に関する規則はありません。このボックスに入力する値は、フリー・フォームです。パラメーター値のストリングの長さは、255 文字を超えることはできません。

9. 以下の操作 (複数可) のほかに InfoSphere CDC で呼び出すユーザー出口プログラムの名前を入力します。

オプション	説明
挿入前	挿入操作を複製する前に、InfoSphere CDC がユーザー出口を実行します。
挿入後	挿入操作を複製した後に、InfoSphere CDC がユーザー出口を実行します。
更新前	更新操作を複製する前に、InfoSphere CDC がユーザー出口を実行します。
更新後	更新操作を複製した後に、InfoSphere CDC がユーザー出口を実行します。
削除前	削除操作を複製する前に、InfoSphere CDC がユーザー出口を実行します。
削除後	削除操作を複製した後に、InfoSphere CDC がユーザー出口を実行します。
リフレッシュ前	リフレッシュ操作を複製する前に、InfoSphere CDC がユーザー出口を実行します。

オプション	説明
リフレッシュ後	リフレッシュ操作を複製した後に、InfoSphere CDC がユーザー出口を実行します。
切り捨て前	切り捨て操作を複製する前に、InfoSphere CDC がユーザー出口を実行します。
切り捨て後	切り捨て操作を複製した後に、InfoSphere CDC がユーザー出口を実行します。

10. 「Apply」をクリックします。

InfoSphere CDC for solidDB のシステム・パラメーター

システム・パラメーターを使用して、InfoSphere CDC の動作を制御できます。レプリケーション環境で特定の構成が必要な場合は、システム・パラメーターを使用して、InfoSphere CDC のデフォルト操作の動作を変更できます。デフォルトのシステム・パラメーター設定は、ほとんどのインストール済み環境に適しています。InfoSphere CDC の構成を理解するまでは、これらのデフォルト設定を維持してください。

InfoSphere CDC には、ソース・データ・ストアおよびターゲット・データ・ストアの動作を制御するシステム・パラメーターがあります。

注:

- アクティブ・レプリケーション時にシステム・パラメーターを変更する場合は、変更を有効にするために、InfoSphere CDC を停止し、再開してください。
- InfoSphere CDC の高位バージョンにアップグレードするときに、システム・パラメーターの既存の設定はすべて維持されます。

一般製品システム・パラメーター

一般製品システム・パラメーターを使用して、InfoSphere CDC の基本機能およびインストール時に指定した情報を制御できます。

retrieve_credentials:

このシステム・パラメーターは、InfoSphere CDC が solidDB Universal Cache で SQL パススルーを目的としてバックエンド・ログイン・データのフェッチを試行するかどうかを定義します。

このパラメーターは、以下のいずれかに設定します。

- true — InfoSphere CDC が、solidDB Universal Cache で SQL パススルーを目的としたバックエンド・ログイン・データのフェッチを試行することを示します。
- false — InfoSphere CDC が solidDB Universal Cache で SQL パススルーを目的としたバックエンド・ログイン・データのフェッチを試行しないことを示します。バックエンド・データ・サーバーが DB2 for iSeries®、または DB2 for z/OS® である場合、「false」に設定する必要があります。

適用先: ソース

デフォルト設定: true

通知システム・パラメーター

通知システム・パラメーターを使用して、特定のイベントに対して「**Event Log**」で InfoSphere CDC メッセージを生成するかどうかを制御できます。

global_shutdown_after_no_heartbeat_response_minutes:

このシステム・パラメーターを使用して、サブスクリプションのアクティブな InfoSphere CDC 処理が停止するまでに、通信が非アクティブである期間を分単位で指定します。許容範囲外の値が指定された場合、デフォルト設定が使用されます。

適用先: ソース

デフォルト設定: 15 分

最小設定値: 3 分

最大設定値: 999 分

global_conversion_not_possible_warning:

このシステム・パラメーターを使用して、以下の状況で、InfoSphere CDC が Management Console の「**Event Log**」に警告を生成するかどうかを制御します。

- 特定のデータ値で、データ変換ができない。
- 範囲外である変換後のデータ型が検出された。

このパラメーターは、以下のいずれかに設定します。

true: 特定のデータ値でデータ変換ができない、または範囲外である変換後のデータ型が検出された場合に、「**Event Log**」に警告を生成します。

false: 特定のデータ値でデータ変換ができない、または範囲外である変換後のデータ型が検出された場合に、「**Event Log**」に警告を生成しません。

適用先: ターゲット

デフォルト設定: False

スループット最大化システム・パラメーター

InfoSphere CDC システム・パラメーターを使用して、ミラーリング時に、ターゲット・データベースの作業負荷を大幅に削減できます。InfoSphere CDC アプライ・プロセスは、ターゲットのトランザクションをグループ化して、作業負荷を削減します。ターゲット・データベース上のどのコミットも、ソース上のコミットに対応します。ただし、ソースで実行されたすべてのコミットが実行されるわけではありません。例えば、ソースが、それぞれに 1 つの操作が含まれる 3 つの小さなトランザクションを実行する場合、ターゲットは 3 つのすべての操作を単一トランザクションの一部としてコミットできます。このシステム・パラメーターのグループ化を使用して、ターゲット・データベースに必要なリソースを大幅に削減できます。デフォルト設定はほとんどのデータベースに適していますが、ターゲット・システムのリソースが限定されていて、待ち時間の増加を許容できる場合は、この設定を調整できます。

mirror_commit_after_max_transactions:

このシステム・パラメーターは、コミットの前にグループ化するトランザクションの最大数を指定します。通常、ターゲット・データベースに発行されるコミットは、ソースで実行されているアプリケーションが発行したコミットに対応します。このシステム・パラメーターを使用して、ターゲット・データベースに発行される頻度を制御することで、コミットを管理できます。このアプローチを使用すると、データベースへの頻繁なコミットのオーバーヘッドを削減できます。

適用先: ターゲット

デフォルト設定: 10

最小設定値: 1

mirror_commit_after_max_seconds:

このシステム・パラメーターは、ターゲット・データベースに小さなトランザクションをコミットするまでの時間を秒単位で指定します。通常、ターゲット・データベースに発行されるコミットは、ソースで実行されているアプリケーションが発行したコミットに対応します。このシステム・パラメーターを使用して、ターゲット・データベースに発行される頻度を制御することで、コミットを管理できます。このアプローチを使用すると、データベースへの頻繁なコミットのオーバーヘッドを削減できます。

適用先: ターゲット

デフォルト設定: 1 秒

最小設定値: 1

mirror_commit_after_max_operations:

このシステム・パラメーターは、コミットを発行する前にターゲット・データベースに適用する必要がある操作の数を指定します。通常、ターゲット・データベースに発行されるコミットは、ソースで実行されているアプリケーションが発行したコミットに対応します。このシステム・パラメーターを使用して、ターゲット・データベースに発行される頻度を制御することで、コミットを管理できます。このアプローチを使用すると、データベースへの頻繁なコミットのオーバーヘッドを削減できます。

適用先: ターゲット

デフォルト設定: 1000

最小設定値: 1

mirror_commit_on_transaction_boundary:

このシステム・パラメーターは、InfoSphere CDC がターゲット・データベースで行うコミットが、常にソース・データベースで発生したコミットに対応するかどうかを示します。ソース・データベースのコミットメント制御を無視すると、InfoSphere CDC で、大きなトランザクションの部分的な結果を表示できるようになります。

このパラメーターは、以下のいずれかに設定します。

- **true:** ソース・データベースのコミットメント制御を無視しません。コミットされたトランザクションのレコードだけがターゲットにミラーリングされます。この設定では、コミットされたトランザクションだけをターゲットに送信することにより、真のトランザクション整合性が提供されます。
- **false:** ソース・データベースのコミットメント制御を無視します。この値は、トランザクション処理のコミットメント制御を無効にします。ミラーリング時に、トランザクション整合性の維持を試みません。

適用先: ターゲット

デフォルト設定: true

refresh_commit_after_max_operations:

このシステム・パラメーターは、リフレッシュ時に各トランザクションを構成する行の数を識別します。リフレッシュ時のターゲット・データベースのワークロードを削減するために、InfoSphere CDC は、リフレッシュを単一の大きなトランザクションとして実行するのではなく、定期的にターゲット・データベースに変更をコミットします。

適用先: ターゲット

デフォルト設定: 1000

最小設定値: 1

エンコード・システム・パラメーター

システム・パラメーターによっては、定義されている Unicode 列のデータを処理するデフォルト方式を設定し、データベースにデフォルトの文字エンコードを設定できます。

global_unicode_as_char:

このシステム・パラメーターは、定義されている Unicode 列のデータを処理するデフォルト方式を示します。サーバー上の InfoSphere CDC インストールごとに、このシステム・パラメーターは、Unicode 列のデータを処理するシステム・デフォルト方式を定義します。Unicode 列がシステム・デフォルトに設定されている場合、このシステム・パラメーターで定義されているとおりに、現行のシステム・デフォルト方式が使用されます。

このパラメーターは、以下のいずれかに設定します。

- **true:** InfoSphere CDC は、Unicode 列のすべてのデータを 1 バイト文字として処理します。この設定は、Unicode 列に 1 バイト文字データが含まれているときに使用します。
- **false:** InfoSphere CDC は、Unicode 列のすべてのデータを連続したビット・ストリームとして処理します。この設定は、Unicode 列に 1 バイト以外の文字データが含まれているときに使用します。このシステム・パラメーターを false に設定すると、InfoSphere CDC は、以前の InfoSphere CDC リリースと同じように 1 バイト以外の文字データを処理します。

注: このパラメーターを `false` に設定することは、複製された Unicode 列の 1 バイト以外の文字データがターゲットで正しく表示されることを保証するわけではありません。複製された 1 バイト以外の文字データについては、Unicode 列のデータが正しく表示されるように、ユーザー出口プログラムまたはその他のカスタマイズの適用が必要になることがあります。ユーザー出口プログラムについて詳しくは、ご使用のプラットフォームの「InfoSphere CDC のエンド・ユーザー向け資料」を参照してください。

適用先: ソース

デフォルト設定: `false`

ディスク・リソース・システム・パラメーター

システム・パラメーターには、InfoSphere CDC のメモリー使用法を制御するものがあります。パフォーマンスを向上させるために、InfoSphere CDC Java 仮想マシンにデフォルト値である 512 MB より大きな値を割り振ることができる場合は、増大したメモリーを使用するようにディスク・リソース・システム・パラメーターを調整できます。

mirror_memory_txqueue_total_mb:

このシステム・パラメーターは、ソースでデータのステージングに使用するメモリーの容量を制御します。パフォーマンスを最適化するために、このシステム・パラメーターは、ソース・データベースに今後存在するコミットされていないデータの最大量を保持するのに十分な大きさの値にする必要があります。

適用先: ソース

デフォルト設定: 15 メガバイト

mirror_memory_txqueue_each_mb:

このシステム・パラメーターは、ソースでデータのステージングに使用するメモリーの容量を制御します。パフォーマンスを最適化するために、このシステム・パラメーターは、ソースで発生する最大のトランザクションのデータを保持するのに十分な大きさの値にする必要があります。

適用先: ソース

デフォルト設定: 3 メガバイト

global_memory_lob_cache_mb:

このシステム・パラメーターは、ターゲットで LOB 値のステージングに使用するメモリーの容量を制御します。パフォーマンスを最適化するために、この値は、複製される最大の LOB 値のデータ全体を保持するのに十分な大きさの値にする必要があります。

デフォルト設定: 2 メガバイト

適用先: ターゲット

mirror_queue_for_buffers_between_cdc_threads_operations:

このシステム・パラメーターは、マルチ・プロセッサを利用する InfoSphere CDC のログの収集機能を制御します。ほとんどの状況で、デフォルト設定を受け入れることができます。拡張が非常に容易な環境では、この値を増やすことができます。

適用先: ソース

デフォルト設定: 100 項目

最小設定値: 100 項目

アプライ・プロセス・システム・パラメーター

システム・パラメーターには、InfoSphere CDC が行、列、データ、およびエラー処理を適用する方法を調整するものがあります。

mirror_end_on_error:

このシステム・パラメーターを使用して、ターゲット・データベースでアプライ・エラーが発生した後で、ミラーリングを終了するかどうかを示します。

このパラメーターは、以下のいずれかに設定します。

- **true:** ターゲット・データベースのアプライ・エラーの後、ミラーリングを終了します。
- **false:** ターゲット・データベースのアプライ・エラーの後、ミラーリングを終了しません。

適用先: ターゲット

デフォルト設定: true

refresh_end_on_error:

このシステム・パラメーターを使用して、アプライ・エラーが発生した後で、リフレッシュを終了するかどうかを示します。

このパラメーターは、以下のいずれかに設定します。

- **true:** アプライ・エラーが発生した後で、リフレッシュを終了します。
- **false:** アプライ・エラーが発生した後で、リフレッシュを終了しません。

適用先: ターゲット

デフォルト設定: true

refresh_with_referential_integrity:

このシステム・パラメーターを使用して、すべてのターゲット表から削除されたデータを再び追加する前に、リフレッシュするかどうかを示します。これは、リフレッシュする表に参照整合性制約がある場合に非常に便利です。

このパラメーターは、以下のいずれかに設定します。

- **true** — InfoSphere CDC は最初に、指定したリフレッシュ順序とは逆にすべてのデータを削除することを示します。リフレッシュ順序を指定する場合、一般的に、参照する子表の前に親表を表示する必要があります。

- false - InfoSphere CDC は、最初に表からすべてのデータを削除せずに、指定された順序で表をリフレッシュすることを示します。

適用先: ソース

デフォルト設定: false

solid_fast_refresh_apply_pipes:

このシステム・パラメーターを使用して、高速リフレッシュのパフォーマンスを向上させます。高速リフレッシュ機能により、バックエンド・データ・サーバーから solidDB フロントエンドへの大容量のデータの複製に必要な時間が削減されます。

このパラメーターの値を、システム内のプロセッサ (コア) の数に設定します。

適用先: ターゲット

デフォルト設定—2

関連資料

『solid_fast_refresh_on』

solid_fast_refresh_on:

このシステム・パラメーターを使用して、高速リフレッシュ機能を制御します。高速リフレッシュ機能により、バックエンド・データ・サーバーから solidDB フロントエンドへの大容量のデータの複製に必要な時間が削減されます。

このパラメーターは、以下のいずれかに設定します。

- true — 高速リフレッシュが有効であることを示します。
- false — 高速リフレッシュが無効であることを示します。

適用先: ターゲット

デフォルト設定: false

関連資料

『solid_fast_refresh_apply_pipes』

特記事項

Copyright © Solid® Information Technology Ltd. 1993, 2010.

All rights reserved.

Solid Information Technology Ltd. または International Business Machines Corporation の書面による明示的な許可がある場合を除き、本製品のいかなる部分も、いかなる方法においても使用することはできません。

本製品は、米国特許

6144941、7136912、6970876、7139775、6978396、7266702、7406489、7502796、および 7587429 により保護されています。

本製品は、米国輸出規制品目分類番号 ECCN=5D992b に指定されています。

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

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

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

〒242-8502

神奈川県大和市下鶴間1623番14号

日本アイ・ビー・エム株式会社

法務・知的財産

知的財産権ライセンス渉外

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

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

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

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

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

IBM Canada Limited
Office of the Lab Director
8200 Warden Avenue
Markham, Ontario
L6G 1C7
CANADA

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

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

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

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

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

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

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

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

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

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

商標

IBM、IBM ロゴ、ibm.com[®]、Solid、solidDB、InfoSphere、DB2、Informix、および WebSphere[®] は、International Business Machines Corporation の米国およびその他の国における商標です。これらおよび他の IBM 商標に、この情報の最初に現れる個所で商標表示 (® または ™) が付されている場合、これらの表示は、この情報が公開された時点で、米国において、IBM が所有する登録商標またはコモン・ロー上の商標であることを示しています。このような商標は、その他の国においても登録商標またはコモン・ロー上の商標である可能性があります。現時点での IBM の商標リストについては、「Copyright and trademark information」(www.ibm.com/legal/copytrade.shtml) をご覧下さい。

Java およびすべての Java 関連の商標およびロゴは Sun Microsystems, Inc. の米国およびその他の国における商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

Microsoft および Windows は、Microsoft Corporation の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。



Printed in Japan

SC88-8169-01



日本アイ・ビー・エム株式会社
〒103-8510 東京都中央区日本橋箱崎町19-21