

WebSphere® WebSphere Process Server for z/OS
バージョン 7.0.0

WebSphere Process Server のインストールおよび構成



WebSphere® WebSphere Process Server for z/OS
バージョン 7.0.0

WebSphere Process Server のインストールおよび構成



本書は、WebSphere Process Server for z/OS バージョン 7、リリース 0、モディフィケーション 0 (製品番号 5655-N53)、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。

本書についてのご意見は、doc-comments@us.ibm.com へ E メールでお寄せください。皆様の率直なご意見をお待ちしています。

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。

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： WebSphere® WebSphere Process Server for z/OS
Version 7.0.0
Installing and Configuring WebSphere Process Server

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 2010.4

© Copyright IBM Corporation 2006, 2010.

目次

表	v
---	---

第 1 章 WebSphere Process Server のインストール 1

WebSphere Process Server のインストールの準備	3
インストールおよび構成の概説	4
必要なスキルの判別	6
スタンドアロン構成	8
Network Deployment 構成	9
zPMT ツールの使用	12
WebSphere Application Server のインストール	15
インストール・メディアから z/OS への製品コードの読み込み	16
IBM SystemPac または ServerPac の使用	17
IBM Custom-Build Product Delivery Option の使用	18
製品データ・セットについての計画	18
WebSphere Process Server のファイル・システム・ディレクトリー	20
WebSphere Process Server のインストール	21
zPMT を使用した構成定義の作成	21
zPMT で生成された JCL を使用したインストール・ジョブの実行	24
USS からのインストール・スクリプトの実行	24
製品保守の適用	26
サービス・レベルの適用または前の受け入れ済みサービス・レベルへの復元	27
アップグレード・プロセスについて	27
JNDILookup Web Service アプリケーションのインストール	31
Message Service クライアントのインストール	33
アンインストール	33
アンインストールの概要	33
WebSphere Process Server for z/OS のアンインストール	34
Business Process Choreographer のアンインストール	37
インストールのトラブルシューティング	37
WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報	37
WebSphere Process Server のログ・ファイル	38

第 2 章 インストール情報 39

スタンドアロン構成と Network Deployment 構成の相違点	39
インストール・メディアのコンテンツ	40
ノード、サーバー、ホスト、およびセルの命名の考慮事項	42
WebSphere Process Server の機能	42
製品のバージョン情報および履歴情報	43
製品のバージョン情報	44

応答ファイルの値	45
zPMT ツールの使用	56

第 3 章 WebSphere Process Server の構成 59

WebSphere Process Server for z/OS の構成	59
DB2 データベース・オブジェクトの設計	60
応答ファイルの作成	63
共通構成の作成	69
データベースの作成および構成	80
DB2 データベースの作成方法の選択	80
メッセージ・エンジンのデータ・ストアの作成	86
JCA 認証別名ユーザー ID への表の特権の付与	88
SIB の正しいスキーマ名の設定	88
サーバントおよび付属 JCL への DB2 ライブラリーの追加	89
Derby JDBC リソースのクリーンアップ	89
DB2 を使用するインストールの検証	90
ConsolidateJAASAuthAliases.py スクリプト	91
Business Process Management コンポーネントの構成	97
デプロイメント環境ウィザードとコンポーネントの 手動構成の違い	97
デプロイメント環境のセットアップ	98
Business Process Management コンポーネントの 手動構成	133
WebSphere Process Server 機能での WebSphere Enterprise Service Bus の再構成	134
サーバーまたはクラスターの SCA サポートの構成	135
サーバーおよびクラスターでの Service Component Architecture サポートに関する考慮事項	137
管理コンソールでのすべての REST サービスの構成	138
サービス・プロバイダーでの REST サービスの 構成	139
サーバー、クラスター、またはコンポーネント用の REST サービスの構成	140
コマンド行を使用した REST サービスの構成	142
Business Process Choreographer の構成	143
Business Space の構成	143
プロファイル管理ツールを使用した Business Space の構成	145
「デプロイメント環境構成」ウィザードの一部として Business Space を構成する	148
ネットワーク・デプロイメント環境のための Business Space の構成	150
Business Space 内で使用するための特定のウィジ エットのセットアップ	181
Business Space のセキュリティーのセットアップ	189
Business Space を構成するためのコマンド (wsadmin スクリプト)	206

ビジネス・ルールおよびセレクターの構成	223	イベント・データベースへのデータの取り込み	258
ビジネス・ルールとセレクターの監査ログの構成	223	WebSphere Business Monitor 用のクロスセル	
コマンドを使用したビジネス・ルールとセレク		Common Event Infrastructure 構成	260
ターの監査の構成	225	WebSphere Business Integration Adapter の構成	261
ビジネス・ルール・マネージャーのインストール		WebSphere Business Integration Adapter の管理の	
の考慮事項	228	セットアップ	261
リレーションシップ・サービスの構成	234	Service Federation Management 用の WebSphere	
リレーションシップ機能に必要な DB2 セットア		Process Server の構成	263
ップ	235	Service Connectivity Management 接続サーバーの	
拡張メッセージング・リソースの構成	236	構成	264
拡張メッセージング・サービスを使用可能にする	236	Service Connectivity Management 接続プロバイダ	
遅延応答を処理するためのリスナー・ポートの拡		の構成	264
張の構成	237	Service Component Architecture モジュールの	
拡張メッセージング・プロバイダーを選択する	239	Service Connectivity Management の使用	268
メッセージング・サーバー環境のセットアップ	245	Service Connectivity Management (SCM) のプロ	
JNDILookup Web Service の構成	246	キシー・ゲートウェイへのマッピング	268
Common Event Infrastructure の構成	246	構成のトラブルシューティング	269
Common Event Infrastructure コンポーネント	247	WebSphere Process Server のエラー	269
管理コンソールを使用した Common Event		検査エラー	279
Infrastructure の構成	249	WebSphere Process Server for z/OS のインストー	
Common Event Infrastructure アプリケーションの		ルおよび構成のメッセージ参照情報	285
デプロイ	251	ログ・ファイル	286
イベント・メッセージングの構成	255		

表

1. zPMT ツールによって使用可能なカスタマイズ環境	13	6. CEI コンポーネントのスタンドアロン・データベース設計の制約事項:.	62
2. 製品のバージョン情報および履歴情報のリンク	44	7. JAAS 認証別名	91
3.	46	8. デプロイメント環境のコンポーネントの関係	112
4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト . . .	47	9.	182
5. zPMT ツールによって使用可能なカスタマイズ環境	57		

第 1 章 WebSphere Process Server のインストール

このセクションでは、WebSphere® Process Server for z/OS® のインストール方法と、その準備についての情報を提供します。

WebSphere Process Server 資料 (PDF 形式)

関連情報

 PDF 資料

WebSphere Process Server 資料 (PDF 形式)

 情報ロードマップ

IBM が提供する Business Process Management 情報ロードマップには、WebSphere Process Server、WebSphere ESB、およびその他の製品に関する情報がポートフォリオに編成されています。

 IBM Education Assistant

IBM Education Assistant により提供される WebSphere Process Server に関するマルチメディア教育モジュール。

 技術情報

WebSphere Process Server サポート > インストール。WebSphere Process Server 製品のインストールについて不明な点がある場合は、このリソースを参照して、製品のインストールからセットアップまで行うことができます。

 概説

製品ライブラリー Web ページの「Overview」タブ。このページを使用して、WebSphere ESB に関連した発表、データ・シート、およびその他の一般ライブラリー資料にアクセスします。

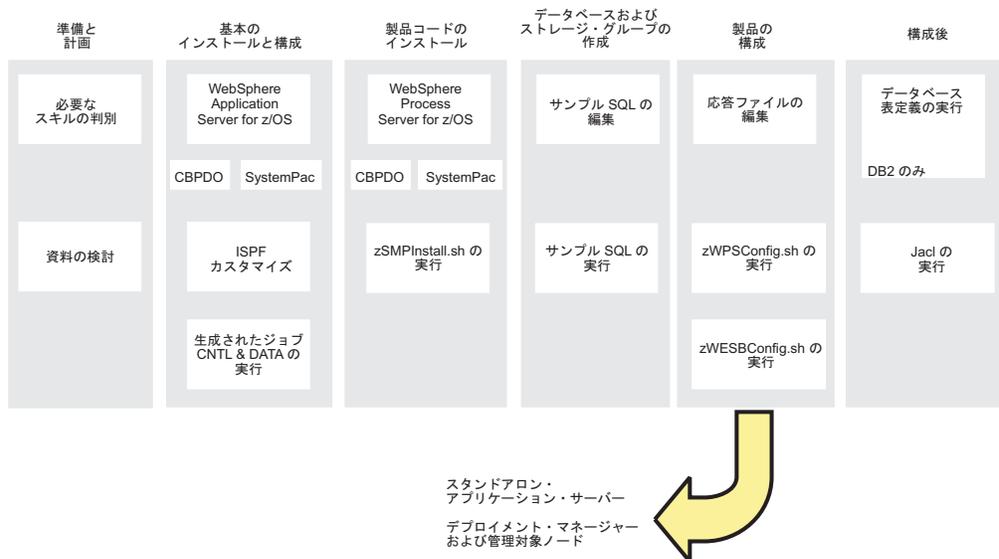
作業の概要: インストールおよび構成

ここでは、サポートされる構成の WebSphere Process Server for z/OS インストールおよび構成作業の流れについて説明します。

始める前に

このセクションでは、インストールの前および後に実行する必要がある作業を含む、IBM® WebSphere Process Server のインストールおよびカスタマイズのコンテキストについて説明します。

以下の図に、製品のインストールおよび構成を行うための作業の流れの概要を示します。



完全な、カスタマイズされた WebSphere Process Server アプリケーションのサービス提供環境を作成するには、以下のステップを完了する必要があります。

1. 製品バイナリーをインストールします。
2. WebSphere Process Server 定義を作成します。
3. 必要に応じてユーザー・プロファイルを拡張します。
4. サーバーを始動します。

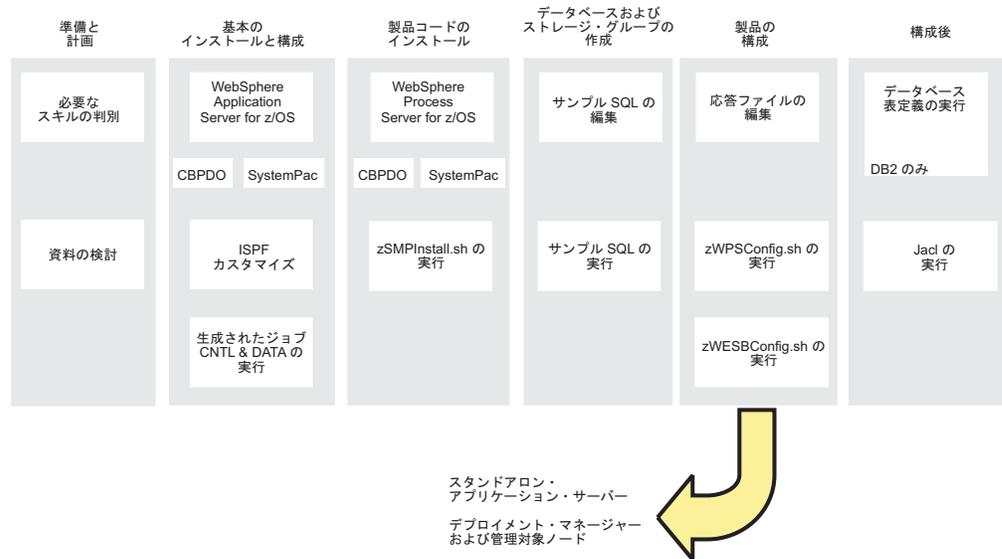
作業の概要: インストールおよび構成

ここでは、サポートされる構成の WebSphere Process Server for z/OS インストールおよび構成作業の流れについて説明します。

始める前に

このセクションでは、インストールの前および後に実行する必要がある作業を含む、IBM WebSphere Process Server のインストールおよびカスタマイズのコンテキストについて説明します。

以下の図に、製品のインストールおよび構成を行うための作業の流れの概要を示します。



完全な、カスタマイズされた WebSphere Process Server アプリケーションのサービス提供環境を作成するには、以下のステップを完了する必要があります。

1. 製品バイナリーをインストールします。
2. WebSphere Process Server 定義を作成します。
3. 必要に応じてユーザー・プロファイルを拡張します。
4. サーバーを始動します。

WebSphere Process Server のインストールの準備

WebSphere Process Server for z/OS のインストールおよび構成には、WebSphere Process Server のほかに WebSphere Application Server for z/OS についての計画アクティビティーが含まれます。

このタスクについて

以降のセクションでは、実行する必要があるタスクを示し、WebSphere Process Server をインストールおよび構成するために役立つ資料リソースの参照先を記載します。

計画を実装するには、以下の事項を考慮しておく必要があります。

タスクの説明	インフォメーション・センターのリソース
必要なスキルを判別する。	『必要なスキルの判別』を参照してください。
システム要件を判別する。	WebSphere Process Server のシステム要件 Web サイトを参照し、 <i>WebSphere Process Server 7.0</i> を選択します。 WebSphere Process Server for z/OS は、WebSphere Application Server for z/OS の上にインストールされます。システム要件のリストについては、 <i>WebSphere Application Server for z/OS</i> インフォメーション・センターの『ハードウェアおよびソフトウェア要件』を参照してください。

タスクの説明	インフォメーション・センターのリソース
セキュリティ・オプションについて理解し、システムの保護に備える。	セキュリティ・オプションは WebSphere Application Server for z/OS で設定されています。WebSphere Application Server for z/OS インフォメーション・センター 内の『セキュリティ計画の概要』情報を参照してください。
必要に応じて、各 z/OS システムでワークロード管理をゴール・モードで実装する。	WebSphere Application Server for z/OS インフォメーション・センター 内の『ワークロード管理 (WLM) の方針』を参照してください。
各 z/OS システム上で Resource Recovery Services を実装する (まだ実装されていない場合)。	WebSphere Application Server for z/OS インフォメーション・センター 内の『Resource Recovery Services の準備』を参照してください。
パフォーマンスおよびモニター・システムについて計画する。	WebSphere Application Server for z/OS インフォメーション・センター 内の『エンド・ユーザー応答時間のモニター』を参照してください。
問題診断手順を計画および定義する。	WebSphere Application Server for z/OS インフォメーション・センター 内の『問題診断計画の方針』を参照してください。
WebSphere Application Server for z/OS をインストールする前に自動再起動管理を検討する。	WebSphere Application Server for z/OS インフォメーション・センター 内の『自動再起動管理』を参照してください。
インストール・メディアから z/OS ヘプログラム・マテリアルをロードする準備としての計画タスクを実行する。	WebSphere Application Server for z/OS インストール・メディアのアンロードに関連する計画タスクについては、WebSphere Application Server for z/OS インフォメーション・センター 内の『インストールの計画』を参照してください。

環境構成変数と応答ファイルの構成の仕方によっては、WebSphere Process Server for z/OS の構成を完了するために、追加の構成作業を行う必要がある場合があります。

インストールおよび構成の概説

WebSphere Process Server for z/OS のインストールおよび構成は、WebSphere Application Server for z/OS のインストールおよび構成と緊密に統合されていて、それに依存しています。したがって、このタスクは、複数のロールにまたがる複数フェーズのプロセスです。

WebSphere Process Server サーバーを作成するには、WebSphere Application Server for z/OS のインストール時に作成されるデフォルト・プロファイルを WebSphere Process Server プロファイルに拡張する必要があります。このプロセスにより、既存の WebSphere Application Server 機能に WebSphere Process Server 機能が追加されます。

z/OS にインストールされる多くの製品とは異なり、WebSphere Process Server のインストールと構成には WebSphere Customization Tool (WCT) を使用しません。その代わりに、2 つのシェル・スクリプトを実行して、製品のインストール・ファイル・システムと製品の構成ファイル・システムとの間のシンボリック・リンクの作成などのタスクを実行します。

インストール

WebSphere Process Server for z/OS をインストールするときには、次の 2 つの主なフェーズがあります。

フェーズ 1

インストール・メディアの内容を z/OS システムに読み込みます。

この最初のフェーズの責任は、インストールの全体的な生産性が向上するようにオペレーティング・システムの使用を計画、保守、および制御するシステム・プログラマーにあります。

インストールの最初のフェーズを完了すると、読み取り専用のインストール・ファイル・システムが生成されます。これは、HFS (階層ファイル構造) または zFS (z-Series ファイル・システム) のいずれかになります。

フェーズ 2

インストール・スクリプト (zWPSInstall.sh) が実行され、製品を使用できるように準備するために必要な定義が作成されます。

この 2 番目のフェーズは、製品管理者が担当します。

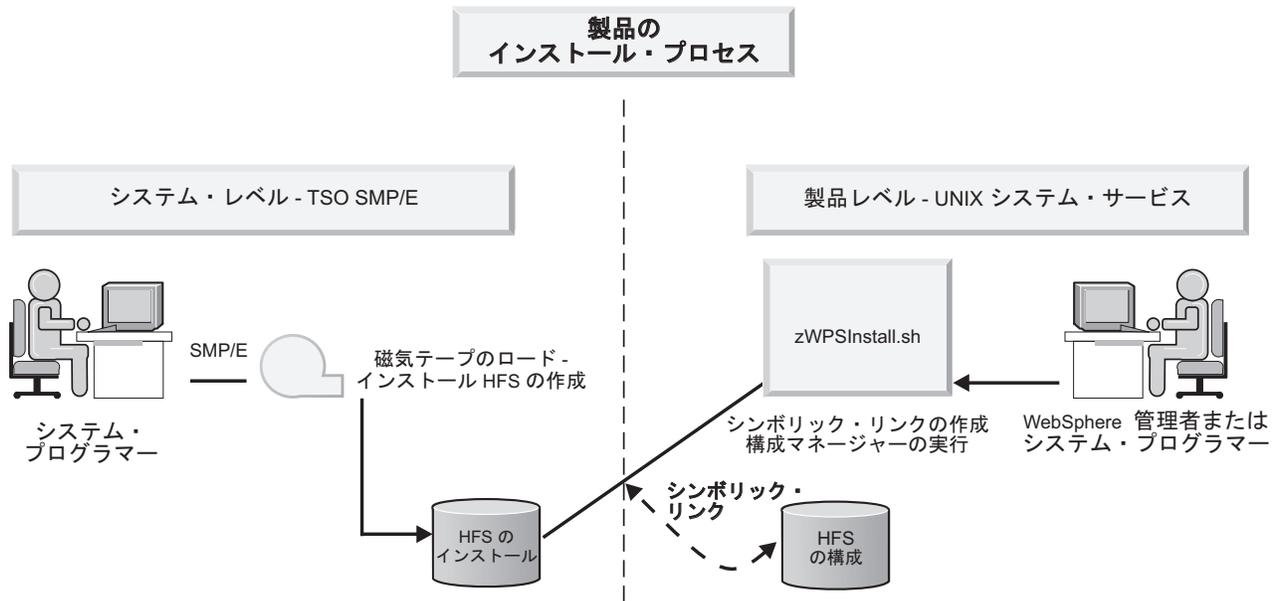
このインストール・フェーズが完了するときには、次のイベントが実行済みです。

- WebSphere Application Server for z/OS 構成ファイル・システムから WebSphere Process Server for z/OS インストール・ファイル・システムへのシンボリック・リンクが作成済みです。
- WebSphere Process Server for z/OS 機能によって管理コンソールを更新する Apache Ant スクリプト・アクションが呼び出されています。
- 必要に応じて applyPTF 処理が実行済みです。

構成ファイル・システムは、構成済みの製品インストールのカスタマイズされた構成文書およびファイルを格納する書き込み可能なファイル・システムです。構成ファイル・システムには、アプリケーション・サーバーのルート・ディレクトリー (/AppServer および /DeploymentManager) と、それらに関連付けられた特定のシンボリック・リンクが含まれます。構成ファイル・システム下にあるノード・タイプごとに、z/OS 下に独自のカタログ式プロシージャが必要です。デフォルトでは、構成ファイル・システムは /WebSphere/V7R0 にマウントされます。

構成ファイル・システムには、WebSphere Process Server インストール・ファイル・システム (デフォルトでは /usr/lpp/zWPS/V7R0) へのシンボリック・リンクがあります。シンボリック・リンクは、JAR ファイルやシェルスクリプトなどのインストール・ファイル・システムの読み取り専用ファイルを指します。

インストール作業の関係者、つまりシステム・プログラマーおよび製品管理者と、インストール・フェーズとの間の関係を以下の図に示します。



製品のインストールは、この両方のフェーズ（製品コードのロードとインストール・スクリプトの実行）が正常に実行されるまで完了しません。

製品のインストールが完了すると、WebSphere 管理者は、製品を使用するための構成を開始できます。

構成

インストール・プロセスの両方の段階を完了すると、WebSphere Process Server for z/OS を構成できます。データベース・タイプによっては、WebSphere Process Server を構成する前に、適切なデータベースとストレージ・グループを作成しておく必要があります。データベースおよびストレージ・グループを作成する方法について詳しくは、83 ページの『SPUFI、DSNTEP2、または DBUtility.sh を使用した DB2 データベースおよびストレージ・グループの作成』を参照してください。

製品の構成スクリプト (zWPSConfig.sh) は、コマンド行から実行されます。

z/OS で WebSphere Process Server for z/OS 構成スクリプトが応答ファイルを使用することによって駆動される構成オプションがいくつかあります。応答ファイルの内容は、WebSphere Process Server for z/OS 構成データで WebSphere Application Server for z/OS プロファイルを拡張するために使用されます。

必要なスキルの判別

プロジェクト・チームを編成するに当たっては、WebSphere Process Server for z/OS の実装に必要なスキルを考慮してください。

プロジェクト・チームを編成するに当たっては、WebSphere Application Server for z/OS の実装に必要なスキルを考慮してください。ここでは、以下の構成をサポートするために必要な推奨される一連のスキルについて説明します。

- 基本構成
- 実稼働環境

ここで説明する z/OS スキルを支援する資料については、Web サイト (z/OS Internet Library) を参照してください。

基本構成の場合

基本構成のサポートに必要な推奨スキルを以下に示します。

- z/OS UNIX[®] システム・サービスおよび階層ファイル・システム (HFS)。HFS と UNIX[®] 機能環境をセットアップするためのスキルです。
- eNetwork Communications Server (TCP/IP) または同等の製品。WebSphere Application Server for z/OS のクライアントおよびサーバーの接続を構成するために必要です。
- Resource Recovery Services (RRS)。Resource Recovery Services を実装し、2 フェーズ・コミット・トランザクションをサポートするために必要です。
- Security Server (RACF[®])。WebSphere Application Server for z/OS クライアントおよびサーバーを認証し、リソースへのアクセスを許可するためのセキュリティー製品です。
- Secure Sockets Layer (SSL)。必要に応じてセキュリティーを有効にする (推奨) ために必要です。
- SMP/E および JCL。
- システム・ロガー。RRS および WebSphere Application Server for z/OS エラー・ログのログ・ストリームをセットアップするために必要です。
- Web サーバー。必要に応じて HTTP クライアントをサポートするために必要です。
- ワークロード管理 (WLM)。
- Java[™] および WebSphere Application Server ツール。アプリケーションの開発とデプロイを支援します。

デプロイするアプリケーションのニーズによっては、アプリケーションに必要なリソース・マネージャーを構成するスキルも必要になります。このようなスキルには、CICS[®]、DB2[®]、IMS[™]、および MQ があります。

実稼働環境の場合

ご使用のシステムを実稼働環境に移行するときには、以下のシステム・スキルが必要です。

- アプリケーション応答測定 (ARM)。
- System Automation (インストールしている場合)、または使用する任意の自動化機能。
- シスプレックス (システムを横断するセルで WebSphere Application Server for z/OS を使用する場合)。
- シスプレックス・ディストリビューター (eNetwork Communications Server の一部。高可用性環境を構築する場合)。
- RMF[™] などのパフォーマンス測定システム。

スタンドアロン構成

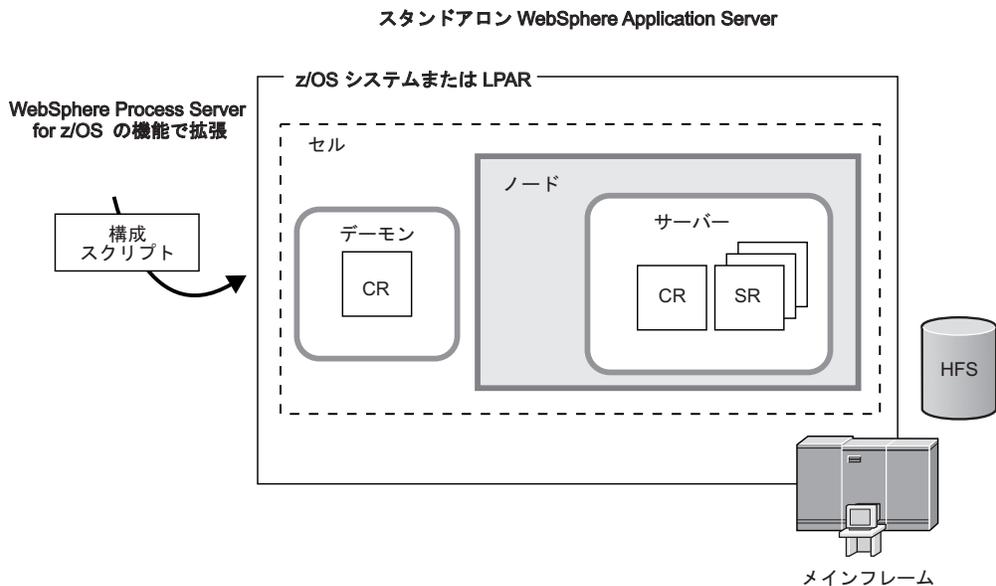
スタンドアロン構成 (基本構成とも呼ばれる) は、WebSphere Process Server for z/OS をデプロイして実行するときに使用できる最も簡単な構成です。

WebSphere Process Server for z/OS のスタンドアロン構成には、単一 z/OS[®] システムまたは LPAR 内に、アプリケーション・サーバーを実行する単一ノードと 1 つのデーモン・サーバーが含まれています。

アプリケーション・サーバーは管理コンソールを実行し、この管理コンソールを使用して、追加のアプリケーションをデプロイおよび実行できます。アプリケーション・サーバーは、ほかのサーバーとは別個に管理されます。

デーモン・サーバーは、常に稼働している固有サーバーで、サーバーのワークロードを分散する 1 つのコントローラー領域を持ちます。

1 つの z/OS[®] システムまたは LPAR 内の、1 つのデーモン・サーバーと、アプリケーション・サーバーを実行する 1 つのノードからなるスタンドアロン構成を次の図に示します。



1 つの z/OS[®] システムまたは LPAR 内の、1 つのデーモン・サーバーと、アプリケーション・サーバーを実行する 1 つのノードからなるスタンドアロン構成を次の図に示します。

スタンドアロン構成を作成する前に、以下の構成を行っておく必要があります。

- WebSphere Application Server for z/OS がスタンドアロン・サーバーとしてインストールされている必要があります。
- シェル内からインストール・スクリプトと拡張スクリプトを実行するために、ご使用の UNIX ユーザー ID が UNIX シェルへのアクセス権を持っている必要があります。シェルへのアクセス権を取得することには、RACF[®] プロファイルに変更を加え、UNIX シェル内にホーム・ディレクトリーを作成する作業が含まれま

す。ホーム・ディレクトリーは、UNIX セッションを開始し、プログラムを実行するために必要な環境変数ファイルを保管する場所です。また、データを保管するためのメイン・ディレクトリーとしてホーム・ディレクトリーを使用することもできます。

- WebSphere Process Server for z/OS 製品コードを使用して WebSphere Process Server for z/OS をインストールし、構成できるように、コードをテープからシステムにロードしておく必要があります。

スタンドアロン構成の利点

企業内でテスト・システムと実動システムを分離する場合は、スタンドアロン構成を使用します。テスト・システムと実動システムを分離することが重要なのは、両システムが分離されていない場合、テスト・システムでテスト中のアプリケーションにより、企業の業務を担っているほかのアプリケーションに影響を及ぼすエラーが発生する可能性があるからです。

以下の状況では、スタンドアロン構成の使用を推奨します。

- 企業規模が非常に大きいため、テスト・グループそれぞれにアプリケーションを実行するための LPAR を提供したい。
- 実稼働環境とテスト環境を 1 つの zSeries® システム上で実行する。例えば、テスト・グループには 1 つの LPAR に対するアクセス権を与え、実動グループにはほかの LPAR すべてに対するアクセス権を与えることができます。
- WebSphere Process Server for z/OS を使用するかどうかを企業で検討中で、適否を検査するために少量のリソースを提供したい。

WebSphere Process Server for z/OS の場合に使用可能な他のタイプの構成は、複数のサーバーで構成される Network Deployment 構成です。Network Deployment 構成は、より複雑な環境に適していて、システム全体でスケーラビリティとセキュリティを向上させることができます。

Network Deployment 構成

初期 Network Deployment 構成に含まれているデプロイメント・マネージャー・サーバーには、デプロイメント・マネージャーが稼働する z/OS® システムのデーモンが含まれています。Network Deployment セルの作成後に、アプリケーション・サーバー・ノードを追加できます。追加するには、新しい空の管理対象ノードを作成して統合するか、またはスタンドアロン・アプリケーション・サーバー・ノードを Network Deployment セルに統合します。

WebSphere Process Server を Network Deployment 環境にインストールするには、統合前に、デプロイメント・マネージャー・ノードと空の管理対象ノードの両方を構成する必要があります。その後、統合を行うために、ジョブ BBOWMNAN を実行します。空のノードをデプロイメント・マネージャーに統合すると、そのノードはデプロイメント・マネージャーによって管理されるため、管理対象ノードになります。管理対象ノードでは、ノード・エージェントは構成できますが、アプリケーション・サーバーは含まれません。アプリケーション・サーバーまたはクラスターは、管理コンソールを使用してノードに追加できます。

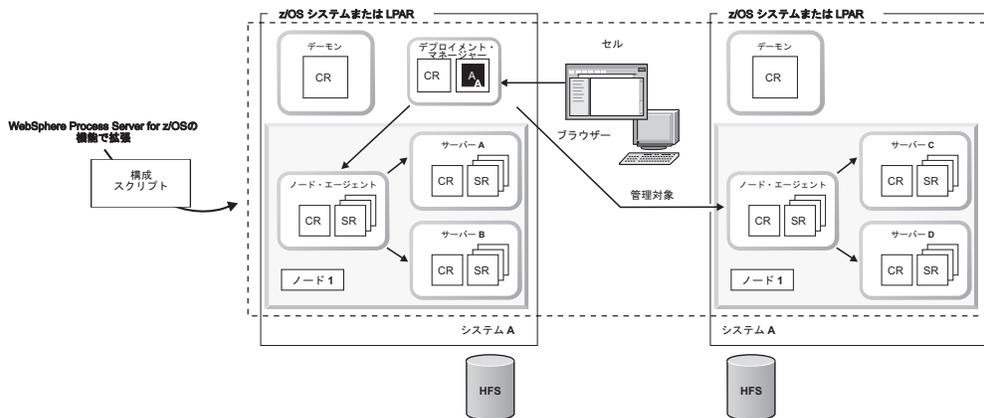
デプロイメント・マネージャーは、管理コンソールのアプリケーションを実行し、セル内のすべての管理対象ノードの構成を管理したり、セル内の選択されたサーバ

ーまたはクラスターにアプリケーションをデプロイしたりするなどの集中管理タスクで使用されます。デプロイメント・マネージャーは 1 つのノード上で稼働し、各アプリケーション・サーバーはさまざまなノード内で稼働します。

基本的な Network Deployment 構成には、次のコンポーネントが含まれます。

- 管理コンソール・アプリケーションを実行する別個のノード内で稼働するデプロイメント・マネージャー・サーバー。これを使用してアプリケーションをデプロイできます。
- セルの一部をホスティングする各 z/OS ターゲット・システム上の 1 つ以上のアプリケーション・サーバー・ノード。各ノードは、ノード・エージェントといくつかのアプリケーション・サーバーで構成されます。各ノードは、デプロイメント・マネージャー・セルに統合する必要があります。
- 各 z/OS システム上の単一ロケーション・サービス・デーモン。セルごとに、常時稼働し、サーバーのワークロードを検査および分散する 1 つのデーモン・サーバーが存在する必要があります。
- DB2® データベース。DB2® Universal Database™ バージョン 8.1 は、Network Deployment 環境のデフォルトのデータベース・タイプです。

次の図に示す Network Deployment 構成では、デプロイメント・マネージャー・サーバーに、デプロイメント・マネージャーが稼働する z/OS システムのためのデーモンが備わっています。このデプロイメント・マネージャーは、2 つのノード・エージェントを通じて、A、B、C、および D という 4 つのサーバーを管理しています。



Network Deployment のトポロジを作成する前に、以下の構成を実行しておく必要があります。

- デプロイメント・マネージャー・ノード、およびそのデプロイメント・マネージャーに統合されていない空の管理対象ノードを持つサーバーで、WebSphere Application Server for z/OS を構成しておく必要があります。インストール・スクリプトの実行前に空の管理対象ノードが統合されていないようにすることは重要です。ノードは、拡張して WebSphere Process Server for z/OS の構成データを含めた後に統合します。
- 使用する UNIX® ユーザー ID に、UNIX シェルへのアクセス権限が付与されている必要があります。これは、インストール・スクリプトと拡張スクリプトをこ

のシェルで実行するためです。アクセス権限を付与するときには、RACF® (セキュリティ) プロファイルを変更し、UNIX シェル内にホーム・ディレクトリーを作成します。ホーム・ディレクトリーは、UNIX セッションを開始したり、プログラムを実行するために必要とされる環境変数ファイルを保管したりする場所です。ホーム・ディレクトリーを、作業データの保存用のメイン・ディレクトリーとして使用することもできます。

- WebSphere Process Server for z/OS 製品コードを使用して WebSphere Process Server for z/OS をインストールおよび構成できるように、コードをテープからシステムにロードしておく必要があります。

デプロイメント・マネージャー・ノードを構成するときは、以下の点に注意してください。

- ターゲット・データ・セットを割り振る場合は、スタンドアロン・アプリケーション・サーバー・ノードで使ったのと同じターゲット・データ・セットを使用することが可能ですが、推奨されていません。各構成のジョブ名は非常に類似しているため、同じターゲット・データ・セットを使用すると、2 つのジョブ・セットを区別しにくくなる場合があります。したがって、デプロイメント・マネージャー・ノードのターゲット・データ・セットを新規作成し、2 つのジョブ・セットを互いに区別しておくことをお勧めします。
- 汎用システム・マウント・ポイントで、ファイル・システム内のデプロイメント・マネージャーの構成以外の、ファイル・システムのルートをすべてのプロセッサで共有します。

Network Deployment セルを構成する場合は特に、使用する WebSphere Process Server for z/OS の構成を計画してから作業に入ることが非常に重要です。多くの選択項目があるので、インストール・プロセス中に正しい決定を下せるように、それらの選択項目に影響する要素について理解しておく必要があります。

Network Deployment 構成の利点

- Network Deployment 構成の主な利点の 1 つは、可用性です。構成に複数の LPAR が含まれていると、Single Point of Failure の削減、および計画された停止および計画外の停止の間の可用性の維持が可能になります。
- メッセージの送信方法を構成できます。例えば、確実に安全な配信 (メッセージが失われないことが保証されており、保護された状態でメッセージが転送される) またはベストエフォート (システム障害発生時にはメッセージが失われる可能性がある) を指定できます。
- Network Deployment セルは、メディエーション・モジュールをホストする数個のサーバーが稼働するようにセットアップできます。メディエーション・モジュールを使用すると、スケーラビリティ (より多くのクライアント接続を処理する能力) とメッセージ・スループットが向上します。
- サーバー・クラスターを構築できます。サーバー・クラスターを使用すると、サーバーのグループの一括管理、およびそれらのサーバーのワークロード管理への参加ができます。
- ご使用のバス環境は、異なる部門に別々の管理ドメインを提供するため、またはテスト機能と実動機能を分離したりするために、数個のスタンドアロン・プロフ

ファイルとデプロイメント・マネージャー・プロファイルで構成されている可能性があります。各プロファイルには、固有の SCA.SYSTEM サービス統合バスが含まれています。

関連資料

zWPSInstall.sh スクリプト

zWPSInstall.sh スクリプトは、WebSphere Application Server プロファイルを変更して WebSphere Process Server をインストールまたはアンインストールするために使用します。デプロイメント・マネージャーを含め、構成内の各ノードでスクリプトを実行します。

45 ページの『応答ファイルの値』

WebSphere Process Server for z/OS 製品定義をインストールするインストール・ジョブを実行すると、サンプル応答ファイルがインストール・ファイル・システムにインストールされます。これらの応答ファイルをコピーし、実現する構成に従って編集し、拡張スクリプトの実行時にその応答ファイルを渡します。

zWPSConfig.sh スクリプト

インストールした WebSphere Process Server for z/OS を構成および拡張するとき使用するスクリプトです。デプロイメント・マネージャーを含め、構成内の各ノードでスクリプトを実行します。

WebSphere Process Server をデプロイメント・マネージャー・ノードへインストールするための JCL

デプロイメント・マネージャー・ノードを拡張するための JCL

WebSphere Process Server を空のノードへインストールするための JCL

空のノードを拡張するための JCL

zPMT ツールの使用

zPMT は、z/OS Profile Management Tool の略です。zPMT ツールはワークステーション・ベースのツールで、情報を取り込み、カスタマイズされた JCL バッチ・ジョブと構成応答ファイル (z/OS システムの WebSphere Process Server の作成に使用) を生成します。

zPMT ツールの重要な機能の一部として、以下のタスクを実行する機能があります。

- 情報の取り込み。
- カスタマイズされた JCL バッチ・ジョブと構成応答ファイルの生成。
- z/OS システムへの JCL バッチ・ジョブと構成応答ファイルのアップロード。

zPMT ツールは、Eclipse ベースのツールである WebSphere Customization Tool (WCT) の一部です。zPMT ツールを使用するには、ワークステーションに WCT をインストールして起動し、そこから zPMT を開始する必要があります。WebSphere Customization Tool のインストール方法については、『WebSphere Customization Tool のインストール』を参照してください。

ワークステーションで zPMT ツールを実行し、一連のウィンドウに情報を入力します。入力した情報は、WCT のカスタマイズ・ロケーション (ハード・ディスク上のディレクトリー構造) に保管されます。生成された JCL ジョブと応答ファイル (z/OS サーバー用の WebSphere Process Server の作成に使用) は、このディレクトリー構造に保管されます。

JCL ファイルは、zPMT ツールのプロセス機能を使用して z/OS システムに送信されます。JCL ファイルは、CNTL と DATA という 2 つの区分データ・セットに格納されます。

- CNTL データ・セットには、WebSphere Process Server for z/OS のインストールと構成に使用される生成済み JCL ジョブが格納されます。
- DATA データ・セットには、拡張スクリプトに渡される構成応答ファイルが格納されます。

zPMT ツールを使用すると、データ・セットをアップロードの一部として割り振ることができます。

zPMT ツールを使用するセルの計画プロセスは、ISPF パネルを使用した際と同じです。zPMT ツールにより、ISPF パネルと同じ情報の入力を要求するプロンプトが表示されますが、順序が若干異なる場合もあります。zPMT ツールを使用する命名規則とポートの割り振りチャートは、ISPF パネルを使用した際と同じです。

zPMT ツールによって作成可能な構成のタイプ

zPMT ツールを使用して、3 つのタイプの構成を作成することができます。以下の表に、zPMT ツール用語と ISPF 用語の相互参照を示します。

表 1. zPMT ツールによって使用可能なカスタマイズ環境

zPMT ツールのオプション	対応する ISPF オプション
WebSphere Process Server for z/OS Process Server	スタンドアロンの Process Server ノード
WebSphere Process Server for z/OS のデプロイメント・マネージャー	Network Deployment セル
WebSphere Process Server for z/OS 管理対象 (カスタム) ノード	空の管理対象ノード

zPMT ツールは、ワークスペース内に保存済み構成定義を作成します。アップロード機能により、z/OS システムにファイルを FTP で転送することができます。

zPMT ツールに入力された変数は、ハード・ディスク上のワークスペースの一部として保存されます。このワークスペースは、プロジェクトに関するすべての情報をツールが保持するエリアです。構成に対して入力した変数は、zPMT ツールの応答ファイルに登録されます。このファイルは、変数とユーザーの値を保持するフラット・ファイルです。この応答ファイルは、ISPF ダイアログの SAVECFG ファイルの概念と似ていますが、形式が異なります。

zPMT ツールの応答ファイルは、拡張に使用される応答ファイルとは異なります。ジョブをカスタマイズする場合は、別の変数 (例えば、ジョブを構成するための JCL ヘッダーの 4 つの変数) を使用します。拡張に使用される応答ファイルは、

「プロセス」をクリックして取得します。このファイルは、カスタマイズしたジョブを z/OS システムまたはローカル・ディレクトリーにアップロードする際にも使用されます。

この応答ファイルの名前は、BPZRSPX の名前形式に従っています。この「X」は、以下のノードによって値が異なります。

- A = スタンドアロン・ノード。
- M = デプロイメント・マネージャー・ノード。
- N = 管理対象ノード。

WebSphere Customization Tool のインストール

WebSphere Customization Tools を使用して、カスタマイズされたジョブおよび拡張応答ファイルを作成すると、WebSphere Process Server のインストールをグラフィカル環境で調整できます。また、WebSphere Customization Tools を使用すると、構成を他のユーザーと共有し、拡張応答ファイルの小規模な更新を素早く簡単に行い、カスタマイズされたスクリプトをターゲットの z/OS システムにアップロードすることができます。

このタスクについて

WebSphere Customization Tools をインストールすると、zPMT ツールを使用して WebSphere Process Server のインストールをカスタマイズすることができます。

手順

1. 以下の該当する WebSphere Customization Tools バージョン 7.0.x Web サイトにアクセスします。
 - WebSphere Customization Tools バージョン 7.0 for Windows
 - WebSphere Customization Tools バージョン 7.0 for Linux
2. 「**Download package**」の下の「**FTP**」または「**DD**」をクリックして、WCT インストール・ファイルをダウンロードします。「**Installation Instructions**」という見出しの下の指示に従って、WCT をインストールします。
3. WebSphere Process Server インストール・ファイル・システムのルートにナビゲートします。デフォルトのパスが使用された場合、これは /usr/lpp/zWPS/V7R0 となります。インストール・ファイル・システムのルートから、util/WCT にナビゲートします。
4. **zWBI.wct** をワークステーションにダウンロードします。これはバイナリー・ファイル・ダウンロードです。
5. ワークステーションで WebSphere Customization Tool アプリケーションを開始します。
6. 「ヘルプ」>「ソフトウェアの更新 (Software Updates)」>「拡張のインストール (Install Extension)」をクリックします。
7. 「WebSphere Customization Tool 拡張ロケーション (WebSphere Customization Tools Extension Locations)」パネルで、「**新規拡張ロケーションのインストール (Install new extension location)**」をクリックします。
8. 「ソース・アーカイブ・ファイル (Source Archive File)」パネルで「参照」をクリックし、**zWBI.wct** ファイルにナビゲートして「開く」をクリックします。

9. 「ソース・アーカイブ・ファイル (Source Archive File)」パネルで、「次へ」をクリックします。インストールの詳細の要約が表示されます。
10. 「拡張ロケーションの要約 (Extension Location Summary)」パネルで「次へ」をクリックします。次のパネルが表示されるまでに少し時間がかかります。
11. 「インストールが正常に完了しました (Install Successful)」パネルで説明を読み、「ロケーション」フィールドに表示されたロケーションを書き留めます。

注: 「ロケーション」フィールド内のテキストは通常かなり長く、すべて読むためには、右方向にスクロールするか、パネルを拡大する必要があります。
12. 「インストールが正常に完了しました (Install Successful)」パネルで、「終了」をクリックします。「製品構成 (Product Configuration)」パネルが開きます。
13. 「拡張のロケーションの追加 (Add an Extension Location)」をクリックします。
14. ステップ 11 で書き留めたロケーションを参照します。
15. 「OK」をクリックします。WebSphere Customization Tool アプリケーションを再始動するように求めるプロンプトが出されたら、「はい」をクリックし、ツールが再始動するまで待機します。
16. 「よろこそ」をクリックします。
17. 「提供されるツールのリスト」の下の「プロファイル管理ツール」をクリックします。「このツールでは WebSphere Process Server for z/OS のカスタマイズ定義も作成されます」というテキストが表示されます。このテキストは、プラグインが正常にインストールされたことを確認するものです。

次のタスク

これで、WebSphere Customization Tool アプリケーションを使用して WebSphere Process Server プロファイルを構成できるようになります。WebSphere Customization Tool の使用について詳しくは、「技術情報 PRS3357」を参照してください。

WebSphere Application Server のインストール

WebSphere Process Server for z/OS をインストールする前に、WebSphere Application Server for z/OS をインストールおよび構成する必要があります。既存の WebSphere Application Server for z/OS Network Deployment セルがある場合は、WebSphere Process Server のインストール先として新しい空の WebSphere Application Server ノードを作成します。WebSphere Process Server をサポートするよう Deployment Manager を構成した後に、そのノードを既存のセルに統合できます。

始める前に

WebSphere Process Server for z/OS の前提条件として、WebSphere Process Server for z/OS プログラム・ディレクトリー で参照されている、以下の WebSphere Application Server フィーチャー・パックもインストールする必要があります。

- XML フィーチャー・パック
- SCA フィーチャー・パック。WebSphere Process Server に必須である、オプションの SDO フィーチャーが含まれています。

このタスクについて

WebSphere Application Server のインストール手順は、選択したインストール・タイプに応じて異なります。

スタンドアロン・サーバー構成

WebSphere Application Server のスタンドアロン・サーバー構成を作成すると、デフォルト・プロファイルが提供され、それを WebSphere Process Server の構成データで拡張できます。詳しくは、WebSphere Application Server インフォメーション・センターの『スタンドアロン・アプリケーション・サーバー・セルの作成』を参照してください。詳しくは、WebSphere Application Server バージョン 7.0 インフォメーション・センターを参照してください。

練習用のスタンドアロン構成の作成方法については、『WebSphere Application Server for z/OS の練習セルの構築』を参照してください。

Network Deployment サーバー構成

WebSphere Application Server のデプロイメント・マネージャーを作成すると、デフォルトの WebSphere Application Server プロファイルが作成され、それを WebSphere Process Server の機能で拡張できます。詳細については、WebSphere Application Server インフォメーション・センターの『Network Deployment セルの作成』を参照してください。詳しくは、WebSphere Application Server バージョン 7.0 インフォメーション・センターを参照してください。

次のタスク

次に、WebSphere Process Server の製品コードを WebSphere Application Server の上にインストールします。『インストール・メディアから z/OS への製品コードの読み込み』を参照してください。

インストール・メディアから z/OS への製品コードの読み込み

WebSphere Process Server for z/OS の製品コードは、IBM SystemPac[®] または ServerPac、あるいは IBM Custom-Built Product Delivery Option (CBPDO) のいずれかを使用してインストールされます。

このタスクについて

z/OS 環境では、製品コードのインストール・メディアからシステムへの読み込みは、通常システム・プログラマーが担当します。

- *IBM SystemPac* または *ServerPac* の使用。IBM SystemPac または ServerPac は、ロード可能な製品ライブラリーおよび対応する SMP/E データ・セットで構成されています。SystemPac または ServerPac を使用して製品コードを読み込む方法についての説明は、SystemPac または ServerPac に付属しています。
- *IBM Custom-Built Product Delivery Offering* の使用。CBPDO には、1 つ以上の製品用の SMP/E 相対ファイル、および保守が含まれます。CBPDO を使用して製品コードを読み込む方法の手順は、*WebSphere Process Server* プログラム・ディレクトリーにあります。

次のタスク

インストール・メディアから製品コードを読み込み、製品データ・セットがシステム上で定義されたら、システムでインストール・スクリプト (zWPSInstall.sh) を製品管理者が実行できるようになります。

IBM SystemPac または ServerPac の使用

IBM CustomPac (SystemPac、ServerPac、または ProductPac[®]) は、ディスクへのデータ・セットのロードおよび初期カスタマイズの実行に使用される IBM ダイアログにバンドルされた、一連のプリインストール済み製品データ・セットです。

このタスクについて

一般に、CustomPac オファリングのインストール中には、SMP/E の作業は必要ありません。代わりに、CustomPac サービス・レベルに対応する SMP/E データ・セットが、製品データ・セットと共にディスクにロードされます。その場合も、CustomPac インストール後に、SMP/E を使用して予防および修正サービスをインストールすることができます。

IBM SystemPac または ServerPac を使用する場合、SystemPac または ServerPac に付属する「*ServerPac: Installing your Order*」の指示に従ってください。

SystemPac または ServerPac のインストールについては、「z/OS IBM ServerPac インストール・ダイアログの使用法 (SA88-8647)」を参照してください。

詳しくは、以下の情報源を参照してください。

- http://www.ibm.com/software/webservers/appserv/zos_os390/support の Web サイト
- PSP バケット (<http://www14.software.ibm.com/webapp/set2/psp/srchBroker>)
- <http://www-306.ibm.com/software/support/> の IBM ソフトウェア・サポート・センターの Web サイト

IBM SystemPac または ServerPac からインストールするときは、以下の要件が満たされていることを確認する必要があります。

手順

- 保守の目的で製品ライブラリーのコピーを少なくとも 2 つ保持および保守できるような製品データ・セット命名規則を選択してください。詳しくは、18 ページの『製品データ・セットについての計画』を参照してください。
- 駆動システムからインストールを実行する場合、ターゲット・システムの保守レベルが WebSphere Process Server for z/OS の要件を満たしていることを確認してください。
- インストールが完了したら、製品データ・セットが z/OS ターゲット・システムで使用可能であり、WebSphere Process Server のインストール・ファイル・システムが各ターゲット・システムの /usr/lpp/zWPS/V7R0 または選択した類似のマウント・ポイントにマウントされていることを確認してください。

IBM Custom-Build Product Delivery Option の使用

IBM Custom-Build Product Delivery Option (CBPDO) は、累積サービスと共にバンドルされている 1 つ以上の IBM ソフトウェア製品のプロダクト・テープのセットです。SMP/E を使用してシステム上に製品およびサービスをインストールします。

このタスクについて

CBPDO を使用する場合、注文品に付属する「*WebSphere Process Server for z/OS: Program Directory*」(GI11-2880-00) の指示に従ってください。

PDF 形式のプログラム・ディレクトリーは、WebSphere Process Server for z/OS のダウンロード・ページ (<http://www-306.ibm.com/software/integration/wps/library/infocenter/>) からダウンロードできます。

プログラム・ディレクトリーには、インストール・メディアから z/OS システムに製品コードを読み込む方法の説明が含まれています。また、プログラム・ディレクトリーには、WebSphere Process Server インストール・スクリプト (zWPSInstall.sh) を実行できるようにするために、(HFS または zFS を使用した) インストール・ファイル・システムをセットアップする方法の手順も含まれています。

詳しくは、以下の情報源を参照してください。

- WebSphere Application Server for z/OS サーバー製品のサポート Web サイト (http://www.ibm.com/software/webservers/appserv/zos_os390/support)
- PSP バケット (<http://www14.software.ibm.com/webapp/set2/psp/srchBroker>)
- <http://www-306.ibm.com/software/support/> の IBM Software Support Center Web サイト

IBM SystemPac または ServerPac からインストールするときは、以下の要件が満たされていることを確認する必要があります。

手順

- 保守の目的で製品ライブラリーのコピーを少なくとも 2 つ保持および保守できるような製品データ・セット命名規則を選択してください。詳しくは、『製品データ・セットについての計画』を参照してください。
- 駆動システムからインストールを実行する場合、ターゲット・システムの保守レベルが WebSphere Process Server for z/OS の要件を満たしていることを確認してください。
- インストールが完了したら、製品データ・セットが z/OS ターゲット・システムで使用可能であり、WebSphere Process Server のインストール・ファイル・システムが各ターゲット・システムの /usr/lpp/zWPS/V7R0 または選択した類似のマウント・ポイントにマウントされていることを確認してください。

製品データ・セットについての計画

WebSphere Process Server for z/OS の製品コードは、区分データ・セット (製品データ・セットが格納されている) および階層ファイル・システム・ディレクトリー (製品ディレクトリーおよびそのサブディレクトリーが格納されている) 内にあります。製品データ・セットのデフォルトの高位修飾子は BPZ です。

製品データ・セットの内容

WebSphere Process Server の製品データ・セットは、ターゲット・データ・セット (製品のカスタマイズおよび実行時に使用) と配布データ・セット (必要に応じて、保守の「バックオフ」に使用) に分けられます。以下の説明では、特定の一連の WebSphere Process Server for z/OS 製品データ・セットに対するデータ・セット名の高位修飾子を表すために *wps_hlq* を使用しています。

WebSphere Process Server には、以下のターゲット・データ・セットがあります。

wps_hlq.SBPZEXEC
REXX 実行スクリプト

wps_hlq.SBPJCL
インストール・ジョブの JCL

WebSphere Process Server には、以下の配布データ・セットがあります。

wps_hlq.ABPZANT
インストール・ファイル・システムのファイル (ASCII)

wps_hlq.ABPZEBCD
インストール・ファイル・システムのファイル (EBCDIC)

wps_hlq.ABPZEXEC
REXX 実行可能ファイル

wps_hlq.ABPJCL
インストール・ジョブの JCL

ターゲット・データ・セットと配布データ・セットそれぞれの割り振り情報については、「WebSphere Process Server for z/OS Program Directory」(GI11-2880-00) を参照してください。この情報の更新は、WebSphere Process Server for z/OS の各リリースの Preventive Service Planning (PSP) バケットに含まれています。Preventive Service Planning (PSP) バケットは、<http://www14.software.ibm.com/webapp/set2/psp/srchBroker> にあります。

製品データ・セットの命名規則

製品が正常に機能するためには、WebSphere Process Server for z/OS の特定のデータ・セットに、同じデータ・セット名高位修飾子が必要です。すべての製品データ・セットが同じ高位修飾子を持っていると、製品の保守やマイグレーションが容易になります。

また、保守が行われている間に WebSphere Process Server for z/OS の実行を継続するには、製品データ・セットのコピーが少なくとも 2 つ必要です。1 つは、実行中のアプリケーションの稼働環境用で、もう 1 つは、サービスを適用先です。

WebSphere Process Server for z/OS の個々のリリースおよび保守レベルに、中間レベルの修飾子を選択してください。この中間レベルの修飾子により、テストと実動の区別を非常に単純に示すことができます。例えば、「BPZ.V6PROD.*」や「BPZ.V6TEST.*」と指定します。また、中間レベルの修飾子には特定のサービス・レベル情報を含めることができます。例えば、「WPS.W610FP1.*」、または「WPS.W610FP2.*」と指定します。

製品データ・セット名は多くの場所で指定する必要があるため、混乱を避けるために、保守の目的を達成するデータ・セット命名スキームの中で最も単純なものを使用してください。

WebSphere Process Server のファイル・システム・ディレクトリー

WebSphere Process Server for z/OS 製品およびインストール・コードは、z/OS 区分データ・セット (製品データ・セット) および z/OS USS ファイル・システム・ディレクトリー (HFS または zFS ファイル・システム) にあります。

WebSphere Process Server のインストール・ファイル・システム

WebSphere Process Server for z/OS 製品のすべてのファイルは、インストール・ファイル・システムのディレクトリーとそのサブディレクトリー内にあります。WebSphere Process Server のインストール・ファイル・システムは、インストール・メディアから製品コードをダウンロードするときに定義されます (インストール・プロセスのフェーズ 1)。通常、インストール・ファイル・システムは、読み取り専用でマウントされます。

WebSphere Process Server インストール・ファイル・システムのデフォルトの場所は `/usr/lpp/zWPS/V7R0` です。

WebSphere Process Server インストール・ファイル・システムの場所は、WebSphere Application Server インストール・ファイル・システムの場所とは異なります。WebSphere Application Server インストール・ファイル・システムのデフォルトの場所は、`/usr/lpp/zWebSphere/V7R0` です。

製品ディレクトリーおよびそのすべてのサブディレクトリーは、同じ階層ファイル・システム (HFS) または zSeries ファイル・システム (zFS) データ・セット内にある必要があります。インストール・ジョブおよびプログラム・ディレクトリーでは、z/OS ルートまたはバージョンのデータ・セットとは別に、WebSphere Process Server で使用されるデータ・セットが割り振られることが想定されています。zFS ファイル・システムまたは HFS ファイル・システムを作成するサンプル・ジョブが、`hlq.SBPZJCL` 内のメンバー `BPZALZFS` および `BPZALHFS` として提供されています。この 2 つのジョブのうちの 1 つは、インストール時にシステム・プログラマーによって実行されます。

構成ファイル・システム

各 WebSphere Process Server for z/OS アプリケーション処理環境 (スタンドアロン・アプリケーション・サーバー・ノードまたは Network Deployment セル) では、1 つ以上の WebSphere Application Server 構成ディレクトリーに構成ファイルが含まれています。これらの構成ディレクトリーには、製品ディレクトリー内のファイルへのシンボリック・リンクが含まれます。

構成ディレクトリーのデフォルトの場所は `/WebSphere/VR70` です。このため、デプロイメント・マネージャー構成ファイルの場所は `/WebSphere/VR70/DeploymentManager/config/bin` になります。

関連情報

 [z/OS basic skills information center](#)

WebSphere Process Server のインストール

インストール・スクリプトは、製品を使用可能にする WebSphere Process Server for z/OS 定義を作成します。

関連資料

 [zWPSInstall.sh スクリプト](#)

zWPSInstall.sh スクリプトは、WebSphere Application Server プロファイルを変更して WebSphere Process Server をインストールまたはアンインストールするために使用します。デプロイメント・マネージャーを含め、構成内の各ノードでスクリプトを実行します。

 [WebSphere Process Server をデプロイメント・マネージャー・ノードへインストールするための JCL](#)

 [WebSphere Process Server を空のノードへインストールするための JCL](#)

zPMT を使用した構成定義の作成

プロファイル管理ツールを使用して、WebSphere Process Server を構成することができます。プロファイル管理ツールにアクセスするには、WebSphere Customization Tool アプリケーションを実行します。このトピックでは、zPMT ツールを使用して構成定義を作成する方法について説明します。WebSphere Process Server を構成するには、製品に同梱されているサンプル応答ファイルをカスタマイズするという方法もあります。

始める前に

- WebSphere Application Server をインストールして構成し、WebSphere Process Server for z/OS 構成のベースとして使用します。
- WebSphere Customization Tool をインストールし、選択したカスタマイズ・タスク用に指示されているいずれかの計画を実行します。詳しくは、『WebSphere Customization Tool のインストール』を参照してください。

このタスクについて

- zPMT ツールの概念情報については、『zPMT ツール』を参照してください。
- WebSphere Process Server for z/OS の構成に使用する応答ファイルのキーワードについて詳しくは、『45 ページの『応答ファイルの値』』を参照してください。

手順

1. WebSphere Customization Tool を起動します。「スタート」>「すべてのプログラム」>「IBM WebSphere」>「WebSphere Customization Tools V7.0」>「WebSphere Customization Tools」を選択します。「ようこそ」画面が表示されます。この画面から、zPMT または zMMT を選択します。「zPMT」を選択します。

- zPMT ツールが起動すると、画面上部のツールバーの「プロファイル管理ツール」ボタンに表示されます。zPMT ツールは、カスタマイズ・ロケーションとカスタマイズ定義という 2 つのセクションから構成されています。
 - カスタマイズ・ロケーションは、カスタマイズ・ツールと zPMT によって作成されるすべてのサブディレクトリーとファイルから構成されるフォルダー構造です。カスタマイズ・ロケーション・フォルダーとしては、ローカル・ドライブまたはネットワーク・ドライブを指定することができます。その際、オペレーティング・システムからそのドライブを認識できて、アクセスできなければなりません。異なる構成ごとに異なるロケーションを作成することができ、1 つのワークスペースに複数の構成を定義することができます。
 - カスタマイズ定義は、ワークステーション上の一連のファイルで、これらのファイルを z/OS にアップロードして、WebSphere Process Server for z/OS を構成するために実行するジョブを作成します。
2. 既存のカスタマイズ・ロケーションを開きます。存在しない場合は、新しく作成します。どちらの場合も、ファイル・パスを指定する必要があります。
 - 以前に作成したロケーションを作業セットに追加できます。これは、WebSphere Customization Tool のインストール全体でカスタマイズ・ロケーションを共有する場合に便利です。この場合、開発部門から構成を変更する部門に構成を送信することができます。また、以前に作成したロケーションを作業セットに追加すると、作業セットをバックアップ目的として使用することもできます。例えば、構成を変更する前にバックアップを作成することができます。さらに、構成を災害時回復システムにバックアップすることにより、何らかの理由でサーバーがダウンした場合に備えることができます。バックアップを準備しておくことにより、構成を再作成するのに必要な時間を大幅に短縮することができます。
 - カスタマイズ・ロケーションのバージョン番号は、ターゲットとなるベースの WebSphere Application Serverを表しています。
 - バージョン 7.0 でカスタマイズ・ロケーションを作成して WebSphere Process Server for z/OS 用に拡張する必要があります。
 3. 「拡張」をクリックします。「環境の選択」パネルが表示されます。
 4. 目的の環境を選択します。「管理」、「アプリケーション・サーバー」、または「管理対象 (カスタム) ノード」を選択できます。選択項目を強調表示し、「次へ」をクリックします。「拡張の選択」パネルが表示されます。
 5. **WebSphere Process Server** オプションを強調表示し、「次へ」をクリックします。間もなくすると、「カスタマイズ定義名」パネルが表示されます。
 6. デフォルトを受け入れることも、zPMT カスタマイズ定義の名前を入力することもできます。名前は 8 文字より多い文字を使用できますが、スペース文字を含めることはできません。異なる定義を識別して、それぞれの定義を区別できるように名前にする必要があります。名前は、z/OS システムでは使用されず、zPMT ツールによってのみ使用されます。カスタマイズ定義名は、カスタマイズ・ロケーション内で固有でなければなりません。
 7. このカスタマイズ定義を作成するために再使用したいパラメーターおよび値が含まれる既存の応答ファイルがある場合には、オプションで、そのファイルのパスを「応答ファイル・パス名」フィールドに指定することができます。この

フィールドはオプションです。このフィールドに既存の zPMT または構成応答ファイルを指定すると、それに含まれる値が zPMT によって読み取られて、後続の構成パネルに設定されます。

8. 「次へ」をクリックします。「ターゲット・データ・セット」パネルが表示されます。ターゲット・データ・セットの高位修飾子を入力します。この値は、生成されたジョブ、応答ファイル、および命令をアップロードするためのターゲット・データ・セットとして使用されるだけでなく、実行する JCL のターゲット・ロケーションを指定して生成されたジョブをカスタマイズするためにも使用されます。
9. カスタマイズ・パネルでの作業を続けます。それぞれのパネルで WebSphere Process Server for z/OS システムの作成に使用するパラメーターの値を指定し、「次へ」をクリックして次のパネルに進みます。
10. 必要なパネルをすべて入力し終わると、「カスタマイズの要約」パネルが表示されます。「拡張」をクリックしてカスタマイズされたジョブを生成し、定義を終了します。「カスタマイズ定義の要約」に、正常に作成されたことを示すメッセージが表示されます。「終了」をクリックしてプロファイル管理ツールに戻ると、「カスタマイズの定義」ビューで新規カスタマイズ定義が強調表示されます。「カスタマイズの要約」ビューには、強調表示された定義に関する基本情報(名前、タイプ、指示ファイルおよび応答ファイルへのパスなど)が表示されます。「カスタマイズの指示」ビューには、システム上でジョブを実行して必要なタイプの構成を作成するために必要な情報が表示されます。最後に、「カスタマイズの応答ファイル」ビューには、カスタマイズ定義を記述するために WCT によって生成されたパラメーターの名前と値のペアが表示されます。
11. 「カスタマイズの定義」タブを選択し、必要なカスタマイズ定義を強調表示し、「カスタマイズの定義」ウィンドウで「プロセス」をクリックし、「ターゲット z/OS システムへアップロード」を選択して、定義を z/OS システムにアップロードします。
12. z/OS へのアクセスに使用するホスト、ID、およびパスワードを入力します。ファイルの保管先フォルダーは使用不可に設定されています。これは、このフォルダーの値と、構成内で指定されているターゲット・データ・セットの値を必ず同じにする必要があるためです。したがって、このフォルダーの場所を編集する場合は、カスタマイズ定義を再生成する必要があります。
13. ターゲット・データ・セットは z/OS システムに既に存在する場合があります。あるいは、プロファイル管理ツールを使用して割り振ることができます。
14. カスタマイズされたジョブおよび拡張応答ファイルが、z/OS システムにアップロードされます。拡張応答ファイルの名前は、BPZRSP x の名前形式に従っています。ここで、 x は以下のように値が異なります。
 - スタンドアロン・アプリケーション・サーバー・ノードの場合は **A**
 - デプロイメント・マネージャー・ノードの場合は **D**
 - 管理対象ノードの場合は **N**

アップロードした応答ファイルを使用してシステムを構成するか、生成したインストール・ジョブと構成ジョブを使用して WebSphere Process Server for z/OS の構成を作成します。

タスクの結果

以上で、WebSphere Process Server for z/OS のカスタマイズは完了です。zPMT ツールを使用して zPMT 応答ファイルを作成し、生成されたジョブと拡張応答ファイルを z/OS にアップロードしました。

zPMT で生成された JCL を使用したインストール・ジョブの実行

インストールを行うには、zPMT で生成されたインストール・ジョブをサブミットするか、または zWPSInstall.sh コマンドを実行します。このトピックでは、インストール・ジョブの実行について説明します。

始める前に

あらかじめ、zPMT ツールを使用して構成定義を作成し、これを処理して、生成された JCL を z/OS システムに転送しておく必要があります。

このタスクについて

構成済みの WebSphere Process Server プロファイルについては、zPMT ツールで「カスタマイズの指示」タブを参照してください。これらの指示に、サブミットするアクションとジョブのシーケンスに関する詳細情報が示されます。

インストール・ジョブは BPZINST と呼ばれ、インストール実行可能プログラムを実行します。前提条件のアクションを実行したら、このジョブをサブミットしてインストールを実行してください。インストール・ジョブで実行するコマンドについて詳しくは、zWPSInstall.sh スクリプトを参照してください。

インストール・ジョブが正常に実行されたら、拡張ジョブを実行する前に、必要に応じて 59 ページの『第 3 章 WebSphere Process Serverの構成』に進んでください。これで、DbDesignGenerator.sh ツールを使用して DB2 データベース構成を設計し、データベース設計ファイルを作成できるようになりました。データベース設計ファイルを拡張プロセスに入力すると、WebSphere Process Server のリソース定義にデータベース構成値を指定することができます。詳しくは、60 ページの『DB2 データベース・オブジェクトの設計』を参照してください。

USS からのインストール・スクリプトの実行

インストール・スクリプト zWPSInstall.sh を実行すると、WebSphere Application Server for z/OS サーバーで WebSphere Process Server for z/OS を実行するために必要なコード定義が作成されます。Network Deployment 構成を作成する場合は、Network Deployment セルに属する各ノードに対して、インストール・スクリプトを実行する必要があります。JCL スクリプトを使用して WebSphere Process Server 製品定義をインストールするか、または zWPSInstall.sh スクリプトを手動で実行することができます。WebSphere Process Server WebSphere Enterprise Service Bus をスタンドアロン・サーバーへインストールするための JCL、『WebSphere Process Server を空のノードにインストールするための JCL』および『WebSphere Process Server をデプロイメント・マネージャー・ノードにインストールするための JCL』を参照してください。

始める前に

サーバーをインストールする前に、以下のステップを実行する必要があります。

1. ノードで WebSphere Application Server for z/OS をインストールしてカスタマイズします。アプリケーション・サーバーが停止していることを確認します。詳しくは、『サーバーの始動と停止』を参照してください。
2. サーバーのファイル・システムをバックアップし、必要になった場合にリストアできるようにします。詳しくは、『WebSphere Application Server for z/OS システムのバックアップ』を参照してください。
3. インストール・メディアから z/OS システムに製品コードを読み込みます。
4. インストール・スクリプトを実行できるように、システムの管理者権限を持っていることを確認します。
5. TSO OMVS からインストール・スクリプトを実行する場合は、TSO ユーザー ID の領域サイズが Java™ の実行に十分であることを確認してください (通常は 150 M)。
6. オプション: **オプション:** 製品構成スクリプトを最後まで実行させるためには、OMVS の時間制限の延長が必要な場合があります。OMVS の時間制限を延長して、セッションがタイムアウトにならないようにするには、以下のシステム・コマンドを入力します。

```
SETOMVS MAXCPUIME=86400
```

インストール・シェル・スクリプトについて詳しくは、『zSMPInstall.sh スクリプト』を参照してください。

手順

1. OMVS で管理者ユーザー ID に切り替えます。以下に例を示します。

```
su wsadmin
```

2. 次のように入力して、インストール・スクリプトを実行するディレクトリーに切り替えます。以下に例を示します。

```
cd /usr/lpp/zWPS/V7R0/zos.config/bin
```

3. 次のように入力して、インストール・スクリプトを実行できるように現行ディレクトリーを PATH に追加します。

```
export PATH=.:$PATH
```

4. コマンド・プロンプトから、インストール・スクリプトを実行します。以下に例を示します。

```
zWPSInstall.sh -smproot /usr/lpp/zWPS/V7R0 -runtime  
/usr/lpp/zWebSphere/V7R0/AppServer -install
```

オプションで、標準出力メッセージをコンソールに表示するのではなく、ファイルにリダイレクトすることもできますが、この説明では、スクリプトの実行中に標準出力メッセージを表示するものと想定しています。

実行中のスクリプトの進行状況が分からない場合は、コンソールの表示を最新表示して、最後のいくつかのメッセージを確認してください。

インストール・スクリプトの実行が完了すると、シェル・プロンプトに戻る前に以下の (または以下と同様の) メッセージが表示されます。

```
CWPIZ0256I: 構成のセットアップが完了しました
CWPIZ0257I: シンボリック・リンクの作成中...
CWPIZ0259I: シンボリック・リンクの作成が完了しました
CWPIZ0260I: インストール後ファイル更新の実行中...
CWPIZ0262I: インストール後更新が完了しました
CWPIZ0263I: 構成マネージャー更新の実行中...
Oct 19, 2007 4:28:16 PM com.ibm.ws390.installer.WPSInstaller
INFO: BBZWI218
Oct 19, 2007 4:38:16 PM com.ibm.ws390.installer.WPSInstaller
WARNING: BBZWI221
CWPIZ0264I: 構成マネージャー更新が完了しました。
```

スクリプトの実行が完了したら、コンソールに表示されるメッセージを確認して、エラー・メッセージが表示されていないかどうかを調べます。

ソフトウェアが正常にインストールされなかった場合は、インストールのトラブルシューティングを参照してインストール上の問題を診断してください。

注: Network Deployment 構成を作成する場合、この時点では、デプロイメント・マネージャーにノードを統合するためのジョブ BBOWMNAN を実行しないでください。ノードの統合は、後続の段階で構成ステップが完了した以降に行います。

タスクの結果

以上で、WebSphere Application Server 構成に、WebSphere Process Server インストール済み環境へのシンボリック・リンクが組み込まれます。WebSphere Application Server 管理コンソールが更新され、これを使用して WebSphere Process Server を管理できるようになります。

次のタスク

インストール・スクリプトの実行が正常に完了したら、次に WebSphere Process Server for z/OS 機能を使用してサーバーを構成できます。詳しくは、59 ページの『WebSphere Process Server for z/OSの構成』を参照してください。

製品保守の適用

WebSphere Process Server for z/OS は、WebSphere Application Server にインストールされて構成されます。したがって、WebSphere Application Server 製品と技術を使用して、WebSphere Process Server for z/OS に製品保守を適用します。

始める前に

製品の Preventive Service Planning (PSP) のアップグレードの情報については、IBM ソフトウェア・サポート・センターにお問い合わせください。WebSphere Process Server for z/OS の PSP のアップグレードの詳細については、「*Program Directory for WebSphere Process Server for z/OS*」を参照してください。プログラム・ディレクトリーには、必要なプログラム一時修正 (PTF) がリストされていますが、IBM ソフトウェア・サポート・センターから最新情報を入手することができます。

このタスクについて

ご使用のシステムに新しいサービス・リリースを適用する場合は、以下に示す手順に従ってください。

手順

製品保守の適用については、WebSphere Application Server for z/OS のインフォメーション・センターにある『製品保守の適用』を参照してください。

次のタスク

WebSphere Application Server for z/OS のホスト・クラスタのアップグレード時に、クライアントに対するサービスを維持することができます。

サービス・レベルの適用または前の受け入れ済みサービス・レベルへの復元

WebSphere Process Server for z/OS は、WebSphere Application Server にインストールされて構成されるため、WebSphere Process Server for z/OS に適用されたサービス・レベルは、WebSphere Application Server 製品を通じて適用されます。サービス・レベルの適用または前の受け入れ済みサービス・レベルへの復元には、WebSphere Application Server の技法を使用します。

このタスクについて

製品データ・セットおよび製品ファイル・システムに適用されるサービスでは、低いサービス・レベルで構成される既存のアプリケーション・サービス提供環境の構成ファイル・システムに対して、対応する変更を加えることが必要な場合があります。これらの保守後またはインストール後の更新のほとんどは、自動的に実行することができます。これは、ポストインストーラーによって実行されます。サービスの適用方法の説明については、WebSphere Application Server for z/OS インフォメーション・センター内の『サービス・レベルの適用または前の受け入れ済みサービス・レベルへの復元』を参照してください。

アップグレード・プロセスについて

WebSphere Process Server for z/OS のアップグレードは、複数のロールにまたがる複数フェーズのプロセスです。

WebSphere Process Server for z/OS 製品をアップグレードするには、次のいずれかの方法を使用します。

- 既存のインストール・ファイル・システムに新しい製品バージョンをオーバーレイする
- アップグレード・スクリプトを実行して、旧バージョンを新規のインストール済み製品バージョンに更新する

シナリオ 1: 既存のインストール・ファイル・システムへの新バージョンのオーバーレイ

このシナリオでは、SMP/E を使用して既存のインストール・ファイル・システム上に製品の新バージョンをロードすることにより、WebSphere Process Server for z/OS をアップグレードします (図 1)。

注: 構成は、その構成ファイル・システムにのみアクセスできるようになっています。構成ファイル・システムのシンボリック・リンクによって、インストール・ファイル・システム内のコードにアクセスできるようになります。

その後、サーバー・コントローラーが始動すると、*applyPTF.sh* スクリプトが実行されます。このスクリプトは、構成ファイル・システムのレベルを、インストール・ファイル・システムのレベルと照らし合わせてチェックします。この 2 つのファイル・システムが同じ保守レベルになっている場合は、サーバーが始動します。構成ファイル・システムのレベルがインストール・ファイル・システムのレベルよりも低い場合、*applyPTF.sh* スクリプトは、インストール・ファイル・システムの保守レベルによって指定される構成ファイル・システムに変更を加え、サーバーを始動します。

注: 構成ファイル・システムのレベルがインストール・ファイル・システムのレベルより高い場合 (例えば、保守がバックオフされる場合)、サーバーは始動できません。

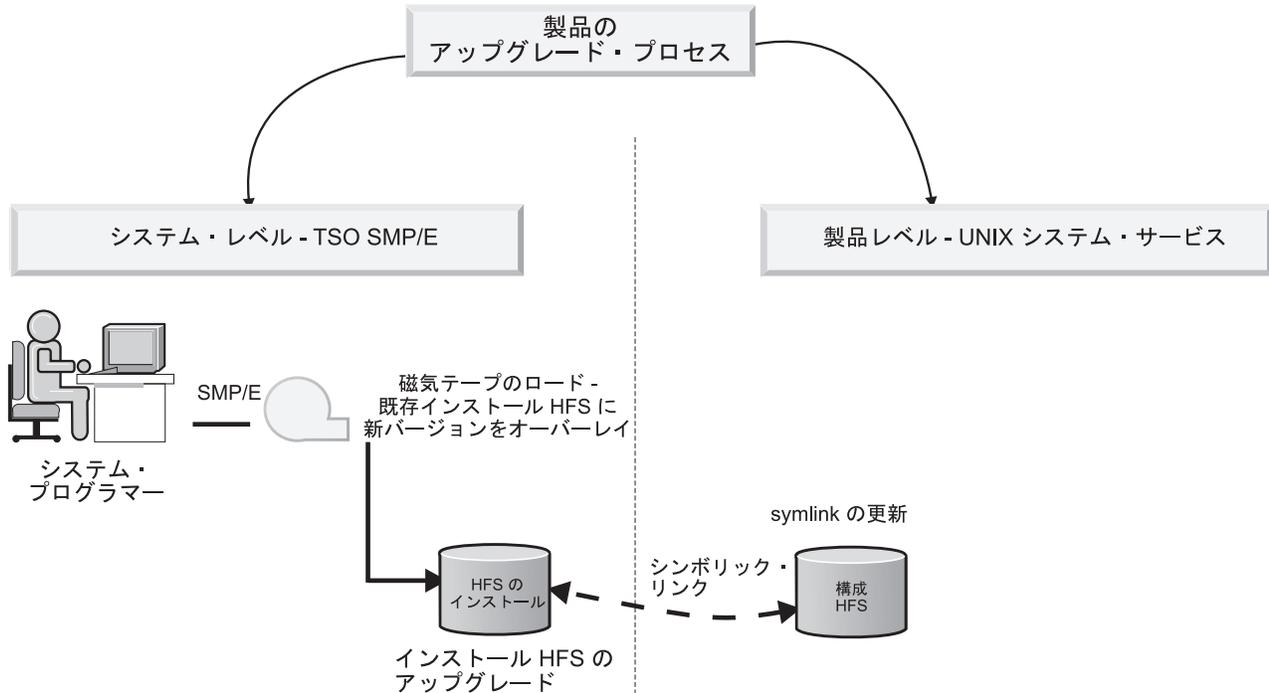


図 1. 既存のインストール・ファイル・システムのオーバーレイによる WebSphere Process Server for z/OS 製品のアップグレード

シナリオ 2: アップグレード・スクリプトの使用

このシナリオでは、SMP/E を使用して、既存のインストール・ファイル・システムとは別に製品の最新バージョンをロードすることにより、WebSphere Process Server for z/OS をアップグレードします (図 2)。

その後、アップグレード対象のアプリケーション・サーバーごとにアップグレード・スクリプトを実行できます。アップグレード・プロセスでは、サービス・レベルのシンボリック・リンクが、アプリケーション・サーバーの既存のインストール・ファイル・システムから新規のインストール・ファイル・システムに変更されます。

その後、サーバー・コントローラーが始動すると、*applyPTF.sh* スクリプトが実行されます。このスクリプトは、構成ファイル・システムのレベルを、インストール・ファイル・システムのレベルと照らし合わせてチェックします。この場合、構成ファイル・システムのレベルはインストール・ファイル・システムのレベルよりも低く、*applyPTF.sh* スクリプトは、インストール・ファイル・システムの保守レベルによって指定される構成ファイル・システムに変更を加え、サーバーを始動します。

図 2 では、WebSphere Process Server for z/OS の最新バージョンはアプリケーション・サーバー A で稼働するためのインストールおよび構成が完了しています。アプリケーション・サーバー B をアップグレードする場合、システム・プログラマーはアップグレード・スクリプトを実行できます。このスクリプトによって、アプリケーション・サーバー B の構成ファイル・システムが更新されて、サービス・レベル・シンボリック・リンクがアプリケーション・サーバー A に関連する新規インストール・ファイル・システムを指すようになります。このインストール・アップグレードは、*applyPTF.sh* スクリプトが実行されると完了します。

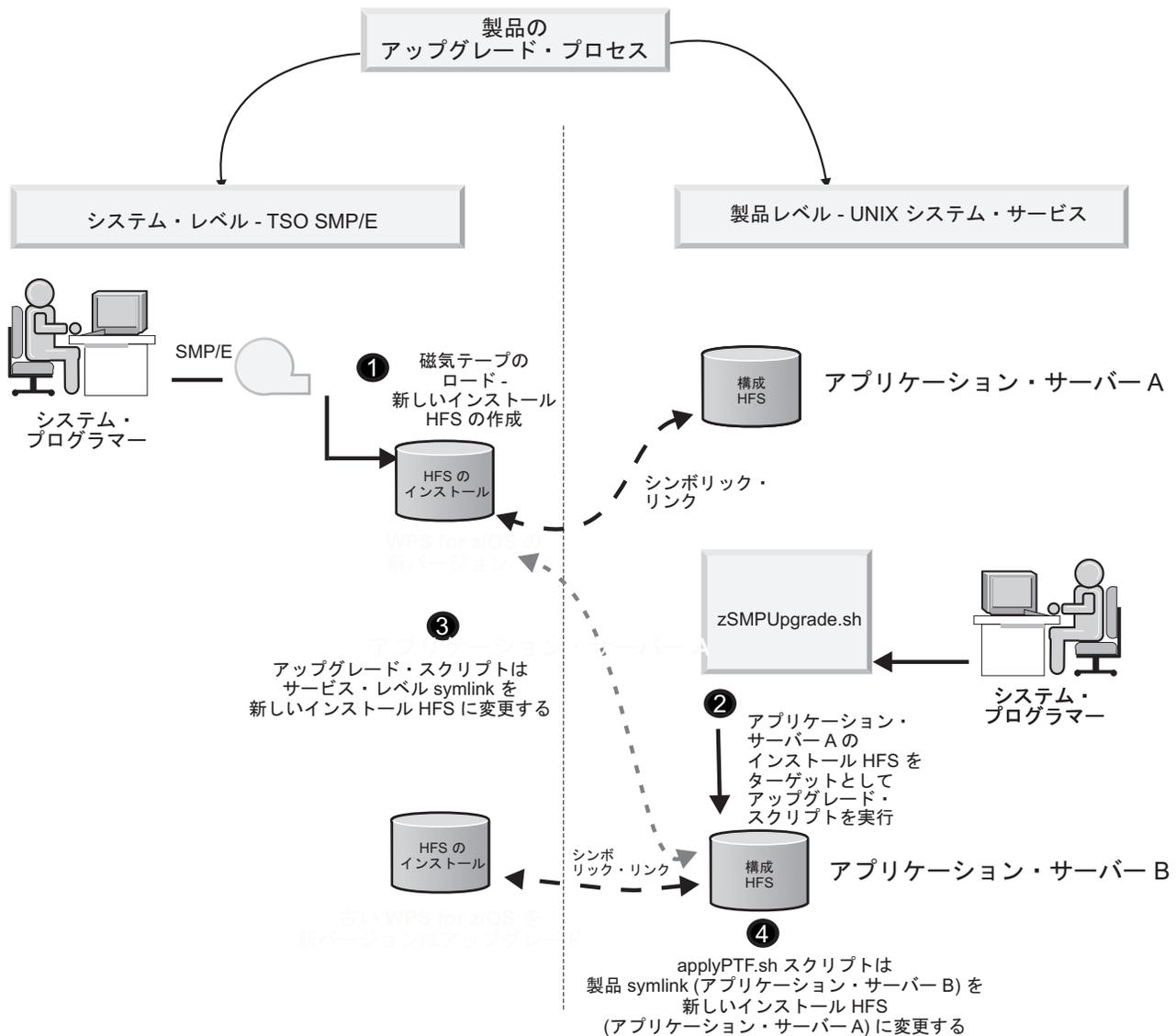


図 2. アップグレード・スクリプトを使用した WebSphere Process Server for z/OS 製品のアップグレード

中間シンボリック・リンクによるアップグレード・スクリプトの使用

この例は、図 2 のシナリオと類似していますが、インストール・ファイル・システムと構成ファイル・システムとの間の直接のシンボリック・リンクではなく、中間シンボリック・リンクが使用されます。中間シンボリック・リンクは、インストール・ファイル・システムを指す標準のシンボリック・リンクであり、構成ファイル・システムはこの中間シンボリック・リンクを指します。ノードのサービス・レベルを変更すると、1 つの中間シンボリック・リンクが変更されます。

アップグレード対象のアプリケーション・サーバーごとにアップグレード・スクリプトを実行できます。このスクリプトでは、新バージョンのインストール済みファイル・システムを使用して、更新対象の構成ファイル・システムのシンボリック・リンクを更新します。構成ファイル・システムは、追加の間接的な層である中間シンボリック・リンクを指します。

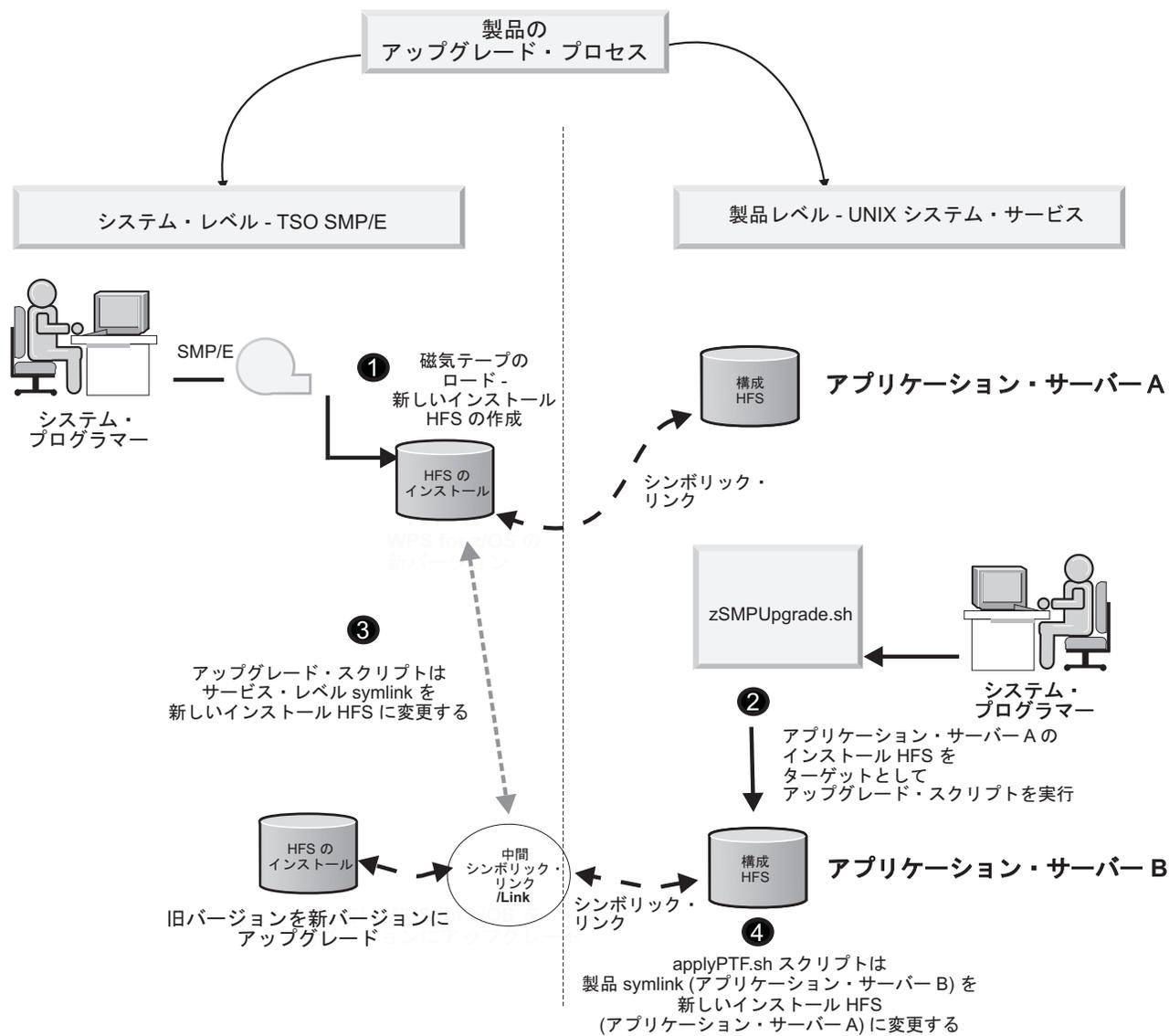


図 3. アップグレード・スクリプトを使用した WebSphere Process Server for z/OS 製品のアップグレード

JNDILookup Web Service アプリケーションのインストール

WebSphere Process Server は、Java 以外のクライアントによって解釈できない JMS 管理対象オブジェクトを保守します。Java 以外のクライアントに管理対象オブジェクトへのアクセスを許可するために、WebSphere Process Server は JNDILookup Web Service を提供します。この Web サービスは、プロキシとして動作して、非 Java クライアントが JMS 管理対象オブジェクトを取得できるようにします。

始める前に

JNDILookup Web Service アプリケーションをインストールする前に、システム上に実行中の WebSphere Process Server のインストールがあることを確認する必要があります。

このタスクについて

WebSphere Process Server インストール済み環境に対して非 Java クライアントからのアクセスがある場合は、JNDILookup Web Service をインストールする必要があります。このアプリケーションは、以下に説明する管理コンソールを使用してインストールできます。

重要: 以下のステップの実行を開始した後に、アプリケーションをインストールしないことを決めた場合は、「**キャンセル (Cancel)**」をクリックして終了してください。別の管理コンソール・ページに移動する場合は、まず、アプリケーション・インストール・ページで「**キャンセル (Cancel)**」をクリックしてください。

手順

1. コンソールのナビゲーション・ツリーで、「**アプリケーション**」 → 「**新規アプリケーション (New Application)**」をクリックします。

「アプリケーション・インストールの準備 (Preparing for application installation)」の 2 つのページのうち最初のページが表示されます。

2. 最初の「アプリケーション・インストールの準備 (Preparing for the application installation)」ページで、新規アプリケーションへのパスを指定します。

- a. `install_root/installableApps` ディレクトリーにブラウズし、`SIBXJndiLookupEAR.ear` を選択します。

- b. 「次へ」をクリックします。

3. 2 番目の「アプリケーション・インストールの準備 (Preparing for application installation)」ページで、次のことを行います。

- a. デフォルトのバインディングとマッピングを生成するかどうかを選択します。

デフォルトのバインディングを使用すると、アプリケーション内の未完成のバインディングにデフォルト値が入力されます。既存のバインディングは変更されません。デフォルト・バインディングの生成に使用されるデフォルト値をカスタマイズすることができます。

- b. 「次へ」をクリックします。

「新規アプリケーションのインストール (Install New Application)」ページが表示されます。

4. 「ステップ 1: インストール・オプションの選択 (Step 1: Select installation options)」パネルで、「**Web サービスのデプロイ (Deploy Web services)**」を選択します。

5. 「**ステップ 5: 要約 (Step 5: Summary)**」をクリックし、「**要約 (Summary)**」パネルに進みます。

6. 「**要約 (Summary)**」パネルで、「**終了**」をクリックします。

次のタスク

アプリケーション・インストールの進行メッセージを検査します。アプリケーションが正常にインストールされた場合は、変更内容をマスター構成に保存します。コンソール・ナビゲーション・ツリーで「**アプリケーション**」 → 「**エンタープライ**

ズ・アプリケーション」の順にクリックしてアクセスされる「エンタープライズ・アプリケーション」ページ上の、デプロイ済みアプリケーションのリストに **SIBXJndiLookup** が表示されるようになります。

「エンタープライズ・アプリケーション」ページからこのアプリケーションを開始するには、「**SIBXJndiLookup**」を選択して、「**開始**」をクリックします。

Message Service クライアントのインストール

C、C++、または .NET アプリケーションが WebSphere Process Server と対話できるようにするには、メッセージ・サービス・クライアントを使用します。

このタスクについて

メッセージ・サービス・クライアントをインストールするために実行すべき手順は、クライアントのタイプおよび使用するインストールのタイプによって異なります。これらの手順については、当該タイプのクライアントに関する資料で説明されています。

手順

- Message Service Client for .NET のインストール
- Message Service Client for C/C++ のインストール

アンインストール

『アンインストール』のセクションでは、WebSphere Process Server for z/OS のアンインストール方法について説明します。

アンインストールの概要

WebSphere Process Server for z/OS は、コマンド行からアンインストール・パラメーターを指定してインストール・スクリプト `zWPSInstall.sh` を実行することによってアンインストールできます。

アンインストール・スクリプトを実行する前に、最新の技術情報を確認してください。

`-uninstall` コマンド引数を指定して `zWPSInstall.sh` コマンドを実行すると、WebSphere 環境は、インストール前のレベルに復元されます。

注: Common Event Infrastructure および Business Process Choreographer コンポーネントは、別々にアンインストールする必要があります。詳しくは適切なヘルプ・トピックを参照してください。

アンインストール・プロセスでは、以下のアクションが実行されます。

- 構成マネージャーのスクリプト・アクションが実行されることにより、WebSphere Process Server フィーチャーが使用不可になります。管理コンソールのプラグイン拡張機能すべてが除去されます。
- `WSPROFILE` スクリプト・アクションにより、デフォルトの WebSphere Application Server プロファイルの拡張プロセスが逆に戻されます。
- インストール後ファイルが削除され、コードの基本許可が除去されます。

WebSphere Process Server for z/OS のアンインストール

-uninstall キーワードを指定してシェル・スクリプト `zWPSInstall.sh` を実行することによって、WebSphere Process Server for z/OS をアンインストールします。この指示では、OS/390® UNIX コマンド・シェルから、uninstall キーワードを付けてインストール・スクリプトを実行する方法について説明します。別の方法としては、Telnet を使用して、プロンプトからスクリプトとキーワードを実行することもできます。ここでは、TSO を使用していると想定しています。Telnet を使用する場合は、相当するコマンドを使用してください。また、ここに示す説明では、CTRL キーを使用したデフォルトのキー・マッピングで入力すると想定しています。この構成を変更した場合は、マップしたキーを使用して情報を TSO に入力してください。

始める前に

注: 部分アンインストールまたはインクリメンタル・アンインストールはサポートされていません。

アンインストール・スクリプトを実行する前に、WebSphere Process Server for z/OS のアンインストールに関する最新の技術情報をお読みください。

WebSphere Process Server for z/OS インストール・プロセスでは、z/OS UNIX System Services の実用的な知識があることを前提としています。さらに詳細な情報が必要な場合には、以下の資料を参照してください。

- z/OS V1.11 UNIX システム・サービス ユーザーズ・ガイド
- z/OS V1.11 UNIX システム・サービス コマンド解説書

手順

1. OS/390® UNIX® コマンド・シェルにアクセスします。Telnet で USS に直接ログインするか、TSO から TSO コマンド `OMVS` を ISPF コマンド・オプション 6 で入力するか、TSO `OMVS` をそれ以外の ISPF パネルから入力します。UNIX シェルに入ると、システムが入力を受け入れる準備ができたことがコマンド・プロンプト (通常はドル (\$) またはポンド (#) 記号) で示されます。
2. 次のように入力して、管理者ユーザー ID に切り替えます。

```
su wsadmin
```

3. 次のようにして、製品インストール・ファイル・システム・ディレクトリー内の `zos.config/bin` に移動します。

```
cd /usr/zWPS/V7R0/zos.config/bin
```

4. 現行ディレクトリーをパスに追加します。

```
export PATH=.:$PATH
```

5. コマンド・プロンプトから、アンインストール・コマンドを指定してインストール・スクリプトを実行します。以下に例を示します。

```
zWPSInstall.sh -smproot /wps/pathofWPScode/zWPS/V7R0 -runtime  
/WebSphere/V7R0/AppServer -uninstall -response  
/yourdrivename/yourfoldername/responsefilename.rsp
```

システム構成に応じて、応答ファイルの絶対パス名を入力します。前述のパス・ファイル名は、アンインストーラーがデフォルト応答ファイルを使用しているこ

とを前提とします。応答ファイルがカスタマイズされた場合、パス名は、カスタマイズされたファイルの絶対パスを表す必要があります。

6. 「アンインストールのための構成マネージャーの実行が完了しました」というメッセージが表示され、コマンド・プロンプトに戻るまで待ちます。

タスクの結果

-uninstall パラメーターを指定してインストール・スクリプトを実行すると、次のアクションが実行されます。

- 構成マネージャーがそのスクリプト・アクションを実行するときに、WebSphere Process Server for z/OS 製品の機能は使用不可になります。WebSphere Process Server for z/OS 製品の機能が使用不可になると、管理コンソールのプラグイン拡張機能が除去されます。
- WSPROFILE スクリプトにより、デフォルト WebSphere Application Server プロファイルでの拡張プロセスが逆に戻され、これによりすべての WebSphere Process Server 機能が除去されます。ユーザーは、拡張されたデフォルト・プロファイルが削除される（基盤となる WebSphere Application Server for z/OS または WebSphere Application Server Network Deployment for z/OS がアンインストールされる場合）こと、またはほかの拡張プロファイルを使用できなくなることを警告されます。
- インストール後のファイルが削除されます。
- コードの基本許可が除去されます。

共用の共通コンポーネントが他のアプリケーションによって使用されている場合は、コマンド行プロンプトにより、製品をアンインストールすると他のアプリケーションが正しく機能しなくなる可能性があることが警告されます。

アンインストール・コマンドの実行に失敗した場合は、関連付けられたログ・ファイルとトレース・ファイルを確認してください。

• 標準出力メッセージ

標準出力メッセージは画面に直接表示されます。コマンド行の末尾でリダイレクト記号とファイル名を使用して、これらのメッセージをファイルにリダイレクトすることもできます。例えば、インストール・コマンドの末尾に構文 `>run.log` を追加すると、標準出力メッセージが現在の作業ディレクトリーの **run.log** という名前のファイルにリダイレクトされます。標準出力メッセージが以下のように表示されます。

```
コマンド引数の構文解析中...
引数の構文解析が完了しました
構成のセットアップ中...
runtimeRootDirName: /WebSphere/V7R0/AppServer
WAS_HOME: /WebSphere/V7R0/AppServer
WBI_HOME: /WebSphere/V7R0/AppServer
アンインストールのために構成マネージャーを実行中...
アンインストールのための構成マネージャーの実行が完了しました
プロファイルの拡大解除中...
プロファイルの拡大解除が完了しました
```

• ログ・ファイル

ログ・メッセージは、ランタイム・ディレクトリー内の `zWPSInstall.log` ファイルに書き込まれます。このファイルのデフォルトの場所は `/WebSphere/V7R0/AppServer/logs/wbi/zWPSInstall.log` です。

• トレース・ファイル

ランタイム・ディレクトリー内の zWPSInstall.trace (ASCII) ファイルを確認します。このファイルのデフォルトの場所は /WebSphere/V7R0/AppServer/logs/wbi/zWPSInstall.trace です。

「E」 サフィックスが付いたエラー・メッセージがないようにする必要があります。

以下のトラブルシューティング・アクションも実行できます。

- **構成マネージャーの更新タスクのアクションを見直します。**アクションは、ASCII フォーマットでログ・ファイル cmtInstall.log に書き込まれます。このファイルのデフォルトの場所は /WebSphere/V7R0/AppServer/logs/wbi です。このログで >SEVERE< または >WARNING< レベルのテキスト・メッセージを検索し、発生したエラーを判別します。インストール・ディレクトリーから実行される Apache Ant スクリプトには、ASCII フォーマットでエラーを書き込むためのログがあります。

Ant スクリプトを含んでいるディレクトリーのデフォルト名は、/WebSphere/V7R0/AppServer/properties/version/install.wbi/6.0.0.0/config/full/uninstall です。生成された Ant ログは、製品ログ・ディレクトリーに書き込まれます。このディレクトリーのデフォルト名は、/WebSphere/V7R0/AppServer/logs/wbi です。Ant ログには、以下のログがあります (これらのログを確認して、処理中のエラーを判別します)。

- 90SDeleteFirstStepsFilesWBI.ant.log
- 90SRemoveJavaOptions.ant.log
- 90SUninstallCEI.ant.log
- 98SUndeployBPCAdminConsolePlugins.ant.log
- 98SUndeployServerAdminConsolePlugins.ant.log
- 99SUndeployCoreAdminConsolePlugins.ant.log

これらのログそれぞれには、アンインストールが正常に行われたときに書き込まれる「build successful」メッセージが含まれている必要があります。

- **拡張解除ログの内容を確認します。**プロファイル拡張解除タスクでは、ログ・ファイル (ASCII) に書き込むことによって、実行したアクションが記録されます。ログ・ファイル名は wasprofile_unaugment_default.log です。このファイルの標準的な場所は WebSphere/V7R0/AppServer/logs/wasprofile です。この WebSphere Application Server プロファイル拡張ログを検索して、>SEVERE< または >WARNING< レベルのメッセージがあるかどうかを調べ、処理中の全体的なエラーを判別します。通常、SEVERE メッセージは存在しません。

次のタスク

WebSphere Process Server for z/OS をシステムから除去したら、Business Process Choreographer 構成を除去する必要があります。

Business Process Choreographer のアンインストール

Business Process Choreographer を WebSphere Process Server のインストール済み環境から除去する方法については、WebSphere Process Server for z/OS バージョン 7.0 インフォメーション・センターにアクセスし、「**WebSphere Process Server のインストール**」>「**ソフトウェアのアンインストール**」>「**Business Process Choreographer 構成の除去**」にあるトピックを参照してください。この情報は *Business Process Choreographer PDF* でも検索できます。

インストールのトラブルシューティング

WebSphere Process Server のインストールに失敗したときに、問題を診断できません。

WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報

WebSphere Process Server for z/OS のメッセージ参照情報には、インストール・スクリプトまたは構成スクリプトの実行時に表示されるメッセージ・コードをリストします。

インストール・エラー・メッセージについて

WebSphere Process Server for z/OS メッセージ・コードのトラブルシューティングを行うときに、「説明」フィールドと「ユーザー応答」フィールドのデータを使用してください。

メッセージ・コードは CWPIZyyyyz という形式で表示されます。各部位には次のような意味があります。

- CWPIZ = WebSphere Process Server for z/OS メッセージ接頭語
- yyyy = 番号に割り当てられている数値 ID
- z = メッセージ・タイプの記述子 (E、I、または W)。各記述子は以下のタイプを示します。
 - E = エラー・メッセージ
 - I = 通知メッセージ
 - W = 警告メッセージ

WebSphere Process Server for z/OS インストール時のエラー・メッセージのリストについては、参照資料の『メッセージ (Messages)』の部分に記載されている CWPIZ を参照してください。

WebSphere Process Server for z/OS インストール・エラー・メッセージは、ランタイム・ディレクトリーの zSMPInstall.log ファイルに書き込まれます。このログ・ファイルの標準のデフォルトの場所は、/WebSphere/V7R0/AppServer/logs/wbi/zSMPInstall.log です。

WebSphere Process Server for z/OS 構成エラー・メッセージは、ランタイム・ディレクトリーの zWPSConfig.log ファイルおよび zWESBConfig.log ファイルに書き込まれます。これらのログ・ファイルの標準のデフォルトの場所は、それぞれ

/WebSphere/V7R0/AppServer/logs/wbi/zWESBConfig.log および
/WebSphere/V7R0/AppServer/logs/wbi/zWPSConfig.log です。

関連資料

270 ページの『T2 ネイティブ・ライブラリー db2jcc2zos のロードの失敗』
このエラーは、DB2 Universal Driver コネクタを使用する場合に発生する可能性があり、WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOAD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

WebSphere Process Server のログ・ファイル

インストールした製品のログ・ファイルは、2 つの異なるグループに分けられます。1 つ目のグループは、製品のインストール、製品の更新、およびプロファイル管理に関する情報が詳細に記録されるログです。もう 1 つのグループは、個別のプロファイルの特性および実行時アクティビティの詳細を記録するログです。

WebSphere Process Server をインストールおよびアンインストールするときや、プロファイルを作成、拡張、および削除するときは、各種のログ・ファイルが作成されます。製品のインストールおよび構成プロセスで問題が発生したときは、これらのログを調べます。製品インストール内のログ・ファイルとそれらの場所について詳しくは、『インストールとプロファイル作成のログ・ファイル』トピックを参照してください。

プロファイルごとに作成されるログ・ファイルも多数存在します。これらのログの一部では、プロファイルの作成に使用されたパラメーターが記述されています。このようなタイプのログ・ファイルは通常、プロファイルの構成が完了した後に変更されることはありません。他のプロファイル固有のログは、実行時に発行されたエラー、警告、および情報メッセージを収集するために継続的に更新されます。これらのログ・ファイルの一部は、モニター対象として選択された Common Base Event (ビジネス・オブジェクト・データを含む場合があります) を取り込むためにも使用されます。これらのログについては、『プロファイル固有のログ・ファイル』トピックを参照してください。

第 2 章 インストール情報

この参照用セクションには、WebSphere Process Server のインストールに関連するサブタスクと、インストールを支援するための概念説明および参照情報が記載されています。

スタンドアロン構成と Network Deployment 構成の相違点

スタンドアロン・アプリケーション・サーバー・セルはベース・アプリケーション・サーバー・ノードとも呼ばれ、WebSphere Process Server for z/OS アプリケーションをデプロイして実行するために使用できる最も単純な構成です。スタンドアロン構成は、即時始動環境または開発環境のためのフレームワークを提供します。小規模な環境を構成する場合は、単一サーバー・トポロジーのみで要件に対応できる可能性があります。ただし、Network Deployment 構成を使用する可能性の方が高くなります。これは、Network Deployment 構成では、ネットワーク環境内の複数の sysplex に分散するアプリケーション・サーバーを管理できるためです。Network Deployment 構成では、クラスター化すると高い可用性と信頼性を提供できる複数のアプリケーション・サーバーがサポートされます。

スタンドアロン構成と Network Deployment 構成の主な相違点を、以下に示します。

スタンドアロン構成	Network Deployment 構成
<p>以下のプロセスに対し 4 つ以上のアドレス・スペースが割り振られる。</p> <ul style="list-style-type: none"> ロケーション・サービス・デーモン コントローラー サーバント 制御領域付加属性 <p>アドレス・スペースの最大容量は、リソースによってのみ限定される。</p>	<p>以下のプロセスに対し 7 つ以上のアドレス・スペースが割り振られる。</p> <ul style="list-style-type: none"> ロケーション・サービス・デーモン アプリケーション・サーバー・コントローラー アプリケーション・サーバー・サーバント アプリケーション・サーバー制御領域付加属性 デプロイメント・マネージャー・コントローラー デプロイメント・マネージャー・サーバント ノード・エージェント <p>アドレス・スペースの最大容量は、リソースによってのみ限定される。</p>
各サーバー・ノードはそれぞれ個別の管理ドメインに配置されている。	すべてのノードが同一管理可能ドメインに配置されている。
サーバーの開始と停止を単独に実行できる。各サーバーには、単独の非共用 JNDI 名前空間がある。	サーバーの開始と停止を単独に実行できるが、JNDI 名前空間はセル内のすべてのサーバーにより共有される。
複数のサーバントを使用できる。	複数のサーバントを使用できる。

スタンダード構成	Network Deployment 構成
クラスター化できない。	クラスター化できる。

インストール・メディアのコンテンツ

z/OS システムにインストールされる WebSphere Process Server for z/OS コンポーネントは、1 つのテープまたはダウンロード・パッケージの形で提供されます。z/OS 以外のシステムにインストールされるコンポーネントは、CD で提供されます。

WebSphere Process Server for z/OS に付属するソフトウェア

WebSphere Process Server for z/OS パッケージには、WebSphere Process Server for z/OS のインストールと構成、およびアプリケーションのアセンブルとデプロイに必要なソフトウェアがすべて含まれています。

テープとダウンロード・パッケージには WebSphere Application Server for z/OS が含まれています。これは、WebSphere Process Server for z/OS のインストール前に、インストールして構成しておく必要があります。テープまたはダウンロード・パッケージで提供されるソフトウェアを、次の表に示します。

ソフトウェア	説明
WebSphere Process Server for z/OS	メインの WebSphere Process Server for z/OS ソフトウェアです。
WebSphere Application Server for z/OS	WebSphere Application Server for z/OS ソフトウェアです。これは、WebSphere Process Server for z/OS ソフトウェアのインストール前に、インストールして構成しておく必要があります。
WebSphere Application Server Application Clients	クライアント・アプリケーションのスタンダード・クライアント・ランタイム環境です。これにより、WebSphere Application Server for z/OS インスタンス全体をインストールする必要はありません。アプリケーション・クライアント・モジュールは、Java アプリケーションにアクセスするためのクライアントが含まれている Java Archive (JAR) ファイルです。
DataDirect Java ^(TM) Database Connectivity (JDBC) ドライバー	Microsoft ^(R) SQL Server との接続を可能にする目的で DataDirect Technologies により開発された 2 つの JDBC ドライバーです。これらのドライバーとは、SequeLink ドライバーおよび Connect JDBC ドライバーです。

CD には、実動環境と開発環境に役立つツールを提供するオプションの補足ソフトウェアが含まれています。CD で提供されるソフトウェアを、次の表に示します。

ソフトウェア	説明
WebSphere Process Server クライアント	<p>SOA コアおよび Business Process Choreographer 機能を含む WebSphere Application Server for z/OS 構成を提供します。完全な WebSphere Process Server のインストールは不要です。</p> <ul style="list-style-type: none"> • WebSphere Process Server 6.2 Client for Windows® • WebSphere Process Server 6.2 Client for AIX® • WebSphere Process Server 6.2 Client for Solaris on SPARC • WebSphere Process Server 6.2 Client for Solaris on x86 64-bit • WebSphere Process Server 6.2 Client for HP-UX • WebSphere Process Server 6.2 Client for HP-UX on Itanium® 64-bit • WebSphere Process Server 6.2 Client for Linux® on x86 32-bit • WebSphere Process Server 6.2 Client for Linux on x86 64-bit • WebSphere Process Server 6.2 Client for Linux on POWER® • WebSphere Process Server 6.2 Client for Linux on System z® 31-bit • WebSphere Process Server 6.2 Client for Linux on System z 64-bit
IBM Message Service Clients	<p>Java 以外の環境にメッセージング機能と Web サービス機能を提供します。提供された以下のクライアントを使用することにより、アプリケーションと WebSphere Process Server 間の対話を拡張します。</p> <ul style="list-style-type: none"> • IBM Message Service Client for C/C++ は JMS モデルを拡張して、C および C++ アプリケーションへのメッセージングを行います。 • IBM Message Service Client for .NET を使用すると、.NET アプリケーションを JMS ベースの情報フローに加えることができます。
WebSphere Application Server Edge Components	<p>ロード・バランシング、キャッシング、および集中セキュリティ機能を提供します。詳細については、WebSphere Application Server Network Deployment Edge Components の Web ページを参照してください。</p>
WebSphere Application Server Network Deployment V7.0 Supplements	<p>WebSphere Process Server と合わせて使用できるようにライセンスされている WebSphere Application Server Network Deployment の機能を提供します。</p>
WebSphere Application Server interim fix 6.2.0.13-WAS-IFPK56164	<p>WebSphere Application Server Network Deployment 6.2 の暫定修正です。</p>

ソフトウェア	説明
IBM Tivoli® Access Manager Servers	ビジネスに不可欠なアプリケーションと拡大した企業内に分散するデータにセキュアにアクセスするための認証および許可用の API と、機能統合を提供します。詳しくは、『IBM Tivoli Access Manager for e-business』を参照してください。
Rational® Agent Controller 6.2.5 for Windows	Rational Agent Controller は、問題判別機能を備えています。Rational Agent Controller は、Autonomic Computing Toolkit の Agent Controller を拡張したものです。Rational Agent Controller を、サポートされている z/OS システム環境のいずれか 1 つにインストールします。別の z/OS システム (管理者のワークステーションなど) で Log and Trace Analyzer を使用して、リモート z/OS システムのエージェントと通信を確立できます。Analyzer は、Rational Agent Controller の特定のログ・ファイルを要求できます。エージェントは、適切なログ・ファイル・パーサーを使用してネイティブ・ログを Common Base Event 形式に正規化し、ログ・ファイルを Log and Trace Analyzer に転送します。Autonomic Computing Toolkit に含まれている Common Base Event モデル・ビルダーを使用して、Rational Agent Controller のログ・ファイル・パーサーを作成します。

WebSphere Process Server for z/OS の入手方法

以下のいずれかの方法で、製品コードを入手できます。

- IBM Custom-Built Product Delivery Option (CBPDO) - システム・プログラマーが SMP/E を使用して、製品コードを z/OS システムにアンロードします。
- IBM SystemPac/ServerPac - システム・プログラマーが、CustomPac サービス・レベルに対応する SMP/E データ・セットを z/OS システムにコピーします。

ソフトウェアを購入するには、IBM の担当員または IBM の販売店に連絡するか、WebSphere Process Server ホーム・ページ (<http://www.ibm.com/software/integration/wps/>) にアクセスして *How to buy* リンクを選択してください。

ノード、サーバー、ホスト、およびセルの命名の考慮事項

WebSphere® Application Server のインストール済み環境をカスタマイズするとき、いくつかの事項について考慮する必要があります。詳しくは、『カスタマイズ変数の設定: スタンドアロン・アプリケーション・サーバー・セル』を参照してください。

WebSphere Process Server の機能

このトピックでは、Installation Manager でインストール対象として選択可能な WebSphere Process Server 機能について説明します。

サンプル・アプリケーション

Installation Manager で WebSphere Process Server の「サンプル・アプリケーション」機能を選択することにより、WebSphere Process Server と WebSphere Application Server の両方のためのサンプル・アプリケーションをインストールに組み込むかどうかが決まります。サンプル・アプリケーションには、ソース・コード・ファイルと、Java Platform Enterprise Edition (Java EE) および WebSphere のいくつかの最新テクノロジーをデモンストレーションする統合エンタープライズ・アプリケーションの両方が含まれています。

サンプル・アプリケーションについて詳しくは、『サンプル・ギャラリーのインストールおよびアクセス』を参照してください。

実稼働環境でパフォーマンスを高めるためには、サンプル・アプリケーションをインストールしないでください。

WebSphere Process Server - Client

機能パネルで「**WebSphere Process Server - Client**」を選択すると、WebSphere Process Server Client と WebSphere Process Server がインストールされます。WebSphere Process Server Client だけをインストールするには、WebSphere Process Server のチェック・ボックスをクリアします。

スタンドアロン WebSphere Process Server または WebSphere Enterprise Service Bus 開発プロファイル

Installation Manager には、WebSphere Process Server と WebSphere Enterprise Service Bus の両方のためのスタンドアロン開発プロファイルを作成するオプション機能が含まれています。これらのプロファイルは、実稼働環境では機能しません。これらは、ユーザーが、運用実動プロファイルを作成することなく、WebSphere Process Server または WebSphere Enterprise Service Bus についての理解を深めることを目的としています。これらのプロファイルを作成するには、管理者セキュリティー ID およびパスワード資格情報を提供する必要があります。

製品のバージョン情報および履歴情報

製品のバージョン情報および履歴情報とそのリンク

properties/version ディレクトリーにある WBI.product ファイルには、製品、バージョン、ビルドの日付、およびビルド・レベルなどの情報が記載されています。以下に例を示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE product SYSTEM "product.dtd">
<product name="IBM WebSphere Process Server">
<id>WBI</id>
<version>7.0.0.0</version>
<build-info date="8/31/09" level="of0935.02"/>
</product>
```

該当する製品バージョン情報および履歴情報については、以下のリンクをクリックしてください。

表 2. 製品のバージョン情報および履歴情報のリンク

リンク
製品のバージョン情報
genVersionReport コマンド
versionInfo コマンド
historyInfo コマンド
genHistoryReport コマンド

製品のバージョン情報

WebSphere Process Server for z/OS インストール・ファイル・システムの `properties/version` ディレクトリーには、製品とインストールされているコンポーネントに関する重要なデータ (ビルド・バージョンやビルド日など) が格納されています。

製品情報ファイル

この情報は、`WBI.product` ファイルと `[component].component` ファイルに記述されています。

インストールされている保守パッケージに関するレポートを作成するには、`historyInfo` コマンドを実行します。`historyInfo` コマンドは、コンソール上にレポートを作成し、`config_root/properties/version/history` ディレクトリーに追跡ファイルも作成します。

タイム・スタンプ付きの詳細なログに、`configuration_root` の `properties/version/log` ディレクトリーでの各更新プロセスが記録されます。

この記事では、WebSphere Process Server for z/OS バージョン 6.0.x の製品バージョン情報を記述した XML データ・ファイルについて説明します。これらのファイルの文書タイプ宣言 (DTD) は、デフォルトではインストール・ファイル・システムのルートの `properties/version/dtd` フォルダー、またはサーバー・ルート・ディレクトリーに格納されています。詳しくは、このトピックのディレクトリーの場所に関する情報を参照してください。

properties/version ディレクトリーの次の XML ファイルにバージョン情報が記述されています。

platform.websphere

このファイルが存在している場合は、WebSphere Application Server 製品がインストールされています。このファイルの例を、次に示します。

```
<?xml version="1.0" encoding="UTF-8">
<!DOCTYPE websphere PUBLIC "websphereId" "websphere.dtd"
<websphere name="IBM WebSphere Application Server" version="6.0"/>
```

`properties/version` ディレクトリーに保管される以下の XML ファイルには、インストールされているコンポーネントと、インストール・イベント (製品のエディション、バージョン、コンポーネント、およびビルド情報など) が記述されています。

WAS.product

このファイルが存在している場合は、特定の WebSphere Application Server 製品がインストールされています。インストールされている製品のタイプは、<id> タグにより示されます。このファイルのデータは、バージョン、ビルド日、およびビルド・レベルを示します。

例えば、<id>ND</id> 製品は、WebSphere Application Server Network Deployment がインストールされていることを示します。このファイルの例を、次に示します。

```
<?xml version="1.0" encoding="UTF-8">
<!DOCTYPE websphere PUBLIC "productId" "product.dtd"
<product name name="IBM WebSphere Application Server - ND">
<id>ND</id>
<version>6.0.0</version>
<build-info date="02/03/05" level="s0461.18"/>
</product>
```

WBI.product

このファイルが存在している場合は、特定の WebSphere Process Server for z/OS 製品がインストールされています。インストールされている製品のタイプは、<id> タグにより示されます。このファイルのデータは、バージョン、ビルド日、およびビルド・レベルを示します。

このファイルの例を、次に示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE product SYSTEM "product.dtd">
<product name="IBM WebSphere Process Server">
<id>WBI</id>
<version>6.2.0.0</version>
<build-info date="11/15/08" level="o0845.22"/>
</product>
```

レポート

インストールが完了すると、WebSphere Application Server for z/OS のレポートには、インストールされた WebSphere Process Server for z/OS 製品が反映されます。詳しくは、WebSphere Application Server for z/OS インフォメーション・センターの『製品バージョン情報 (product version information)』を参照してください。

応答ファイルの値

WebSphere Process Server for z/OS 製品定義をインストールするインストール・ジョブを実行すると、サンプル応答ファイルがインストール・ファイル・システムにインストールされます。これらの応答ファイルをコピーし、実現する構成に従って編集し、拡張スクリプトの実行時にその応答ファイルを渡します。

目的

応答ファイルには、WebSphere Process Server を構成するために使用できるキーワードが含まれています。応答ファイルのコピーを作成し、そのファイルを書き込み可能にしてから、作成する構成の要件を満たすようにファイルの調整を開始します。作業が完了したら、変更を保存します。

応答ファイルを編集するときは、以下の規則とガイドラインを使用します。

- キーワードの最後に、読めない余分な文字を残さないでください。
- 応答ファイルの編集には、TSO oedit または vi を使用します。
- コメントは、新しい行で開始します。キーワードの右側からコメントを書き始めないでください。
- コメントは、1 桁めから番号記号 (#) を付けて記入します。
- 応答ファイルは EBCDIC フォーマットでなければなりません。ワークステーションを使用してファイルを編集する場合は、FTP によるダウンロードおよびアップロード時にテキスト形式を使用してください。

応答ファイルの先頭にはグローバル変数が表示され、ドル (\$) 記号の付いたファイル内の他の変数によって参照されます。serverName 変数は、グローバル変数を参照する変数の一例です。

以下の表に 4 つのサンプル応答ファイルの名前を示します。これらのファイルを使用することにより、さまざまな WebSphere Process Server for z/OS 構成を作成できます。

表 3.

standAloneProfile.rsp	このファイルを使用して、Derby™ データベースを使用したスタンドアロン構成を作成します。スタンドアロン構成には、アプリケーション・サーバーを実行する単一ノードと、単一 z/OS® システムまたは LPAR で稼働する 1 つのデーモン・サーバーが含まれています。Derby データベースは、テスト・システムにのみ使用してください。
standAloneProfileDB2.rsp	このファイルを使用して、DB2® データベースを使用したスタンドアロン構成を作成します。スタンドアロン構成には、アプリケーション・サーバーを実行する単一ノードと、単一 z/OS® システムまたは LPAR で稼働する 1 つのデーモン・サーバーが含まれています。実動システムには DB2® データベースを使用してください。
DmgrDB2.rsp	このファイルを使用して、DB2 データベースを使用する Network Deployment 構成を備えたデプロイメント・マネージャー・サーバーを作成します。基本的な Network Deployment 構成では、1 つのノードにデプロイメント・マネージャー・サーバー、もう 1 つのノードにアプリケーション・サーバーがあります。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合されるため、デプロイメント・マネージャーで管理できます。Network Deployment 構成では、両方のノードが WebSphere Process Server 機能で拡張されます。

表 3. (続き)

ManagedDB2.rsp	このファイルを使用して、DB2 データベースを使用する Network Deployment 構成を備えたノードを作成します。基本的な Network Deployment 構成では、1 つのノードにデプロイメント・マネージャー・サーバー、もう 1 つのノードにアプリケーション・サーバーがあります。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合されるため、デプロイメント・マネージャーで管理できます。Network Deployment 構成では、両方のノードが WebSphere Process Server 機能で拡張されます。
----------------	--

次の表に、WebSphere Process Server for z/OS 構成を作成するために使用できるデフォルトの応答ファイル・キーワードを、アルファベット順に示します。これらのキーワードの多くは、応答ファイルで未設定にすると、デフォルトの適切な値に設定されます。また、WebSphere Process Server for z/OS のインストール済み環境を生成する場合、インストール時にこれらのキーワードのすべてを応答ファイルに組み込む必要はありません。

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト

-adminPassword	-adminUserName キーワードで指定されている管理セキュリティ・ユーザー ID のパスワード。管理セキュリティが有効になっている既存のプロファイルを拡張するには、このキーワードが必須です。
-adminUserName	管理セキュリティに使用されるユーザー ID。管理セキュリティが有効になっている既存のプロファイルを拡張するには、このキーワードが必須です。
-augment	augment パラメーターは、拡張テンプレートを使用して、WebSphere Process Server for z/OS 機能を備えたノードを構成します。
-bpcDbConnectionLocation	Business Process Choreographer を構成するためのデータベース・ロケーション。この値は、WebSphere Process Server データ・ソース定義のデータベース名の代わりに使用されます。例えば、スタンドアロン構成の場合は DB2=LOC1 です。
-bpcDbHostName	ビジネス・プロセス・コンテナー・データベースの構成のためのデータベース・サーバー・ホスト名または IP アドレス。
-bpcDbJDBCClasspath	JDBC ドライバー・ファイルのロケーション。
-bpcDbName	ビジネス・プロセス・コンテナー・データベース構成用のデータベース名。

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト (続き)

-bpcDbProduct	<p>ビジネス・プロセス・コンテナの構成用のデータベース・ロケーション。例えば、スタンドアロン構成の場合は DB2=DB2UDBOS390_V8_1 です。このキーワードに使用可能な値は、以下のとおりです。</p> <ul style="list-style-type: none"> • DERBY_EMBEDDED • DERBY_NETWORKSERVER • DB2_UNIVERSAL • DB2UDBOS390_V7_1 • DB2UDBOS390_V8_1 • DB2UDBOS390_V9_1 • DB2UDBISERIES_NATIVE • DB2UDBISERIES_TOOLBOX • INFORMIX • MSSQLSERVER_EMBEDDED • MSSQLSERVER_DATADIRECT • ORACLE9I、ORACLE10G
-bpcDbPassword	<p>ビジネス・プロセス・コンテナの JDBC リソースを認証するためのパスワード。</p>
-bpcDbSqlId	<p>ビジネス・プロセス・コンテナが使用する DB2 スキーマ修飾子。この値は、インストールによって作成される ddl/sql 内で置換されます。</p>
-bpcDbUser	<p>ビジネス・プロセス・コンテナ・データベースの作成および除去の特権を持つ DB2 ユーザー ID。</p>
-bpcDbServerPort	<p>ビジネス・プロセス・コンテナ・データベースのポート番号。</p>
-bpcDbStorageGroup	<p>ビジネス・プロセス・コンテナが使用する DB2 スキーマ修飾子。この値は、インストール中に作成される ddl/sql 内で置換されます。例えば、スタンドアロン構成の場合は DB2=BPEDBSTO です。</p>
-bpcmqPassword	<p>ビジネス・プロセス・コンテナの MQ リソースを認証するためのパスワード。</p>
-bpcmqUser	<p>WebSphere Process Server MQ を使用するビジネス・プロセス・コンテナのリソースの構成におけるユーザー名。</p>
-bspaceAlreadyConfigured	<p>このプロファイルに Business Space が構成済みかどうかを指定する boolean のキーワード。</p>
-bspaceAlreadyDeployed	<p>このサーバーに Business Space がデプロイ済みかどうかを指定する boolean のキーワード。</p>
-bspaceSchemaName	<p>Business Space テーブルに使用される DB2 スキーマ修飾子。このキーワードは、インストールによって作成される ddl/sql 内で置換されます。</p>
-cbeServerName	<p>Common Base Event ブラウザーの実行が構成されているサーバーの名前。</p>

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト (続き)

-cdbSchemaName	<p>共通データベースのスキーマ名。このパラメーターは、Oracle/Informix/MSSQLSERVER_DataDirect/MSSQLSERVER_MICROSOFT 以外のすべての dbType で有効です。dbSchemaName を設定する場合は、このキーワードを DB2UDBOS390_V8_1/DB2UDBOS390_V9_1/DB2UDBISERIES_TOOLBOX の dbType に使用しないでください。データベースのデフォルトのスキーマ値は、以下のとおりです。</p> <ul style="list-style-type: none"> • Derby_Embedded/Derby_Embedded40 は、「APP」 • Derby_NetwrokServer/Derby_NetwrokServer40 は、「dbUserId」 • DB2_Universal/DB2_dataServer は、「dbUserId」 • DB2UDBOS390_V8_1/DB2UDBOS390_V9_1 は、「dbUserId」 • DB2UDBISERIES_TOOLBOX は、「dbUserId」 <p>注: サポートされるデータベースに別のスキーマを指定することは可能ですが、dbType が Derby_Embedded/Derby_Embedded40/DB2_Universal/DB2_dataServer/Derby_NetwrokServer/Derby_NetwrokServer40 の場合、リカバリー・テーブルは常にデフォルトのスキーマを使用して作成されます。例えば、cdSchemaName が「mySchema」で dbUserId が「myUser」であっても、derby_embedded の場合、リカバリー表スキーマは mySchema でなく APP になります。同様に、derby_networkservr の場合、リカバリー表スキーマは mySchema でなく dbUserId になり、db2_universal/db2_dataserver の場合、リカバリー表スキーマは mySchema でなく dbUserId になります。</p>
-ceiAlreadyDeployed	このサーバーに Common Event Infrastructure がデプロイ済みかどうかを指定する boolean のキーワード。
-ceiBufferPool4k	Common Event Infrastructure 用の 4K バッファ・プールの名前。このバッファ・プールは、データベース DDL スクリプトを実行する前にアクティブにする必要があります。
-ceiBufferPool8k	Common Event Infrastructure 用の 8K バッファ・プールの名前。このバッファ・プールは、データベース DDL スクリプトを実行する前にアクティブにする必要があります。
-ceiBufferPool16k	Common Event Infrastructure 用の 16K バッファ・プールの名前。このバッファ・プールは、データベース DDL スクリプトを実行する前にアクティブにする必要があります。
-ceiCacheSizeInMB	トランザクション・ログに使用されるキャッシュのサイズ (MB)。
-ceiCreateLogin	オプション・パラメーター。true の場合は、Common Event Infrastructure 用のイベント・サービス Sybase テーブルを所有するログイン・ユーザー ID が作成されます。
-ceiDbInstallDir	Common Event Infrastructure 用にデータベースがインストールされているディレクトリー。
-ceiDbName	Common Event Infrastructure の DB2 名。
-ceiDiskSizeInMB	イベント・サービス用に作成されるデータベースのサイズ (MB)。DB2 for z/OS の場合、デフォルト値は 10 です。別の値を指定する場合は、10 以上の値を指定する必要があります。
-ceiEventCatalogDBName	イベント・カタログ・データベースの名前。
-ceiInstancePrefix	CEI インスタンスで使用される接頭部。
-ceiFindDeviceNumber	イベント・データベースによって、Common Event Infrastructure 用に 6 つのデバイスが作成されます。このキーワードは、新規デバイスに割り当てる必要のある最初のデバイス番号の値を識別します。指定されない場合、デフォルト値は 10 です。

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト (続き)

-ceiOverrideDataSource	指定された有効範囲の既存の Common Event Infrastructure サービス・データ・ソースを除去した後に、新規作成するかどうかを指定するキーワード。このキーワードが true に設定されると、コマンドは指定された有効範囲の既存の Common Event Infrastructure サービス・データ・ソースを除去した後に、新規作成します。このキーワードが false に設定されると、指定された有効範囲に別のイベント・サービス・データ・ソースが検出された場合、コマンドは同じ有効範囲にイベント・サービス・データ・ソースを作成しません。指定されない場合、デフォルト値は FALSE です。
-ceiOracleHome	(非推奨) ORACLE_HOME のディレクトリー。
-ceiSaUser	Common Event Infrastructure のテーブル、デバイス、およびキャッシュを作成するための特権を持つ Microsoft SQL Server ID。
-ceiSaPassword	Common Event Infrastructure のテーブル、デバイス、およびキャッシュを作成するための特権を持つ Microsoft SQL Server ID のパスワード。ceiSaUser キーワードに値を指定したときにこのキーワードは必須ですが、sa ユーザー ID がパスワードを持たない場合は除きます。
-cellName	プロファイルのセル名。これは、拡張対象の WebSphere Application Server プロファイルのセル名と一致しなければなりません。このキーワードの値には、スペース、コンマ、または無効な文字 (/, ¥, *, :, ;, =, +, ?, !, <, >, &, %, ', ",]]>, #, \$, ^, {, } など) を含めないでください。ピリオド (.) は、先頭文字には使用できません。
-configureBPC	Business Process Choreographer のサンプル構成が作成されたかどうかを判別するキーワード。このキーワードを true に設定する場合は、-adminUserName および -adminPassword キーワードも設定する必要があります。このキーワードのデフォルトは -enableAdminSecurity キーワードと同じ値です。 注: Business Process Choreographer のサンプル構成では、共通データベース (WPRCSDB) を使用しません。この構成では、必ず Derby データベースが使用されますが、このデータベースは Network Deployment 環境ではサポートされていません。このスタンドアロン・サーバーを後で統合する予定の場合は、-configureBPC に false を設定してください。
-configureBRM	ビジネス・ルール・マネージャーを構成するかどうかを判別するキーワード。デフォルト値は false です。
-configureBSpace	Business Space を構成するかどうかを判別するキーワード。デフォルト値は false です。
-createDefaultProfileForMigration	サーバーがバージョン間マイグレーションのターゲットとなることを指定する boolean のキーワード。スタンドアロン・サーバーを拡張してマイグレーション・ターゲットとして使用する場合は、この値を true に設定してください。このキーワードは、重複する CEI メッセージング・エンジンが作成されるのを防ぎます。
-dbAlreadyConfigured	データベースが構成済みかどうかを指定する boolean のキーワード。
-dbAppMeUserId	メッセージング・エンジンが共通 DB を使用している場合に認証に必要なユーザー ID。
-dbAppMePassword	メッセージング・エンジンが共通 DB を使用している場合に認証に必要なパスワード。
-dbBPCMeUserId	Business Process Choreographer メッセージング・エンジン・オブジェクトの作成に必要なユーザー ID。このキーワードは、-configureBPC が true に設定されている場合にのみ有効です。

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト (続き)

-dbBPCMePassword	Business Process Choreographer メッセージング・エンジン・オブジェクトの作成に必要なパスワード。このキーワードは、-configureBPC が true に設定されている場合にのみ有効です。
-dbCeiMeUserId	CEI メッセージング・エンジンが共通 DB を使用している場合に認証に必要なユーザー ID。
-dbCeiMePassword	CEI メッセージング・エンジンが共通 DB を使用している場合に認証に必要なパスワード。
-dbCommonForME	メッセージング・エンジンの表に共通データベースを使用するかどうかを指示するキーワード。デフォルト値は false です。このキーワードが true に設定された場合、メッセージング・エンジンの表に共通データベースが使用されますが、メッセージング・エンジンの表は共通データベース内で手動で作成する必要があります。詳しくは、86 ページの『メッセージ・エンジンのデータ・ストアの作成』を参照してください。
-dbConnectionLocation	DB2 for z/OS データベースの場所。
-dbCeiUserId	CEI データベースの認証に必要なユーザー ID。
-dbCommonUserId	共通データベースの認証に必要なユーザー ID。
-dbCeiPassword	CEI データベースの認証に必要なパスワード。
-dbCommonPassword	共通データベースの認証に必要なパスワード。
-dbCreateNew	共通 DB を格納する新規データベースを作成するか、既存データベースを使用するかを示す boolean のフラグ。
-dbDelayConfig	プロファイルの作成後までテーブル作成を延期するかどうかを示すキーワード。有効な値は true または false です。このキーワードは、デフォルトでは false に設定されています。このキーワードを true に設定するか false に設定するかは、構成内容によって異なります。すべての必要なデータベースがセットアップ済みで準備できている場合は、-dbDelayConfig を true に設定できます。すべての必要なデータベースがセットアップ済みで準備できていない場合は、-dbDelayConfig を false に設定して、拡張後にデータベースをセットアップできます。
-dbDriverType	データベースのドライバー・タイプ。Oracle データベースの場合、有効な値は THIN または OCI です。DB2 データベースの場合、有効な値は 2 または 4 です。
-dbHostName	データベース・サーバーのホスト名または IP アドレス。デフォルト値は localhost です。
-dbJDBCClasspath	共通 DB に必要な JDBC クラス・パス。
-dbInstance	Informix データベースのデータベース・インスタンス名。
-dbLocation	データベース・サーバーのロケーション (データベース製品のインストール・ルート)。
-dbName	WebSphere Process Server データベースの名前。デフォルト値は WPRCSDB です。
-dbOutputScriptDir	出力スクリプトのディレクトリー・ロケーション。
-dbPassword	データベース認証に必要なパスワード。このキーワードは、Derby Embedded を除くすべてのデータベースに必要です。
-dbProviderType	現行の dbType のプロバイダー・タイプを指定するオプション・パラメーター。現在は、Informix dbType にのみ適用されます。
-dbSchemaName	データベース・スキーマの名前。このキーワードは、DB2UDBOS390_V8_1/DB2UDBOS390_V9_1/DB2UDBISERIES_TOOLBOX の dbType で有効です。バージョン 7 では非推奨であるため、代わりに cdbSchemaName キーワードを使用してください。

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト (続き)

-dbServerPort	データベース・サーバーのポート番号。使用しているデータベースに応じて、デフォルトのポート番号 446 以外の別のポート番号を指定できます。
-dbStorageGroup	DB2 for z/OS データベースのストレージ・グループの名前。
-dbSysMeUserId	データベース・システム・メッセージング・エンジン・ユーザー ID。
-dbSysMePassword	データベース・システム・メッセージング・エンジン・パスワード。
-dbSysUserId	データベースにアクセスするためのユーザー ID。
-dbSysPassword	データベースにアクセスするためのパスワード。
-dbType	データベース・タイプ。WebSphere Process Server データベースと併用しているデータベース製品のタイプとして、以下の値のいずれかを設定します。 <ul style="list-style-type: none"> • Derby Embedded データベースの場合は DERBY_EMBEDDED • DB2UDBOS390_V8_1 (DB2 for z/OS v8 データベース) • DB2UDBOS390_V9_1 (DB2 for z/OS v9 データベース)
-dbUserId	データベース認証に必要なユーザー ID。このキーワードは、Derby Embedded を除くすべてのデータベースに必要です。
-dmgrAdminPassword	デプロイメント・マネージャーが実行されているシステムへのユーザー管理アクセス権限を取得するためのパスワード。
-dmgrAdminUserName	デプロイメント・マネージャーが実行されているシステムへのユーザー管理アクセス権限を取得するためのユーザー名。
-dmgrHost	<p>デプロイメント・マネージャーが稼働しているシステムを識別するキーワード。空ノードの作成時にそのノードを統合するには、このキーワードを dmgrPort キーワードと一緒に指定します。ホスト名には、デプロイメント・マネージャー・システムの長い DNS 名、短い DNS 名、または IP アドレスを指定できます。このオプション・キーワードを指定すると、空ノードの作成時に空のノードをデプロイメント・マネージャー・セルに統合するように、構成プロセスに指示することになります。</p> <p>デプロイメント・マネージャーまたはアプリケーション・サーバーの作成時には、このキーワードは無視されます。デプロイメント・マネージャーが実行されていない場合、またはセキュリティが有効になっているなどの理由でデプロイメント・マネージャーを使用できない場合に空ノードを統合すると、ログ内のインストール標識は、完全な失敗を示す INSTCONFFAIL になります。結果としての空のノードは、使用できません。同じ名前の空のノードをもう 1 つ作成するには、プロファイル・リポジトリ (プロファイルのインストール・ルート・ディレクトリー) から、この空のノードのプロファイル・ディレクトリーを移動しておく必要があります。</p> <p>セキュリティを有効にしてある場合、またはデフォルトの JMX コネクター・タイプを変更した場合は、構成プロセスの実行中に統合することはできません。代わりに addNode コマンドを使用してください。このキーワードのデフォルト値は localhost です。このキーワードの値は、適切な形式のホスト名でなければなりません。また、スペースや無効な文字 (*、?、",、V、<、>、,、/、¥、 など) を含めることはできません。さらに、dmgrPort キーワードと一緒に指定することによって、デプロイメント・マネージャーへの接続を使用可能にする必要があります。例: dmgr_host_name。</p>

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト (続き)

-dmgrPort	<p>デプロイメント・マネージャーの SOAP ポートを識別するキーワード。空ノードの作成時にそのノードを統合するには、このキーワードを dmgrHost キーワードと一緒に指定します。デプロイメント・マネージャーは、稼働していてアクセス可能である必要があります。セキュリティを有効にしてある場合、またはデフォルトの JMX コネクタ・タイプを変更した場合は、構成プロセスの実行中に統合することはできません。代わりに addNode コマンドを使用してください。このキーワードのデフォルト値は 8879 です。指示するポートは正整数である必要があり、dmgrHost キーワードと一緒に指定することによって、デプロイメント・マネージャーへの接続を使用可能にする必要があります。例: dmgr_port_number。</p>
-enableAdminSecurity	<p>管理セキュリティを使用可能にするキーワード。考えられる値は次のとおりです。</p> <ul style="list-style-type: none"> • True: enableAdminSecurity を true に設定した場合は、-adminUserName および -adminPassword キーワードも、それらの値と共に指定する必要があります。 • False: enableAdminSecurity を false に設定した場合は、他のキーワードを設定する必要はありません。 <p>デフォルト値は false です。</p>
-fileStoreForME	<p>メッセージング・エンジンのファイル・ストアのロケーション。このキーワードは、スタンドアロン・サーバーの構成にのみ適用されます。デフォルト値は false で、-dbCommonForME も true に設定している場合は、このキーワードを true に設定できません。</p>
-isDeveloperServer	<p>サーバーを開発の目的にのみ使用するかどうかを指定します。</p>
-portsFile	<p>新規プロファイルのポート設定を定義しているファイルへのパスを指定するオプション・キーワード。</p>
-profileName	<p>プロファイル拡張に使用されるキーワード。各構成で、プロファイルは常にデフォルトの名前になります。プロファイルが統合済みであってはけません。プロファイルの拡張時にサーバーが稼働中であってはなりません。</p>
-profilePath	<p>WebSphere Application Server for z/OS のデフォルト・プロファイル・ルート・ディレクトリー。このキーワードは必須です。</p> <p>WebSphere Application Server for z/OS のサーバー構成は、profiles ディレクトリーの下のディレクトリー構造内にあります。プロファイル・パスには、ランタイム環境を定義するファイル (コマンド、構成ファイル、ログ・ファイルなど) が含まれます。Apache Ant スクリプトの制限によって、プロファイルのセルへの統合時に障害が発生することを回避するために、絶対パスを指定してください。このキーワードを指定しない場合は、拡張手順で、プロファイルの場所を示すパスが WebSphere Application Server for z/OS 構成の中で検索されます。例えば、次のようになります。</p> <ul style="list-style-type: none"> • スタンドアロン Derby=/WebSphere/V7R0/AppServer/profiles/default • スタンドアロン DB2=/WebSphere/V7R0/AppServer/profiles/default • Network Deployment=/WebSphere/V7R0/DeploymentManager/profiles/default • Network Deployment の管理対象ノード=/WebSphere/V7R0/AppServer/profiles/default
-serverNameplatform	<p>サーバーが稼働するプラットフォームの名前。</p>
-serverType	<p>管理プロファイルのタイプを指定するキーワード。管理プロファイルに DMGR を指定します。このキーワードは、管理プロファイルを作成する場合は必須です。</p>

表 4. WebSphere Process Server for z/OS 応答ファイルのキーワードのアルファベット順リスト (続き)

-soaCoreAugmentType	SOA コア拡張情報のタイプを指定するキーワード。
-topologyPattern	デプロイメント・マネージャーのトポロジー・パターン、つまり、なし (デフォルト値)、CondensedSync、CondensedAsync、または Reference を決定するキーワード。
-topologyRole	作成済みのプロファイルを統合するときに、プロファイルがデプロイメント環境で果たす機能を示すキーワード。有効な値は、デプロイメント・ターゲットの場合は ADT、ホスト・メッセージングの場合は Messaging、またはサービスをサポートする場合は Support です。1 つの値を指定するか、または複数の値をそれぞれスペースで区切って指定できます。例えば、ADT Messaging Support または Messaging または ADT Support とします。
-wbidbDesign	プロファイル作成中に、すべてのコンポーネントの単一設計ファイルを指定するために使用するキーワード。design_file への完全修飾パスを指定する必要があります。

応答ファイルの使用

- 応答ファイルにコメントを追加する場合は注意してください。

応答ファイルにコメントを書き込む場合、そのコメントをプロパティー値と同じ行に置いてしまうと、コメントの前にクロス・ハッチ文字 (#) を置いた場合でも、Ant スクリプトはコメントをキーワード値の一部として読み取ろうとするため、予測不能な結果になってしまいます。例えば、この例に示すようなコメントを入力してはなりません。

```
#####
# DB2 Properties
#####
dbJDBCClasspath=/shared/db2810/jcc/classes # DB2 ClassPath Location
dbJDBCProperties=/u/hutch/wpswork/ # DB2JccConfiguration.properties
```

上記のサンプルに示すコメントを正しく追加する方法は、以下のようになります。

```
#####
# DB2 Properties
#####
# DB2 ClassPath Location
dbJDBCClasspath=/shared/db2810/jcc/classes
# DB2JccConfiguration.properties
dbJDBCProperties=/u/hutch/wpswork/
```

グローバル・プロパティー

応答ファイルの先頭には、**GLOBAL Properties** という名前のセクションが含まれています。以下に例を示します。

```
#####
# GLOBAL Properties
#####
JMSUSER=ibmuser
JMSPASS=ibmuser
DBUSER=wsadmin
DBPASS=wsadmin
CONFIGSERVER=server1
DBLOCATION=LOC1
#####
```

GLOBAL Properties セクションは、共通の値の中央の場所として機能します。共通の値は複数のコンポーネントで使用されるため、応答ファイルの複数の場所に表示されます。 **GLOBAL Properties** セクションは、これらの共通の値のための中央の場所であり、これらの値を編集するのに便利です。例えば、グローバル・プロパティを使用すると、サーバー・プロパティのインスタンスをすべて **server1** に設定でき、変更するプロパティのすべてのインスタンスを応答ファイルで検索せずに済みます。

グローバル・プロパティを使用するには、先頭に \$ 記号を付けたグローバル・キーワードを、後続のプロパティの値として指定します。例えば、`serverName=$CONFIGSERVER` のようになります。

グローバル・プロパティは、応答ファイル内で \$ 記号によって参照される箇所よりも前に出現していなければなりません。

グローバル・プロパティを使用する場合は、プロパティの値全体を指定する必要があります。以下の例は、許可されません。

```
templatePath=/usr/$USERPATH/dir
```

グローバル・プロパティでは、コマンド行の構文の数を削減することにより、製品構成コマンドのオーバーライド引数 (-Z) の使用を簡略化しています。

プロパティのコメントは、プロパティの上に表示され、プロパティ名とそのデフォルト値が太字で示されます。以下にプロパティの例を示します。

```
#####  
#  
# Profile name  
#  
# On z/OS, there is always one and only one profile and that profile is named  
# default in each of the configurations.  
#  
# The profile referred to here is the default profile installed and  
# and configured during the WebSphere Application Server for z/OS install.  
#  
profileName=default
```

関連概念

9 ページの『Network Deployment 構成』

初期 Network Deployment 構成に含まれているデプロイメント・マネージャー・サーバーには、デプロイメント・マネージャーが稼働する z/OS® システムのデーモンが含まれています。Network Deployment セルの作成後に、アプリケーション・サーバー・ノードを追加できます。追加するには、新しい空の管理対象ノードを作成して統合するか、またはスタンドアロン・アプリケーション・サーバー・ノードを Network Deployment セルに統合します。

関連タスク

 Derby データベースでのスタンドアロン構成の作成

スタンドアロン構成は、WebSphere Process Server for z/OS の最も単純な構成タイプです。スタンドアロン構成には、アプリケーション・サーバーを実行する単一ノードと、単一 z/OS® システムまたは LPAR で稼働する 1 つのデーモン・サーバーが含まれています。Derby データベースは、テスト・システムにのみ使用してください。実動システムには DB2 データベースを使用してください。

74 ページの『WebSphere Process Server を使用した Network Deployment 構成の作成』

基本的な Network Deployment 構成では、1 つのノードにデプロイメント・マネージャー・サーバー、もう 1 つのノードにアプリケーション・サーバーがあります。WebSphere Process Server Network Deployment 構成では、両方のノードが WebSphere Process Server 機能によって拡張されます。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合されるため、デプロイメント・マネージャーで管理できます。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合される前に、WebSphere Process Server 機能で拡張されている必要があります。

72 ページの『スタンドアロン構成の作成』

スタンドアロン構成は、WebSphere Process Server for z/OS の最も単純な構成タイプです。スタンドアロン構成には、アプリケーション・サーバーを実行する単一ノードと、単一 z/OS システムまたは LPAR で稼働する 1 つのデーモン・サーバーが含まれています。Derby または DB2 タイプのデータベースを使用するスタンドアロン構成を作成できます。Derby データベースは、テスト・システムにのみ使用してください。実動システムには DB2 データベースを使用してください。DB2 データベースを使用する場合は、応答ファイルを大幅に変更してデータベースをセットアップおよび構成する必要があります。評価やデモの目的でスタンドアロン構成を迅速にセットアップするには、代わりに Derby データベースを使用するのが便利です。

zPMT ツールの使用

zPMT は、z/OS Profile Management Tool の略です。zPMT ツールはワークステーション・ベースのツールで、情報を取り込み、カスタマイズされた JCL バッチ・ジョブと構成応答ファイル (z/OS システムの WebSphere Process Server の作成に使用) を生成します。

zPMT ツールの重要な機能の一部として、以下のタスクを実行する機能があります。

- 情報の取り込み。
- カスタマイズされた JCL バッチ・ジョブと構成応答ファイルの生成。
- z/OS システムへの JCL バッチ・ジョブと構成応答ファイルのアップロード。

zPMT ツールは、Eclipse ベースのツールである WebSphere Customization Tool (WCT) の一部です。zPMT ツールを使用するには、ワークステーションに WCT をインストールして起動し、そこから zPMT を開始する必要があります。WebSphere Customization Tool のインストール方法については、『WebSphere Customization Tool のインストール』を参照してください。

ワークステーションで zPMT ツールを実行し、一連のウィンドウに情報を入力します。入力した情報は、WCT のカスタマイズ・ロケーション (ハード・ディスク上のディレクトリー構造) に保管されます。生成された JCL ジョブと応答ファイル (z/OS サーバー用の WebSphere Process Server の作成に使用) は、このディレクトリー構造に保管されます。

JCL ファイルは、zPMT ツールのプロセス機能を使用して z/OS システムに送信されます。JCL ファイルは、CNTL と DATA という 2 つの区分データ・セットに格納されます。

- CNTL データ・セットには、WebSphere Process Server for z/OS のインストールと構成に使用される生成済み JCL ジョブが格納されます。
- DATA データ・セットには、拡張スクリプトに渡される構成応答ファイルが格納されます。

zPMT ツールを使用すると、データ・セットをアップロードの一部として割り振ることができます。

zPMT ツールを使用するセルの計画プロセスは、ISPF パネルを使用した際と同じです。zPMT ツールにより、ISPF パネルと同じ情報の入力を要求するプロンプトが表示されますが、順序が若干異なる場合もあります。zPMT ツールを使用する命名規則とポートの割り振りチャートは、ISPF パネルを使用した際と同じです。

zPMT ツールによって作成可能な構成のタイプ

zPMT ツールを使用して、3 つのタイプの構成を作成することができます。以下の表に、zPMT ツール用語と ISPF 用語の相互参照を示します。

表 5. zPMT ツールによって使用可能なカスタマイズ環境

zPMT ツールのオプション	対応する ISPF オプション
WebSphere Process Server for z/OS Process Server	スタンドアロンの Process Server ノード
WebSphere Process Server for z/OS のデプロイメント・マネージャー	Network Deployment セル
WebSphere Process Server for z/OS 管理対象 (カスタム) ノード	空の管理対象ノード

zPMT ツールは、ワークスペース内に保存済み構成定義を作成します。アップロード機能により、z/OS システムにファイルを FTP で転送することができます。

zPMT ツールに入力された変数は、ハード・ディスク上のワークスペースの一部として保存されます。このワークスペースは、プロジェクトに関するすべての情報をツールが保持するエリアです。構成に対して入力した変数は、zPMT ツールの応答ファイルに登録されます。このファイルは、変数とユーザーの値を保持するフラット・ファイルです。この応答ファイルは、ISPF ダイアログの SAVECFG ファイルの概念と似ていますが、形式が異なります。

zPMT ツールの応答ファイルは、拡張に使用される応答ファイルとは異なります。ジョブをカスタマイズする場合は、別の変数 (例えば、ジョブを構成するための JCL ヘッダーの 4 つの変数) を使用します。拡張に使用される応答ファイルは、「プロセス」をクリックして取得します。このファイルは、カスタマイズしたジョブを z/OS システムまたはローカル・ディレクトリーにアップロードする際にも使用されます。

この応答ファイルの名前は、BPZRSPX の名前形式に従っています。この「X」は、以下のノードによって値が異なります。

- A = スタンドアロン・ノード。
- M = デプロイメント・マネージャー・ノード。
- N = 管理対象ノード。

第 3 章 WebSphere Process Serverの構成

WebSphere Process Server をインストールしたら、追加の構成タスクを完了して、ランタイム環境を完全に準備する必要があります。

関連情報

 PDF 資料

WebSphere Process Server 資料 (PDF 形式)

 情報ロードマップ

IBM が提供する Business Process Management 情報ロードマップには、WebSphere Process Server、WebSphere ESB、およびその他の製品に関する情報がポートフォリオに編成されています。

 IBM Education Assistant

IBM Education Assistant により提供される WebSphere Process Server に関するマルチメディア教育モジュール。

 技術情報

WebSphere Process Server サポート > インストール。WebSphere Process Server 製品のインストールについて不明な点がある場合は、このリソースを参照して、製品のインストールからセットアップまで行うことができます。

 概説

製品ライブラリー Web ページの「Overview」タブ。このページを使用して、WebSphere ESB に関連した発表、データ・シート、およびその他の一般ライブラリー資料にアクセスします。

WebSphere Process Server for z/OSの構成

構成スクリプトは、ノードを拡張して WebSphere Process Server for z/OS 機能のサポートを追加します。ご使用の WebSphere Process Server for z/OS セルのトポロジーを決定し、WebSphere Process Server for z/OS 機能を実行するすべてのノードを定義して拡張する必要があります。

このタスクについて

各種パラメーターの値は、構成シェル・スクリプトに対する入力として渡す応答ファイルで設定します。応答ファイルを作成する方法は 2 つあり、zPMT ツールを使用して応答ファイルを生成することも、製品に付属するサンプルの応答ファイルを編集することもできます。『共通構成の作成』で示す上位の作業のリストでは、各種 WebSphere Process Server トポロジーの計画および実装について順を追って説明しています。

DB2 データベース・オブジェクトの設計

WebSphere Process Server インスタンスのサポートに必要な DB2 for z/OS オブジェクトを編成するには、さまざまな方法があります。どの方法を選ぶかは、組織の標準や規則、または DB2 サブシステムを管理しやすくするための慣例によって決まる場合があります。DB2 サブシステム管理者と協力して、データベース、ストレージ・グループ、バッファ・プールなどの DB2 リソースの命名規則と使用方法に関する事柄を設計し、DbDesignGenerator.sh ツールを使用して、そのデータベース設計を実装してください。

DbDesignGenerator.sh ツール

WebSphere Process Server データベース・オブジェクトのセットに対してデータベース構成値を指定するには、DbDesignGenerator.sh ツールを使用します。

DbDesignGenerator.sh ツールを使用すると、以下の 2 つのものが生成されます。

1. DbDesignGenerator.sh ツールに入力した値の格納に使用されるデータベース設計ファイル。このファイルを拡張応答ファイルの `wbidbDesign` パラメーターで参照して、WebSphere Process Server リソース定義の属性値を指定できます。拡張応答ファイルの作成方法については、63 ページの『応答ファイルの作成』を参照してください。
2. 各種の WebSphere Process Server コンポーネントの DB2 データベース・オブジェクトを作成するために実行する必要がある DDL。DDL を実行してデータベース・オブジェクトを作成する方法については、80 ページの『データベースの作成および構成』を参照してください。

DB2 の決定

WebSphere Process Server for z/OS を構成するときに、いくつかの決定を行う必要があります。

データベースの命名規則の決定:

WebSphere Process Server for z/OS が構成されているサーバー 1 つのみが DB2 サブシステムを使用する場合は、サンプルの応答ファイルに指定されているデフォルトのデータベース名を使用できます。WebSphere Process Server for z/OS が構成されている複数のサーバー (同一セル内または異なるセル内) が同じ DB2 サブシステムまたは同じデータ共有グループを使用する場合は、以下の DB2 コンポーネントを切り分けるための命名規則を計画する必要があります。

- データベース名
- ストレージ・グループ名
- テーブルのスキーマ修飾子
- VCAT。VSAM のカタログ名を表します。VCAT は、DB2 サブシステムの DB2 のテーブル・スペースとテーブルにプレフィックスを付けるために使用される高位修飾子です。
- 上記データ・セットのボリュームまたは SMS ストレージ・グループ

以下の例は、セル名が `S5CELL` の場合の (1 つのデータベースの) データベース命名規則を示したものです。

Database	Dbase Name	Storage Group	Schema -Owner	VCAT DSN-h1q
-----	-----	-----	-----	-----
WPS:	S5CELLDB	S5DBSTO	S5CELL	DSN810PP
BPC:	S5CELLDB	S5DBSTO	S5CELL	DSN810PP
APP SIB:	S5CELLDB	S5DBSTO	S5S1A	DSN810PP
BPC SIB:	S5CELLDB	S5DBSTO	S5S1B	DSN810PP
CEI SIB:	S5CELLDB	S5DBSTO	S5S1C	DSN810PP
SCA SIB:	S5CELLDB	S5DBSTO	S5S1S	DSN810PP
CEI:	S5CELLDB	S5DBSTO	S5CELL	DSN810PP
BSPACE:	S5CELLDB	S5DBSTO	S5CELL	DSN810PP

以下の例は、WebSphere Application Server セル名が B6CELL の場合の (9 つのデータベースの) データベース命名規則を示したものです。

Database	Dbase Name	Storage Group	Schema -Owner	VCAT DSN-h1q
-----	-----	-----	-----	-----
WPS:	B6WPSDB	B6WPSSTO	B6CELL	B6WPS
BPE:	B6BPEDB	B6PESTO	B6CELL	B6WPS
SIBs:	B6SIBAPP	B6SIBSTO	B6CELLA	B6WPS
	B6SIBSCA	B6SIBSTO	B6CELLS	B6WPS
	B6SIBBPC	B6SIBSTO	B6CELLB	B6WPS
	B6SIBCEI	B6SIBSTO	B6CELLC	B6WPS
CEI:	B6EVTDB	B6EVTSTO	B6CELL	B6WPS
BSPACE:	B6BSPACE	B6BSPSTO	B6CELL	B6WPS

スキーマ名および SQL ID の決定:

DB2 テーブルでは、スキーマ名および SQL ID に一意の名前を選択することが重要です。

DB2 テーブルが同一 DB2 サブシステム内でほかの WebSphere Process Server for z/OS セルまたは WebSphere Business Integration Server セルと共存するには、その DB2 テーブルの修飾スキーマ名が一意でなければなりません。管理コンソールを使用して、「データ・ソース」>「カスタム・プロパティ」の定義内で現在のスキーマまたは SQL ID 値を設定できます。

これらの名前を DDL 定義内で DB2 テーブル名および索引名のプレフィックスとして付けたり、テーブルの作成に使用する DDL の前に SET CURRENT SQLID ステートメントを挿入したりできます。以下に例を示します。

```
SET CURRENT SQLID = 'B6CELL';
```

SIB データベースでは、すべてのバスに同じテーブル名が使用されるため、それぞれのテーブルに一意のスキーマ名が必要です。これと同じスキーマ名を SIB のメッセージング・エンジンのデータ・ストア・プロパティにも設定する必要があります。WebSphere Application Server for z/OS インフォメーション・センターの『メッセージング・エンジン』を参照してください。

データベース設計ツールを使用したデータベース設計ファイルの作成

データベース設計ツール (DDT) を使用すると、WebSphere Process Server に必要なデータベース表の作成に使用する設計ファイルを生成できます。DDT では、ユーザー指定のプロパティ・ファイルまたは対話式のユーザー入力に基づいて設計ファイルが生成されます。生成された設計ファイルは、DDT がデータベース表の作成に使用されるデータベース・スクリプトを作成するのに使用します。また、プロ

ファイル作成時やデプロイメント環境構成時の入力として設計ファイルを使用し、データベース構成プロパティを指定することもできます。

始める前に

すべてのデータベース要件およびスキーマ名のリストを用意する必要があります。

このタスクについて

以下のステップでは、DDT を使用して設計ファイルおよびデータベース・スクリプトを生成する方法について説明します。DDT の入力は、ユーザー指定のプロパティ・ファイルまたは対話式ユーザー入力のいずれかです。

DbDesignGenerator コマンドのオプションは以下のとおりです。

-? , -help
ヘルプ情報を表示します。

-e db_design_file_name
指定したデータベース設計ファイル (*.dbDesign、*.properties など) を編集します。

-v db_design_file | db_scripts_output_directory
db_design_file が指定されている場合は、データベースの仕様に基づいて、指定されたデータベース設計ファイルに対して検証が行われます。
db_scripts_output_directory が指定されている場合は、指定されたディレクトリ内のデータベース・スクリプトが検証されます。現在のところ、検証が可能なのは、

テンプレート ddl 生成プログラムから生成されたスクリプトのみです。

-g db_design_file [-d output_directory] [db_design_file2] [-d output_directory2] ... [db_design_fileN] [-d output_directoryN]
指定された設計ファイルからデータベース・スクリプトをバッチ・モードで生成します。生成されたスクリプトは、対応する出力ディレクトリに格納されるか、出力ディレクトリがない場合はデフォルトのロケーションに格納されます。

注: DDT は、Common Event Infrastructure 用のデータベース・スクリプトの生成をサポートしません。

注: 以下の制約事項は、Common Event Infrastructure コンポーネントのスタンドアロン・データベース設計に適用されます。

表 6. CEI コンポーネントのスタンドアロン・データベース設計の制約事項:

データベース・タイプ	CEI 制約事項
DB2 Distributed	データベース名は、commonDB 名と同じであってはなりません。CEI データベース設計を編集し、別の名前を選択してください。

表 6. CEI コンポーネントのスタンドアロン・データベース設計の制約事項: (続き)

データベース・タイプ	CEI 制約事項
SQL サーバー	<ul style="list-style-type: none"> • dbServerName は空であってはなりません。CEI データベース設計を編集し、データベース・サーバー名を指定してください。 • dbUser は、CommonDB ユーザー ID と同じにすることはできません。CEI データベース設計を編集し、別のユーザー ID を指定してください。 • sysUser および sysPassword は空であってはなりません。CEI データベース設計を編集し、システム・ユーザー ID およびシステム・パスワードを指定してください。
Oracle	<ul style="list-style-type: none"> • dbUser は、CommonDB ユーザー ID と同じにすることはできません。CEI データベース設計を編集し、別のユーザー ID を指定してください。 • sysUser および sysPassword は空であってはなりません。CEI データベース設計を編集し、システム・ユーザー ID およびシステム・パスワードを指定してください。

手順

以下のように **DbDesignGenerator** コマンドを使用して、設計ファイルおよびデータベース・スクリプトを生成します。このコマンドは、以下にあります。

```
DbDesignGenerator.bat - for Windows
DbDesignGenerator.sh - for Unix and z/OS
```

以下のメインメニューに戻ります

[通知] 対話モードで DbDesignGenerator を実行中...

[通知] 保存しないで終了する場合は「q」、直前のメニューに戻るには「-」、ヘルプを随時表示するには「?」を入力してください。

(Enter 'q' to quit without saving; '-' for back to previous menu; '?' for help at any time.)

[通知] 提供されたデフォルト値を受け入れるには、単に「Enter」キーを押します。

[通知]

以下のいずれかの [設計オプション] を選んでください

(Please pick one of the following [design option(s)]):

- (1) スタンドアロン・プロファイルまたはデプロイメント環境のデータベース設計の作成
- (2) 単一コンポーネントのデータベース設計の作成 (例: BPC、CEI など)
- (3) 既存のデータベース設計の編集
- (4) データベース設計からのデータベース・スクリプトの生成
- (5) 終了 [q]

応答ファイルの作成

システムの構成方法の詳細がすべて含まれている応答ファイルは、2 つの方法で作成できます。zPMT ツールを使用して応答ファイルを生成することも、製品に付属するサンプルの応答ファイルを編集することもできます。詳しくは、64 ページの

『zPMT ツールを使用した応答ファイルの作成』および『サンプルを使用した応答ファイルの作成』を参照してください。

zPMT ツールを使用した応答ファイルの作成

zPMT は z/OS プロファイル管理ツールです。zPMT ツールはワークステーション・ベースのツールで、情報を取り込み、カスタマイズされた JCL バッチ・ジョブと構成応答ファイル (z/OS システムの WebSphere Process Server の作成に使用) を生成します。

既に zPMT を使用して応答ファイルを作成している場合もあります。zPMT を使用してインストール・ジョブを生成した場合は、このプロセスによって応答ファイルが生成されています。バッチ・ジョブとしてではなく USS コマンドからインストールを実行した場合は、WebSphere Process Server 構成定義および関連する拡張応答ファイルの作成に zPMT を使用していない可能性があります。

zPMT を使用して応答ファイルを作成する場合の説明については、12 ページの『zPMT ツールの使用』を参照してください。また、付属のサンプル応答ファイルのいずれかをカスタマイズする方法については、『サンプルを使用した応答ファイルの作成』を参照してください。

サンプルを使用した応答ファイルの作成

WebSphere Process Server for z/OSには、4 つのサンプル応答ファイルが付属しており、これらのファイルを使用して、ご使用のシステムの構成方法の詳細すべてを記述した応答ファイルを作成できます。

始める前に

- WebSphere Application Server をインストールして構成し、WebSphere Process Server 構成のベースとして使用します。

このタスクについて

WebSphere Process Server 構成の作成に使用できる応答ファイルを作成することができます。作成する応答ファイルは、次の 4 つのサンプル応答ファイルのいずれかに基づきます。

standAloneProfile.rsp

Derby™ データベースを使用してスタンドアロン構成を作成します。

standAloneProfileDB2.rsp

DB2® データベースを使用してスタンドアロン構成を作成します。

DmgrDB2.rsp

DB2 データベースを使用して Network Deployment 構成を備えたデプロイメント・マネージャー・サーバーを作成します。

ManagedDB2.rsp

DB2 データベースを使用して Network Deployment 構成を備えたノードを作成します。

これらのサンプル応答ファイルに含まれている、ご使用の構成のニーズに合わせて調整可能なキーワードについては、『応答ファイルの値』を参照してください。

Derby データベースを使用するスタンドアロン構成の応答ファイルの作成:

Derby データベースを使用するスタンドアロン構成で WebSphere Process Server for z/OS を構成するときに使用する応答ファイルを作成できます。スタンドアロン構成は、WebSphere Process Server for z/OS の最も単純な構成タイプです。スタンドアロン構成には、アプリケーション・サーバーを実行する単一ノードと、単一 z/OS® システムまたは LPAR で稼働する 1 つのデーモン・サーバーが含まれています。Derby データベースは、テスト・システムにのみ使用してください。実動システムには DB2® データベースを使用してください。

始める前に

Derby データベースを使用する WebSphere Process Server のスタンドアロン構成の応答ファイルを作成できるようにするには、まず、以下のタスクを実行する必要があります。

- WebSphere Process Server のインストール

手順

1. 構成スクリプト `zWPSConfig.sh` への入力を提供する応答ファイルを作成します。Derby データベースを使用するスタンドアロン・サーバー用のサンプル応答ファイルが、`/usr/lpp/zWPS/V7R0/zos.config/standAloneProfile.rsp` にあります。
 - a. サンプル応答ファイル `standAloneProfile.rsp` を作業ディレクトリーにコピーします。以下に例を示します。

```
cp /usr/lpp/zWPS/V7R0/zos.config/standAloneProfile.rsp /u/work/
```
 - b. **chmod** コマンドを使用して、応答ファイルのコピーに適切な許可を割り当てます。以下に例を示します。

```
chmod 755 standAloneProfile.rsp
```
 - c. 使用している構成に合わせて、応答ファイル内のパラメーターを編集します。
 - d. 編集した応答ファイルを保存します。
2. サーバーが開始済みである場合は停止します。
3. 製品構成スクリプトを最後まで実行させるために、OMVS の時間制限を延長します。MVS コンソールで、次のコマンドを入力します。

```
SETOMVS MAXCPU=86400
```
4. USS コマンド・シェルにアクセスしてから、管理者ユーザー ID に切り替えます。以下に例を示します。

```
su wsadmin
```
5. 次のように入力して、ディレクトリーをアプリケーション・サーバーの `bin` ディレクトリーに変更します。

```
cd /WebSphere/V7R0/AppServer/bin
```

タスクの結果

WebSphere Process Server のインストール環境を構成する準備ができました。詳しくは、69 ページの『共通構成の作成』を参照してください。

DB2 データベースを使用するスタンドアロン構成の応答ファイルの作成:

DB2 データベースを使用するスタンドアロン構成で WebSphere Process Server for z/OS を構成するときに使用する応答ファイルを作成できます。スタンドアロン構成には、アプリケーション・サーバーを実行する単一ノードと、単一 z/OS® システムまたは LPAR で稼働する 1 つのデーモン・サーバーが含まれています。DB2 データベースを使用する場合は、応答ファイルを大幅に変更し、データベースをセットアップおよび構成する必要があります。評価やデモの目的でスタンドアロン構成を迅速にセットアップするには、代わりに Derby データベースを使用するのが便利です。

手順

1. 次のように、DB2JccConfiguration.properties ファイルを作成およびカスタマイズして、WebSphere Application Server に対して DB2 サブシステムに関する情報を提供します。
 - a. このファイルを、適切なディレクトリーに作成します。例: /etc/db2cfg
 - b. WebSphere 制御領域ユーザー ID、および WebSphere Process Server 構成ジョブを実行するユーザー ID がプロパティー・ファイルを読み取れるよう、上記のディレクトリーに対するアクセス権を設定します。
 - c. DB2JccConfiguration.properties ファイルに次の行が含まれていることを確認します。

```
db2.jcc.ssid=DB15
```

ここで、*dbn* はインストールの SSID です (例: DB15)。それ以外の、DB2JccConfiguration.properties ファイル内で変更できるすべてのプロパティーの詳細については、「DB2 for z/OS アプリケーション・プログラミング ガイドおよびリファレンス (Java 用) バージョン 8.1」(SC88-9807) を参照してください。

2. 構成スクリプト zWPSConfig.sh への入力を提供する応答ファイルを作成します。DB2 を使用したスタンドアロン・サーバー用のサンプル応答ファイルが `usr/lpp/zWPS/V7R0/zos.config/standaloneProfileDB2.rsp` にあります。
 - a. サンプルの応答ファイル `standaloneProfileDB2.rsp` を作業ディレクトリーにコピーします。以下に例を示します。

```
cp /usr/lpp/zWPS/V7R0/zos.config/standaloneProfileDB2.rsp /u/work/
```
 - b. **chmod** コマンドを使用して、応答ファイルのコピーに適切な許可を割り当てます。以下に例を示します。

```
chmod 755 standaloneProfileDB2.rsp
```
 - c. `-dbDelayConfig` パラメーターを `true` に設定し、構成スクリプトによって、データベース・オブジェクトを作成する DDL スクリプトが自動的に実行されないようにします。DDL スクリプトは、後で実行します。
 - d. 応答ファイルのその他のパラメーターを、ご使用の構成に合わせて編集します。
 - e. 編集した応答ファイルを保存します。

タスクの結果

WebSphere Process Server のインストール環境を構成する準備ができました。詳しくは、69 ページの『共通構成の作成』を参照してください。

Network Deployment マネージャー構成の応答ファイルの作成:

基本的な Network Deployment 構成では、1 つのノードにデプロイメント・マネージャー・サーバー、もう 1 つのノードにアプリケーション・サーバーがあります。WebSphere Process Server Network Deployment 構成では、両方のノードが WebSphere Process Server 機能によって拡張されます。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合されるため、デプロイメント・マネージャーで管理できます。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合される前に、WebSphere Process Server 機能で拡張されている必要があります。

手順

1. 次のように、DB2JccConfiguration.properties ファイルを作成およびカスタマイズして、WebSphere Application Server に対して DB2 サブシステムに関する情報を提供します。
 - a. このファイルを、適切なディレクトリーに作成します。例: /etc/db2cfg ディレクトリー。
 - b. WebSphere Controller Region のユーザー ID と WebSphere Process Server 構成ジョブを実行するユーザー ID がプロパティー・ファイルを読み取ることができるように、ディレクトリーに対する許可を設定します。
 - c. DB2JccConfiguration.properties ファイルに次の行が含まれていることを確認します。

```
db2.jcc.ssid=DB15
```

ここで、DB15 はインストールの SSID です。それ以外の、DB2JccConfiguration.properties ファイル内で変更できるすべてのプロパティーの詳細については、「DB2 for z/OS アプリケーション・プログラミング ガイドおよびリファレンス (Java 用) バージョン 8.1」(SC88-9807) を参照してください。

2. 構成スクリプト zWPSConfig.sh への入力を提供する応答ファイルを作成します。DB2 を使用したスタンドアロン・サーバー用のサンプルの応答ファイルが /usr/lpp/zWPS/V7R0/zos.config/DmgrDB2.rsp にあります。
 - a. サンプルの応答ファイル DmgrDB2.rsp を作業ディレクトリーにコピーします。以下に例を示します。

```
cp /usr/lpp/zWPS/V7R0/zos.config/DmgrDB2.rsp /u/work/
```
 - b. **chmod** コマンドを使用して、応答ファイルのコピーに適切な許可を割り当てます。以下に例を示します。

```
chmod 755 DmgrDB2.rsp
```
 - c. **-dbDelayConfig** パラメーターを **true** に設定し、構成スクリプトによって、データベース・オブジェクトを作成する DDL スクリプトが自動的に実行されないようにします。DDL スクリプトは、後で実行します。
 - d. 使用するシステムに合わせて、応答のその他のパラメーターを編集します。

- e. 編集した応答ファイルを保存します。

次のタスク

WebSphere Process Server のインストール環境を構成する準備ができました。詳しくは、69 ページの『共通構成の作成』を参照してください。

空のノードの応答ファイルを作成します。:

基本的な Network Deployment 構成には、デプロイメント・マネージャー・サーバーが 1 つのノードに含まれると共に、アプリケーション・サーバーを作成できる別の管理対象ノードが含まれています。WebSphere Process Server Network Deployment 構成では、両方のノードが WebSphere Process Server 機能によって拡張されます。その後、管理対象ノードはデプロイメント・マネージャー・セルに統合されます。これにより、デプロイメント・マネージャーでアプリケーション・サーバーを作成し、管理できるようになります。管理対象ノードは、デプロイメント・マネージャー・セルに統合される前に、WebSphere Process Server 機能で拡張されている必要があります。

手順

1. 次のように、DB2JccConfiguration.properties ファイルを作成およびカスタマイズして、WebSphere Application Server に対して DB2 サブシステムに関する情報を提供します。
 - a. このファイルを、適切なディレクトリーに作成します。例: /etc/db2cfg ディレクトリー。
 - b. WebSphere Controller Region のユーザー ID と WebSphere Process Server 構成ジョブを実行するユーザー ID がプロパティ・ファイルを読み取ることができるように、ディレクトリーに対する許可を設定します。
 - c. DB2JccConfiguration.properties ファイルに次の行が含まれていることを確認します。

```
db2.jcc.ssid=DB15
```

ここで、DB15 はインストールの SSID です。それ以外の、DB2JccConfiguration.properties ファイル内で変更できるすべてのプロパティの詳細については、「*DB2 for z/OS アプリケーション・プログラミング ガイドおよびリファレンス (Java 用) バージョン 8.1*」(SC88-9807) を参照してください。

2. 構成スクリプト zWPSConfig.sh への入力を提供する応答ファイルを作成します。空の管理対象ノード用のサンプル応答ファイルが /usr/lpp/zWPS/V7R0/zos.config/ManagedDB2.rsp にあります。
 - a. サンプルの応答ファイル ManagedDB2.rsp を作業ディレクトリーにコピーします。以下に例を示します。

```
cp /usr/lpp/zWPS/V7R0/zos.config/ManagedDB2.rsp /u/work/
```
 - b. **chmod** コマンドを使用して、応答ファイルのコピーに適切な許可を割り当てます。以下に例を示します。

```
chmod 755 ManagedDB2.rsp
```
 - c. 応答ファイルのその他のパラメーターを、ご使用の構成に合わせて編集します。45 ページの『応答ファイルの値』を参照してください。

- d. 編集した応答ファイルを保存します。
3. 製品構成スクリプトを最後まで実行させるために、OMVS の時間制限を延長します。MVS コンソールで、次のコマンドを入力します。

```
SETOMVS MAXCPUIME=86400
```
4. USS コマンド・シェルにアクセスしてから、管理者ユーザー ID に切り替えます。以下に例を示します。

```
su wsadmin
```
5. 次のように入力して、ディレクトリーをアプリケーション・サーバーの bin ディレクトリーに変更します。

```
cd /WebSphere/V7R0/AppServer/bin
```

次のタスク

WebSphere Process Server のインストール環境を構成する準備ができました。詳しくは、『共通構成の作成』を参照してください。

共通構成の作成

このセクションでは、WebSphere Process Server for z/OS で作成可能な共通の構成のいくつかについて説明します。

WebSphere Process Server を使用した拡張

WebSphere Process Server コードを WebSphere Application Server 構成にインストールしたら、既存の WebSphere Application Server プロファイルを WebSphere Process Server プロファイルに拡張して、WebSphere Process Server 機能を使用可能にする必要があります。

WebSphere Process Server for z/OS のサーバー構成は、WebSphere Application Server for z/OS プロファイル・ディレクトリーの下ディレクトリー構造 (/WebSphere/V7R0/AppServer/profiles/default または /WebSphere/V7R0/DeploymentManager/profiles/default) に格納されます。このディレクトリーには、サーバーに固有のディレクトリーとファイルがすべて格納されます。このディレクトリーは、サーバー・ログと FFDC ログの格納にも使用されます。

WebSphere Application Server for z/OS では、すべてのランタイム環境が default というプロファイル名を使用して作成されます。manageprofiles コマンド、および他の管理コマンドでの -profile オプションは、WebSphere Application Server for z/OS では使用されません。

始める前に

WebSphere Process Server の実装をサポートするために使用するデータベースを決定します。使用するデータベースの選択は、実装トポロジーによって異なります。

- Network Deployment 環境を構成する場合は、DB2 データベースを使用する必要があります。
- スタンドアロン・サーバーを構成する場合は、Derby または DB2 のいずれかのデータベースを使用できます。
- クラスタリングを構成する場合は、DB2 z/OS システムがデータ共用モードで実行されている必要があります。

これが WebSphere Process Server for z/OS の最初のインストールである場合は、まず、Derby データベースを使用するスタンドアロン構成を作成してください。Derby データベースを使用するスタンドアロン構成をインストールした後で、DB2 データベースを使用するスタンドアロン・サーバーを構成し、DB2 の使用に慣れていくことができます。その後、より複雑な手順である、DB2 タイプのデータベースを使用するネットワーク・デプロイメント構成を作成できます。

構成の計画に zPMT を使用している場合は、拡張を実行するために一連の JCL ジョブが生成されています。詳しくは、71 ページの『zPMT で生成された拡張ジョブによるプロファイルの拡張』を参照するか、または zPMT によって生成される説明を参照してください。

zPMT を使用していない場合は、まず、拡張プロセスで使用されるプロファイル固有のパラメーターが含まれた応答ファイルを構成 (または既存のファイルやサンプル・ファイルを変更) して、システムを構成する必要があります。このプロセスについて詳しくは、72 ページの『スタンドアロン構成の作成』または 74 ページの『WebSphere Process Server を使用した Network Deployment 構成の作成』を参照してください。応答ファイルがある場合は、WebSphere Process Server によって提供された USS スクリプトを使用して、拡張を実行する必要があります。詳しくは、『USS コマンドを使用したプロファイルの拡張』を参照してください。

USS コマンドを使用したプロファイルの拡張:

拡張は、拡張テンプレートで既存のプロファイルを変更する機能です。既存の WebSphere Application Server プロファイルまたは WebSphere Application Server Network Deployment プロファイルを WebSphere Process Server プロファイルに拡張できます。

始める前に

この手順を実行する前に、以下の作業が完了していることを確認してください。

- プロファイルを作成または拡張するための前提条件を確認してください。
- 拡張するプロファイルに関連付けられたすべてのサーバーをシャットダウンしてください。
- 拡張するプロファイルのタイプを確認してください。
- 応答ファイルを作成してください。

このタスクについて

構成するプロファイルのタイプに応じて zWPSConfig.sh スクリプトまたは zWESBConfig.sh スクリプトを使用して、USS からプロファイルを拡張できます。USS からのプロファイルの拡張を実行するには、以下のステップを実行します。

手順

1. サーバーの停止方法
2. 製品構成スクリプトを最後まで実行させるために、OMVS の時間制限を延長します。MVS コンソールで、次のコマンドを入力します。

```
SETOMVS MAXCPUTIME=86400
```

3. USS コマンド・シェルにアクセスし、管理者ユーザー ID に切り替えます。以下に例を示します。

```
su wsadmin
```

4. 次のように入力して、ディレクトリーをサーバーの bin ディレクトリーに変更します。以下に例を示します。

```
cd /WebSphere/V7R0/AppServer/bin
```

または

```
cd /WebSphere/V7R0/DeploymentManager/bin
```

5. zWPSConfig.sh 構成スクリプトを、編集した応答ファイルの絶対パスを使用して実行します。以下に例を示します。

```
zWPSConfig.sh -augment -response /working_directory_path/standAloneProfileDB2.rsp
```

これにより、DB2 データベース用に拡張されます。ここで、`working_directory_path` は、編集した `standAloneProfileDB2.rsp` ファイルのロケーションです。コマンドについて詳しくは、『zWPSConfig.sh スクリプト』を参照してください。

6. 構成が実行されるのを待ちます。

次のタスク

スクリプトの実行が完了したら、コンソールに書き込まれたメッセージを確認します。以下のメッセージが返された場合は、プロファイルの拡張が正常に完了したことになります。

INSTCONFSUCCESS: プロファイルの拡張は正常に終了しました (Profile augmentation succeeded)。

以下のログ・ファイルを確認できます。

```
install_root/logs/manageprofiles/profile_name_augment.log
```

zPMT で生成された拡張ジョブによるプロファイルの拡張:

WebSphere Process Server zPMT のカスタマイズ定義が処理されると、既存の WebSphere Application Server 構成のインストールおよび WebSphere Process Server 構成への拡張を制御するために、いくつかの JCL ジョブが生成されます。このトピックでは、このプロセスの拡張フェーズについて説明します。zPMT カスタマイズ定義の「カスタマイズの指示」タブでも、詳細なカスタマイズの指示が示されます。

このタスクについて

zPMT で生成されたジョブを使用してプロファイルを拡張する前に、以下のタスクを完了しておく必要があります。

- zPMT で構成計画を完了し、カスタマイズ定義を処理して、その結果として生成されたデータ・セット・メンバーをターゲットの z/OS システムにアップロードする。
- ターゲットの WebSphere Application Server 構成が WebSphere Application Server を使用して完全に構成されており、WebSphere Process Server コードを使用して

インストールされていることを確認する。詳しくは、21 ページの『WebSphere Process Server のインストール』を参照してください。

手順

1. ターゲットの構成を停止し、バックアップを作成します。
2. zPMT によってアップロードされたカスタマイズ・ジョブのロケーションにナビゲートします。
3. WebSphere Process Server の機能を使用して、ターゲットの WebSphere Application Server プロファイルを拡張するのに必要な拡張ジョブを実行します。実行するジョブの名前は、zPMT カスタマイズ定義の「カスタマイズの指示」タブに表示されます。
 - a. これを 1 つのジョブで実行したい場合は、「MVS 単一ジョブの拡張ステップ (MVS Single-job Augmentation Steps)」の指示に従ってください。
 - b. これを一連のジョブで実行したい場合は、「MVS ステップ形式の拡張ステップ (MVS Step by Step Augmentation Steps)」の指示に従ってください。
4. ターゲットの構成を再始動します。

タスクの結果

これで、ターゲットの WebSphere Application Server 構成が WebSphere Process Server 構成に正常に拡張されました。

スタンドアロン構成の作成

スタンドアロン構成は、WebSphere Process Server for z/OS の最も単純な構成タイプです。スタンドアロン構成には、アプリケーション・サーバーを実行する単一ノードと、単一 z/OS システムまたは LPAR で稼働する 1 つのデーモン・サーバーが含まれています。Derby または DB2 タイプのデータベースを使用するスタンドアロン構成を作成できます。Derby データベースは、テスト・システムにのみ使用してください。実動システムには DB2 データベースを使用してください。DB2 データベースを使用する場合は、応答ファイルを大幅に変更してデータベースをセットアップおよび構成する必要があります。評価やデモの目的でスタンドアロン構成を迅速にセットアップするには、代わりに Derby データベースを使用するのが便利です。

始める前に

WebSphere Process Server を構成するには、事前に以下の作業を完了しておく必要があります。

- WebSphere Process Server をインストールします。詳しくは、『WebSphere Process Server のインストール』を参照してください。
- サンプル・ファイルまたは zPMT ツールを使用して、Derby または DB2 データベースを使用するスタンドアロン構成用の応答ファイルを作成します。

手順

1. サーバーを停止します。
2. コマンド行から zWPSCfg.sh スクリプトを実行します。以下に例を示します。

```
install_root/zWPSConfig.sh -augment -response responseFile
```

コマンドにより、実行につれて状況が表示されます。コマンドが終了するまで待機します。

3. USS コマンドを使用するか、zPMT によって生成された JCL を使用して、拡張ジョブを実行します。詳しくは、70 ページの『USS コマンドを使用したプロファイルの拡張』または 71 ページの『zPMT で生成された拡張ジョブによるプロファイルの拡張』を参照してください。
4. 構成が実行されるのを待ちます。

スクリプトの実行が終了したら、コンソールに表示されるメッセージを確認します。スクリプトが正常に実行された場合、エラー・メッセージは表示されず、「プロファイルの拡張が完了しました (augmenting profile(s) complete)」という情報メッセージが表示されます。

5. サーバーを始動します。詳しくは、『スタンドアロン・サーバーの開始』を参照してください。
6. 以下の手順で、管理コンソールに WebSphere Process Server の情報が表示されることを確認します。
 - a. ブラウザー・ウィンドウを開き、表示するサーバーの URL を入力して管理コンソールを開きます。以下に例を示します。

```
http://server_name.domain_name:port_number/admin
```
 - b. 管理コンソールにログインします。
 - c. ウェルカム・ページに WebSphere Process Server が表示されることを確認します。WebSphere Process Server をクリックして詳細を表示することもできます。
 - d. 管理コンソール内を移動して、アプリケーションおよびメッセージング・エンジンが開始済みであることを確認します。
7. スタンドアロン・サーバー構成が入っているデータ・セットをバックアップします。

タスクの結果

Derby データベースを使用する構成で、スタンドアロン構成の応答ファイルに Business Process Choreographer のサンプル構成の値を指定した場合は、ビジネス・プロセス・コンテナ、ヒューマン・タスク・コンテナ、および Business Process Choreographer Explorer が含まれるサンプル構成が作成されています。このサンプル構成は、Process Server 構成に含まれています。管理コンソールを調べて、BPEContainer、BPCEXplorer、および TaskContainer で始まる名前のエンタープライズ・アプリケーションを探すことによって、これらのコンポーネントが構成されていることを確認することができます。

Derby サンプル構成は、実動システムには適していません。Business Process Choreographer の構成は、1 つに限られるため、Business Process Choreographer の構成を続行するには、「Business Process Choreographer 構成の除去」での説明に従って、サンプル構成を削除する必要があります。

DB2 データベースを使用する構成の場合は、WebSphere Process Server 機能を使用してスタンドアロン・サーバーが拡張され、データベース・オブジェクトを作成す

るために実行する必要がある DDL が生成されています。

次のタスク

Derby データベースを使用する構成の場合は、この時点でアプリケーションをスタンドアロン・サーバーにデプロイできます。

DB2 データベースを使用する構成の場合は、この時点で DDL スクリプトを実行して、DB2 データベース・オブジェクトを作成する必要があります。『DB2 データベース・オブジェクトの作成および構成』を参照してください。詳しくは、80 ページの『データベースの作成および構成』を参照してください。

関連資料

45 ページの『応答ファイルの値』

WebSphere Process Server for z/OS 製品定義をインストールするインストール・ジョブを実行すると、サンプル応答ファイルがインストール・ファイル・システムにインストールされます。これらの応答ファイルをコピーし、実現する構成に従って編集し、拡張スクリプトの実行時にその応答ファイルを渡します。

 zWPSConfig.sh スクリプト

インストールした WebSphere Process Server for z/OS を構成および拡張するとき使用するスクリプトです。デプロイメント・マネージャーを含め、構成内の各ノードでスクリプトを実行します。

 WebSphere Process Server をデプロイメント・マネージャー・ノードへインストールするための JCL

 デプロイメント・マネージャー・ノードを拡張するための JCL

 WebSphere Process Server を空のノードへインストールするための JCL

 空のノードを拡張するための JCL

 zWPSInstall.sh スクリプト

zWPSInstall.sh スクリプトは、WebSphere Application Server プロファイルを変更して WebSphere Process Server をインストールまたはアンインストールするために使用します。デプロイメント・マネージャーを含め、構成内の各ノードでスクリプトを実行します。

WebSphere Process Server を使用した Network Deployment 構成の作成

基本的な Network Deployment 構成では、1 つのノードにデプロイメント・マネージャー・サーバー、もう 1 つのノードにアプリケーション・サーバーがあります。WebSphere Process Server Network Deployment 構成では、両方のノードが WebSphere Process Server 機能によって拡張されます。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合されるため、デプロイメント・マネージャーで管理できます。アプリケーション・サーバーは、デプロイメント・マネージャー・セルに統合される前に、WebSphere Process Server 機能で拡張されている必要があります。

始める前に

ネットワーク・デプロイメント構成を作成する前に、以下の作業を実行します。

- デプロイメント・マネージャー・セル内の各ノードに WebSphere Process Server をインストールします。
- データベース、ストレージ・グループ、およびスキーマ (または SQL ID) の命名規則を決定します。DB2 データベース・オブジェクトの作成および構成は構成の後で行いますが、構成スクリプトへの入力を提供する応答ファイルを編集するときに、それらのデータベース・オブジェクトの名前が必要になります。

手順

1. デプロイメント・マネージャーを作成します。
2. 空のノードを構成します。
3. 空のノードをデプロイメント・マネージャー・セルに統合します。
4. クラスターを作成します。
5. SCA を構成します。また、必要に応じて他の Business Process Management コンポーネントを、『ソフトウェアの構成』セクションの説明にあるとおりに構成します。
6. 構成用に 1 つ以上の DB2 データベースを作成します。『データベースの作成および構成』を参照してください。

関連概念

9 ページの『Network Deployment 構成』

初期 Network Deployment 構成に含まれているデプロイメント・マネージャー・サーバーには、デプロイメント・マネージャーが稼働する z/OS® システムのデーモンが含まれています。Network Deployment セルの作成後に、アプリケーション・サーバー・ノードを追加できます。追加するには、新しい空の管理対象ノードを作成して統合するか、またはスタンドアロン・アプリケーション・サーバー・ノードを Network Deployment セルに統合します。

関連資料

45 ページの『応答ファイルの値』

WebSphere Process Server for z/OS 製品定義をインストールするインストール・ジョブを実行すると、サンプル応答ファイルがインストール・ファイル・システムにインストールされます。これらの応答ファイルをコピーし、実現する構成に従って編集し、拡張スクリプトの実行時にその応答ファイルを渡します。

 zWPSConfig.sh スクリプト

インストールした WebSphere Process Server for z/OS を構成および拡張するとき使用するスクリプトです。デプロイメント・マネージャーを含め、構成内の各ノードでスクリプトを実行します。

 WebSphere Process Server をデプロイメント・マネージャー・ノードへインストールするための JCL

 デプロイメント・マネージャー・ノードを拡張するための JCL

 WebSphere Process Server を空のノードへインストールするための JCL

 空のノードを拡張するための JCL

 zWPSInstall.sh スクリプト

zWPSInstall.sh スクリプトは、WebSphere Application Server プロファイルを変更して WebSphere Process Server をインストールまたはアンインストールするために使用します。デプロイメント・マネージャーを含め、構成内の各ノードでスクリプトを実行します。

WebSphere Process Server によるデプロイメント・マネージャーの構成:

デプロイメント・マネージャーは Network Deployment セルを管理します。Network Deployment 構成内のすべてのノードは、DB2 データベース・オブジェクトを使用します。

始める前に

デプロイメント・マネージャー・ノードを構成する前に、以下のタスクを完了しておく必要があります。

- WebSphere Application Server 内にデプロイメント・マネージャー・ノードを作成します。『Network Deployment セルの作成』を参照してください。
- デプロイメント・マネージャー・ノードに WebSphere Process Server をインストールします。

- デプロイメント・マネージャー構成を含むデータ・セットをバックアップします。
- サンプル・ファイルまたは zPMT ツールのいずれかを使用して、DB2 データベースを使用するデプロイメント・マネージャーの応答ファイルを作成します。

手順

1. サーバーを停止します。
2. 製品構成スクリプトを最後まで実行させるために、OMVS の時間制限を延長します。MVS コンソールで、次のコマンドを入力します。
SETOMVS MAXCPUIME=86400
3. USS コマンド・シェルにアクセスしてから、管理者ユーザー ID に切り替えます。以下に例を示します。
su wsadmin
4. 次のように入力して、ディレクトリーをデプロイメント・マネージャーの bin ディレクトリーに変更します。
cd /WebSphere/V7R0/DeploymentManager/bin
5. zWPSConfig.sh 構成スクリプトを実行します。

JCL を使用して、デプロイメント・マネージャー・ノードで構成を実行できます。JCL スクリプトの例については、デプロイメント・マネージャー・ノードを拡張するための JCLを参照してください。

別の方法として、編集した応答ファイルの絶対パスを使用して USS から直接、構成スクリプトを実行することもできます。以下に例を示します。

```
zWPSConfig.sh -augment -response /working_directory_path/DmgrDB2.rsp
```

ここで、*working_directory_path* は編集した DmgrDB2.rsp ファイルのロケーションです。

6. 構成が実行されるのを待ちます。

スクリプトの実行が終了したら、コンソールに表示されるメッセージを確認します。スクリプトが正常に実行された場合、エラー・メッセージは表示されず、「プロファイルの拡張が完了しました (augmenting profile(s) complete)」という情報メッセージが表示されます。

7. デプロイメント・マネージャー構成を含むデータ・セットをバックアップします。

タスクの結果

WebSphere Process Server でデプロイメント・マネージャーが構成されます。

次のタスク

以下のいずれかの方法で DB2 共通データベースを作成できます。

- 提供された createDB.sh スクリプトを実行して、単一の共通データベースを作成します。構成用のすべてのデータベース・オブジェクトは、その中に作成されます。createDB.sh スクリプトは、データベースを作成して、そのデータベースに、例えば Business Process Choreographer、Common Event Infrastructure、および

SCA などのフィーチャーに必要なデータベース・オブジェクトを設定します。
81 ページの『createDB.sh スクリプトによる DB2 データベース・オブジェクトの作成』を参照してください。

- SPUFI、DSNTEP2、または DBUtility.sh を使用して、構成スクリプトにより生成された DDL スクリプトを実行します。この方法は、DDL スクリプトを使用してデータベースを作成した後、別の DDL スクリプトを実行してデータベースを設定するので、少し複雑になります。しかし、この方法を使用すると、データベース・オブジェクトを単一の共通データベース内にすべて作成するのではなく、複数のデータベース内に作成するよう指定できます。83 ページの『SPUFI、DSNTEP2、または DButility.sh を使用した DB2 データベースおよびストレージ・グループの作成』を参照してください。

WebSphere Process Server での空のノードの構成:

空の WebSphere Application Server ノードをデプロイメント・マネージャーに統合する前に、WebSphere Process Server for z/OS 機能を使用して空のノードを構成する必要があります。

始める前に

重要: この段階では、管理対象ノードをデプロイメント・マネージャーに統合しないでください。

空のノードを構成する前に、以下のステップを完了しておく必要があります。

- WebSphere Application Server を使用して空のノードを作成します。『ノードの追加、管理、および除去』を参照してください。この段階では、管理対象ノードを統合しないでください。
- 空のノードに WebSphere Process Server をインストールします。
- デプロイメント・マネージャー構成を含むデータ・セットをバックアップします。
- DB2 データベースを使用する空のノードの応答ファイルを作成します。これを行うには、サンプル・ファイルを編集するか、zPMTtool を使用します。

手順

1. zWPSConfig.sh 構成スクリプトを実行します。

JCL を使用して空のノード上で構成を実行できます。JCL スクリプトの例については、空のノードを拡張するための JCL を参照してください。

別の方法として、編集した応答ファイルの絶対パスを使用して USS から直接、構成スクリプトを実行することもできます。以下に例を示します。

```
zWPSConfig.sh -augment -response /working_directory_path/ManagedDB2.rsp
```

ここで、*working_directory_path* は、編集した ManagedDB2.rsp ファイルのロケーションです。

コマンドについて詳しくは、『zWPSConfig.sh および zWESBConfig.sh スクリプト』を参照してください。

2. 構成が実行されるのを待ちます。

3. **オプション:** 空のノード構成を含んでいるファイル・システムをバックアップします。

タスクの結果

これで、空のノードが WebSphere Process Server for z/OS の機能で構成されました。

次のタスク

必要に応じてその他の空のノードを同じ方法で作成できます。その後、それぞれの空のノードをデプロイメント・マネージャー・セルに統合します (『デプロイメント・マネージャー・セルへの空のノードの統合』を参照してください)。

デプロイメント・マネージャー・セルへの空のノードの統合:

空のノードをデプロイメント・マネージャー・セルに統合すると、その空のノードがデプロイメント・マネージャーと関連付けられ、ノードの管理にデプロイメント・マネージャーを使用できるようになります。空のノードを統合するまでは、作業を処理するためにそのノードを使用することはできません。

始める前に

空のノードをデプロイメント・マネージャー・セルに統合する前に、以下のタスクを実行する必要があります。

- 1 つ以上の空のノードを作成します。

手順

1. デプロイメント・マネージャーを開始します。『MVS コンソールからのサーバーの始動』を参照してください。
2. ノードがあるシステムで、JCL メンバー BBOWMNAN を見つけます。
3. BBOWMNAN ジョブを実行依頼して、戻りコードが 0 で完了したことを確認します。
4. WebSphere Customization Tools (WCT) アプリケーションを使用して空の管理対象ノードを構成するときにノード・エージェントの自動開始を選択しなかった場合は、この時点で、手動でノード・エージェントを開始してください。『MVS コンソールからのサーバーの始動』を参照してください。
5. デプロイメント・マネージャーおよびすべての管理対象ノードの構成ファイル・システムのデータ・セットをバックアップします。

タスクの結果

これで、空のノードがデプロイメント・マネージャー・セルに統合され、管理コンソールでデプロイメント・マネージャーを使用してノードを管理できるようになりました。

次のタスク

管理コンソールを使用してクラスターを作成できます。詳しくは、133 ページの『クラスターの作成』を参照してください。

データベースの作成および構成

WebSphere Process Server インスタンスが機能するためには、それらをサポートするデータベースを構成し、データベース・オブジェクトを定義する必要があります。

このタスクについて

WebSphere Process Server インスタンスをサポートするためのデータベースは Derby と DB2 のいずれかに実装することができます。Derby データベースはスタンドアロン・サーバーでのみ使用することができます。DB2 データベースは、スタンドアロン・サーバーと Network Deployment 環境のいずれかで使用することができます。使用するデータベースを決定するための情報については、69 ページの『WebSphere Process Server を使用した拡張』を参照してください。

Derby データベースは、スタンドアロン・サーバーを拡張すると、自動的に作成されます。DB2 データベース・オブジェクトは、別途、作成する必要があります。

DB2 データベースを作成するための DDL は以下の 2 つの場所のいずれかに格納されます。1 つは、DbDesignGenerator.sh ツールによって DDL が生成されるデフォルト・ディレクトリー (<WAS home>/util/dbUtils) です。データベース設計ツールを使用して DDL を作成する方法については、61 ページの『データベース設計ツールを使用したデータベース設計ファイルの作成』を参照してください。また、DDL は、BPM コンポーネントの構成時または拡張時に、管理コンソールで利用可能な各種のウィザードを使用することによって生成されることもあります。この DDL は、<WAS home>/profiles/default/dbscripts の下の WebSphere Process Server プロファイル・ディレクトリーに配置されます。

DB2 データベースの作成方法の選択

DDL を実行して DB2 データベースを作成するために使用できる、さまざまなツールが提供されています。WebSphere Process Server では、サンプルの createDB.sh スクリプトが提供されています。このスクリプトを USS コマンド環境から実行して、データベース・オブジェクトを作成できます。また、z/OS 環境から実行する DB2 ツール (SPUFI や DSNTEP2 など) を使用して、DDL を実行することもできます。

使用するツールの選択

使用するツールの選択は、経験や慣れ、または個人的な好みに基づいて行う場合があります。また、DB2 オブジェクトの作成に使用するツール (特に、実稼働環境で使用するもの) について、組織が標準や規則を導入している場合もあります。

createDB.sh サンプルの選択に関する考慮事項

- createDB.sh を使用すると、ツールを 1 回実行するだけですべてのデータベース・オブジェクトを作成できます。サーバーを初めて実装する場合は、これを選択することをお勧めします。
- createDB.sh は、DbDesigngenerator.sh ツールによって生成された DDL を実行します。

- createDB.sh は、定義済みの命名規則に従ってデータベース・オブジェクトを作成します。ただし、これはサンプル・スクリプトとして提供されているため、実際の命名編成に合わせてカスタマイズが必要になる場合があります。
- createDB.sh は、CEL DDL および SIB DDL のカスタマイズを自動的に完了します。
- createDB.sh は USS 環境から実行します。
- createDB.sh では、作成したオブジェクトの監査証跡は生成されません。

他のツールの選択に関する考慮事項

- DDL ファイルを、WebSphere Process Server ファイル・システムから区分データ・セット (PDS) にコピーすることが必要になる場合があります。ファイルのコピーには Ddl2Pds.sh ツールを使用できます。詳しくは、Ddl2Pds.sh スクリプトを参照してください。
- データベース・オブジェクトに適用する命名規則や編成規則に制限はありません。
- CEI DDL ファイルと SIB DDL ファイルは、実行する前にカスタマイズが必要です。(注: -RunSQL オプションを指定して createDB.sh を使用すると、CEI DDL および SIB DDL を実行せずに生成することができます。その後、createDB.sh によって生成された DDL を使用できます。)
- 一部のツールは、z/OS 環境から実行される場合があります。
- 各ツールによって、発行された DB2 データベース・コマンドの監査証跡が生成される場合があります。

createDB.sh スクリプトによる DB2 データベース・オブジェクトの作成

createDB.sh スクリプトを実行すると、DB2 データベースを作成し、WebSphere Process Server 用に構成した後、データベースにオブジェクトを設定することができます。createDB.sh スクリプトは、DB2 を使用するスタンドアロン構成、または Network Deployment 構成に使用できます。createDB.sh を使用すると、構成用のすべてのデータベース・オブジェクトは単一のデータベース内に作成されます。複数のデータベースを使用する場合は、別の方法を使用してデータベースとデータベース・オブジェクトを作成します。例えば、提供されている SPUFI や DBUtility.sh スクリプトを使用します。

始める前に

createDB.sh スクリプトを実行するには、事前に以下の手順を完了しておく必要があります。

- データベースの命名規則を決定します。
- スキーマ名および SQL ID を決定します。
- スタンドアロン構成の場合: DB2 と一緒に使用するスタンドアロン・サーバーを作成します。Network Deployment 構成の場合: デプロイメント・サーバーを作成します。ノードを構成したときに、応答ファイル内の dbDelayConfig パラメーターが true に設定されたことを確認します。
- 必要なバッファ・プールを作成します。

- DbDesignGenerator.sh ツールを実行して、構成しようとするすべてのコンポーネントのデータベース表を作成するための DDL を生成します。DbDesignGenerator.sh は <WAS_HOME>/util/dbUtils ディレクトリーから実行し、DDL をデフォルトのディレクトリー名に生成する必要があります。Process Choreographer および Process Choreographer Reporter 用に DB2 for z/OS DDL を生成する場合は、データベースをスタンドアロン・サーバー用にセットアップする場合であってもネットワーク・デプロイメント設計を作成する必要があります。

このタスクについて

createDB.sh スクリプトはデータベースを作成し、それに WebSphere Process Server が必要とするすべての DB2 データベース・オブジェクトを設定します。これには、SCA、Business Process Choreographer、および Common Event Infrastructure が使用するデータベース・オブジェクトも含まれます。sysplex 間でデータベースが共用されず、WebSphere Process Server と DB2 が異なる LPAR 上にインストールされている sysplex 環境では、WebSphere Process Server ではなく DB2 がインストールされているシステムで createDB.sh を実行する必要があります。

手順

1. スタンドアロン・サーバー・ノードまたはデプロイメント・マネージャー・ノードでは、USS コマンド・シェルにアクセスしてから、管理者ユーザー ID に切り替えます。以下に例を示します。

```
su wsadmin
```

2. サンプルの createDB.sh ファイルを作業ディレクトリーにコピーします。以下に例を示します。

```
cp /usr/lpp/zWPS/V7R0/zos.config/samples/createDB.sh /u/work
```

3. createDB.sh ファイルのコピーに適切な許可を割り当てます。

```
chmod 755 createDB.sh
```

4. システムの必要に応じて、createDB.sh ファイルのコピー内のパラメーターをカスタマイズします。詳しくは、createDB.sh スクリプト のトピックを参照してください。

5. 編集したファイルを保存します。

6. sibDropxx.sql および sibSchemaxx.sql を作業中のスクリプト・ディレクトリーにコピーします。以下に例を示します。

```
cp /usr/lpp/zWPS/V7R0/zos.config/samples/sibDropxx.sql /u/work
cp /usr/lpp/zWPS/V7R0/zos.config/samples/sibSchemaxx.sql /u/work
```

7. WAS_HOME 環境変数を適切に設定します。以下に例を示します。

```
export WAS_HOME=/WebSphere/V6S01Z1/AppServer
```

8. 必要に応じて LIBPATH 変数と STEPLIB 変数を設定して DB2 コードにアクセスします。以下に例を示します。

```
export LIBPATH=/ZOS180/usr/lpp/db2910/lib:$PATH
export STEPLIB=SYS2.DB2.V910.SDSNEXIT:SYS2.DB2.V910.SDSNLOAD2:
SYS2.DB2.V910.SDSNLOAD:$STEPLIB
```

9. カスタマイズした createDB.sh スクリプト を実行します。エラーがないかどうかを、cdbtmp ディレクトリーの error.out ファイルで確認します。このファイルは、/u/work ディレクトリーの createDB.sh スクリプトによって作成されません。

10. スタンドアロン構成を作成する場合は、次のようにして WebSphere Process Server のインストールを確認します。
 - a. サーバーを始動します。
 - b. ブラウザー・ウィンドウを開き、表示するサーバーの URL を入力して管理コンソールを開きます。以下に例を示します。

```
http://server_name.domain_name:port_number/admin
```
 - c. 管理コンソールにログインします。
 - d. 「ようこそ」ページに WebSphere Process Server が表示されることを確認します。これをクリックして詳細を表示することもできます。
 - e. コンソール内を移動して、以下の項目を確認します。
 - エンタープライズ・アプリケーションが起動している。
 - メッセージング・エンジンが起動している。
 - データ・ソース接続のテストを行う。
- 起動できなかったものがある場合は、サーバーのジョブ・ログで「SEVERE」または「WARNING」のメッセージを探して、失敗に関する詳細情報を確認します。

タスクの結果

データベースが作成され、構成に必要なデータベース・オブジェクトが設定されます。

次のタスク

スタンドアロン構成を作成する場合は、この時点でアプリケーションをサーバーにデプロイすることができます。

Network Deployment 構成を作成する場合は、次に、1 つ以上の空のノードを作成して、デプロイメント・マネージャー・セルに追加する必要があります。78 ページの『WebSphere Process Server での空のノードの構成』を参照してください。

SPUFI、DSNTEP2、または DBUtility.sh を使用した DB2 データベースおよびストレージ・グループの作成

zWPSConfig.sh 構成スクリプトにより、データ定義言語 (DDL) スクリプトが生成され、これを使用して、その構成に対する DB2 のデータベース・オブジェクトを作成できます。ご使用の構成に対するデータベース・オブジェクトを作成するための DDL スクリプトを実行するために使用できるツールがいくつかあります。

始める前に

DB2 のデータベースおよびストレージ・グループを作成する前に、以下のタスクを完了する必要があります。

- サーバー構成を作成します。スタンドアロン・サーバーの場合は、72 ページの『スタンドアロン構成の作成』を参照してください。デプロイメント・マネージャー・ノードの場合は、76 ページの『WebSphere Process Server によるデプロイメント・マネージャーの構成』を参照してください。

- データベースの構成で使用するすべてのコンポーネントの DDL が生成されていることを確認してください。以下のタスクを実行して、DDL を生成することができます。
 - zWPSConfig.sh スクリプト スクリプトを実行する。
 - 適切なコンポーネントを構成してデプロイメント環境を作成および生成する。
 - 個々のコンポーネント構成ウィザードを実行する。
 - DbDesignGenerator.sh ツールを使用する。

このタスクについて

DBUtility.sh、SPUFI、または DSNTEP2 を使用して DDL スクリプトを実行できます。

USS 環境内で処理する場合は、DBUtility.sh スクリプトを使用して DDL スクリプトを実行できます。DBUtility.sh スクリプトも、WebSphere Process Server に付属しています。createDB.sh スクリプトは、単一データベース内にすべての DB2 データベース・オブジェクトを作成します。複数のデータベースにまたがるデータベース・オブジェクトを作成し、USS 環境内で作業を行う場合は、DBUtility.sh スクリプトを使用して DDL スクリプトを実行するか、データベース名ごとに異なるコンポーネントを指定して createDB.sh を複数回実行します。

重要: ASCII から EBCDIC に変換した後に、71 文字を超える長さの SQL ステートメントがないことを確認してください。このようなステートメントがあると、固定幅の MVS データ・セットにコピーするときに行が切り捨てられ、ステートメントが無効になるためです。

手順

1. データベースおよびストレージ・グループを作成します。
2. 生成された DDL スクリプトを使用して、データベースにデータを追加します。生成された DDL スクリプトの場所は、生成された方法によって異なります。

DbDesignGenerator.sh によって生成された DDL のデフォルトの場所は、以下の場所にあるディレクトリーです。

- スタンドアロン構成の場合は、<WAS_ROOT>/AppServer/util/dbUtils。
- Network Deployment 構成の場合は、<WAS_ROOT>/DeploymentManager/util/dbUtils。

他の方法で生成された DDL の場合、DDL は以下の場所のディレクトリーに存在します。

- スタンドアロン構成の場合は、<WAS_ROOT>/AppServer/profiles/default/dbScripts。
- Network Deployment 構成の場合は、<WAS_ROOT>/DeploymentManager/profiles/default/dbScripts。

ここで、<WAS_ROOT> は、WebSphere Application Server 構成の上位ディレクトリーです。

3. DDL を USS 環境から実行している場合は、ファイルのコピーに適切なアクセス権を割り当てます。以下に例を示します。

```
chmod 755 createTable_AppScheduler.sql
```

- 必要に応じてファイル内の値を編集します。 データベース名、ストレージ・グループ、およびスキーマ名は、製品の構成処理によってカスタマイズされます。各ファイルの値を調べ、構成スクリプトへの入力を提供する応答ファイルに入力した値に一致しており、かつデータベースに適していることを確認します。

注: ファイルは ASCII フォーマットでも構いません。スクリプトの表示、編集、および実行に使用するツールで、スクリプトが EBCDIC 形式であることが必要な場合は、`iconv` コマンドを使用して、ファイルを EBCDIC に変換します。例えば、次のようにします。

```
iconv -t IBM-1047 -f ISO8859-1 createTable_AppScheduler.sql >  
createTable_AppScheduler_EBCDIC.sql
```

ファイルを ASCII フォーマットから EBCDIC に変換してある場合で、そのファイルを ASCII フォーマットで実行する必要がある場合は、`iconv` を使用して、ファイルを ASCII に戻します。以下に例を示します。

```
iconv -t ISO8859-1 -f IBM-1047 createTable_AppScheduler_EBCDIC.sql >  
createTable_AppScheduler.sql
```

- オプション:** USS 環境外で SPUFI または DSNTEP2 を介してデータベース・オブジェクトを作成する場合は、提供されている `Ddl2Pds.sh` スクリプトを使用して、カスタマイズ済み DDL を USS から区分データ・セットにコピーすることができます。例えば、WebSphere Process Server 共通コンポーネント用の DDL をコピーするには、以下のようなコマンドを `/usr/lpp/zWPS/V7R0/zos.config/samples` ディレクトリーから入力します。

```
./Ddl2Pds.sh -Source  
/WebSphere/V7S05Z1/AppServer/profiles/default/dbscripts/CommonDB/DB2z0SV8/S5CELLDB -PDS HEALDR.DDL2PDS.TEST -Component  
WPS
```

- 選択したツールを使用して、カスタマイズ済みのスクリプトを実行します。例えば、次のようにします。

SPUFI z/OS から SQL スクリプトを実行するユーティリティです。SPUFI は EBCDIC 入力を使用します。

DSNTEP2

DB2 for z/OS 製品で提供される動的 SQL プログラムのサンプル。

DBUtility.sh

DBUtility.sh は、WebSphere Process Server for z/OS に付属のユーティリティで、インストール・ファイル・システムにインストールされます。例えば、`/usr/lpp/zWESB/V7R0/bin/DBUtility.sh` です。このユーティリティを使用して、データベースおよびストレージ・グループを作成でき、さらに、後で USS から SQL を実行してデータベース表を作成することもできます。DBUtility.sh は ASCII 入力を使用します。

DBUtility.sh スクリプトで使用される構文の例を以下に示します。

```
/WebSphere/V7S03Z1/AppServer/profiles/default/bin/DBUtility.sh  
createTable  
-DdbStorageGroup=S3DBSTO  
-DdbSchemaName=S3CELL  
-DsqlScriptName.default=createTable_AppScheduler.sql  
-DsqlScriptPath.default=/WebSphere/V7S03Z1/AppServer/profiles/default/dbscripts/CommonDB/DB2z0SV8/S3CELLDB  
/createTable_AppScheduler.sql  
-DdbType=DB2UDBOS390_V8_1  
-DdbName=S3CELLDB  
-DprofileName=default  
-DprofilePath=/WebSphere/V7S03Z1/AppServer/profiles/default
```

```
-DdbJDBCProperties=/wps/dbscripts/db2v8
-DdbConnectionLocation=DSN810PP
-DdbJDBCClasspath=/usr/1pp/db2810/db2810/jcc/classes
-DdbUserId=wsadmin
-DdbPassword=password
-DdbDelayConfig=false
-DdbCreateNew=false
-DdbHostName=winmvsp1.hursley.ibm.com
-DdbServerPort=448
>/tmp/output.out 2>>/tmp/error.out
```

7. 出力を調べることによって、エラーが発生せずに正常にデータベース、ストレージ・グループ、および表が作成されたことを確認します。
8. スタンドアロン構成を作成する場合は、WebSphere Process Server のインストール済み環境を検証します。
 - a. サーバーを始動します。
 - b. ブラウザー・ウィンドウを開き、表示するサーバーの URL を入力して、管理コンソールを開きます。以下に例を示します。
`http://server_name.domain_name:port_number/admin`
 - c. 管理コンソールにログインします。
 - d. 「ようこそ」ページに WebSphere Process Server が表示されることを確認します。これをクリックして、詳細情報を表示できます。
 - e. コンソール内を移動して、サーバーの状況が開始済みであることを確認します。すべてのアプリケーションが開始済みであること、およびメッセージング・エンジンが開始済みであることも確認します。開始に失敗したものがあ
る場合、サーバーのジョブ・ログで「SEVERE」または「WARNING」のメッセージを探し、失敗についての詳細を確認できます。

タスクの結果

DB2 データベースおよびストレージ・グループが作成され、テーブル、索引などの必要なデータベース・オブジェクトが設定されます。

次のタスク

スタンドアロン構成を作成している場合は、これでアプリケーションをサーバーにデプロイできます。

Network Deployment 構成を作成している場合は、次に、1 つ以上の空のノードを作成し、デプロイメント・マネージャーのセルに追加する必要があります。78 ページの『WebSphere Process Server での空のノードの構成』を参照してください。

メッセージ・エンジンのデータ・ストアの作成

まだ createDB.sh スクリプト を使用してメッセージング・エンジンのデータ・ストアを作成していない場合は、sibDDLGenerator コマンドを使用して、データベース管理者がメッセージング・エンジンのデータ・ストアのテーブルを作成するために使用する DDL ステートメントを生成します。

始める前に

SQL スクリプトを作成して実行する前に、以下を行います。

- クラスタを作成します。

- Network Deployment 構成内に作成する SIBus コンポーネントを決定します。

このタスクについて

構成する以下のコンポーネントごとにメッセージング・エンジンのデータ・ソースを作成し、設定します。

- SCA (システム・バスおよびアプリケーション・バス)
- Business Process Choreographer
- Common Event Infrastructure

すべてのデータベース・オブジェクトを単一のデータベース内に作成することも、1つの既存データベース内に作成することも、1つ以上の新規データベース内に作成することもできます。以下の点に注意してください。

- 各バスには同一のテーブル名が付いています。このため、各バスのテーブルには、固有のスキーマ修飾子を付けてください。
- 各バスには同一のテーブル・スペース名が付いています。このため、バスごとに異なるデータベースを使用してください。単一のデータベースを使用する場合は、テーブル・スペース名を編集して、データベース内部で固有になるようにしてください。
- createDB.sh を使用してバス・データベースを作成する場合は、5文字のスキーマ修飾子を組み込むようにテーブル・スペース名が修正されます。以下に例を示します。
 - OWNERTS => <スキーマ>OWN、例: A6S1AOWN
 - OWNEROTS => <スキーマ>OTS、例: A6S1AOTS
 - MAPTS => <スキーマ>MAP、例: A6S1AMAP
 - LISTTS => <スキーマ>LST、例: A6S1ALST
 - SIB000TS => <スキーマ>00T、例: A6S1A00T
 - SIB000LS => <スキーマ>00L、例: A6S1A00L
 - SIB001TS => <スキーマ> 01T、例: A6S1A01T
 - SIB001LS => <スキーマ> 01L、例: A6S1A01L
 - SIB002TS => <スキーマ> 02T、例: A6S1A02T
 - SIB002LS => <スキーマ> 02L、例: A6S1A02L
 - XACTSTS => <スキーマ>XAT、例: A6S1AXAT
 - KEYTS => <スキーマ>KEY、例: A6S1AKEY

手順

必要なデータ・ソースごとに、WebSphere Application Server インフォメーション・センターの次の説明に従って SQL スクリプトを作成および実行します。『データベース管理者によるデータ・ストア・テーブルの作成を可能にする』。例えば、S2SIBSCA という新しいデータベースを使用して SCA の SIB データ・ソースを作成するには、以下のコマンドを実行します。

```
sibDDLGenerator.sh -system db2 -version 8.1 -platform zos -schema
S2CELLS -user wsadmin -create -database S2SIBSCA -storagegroup
S2SIBST0 -buffer pool BP3 -statementend ";" > /u/hssd/SIBSCA.dd1
```

DDL の作成時には、-buffer pool オプション (この例では -buffer pool BP3) を使用してデフォルト以外のバッファ・プールを設定できます。

タスクの結果

メッセージング・エンジンのデータ・ストアが作成されます。

次のタスク

次に以下を実行して、SCA、Business Process Choreographer、および Common Event Infrastructure を構成します。

- 135 ページの『サーバーまたはクラスターの SCA サポートの構成』
- Business Process Choreographer の構成
- Common Event Infrastructure の構成

JCA 認証別名ユーザー ID への表の特権の付与

使用しているスキーマ名が JCA 認証別名ユーザー ID と同じものでない場合は、DB2 特権のサブセットを JCA 認証別名ユーザー ID に付与する必要があります。

このタスクについて

サービス統合バス用の DDL には、コメント化された GRANT コマンドが既に含まれていて、これを SIB テーブルへのアクセス権を付与するための基盤として使用できます。ただし、他の WebSphere Process Server for z/OS コンポーネントは GRANT ステートメントを何も提供しません。

JCA 認証別名と同じものでないスキーマ名を使用して、別名ユーザー ID が表を除去する能力を持たないようにしてください (テーブルをドロップする権限は、作成者つまりスキーマに暗黙的に付与されます)。DBADM のような特権を JCA 認証別名ユーザー ID に付与することには意味がありません。これは、DBADM 特権でもテーブルを DROP することができるからです。

別名ユーザー ID で DROP を実行できないようにしたまま、WebSphere Process Server を機能させるには、DDL をコピーし、それを編集して、CREATE コマンドから GRANT ステートメントを作成することにより、いくつかの GRANT ステートメントを作成します。次のような GRANT コマンドを作成します。

```
GRANT ALL PRIVILEGES ON TABLE  
cell.tablename TO userid/sqlid
```

ここで、userid/sqlid は JCA 認証別名ユーザー ID です。

SIB の正しいスキーマ名の設定

SIB メッセージング・エンジンが適切な DB2 表にアクセスできるようにするには、SIB に正しいスキーマ名を設定します。

始める前に

- サーバーを始動します。

このタスクについて

管理コンソールを使用して、スキーマ名を変更します。

手順

1. 管理コンソールにログインします。
2. 「サービス統合バス」 → 「バス」にナビゲートします。
3. 各バスについて、次のようにします。
 - a. 「メッセージング・エンジン」を選択し、表示される名前をクリックします。
 - b. 「メッセージ・ストア」をクリックします。
 - c. 「スキーマ名」の値を、この SIB 用の DB2 表を作成したときに使用した名前に変更します。
 - d. 「適用」をクリックします。
 - e. 構成変更を保存します。
4. 管理コンソールからログアウトします。
5. サーバーを停止した後で、再始動します。
6. 付属ジョブ・ログの出力を参照して、成功を示す SIB メッセージング・エンジン始動メッセージを探します。以下に例を示します。

```
BB000222I: "BusName"  
CWSID0016I: Messaging engine MessagingEngineName is in state Started.
```

タスクの結果

SIB メッセージング表が DB2 表へのアクセスに使用するスキーマ名が変更されました。

サーバントおよび付属 JCL への DB2 ライブラリーの追加

ご使用の DB2 システムが LNKLST で SDSNEXIT、SDSNLOAD、および SDSNLOD2 を指定して実行されていない場合は、WebSphere サーバントおよび付属 JCL を更新して、DB2 のカスタマイズ済みの SDSNEXIT、SDSNLOAD、および SDSNLOD2 が STEPLIB に含まれるようにします。

このタスクについて

STEPLIB を正しく設定していない場合は、SDSNLOD2 内の DB2 プログラムをロードできないために、サーバーの systout に「T2 ネイティブ・ライブラリー db2jct2zos のロード障害」の問題が出力されます。このエラーは、初期化時に SIB が失敗する原因になります。

構成が完了したら、WebSphere Application Server の構成ファイル・システムをバックアップします。

Derby JDBC リソースのクリーンアップ

Derby JDBC プロバイダー用に定義された JDBC データ・ソースが見つかった場合は、それらを削除できます。一部の環境では、SIB データ・ソースが Derby JDBC プロバイダーの下で定義されます。

このタスクについて

以下の手順を実行して、Derby JDBC リソースをクリーンアップします。

手順

1. 管理コンソールにログインし、「リソース」 → 「JDBC プロバイダー」にナビゲートします。
2. ノード名またはサーバー名を除去し、「適用」をクリックすることにより、セル・レベルにスコープを設定します。
3. Derby JDBC プロバイダーまたは Derby JDBC プロバイダー (XA) が見つかったらそれらをクリックし、右側の「データ・ソース」をクリックします。
4. WebSphere Process Server に関連する文字ストリング (例えば WPS、BPE、SCA、SIB など) を含んでいる名前のデータ・ソースがあれば、それらを削除します。
5. 構成変更を保存します。
6. 再び「リソース」 → 「JDBC プロバイダー」にナビゲートし、スコープをアプリケーション・サーバー・ノードに変更します。
7. WebSphere Process Server に関連するデータ・ソースの Derby JDBC プロバイダーがないかどうか検査し、あれば削除します。
8. 構成変更を保存します。
9. 「リソース」 → 「JDBC プロバイダー」にナビゲートし、スコープをノード・レベルに変更し、アプリケーション・サーバーを選択します。
10. WebSphere Process Server に関連するデータ・ソースの Derby JDBC プロバイダーがないかどうか検査し、あれば削除します。
11. 構成変更を保存します。

次のタスク

次のステップでは、インストールを検査します。詳しくは、www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101218 を参照してください。

DB2 を使用するインストールの検証

DB2 データベースを使用するインストールを検証する場合は、サーバント・ジョブと付属ジョブのログを確認し、データ・ストアへのアクセスに問題があることを示すエラー・メッセージがないかどうかを調べることが重要です。

手順

1. 付与されている権限を調べ、どのユーザー ID にも必要以上の権限を付与していないことを確認するように、DB2 システム管理者に依頼してください。構成時に起こる可能性がある DB2 のセキュリティーに関する問題を回避するために、JCA 認証別名に DB2 の SYSADM 権限を付与したくなる場合もあります。
2. ストレージ・グループの割り当てとバッファ・プールの使用状況を調べるように、DB2 システム管理者に依頼してください。ストレージ・グループの割り当てとバッファ・プールの使用量が正しくなくても、ログにエラー・メッセージとして表示されない場合がありますが、後で問題の原因になる可能性があります。このような問題は、システムを使用するユーザーに引き渡した後でなく、こ

の時点で解決しておくことを推奨します。例えば、ストレージ・グループおよび VCAT の修正は、表と索引が使用された後では容易ではありません。

3. 管理コンソールにログインします。『管理コンソールの始動と停止』を参照してください。
4. 管理コンソールで、すべてのアプリケーションが開始されていることと、メッセージング・エンジンが開始されていること、およびすべてのデータ・ソースに対して「**テスト接続**」オプションを使用してアクセスできることを確認します。開始に失敗したアプリケーションがある場合、サーバント・ジョブおよび付属ジョブのログで「SEVERE」または「WARNING」のメッセージを探し、失敗についての詳細を確認できます。
 - 管理コンソールに SQLCODE -204 などの DB2 エラーが表示された場合は、データ・ソースのカスタム・プロパティ・セクションに正しいスキーマ名または currentSQLID 値を設定します。スキーマ名が JCA 認証別名のユーザー ID と異なる場合、SQL 要求は、JCA 認証別名のユーザー ID によって修飾されている表を探そうとします。
 - SQLCODE -913 理由コード 00C90088 などの DB2 デッドロック・エラーが表示された場合は、『調整チェックリスト』セクションの説明のとおり RRULOCK DB2 パラメーターが YES に設定されていることを確認してください。

次のタスク

すべての SIB が正しく初期化済みで、JDBC 接続を開くことに関連するその他のエラーが出力されていない場合は、WebSphere Process Server の構成のカスタマイズを続行できます。

ConsolidateJAASAuthAliases.py スクリプト

ConsolidateJAASAuthAliases.py は wsadmin スクリプトで、拡張プロセスによってデータベース・アクセスのために定義された、いくつかの JAAS 認証別名を統合するために使用します。

概要

WebSphere Process Server for z/OS サーバーがセキュア・データベース・サブシステムにアクセスする際、使用可能なセキュリティー・メカニズムの 1 つに JAAS 認証別名の使用があります。JAAS 認証別名は、データベース・サブシステムが認証資格情報を要求する際に、提供されるユーザー ID およびパスワードを指定します。拡張プロセスは、データベースにアクセスする際に使用するいくつかのデータ・ソースおよびサービス統合バスに関連付けられた JAAS 認証別名のセットを定義します。また、別名は多数の WebSphere Relational Resource Adapter CMP 接続ファクトリーに割り当てられます。

完全に構成済みの WebSphere Process Server システムは、以下のリソースと拡張プロセスによって定義された JAAS 認証別名から構成されています。

表 7. JAAS 認証別名

データ・ソース	JAAS 認証別名
ESBLoggerMediationDataSource	WPSDB_Auth_Alias

表 7. JAAS 認証別名 (続き)

データ・ソース	JAAS 認証別名
WBI_DataSource	WPSDB_Auth_Alias
event	<qualifier>/EventAuthDataAliasDB2
CEI ME データ・ソース	CEIME_<qualifier>_Auth_Alias
SCA システム・バス ME データ・ソース	SCASYSME00_Auth_Alias
SCA アプリケーション・バス ME データ・ソース	SCAAPPME00_Auth_Alias
Business Process Choreographer データ・ソース	BPCDB_<qualifier>_Auth_Alias
Business Process Choreographer ME データ・ソース	BPCME_00_Auth_Alias
Business Process Choreographer レポート作成機能のソース	BPCEDB_<qualifier>_Auth_Alias
Business Space データ・ソース	BSPACE_Auth_Alias
CMP 接続ファクトリー	JAAS 認証別名
WBI_DataSource_CF	WPSDB_Auth_Alias (コンポーネント管理)
event_CF	<qualifier>/EventAuthDataAliasDB2 (コンポーネント管理)
CEI ME data source_CF	CEIME_<qualifier>_Auth_Alias (コンポーネント管理)
SCA システム・バス ME data source_CF	SCASYSME00_Auth_Alias (コンポーネント管理)
SCA アプリケーション・バス ME data source_CF	SCAAPPME00_Auth_Alias (コンポーネント管理)
Business Process Choreographer data source_CF	BPCDB_<qualifier>_Auth_Alias (コンポーネント管理)
Business Process Choreographer ME data source_CF	BPCME_00_Auth_Alias (コンポーネント管理)
Business Process Choreographer reporting function source_CF	BPCEDB_<qualifier>_Auth_Alias (コンポーネント管理)
Business Space data source_CF	BSPACE_Auth_Alias (コンポーネント管理)
SIBuses	JAAS 認証別名
<qualifier>-CEI<cell>BUS	CEIME_<qualifier>_Auth_Alias
<qualifier>-SCA.SYSTEM.<cell>.Bus	SCASYSME00_Auth_Alias
<qualifier>-SCA.APPLICATION.<cell>.Bus	SCAAPPME00_Auth_Alias
<qualifier>-BPC.<cell>.Bus	BPCME_00_Auth_Alias

z/OS では、さまざまなデータ・リポジトリのすべてが同じ z/OS データベース・サブシステム (DB2 for z/OS など) にアクセスするように定義されています。また、この共通データベース・サブシステムへの認証は、同じユーザー ID およびパスワードを使用して実行されます。すべてではありませんが、拡張プロセスで定義されている JAAS 認証別名の大半が同じユーザー ID およびパスワードで定義されていることは稀ではありません。

同じユーザー ID およびパスワード・パラメーターで多数の JAAS 認証別名が定義されていることで、懸念がいくつか生じます。

- データベース・アクセスに必要なパスワードは通常期限切れになりませんが、何らかの理由で変更が必要になった場合、すべての JAAS 認証別名で変更する必要があります。
- JAAS 認証別名を扱う管理コンソールのパネルが雑然となるため、使いにくくなります。
- JAAS 認証別名の名前がローカルの命名規則に準拠しない可能性があります。

目的

これらの問題に対処するには、`ConsolidateJAASAuthAliases.py` スクリプトを実行することもできます。このスクリプトは、WebSphere wsadmin ユーティリティによって呼び出され、以下のアクションを実行します。

- 表 7 にリストされたさまざまな JAAS 認証別名を単一のエントリーに統合します。
- 元の別名を参照するすべてのリソースの再割り当てを行い、これらのリソースが新しい別名を使用するようにします。
- 元の別名を削除します。

その結果、単一の JAAS 認証別名を使用して、WebSphere Process Server for z/OS 構成プロセスで作成されたすべてのリソースについて、データベース・アクセスの認証が行われるようになります。

ConsolidateJAASAuthAliases スクリプト

wsadmin Jython スクリプトを使用して、WebSphere Process Server または WESB 構成で作成された、さまざまな JAAS 認証別名を単一のエントリーに統合できます。

デフォルトでは、スクリプトの場所は、`/usr/lpp/zWPS/V7R0/zos.config/samples` です。

スクリプトの有効範囲

スクリプトは元々、WebSphere Application Server for z/OS V6.1.0.15 上で稼働する WebSphere Process Server または WESB for z/OS V6.1.0.1 用に開発されました。スクリプトは、スタンドアロン・サーバー、およびデプロイメント・マネージャー・ノードと単一のアプリケーション・サーバー・ノードで構成された Network Deployment セルを対象にテストされました。

スクリプトの呼び出し

スクリプトは、WebSphere wsadmin ツールの引数として提供されています。スクリプトには、7 つの必須パラメーターと 1 つのオプション・パラメーターを指定できます。以下のコードに、wsadmin Jython スクリプトの構文を示します (見やすくするため、複数の行に分割してあります)。

```
/AppServerRoot/bin/wsadmin.sh
-host ホスト名
-port ホスト・ポート
-lang jython
```

`-f ConsolidateJAASAuthAliases.py`
JAAS 認証別名
ユーザー ID
パスワード
[scan mode]

パラメーター

-host ホスト名

ターゲット・サーバーまたは Network Deployment セルのデプロイメント・マネージャーのホスト・アドレス。

-port ホスト・ポート

ターゲット・サーバーの SOAP ポート番号。

-lang jython

スクリプトが書かれている言語 (Jython)。

-f ConsolidateJAASAuthAliases.py

スクリプトが現行ディレクトリーに存在しない場合、スクリプトが格納されているパスを含める必要があります。

JAAS 認証別名

作成する新しい JAAS 認証別名の名前。任意の名前でかまいませんが、説明的な名前を選択した方が適切です (WPSDBAccess など)。

ユーザー ID

データベース・サブシステムへの認証で指定されるユーザー ID。

password

データベース・サブシステムへの認証で指定されるパスワード。

[scan mode]

オプション・パラメーター。このパラメーターが欠落している場合 (つまり、スクリプトにパラメーターが 3 つしか指定されていない場合)、スクリプトによるあらゆる変更は、スクリプトが処理を完了した際にコミットされます。8 番目のパラメーターとして任意のストリングが指定された場合、スクリプトは行ったすべての変更を報告しますが、スクリプトが処理を完了する際に、それらの変更はロールバックされます。スキャン・モードは、スクリプトが影響を与える変更の範囲を評価する場合に役立ちます。

このスクリプトは、実行したすべてのアクションに関する報告を提供します。

スクリプト処理

スクリプトの処理は、以下のステップで構成されています。

1. スクリプトに指定されたパラメーターに基づいて、新しい JAAS 認証別名を作成します。
2. 関連するすべての JAAS 認証別名を識別します。スクリプトはすべての別名リストを検索し、以下のパターンと一致する別名を探します。
 - 先頭が「BPCDB_」
 - 先頭が「BPCME_」
 - 先頭が「CEIME_」
 - 先頭が「SCAAPPME」

- 先頭が「SCASYSME」
 - 最後が「EventAuthDataAliasDB2ZOS」
 - 「WPSDB_Auth_Alias」と一致
3. 識別された別名ごとに、該当するすべての箇所が JDBC データ・ソース定義の新しい別名に置換されます。例外は、event_catalog などのコンテナ管理別名を持つデータ・ソースです。コンテナ管理別名は非推奨であるため、コンポーネント管理別名として新しい別名に置換されます。
 4. 識別された別名ごとに、該当するすべての箇所が SIBus メッセージング・エンジン・データ・ストア定義で新しい別名に置換されます。
 5. 識別された別名ごとに、該当するすべての箇所が WebSphere Relational Resource Adapter CMP 接続ファクトリー定義から削除されます。
 6. 識別されたすべての別名が WebSphere 構成から削除されます。
- 4 番目のパラメーターがスクリプトに指定されていない場合、変更はコミットされます。8 番目のパラメーターに任意のストリングが指定されている場合、変更はバックアウトされます。ただし、スクリプトは行うはずだった変更について報告を行います。

出力例

以下の出力に、スクリプトの実行サンプルを示します。

```
/WebSphere/V7T5DM/DeploymentManager/bin:>./wsadmin.sh -lang jython -host winmvsp1 -port 20502
-f /u/healdr/Jython/ConsolidateJAASAuthAliases.py DB2Alias wsadmin gadzooks
WASX7209I: Connected to process "dmgr" on node T5NodeDmgrMVP1 using SOAP connector; The type of
process is: DeploymentManager
WASX7303I: The following options are passed to the scripting environment and are available as
arguments that are stored in the argv variable: "[DB2Alias, wsadmin, gadzooks]"

ConsolidateJAASAuthAliases: Starting

Created JAAS alias: DB2Alias

Replacing alias reference in data source: ESLoggerMediationDataSource
WPSDB_Auth_Alias => DB2Alias

Replacing alias reference in data source: WBI_DataSource
WPSDB_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: WBI_DataSource_CF
Component-managed WPSDB_Auth_Alias

Removing alias: WPSDB_Auth_Alias

Replacing alias reference in data source: event
T5Cell/T5DepEnv.AppTarget/EventAuthDataAliasDB2 => DB2Alias

Removing alias reference from CMP connection factory: event_CF
Component-managed T5Cell/T5DepEnv.AppTarget/EventAuthDataAliasDB2

Removing alias: T5Cell/T5DepEnv.AppTarget/EventAuthDataAliasDB2

Replacing alias reference in data source: CEI ME data source
CEIME_T5DepEnv.AppTarget_Auth_Alias => DB2Alias

Replacing alias reference in SIBus data store of ME: T5DepEnv.AppTarget.000-C
EI.T5Cell.BUS CEIME_T5DepEnv.AppTarget_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: CEI ME data source_CF
Component-managed CEIME_T5DepEnv.AppTarget_Auth_Alias

Removing alias: CEIME_T5DepEnv.AppTarget_Auth_Alias

Replacing alias reference in data source: SCA System Bus ME data source
SCASYSME00_Auth_Alias => DB2Alias
```

```

Replacing alias reference in SIBus data store of ME:
T5DepEnv.AppTarget.000-SCA.SYSTEM.T5Cell.Bus SCASYSME00_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: SCA System Bus ME data
source_CF Component-managed SCASYSME00_Auth_Alias

Removing alias: SCASYSME00_Auth_Alias

Replacing alias reference in data source: SCA Application Bus ME data source
SCAAPME00_Auth_Alias => DB2Alias

Replacing alias reference in SIBus data store of ME:
T5DepEnv.AppTarget.000-SCA.APPLICATION.T5Cell.Bus SCAAPME00_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: SCA Application Bus ME
data source_CF Component-managed SCAAPME00_Auth_Alias

Removing alias: SCAAPME00_Auth_Alias

Replacing alias reference in data source: Business Process Choreographer data source
BPCDB_T5DepEnv.AppTarget_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: Business Process Choreographer
data source_CF Component-managed BPCDB_T5DepEnv.AppTarget_Auth_Alias

Removing alias: BPCDB_T5DepEnv.AppTarget_Auth_Alias

Replacing alias reference in data source: Business Process Choreographer ME data source
BPCME_00_Auth_Alias => DB2Alias

Replacing alias reference in SIBus data store of ME:
T5DepEnv.AppTarget.000-BPC.T5Cell.Bus BPCME_00_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: Business Process Choreographer ME
data source_CF Component-managed BPCME_00_Auth_Alias

Removing alias: BPCME_00_Auth_Alias

Replacing alias reference in data source: Business Process Choreographer reporting function
source BPCEDB_T5DepEnv.AppTarget_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: Business Process Choreographer
reporting function source_CF Component-managed BPCEDB_T5DepEnv.AppTarget_Auth_Alias

Removing alias: BPCEDB_T5DepEnv.AppTarget_Auth_Alias

Replacing alias reference in data source: Business Space data source
BSPACE_Auth_Alias => DB2Alias

Removing alias reference from CMP connection factory: Business Space data source_CF
Component-managed BSPACE_Auth_Alias

Removing alias: BSPACE_Auth_Alias

Saving configuration

ConsolidateJAASAuthAliases: Completed

```

4 番目のスキャン・モード・パラメーターが指定されている場合、Saving configuration のメッセージは、Running in scan mode, no updates committed のメッセージに置換されます。以下に例を示します。

```

./wsadmin.sh -lang jython -host winmvsp1 -port 20502
-f /u/healdr/Jython/ConsolidateJAASAuthAliases.py DB2Alias wsadmin admn4was y

/WebSphere/V7T5DM/DeploymentManager/bin:>./wsadmin.sh -lang jython -host winmvsp 1 -port 20502
-f /u/healdr/Jython/ConsolidateJAASAuthAliases.py DB2Alias wsadmin gadook y
WASX7209I: Connected to process "dmgr" on node T5NodeDmgrMVP1 using SOAP connector;
The type of process is: DeploymentManager
WASX7303I: The following options are passed to the scripting environment and are available as
arguments that are stored in the argv variable: "[DB2Alias, wsadmin, gadook, y]"

ConsolidateJAASAuthAliases: Starting

Created JAAS alias: DB2Alias

```

...

Removing alias: BSPACE_Auth_Alias

Running in scan mode, no updates committed

ConsolidateJAASAuthAliases: Completed

参照

wsadmin および Jython スクリプト記述について詳しくは、以下を参照してください。

- WebSphere for z/OS V6.1 インフォメーション・センター
- WebSphere Process Server および ESB for z/OS インフォメーション・センター
- WebSphere z/OS V6.1 – WSADMIN Scripting Primer (with Jython)
- Using Jython Scripting Language With WSADMIN
- Introduction to Jython Part 1: Java programming made easier
- Introduction to Jython Part 2: Programming essentials

Business Process Management コンポーネントの構成

現在、 WebSphere Process Server インスタンスを始動できる位置にいて、このインスタンスがデータベースに接続されている必要があります。次のステップでは、必須の BPM コンポーネントがすべて構成されていることを確認します。

Business Process Management コンポーネントは、以下の 3 つの方法のいずれかで構成される場合があります。

1. 一部のコンポーネントは、拡張時に構成される場合があります。これは、特にスタンドアロン・サーバーの場合に当てはまります。スタンドアロン・サーバーでは、大半のコンポーネントが拡張時に構成される場合があります。
2. Network Deployment セルでは、デプロイメント環境のビルドおよび生成によって、コンポーネントが構成される場合があります。デプロイメント環境をセットアップすると、サーバー・クラスターがビルドされ、標準トポロジーまたはカスタム・トポロジーに従って、それらのサーバー・クラスター上に BPM コンポーネントが構成されます。スタンドアロン・サーバーではデプロイメント環境を使用できないことに注意してください。
3. 管理アプリケーションを使用して、コンポーネントが個別に構成される場合があります。管理コンソールでは、このタスクを支援する各種ウィザードを使用できます。

ランタイム環境で必要なコンポーネントを決定します。次に、デプロイメント環境 (Network Deployment のみ) または管理アプリケーションを構成して、拡張によってセットアップされなかったコンポーネントを構成します。

デプロイメント環境ウィザードとコンポーネントの手動構成の違い

拡張の完了後に Business Process Management コンポーネントを構成するには、2 つの方法があります。1 つは、デプロイメント環境ウィザードを使用して、クラスターとサーバーを作成し、標準トポロジーまたはカスタム・トポロジーに従ってそれらのクラスターとサーバー全体にコンポーネントをデプロイする方法です。もう

1 つは、管理アプリケーションで提供されている各種ウィザードを使用して、各コンポーネントを個別に構成する方法です。

スタンドアロン・サーバーを構成する場合は、コンポーネントを個別に構成する必要があります。コンポーネントの多くは拡張時に構成されている可能性があることに留意してください。

Network Deployment 環境を構成する場合は、コンポーネントを個別に構成することも、デプロイメント環境を作成および生成することも可能です。デプロイメント環境ウィザードを使用すると、いくつかの事前定義トポロジーに従ってクラスターとサーバーを生成し、それらのクラスターとサーバー全体に複数のコンポーネントをすべて同時に構成することができます。事前定義トポロジーでは要件が満たされない場合は、カスタムのデプロイメント環境トポロジーを作成することもできます。

WebSphere Process Server for z/OS の場合は、単一クラスター・トポロジーから始めることをお勧めします。

デプロイメント環境のビルドに関する考慮事項

- Network Deployment 環境を構成します。
- 管理アプリケーションで単一のウィザードを実行して、複数のコンポーネントを構成します。
- データベース設計ファイルをインポートして、データベース関連リソース定義の値を指定します。データベース設計ファイルの作成については、61 ページの『データベース設計ツールを使用したデータベース設計ファイルの作成』を参照してください。
- 現在の環境にインポートして、必要に応じてカスタマイズできる、事前定義デプロイメント環境が提供されています。

コンポーネントの個別構成に関する考慮事項

- スタンドアロン・サーバーまたは Network Deployment 環境を構成します。
- デプロイメント環境に含まれるコンポーネントをすべて構成するわけではありません。
- 任意のコンポーネントを構成する前に、そのコンポーネントをデプロイするクラスターまたはサーバーを構成します。

Network Deployment 環境の場合、単一クラスター・トポロジーからなるデプロイメント環境を構成することで、大半の WebSphere Process Server for z/OS 要件が満たされます。

デプロイメント環境のセットアップ

デプロイメント環境をセットアップするには、デプロイメント環境定義を作成してから、環境を生成する必要があります。

このタスクについて

デプロイメント環境は、「デプロイメント環境構成」ウィザードを使用するか、または `wsadmin` を使用したスクリプトによって作成できます。デプロイメント環境の作成が終わった後、デプロイメント環境のセットアップを完了する追加タスクを実行できます。

デプロイメント環境の作成方法の選択については、『デプロイメント環境の計画』を参照してください。

関連タスク

デプロイメント環境の計画

デプロイメント環境のセットアップには、物理サーバーの数から選択するパターンのタイプまで、あらゆる事柄に影響を与える多くの決定が関係しています。それぞれの決定はデプロイメント環境をセットアップする方法に影響を与えます。

デプロイメント環境の作成

デプロイメント環境を作成するには、デプロイメント環境定義を作成してから、環境を生成する必要があります。デプロイメント環境は、「デプロイメント環境構成」ウィザード、または `wsadmin` を使用して作成できます。

「デプロイメント環境構成」ウィザードは、デプロイメント環境を構成できるコンポーネントおよびクラスターを構成するための一連のパネルを表示します。「デプロイメント環境構成」ウィザードのパネルで情報を入力し終わってから、「生成」ではなく「終了」をクリックすると、デプロイメント環境定義が作成されます。

「デプロイメント環境構成」ウィザードで「生成」をクリックした場合に限り、環境が「構成」されます。「デプロイメント環境構成」ウィザードでデプロイメント環境定義を生成すると、システムは、生成された定義に含まれるデータに従って、すべてのクラスターおよびコンポーネントを構成します。

デプロイメント環境は、「デプロイメント環境構成」ウィザードで作成できるだけでなく、`wsadmin` スクリプトを使用して作成することもできます。「デプロイメント環境構成」ウィザードを使用する場合と同様に、`wsadmin` 関数でもデプロイメント環境を作成するには 2 つの段階があります。つまり、最初にデプロイメント環境定義を作成し、次にその定義からデプロイメント環境を生成します。

「デプロイメント環境構成」ウィザードを使用したデプロイメント環境の作成:

デプロイメント環境は、「デプロイメント環境構成」ウィザードを使用によって作成できます。

パターンを使用したデプロイメント環境の作成:

デプロイメント・パターンを選択したら、「デプロイメント環境構成」ウィザードを使用して、そのパターンに基づいたデプロイメント環境を作成します。

始める前に

デプロイメント・マネージャーの管理コンソールで、「サーバー」>「デプロイメント環境」にナビゲートします。

この作業に必要なセキュリティ・ロール: セキュリティーおよびロール・ベースの許可が使用可能になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとして管理コンソールにログインする必要があります。

デプロイメント環境ウィザードを使用してデプロイメント環境を作成する手順には、パターンおよびフィーチャーを選択するステップが含まれています。したがって、『計画』のセクションに記載されているパターンおよびフィーチャーに関する情報を読んで理解していることを前提としています。

製品がインストール済みであり、デプロイメント・マネージャー・プロファイルおよび関連ノードが作成済みであることを前提とします。

さらに、「デプロイメント環境構成」ウィザードのステップの 1 つは、データベース設計文書のインポートを含みます。データベース設計文書は、選択されたデプロイメント環境フィーチャーのデータベース構成を定義します。WebSphere Process Server には、ユーザーの入力に基づいてデータベース設計文書を作成する応答式データベース設計ツール (DDT) が用意されています。データベース設計文書を使用して、DDT によってデータベース・スクリプトを作成することや、デプロイメント環境で使用されるデータベースを WebSphere Process Server デプロイメント環境ウィザードによって構成することができます。WebSphere Process Server には、ユーザーの入力に基づいてデータベース設計文書を作成する応答式データベース設計ツール (DDT) が用意されています。データベース設計文書を使用して、DDT によってデータベース・スクリプトを作成することや、デプロイメント環境で使用されるデータベースを WebSphere Process Server デプロイメント環境ウィザードによって構成することができます。DDT およびデータベース構成全般について詳しくは、『データベースの構成』を参照してください。

このタスクについて

このタスクでは、「デプロイメント環境構成」ウィザードを使用して、特定のパターンに基づくデプロイメント環境を作成する手順について説明します。

注: ウィザードで作業中エラーが発生した場合は、「戻る」をクリックして戻ることができます。

手順

1. 管理コンソールで「サーバー」 → 「デプロイメント環境」をクリックして、「デプロイメント環境」ページを表示します。
2. 「デプロイメント環境」ページで「新規」をクリックして、「デプロイメント環境構成」ウィザードを起動します。
 - a. 「パターンに基づくデプロイメント環境の作成」オプションが選択されています。「パターンに基づくデプロイメント環境の作成」はシステム・デフォルトであり、このトピックで説明するオプションです。

デプロイメント環境パターンを利用して、よく使用するビジネス・インテグレーション・トポロジーを記録します。パターンは、作成するデプロイメント環境のテンプレートを提供します。

注: パターンは、構成されたデプロイメント・マネージャーがサポートする製品と直接に関連しています。WebSphere Process Server は特定のパター

ン・セットをサポートします。その中のリモート・メッセージングおよびリモート・サポート・パターンがシステム・デフォルトです。デプロイメント・マネージャーが WebSphere Process Server 以外の製品もサポートしている場合は、追加のパターンを適用できます。特定の製品に適用されるパターンについては、その製品固有の文書を参照してください。

WebSphere Process Server に組み込まれてサポートされているパターンのタイプについては、『計画』セクションの『トポロジー・タイプおよびデプロイメント環境パターン』を参照してください。

「カスタム・デプロイメント・トポロジーの詳細」ページを使用してカスタム・デプロイメント環境を構成する方法については、『カスタム・デプロイメント環境のレイアウト構成』を参照してください。

- b. 「**デプロイメント環境名**」フィールドに、デプロイメント環境の固有の名前を入力します。
- c. **オプション:** ウィザードですべての構成ステップを表示するには、「**詳細: すべてのステップを表示します**」を選択します。

「**ファースト・パス: 必要なステップのみを表示します**」を選択した場合は、ウィザードにはデフォルト値が割り当てられていないページのみが表示されます。「**ファースト・パス: 必要なステップのみを表示します**」は、デプロイメント環境構成でシステム提供のデフォルト値を受け入れることに同意する場合にのみ選択してください。

このトピックでは、「**詳細: すべてのステップを表示します**」を選択したものと想定します。

- d. 「**次へ**」をクリックして、「**デプロイメント環境フィーチャー (Deployment Environment Features)**」ページを表示します。
3. 「**デプロイメント環境フィーチャー**」ページでデプロイメント環境のフィーチャーを選択し、「**次へ**」をクリックして、互換性のあるフィーチャーのリストを表示するか、デプロイメント環境パターンのリストを表示します。フィーチャーはデプロイメント環境のランタイム処理機能を表します。

「**デプロイメント環境フィーチャー**」ページの使用可能なフィーチャーのリストは、デプロイメント・マネージャー・プロファイルに基づきます。デプロイメント・マネージャー・プロファイルを拡張して WebSphere Process Server と一緒に別の製品を組み込んだ場合 (WebSphere Business Monitor や WebSphere Business Services Fabric など)、「**デプロイメント環境フィーチャー**」ページにはこれらのフィーチャーもリストされます。

WebSphere Process Server のプロファイルをインストールして構成した場合、「**デプロイメント環境フィーチャー**」ページには以下が含まれます。

- **WESB.** WebSphere Enterprise Service Bus 用。メディエーションをサポートするデプロイメント環境を提供します。
- **WPS.** WebSphere Process Server 用。メディエーション、ビジネス・プロセス、ヒューマン・タスク、およびビジネス・ルールをサポートするデプロイメント環境を提供します。

デプロイメント環境フィーチャーのデフォルト値は、デプロイメント・マネージャーのランタイム機能に一致します。

4. 「互換性のあるデプロイメント環境フィーチャーの選択」 ページで必要に応じて追加のフィーチャーを選択し、「次へ」をクリックして、1 次フィーチャーおよび補助フィーチャーの選択に関連したパターンのリストを表示します。

注: 「互換性のあるデプロイメント環境フィーチャーの選択」 ページは、デプロイメント・マネージャーが他のビジネス・プロセス・マネジメント (BPM) フィーチャー (WebSphere Business Monitor など) で拡張されている場合にのみ表示されます。

フィーチャーと互換フィーチャーの関係を理解するには、『計画』セクションのデプロイメント環境に関する説明を参照してください。

5. 「デプロイメント環境パターンの選択 (Select the deployment environment pattern)」 ページで、選択したデプロイメント環境のパターンを選択し、「次へ」をクリックして「ノードの選択」 ページを表示します。

「デプロイメント環境パターン」 ページに表示されるパターンのリストは動的です。このリストは、以下の環境条件および構成に関する決定によってアクティブになるため、その内容はこれらの条件および決定に依存します。

- ソフトウェアをインストールしたプラットフォーム
- 「デプロイメント環境フィーチャーの選択」 ページおよび「互換性のあるデプロイメント環境フィーチャーの選択」 ページで行った選択

パターンからフィーチャーへ関係について詳しくは、『トポロジー・パターンおよびサポート対象の BPM 製品フィーチャー』を参照してください。

6. オプション: 「ノードの選択」 ページでこのデプロイメント環境に含めるノードを選択し、「次へ」をクリックして「クラスター」 ページを表示します。

デプロイメント環境用に少なくとも 1 つのノードを選択します。高可用性環境およびフェイルオーバー環境では、少なくとも 2 つのノードを選択します。スケラビリティを実現するには、すべてのノードを選択します。

ノードを組み込むには、ノード名の隣にあるチェック・ボックスを選択します。選択したノードを別のノード名にマップするには、「ノード・マッピング」を使用します。

7. オプション: 「クラスター」 ページで、デプロイメント環境のクラスター・タイプ (アプリケーション・デプロイメント・ターゲット、メッセージング・インフラストラクチャー、サポート・インフラストラクチャー) ごとに各ノードに必要な数のクラスター・メンバーを割り当てます。

デフォルトでは、機能ごとに各ノード上で 1 つのクラスター・メンバーが割り当てられます。数値を変更するには、各列の数値を置き換えます。それぞれのクラスター・タイプで提供される各種のクラスター・ロールおよび機能について十分な知識がない場合は、『トポロジー・タイプおよびデプロイメント環境パターン』を参照してください。

ノードの値が 0 (ゼロ) の場合、選択したフィーチャーに関して、その選択した機能にノードが寄与しないことを意味します。

クラスター・メンバーを割り当てた後、「次へ」をクリックすると、デプロイメント環境のクラスター・タイプごとに「クラスター命名」ページを表示することができます。表示される「クラスター命名」サブステップは、選択したデプロイメント環境パターンによって異なります。

クラスター名およびクラスター・メンバー名のデフォルト値は、システムによって生成されます。また、クラスター・ショート・ネームおよびクラスター・メンバー・ショート・ネームのデフォルト値もシステムによって生成されます。

クラスター名またはクラスター・メンバー名をカスタマイズしない場合は、以下の手順で、ウィザード・ナビゲーション・ペインを使用して、直接、「REST サービス」ページに移動します。

それぞれのサブステップ・ページは、同じ構造であり、『クラスター名およびクラスター・メンバー名をカスタマイズします』で説明しています。

- a. オプション: クラスター名およびクラスター・メンバー名をカスタマイズします。

「クラスター命名」ページを使用して、クラスター・タイプのクラスター名またはクラスター・メンバー名をカスタマイズします。クラスター・ショート・ネームおよびクラスター・メンバー・ショート・ネームを変更することもできます。選択したパターンのクラスター・タイプごとに1つのサブステップ・ページがあります。例えば、「リモート・メッセージングおよびリモート・サポート」パターンを選択した場合は、3つのサブステップがあります。すなわち、そのパターンのクラスター・タイプ (アプリケーション・デプロイメント・ターゲット、メッセージング・インフラストラクチャー、およびサポート・インフラストラクチャー) ごとに1つのサブステップがあります。

それぞれのサブステップ・ページの情報、は、以下のとおりです。

クラスター

クラスターの機能ロールを指定する読み取り専用フィールド。

値は、以下のようにクラスター・タイプによって異なります。

- アプリケーション・デプロイメントのターゲット
- サポート・インフラストラクチャー
- メッセージング・インフラストラクチャー

クラスター・タイプごとの機能ロールに関する情報については、トポロジーおよびデプロイメントの環境パターンを参照してください。

クラスター名

クラスター名のシステム生成デフォルト値が入ります。

デフォルト値は、<デプロイメント環境名>.<クラスター・タイプ名> という命名規則に従っています。ここで、クラスター・タイプ名は、以下のいずれかの値です。

- AppTarget

アプリケーション・デプロイメントのターゲットのロールを実行する
クラスターの場合

- メッセージング

メッセージング・インフラストラクチャーのロールを実行するクラス
ターの場合

- サポート

サポート・インフラストラクチャーのロールを実行するクラスター
の場合

- Web

サポート Web アプリケーションのロールを実行するクラスター
の場合

注: このクラスター・タイプ名は、WebSphere Business Monitor が主
なフィーチャーまたは製品である BPM 構成に適用されます。

クラスター・ショート・ネーム

このフィールドは、空白のままにすることも、任意のショート・ネ
ームを入力することもできます。

クラスター・ショート・ネームは、8 文字を超えることはできず、名前
の競合を避けるため、固有でなければなりません。クラスター・ショ
ート・ネームの作成時には、以下の規則と照らし合わせて妥当性がチェッ
クされます。

- 長さは 1 文字から 8 文字であること。
- 含まれているのは英数字または各国語の文字のみであること。
- 先頭が数値ではないこと。
- セル内で固有であること。
- 非クラスター・サーバーの **ClusterTransitionName** カスタム・プロパ
ティーで指定されている値と同じではないこと。

クラスター・メンバー名

システム生成デフォルト値を受け入れるか、任意の名前を指定します。

クラスター・メンバー名のデフォルト値は、<クラスター名>.<ノード名
>.<ノード番号シーケンス> という命名規則に従っています。

表に表示されるクラスター・メンバー名の数は、「クラスター」ペー
ジのクラスター・タイプ列およびノード行に入力したクラスター・メンバ
ーの数と一致します。「クラスター」ページについては、前のステップ
を参照してください。

クラスター・メンバー・ショート・ネーム

システム生成デフォルト値を受け入れるか、任意の名前を指定します。

クラスター・メンバー・ショート・ネームのシステム生成値は、<デプ
ロイメント環境名>[0:5]<クラスター・タイプ名> という命名規則に従っ
ています。

クラスター・メンバー・ショート・ネームは、最大 7 文字で、固有でな
ければなりません。

クラスター・メンバー・ショート・ネームが固有でない場合は、システムによって名前に固有の文字が付加されます。

例えば、デプロイメント環境の名前が DEMO である場合、システムが生成するアプリケーション・ターゲット・クラスター・メンバーのショート・ネームは、DEMOAT になります。

以下の構成条件が存在するときは、クラスター・メンバー・ショート・ネームのオプションが表示されます。

- セル内で認識されているノードのいずれかが z/OS プラットフォーム上にある場合は、クラスター・メンバー・ショート・ネームが表示されます。ノードのメタデータは、ノードが存在するプラットフォームをサポートする必要があります。
- デプロイメント・マネージャーが z/OS プラットフォームにある場合。

8. 「REST サービス」ページで、Representational State Transfer (REST) アプリケーション・プログラミング・インターフェース (API) のサービス・エンドポイントを構成します。

ウィジェットを Business Space で使用できるようにしたい場合は、それらのウィジェットについて REST サービス・エンドポイントを構成する必要があります。

- a. 「プロトコル」リストから「<https://>」または「<http://>」を選択することにより、すべての REST サービスのフル URL パスを構成します。
 - b. 「負荷分散環境でのホスト名または仮想ホスト」フィールドに名前を入力します。
 - c. 「ポート」フィールドに、クライアントがサーバーまたはクラスターと通信するために必要なポートを入力します。
 - d. REST サービス・エンドポイントの説明を変更する場合は、REST サービスの表の「説明」フィールドの項目を上書きします。その他のフィールドは読み取り専用です。
 - e. 「次へ」をクリックして、「データベース構成のインポート」ページに進みます。
9. オプション: 「データベース構成のインポート」ページで、「参照」をクリックしてデータベース設計文書に進むか、またはデータベース設計文書へのパスを入力した後、「次へ」をクリックして、「データ・ソース」ページに進みます。設計文書は、データベース設計ツール (DDT) を使用して作成したデータベース設計に基づく設計文書にすることも、選択したパターンとフィーチャーに基づいて提供された設計文書にすることもできます。

注: プロファイルの作成時に作成された commonDB は、デプロイメント環境用にインポートされたデータベース設計文書によって変更されることはありません。

10. 条件付きオプション: 「データベース」ページでデプロイメント環境のデータ・ソースのデータベース・パラメーターを構成し、「次へ」をクリックして「セキュリティ」ページに移動します。

このページで、このデプロイメント環境に組み込まれたコンポーネントに対してデータベース情報を定義します。可能であれば、ウィザードにパラメーターのデフォルト情報が表示されますが、環境の計画時に定義した値に合うようにそれらの値を変更してください。

注: データベース設計文書をインポートしてある場合は、その文書に存在するデータ・ソース構成が「データベース」ページの情報に反映されます。

ファースト・パス・デプロイメント環境構成用にこのステップが表示されるかどうかは、条件に依存します。複数のデータベースが定義されている場合は、ファースト・パス・デプロイメント環境構成用にこのステップが表示されません。

DB2 for z/OS または Oracle データベース・プロバイダーを使用している場合には、このステップが常に表示されます。

このページに表示されるデフォルト・スキーマ名は、お客様のサイトの命名規則と矛盾したり、既存のスキーマと矛盾したりする場合があります。そのような場合は、スキーマ名の変更が必要になります。

Oracle データベースの考慮事項:

- Oracle の使用時にすべてのコンポーネントに対して 1 つの DBA ユーザー名とパスワードを指定したくない場合は、「**テーブルの作成**」をクリアし、コンポーネントごとに以前から存在する固有のユーザー名とパスワードを指定します。すべてのコンポーネントに対して 1 つの DBA ユーザー名とパスワードを指定できる場合は、「**テーブルの作成**」を選択し、必要なスキーマとユーザーを構成プロセスで作成できるようにします。

実稼働環境では、「**ユーザー名**」と「**スキーマ名**」に同じ値を設定し、「**テーブルの作成**」の選択は解除してください。実稼働環境では、必要なスキーマを手動で作成し、生成された SQL ファイルを使用してテーブルを作成します。

注: Business Space では、「**テーブルの作成**」を選択できません (このオプションは選択不可になっています)。Business Space 用の SQL ファイルは、手動で実行する必要があります。Business Space 用の SQL を手動で実行する場合の情報については、「*Business Space* のデータベース表の構成」を参照してください。

キー・パラメーター (データベース名、テーブル作成の有無、データ・ソースの実行時ユーザー名、デプロイメント環境のパスワードなど) は、すべて編集することができます。

特定のコンポーネントに使用するデータベースを選択できます。

DB2 for z/OS: DB2 for z/OS データベース・プロバイダーを使用している場合、「**テーブルの作成**」オプションは使用できません。

「デプロイメント環境構成」ウィザードでは実行できず、手動で実行する必要のあるステップは、「**据え置かれた構成**」ページにリスト表示されます。

11. 「**セキュリティ**」ページで、セキュア・コンポーネントへのアクセス時に WebSphere が使用する認証別名を構成します。

認証別名のユーザー名とパスワードは、このページで変更することができます。これらの別名を使用してセキュア・コンポーネントにアクセスしますが、データ・ソースにアクセスすることはできません。

12. 「Business Process Choreographer」ページで、Business Process Choreographer 構成のパラメーターを設定し、「次へ」をクリックして「システム Web アプリケーション (System web applications)」ページを表示します。このページで、以下の値を指定します。

- セキュリティー・ロール
- 認証別名

13. オプション: 「システム Web アプリケーション (System web applications)」ページで、デプロイメント環境のコンポーネント・ベース Web アプリケーションのコンテキスト・ルートを設定するか、システム提供のコンテキスト・ルートのデフォルト値を受け入れます。「次へ」をクリックして、「要約」ページを表示します。

「システム Web アプリケーション」ページは、デプロイメント環境でリモート・メッセージング、サポート、および Web アプリケーション・パターンを使用している場合に表示されます。リモート・メッセージング、サポート、および Web アプリケーション・パターンが適用されるのは、デプロイメント環境が WebSphere Business Monitor を組み込むために拡張されたデプロイメント・マネージャー用のものである場合です。

テーブルには以下のコントロール情報が含まれています。

Web アプリケーション

Web アプリケーションの名前。

作成しているデプロイメント環境を構成するコンポーネントのなかには、作成しているデプロイメント環境を構成するコンポーネントのなかには、Web アプリケーションが含まれるものがあります。「Web アプリケーション」列には、以下のコンポーネントが含まれます。

- Business Space
- Business Process Choreographer Explorer
- ビジネス・ルール・マネージャー

コンテキスト・ルート

コンポーネントのコンテキスト・ルートの現行値。

デフォルトでは、Web アプリケーションのデフォルトのコンテキスト・ルートが適用されます。「コンテキスト・ルート」フィールドの値に上書きして、コンテキスト・ルートを変更できます。

注: Business Space のコンテキスト・ルートは読み取り専用であるため、編集することはできません。

説明

Web アプリケーション・コンテキスト・ルートの説明。

14. 「要約」ページの情報が正しいことを確認したあと、「終了して環境を生成」をクリックして、デプロイメント環境の構成を保存して完了します。構成を完了せずに終了するには、「終了」をクリックします。

「終了」をクリックすると、デプロイメント環境構成が保存されますが、生成はされません。

「取り消し」をクリックすると、デプロイメントの構成が取り消され、構成は保存されません。

- a. 据え置かれた構成ステップがないかどうかを確認します。

「デプロイメント環境」 → デプロイメント環境の名前 → 「据え置かれた構成」を選択します。

据え置かれた構成ステップが存在する場合は、デプロイメント環境を始動する前に、そのステップを処理する必要があります。

15. z/OS システムの命名規則に従い、サーバー・ショート・ネームを変更します。

サーバー・ショート・ネームは 7 文字以内で指定する必要があります。これは、実行時に S または A をショート・ネームに追加して、サーバント領域または付属領域を指定できるようにするためです。

z/OS のクラスター・ショート・ネームを変更するには、以下の手順を実行します。

- a. 管理コンソールから、「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」にナビゲートします。
- b. 変更するサーバーの名前をクリックします。
- c. 新しいサーバー・ショート・ネームを「ショート・ネーム」領域に入力します。

例えば、ショート・ネームを BBOS001 から WZxxZ1 に変更します。この xx は、セルの接頭部です（「WZT5Z1」など）。

z/OS のサーバー命名規則について詳しくは、WebSphere Application Server インフォメーション・センターを参照してください。

タスクの結果

構成が完了すると、構成ファイルを調べて、変更を表示できます。

次のタスク

変更内容をマスター構成に保存するか、破棄します。

関連概念

デプロイメント環境

デプロイメント環境とは、Service Component Architecture (SCA) の対話をホストするための環境を共同して提供する、構成済みのクラスター、サーバー、およびミドルウェアの集合のことです。例えば、デプロイメント環境には、メッセージの宛先用のホスト、ビジネス・イベントの処理プログラム、および管理プログラムが組み込まれている場合があります。

トポロジーおよびデプロイメントの環境パターン

トポロジーには、さまざまなレイアウトがあります。WebSphere Process Server をインストールして構成する前に、このセクションの情報を確認してください。トポロジーの概念を理解しておく、製品のインストールおよび構成方法について、知識に基づいた判断を行うのに役立ちます。

関連タスク

デプロイメント環境実装の汎用ステップ

デプロイメント環境を設計したら、設計を実現するための作業を行います。デプロイメント環境の実装に使用する方法にかかわらず、同じ汎用の手順を行います。

125 ページの『デプロイメント環境の据え置かれた構成の設定』

データベースとテーブルの作成を据え置く必要がある場合は、「据え置かれた構成」ページを使用してください。このページには、データベースとテーブルを作成するためのスクリプトを見つけて実行する方法の説明が示されています。

61 ページの『データベース設計ツールを使用したデータベース設計ファイルの作成』

データベース設計ツール (DDT) を使用すると、WebSphere Process Server に必要なデータベース表の作成に使用する設計ファイルを生成できます。DDT では、ユーザー指定のプロパティ・ファイルまたは対話式のユーザー入力に基づいて設計ファイルが生成されます。生成された設計ファイルは、DDT がデータベース表の作成に使用されるデータベース・スクリプトを作成するのに使用します。また、プロファイル作成時やデプロイメント環境構成時の入力として設計ファイルを使用し、データベース構成プロパティを指定することもできます。

116 ページの『カスタム・デプロイメント環境の構成』

カスタム・デプロイメント環境を構成するには、「カスタム・デプロイメント・トポロジーの詳細」ページを使用します。

関連情報

z/OS アプリケーション・サーバーの命名規則

Business Process Choreographer の構成

カスタム・デプロイメント環境のレイアウト構成:

この概要では、カスタム・デプロイメント環境の構成における 2 つの主要な考慮事項について説明します。環境で使用するクラスターおよび単一サーバーの選択と、デプロイメント環境構成の指定です。これらの考慮事項を理解すれば、デプロイメント環境を効率的に計画および実装できます。

『デプロイメント環境で使用するクラスターと単一サーバーの選択』では、デプロイメント環境を構成するクラスターおよびサーバーを定義します。機能ごとにクラスターを作成するパターン・デプロイメント環境とは異なり、カスタム・デプロイメント環境では、機能を実行するために必要なクラスターとサーバーを追加します。

111 ページの『デプロイメント環境構成の定義』では、クラスターとサーバーに構成する機能について説明します。これらの機能は、メッセージング、Common Event Infrastructure、およびアプリケーション・サポートです。

システム内でデプロイメント環境構成を生成して構成を完了する前であれば、構成に戻って変更することができます。システム内でデプロイメント環境構成を生成した後でも、現行構成を確認できます。また、サーバーとクラスターを追加したり、追加の機能を構成したりすることもできます。このデプロイメント環境での管理対象からサーバーとクラスターを除去することもできます。生成済みの機能構成を取り消すことや、デプロイメント環境の別のサーバーまたはクラスターにまだ必要なデプロイメント環境定義からサーバーまたはクラスターを除去することはできません。

すべてのカスタム・デプロイメント環境における要件

カスタム・デプロイメント環境レイアウトには、以下の制限があります。

- ユーザーがデプロイメント環境を生成して構成を完了すると、関連するコントロールがチェックされ、使用不可になります。これは、構成を元に戻すことができないことを意味しています。
- デプロイメント環境を生成後、コンポーネントのコントロールがチェックされず、使用不可にならない場合は、次の順番で関連する機能を構成する必要があります。関連するメッセージング・エンジンを構成して、Common Event Infrastructure (CEI) を構成し、次にアプリケーション・サポート (このトピックの以降で説明します) を構成します。
- システムに存在する構成は、トポロジー・レイアウト構成よりも優先されます。したがって、カスタム・トポロジーをエクスポートすると、トポロジーに含まれるサーバーの実際の構成が反映されます。

管理コンソールの「トポロジー・レイアウト」ページには、カスタム・トポロジーにおいて構成する必要のある 4 つのセクションがあります。

- クラスターおよび単一サーバーの選択
- メッセージング
- Common Event Infrastructure
- コンポーネント

以下のセクションで、カスタム・トポロジー・レイアウト構成を完了するためのその他の要件を説明しています。

デプロイメント環境で使用するクラスターと単一サーバーの選択

「トポロジー・レイアウト」ページの「クラスターおよび単一サーバーの選択」セクションを使用して、デプロイメント環境内のクラスターとサーバーを管理し、クラスターとサーバーが提供する機能を定義します。

「トポロジー・レイアウト」ページの「クラスターおよび単一サーバーの選択」セクションには、デプロイメント環境の一部として構成する、使用可能なクラスターおよびサーバーのリストが含まれています。機能構成のコラボレーション単位にクラスターとサーバーを割り当てます。各コラボレーション単位は、全体としてデプロイメント環境内で 1 つの機能を提供するクラスターおよびサーバーのグループを表します。デプロイメント環境からクラスターまたはサーバーを削除することもできます。ただし、除去できるクラスターおよびサーバーは、構成内の他のクラスターまたはサーバーで必要としなくなったもののみです。

デプロイメント環境構成の定義

「トポロジー・レイアウト」ページの「デプロイメント環境の構成の指定」セクションを使用して、デプロイメント環境の特定の機能に参加するクラスターまたはサーバーを定義します。

メッセージング

注: 分割されたメッセージング・エンジンはサポートされません。

選択したターゲットのメッセージング宛先を構成するには、「メッセージング」タブのフィールドを使用します。各テーブルはコラボレーション単位を表し、「メッセージング」セクションには複数のテーブルを含むことができます。それぞれのユニットのローカル構成のオプションには、ターゲット (クラスター/サーバー) を 1 つのみ選択する必要があります。このユニットのその他のすべてのターゲットでは、リモート宛先が想定されます。アプリケーションがリモート宛先構成のターゲットにメッセージを送信する際、システムはメッセージをそのユニットのローカル・ターゲットにルーティングします。

メッセージング構成は、Service Component Architecture (SCA)、CEI、Business Process Choreographer システム・バスに適用されます。

トポロジー構成内のローカル宛先との競合を避けるために、以下のルールが適用されます。

- SCA システム・バスのメッセージング・エンジン構成により、ローカル宛先とリモート宛先のロケーションが決定されます。SCA アプリケーション、CEI バス構成、および Business Process Choreographer バス構成は、SCA システム・バス構成に従います。
- 他のバス用のメッセージング・エンジンを単位内の別のターゲットに配置する場合、その単位内の他のターゲットはリモート宛先の役割であると想定されます。CEI または Business Process Choreographer バスの構成が異なる場合、情報メッセージは指定されたバスのメッセージング・エンジンが SCA メッセージング・エンジンと同じターゲットに見つからないことを示します。
- 追加しようとしているターゲットにある既に構成されているリモート宛先またはローカル宛先が、指定のユニットの現在のバス設定と競合する場合、システムはエラー・メッセージを生成します。

Common Event Infrastructure

CEI タブに CEI を構成します。「メッセージング」と同じです。CEI は複数のテーブルを保有でき、各テーブルが単位を表します。各テーブルで、CEI クラスターまたはサーバーを 1 つ選択します (「クラスター/サーバー」列)。これ

は、「サーバー」ラジオ・ボタンを選択すると、サーバーとして機能します。サーバーとして構成されていないすべてのターゲットでは、宛先ロールが想定されます。対応するターゲットでは、このターゲットで発行される Common Base Event が該当するコラボレーション単位のサーバーに送信されるように、Event Infrastructure エミッター・ファクトリーの Java Naming and Directory Interface (JNDI) 名が構成されています。

アプリケーション・サポート

「アプリケーション・サポート」タブには、特定のデプロイメント・ターゲットに対して構成可能なすべてのコンポーネントがリストされます。コンポーネントの機能を関連するコラボレーション単位内で構成します。例えば、単位内で構成されている Business Process Choreographer Container から発行される Common Base Event を収集するには、同じ単位内で Business Process Choreographer Event Collector を構成します。各コンポーネント構成には、他のコンポーネント構成に対する要件と依存関係があります。依存関係は、クリアされて使用不可のコントロールにより表現されます。これらのコントロールを使用可能にするには、まず依存するコントロールを構成する必要があります。

注: 依存コントロールは、「メッセージング」タブまたは「CEI」タブに構成されます。

表 8 に、コンポーネント間の関係を示します。

表 8. デプロイメント環境のコンポーネントの関係

コンポーネント	目的	関連コンポーネント	考慮事項
サービス・コンポーネント・アーキテクチャ (SCA)	SCA アプリケーション・サポートのためのデプロイメント・ターゲットを構成します。 SCA システムおよびアプリケーションのバス・メンバーは、対応するメッセージング構成がローカルの場合はローカルに構成されます。それ以外の場合は、対応するメッセージング単位の指定に基づき、リモート宛先を使用してリモートに構成されます。	メッセージング	メッセージング用のデプロイメント・ターゲットを構成していない場合は、SCA 構成は使用できません。

表 8. デプロイメント環境のコンポーネントの関係 (続き)

コンポーネント	目的	関連コンポーネント	考慮事項
Business Process Choreographer Container	<p>ビジネス・フローと ヒューマン・タスク の両方をサポートす るためのデプロイメ ント・ターゲットを 構成します。</p> <p>この構成は、Business Process Choreographer システム・バスのセ ットアップ用の SCA 構成に準拠します。</p>	<p>メッセージング Service Component Architecture Business Process Choreographer Explorer</p>	<p>デプロイメント・タ ーゲットがメッセー ジングに構成されて いないか、またはデ プロイメント・ター ゲットが Service Component Architecture のサポー トのために構成され ていない場合は、 Business Process Choreographer 構成は 使用できません。</p> <p>1 つのコラボレーシ ョン単位により、1 つの Business Process Choreographer 構成が サポートされます。 「アプリケーション・サ ポート」 タブ で、必要なだけ単位 を追加します。</p> <p>Container を管理する 場合は、Business Process Choreographer Explorer を構成する ことを検討してくだ さい。</p>

表 8. デプロイメント環境のコンポーネントの関係 (続き)

コンポーネント	目的	関連コンポーネント	考慮事項
Business Process Choreographer Explorer	<p>選択されているデプロイメント・ターゲットで Business Process Choreographer Explorer を構成します。</p> <p>Business Process Choreographer Explorer は、同じコラボレーション単位内で構成されている Business Process Choreographer Container を管理する Web アプリケーションです。</p> <p>これには、以前は <i>Business Process Choreographer Observer</i> と呼ばれていたオプションのレポート作成機能 (Business Process Choreographer Explorer レポート作成) が組み込まれています。</p>	Business Process Choreographer Container	<p>同じコラボレーション単位内の Business Process Choreographer Container 構成を選択すると、Business Process Choreographer Explorer 構成が使用可能になります。</p> <p>Web アプリケーション・サポート用にデプロイメント・ターゲットを構成する必要があります。</p> <p>任意のデプロイメント・ターゲットで、必要な数の Business Process Choreographer Explorer のインスタンスを構成できます。コンテナが構成されているコラボレーション単位にデプロイメント・ターゲットを追加し、Business Process Choreographer Explorer の構成コントロールをチェックします。</p>

表 8. デプロイメント環境のコンポーネントの関係 (続き)

コンポーネント	目的	関連コンポーネント	考慮事項
Business Process Choreographer Event Collector	<p>選択されているデプロイメント・ターゲットで Business Process Choreographer Event Collector を構成します。</p> <p>Business Process Choreographer Event Collector は、同じコラボレーション単位内で構成されている Business Process Choreographer Container から発行される Common Base Event を収集します。監視されたコンテナに関する統計情報は、データベースに記録されます。</p>	Business Process Choreographer Container Common Event Infrastructure	<p>最初に、Business Process Choreographer Event Collector に使用するのと同じデプロイメント・ターゲットで Common Event Infrastructure サーバーを構成します。Business Process Choreographer Event Collector を使用可能にするには、同じコラボレーション単位内で Business Process Choreographer Container を構成する必要があります。</p> <p>特定の Business Process Choreographer Container を監視する必要があるかどうかはわからない場合は、この機能を後で構成できます。</p>
ビジネス・ルール・マネージャー	<p>選択したデプロイメント・ターゲットにビジネス・ルール・マネージャーを構成します。</p> <p>ビジネス・ルール・マネージャーを使用すれば、ビジネス・プロセスの動作が決定されるビジネス・ルールを構成することができます。</p>	Service Component Architecture	<p>同じデプロイメント・ターゲットで SCA サポートを構成すると、ビジネス・ルール・マネージャー構成コントロールが使用可能になります。</p> <p>ビジネス・ルール・マネージャーは、1つのデプロイメント環境で1つのみ構成できます。</p> <p>1つのインスタンスでセル全体のビジネス・ルール構成を管理できるため、システムで構成する必要があるビジネス・ルール・マネージャーは1つのみです。</p>

カスタム・デプロイメント環境の構成:

カスタム・デプロイメント環境を構成するには、「カスタム・デプロイメント・トポロジーの詳細」ページを使用します。

始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。

デプロイメント・マネージャーの管理コンソールから、「サーバー」→「デプロイメント環境」→「*deployment_environment_name*」→「プロパティ」→「カスタム・デプロイメント・トポロジーの詳細」にナビゲートします。

この作業に必要なセキュリティ・ロール: セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

制約事項:

- システムに存在する構成は、デプロイメント環境の構成より優先されます。したがって、カスタム・デプロイメント環境をエクスポートすると、デプロイメント環境に含まれるサーバーの実際の構成が反映されます。
- コンポーネント・ユニットを構成する前にメッセージング・ユニットを構成する必要があります。チェック・ボックスが使用不可になっている場合、メッセージング・サポートがまだ構成されていません。

このタスクについて

カスタム・デプロイメント環境の場合、必要に応じてそれぞれの機能を構成する方法を決定できます。各機能は、クラスターまたは単一サーバーに対して構成します。カスタム・デプロイメント環境トポロジーの構成には、大きく分けて 3 つの領域があります。

- コンポーネント内通信をサポートするメッセージング。
- イベントおよびモニター機能を統合する Common Event Infrastructure。
- ビジネス・インテグレーションのサービス・コンポーネント (ビジネス・プロセスやヒューマン・タスクなど) をサポートするアプリケーション・サポート。

詳しくは、『カスタム・デプロイメント環境のレイアウト構成の概要』を参照してください。

手順

1. 「このデプロイメント環境で使用するクラスターとサーバーの選択」のリストから、クラスターまたはサーバーを選択します。
2. 「追加」をクリックします。下のテーブルにクラスターまたは単一サーバーが追加されます。
3. このデプロイメント環境に必要なクラスターとサーバーをすべて選択するまで、ステップ 1 と 2 を繰り返します。
4. 「メッセージング」タブを選択します。

- a. デプロイメント環境に必要な個別のメッセージング単位の数を決定し、「**新規単位の追加**」をクリックしてその数を追加します。

各単位に「メッセージング単位 x 」という名前が付けられます。この x は、単位の番号です。

- b. ステップ 2 (116 ページ) で作成されたテーブルからクラスターとサーバーを各単位に割り当てます。

単位に追加するクラスターまたはサーバーを選択してから、「**選択したものを単位に追加**」から単位を選択します。

- c. 各単位でどのデプロイメント・ターゲットがローカル・メッセージング・サポートをホストするかを決定し、単位内の該当デプロイメント・ターゲットを定義する行の「**ローカル・バス・メンバー**」をクリックし、ローカル・メッセージング・ホストを構成します。

その他のクラスターまたはサーバーはすべて、リモート・メッセージング宛先として自動的に構成されます。

5. 「**Common Events Infrastructure**」タブをクリックします。

- a. デプロイメント環境に必要な個別の Common Event Infrastructure 単位の数を決定し、「**新規単位の追加**」をクリックしてその数を追加します。

各単位に「Common Event Infrastructure 単位 x 」という名前が付けられます。この x は、単位の番号です。

- b. ステップ 2 (116 ページ) で作成されたテーブルからクラスターとサーバーを各単位に割り当てます。

単位に追加するクラスターまたはサーバーを選択してから、「**選択したものを単位に追加**」から単位を選択します。

- c. 各単位でどのデプロイメント・ターゲットが Common Event Infrastructure サーバーをホストするかを決定し、単位内の該当デプロイメント・ターゲットを定義する行の「**サーバー**」をクリックし、Common Event Infrastructure サーバーを構成します。

その他のクラスターまたはサーバーはすべて、リモート Common Event Infrastructure 宛先として自動的に構成されます。

6. 「**アプリケーション・サポート**」タブをクリックします。このタブには、指定のデプロイメント・ターゲットに対して構成可能なコンポーネントがすべて表示されます。

制約事項: このセクションでコンポーネントを構成する前に、各コンポーネントのメッセージング・ユニットを完全な状態にしておく必要があります。例えば、Service Component Architecture についてチェック・ボックスが使用不可の場合は、関連するメッセージング・ユニットが構成されていません。制限については、『カスタム・デプロイメント環境のレイアウト構成の概要』を参照してください。

- a. デプロイメント環境に必要な個別のアプリケーション・サポート単位の数を決定し、「**新規単位の追加**」をクリックしてその数を追加します。

必要な単位の数は、必要な Business Process Choreographer Container の数によって決まります。Business Process Choreographer Container が不要な場合は、Service Component Architecture アプリケーションには 1 つの単位で十分です。

各単元に「アプリケーション・サポート単位 x 」という名前が付けられます。この x は、単位の番号です。

- b. ステップ 2 (116 ページ) で作成されたテーブルからクラスターとサーバーを各単元に割り当てます。

単元に追加するクラスターまたはサーバーを選択してから、「**選択したものを単元に追加**」から単位を選択します。

- c. ユニット内で、デプロイメント環境の各コンポーネントに属させるクラスターまたはサーバーを選択します。
- d. 各ユニットでデプロイメント環境に必要なすべてのコンポーネントが構成されるまで、ステップ 6b と 6c を繰り返します。

次のタスク

デプロイメント環境が完成するか、または既存のデプロイメント環境を編集した後、「カスタム・デプロイメント環境の構成 (Custom Deployment Environment Configuration)」ウィザードが開きます。情報を検討して、必要があれば変更を加えることができます。

関連概念



デプロイメント環境

デプロイメント環境とは、Service Component Architecture (SCA) の対話をホストするための環境を共同して提供する、構成済みのクラスター、サーバー、およびミドルウェアの集合のことです。例えば、デプロイメント環境には、メッセージの宛先用のホスト、ビジネス・イベントの処理プログラム、および管理プログラムが組み込まれている場合があります。



トポロジーおよびデプロイメントの環境パターン

トポロジーには、さまざまなレイアウトがあります。WebSphere Process Server をインストールして構成する前に、このセクションの情報を確認してください。トポロジーの概念を理解しておく、製品のインストールおよび構成方法について、知識に基づいた判断を行うのに役立ちます。

コマンド行を使用したデプロイメント環境の作成:

`wsadmin` を使用して、デプロイメント環境を作成できます。

`createDeploymentEnvDef` および `generateDeploymentEnv` は、デプロイメント環境ウィザードを使用してデプロイメント環境を作成するのと同等のコマンド行を提供します。

コマンド行を使用したデプロイメント環境定義へのノードの追加:

`wsadmin.sh` コマンドを使用して、デプロイメント環境定義にノードを追加することができます。

始める前に

このタスクでは、ノードがデプロイメント・マネージャーに統合されていることを前提としています。

トポロジーが既に構成されている場合、このコマンドを実行しても、デプロイメント環境定義にノードは追加されません。

ノードを追加する場合は、ノードの追加先となるデプロイメント・マネージャーを起動しておく必要があります。

この作業に必要なセキュリティ・ロール: セキュリティとロール・ベースの許可が有効になっている場合、管理者権限またはオペレーター権限を持つユーザー ID とパスワードを使用してこのタスクを実行する必要があります。

このタスクについて

このタスクでは、`wsadmin.sh` コマンドを使用して、統合ノードをデプロイメント環境定義に追加します。

手順

1. コマンド・ウィンドウを開きます。

`wsadmin.sh` コマンドは、`<WPS>/profiles/<dmgr profile>/bin` ディレクトリーか `<WPS>/bin` ディレクトリーのいずれかに格納されています。

2. コマンド・プロンプトから `wsadmin.sh` コマンドを入力して `wsadmin.sh` 環境に入ります。
3. `addNodeToDeploymentEnvDef` コマンドを入力して、デプロイメント環境定義にノードを追加します。

注: 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力画面が表示されます (コマンド内で指定していない場合)。

例

この例では、管理セキュリティが有効な状態で、ノード (**MyNode**) をデプロイメント環境定義 (**myDepEnv**) に追加します。

重要: 単一クラスターのトポロジー・パターンにノードを追加する場合は、`-topologyRole` の値を **ADT** に設定する必要があります。デプロイメント環境のトポロジー・パターンは、`createDeploymentEnvDef` コマンドまたは「デプロイメント環境構成」ウィザードを使用してデプロイメント環境を作成するときに指定します。

```
wsadmin.sh -connType SOAP -host myDmgr -port 8879 -user dmgrAdmin -password dmgrPass
> $AdminTask addNodeToDeploymentEnvDef {-topologyName myDepEnv -nodeRuntime WPS
-topologyRole Messaging -nodeName MyNode -serverCount 3}
```

注: 管理セキュリティを無効にする場合は、ユーザー ID とパスワードを指定する必要はありません。

関連資料



addNodeToDeploymentEnvDef コマンド

addNodeToDeploymentEnvDef コマンドを使用して、既存のデプロイメント環境定義にノードを追加します。

コマンド行を使用したデプロイメント環境の生成:

wsadmin.sh インターフェースを使用してデプロイメント環境を生成することができます。この機能により、スクリプトを使用して、デプロイメント・マネージャー上の複数のデプロイメント環境を無人で構成できます。

始める前に

デプロイメント環境を構成するデプロイメント・マネージャーでコマンドを入力する必要があります。

この作業に必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとして管理コンソールにログインする必要があります。

このタスクについて

デプロイメント・マネージャー上にデプロイメント環境をインポートまたは作成したあと、generateDeploymentEnv コマンドを使用してデプロイメント環境を構成できます。

手順

1. wsadmin.sh 環境に入ります。
2. 構成するトポロジーごとに generateDeploymentEnv コマンドを入力します。
3. クラスター・ショート・ネームを定義するには、以下の手順を実行します。
 - a. 管理コンソールにログインして、「**サーバー**」>「**クラスター**」>「**WebSphere Application Server クラスター**」にナビゲートします。
 - b. 変更するクラスターの名前をクリックします。
 - c. 新しいクラスター・ショート・ネームを「ショート・ネーム」領域に入力します。例えば、ショート・ネームを BBOC001 (通常はこれがデフォルト値) から WCLxx に変更します。ここで xx は、セルの接頭部です (例: WCLT5)。
4. サーバー・ショート・ネームを定義するには、以下の手順を実行します。
 - a. 管理コンソールにログインして、「**サーバー**」>「**サーバー・タイプ**」>「**WebSphere Application Server**」にナビゲートします。
 - b. 変更するサーバーの名前をクリックします。
 - c. 新しいサーバー・ショート・ネームを「ショート・ネーム」領域に入力します。例えば、ショート・ネームを BBOS001 から WZxxZ1 に変更します。ここで xx は、セルの接頭部です (例: WZT5Z1)。

サーバー・ショート・ネームの定義について詳しくは、『z/OS アプリケーション・サーバーの命名規則』を参照してください。

例

以下のコマンドは、ホスト myDmgr 上で eastEnvironment トポロジーおよび westEnvironment トポロジーを構成します。

```
wsadmin.sh -connType SOAP -host myDmgr -port 8879
> $AdminTask generateDeploymentEnv {-topologyName eastTopology}
> $AdminTask generateDeploymentEnv {-topologyName westTopology}
> $AdminConfig save
```

注: 管理セキュリティが有効である場合、 wsadmin.sh コマンドの処理後にユーザー ID とパスワードの入力を求めるプロンプトが表示されます。

次のタスク

構成されたデプロイメント環境を保存します。コマンド行から、\$AdminConfig save と入力します。

関連情報

generateDeploymentEnvFromDef コマンド

コマンド行を使用したデプロイメント環境定義のインポート

 ノード・エージェントの管理

コマンド行からのデプロイメント環境定義の検証:

wsadmin.sh コマンドを使用して、デプロイメント環境定義を検証することができます。

始める前に

このタスクでは、ノードがデプロイメント・マネージャーに統合されていることを前提としています。

デプロイメント環境定義の検証の対象となるデプロイメント・マネージャーを起動しておく必要があります。

この作業に必要なセキュリティ・ロール: セキュリティとロール・ベースの許可が有効になっている場合、管理者権限またはオペレーター権限を持つユーザー ID とパスワードを使用してこのタスクを実行する必要があります。

このタスクについて

このタスクは、wsadmin.sh コマンドを使用して、デプロイメント環境定義を検証します。

手順

1. コマンド・ウィンドウを開きます。

wsadmin.sh コマンドは、<WPS>/profiles/<dmgr profile>/bin ディレクトリーか <WPS>/bin ディレクトリーのいずれかに格納されています。

2. コマンド・プロンプトから wsadmin.sh コマンドを入力して wsadmin.sh 環境に入ります。

3. `validateDeploymentEnvDef` コマンドを入力して、デプロイメント環境定義を検証します。

注: 管理セキュリティが有効になっている場合は、ユーザー ID とパスワードの入力画面が表示されます (コマンド内で指定していない場合)。

例

この例では、管理セキュリティが有効な状態で、デプロイメント環境定義 (**myDepEnv**) を検証します。

```
wsadmin.sh -connType SOAP -host myDmgr -port 8879 -user dmgrAdmin -password -dmgrPass  
> $AdminTask validateDeploymentEnvDef { -topologyName topOne}
```

注: 管理セキュリティを無効にする場合は、ユーザー ID とパスワードを指定する必要はありません。

デプロイメント環境設定の編集

デプロイメント環境設定を編集および変更できます。

ホスト別名の構成:

IBM HTTP Server または選択したサーバーを構成して、管理対象ノードとデプロイメント・マネージャー間の通信を可能にします。

始める前に

デプロイメント・マネージャーおよび関連ノードを作成および構成。

このタスクについて

管理対象ノードとデプロイメント・マネージャー間の相互通信が可能になる必要があります。そのために、デプロイメント・マネージャーはデプロイメント・ターゲット・クラスター内の各ノードのホスト別名を認識する必要があります。ホスト別名には、DNS ホスト名およびポート番号が含まれます。デプロイメント・ターゲットで実行中のアプリケーションには、この別名を URL の一部として使用してアクセスします。

注: この手順は、`AppCluster_member1` および `AppCluster_member2` という 2 つのアプリケーション・クラスター・メンバーを使用します。説明内ではご使用のサーバー名と置換して理解してください。

手順

1. 管理コンソールから、「サーバー」 → 「サーバー・タイプ」 → 「**WebSphere Application Server**」 → 「**AppCluster_member1**」にナビゲートします。
2. 名前をクリックします。
3. 「通信」の見出しの下の「ポート」を展開し、`WC_defaulthost` にリストされているポート値をメモします。この値は後に使用する必要があります。
4. クラスター・メンバーごとにステップ 1 から 3 を繰り返します。追加のアプリケーション・クラスター・メンバーごとにこれを繰り返します。

完了すると、クラスター・メンバーとそれらのデフォルト・ホスト用のポート番号のリストが表示されます。

5. 管理コンソールから、「環境」 → 「仮想ホスト」 → 「default_host」にナビゲートします。
6. 「追加プロパティ」で、「ホスト別名」をクリックします。
7. クラスター・メンバーに対するホスト名とポート値の正しい組み合わせのエントリーが表示されない場合、欠落しているエントリーをリストに追加します。
8. リストに新規エントリーを追加した場合、「保存」および「同期化」をクリックします。

次のタスク

テスト・アプリケーションをインストールして、インストール済み環境を検査します。

デプロイメント環境のデータ・ソースの構成:

「データベース・プロバイダー構成」ページを使用して、ビジネス・インテグレーション・データ・ソースの初回の構成を行います。

始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。
- デプロイメント・マネージャーの管理コンソールから、「サーバー」 → 「デプロイメント環境」 → 「*deployment_environment_name*」 → 「関連項目」 → 「データ・ソース」にナビゲートします。

この作業に必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとして管理コンソールにログインする必要があります。

このタスクについて

「データ・ソース」ページを使用して、デプロイメント環境に必要なすべてのデータ・ソースのコレクションを構成します。

データ・ソースを必要とするコンポーネントは、選択された「データベース・プロバイダー」に基づいて、すべての必須フィールドを決定します。これらの必須フィールドにはデータを入力する必要があります。コンポーネントは、残りのフィールドにデフォルト値を設定します。ユーザーは、デフォルト値を保持することもできますし、必要に応じてそれを変更することもできます。通常、コンポーネントは、「有効範囲」の値を決定します。

ビジネス・インテグレーションのデータ・ソースは、一度だけ構成できます。データ・ソースを構成して保存すると、一部のテキスト・ボックスが使用不可になり、値を変更できなくなります。ページ内のそれ以外のテキスト・ボックスは、すべて編集できます。

手順

1. 「データ・ソース」ページで、構成するデータ・ソースの横にあるチェック・ボックスを選択します。
2. このページに表示されない追加のデータ・ソース・フィールドを編集する必要がある場合は「プロバイダーの編集」をクリックします。

注: 「データ・ソース」列のデータ・ソース名をクリックするだけの方法も可能です。

3. 情報を入力します。サポートされているデータベース・タイプのリストについては、『データベース仕様書』を参照してください。
4. 「適用」または「OK」をクリックして、変更を保存します。

関連情報

データベースの構成

共通データベースの仕様

デプロイメント環境の認証別名の構成:

1 つの管理コンソール・ページから、すべての認証別名を検討または編集できます。

始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。

デプロイメント・マネージャーの管理コンソールから、「サーバー」 → 「デプロイメント環境」 → 「*deployment_environment_name*」 → 「関連項目」 → 「認証別名」にナビゲートします。

この作業に必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとして管理コンソールにログインする必要があります。

このタスクについて

この統合された認証別名リストから、以下を行うことができます。

- 指定のデプロイメント環境のすべての別名の検討
- *Alias_name* リンクを使用した認証構成ページへのアクセス

「リセット」ボタンは、選択された行を現在構成されている値にリセットします。*Alias_name* をクリックして、変更を行う認証構成ページにアクセスしてください。

手順

1. 変更する行を選択します。
2. 以下のいずれかを実行します。

オプション	説明
行の編集	「 <i>Alias_name</i> 」をクリックします。
行のリセット	「リセット」をクリックします。

行の編集では、「認証構成」ページが表示され、ここで変更を行うことができます。

3. 変更を保存するには、「OK」または「適用」をクリックします。

関連情報

認証

デプロイメント環境の据え置かれた構成の設定:

データベースとテーブルの作成を据え置く必要がある場合は、「据え置かれた構成」ページを使用してください。このページには、データベースとテーブルを作成するためのスクリプトを見つけて実行する方法の説明が示されています。

始める前に

- デプロイメント環境が、このデプロイメント・マネージャー上に存在することを確認してください。

デプロイメント・マネージャーの管理コンソールから、「サーバー」 → 「デプロイメント環境」 → 「*deployment_environment_name*」 → 「プロパティ」 → 「据え置かれた構成」にナビゲートします。

この作業に必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとして管理コンソールにログインする必要があります。

このタスクについて

デプロイメント環境の構成とは別に、データベース表またはスキーマを作成する必要がある場合に、この手順を使用します。

「据え置かれた構成」ページには、トポロジーのデータベースを正しく構成するために必要な構成ステップが示されています。通常、このページには、以下の内容が示されます。

- スクリプトの場所
- スクリプトの実行方法の説明

手順

1. 「据え置かれた構成」ページの指示を実行します。
2. 完了したら、「実行された構成」をクリックします。

次のタスク

据え置かれた構成をいつ誰が最後に実行したかを示すテキスト・ボックスが表示されます。表示された説明は、将来参照できるようにこのページに残ります。

関連タスク

99 ページの『パターンを使用したデプロイメント環境の作成』
デプロイメント・パターンを選択したら、「デプロイメント環境構成」ウィザード
を使用して、そのパターンに基づいたデプロイメント環境を作成します。

デプロイメント環境の検証

実動アプリケーションを新しい環境に移動する前に、テストを行ってすべてのコン
ポーネントが正常に動作することを確認してください。

始める前に

『デプロイメント環境の実装』に記載されている説明に従って、デプロイメント環
境の実装を完了します。

1. ソフトウェアをインストールする
2. デプロイメント・マネージャーをホストするノードを構成する
3. ノードを構成する
4. ノードをデプロイメント・マネージャーに統合する
5. デプロイメント環境に機能を提供するノードをクラスター化する

このタスクについて

デプロイメント環境の検証方法は、実装した環境が IBM 提供のデプロイメント環
境であるかカスタム・デプロイメント環境であるかによって異なります。IBM 提供
のデプロイメント環境は、管理コンソールで単一のパネルから管理できます。カス
タム・デプロイメント環境は、管理コンソールで手動で作成し管理する必要があります。

手順

1. 検証するデプロイメント環境のタイプを識別します。

この情報は、元の計画に基づいて用意してください。

2. デプロイメント環境を開始します。

デプロイメント環境のタイプ	始動方法
IBM 提供のパターン	「システム管理」>「デプロイメント環境」> 「デプロイメント環境構成」（『デプロイメ ント環境の開始と停止』の説明を参照してく ださい）。
カスタム	「サーバー」>「クラスター」（『カスタ ム・デプロイメント環境の始動の検証 (Verifying a custom deployment environment starts)』を参照してください）。 注: デプロイメント環境に定義されているす べてのサーバーとクラスターを始動する必要 があります。

3. テスト・アプリケーションをインストールします。
4. ルーティング用のテスト・アプリケーションを構成します。

5. テスト・アプリケーションを開始します。
6. テスト・アプリケーションを実行し、その結果を検証します。

次のタスク

実動アプリケーションをインストールします。

アプリケーション・デプロイメント・ターゲット・クラスターの始動の確認:

アプリケーション・デプロイメント・ターゲット・クラスターが始動可能なことを確認するには、デプロイメント環境のクラスターをすべて始動する必要があります。このセクションでは、3 つのクラスターで構成されるデプロイメント環境についての事例を説明します。

始める前に

メッセージング・エンジン、Common Event Infrastructure (CEI) イベント・サーバー・アプリケーション、およびアプリケーション・デプロイメント・ターゲットのクラスターを作成して構成する必要があります。

このタスクについて

アプリケーション・デプロイメント・クラスターが始動可能なことを確認するには、各クラスターを順に始動します。

注意:

- ここでは、トポロジーで 3 つのクラスターを構成しており、それぞれ **MECluster**、**SupportCluster**、**AppCluster** という名前になっているとして説明します。これらは実際のクラスター名に置き換えてください。また、デプロイメント環境に追加のクラスターがある場合は、該当するステップを繰り返してください。

手順

1. デプロイメント・マネージャーの管理コンソールで「サーバー」を展開し、「クラスター」を選択します。
2. クラスターを開始します。
 - a. 「**MECluster**」の横にあるチェック・ボックスを選択します。
 - b. 「開始」を選択し、MECluster が始動して緑色の矢印が表示されるまで待ちます。
 - c. 「**SupportCluster**」の横にあるチェック・ボックスを選択します。
 - d. 「開始」を選択し、SupportCluster が始動して緑色の矢印がもう 1 つ表示されるまで待ちます。
 - e. 「**AppCluster**」の横にあるチェック・ボックスを選択します。
 - f. 「開始」を選択し、AppCluster が始動して緑色の矢印がもう 1 つ表示されるまで待ちます。
3. メッセージング・バスをクリックします。
 - a. すべてのクラスターが開始するまで待ちます。
 - b. 「サービス統合」 → 「バス」の順にクリックします。

- c. バスごとに、メッセージング・エンジンが稼働していることを確認します。
 - 1) バス名を選択します。
 - 2) 「ローカル・トポロジ」をクリックしてバス・トポロジを表示します。
 - 3) メッセージング・エンジンの状況が表示されるまでバスを拡張します。
4. コントローラー、サーバント、および付属に関するクラスター・メンバーのジョブ・ログを調べます。エラーがないことを確認し、「サーバー AppCluster_member1 が e-business 用に使われています (Server AppCluster_member1 is open for e-business)」または「サーバー AppCluster_member2 が e-business 用に使われています (Server AppCluster_member2 is open for e-business)」という行を探します。これらの行があれば、クラスターは正常に始動しています。エラーがある場合は、訂正してから続行してください。

次のタスク

エラーを訂正した後には、ホストの別名を構成します。

注: 構成エラーを訂正した後は、必ずクラスターを停止してから再始動して、構成の変更を有効にする必要があります。

トラブルシューティング・ヒント: ログを調べていると、メッセージング・エンジンが特定のバスを検出できなかったために起動に失敗したことを示すメッセージが見つかることがあります。このメッセージは、クラスターを再始動すると除去されます。

テスト・アプリケーションのインストール:

テスト・アプリケーションをインストールして、デプロイメント環境の検証のプロセスを開始します。

始める前に

- 完成したデプロイメント環境を作成し、インストールする必要があります。
- デプロイメント・マネージャー管理コンソールにログインします。

このタスクについて

WebSphere Process Server で提供される BPCIVTApp (Business Process Choreographer インストール検査テスト) というアプリケーションを使用して、WebSphere Process Server 環境を正しくインストールおよび構成したことを確認します。まず、アプリケーションをインストールする必要があります。

このアプリケーションのインストールについて詳しくは、『Business Process Choreographer の動作の検証 (Verifying that Business Process Choreographer works)』を参照してください。管理コンソールからのアプリケーションのインストールについて詳しくは、『コンソールへのアプリケーション・ファイルのインストール』を参照してください。

注: ビジネス・プロセスおよびヒューマン・タスクを使用可能にしていない場合は、BPCIVTApp を使用してデプロイメント環境をテストすることはできません。こ

の場合、デプロイメント環境を使用するには、ビジネス・ルールおよびセレクターを使用する Service Component Architecture アプリケーションをインストールして実行する必要があります。デプロイメント環境をテストするプロセスを、ご使用のアプリケーションに適合するように変更してください。

手順

1. 管理コンソールから、「アプリケーション」 → 「新規アプリケーション (New Application)」 → 「新規エンタープライズ・アプリケーション (New Enterprise Application)」を選択します。
 2. 「ローカル・ファイル・システム」が選択されていることを確認し、`bpcivt.ear` ファイルを参照します。このファイルは、`install_root/installableApps` ディレクトリーにあります。
 3. `bpcivt.ear` ファイルを選択してから、「開く」を選択します。
 4. 以下のステップでは、デフォルト構成を使用することを前提とします。「要約」ページに達するまで、後続のパネルで「次へ」を選択します。この手順を実行中に、他のトピックで説明するように、さまざまなオプションを選択し、モジュールをサーバーにマップすることになります。テストのためには、このモジュールをアプリケーション・デプロイメント・ターゲット・クラスターにマップしてください。
- 注: スタンドアロン・サーバーで、モジュールをアプリケーション・ターゲット・クラスターにマップする必要はありません。
5. 「終了」を選択します。
 6. 「保存」を選択し、次に「同期化」を選択します。

次のタスク

ルーティング用テスト・アプリケーションの構成:

この手順を使用して、ルーティング用のテスト・アプリケーションを構成します。

始める前に

テスト・アプリケーションをインストールする必要があります。

このタスクについて

先にアプリケーションを構成してから、プラグイン構成ファイルを生成します。

注: 説明では、`AppCluster` という名前のクラスターと `Webserver1` という名前の Web サーバーを使用することを前提とします。ご使用のテスト・アプリケーションが、ヒューマン・タスクまたはビジネス・プロセスを使用する場合、アプリケーション・クラスター上で既に `Business Process Choreographer` を構成済みであることを確認してください。

モジュールの管理、モジュール設定、およびモジュールのマッピングについては、`WebSphere Application Server` インフォメーション・センターを参照してください。

手順

1. 以下のようにして、Web サーバーとアプリケーションへのデプロイメント・ターゲットを識別するために実行するアプリケーション (1 つまたは複数) を構成します。
 - a. 管理コンソールから「アプリケーション」 → 「アプリケーション・タイプ」 → 「WebSphere エンタープライズ・アプリケーション」を選択します。
 - b. アプリケーションの名前を選択します。
 - c. 「モジュールの管理 (Manage modules)」を選択します。

このパネルで、各モジュールは、「サーバー」の下に示される 1 つ以上のターゲットにマップされていなければなりません。
 - d. 「クラスターとサーバー (Clusters and Servers)」にリストされた選択肢から、*Webserver1* (前に構成した Web サーバー) および *AppCluster* (アプリケーション・デプロイメント・ターゲット) を選択します。
 - e. 「適用」を選択してから、「OK」を選択します。
 - f. ご使用のデプロイメント環境のすべての Web サーバーとデプロイメント・ターゲットの構成を完了するまで、ステップ 1d から 1e までを繰り返します。
 - g. 「保存」を選択し、次に「同期化」を選択します。
2. プラグイン構成ファイルを生成してください。
 - a. 管理コンソールから「サーバー」 → 「サーバー・タイプ」 → 「Web サーバー (Web servers)」を選択します。
 - b. *Webserver1* という名前の横にあるチェック・ボックスを選択します。
 - c. 「プラグインの生成 (Generate plug-in)」を選択します。ウィンドウの上部のメッセージで示すように、プラグイン構成ファイルが作成されます。
 - d. デプロイメント環境で必要な回数だけ、ステップ 2b および 2c を繰り返します。

次のタスク

デプロイメント・マネージャーおよびノード・エージェントを停止して再始動します。次にテスト・アプリケーションを開始します。

テスト・アプリケーションの開始:

この手順を使用して、テスト・アプリケーションを開始し、実装環境をテストします。

始める前に

ルーティング用のテスト・アプリケーションをインストールおよび構成しておく必要があります。

このタスクについて

管理コンソールからテスト・アプリケーションを開始します。

手順

1. 管理コンソールから「アプリケーション」 → 「アプリケーション・タイプ」 → 「WebSphere エンタープライズ・アプリケーション」を選択します。
2. アプリケーション名の横にあるチェック・ボックスを選択し、「開始」を選択します。アプリケーションが正常に開始したことを示す緑色の矢印が表示されるまで待機します。

次のタスク

テスト・アプリケーションを開始した後に、このアプリケーションを実行します。

注: アプリケーションが正常に開始しない場合は、ログ・ファイルを参照して、問題を示すエラー・メッセージを見つけてください。

テスト・アプリケーションの実行:

この手順を使用して、テスト・アプリケーションを実行し、ご使用のデプロイメント環境が正しく動作しているかどうかを判別します。

始める前に

テスト・アプリケーションを開始する必要があります。

このタスクについて

このアプリケーションが正常に実行された場合、ご使用のデプロイメント環境は正しく動作しています。アプリケーション・デプロイメント・ターゲット・クラスターのもう一方のメンバーで同じ手順を実行して、そのメンバーも正しく機能することを確認します。

手順

1. ブラウザー・ウィンドウで `http://hostname:portnumber/testapp` という形式の URL を入力します。ここで、*hostname* は、アプリケーションがインストールされたクラスター・メンバーをホストするシステムの完全修飾 DNS 名または IP アドレス、*portnumber* は、そのクラスター・メンバーのデフォルト・ホストに関連付けられたポート番号、*testapp* は、ご使用のテスト・アプリケーションの名前です。
2. 画面上のロギング・メッセージを調べます。

ご使用のテスト・アプリケーションにヒューマン・タスクが含まれる場合、「HumanTaskManager API EJB を検索中... (Looking up the HumanTaskManager API EJB...)」で始まるロギング・メッセージが画面に表示されます。アプリケーションは処理を続行して、タスクを作成し、それを要求し、入力および出力データをチェックし、タスクを完了し、それを削除します。「合格」という単語がログ・メッセージの終わりの方に表示され、アプリケーションが正常に実行されたことを示します。

成功したことを示す、アプリケーションに組み込んだすべてのメッセージが表示されることを確認してください。

次のタスク

他のテスト・アプリケーションをインストールして開始します。

その他のアプリケーションのインストールおよびアクセス:

管理コンソールまたは Business Process Choreographer Explorer からアプリケーションのインストールとアプリケーションへのアクセスを行って、さらにデプロイメント環境をテストします。

始める前に

デプロイメント環境のインストールおよび構成を正常に完了している必要があります。

このタスクについて

テスト・アプリケーションをインストールした方法と同様に、他のアプリケーションをインストールし、開始することができます。これらのアプリケーションにアクセスするには、管理コンソールまたは Business Process Choreographer Explorer を使用します。

手順

1. アプリケーションを見つけます。

管理コンソールで「**アプリケーション**」 → 「**新規アプリケーション (New Application)**」をクリックし、インストールするアプリケーションを見つけます。

2. アプリケーションをインストールします。
3. アプリケーションを開始します。
4. アプリケーションにアクセスします。

ブラウザ・ウィンドウでアプリケーションの URL を入力します。例えば、`http://hostname:portnumber/myapp` のように入力します。ここで、*hostname* は、アプリケーションがインストールされたクラスター・メンバーに対応するシステムの完全修飾 DNS 名 (または IP アドレス)、*portnumber* は、そのクラスター・メンバーの `default_host` に関連付けられたポート番号、*myapp* は、アクセスしたいアプリケーションの名前です。

Business Process Choreographer Explorer から、以下の操作を実行します。

- a. ブラウザー・ウィンドウで `http://hostname:portnumber/bpc` という形式の URL を入力します。ここで、*hostname* は、アプリケーションがインストールされたクラスター・メンバーに対応するシステムの完全修飾 DNS 名 (または IP アドレス)、*portnumber* は、そのクラスター・メンバーの `default_host` に関連付けられたポート番号です。

「**マイ・タスク**」というラベルのページが表示されますが、タスクはリストされていません。

- b. 「マイ・プロセス・テンプレート (My Process Templates)」を選択します。インストールした任意のアプリケーションに対応するテンプレートがリストされます。
 - c. そのページのインターフェース・コントロールを使用して、タスクを開始したり、タスクを操作したり、タスクを完了したりします。Business Process Choreographer タスクの実行について詳しくは、『ビジネス・プロセスおよびヒューマン・タスクの管理』を参照してください。
5. 必要な場合は、クラスター・メンバーのサーバント・ログを調べて、アプリケーションのレコードを表示し、エラーがあるかどうかをチェックすることができます。

Business Process Management コンポーネントの手動構成

Business Process Management コンポーネントを手動で構成するには、管理コンソールを使用して、以下のサブトピックで詳細説明するタスクを実行する必要があります。

クラスターの作成

以下の手順では、空の管理対象ノードに 1 つのクラスター・メンバーを持つクラスターを作成する方法について説明します。管理コンソールを使用してクラスターを作成することの利点は、作業中に変更を元に戻せることと、グラフィカル・ユーザー・インターフェースを使用できることです。クラスターを作成しておく、後でクラスターを発展させる (クラスターにサーバーを追加してスケラビリティを向上させるなど) ことが容易になるため、作成することを推奨します。

始める前に

クラスターを作成する前に、管理対象ノードのデプロイメント・マネージャーおよびノード・エージェントを開始してください。詳しくは、『MVS コンソールからのサーバーの開始』を参照してください。

手順

1. 管理コンソールにログインし、「サーバー」 → 「クラスター」 → 「Websphere Application Server クラスター」にナビゲートします。
2. 「新規」をクリックします。
3. クラスターの名前とショート・ネームを入力します。
4. 「次へ」をクリックします。
5. サーバーのロング・ネームとショート・ネームを入力します。
6. 「ノードの選択」リストから、サーバーを定義するノードをクリックします。
7. アプリケーション・サーバー・テンプレートを使用してメンバーを作成するためにボタンを選択します。
8. クラスター・メンバーの作成に使用するアプリケーション・サーバー・テンプレートとして、ドロップダウンから「defaultProcessServerZOS」を選択します。
9. 「次へ」をクリックします。空白のフォームが表示され、これを使用して追加のクラスター・メンバーを定義できます。作成したサーバーが画面の下部にリスト表示されます。

10. 「次へ」をクリックします。
11. 要約画面の詳細を確認し、「次へ」をクリックします。
12. 構成変更を保存します。作成したクラスターがリストに表示されます。
13. ご使用の構成に合わせてポート番号をカスタマイズします。方法、および Jython スクリプトについては、技術白書 TD104066 を参照してください。技術情報については、『WAS V6.1 for z/OS での新規アプリケーション・サーバーの作成 (Creating new Application Servers in WAS V6.1 for z/OS)』を参照してください。

タスクの結果

選択したサーバーを使用して、選択した管理対象ノード内に最初のクラスター・メンバーとしてクラスターが作成されます。

次のタスク

次に、メッセージング・エンジン・データ・ソースを作成します。

コンポーネントの構成

以下の Business Process Management コンポーネントを構成できます。

- サーバーまたはクラスターの SCA サポートの構成
- 管理コンソールでのすべての REST サービスの構成
- Business Process Choreographer の構成
- Business Space の構成
- WebSphere Portal での Business Space の構成
- ビジネス・ルールおよびセレクターの構成
- 234 ページの『リレーションシップ・サービスの構成』
- 拡張メッセージング・リソースの構成
- メッセージング・サーバー環境のセットアップ
- JNDILookup Web Service の構成
- 246 ページの『Common Event Infrastructure の構成』
- WebSphere Business Integration Adapter の構成
- Service Federation Management 用の WebSphere Process Server の構成

WebSphere Process Server 機能での WebSphere Enterprise Service Bus の再構成

前に、WebSphere Enterprise Service Bus のフィーチャーのみを使用するように WebSphere Process Server ノードを構成したが、それ以降、ビジネス要件が変化した場合は、WebSphere Process Server の追加コンポーネント (Business Process Choreographer フィーチャーなど) にアクセスするようにサーバーを再構成できます。

このタスクについて

WebSphere Process Server メディアを z/OS システムにアンロードして zWPSInstall.sh スクリプトを実行した時点で、WebSphere Process Server に必要な製品定義がセットアップされています。その後 zWESBConfig.sh スクリプトを実行してノードを WebSphere ESB ノードとして構成していても、問題ありません。したがって、zWPSConfig.sh スクリプトを実行することにより、WebSphere Process Server の追加フィーチャーを使用するようにノードを再構成できます。

手順

WebSphere ESB ノード上で zWPSConfig.sh スクリプトを実行します。詳しくは、zWPSConfig.sh および zWESBConfig.sh スクリプト を参照してください。zWPSConfig.sh スクリプトは、サーバーが WebSphere ESB 用に構成されていることを確認してから、WebSphere Process Server に必要な追加コンポーネント (Business Process Choreographer など) を使用するようにサーバーを構成します。

タスクの結果

これで、ノードがインストールされ、WebSphere Process Server を使用するように構成されました。

サーバーまたはクラスタの SCA サポートの構成

「Service Component Architecture (SCA)」コンソール・ページで、Network Deployment 環境内のサーバーまたはクラスタが、サーバー・アプリケーションと、そのアプリケーションに必要なメッセージング・エンジンおよび宛先をホストできるように設定します。

始める前に

SCA サポートを構成する前に、以下の点を確認します。

- メッセージング・エンジンと宛先をホストする場所 (ローカル・バス・メンバーまたはリモート・バス・メンバーを使用)。
- SCA システム・バスのみを構成する必要があるのか、それとも SCA アプリケーション・バスも構成する必要があるのか。アプリケーション・バスはデフォルトで構成されます。WebSphere Business Integration Adapter を使用する SCA アプリケーションをデプロイする場合は、アプリケーション・バスが必要です。

このタスクに必要なセキュリティ・ロール: 以下のタスクを実行するには、管理者またはコンフィギュレーターとしてログインしている必要があります。

このタスクについて

サービス・アプリケーションでは、自動的に作成されるサービス統合バスを 1 つ以上使用する必要があります。これらのバスでは、宛先のメッセージング・エンジンを構成済みでなければなりません。Network Deployment 構成の新規のサーバーとクラスタは、デフォルトでは SCA アプリケーションとそれらの宛先をホストするには構成されません。

サーバーまたはクラスター上で SCA サポートを構成するには、以下のステップを実行します。

手順

1. 管理コンソール内で、スコープに応じて以下のいずれかをクリックします。
 - 「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → *server_name* → 「Service Component Architecture」
 - 「サーバー」 → 「クラスター」 → 「WebSphere Application Server クラスター (WebSphere Application Server clusters)」 → *cluster_name* → 「Service Component Architecture」
2. 「Service Component Architecture コンポーネントのサポート」をクリックします。
3. 「バス・メンバー・ロケーション」パネルで、SCA アプリケーションに必要な宛先とメッセージング・エンジンをホストする場所を指定します。2 つのオプションがあります。
 - **ローカル**。SCA アプリケーション、宛先、およびメッセージング・エンジンを現行サーバーまたはクラスターでホストすることを指定します。
 - **リモート**。SCA アプリケーションを現行サーバーまたはクラスターでホストし、宛先およびメッセージング・エンジンをリモート・サーバーまたはクラスター (デプロイメント・ターゲット ともいう) でホストすることを指定します。
4. (リモート・バス・メンバーのみ) 前のステップで「リモート」を選択した場合は、ホスト・アプリケーションの宛先およびメッセージング・エンジンをホストするリモート・サーバーまたはクラスターを指定します。ドロップダウン・メニューから、既存のデプロイメント・ターゲット (既に SCA システム・バスのメンバーとして構成されているターゲット) を選択するか、または「新規」をクリックして、「デプロイメント・ターゲットのブラウズ」ページから新規サーバーまたはクラスターを選択します。

「デプロイメント・ターゲットのブラウズ」ページから新規サーバーまたはクラスターを選択する場合は、このトピックで説明する SCA の構成が完了すると、そのターゲットに必要なメッセージング・エンジンが自動的に構成されます。
5. 「システム・バス・メンバー」パネルの表で、システム・バス・データ・ソース構成を確認または変更します。
 - a. 「データベース名」、「スキーマ」、「テーブルの作成」、「ユーザー名」、「パスワード」、「サーバー」、および「プロバイダー」の各フィールドのデフォルト値を確認します。各フィールドと受け入れ可能な値の詳細については、オンライン・ヘルプを参照してください。
 - b. これらのフィールドにデフォルト値がない場合、またはデフォルト値が適切でない場合は、システム・バス・データ・ソースに対する適切な値を入力します。フィールドに値を直接入力するか、または「編集」をクリックしてデータ・ソースの詳細ページで値を編集します。
 - c. オプション: 「接続のテスト」をクリックして、データ・ソースがデータベースに接続でき、正しく認証されることを確認します。
6. 「アプリケーション・バス・メンバー」パネルの表で、アプリケーション・バス・データ・ソース構成を確認または変更します。

- a. 「**WebSphere Business Integration Adapter コンポーネントの使用可能化**」オプションが選択されていることを確認します。

注: アプリケーション・バスを使用しない場合は、「**WebSphere Business Integration Adapter コンポーネントの使用可能化**」オプションをクリアし、ステップ 7 に進みます。
 - b. 「データベース名」、「スキーマ」、「テーブルの作成」、「ユーザー名」、「パスワード」、「サーバー」、および「プロバイダー」の各フィールドのデフォルト値を確認します。各フィールドと受け入れ可能な値の詳細については、オンライン・ヘルプを参照してください。
 - c. これらのフィールドにデフォルト値がない場合、またはデフォルト値が適切でない場合は、アプリケーション・バス・データ・ソースに対する適切な値を入力します。フィールドに値を直接入力するか、または「編集」をクリックしてデータ・ソースの詳細ページで値を編集します。
7. 「OK」をクリックすると、SCA 構成が完了します。
 8. 変更を保存します。必要に応じて、加えた変更を見直すこともできます。

次のタスク

次に、必要に応じて構成を続行できます。

- Business Process Choreographer の構成
- Common Event Infrastructure の構成

関連情報

サーバーおよびクラスターでの Service Component Architecture サポートに関する考慮事項

サーバーおよびクラスターでは、Service Component Architecture (SCA) アプリケーション、アプリケーション宛先、またはその両方をサポートできます。

サーバーおよびクラスターでの Service Component Architecture サポートに関する考慮事項

サーバーおよびクラスターでは、Service Component Architecture (SCA) アプリケーション、アプリケーション宛先、またはその両方をサポートできます。

SCA アプリケーション (サービス・アプリケーションとも呼ばれる) では、自動的に作成される 1 つ以上のサービス統合バスを使用する必要があります。各アプリケーションでは、一連のメッセージング・リソース (宛先 と呼ばれる) が使用されます。これらの宛先には構成されたメッセージング・エンジンが必要であり、またこれらの宛先はアプリケーションと同じサーバーまたはクラスター、あるいはリモート・サーバーまたはリモート・クラスター上でホストすることができます。メッセージング・エンジンはデータベースのデータ・ストアを使用します。

Network Deployment 構成の新規のサーバーとクラスターは、デフォルトでは SCA アプリケーションとそれらの宛先をホストするようには構成されません。

注: スタンドアロン・サーバーでは、SCA サポートが自動的に構成されます。この構成を使用不可にすることはできません。

このサポートを有効にするには、管理コンソールの「Service Component

Architecture」ページを使用します。サーバーの場合、アプリケーション・クラス・ローダー・ポリシーが「複数」に設定されていることを確認します。

Network Deployment 環境または管理対象ノード環境内にあるサーバーまたはクラスターに対して、SCA サポートを使用可能にする前に、以下の可能な構成のいずれを実装するかを決定します。

- **リモート・バス・メンバーの構成:** サーバーまたはクラスターでは SCA アプリケーションがホストされますが、宛先はリモート・サーバーまたはリモート・クラスター上でホストされます。このシナリオでは、宛先をホストするために必要なメッセージング・エンジンを使って、リモート・サービス統合バス・メンバーを構成する必要があります。

リモート・メッセージングの使用には、サービス統合バスとそのメンバーの計画を立てて構成するための初期投資が必要になりますが、この構成はアプリケーション・クラスター内の複数のメンバーで再利用できます。メッセージは、すべてのメンバーに配布されます。また、フェイルオーバー・サポートを提供するように初期構成を構造化することもできます。

- **ローカル・バス・メンバーの構成:** サーバーまたはクラスターでは SCA アプリケーションおよびアプリケーション宛先の両方がホストされます。必要なメッセージング・エンジンは、サーバーまたはクラスター上のローカル・バス・メンバーを使用して構成されます。

計画の各トピックを参照して、ご使用の環境にいずれの構成が適しているかを判断してください。

関連情報

-  [サーバーのクラス・ローダーの構成](#)
-  [サービス統合バスについて](#)
-  [メッセージング・エンジン](#)

管理コンソールでのすべての REST サービスの構成

すべての Representational State Transfer (REST) サービスを環境に合わせて構成するには、REST サービスの管理コンソール・ページを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

このタスクについて

REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他の構成タイプの場合は、管理コンソール・ページを使用すると、Business Space内のすべての製品ウィジェットに関する REST サービスを構成することができます。「REST サービス」ページでは、現在の環境に対するすべてのサービスを表示したり、各サービスを個々に有効化/無効化したりすることができます。

手順

1. 「サービス」 → 「REST サービス」 → 「REST サービス」をクリックします。

「REST サービス」ページが開き、現在の環境内での REST サービスがすべて表示されます。

2. 「スコープセクション」で、すべてを指定して、現在の環境内の REST サービスをすべて表示するか、REST サービスが有効になっているサーバーまたはクラスターを選択します。
3. プロバイダーに対する REST サービスがリストされたテーブル内の各行で、個々の REST サービスを有効化する場合は「使用可能」チェック・ボックスを選択し、個々の REST サービスを無効化する場合は「使用可能」チェック・ボックスをクリアします。
4. 有効化する個々のサービスについて、「説明」列にわかりやすい説明を入力します。
5. 「OK」をクリックして、サービスへの変更をコミットします。

サービス・プロバイダーでの REST サービスの構成

サービス・プロバイダー内の Representational State Transfer (REST) サービスを構成するには、REST サービス・プロバイダー構成の管理コンソール・ページを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

このタスクについて

REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他の構成タイプの場合は、管理コンソール・ページを使用して、REST サービスを構成することができます。REST サービス・プロバイダー構成ページでは、選択したサービス・プロバイダーのすべてのサービスを表示したり、各サービスを個々に有効化/無効化したりすることができます。サーバーまたはクラスター (またはビジネス・プロセスまたはヒューマン・タスク・コンポーネント) を使用して REST サービスを管理する場合は、REST サービスの管理コンソール・ページを使用します。

手順

1. 「サービス」 → 「REST サービス」 → 「REST サービス・プロバイダー」 → をクリックします。

「REST サービス・プロバイダー (REST service providers)」ページが開き、REST サービス・プロバイダーがすべて表示されます。

2. プロバイダー・リンクをクリックし、そのプロバイダーによって管理される REST サービス・グループ用のサービスを構成します。

REST サービス・プロバイダーの構成ページが開き、そのプロバイダーでの REST サービスがすべて表示されます。

3. ランタイム環境で使用できるように構成するすべての REST サービスについて、リストから「**プロトコル**」を選択します。**https://** または **http://** を選択することによりフル URL パスを設定し、「**ホスト名または負荷分散環境の仮想ホスト (Host Name or Virtual Host in a Load-Balanced Environment)**」と「**ポート**」を入力します。完全修飾ホスト名を使用します。

REST 要求をアプリケーション・サーバーに直接送信する場合は、アプリケーション・サーバーのホスト名とポートを入力します。REST 要求を、1 台以上のアプリケーション・サーバーの前に配置されたプロキシ・サーバーまたは HTTP サーバーに送信する場合は、セットアップ済みのプロキシ・サーバーまたは HTTP サーバーのホスト名とポートを入力します。ブラウザと、Business Space および REST サービスの間に、ロード・バランサーまたはプロキシ・サーバーがある環境では、プロトコル、ホスト、およびポートの指定内容が、Business Space にアクセスするためのブラウザ URL と一致していることを確認してください。

4. プロバイダーに対する REST サービスがリストされたテーブル内の各行で、個々の REST サービスを有効化する場合は「**使用可能**」チェック・ボックスを選択し、個々の REST サービスを無効化する場合は「**使用可能**」チェック・ボックスをクリアします。
5. 有効化する個々のサービスについて、「**説明**」列にわかりやすい説明を入力します。
6. 「**OK**」をクリックして、サービスへの変更をコミットします。

サーバー、クラスター、またはコンポーネント用の REST サービスの構成

サーバー、クラスター、またはコンポーネント用の Representational State Transfer (REST) サービスを構成するには、REST サービスの管理コンソール・ページを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

このタスクについて

REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他の構成タイプの場合は、REST サービスの管理コンソール・ページを使用すると、サーバー、クラスター、またはコンポーネント用のサービスを構成することができます。

手順

1. 次のいずれかをクリックします。
 - サーバー上のシステム REST サービスの場合、「**サーバー**」 → 「**サーバー・タイプ**」 → 「**WebSphere Application Server**」 → 「**name_of_server**」 → 「**ビジネス・インテグレーション**」 → 「**REST サービス**」をクリックします。

- クラスター上のシステム REST サービスの場合、「サーバー」 → 「クラスター」 → 「WebSphere Application Server クラスター」 → 「*name_of_cluster*」 → 「ビジネス・インテグレーション」 → 「REST サービス」をクリックします。
- サーバー上のビジネス・プロセス REST サービスの場合は、「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「*name_of_server*」 → 「ビジネス・インテグレーション」 → 「Business Flow Manager」 → 「REST サービス」をクリックします。
- クラスター上のビジネス・プロセス REST サービスの場合は、「サーバー」 → 「クラスター」 → 「WebSphere Application Server クラスター」 → 「*name_of_cluster*」 → 「ビジネス・インテグレーション」 → 「Business Flow Manager」 → 「REST サービス」をクリックします。
- サーバー上のヒューマン・タスク REST サービスの場合は、「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「*name_of_server*」 → 「ビジネス・インテグレーション」 → 「Human Task Manager」 → 「REST サービス」をクリックします。
- クラスター上のヒューマン・タスク REST サービスの場合は、「サーバー」 → 「クラスター」 → 「WebSphere Application Server クラスター」 → 「*name_of_cluster*」 → 「ビジネス・インテグレーション」 → 「Human Task Manager」 → 「REST サービス」をクリックします。

「REST サービス」ページが表示され、サーバーまたはクラスター (または Business Flow Manager または Human Task Manager コンポーネント) で使用するために構成できるデフォルトの REST サービスがすべて表示されます。REST サービスが既に構成されている場合は、メッセージが表示されます。

2. ランタイム環境で使用できるように構成するすべての REST サービスについて、リストから「プロトコル」を選択します。**https://** または **http://** を選択することによりフル URL パスを設定し、「ホスト名または負荷分散環境の仮想ホスト (Host Name or Virtual Host in a Load-Balanced Environment)」と「ポート」を入力します。完全修飾ホスト名を使用します。

REST 要求をアプリケーション・サーバーに直接送信する場合は、アプリケーション・サーバーのホスト名とポートを入力します。REST 要求を、1 台以上のアプリケーション・サーバーの前に配置されたプロキシ・サーバーまたは HTTP サーバーに送信する場合は、セットアップ済みのプロキシ・サーバーまたは HTTP サーバーのホスト名とポートを入力します。ブラウザと、Business Space および REST サービスの間に、ロード・バランサーまたはプロキシ・サーバーがある環境では、プロトコル、ホスト、およびポートの指定内容が、Business Space にアクセスするためのブラウザ URL と一致していることを確認してください。

3. REST サービス・テーブル内の各行で、個々の REST サービスを有効化する場合は「使用可能」チェック・ボックスを選択し、個々の REST サービスを無効化する場合は「使用可能」チェック・ボックスをクリアします。
4. REST サービスの表で、それぞれの REST サービスの意味のある説明を「説明」フィールドに入力します。
5. 「OK」をクリックして、サービスへの変更をコミットします。

REST サービス構成を後で変更する場合は、「REST サービス」ページに戻るか、REST サービス・エンドポイントの構成を管理するための他の管理コンソール・ページを使用します。「REST サービス・プロバイダー」ページでは、構成するサービス・プロバイダーを選択できます。「サービス」→「REST サービス」からアクセスする「REST サービス」ページでは、現在の環境内での REST サービスをすべて構成することができます。

コマンド行を使用した REST サービスの構成

Representational State Transfer (REST) サービスをランタイム環境で使用するには、その前に構成を行う必要があります。REST サービスの管理コンソール・ページを使用しない場合は、`updateRESTGatewayService` コマンドを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

WebSphere Process Server で、Business Process Choreographer を構成済みの場合は、ヒューマン・タスク管理 REST サービスが既に構成されています。ただし、他の REST サービスのサービス・プロバイダーとなる REST サービス・ゲートウェイ・アプリケーションは、`updateRESTGatewayService` コマンドを使用して構成する必要があります。

手順

1. コマンド・ウィンドウを開きます。

`wsadmin` コマンドは、スタンドアロン・サーバー環境の場合は `profile_root/bin` ディレクトリにあり、Network Deployment 環境の場合は `deployment_manager_profile_root/bin` ディレクトリにあります。

2. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を開始します。
3. `updateRESTGatewayService` コマンドを使用して、クラスターまたはサーバーとノードを指定する REST サービスを構成します。 `-enable` パラメーターはオプションです。このパラメーターを指定しない場合は、デフォルト値の `true` に設定されます。
4. `save` コマンドを実行します。

例

次の例では、Jython を使用して、`updateRESTGatewayService` コマンドを実行し、変更を保存しています。クラスター上に REST サービスを構成します。

```
AdminTask.updateRESTGatewayService(['-clusterName
  cluster_name'])
AdminConfig.save()
```

次の例では Jacl を使用しています。

```
$AdminTask updateRESTGatewayService {-clusterName
  cluster_name}
$AdminConfig save
```

Business Process Choreographer の構成

Business Process Choreographer を構成する方法については、WebSphere Process Server for z/OS バージョン 7.0 インフォメーション・センターにアクセスし、「**WebSphere Process Server の構成**」>「**Business Process Choreographer の構成**」にあるトピックを参照してください。この情報は *Business Process Choreographer PDF* でも検索できます。

Business Spaceの構成

WebSphere が提供する Business Space をインストールおよび構成して、アプリケーション・ユーザーが IBM WebSphere ビジネス・プロセス・マネジメント・ポートフォリオ全体で Web インターフェースを作成、管理、および統合するための共通インターフェースをセットアップします。

始める前に

製品ソフトウェアをインストールする必要があります。製品をインストールすると、Business Spaceのファイルは、構成したプロファイルに対応するインストール済み環境に組み込まれます。

ヒューマン・タスク管理ウィジェットを必要とする WebSphere Process Server ランタイム環境では、Business Process Choreographer を構成する必要があります。詳しくは、WebSphere Process Server の資料の『Business Process Choreographer の構成』を参照してください。

このタスクについて

Business Space は、ご使用の WebSphere 製品のサポートに適合するように、以下のデータベース製品でサポートされます。

- Derby Embedded (WebSphere Business Monitor、WebSphere Business Services Fabric、WebSphere Enterprise Service Bus、および WebSphere Process Server の場合)。
- Derby Network Server (WebSphere Business Monitor、WebSphere Enterprise Service Bus、および WebSphere Process Server の場合)
- DB2 Universal (WebSphere Business Compass、WebSphere Business Monitor、WebSphere Business Services Fabric、WebSphere Enterprise Service Bus、および WebSphere Process Server の場合)
- DB2 for IBM i (WebSphere Enterprise Service Bus および WebSphere Process Server の場合)
- DB2 for z/OS (WebSphere Business Monitor、WebSphere Business Services Fabric、WebSphere Enterprise Service Bus、および WebSphere Process Server 用)。
- Microsoft® SQL Server Enterprise 2005 SP 2 および 2008 (WebSphere Business Services Fabric、WebSphere Enterprise Service Bus、および WebSphere Process Server の場合)。

- Oracle 11g (WebSphere Business Compass、WebSphere Business Monitor、WebSphere Business Services Fabric、WebSphere Enterprise Service Bus、および WebSphere Process Server の場合)

Monitor

Process Server / ESB

WebSphere Process

Server、WebSphere Enterprise Service Bus、または WebSphere Business Monitor をインストールしてスタンドアロン・サーバー・プロファイルを標準的なオプションを指定して作成する場合、Business Space のインストールと構成が Derby Embedded データベースを使用して自動的に行われます。スタンドアロン・サーバー・プロファイルを使用する場合、拡張オプションを持つプロファイル管理ツールを使用して Business Space を構成し、現在のランタイム環境で作業することができます。詳しくは、『プロファイル管理ツールを使用した Business Space の構成』を参照してください。

いずれの製品の場合も、デプロイメント・マネージャーおよびカスタム・プロファイルをセットアップしている場合、Business Space を構成する最も簡単な方法は、デプロイメント環境構成ウィザードを使用する方法です。詳しくは、『「デプロイメント環境構成」ウィザードを使用した Business Space の構成』を参照してください。

スタンドアロン・サーバー環境を使用しているか、または「デプロイメント環境構成」ウィザードを使用してランタイム環境を構成している場合、Representational State Transfer (REST) サービス・エンドポイントは自動的に構成されて使用可能になります。その他の環境については、REST サービス管理コンソールのページを使用して REST サービスを構成します。Business Space でウィジェットを使用可能にする場合は、それらのウィジェットに対する REST サービス・エンドポイントを構成する必要があります。REST エンドポイントは、Business Space がウィジェットをエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようにするために、登録する必要があります。

デプロイメント・マネージャーおよびカスタム・プロファイルを使用している場合は、管理コンソールを使用して Business Space を構成できます。

プロファイル管理ツールまたは管理コンソールで最初のセットアップ作業を行った後で、Business Space 用のデータベース表を構成する必要があります。詳しくは、『Business Space のデータベース表の構成』を参照してください。

Business Space を構成するためにどのツールを使用するかに関わらず、Business Space がご使用の環境のセキュリティーを伴って機能することを確認する必要があります。詳しくは、『Business Space のセキュリティーのセットアップ』を参照してください。

Business Space は Lotus® Mashups テクノロジーを基盤としてビルドされています。Lotus Mashups のよくある質問および一般的なトラブルシューティング情報については、<http://www.lotus.com/ldd/mashupswiki.nsf/xpViewCategories.xsp?lookupName=Troubleshooting&SessionID=CDFG4HK6EQ> を参照してください。

次のタスク

Business Space をインストールおよび構成した後に、ランタイム環境のユーザーは、URL: `http://host:port/BusinessSpace` からそれを開くことができます。ここで *host* はサーバーが実行されているホストの名前であり、*port* はサーバーのポート番号です。

プロファイル管理ツールを使用したBusiness Spaceの構成

プロファイル管理ツールを使用して、WebSphere が提供する Business Space を構成できます。

このタスクについて

製品のインストール後に、プロファイル管理ツールを開始できます。また、製品のインストール後に `manageprofiles` コマンド行ユーティリティのパラメーター `-configureBSpace` を使用することにより、コマンド行からプロファイル管理ツール機能を使用することができます。いずれの場合も、Business Spaceは、共通データベース用に指定されたデータベース製品と同じデータベース製品と共にインストールされます。Business Spaceでサポートされていないデータベースを選択した場合、プロファイル管理ツールは Derby Embedded データベースでBusiness Spaceを構成します。

Process Server / ESB プロファイル管理ツールは、WebSphere Process Server for z/OS および WebSphere Enterprise Service Bus for z/OS では使用できません。これらの製品では、管理コンソールを使用してBusiness Spaceを構成します。

いずれの製品の場合も、デプロイメント・マネージャーおよびカスタム・プロファイルに対しては、管理コンソールまたはデプロイメント環境構成ウィザードを使用できます。『管理コンソールを使用したBusiness Spaceの構成』または『デプロイメント環境構成ウィザードを使用したBusiness Spaceの構成』を参照してください。プロファイル管理ツールを使用して、「**デプロイメント環境**」プロファイル作成オプションを指定してデプロイメント・マネージャーおよびカスタム・プロファイル (管理対象ノード) を作成する場合、Business Space はご使用のデプロイメント環境で自動的に構成されますが、データベース表を構成するためのスクリプトは手動で実行する必要があります。

スタンドアロン・サーバー・プロファイルで拡張構成を行う場合は、管理コンソールのページを使用してBusiness Spaceを構成する必要があります。例えば、プロファイル用に選択したデータベース (WebSphere Business Monitor データベース、WebSphere Business Compass データベース、または WebSphere Process Server 共通データベース) とは異なるデータ・ソースを指定する場合は、管理コンソールを使用してBusiness Spaceを構成する必要があります。

管理コンソールを使用する必要がある拡張構成オプションを使用することにした場合は、必ず以下のステップを実行してください。

- プロファイル管理ツールを使用してスタンドアロン・サーバー・プロファイルを作成する場合は、「**詳細プロファイル作成**」オプションを使用して「**ビジネス・スペースの構成**」チェック・ボックスをクリアします。この操作により、後から管理コンソールを使用してBusiness Spaceを構成することができます。

- 『管理コンソールを使用したBusiness Spaceの構成』を参照してください。

スタンドアロン・サーバーを構成する場合は、ステップ 1 を実行します。デプロイメント環境を構成する場合は、ステップ 2 を実行します。

手順

1. スタンドアロン・サーバーの場合は、プロファイル管理ツールを開始し、「**スタンドアロン・サーバー・プロファイル (Stand-alone server profile)**」オプションを選択して、以下のステップを実行します。
 - a. 「プロファイル作成オプション」ページで、以下のいずれかのステップを実行します。
 - Derby Embedded データベースを使用したBusiness Spaceのデフォルトのインストールと構成をそのまま使用する場合は、「**標準プロファイル作成**」オプションを選択します。
 - 作成しているプロファイルの拡張オプションを構成する場合は、「**拡張**」オプションを選択します。「Business Spaceの構成」ページで、「**ビジネス・スペースの構成**」チェック・ボックスが選択されていることを確認します。Business Space 内のヒューマン・タスク管理ウィジェットとともに動作するよう Lotus Webform Server を構成する場合、「**Lotus Webform Server の構成 (Configure Lotus Webform Server)**」チェック・ボックスを選択し、Webform Server Translator およびインストールのルートを入力します。

Business Space は、ご使用の製品のデータ・ソースを使用して構成されます。プロファイル管理ツールと IBM WebSphere Dynamic Process Edition を使用している場合、Business Space は WebSphere Process Server のデータ・ソースを使用して構成されます。
 - b. プロファイルのホスト名を指定するときには、完全修飾ホスト名を使用してください。
 - c. 「データベース設計」ページでは、Business Spaceのデータベース構成情報を含め、製品のすべてのデータベース構成が含まれる、データベース設計ツールを使用して作成したデータベース設計ファイルを使用するオプションを選択できます。
 - d. プロファイル管理ツールを使用して、プロファイルを作成します。Business Spaceがインストールされました。これは、共通データベース用に指定したデータベース製品と同じデータベース製品用に構成されます (または、データベース製品がサポートされていない場合は、Derby Embedded データベース用に構成されます)。
 - e. データベースがリモートにある場合、プロファイル管理ツールの実行後にデータベース表を構成する必要があります。『Business Space のデータベース表の構成』を参照してください。
2. デプロイメント環境の場合は、プロファイル管理ツールを開始し、「**デプロイメント・マネージャー・プロファイル**」または「**カスタム・プロファイル**」オプションを選択して、以下のステップを実行します。

- a. 「プロファイル作成オプション」 ページで、「**デプロイメント環境**」 オプションを選択し、カスタマイズ済みの構成値を使用して各プロファイルを構成し、提供されたパターンに基づいてデプロイメント環境でそのプロファイルを使用します。
- b. プロファイル管理ツールのステップを実行して、デプロイメント・マネージャー・プロファイルおよびカスタム・プロファイル (管理対象ノード) を作成します。
- c. すべてのカスタム・ノードを統合した後に、データベース表を構成するためのスクリプトを手動で実行します。『Business Space のデータベース表の構成』を参照してください。

次のタスク

注: 製品データベースが Oracle データベースの場合、Business Space は、プロファイル管理ツールまたは `manageprofiles` コマンド行ユーティリティーで、デフォルトのスキーマ `IBMBUSSP` とプロファイル作成時に入力したデフォルトのパスワードを使用して、同じデータベースを使用するように構成されます。`IBMBUSSP` ユーザー名に別のパスワードを使用する場合は、管理コンソールを使用して、JDBC リソースを以下のように更新する必要があります。データ・ソース `jdbc/mashupsDS` を探します。Business Space スキーマ名のパスワードと同じようになるように、認証別名の値を変更します。変更内容を保存して、サーバーを再始動します。

Business Space を使用する前に、チームが使用する Business Space とウィジェットで使用する必要のあるセキュリティーをセットアップしてください。詳しくは、『Business Space のセキュリティーのセットアップ』を参照してください。

注: Business Space は、プロキシ・コンポーネントを使用して REST サービスに接続します。REST サービスの反応がない場合、REST サービス・サーバーのパフォーマンスに基づいて、Business Space から REST サービスへの接続タイムアウト設定を更新する必要があります。REST サービス接続がタイムアウトになる場合は、以下の設定を更新します。デフォルトで、`socket-timeout` 値は 30 秒に設定されています。この値を状況に適した値に変更します。

1. ファイル `profile_root/BusinessSpace/node_name/server_name/mm.runtime.prof/config/proxy-config.xml`を開きます。
2. `socket-timeout` の `proxy:value` を変更します。時間はミリ秒で指定します。

```
<proxy:meta-data>
  <proxy:name>socket-timeout</proxy:name>
  <proxy:value>30000</proxy:value>
</proxy:meta-data>
```
3. `wsadmin` スクリプト・クライアントを使用して、`updateBlobConfig` コマンドを実行します。その際には、**-serverName** および **-nodeName** パラメーター (スタンドアロン・サーバーの場合)、または **-clusterName** パラメーター (クラスタの場合) を指定すると共に、**-propertyFileName** パラメーターに `proxy-config.xml` ファイルのパスを値として指定し、**-prefix** パラメーターに値 `Mashups_` を指定します。
4. 管理コンソールから `mm_was_node_server` アプリケーションを再始動するか、サーバー・アプリケーション全体を再始動します。

「デプロイメント環境構成」ウィザードの一部としてBusiness Spaceを構成する

Business Spaceの構成や、Business Spaceのウィジェット用の Representational State Transfer (REST) サービス構成は、「デプロイメント環境構成」ウィザードに自動的に組み込まれます。どの REST サービスを構成するかを選択することができます。

始める前に

タスクを開始する前に、以下のタスクを完了しておく必要があります。

- 製品をインストールします。
- プロファイルを作成します。このとき、必ずプロファイルの完全修飾ホスト名を指定してください。
- Business Space用に機密保護機能のある環境をセットアップする場合は、セキュリティーを使用可能にします。

このタスクについて

デプロイメント・マネージャーおよびカスタム・プロファイルをセットアップしている場合、これがBusiness Spaceを構成するための最も簡単な方法です。

手順

1. 管理コンソールで、「サーバー」 → 「デプロイメント環境」 → 「新規」をクリックします。デプロイメント環境の作成プロセスがウィザードの一連のページで示されます。
2. 新規デプロイメント環境を定義するか、デプロイメント環境定義が記述されたファイルをインポートします。IBM 提供のパターンに基づいてデプロイメント環境を作成することができます。カスタムのデプロイメント環境を作成することもできます。
3. 「デプロイメント環境パターン」ページで、任意のデプロイメント環境パターンを選択します。
4. 「ノードの選択」ページで、デプロイメント環境に参加するノードを指定します。
5. 「クラスター」ページで、特定のデプロイメント環境機能に割り当てる各ノードのクラスター・メンバー数を指定します。
6. 「データベース」ページで、表にリストされるコンポーネントの 1 つである Business Spaceのデータ・ソースを構成します。説明の編集、接続のテスト、プロバイダーに対して使用するデータベース製品をそれぞれ設定することができます。Business Spaceの場合は、このページの「テーブルの作成」チェック・ボックスを選択することはできません。Business Spaceでは、データベース表を手動で構成する必要があります。データベース製品リストには、各コンポーネントでサポートされるすべてのデータベースが表示されます。
7. 「セキュリティー」ページで、セキュア・コンポーネントへのアクセス時に WebSphere が使用する認証別名を構成します。認証別名のユーザー名とパスワードは、このページで変更することができます。これらの別名を使用してセキュア・コンポーネントにアクセスしますが、データ・ソースにアクセスすることはできません。

8. WebSphere Process Server 構成の場合は、アプリケーション・デプロイメント・ターゲットの構成に必要な情報を指定して、 Business Process Choreographer コンポーネントのデプロイメントをサポートします。このデプロイメント環境の Business Process Choreographer の構成に使用されるコンテキスト・ルート、セキュリティー、および Human Task Manager メール・セッションの値を指定します。
9. WebSphere Process Server 構成の場合は、クラスターまたはサーバーで稼働するようにビジネス・ルール・マネージャーを構成します。
10. REST サービス・ページで、ランタイム環境用の Business Space で使用できるようにするウィジェットのサービスを構成します。
 - クライアントがサーバーまたはクラスターと通信するために必要なポート番号とホストまたは仮想ホストを入力します。
 - ホスト・フィールドとポート・フィールドを空のままにした場合、それらの値は、個々のクラスター・メンバーのホストとその HTTP ポートの値へとデフォルト設定されます。負荷分散環境の場合は、後でこれらのデフォルト値を、環境の仮想ホスト名およびポートへと変更する必要があります。必ず完全修飾ホスト名を指定してください。
 - 必要に応じて、ウィジェットの説明を設定します。
11. 次のページで、「終了」または「終了して環境を生成」をクリックします。
12. デプロイメント環境またはクラスターを始動する前に、Business Space のデータベース表を構成するスクリプトを実行します。詳しくは、『Business Space のデータベース表の構成』を参照してください。

次のタスク

Business Space は、プロキシ・コンポーネントを使用して REST サービスに接続します。REST サービスの反応がない場合、REST サービス・サーバーのパフォーマンスに基づいて、Business Space から REST サービスへの接続タイムアウト設定を更新する必要があります。REST サービス接続がタイムアウトになる場合は、以下の設定を更新します。デフォルトで、socket-timeout 値は 30 秒に設定されています。この値を状況に適した値に変更します。

1. ファイル `profile_root/BusinessSpace/node_name/server_name/mm.runtime.prof/config/proxy-config.xml`を開きます。
2. socket-timeout の proxy:value を変更します。時間はミリ秒で指定します。


```
<proxy:meta-data>
  <proxy:name>socket-timeout</proxy:name>
  <proxy:value>30000</proxy:value>
</proxy:meta-data>
```
3. wsadmin スクリプト・クライアントを使用して、updateBlobConfig コマンドを実行します。その際には、**-serverName** および **-nodeName** パラメーター (スタンドアロン・サーバーの場合)、または **-clusterName** パラメーター (クラスターの場合) を指定すると共に、**-propertyFileName** パラメーターに proxy-config.xml ファイルのパスを値として指定し、**-prefix** パラメーターに値 Mashups_ を指定します。
4. 管理コンソールから mm_was_node_server アプリケーションを再始動するか、サーバー・アプリケーション全体を再始動します。

ネットワーク・デプロイメント環境のための Business Space の構成

分散環境またはネットワーク・デプロイメント環境を使用している場合は、管理コンソールまたは管理コマンドを使用して Business Space を構成します。

このタスクについて

デプロイメント・マネージャーおよびカスタム・プロファイルを使用している場合、Representational State Transfer (REST) エンドポイントを構成し、Business Space を構成し、REST エンドポイントを登録し、データベース表を構成する必要があります。

REST サービスの構成

スタンドアロン・サーバー環境を使用しているか、または「デプロイメント環境構成」ウィザードを使用してランタイム環境を構成している場合、Representational State Transfer (REST) サービスは自動的に構成されて使用可能になります。その他の環境については、管理コンソールを使用して REST サービスを構成します。

このタスクについて

Business Space でウィジェットを使用可能にする場合は、それらのウィジェットに対する REST サービスを構成する必要があります。後で、REST エンドポイントを登録して、Business Space がウィジェットをエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようにする必要があります。

特定のサーバーまたはクラスターに対してすべての REST サービスを構成することができます。または、個々のサービスを選択して構成することもできます。個々のサービスの構成は、あるサービス・プロバイダーについてのすべてのサービスを表示したり、あるいは環境のすべてのサービスを表示したりすることによって管理できます。

管理コンソールでのすべての REST サービスの構成:

すべての Representational State Transfer (REST) サービスを環境に合わせて構成するには、REST サービスの管理コンソール・ページを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

このタスクについて

REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他の構成タイプの場合は、管理コンソール・ページを使用すると、Business Space 内のすべての製品ウィジェットに関する REST サービスを構成することができます。「REST サービス」ページでは、現在の環境に対するすべてのサービスを表示したり、各サービスを個々に有効化/無効化したりすることができます。

また、REST エンドポイントはBusiness Spaceに登録する必要があります。すると、Business Spaceがウィジェットをこれらのエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようになります。

同じ REST サービス・エンドポイントの複数のインスタンスを構成する場合は、エンドポイント・ファイルとウィジェット・メタデータ・ファイルを手動で編集する必要があります。詳しくは、『複数のエンドポイント用にBusiness Space・ウィジェットを使用可能にする』を参照してください。

手順

1. 「サービス」 → 「REST サービス」 → 「REST サービス」をクリックします。

「REST サービス」ページが開き、現在の環境内の REST サービスがすべて表示されます。

2. 「スコープセクション」で、すべてを指定して、現在の環境内の REST サービスをすべて表示するか、REST サービスが有効になっているサーバーまたはクラスターを選択します。
3. プロバイダーに対する REST サービスがリストされたテーブル内の各行で、個々の REST サービスを有効化する場合は「使用可能」チェック・ボックスを選択し、個々の REST サービスを無効化する場合は「使用可能」チェック・ボックスをクリアします。
4. 有効化する個々のサービスについて、「説明」列にわかりやすい説明を入力します。
5. 「OK」をクリックして、サービスへの変更をコミットします。

次のタスク

- Business Spaceを構成します。
- データベース表を構成します (リモート・データベースまたはネットワーク・デプロイメント環境を使用する場合)。
- REST サービス・エンドポイントに登録します。
- サービス・エンドポイントのインスタンスが複数ある場合 (例えば 2 つのクラスターで作業を分割しており、ウィジェットで各クラスターのデータを表示する場合など)、追加のクラスターごとに手動で追加のウィジェットを使用可能にする必要があります。
- Business Spaceのセキュリティーをセットアップします。

サービス・プロバイダーでの REST サービスの構成:

サービス・プロバイダー内の Representational State Transfer (REST) サービスを構成するには、REST サービス・プロバイダー構成の管理コンソール・ページを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

このタスクについて

REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他のタイプの構成では、管理コンソールでBusiness Space内の製品のすべてのウィジェットに対して REST サービスを構成することができます。管理コンソールの「REST サービス・プロバイダー構成」ページでは、選択されたサービス・プロバイダーのすべてのサービスを表示し、各サービスを個別に使用可能または使用不可に設定できます。このページでは、サービス・プロバイダーについてのすべてのサービスを処理することで個々のサービス構成を管理できます。サーバーまたはクラスター（あるいはビジネス・プロセスまたはヒューマン・タスク・コンポーネント）ごとに REST サービスを管理したい場合は、管理コンソールの「REST サービス」ページを使用してください。

また、REST エンドポイントはBusiness Spaceに登録する必要があります。すると、Business Spaceがウィジェットをこれらのエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようになります。

同じ REST サービス・エンドポイントの複数のインスタンスを構成する場合は、エンドポイント・ファイルとウィジェット・メタデータ・ファイルを手動で編集する必要があります。詳しくは、『Business Space・ウィジェットが複数のエンドポイントを処理できるようにする』を参照してください。

手順

1. 「サービス」 → 「REST サービス」 → 「REST サービス・プロバイダー」 → をクリックします。

「REST サービス・プロバイダー (REST service providers)」ページが開き、REST サービス・プロバイダーがすべて表示されます。

2. プロバイダー・リンクをクリックし、そのプロバイダーによって管理される REST サービス・グループ用のサービスを構成します。

REST サービス・プロバイダーの構成ページが開き、そのプロバイダーでの REST サービスがすべて表示されます。

3. Business Spaceで使用するすべての REST サービスについての**プロトコル**をリストから選択します。「**https://**」または「**http://**」のいずれかを選択してから、「**負荷分散環境でのホスト名または仮想ホスト**」フィールドと「**ポート**」フィールドに入力して、絶対 URL パスを構成します。完全修飾ホスト名を使用しません。

REST 要求をアプリケーション・サーバーに直接送信する場合は、アプリケーション・サーバーのホスト名とポートを入力します。REST 要求を、1 台以上のアプリケーション・サーバーの前に配置されたプロキシ・サーバーまたは HTTP サーバーに送信する場合は、セットアップ済みのプロキシ・サーバーまたは HTTP サーバーのホスト名とポートを入力します。ブラウザと、Business Space および REST サービスの間に、ロード・バランサーまたはプロキシ・サーバーがある環境では、プロトコル、ホスト、およびポートの指定内容が、Business Space にアクセスするためのブラウザ URL と一致していることを確認してください。

4. プロバイダーに対する REST サービスがリストされたテーブル内の各行で、個々の REST サービスを有効化する場合は「**使用可能**」チェック・ボックスを選択し、個々の REST サービスを無効化する場合は「**使用可能**」チェック・ボックスをクリアします。
5. 有効化する個々のサービスについて、「**説明**」列にわかりやすい説明を入力します。
6. 「**OK**」をクリックして、サービスへの変更をコミットします。

次のタスク

- Business Spaceを構成します。
- データベース表を構成します (リモート・データベースまたはネットワーク・デプロイメント環境を使用する場合)。
- REST サービス・エンドポイントを登録します。
- サービス・エンドポイントのインスタンスが複数ある場合 (例えば 2 つのクラスターで作業を分割しており、ウィジェットで各クラスターのデータを表示する場合など)、追加のクラスターごとに手動で追加のウィジェットを使用可能にする必要があります。
- Business Spaceのセキュリティーをセットアップします。

サーバー、クラスター、またはコンポーネント用の REST サービスの構成:

サーバー、クラスター、またはコンポーネント用の Representational State Transfer (REST) サービスを構成するには、REST サービスの管理コンソール・ページを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

このタスクについて

REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他の構成タイプの場合は、REST サービスの管理コンソール・ページを使用すると、サーバー、クラスター、またはコンポーネント用のサービスを構成することができます。

また、REST エンドポイントはBusiness Spaceに登録する必要があります。すると、Business Spaceがウィジェットをこれらのエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようになります。

同じ REST サービス・エンドポイントの複数のインスタンスを構成する場合は、エンドポイント・ファイルとウィジェット・メタデータ・ファイルを手動で編集する必要があります。詳しくは、『Business Space・ウィジェットが複数のエンドポイントを処理できるようにする』を参照してください。

手順

1. 次のいずれかをクリックします。

- サーバー上の REST サービスの場合、「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「name_of_server」 → 「ビジネス・インテグレーション」 → 「REST サービス」をクリックします。
- クラスター上の REST サービスの場合、「サーバー」 → 「クラスター」 → 「Websphere Application Server クラスター (WebSphere application server clusters)」 → 「name_of_cluster」 → 「ビジネス・インテグレーション」 → 「REST サービス」をクリックします。
- サーバー上のビジネス・プロセス REST サービスの場合は、「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「name_of_server」 → 「ビジネス・インテグレーション」 → 「Business Flow Manager」 → 「REST サービス」をクリックします。
- クラスター上のビジネス・プロセス REST サービスの場合は、「サーバー」 → 「クラスター」 → 「Websphere Application Server クラスター」 → 「name_of_cluster」 → 「ビジネス・インテグレーション」 → 「Business Flow Manager」 → 「REST サービス」をクリックします。
- サーバー上のヒューマン・タスク REST サービスの場合は、「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「name_of_server」 → 「ビジネス・インテグレーション」 → 「Human Task Manager」 → 「REST サービス」をクリックします。
- クラスター上のヒューマン・タスク REST サービスの場合は、「サーバー」 → 「クラスター」 → 「Websphere Application Server クラスター」 → 「name_of_cluster」 → 「ビジネス・インテグレーション」 → 「Human Task Manager」 → 「REST サービス」をクリックします。

「REST サービス」ページが表示され、ご使用の製品またはコンポーネント (Business Flow Manager または Human Task Manager) とともに使用する Business Space・ウィジェット用に構成できるすべてのデフォルト REST サービスが表示されます。REST サービスが既に構成されている場合は、メッセージが表示されます。

2. Business Spaceで使用するすべての REST サービスについてのプロトコルをリストから選択します。「https://」または「http://」のいずれかを選択してから、「負荷分散環境でのホスト名または仮想ホスト」フィールドと「ポート」フィールドに入力して、絶対 URL パスを構成します。完全修飾ホスト名を使用します。

REST 要求をアプリケーション・サーバーに直接送信する場合は、アプリケーション・サーバーのホスト名とポートを入力します。REST 要求を、1 台以上のアプリケーション・サーバーの前に配置されたプロキシ・サーバーまたは HTTP サーバーに送信する場合は、セットアップ済みのプロキシ・サーバーまたは HTTP サーバーのホスト名とポートを入力します。ブラウザーと、Business Space および REST サービスの間に、ロード・バランサーまたはプロキシ・サーバーがある環境では、プロトコル、ホスト、およびポートの指定内容が、Business Space にアクセスするためのブラウザー URL と一致していることを確認してください。

3. REST サービス・テーブル内の各行で、個々の REST サービスを有効化する場合は「使用可能」チェック・ボックスを選択し、個々の REST サービスを無効化する場合は「使用可能」チェック・ボックスをクリアします。

4. REST サービスの表で、それぞれの REST サービスの意味のある説明を「説明」フィールドに入力します。
5. 「OK」をクリックして、サービスへの変更をコミットします。

REST サービス構成を後で変更する場合は、「REST サービス」ページに戻るか、REST サービス・エンドポイントの構成を管理するための他の管理コンソール・ページを使用します。「REST サービス・プロバイダー」ページでは、構成するサービス・プロバイダーを選択できます。「サービス」→「REST サービス」からアクセスする「REST サービス」ページでは、現在の環境内での REST サービスをすべて構成することができます。

次のタスク

- Business Spaceを構成します。
- データベース表を構成します (リモート・データベースまたはネットワーク・デプロイメント環境を使用する場合)。
- REST サービス・エンドポイントを登録します。
- サービス・エンドポイントのインスタンスが複数ある場合 (例えば 2 つのクラスターで作業を分割しており、ウィジェットで各クラスターのデータを表示する場合)、追加のクラスターごとに手動で追加のウィジェットを使用可能にする必要があります。
- Business Spaceのセキュリティーをセットアップします。

コマンド行を使用した REST サービスの構成:

ご使用の製品で必要となるすべてのウィジェットは、WebSphere が提供する Business Space と共にインストールされます。チームがBusiness Space内でウィジェットを使用するには、その前にウィジェットの Representational State Transfer (REST) サービスを構成し、使用可能にして、Business Spaceに登録しておく必要があります。REST サービス管理コンソールのページを使用しない場合、updateRESTGatewayService コマンドを使用します。

始める前に

このタスクを実行する前に、WebSphere Business Process Management 製品をインストールしておく必要があります。

WebSphere Process Server で、Business Process Choreographer を構成済みの場合は、ヒューマン・タスク管理 REST サービスが既に構成されています。ただし、他の REST サービスのサービス・プロバイダーとなる REST サービス・ゲートウェイ・アプリケーションは、updateRESTGatewayService コマンドを使用して構成する必要があります。

このタスクについて

REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他のタイプの構成では、REST サービス管理コンソールのページまたは updateRESTGatewayService コマンドによって、Business Space内の製品のすべてのウィジェットに対して REST アプリケーション・プログラミング・インターフェース (API) のサービスを構成することができます。

また、REST エンドポイントはBusiness Spaceに登録する必要があります。すると、Business Spaceがウィジェットをこれらのエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようになります。

同じ REST サービス・エンドポイントの複数のインスタンスを構成する場合は、エンドポイント・ファイルとウィジェット・メタデータ・ファイルを手動で編集する必要があります。詳しくは、『複数のエンドポイント用にBusiness Space・ウィジェットを使用可能にする』を参照してください。

手順

1. コマンド・ウィンドウを開きます。

`wsadmin` コマンドは、スタンドアロン・サーバー環境の場合は `profile_root/bin` ディレクトリにあり、Network Deployment 環境の場合は `deployment_manager_profile_root/bin` ディレクトリにあります。

2. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を開始します。
3. `updateRESTGatewayService` コマンドを使用して、クラスターまたはサーバーとノードを指定する REST サービスを構成します。 **-enable** パラメーターはオプションです。このパラメーターを指定しない場合は、デフォルト値の `true` に設定されます。
4. `save` コマンドを実行します。

例

次の例では、`Jython` を使用して、`updateRESTGatewayService` コマンドを実行し、変更を保存しています。クラスター上に REST サービスを構成します。

```
AdminTask.updateRESTGatewayService('[-clusterName
  cluster_name]')
AdminConfig.save()
```

`Jacl` を使用した例を以下に示します。

```
$AdminTask updateRESTGatewayService {-clusterName
  cluster_name}
$AdminConfig save
```

次のタスク

- Business Spaceを構成します。
- データベース表を構成します (リモート・データベースまたはネットワーク・デプロイメント環境を使用する場合)。
- REST サービス・エンドポイントを登録します。
- サービス・エンドポイントのインスタンスが複数ある場合 (例えば 2 つのクラスターで作業を分割しており、ウィジェットで各クラスターのデータを表示する場合など)、追加のクラスターごとに手動で追加のウィジェットを使用可能にする必要があります。
- Business Spaceのセキュリティーをセットアップします。

管理コンソールでの Business Space の構成および REST エンドポイントの登録

管理コンソールを使用して、WebSphere が提供する Business Space をインストールして構成することができます。

始める前に

タスクを開始する前に、以下のタスクを完了しておく必要があります。

- 製品ソフトウェアをインストールしてプロファイルを作成します。製品をインストールすると、Business Space のファイルは、セットアップしたプロファイルに対応するインストール済み環境に組み込まれます。プロファイルで明示的に Business Space を構成するまでは、プロファイルは Business Space 用に構成されません。
- ヒューマン・タスク管理ウィジェットを必要とする WebSphere Process Server ランタイム環境の Business Process Choreographer を構成します。詳しくは、WebSphere Process Server の資料の『Business Process Choreographer の構成』を参照してください。
- Business Space 用に機密保護機能のある環境をセットアップする場合は、セキュリティーを使用可能にします。
- Representational State Transfer (REST) サービスを構成します。スタンドアロン・サーバー環境を使用しているか、または「デプロイメント環境構成」ウィザードを使用してランタイム環境を構成している場合、REST サービス・エンドポイントは自動的に構成されて使用可能になります。その他の環境については、REST サービス管理コンソールのページを使用して REST サービスを構成します。Business Space でウィジェットを使用可能にする場合は、それらのウィジェットに対する REST サービスを構成する必要があります。「ビジネス・スペース構成」管理コンソール・ページで REST エンドポイントを登録して、Business Space がウィジェットをエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようにします。
- 製品のデータ・ソース以外のデータ・ソースを使用して、サーバーまたはクラスターに Business Space を構成する場合は、管理コンソールを使用して Business Space を構成する前に、jdbc/mashupDS の正しい JNDI 名を指定して、サーバーまたはクラスター・スコープにデータ・ソースを作成します。
- Oracle の場合、製品データベースで使用されるものとは別のスキーマを Business Space テーブルに使用するには、「Business Space 構成」ページを開く前に、以下のステップを実行してデータ・ソースを手動で作成します。
 - データベース製品ソフトウェアを使用してスキーマを作成します。
 - 管理コンソールを使用して JDBC プロバイダーを構成します。
 - 管理コンソールを使用して、JNDI 名が jdbc/mashupDS のデータ・ソースを、ご使用の環境に応じてサーバー・スコープまたはクラスター・スコープで作成します。
 - 管理コンソールを使用して認証別名を作成します。作成したスキーマにユーザー名を設定し、使用する Oracle のセットアップに応じて認証を設定します。
 - データ・ソースの認証別名を設定します。

このタスクについて

デプロイメント環境やその他の拡張プロファイル構成を使用する場合は、管理コンソールを使用して、Business Spaceをランタイム環境で稼働するよう設定する必要があります。Business Spaceは、セットアップしたプロファイルで稼働するアプリケーションのビジネス・ユーザー向けの、ブラウザ・ベースのグラフィカル・ユーザー・インターフェースです。Business Spaceでは、管理者およびアプリケーション・ユーザーは、WebSphere Business Process Management ポートフォリオの製品からコンテンツをカスタマイズできます。

手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「サーバー」 → 「サーバー・タイプ (Server Types)」 → 「WebSphere アプリケーション・サーバー (WebSphere application servers)」または「サーバー」 → 「クラスター」 → 「Websphere Application Server クラスター (WebSphere application server clusters)」をクリックします。
3. サーバー・ターゲットまたはクラスター・ターゲットの名前を選択します。
4. 「構成」 ページで、「ビジネス・インテグレーション」 の下の「ビジネス・スペース構成」 をクリックします。「Business Spaceの構成」 ページが表示されます。既にBusiness Spaceが構成されている場合は、このページが表示されてもフィールドを編集することはできません。
5. 「ビジネス・スペース・サービスのインストール」 チェック・ボックスを選択します。
6. 「データベース・スキーマ名」 のボックスに、Business Space・データベースで使用するデータベース・スキーマの名前を入力します。

注: Oracle の場合、スキーマはデータ・ソースの認証別名に設定されたユーザー名と同じです。

7. 「既存のビジネス・スペース・データ・ソース」 フィールドでデータ・ソースが指定されていない場合は、「以下を使用して、ビジネス・スペース・データ・ソースを作成:」 に移動して、Business Spaceで使用するデータベースに接続するデータ・ソースを選択します。

「以下を使用して、ビジネス・スペース・データ・ソースを作成:」 の下でデータ・ソースを指定すると、選択したデータ・ソース上でモデル化された、JNDI 名が jdbc/mashupDS の Business Space用のデータ・ソースが作成されます。

Business Spaceのデータ・ソースは、製品のデータ・ソースが別のサーバーまたはクラスターにあるとしても、Business Spaceを構成しているサーバーまたはクラスター上に作成されます。

注: 使用したい既存のデータ・ソースがリストにない場合は、「Business Spaceの構成」 ページをキャンセルして使用したいデータベースおよびデータ・ソースを設定し、「Business Spaceの構成」 ページを再開して構成を完了する必要があります。詳しくは、『始める前に』のセクションを参照してください。

8. 「OK」 をクリックします。

9. Business Spaceで使用する各ウィジェットのための、システム Representational State Transfer (REST) エンドポイントに対する適切なデプロイメント・ターゲット (クラスターまたはサーバー) を登録するには、「**REST サービス・エンドポイント登録**」をクリックします。REST サービス・エンドポイント・タイプに対して選択するターゲットによって、一部のウィジェットに表示されるデータの有効範囲を設定することができます。あるいは、パフォーマンスまたは可用性を向上させるために、特定のクラスターまたはサーバーを選択してもよいでしょう。ターゲットを指定しない場合、このタイプの REST エンドポイントは Business Space に登録されず、このタイプの REST サービス・エンドポイントが必要なすべてのウィジェットは Business Space に表示されません。
10. 構成を保管します。
11. デプロイメント環境またはクラスターを始動する前に、Business Space のデータベース表を構成するスクリプトを実行します。このスクリプトは、構成が完了したときに生成されました。詳しくは、『Business Space のデータベース表の構成』を参照してください。

次のタスク

注: Oracle を使用している場合、Business Space データ・ソースの認証別名のパスワードは、Business Space のスキーマ名と同じに設定されます。スキーマのデフォルト値は `IBMBUSSP` です。Business Space の構成時に、管理コンソールまたはコマンド行で別のスキーマを指定できます。その場合、デフォルトのパスワードは、指定するスキーマと同じです。Business Space ユーザー名に別のパスワードを使用する場合は、管理コンソールを使用して、JDBC リソースを更新する必要があります。データ・ソース `jdbc/mashupsDS` を検索します。Business Space スキーマ名のパスワードと同じようになるように、認証別名の値を変更します。変更内容を保存して、サーバーを再始動します。

注: Business Space は、プロキシ・コンポーネントを使用して REST サービスに接続します。REST サービスの反応がない場合、REST サービス・サーバーのパフォーマンスに基づいて、Business Space から REST サービスへの接続タイムアウト設定を更新する必要があります。REST サービス接続がタイムアウトになる場合は、以下の設定を更新します。デフォルトで、`socket-timeout` 値は 30 秒に設定されています。この値を状況に適した値に変更します。

1. ファイル `profile_root/BusinessSpace/node_name/server_name/mm.runtime.prof/config/proxy-config.xml`を開きます。
2. `socket-timeout` の `proxy:value` を変更します。時間はミリ秒で指定します。

```
<proxy:meta-data>
  <proxy:name>socket-timeout</proxy:name>
  <proxy:value>30000</proxy:value>
</proxy:meta-data>
```
3. `wsadmin` スクリプト・クライアントを使用して、`updateBlobConfig` コマンドを実行します。その際には、**-serverName** および **-nodeName** パラメーター (スタンドアロン・サーバーの場合)、または **-clusterName** パラメーター (クラスターの場合) を指定すると共に、**-propertyFileName** パラメーターに `proxy-config.xml` ファイルのパスを値として指定し、**-prefix** パラメーターに値 `Mashups_` を指定します。

4. 管理コンソールから `mm_was_node_server` アプリケーションを再始動するか、サーバー・アプリケーション全体を再始動します。

コマンド行を使用したBusiness Spaceの構成

`wsadmin` コマンドを使用して、WebSphere が提供する Business Spaceをセットアップおよび構成できます。 `wsadmin` コマンドを使用して、管理コンソールで実行できるのと同じBusiness Spaceの構成を実行できます。

始める前に

タスクを開始する前に、以下のタスクを完了しておく必要があります。

- 製品ソフトウェアをインストールしてプロファイルを作成します。製品をインストールすると、Business Spaceのファイルは、セットアップしたプロファイルに対応するインストール済み環境に組み込まれます。プロファイルで明示的にBusiness Spaceを構成するまでは、プロファイルはBusiness Space用に構成されません。
- タスクの管理ウィジェットとワークフロー・ウィジェットを必要とする WebSphere Process Server ランタイム環境の Business Process Choreographer を構成します。詳しくは、WebSphere Process Server の資料の『Business Process Choreographer の構成』を参照してください。
- Business Space用に機密保護機能のある環境をセットアップする場合は、セキュリティーを使用可能にします。
- Representational State Transfer (REST) サービスを構成します。スタンドアロン・サーバー環境を使用しているか、または「デプロイメント環境構成」ウィザードを使用してランタイム環境を構成している場合、REST サービス・エンドポイントは自動的に構成されて使用可能になります。その他の環境については、REST サービス管理コンソールのページを使用して REST サービスを構成します。Business Space でウィジェットを使用可能にする場合は、それらのウィジェットに対する REST サービス・エンドポイントを構成する必要があります。REST エンドポイントは、Business Space がウィジェットをエンドポイントに関連付け、ウィジェットがパレットに表示されて使用できるようにするために、登録する必要があります。
- 製品のデータ・ソース以外のデータ・ソースを使用して、サーバーまたはクラスターに Business Space を構成する場合は、Business Space を構成する前 (`configureBusinessSpace` コマンドを実行する前) に、正しい JNDI 名である `jdbc/mashupDS` を指定して、サーバーまたはクラスター・スコープにデータ・ソースを作成します。
- Oracle の場合、製品データベースで使用されるスキーマとは異なるスキーマを Business Space テーブルに使用するには、Business Space をインストールおよび構成するためのコマンドをこの後の手順で実行する前に、以下のステップを実行してデータ・ソースを手動で作成します。
 - 管理コンソールを使用して JDBC プロバイダーを構成します。
 - 管理コンソールを使用して、JNDI 名が `jdbc/mashupDS` のデータ・ソースを、ご使用の環境に応じてサーバー・スコープまたはクラスター・スコープで作成します。

このタスクについて

管理コンソールを使用して Business Space を構成する代わりにスクリプトを作成する場合は、コマンド行を使用して Business Space を構成できます。

Business Spaceが既に構成されているかどうか不明な場合は、`getBusinessSpaceDeployStatus` コマンドを実行することで、サーバー、クラスター、またはセルでBusiness Spaceが構成されているかどうかを確認できます。このコマンドについて詳しくは、『`getBusinessSpaceDeployStatus` コマンド』を参照してください。

手順

1. コマンド・ウィンドウを開きます。

`wsadmin` コマンドは、スタンドアロン・サーバー環境の場合は `profile_root/bin` ディレクトリにあり、Network Deployment 環境の場合は `deployment_manager_profile_root/bin` ディレクトリにあります。

2. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を開始します。
3. `installBusinessSpace` コマンドを使用して、Business Spaceのエンタープライズ・アーカイブ (EAR) ファイルをランタイム環境にインストールします。
4. `configureBusinessSpace` コマンドを使用してBusiness Space用のデータ・ソースを構成し、データベース表を構成するスクリプトを `profile_root/dbscripts/BusinessSpace/node_name_server_name/database_type/database_name` (スタンドアロン・サーバーの場合) または `profile_root/dbscripts/BusinessSpace/cluster_name/database_type/database_name` (クラスターの場合) にコピーします。データベース表を構成するスクリプトを実行する必要があります。スクリプトについて詳しくは、『Business Space のデータベース表の構成』を参照してください。

データベース構成にデータベース設計ファイルを使用する場合は、`configureBusinessSpace` コマンドを実行する際に、**`-bspacedbDesign`** パラメーターを使用してそのファイルを指定できます。

5. それぞれのコマンドの実行後に、`AdminConfig.save()` (Jython) または `$AdminConfig save (Jacl)` を実行します。
6. デプロイメント環境またはクラスターを始動する前に、Business Space のデータベース表を構成するスクリプトを実行します。詳しくは、『Business Space のデータベース表の構成』を参照してください。

タスクの結果

Business Spaceを構成すると、セットアップされたプロファイルで稼働するアプリケーションのビジネス・ユーザー向けに、ブラウザー・ベースのグラフィカル・ユーザー・インターフェースがセットアップされます。Business Spaceでは、管理者およびアプリケーション・ユーザーは、WebSphere Business Process Management ポートフォリオの製品からコンテンツをカスタマイズできます。

例

以下の例では、Jython を使用して `installBusinessSpace` コマンドおよび `configureBusinessSpace` コマンドを実行することで、EAR ファイルをクラスターにインストールし、クラスターで Business Space のデータ・ソースを構成します。この例では、複数の製品がインストールされている場合に Business Space で使用するスキーマおよび製品データベースを指定します。WebSphere Process Server と WebSphere Business Monitor の両方がインストールされている場合、この例では WebSphere Process Server データ・ソースのプロパティを使用して Business Space データ・ソースを作成します。

```
AdminTask.installBusinessSpace(['-clusterName myCluster -save true'])
```

```
AdminTask.configureBusinessSpace(['-clusterName myCluster -schemaName mySchema -productTypeForDatasource WPS -save true'])
```

Jacl を使用した例を以下に示します。

```
$AdminTask installBusinessSpace {-clusterName myCluster -save true}
```

```
$AdminTask configureBusinessSpace {-clusterName myCluster -schemaName mySchema -productTypeForDatasource WPS -save true}
```

次のタスク

注: Oracle を使用している場合、Business Space データ・ソースの認証別名のパスワードは、Business Space のスキーマ名と同じに設定されます。スキーマのデフォルト値は `IBMBUSSP` です。Business Space の構成時に、管理コンソールまたはコマンド行で別のスキーマを指定できます。その場合、デフォルトのパスワードは、指定するスキーマと同じです。Business Space ユーザー名に別のパスワードを使用する場合は、管理コンソールを使用して、JDBC リソースを更新する必要があります。データ・ソース `jdbc/mashupsDS` を検索します。Business Space スキーマ名のパスワードと同じようになるように、認証別名の値を変更します。変更内容を保存して、サーバーを再始動します。

ランタイム環境で Business Space を使用可能にするには、コマンド行で Business Space を構成した後、次の手順を実行する必要があります。

- `registerRESTserviceEndpoint` コマンドを使用してエンドポイントを登録します。
- チームが使用する Business Space とウィジェットで使用する必要のあるセキュリティーをセットアップします。詳しくは、『Business Space のセキュリティーのセットアップ』を参照してください。

注: Business Space は、プロキシ・コンポーネントを使用して REST サービスに接続します。REST サービスの反応がない場合、REST サービス・サーバーのパフォーマンスに基づいて、Business Space から REST サービスへの接続タイムアウト設定を更新する必要があります。REST サービス接続がタイムアウトになる場合は、以下の設定を更新します。デフォルトで、`socket-timeout` 値は 30 秒に設定されています。この値を状況に適した値に変更します。

1. ファイル `profile_root/BusinessSpace/node_name/server_name/mm.runtime.prof/config/proxy-config.xml` を開きます。

2. socket-timeout の proxy:value を変更します。時間はミリ秒で指定します。

```
<proxy:meta-data>
  <proxy:name>socket-timeout</proxy:name>
  <proxy:value>30000</proxy:value>
</proxy:meta-data>
```

3. wsadmin スクリプト・クライアントを使用して、updateBlobConfig コマンドを実行します。その際には、**-serverName** および **-nodeName** パラメーター (スタンドアロン・サーバーの場合)、または **-clusterName** パラメーター (クラスターの場合) を指定すると共に、**-propertyFileName** パラメーターに proxy-config.xml ファイルのパスを値として指定し、**-prefix** パラメーターに値 Mashups_ を指定します。
4. 管理コンソールから mm_was_node_server アプリケーションを再始動するか、サーバー・アプリケーション全体を再始動します。

Business Spaceのデータベース表の構成

WebSphere が提供する Business Space 用のデータベース表は、インストール・プログラムによって生成されるスクリプトを使用して、リモート・データベース・サーバーに手でインストールすることができます。デプロイメント環境を使用する場合、またはデータベースがリモートの場合は、Business Spaceの構成後にこれらのテーブルをインストールする必要があります。

始める前に

このタスクを完了する前に、以下のタスクを完了しておく必要があります。

- 製品のインストール。
- プロファイルの作成および Business Space 用のサーバーまたはクラスターの構成。
- Oracle: データベースの作成。
- Microsoft SQL Server: SQL Server インスタンス認証の設定。SQL Server JDBC ドライバーは、混合認証モードのみをサポートします。このため、SQL Server インスタンスの作成時には、認証を「**SQL Server および Windows (SQL Server and Windows)**」に設定する必要があります。
- Business Space を使用するアプリケーション・サーバーが停止していることの確認。

Monitor **Process Server / ESB** DB2 for z/OS を使用しており、必要なリソースがまだコア製品インストールの一部としてセットアップされていない場合は、このタスクの開始前に以下の追加操作を実行してください。

- 一時データベースおよび一時テーブル・スペースを作成し、スクロール可能カーソルを処理するための宣言済み一時テーブルを格納する。
- 専用 STOGROUP を作成し、Business Space データを格納する。

Monitor **Process Server / ESB** DB2 for z/OS で、別のストレージ・グループを使用する場合 (例えば、共通データベースと同じデータベースおよびストレージ・グループに Business Space データベース表を追加したくない場合)

は、createStorageGroup.sql スクリプトを編集し、そのスクリプトを Business Space の構成後、および Business Space データベース表の構成前に実行する必要があります。

- createStorageGroup.sql ファイルを編集する。このファイルは `profile_root/dbscripts/BusinessSpace/node_name_server_name/database_type/database_name` (スタンドアロン・サーバーの場合) または `profile_root/dbscripts/BusinessSpace/cluster_name/database_type/database_name` (クラスターの場合) にあります。ここで、`database_type` は DB2zOSV8 または DB2zOSV9 です。
- VCAT の値を、@VCAT@ から、使用するストレージ・グループの統合カタログ機能のカタログの名前または別名に変更する。

DB2 V9.x を使用していて、パフォーマンスを改善したい場合は、createTableSpace.sql ファイルを編集します。createTableSpace.sql ファイルは、`profile_root/dbscripts/BusinessSpace/node_name_server_name/database_type/database_name` (スタンドアロン・サーバーの場合) または `profile_root/dbscripts/BusinessSpace/cluster_name/database_type/database_name` (クラスターの場合) にあります。

- IMMEDIATE SIZE 8000 PAGESIZE 32K を IMMEDIATE SIZE 8000 AUTOMATIC PAGESIZE 32K に変更します。
- CREATE SYSTEM TEMPORARY TABLESPACE @TSDIR@TMTP と CREATE REGULAR TABLESPACE @TSDIR@REGTP の両方の下にある EXTENTSIZE 16 の後ろに、PREFETCHSIZE AUTOMATIC という行を追加します。

このタスクについて

configBusinessSpaceDB スクリプトは、特定のデータベースを使用して Business Space 用のテーブルをセットアップします。特定のデータベースではなく既存のデータベース上にテーブルを作成する場合は、使用している製品で createDBTables スクリプトを使用します。

手順

1. テーブルを作成できる十分な権限を持つユーザー ID を使用していることを確認します。
2. 最後に構成したプロファイル内のスクリプトを探し、データベースと同じシステム上の任意のロケーションに保存します。
 - DB2 for z/OS を除くすべてのデータベースの場合、configBusinessSpaceDB.bat スクリプトまたは configBusinessSpaceDB.sh スクリプトを探します。
 - **Process Server / ESB** WebSphere Process Server for z/OS および WebSphere Enterprise Service Bus for z/OS で、他のすべてのデータベース・オブジェクトと共に Business Space データベース表を構成する場合は、createDB.sh スクリプトを探します。
 - **Monitor** **Process Server / ESB** DB2 for z/OS で、createDB.sh スクリプトを実行しない場合は、Business Space のファイルを個別に実行する

必要があります。createStorageGroup.sql、createDatabase.sql、createTablespace.sql、createTables_BusinessSpace.sql、およびcreateTable.sql を探します。

デフォルトでは、このスクリプトは、スタンドアロン・サーバーの場合は `profile_root/dbscripts/BusinessSpace/node_name_server_name/database_type/database_name`、クラスターの場合は `profile_root/dbscripts/BusinessSpace/cluster_name/database_type/database_name` に存在します。更新されたスクリプト（プロファイル作成時に入力した情報が含まれている）は、最後に構成したサーバーまたはクラスターのプロファイル内に存在します。「デプロイメント環境構成」ウィザードを使用した場合、スクリプトはデプロイメント・マネージャー・プロファイルに格納されます。リモート・データベースを構成する場合は、製品がインストールされているシステムからリモート・システム上にスクリプトをコピーしてください。

3. **Process Server / ESB** WebSphere Process Server for z/OS および WebSphere Enterprise Service Bus for z/OS の場合: DB2 for z/OS を構成する場合は、createDB.sh スクリプトを使用して、Business Space データベース表を、1 つのデータベース内のその他のすべてのデータベース・オブジェクトで構成することができます。詳しくは、WebSphere Process Server for z/OS の資料の『createDB.sh スクリプトによる DB2 データベース・オブジェクトの作成』を参照してください。
4. コマンド・プロンプトを開き、使用するプラットフォームに基づいて以下のいずれかのコマンドを実行します。

Derby の場合は、デフォルトの場所（スタンドアロン・サーバーの場合、`profile_root/dbscripts/BusinessSpace/node_name_server_name/database_type/database_name`）でコマンドを実行してください。

その他のデータベース・タイプの場合は、バッチ・ファイルおよびスクリプトが含まれているフォルダーを、データベースと同じロケーションにコピーし、そこでコマンドを実行します。ユーザー ID は、該当するデータベース・タイプ用のコマンド行インタープリターに対するアクセス権限およびコマンドを実行する権限を持っている必要があります。

• **Linux、UNIX、および z/OS プラットフォームの場合:**

`configBusinessSpaceDB.sh`

Derby、DB2、SQL Server で、既存のデータベースを使用せずに別のデータベースを作成する場合は、オプションの **-createDB** パラメーターを使用します。

注: SQL Server を使用する場合は、データベース・スクリプトの実行後、systemout.log ファイルに「... 警告! キーの最大長は 900 バイトです (Warning! The maximum key length is 900 bytes) ...」という警告ステートメントが記録されます。フェデレーテッド・リポジトリをユーザー・レジストリーとして使用している場合は、この警告を無視できます。スタンドアロン LDAP レジストリーを使用している場合は、組織内のすべてのユーザー識別名 (DN) 項目が、131 文字の制限より少ないことを確認してください。いずれかのユーザー DN 項目が 131 文字以上である場合は、ユーザー・アカウント・レジストリーをフェデレーテッド・リポジトリ・オプションに変更する必要があります。

z/OS の場合、以下のファイルを順番に実行します。

- createStorageGroup.sql
- createDatabase.sql
- createTablespace.sql
- createTables_BusinessSpace.sql
- createTable.sql

次のタスク

- Business Spaceで使用できるようにしたいウィジェットのエンドポイントを更新します。
- Business Spaceと、チームが使用しているウィジェット用に、セキュリティーをセットアップします。

コマンド行を使用した Business Space ウィジェットの REST サービス・エンドポイントの登録

管理コンソールを使用してBusiness Spaceを構成する場合、Representational State Transfer (REST) エンドポイントを登録して、チームがBusiness Space内でウィジェットを使用できるようにする必要があります。「ビジネス・スペースの構成」ページおよび「システム REST サービス・エンドポイント登録 (System REST service endpoint registration)」ページを使用して管理コンソール上でエンドポイントを登録しない場合、registerRESTServiceEndpoint コマンドを使用して登録できます。

始める前に

このタスクを完了する前に、以下のタスクを完了しておく必要があります。

- 製品のインストール。
- Business Space内で使用するウィジェットのための REST サービスの構成。これは「REST サービス」管理コンソール・ページまたは updateRESTGatewayService コマンドを使用して行います。スタンドアロン・サーバー環境を使用しているか、または「デプロイメント環境構成」ウィザードを使用してランタイム環境を構成している場合、REST サービスは自動的に構成されて使用可能になります。ビジネス・プロセスおよびヒューマン・タスクのための REST サービスの構成は、Business Process Choreographer および Human Task Manager コンテナを構成することによって行います。
- 「ビジネス・スペースの構成」管理コンソール・ページを使用するか、または installBusinessSpace コマンドおよび configureBusinessSpace コマンドを使用することによるBusiness Spaceの構成。
- データベース表の構成 (リモート・データベースまたはネットワーク・デプロイメント環境を使用する場合)。

このタスクについて

スタンドアロン・サーバー環境で管理コンソールまたはプロファイル管理ツールを使用して Business Space を構成している場合、または「デプロイメント環境構成」ウィザードを使用してランタイム環境を構成している場合、REST サービスは自動的に登録されます。これ以外の場合は、REST サービスを構成して登録する必要があります。

「システム REST サービス・エンドポイント登録 (System REST service endpoint registration)」管理コンソール・ページまたは `registerRESTServiceEndpoint` コマンドによって、Business Space内の製品のすべてのウィジェットに対して REST サービスのエンドポイントを登録することができます。すると、Business Spaceが自動的にウィジェットをこれらのエンドポイントに関連付け、ウィジェットがBusiness Space・パレットに表示されて使用できるようになります。

`registerRESTServiceEndpoint` コマンドでは、指定のプロバイダー、デプロイメント・ターゲット、またはセル内のすべての固有のエンドポイントについて、エンドポイントのセットを登録できます。このコマンドは、Business Spaceと同じセル内にある REST サービスのエンドポイントを登録します。

手順

1. コマンド・ウィンドウを開きます。

`wsadmin` コマンドは、スタンドアロン・サーバー環境の場合は `profile_root/bin` ディレクトリにあり、ネットワーク・デプロイメント環境の場合は `deployment_manager_profile_root/bin` ディレクトリにあります。

2. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を開始します。
3. `registerRESTServiceEndpoint` コマンドを使用して、ご使用の製品のすべてのウィジェットに対して REST サービスのBusiness Space・エンドポイントを登録します。
4. 各コマンドの実行後に、`save` コマンドを実行します。

例

以下の例では、`Jython` を使用して `registerRESTServiceEndpoint` コマンドを実行した後、変更内容を保存します。ここでは、クラスター上で構成済みで使用可能なすべての REST サービスをBusiness Spaceに登録します。

```
AdminTask.registerRESTServiceEndpoint('[-clusterName
name_of_rest_services_cluster -businessSpaceClusterName
name_of_business_space_cluster]')
AdminConfig.save()
```

ここで、`name_of_rest_services_cluster` は REST サービスが構成されたクラスター名で、`name_of_business_space_cluster` は Business Space がデプロイされたクラスター名です。

次の例では `Jacl` を使用しています。

```
$AdminTask registerRESTServiceEndpoint
{-clusterName name_of_rest_services_cluster
-businessSpaceClusterName name_of_business_space_cluster}
$AdminConfig save
```

ここで、`name_of_rest_services_cluster` は REST サービスが構成されたクラスター名で、`name_of_business_space_cluster` は Business Space がデプロイされたクラスター名です。

`appName`、`webModuleName`、`type`、`version`、`nodeName`、`serverName`、および `clusterName` の各パラメーターはオプションです。

type、**appName**、および **webModuleName** の各パラメーターを指定しない場合、デプロイメント・ターゲットに構成されたすべての固有の REST サービス・エンドポイントが登録されます。

これらのどのパラメーターも指定しない場合、いずれかのデプロイメント・ターゲットに構成されているすべての固有の REST サービス・エンドポイントが登録されます。

次のタスク

Business Space は、プロキシ・コンポーネントを使用して REST サービスに接続します。REST サービスの反応がない場合、REST サービス・サーバーのパフォーマンスに基づいて、Business Space から REST サービスへの接続タイムアウト設定を更新する必要があります。REST サービス接続がタイムアウトになる場合は、以下の設定を更新します。デフォルトで、`socket-timeout` 値は 30 秒に設定されています。この値を状況に適した値に変更します。

1. ファイル `profile_root/BusinessSpace/node_name/server_name/mm.runtime.prof/config/proxy-config.xml` を開きます。
2. `socket-timeout` の `proxy:value` を変更します。時間はミリ秒で指定します。

```
<proxy:meta-data>
  <proxy:name>socket-timeout</proxy:name>
  <proxy:value>30000</proxy:value>
</proxy:meta-data>
```
3. `wsadmin` スクリプト・クライアントを使用して、`updateBlobConfig` コマンドを実行します。その際には、`-serverName` および `-nodeName` パラメーター (スタンドアロン・サーバーの場合)、または `-clusterName` パラメーター (クラスターの場合) を指定すると共に、`-propertyFileName` パラメーターに `proxy-config.xml` ファイルのパスを値として指定し、`-prefix` パラメーターに値 `Mashups_` を指定します。
4. 管理コンソールから `mm_was_node_server` アプリケーションを再始動するか、サーバー・アプリケーション全体を再始動します。

Business Space と連携するためのプロキシ・サーバーまたはロード・ balancer の構成

プロキシ・サーバーまたはロード・ balancer ・サーバーが存在する環境で Business Space を使用している場合は、Business Space およびウィジェットが正しく機能するように、ご使用の環境をセットアップする必要があります。

このタスクについて

Network Deployment 環境またはクラスター環境では、セキュリティ上の理由から、またはワークロード・ balancer を目的として、プロキシ・サーバーまたは HTTP サーバーをセットアップする場合があります。着信 HTTP 要求は、アプリケーション・サーバーに直接送信されるのではなく、プロキシ・サーバーに送信されます。このプロキシ・サーバーによって、処理を実行する複数のアプリケーション・サーバーの間で要求を分散することができます。

プロキシ・サーバーの代わりに (またはプロキシ・サーバーの前に)、IBM HTTP Server などのルーティング・サーバーを使用することができます。

重要: 複数のクラスター・メンバー間で HTTP 要求のワークロード・バランシングを行うには、プロキシ・サーバー (または代替ルーティング・サーバー) が必要です。プロキシ・サーバーによって、クライアントはこのトポロジー内のアプリケーションにアクセスできるようになります。

ブラウザと、Business Space サービスおよび REST サービスの間に、ロード・バランサーまたはプロキシ・サーバーがある環境では、REST サービス・プロトコル、ホスト、およびポートの指定内容が、Business Space にアクセスするためのブラウザ URL と一致していることを確認してください。管理コンソールの REST サービス・プロバイダー・ページで、すべてのプロバイダー (Business Flow Manager や Human Task Manager など) のプロトコル、ホスト、およびポートが正しいことを確認してください。REST サービスの変更については、『サービス・プロバイダーでの REST サービスの構成』を参照してください。

IBM HTTP Server を使用する場合は、追加のマッピング手順を実行して、モジュールが Web サーバーにマッピングされていること、およびホスト別名が構成されていることを確認する必要があります。

HTTP サーバーにリバース・プロキシ・セットアップを使用する場合は、Business Space およびウィジェットの URL をマップする必要があります。

IBM HTTP Server の Business Space 用の構成:

IBM HTTP Server を使用する場合は、ご使用の環境で Business Space が機能するように、追加のマッピング・ステップを完了する必要があります。

始める前に

Business Space と連携するように IBM HTTP Server を構成する前に、以下の手順を完了してください。

- IBM HTTP Server をインストールします。
- Secure Sockets Layer (SSL) が IBM HTTP Server で有効になっていることを確認します。
- IBM HTTP Server の Web サーバー定義が、アプリケーション・サーバーに追加されていることを確認します。

IBM HTTP Server プラグインのインストール中に、Web サーバー・マシンのインストール・プロセスによって、`configureWeb_server` スクリプトが生成されます。`configureWeb_server` スクリプトは、Web アプリケーション・モジュールを Web サーバーにマップするためのものです。したがって、このスクリプトは、デプロイメント環境の生成後に実行してください。

手順

1. モジュールが Web サーバーにマップされていることを確認します。Business Space で必要となるアプリケーションごとに、選択されたターゲットの中に Web サーバーが含まれていることを確認します。
 - a. 管理ユーザーとして、管理コンソールにログインします。
 - b. 「アプリケーション」 → 「アプリケーション・タイプ」 → 「WebSphere エンタープライズ・アプリケーション」をクリックします。

- c. 「エンタープライズ・アプリケーション」パネルで、アプリケーションの名前をクリックします。

以下のアプリケーションを確認します。 Business Space で使用する製品によっては、このリスト内のアプリケーションの一部またはすべてが揃っている可能性があります。

- **BPMAAdministrationWidgets_nodename_servername** (WebSphere Enterprise Service Bus および WebSphere Process Server の場合)
 - **BusinessSpaceHelpEAR_nodename_servername** (すべての製品の場合)
 - **BSpaceEAR_nodename_servername**(すべての製品の場合)
 - **BSpaceWebformsEnabler_nodename_servername** (すべての製品の場合)
 - **HumanTaskManagementWidgets_nodename_servername** (WebSphere Process Server および WebSphere Business Monitor の場合)
 - **REST サービス・ゲートウェイ** (すべての製品の場合)
 - **REST サービス・ゲートウェイ Dmgr** (WebSphere Enterprise Service Bus および WebSphere Process Server の場合)
 - **mm.was_nodename_servername** (すべての製品の場合)
 - **WBMDashboardWeb_nodename_servername** (WebSphere Business Monitor の場合)
 - **wesbWidgets_nodename_servername** (WebSphere Enterprise Service Bus の場合)
 - **widgets_busleader_nodename_servername** (WebSphere Business Compass の場合)
 - **widgets_pubserver_nodename_servername** (WebSphere Business Compass の場合)
 - **widgets_fabric_nodename_servername** (WebSphere Business Services Fabric の場合)
- d. アプリケーションごとに、「構成」タブの「モジュール」の下で「モジュールの管理」をクリックします。
- e. アプリケーションの「モジュールの管理」ページで、選択されたターゲットの中に Web サーバーが含まれていることをモジュールごとに確認します。
- 表の各モジュールの「サーバー」列を調べて、選択されたターゲットの中に Web サーバーが含まれていることをモジュールごとに確認します。例えば、mm.was_nodename_servername アプリケーションについては、「サーバー」列に表示される Web サーバー
WebSphere:cell=qaxs41Cell02,node=qaxs41Node03,server=httpserver
WebSphere:cell=qaxs41Cell02,cluster=Golden.WebApp を探します。
 - Web サーバーを追加する必要がある場合は、モジュールの名前の横にあるチェック・ボックスを選択します。次に、「クラスターおよびサーバー」リストで、Ctrl キーを使用して複数のターゲットを選択します。例えば、Web サーバーがアプリケーションにサービスを提供するには、Ctrl キーを押しながら、アプリケーション・サーバー・クラスターと Web サーバーを一緒に選択します。変更を保存するには、「適用」、「OK」および「保存」をクリックします。

2. ホスト名の別名 `default_host` に、すべてのクラスター・メンバー、Web サーバー、またはプロキシ・サーバーの正しい情報が含まれていることを確認します。
 - a. 管理ユーザーとして、管理コンソールにログインします。
 - b. 「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」をクリックします。
 - c. すべてのクラスター・メンバーについて、アプリケーション・サーバーの名前をクリックして、`WC_defaulthost` ポート名のポート番号を表示します。
 - 「通信」の下で、「ポート」を展開します。
 - ポート名 `WC_defaulthost` のポート番号を確認します。
 - d. 管理コンソールの左側のナビゲーション領域で、「環境」 → 「仮想ホスト」をクリックします。
 - e. 「`default_host`」の名前をクリックします。
 - f. 「追加プロパティ」で、「ホスト別名」をクリックします。
 - g. クラスター・メンバーのホスト名およびポート番号がリストに表示されていない場合は、「新規」をクリックして、欠落している項目をリストに追加します。ホスト名では、ワイルドカード文字 * (アスタリスク) がサポートされます。
 - h. 新規項目を追加する場合は、「保存」および「同期化」をクリックします。

リバース・プロキシ・サーバーのためのBusiness Space URL のマッピング:

HTTP サーバーにリバース・プロキシがセットアップされている場合、Business Spaceと連携するように HTTP サーバーを構成するときに、Business Spaceとチームで使用するウィジェットの URL をマップする必要があります。

手順

1. HTTP サーバーの構成ファイルを編集します。
2. Business Spaceと、ランタイム・ソリューションでビジネス・ユーザーが使用するウィジェットのすべての URL をマップします。

一般的なBusiness Space・フレームワーク (全製品) の URL:

- `/BusinessSpace/*`
- `/mum/*`
- `/help/*`
- `/BspaceWebformsProxy/*`
- `/themes/*`

WebSphere Business Compass ウィジェットの追加 URL:

- `/WBPublishingDRAFT/*`
- `/BusinessLeader/*`
- `/BusinessLeaderWidgets/*`

WebSphere Business Services Fabric ウィジェットの追加 URL:

- `/bpm/bslm/v1/*`

- /bpm/glossary/v1/*
- /bpm/governance/v1/*
- /bpm/bvars/v1/var/*
- /rest/*

WebSphere Business Monitor ウィジェットの追加 URL:

- /BusinessDashboard/*
- /DashboardABX/*
- /monitorServerComponent/*
- /mobile/*
- /rest/*
- /AlphabloxServer/*
- /AlphabloxAdmin/*
- /AlphabloxTooling/*
- /BloxBuilder/*

WebSphere Enterprise Service Bus ウィジェットの追加 URL:

- /BspaceWidgetsHM/*
- /rest/*
- /PolymorphicWidget/*
- /scaWidget/*
- /ServiceMonitorGraphWidget/*
- /StoreAndForward/*

WebSphere Process Server ウィジェットの追加 URL:

- /BspaceWidgetsHM/*
- /SecurityManagerWidgets/*
- /BspaceWidgetsBCM/*
- /rest/*
- /PolymorphicWidget/*
- /scaWidget/*
- /ServiceMonitorGraphWidget/*
- /StoreAndForward/*

クロスセル環境に対する Business Space ウィジェットの有効化

Business Spaceが、Representational State Transfer (REST) サービスが稼働しているセルと異なるセルで稼働している場合、またはウィジェットがBusiness Spaceと異なるセル上にある場合、エンドポイント・ファイルを手動で編集する必要があります。

始める前に

このタスクを完了する前に、以下のタスクを完了しておく必要があります。

- 製品のインストール。

- デプロイメント・ターゲット (サーバーまたはクラスター) 上の作成済みプロフィール、および構成済みの、 Business Space。
- データベース表の構成 (リモート・データベースまたはデプロイメント環境を使用する場合)。
- WebSphere Business Compass の場合は、最初に WebSphere Business Compass データベースの Endpoints 表を更新します。Server_Name 列にはBusiness Spaceのインターネット・プロトコルを設定し、Port 列にはBusiness Spaceのポートを設定します。

このタスクについて

ご使用の製品で必要となるすべてのウィジェットは Business Space と共にインストールされますが、チームがBusiness Spaceでウィジェットを使用するには、ウィジェットで必要となるエンドポイントを構成および登録する必要があります。管理コンソール・ページを使用して、エンドポイントを構成および登録できます。しかし、ご使用の製品および REST サービスが Business Space とは異なるセルにインストールされている場合は、REST サービス・エンドポイント・ファイルを編集して、Business Space で REST サービスにアクセスし、ウィジェットが適切に動作するようにする必要があります。

インストール済みの製品、およびBusiness Spaceで使用しているウィジェットに基づいて、以下のエンドポイント・ファイルの 1 つ以上を編集します。

- WebSphere Business Compass: pubserverEndpoints.xml および busLeaderEndpoints.xml
- WebSphere Business Monitor: monitorEndpoints.xml
- WebSphere Business Monitor with Alphablox: monitorABXEndpoints.xml
- WebSphere Business Services Fabric: fabricEndpoints.xml
- WebSphere Enterprise Service Bus: wesbWidgetEndpoints.xml (メディエーション・ポリシーの管理ウィジェット、サービス・ブラウザー・ウィジェット、およびプロキシ・ゲートウェイ・ウィジェットの場合同)、 bpmAdministrationEndpoints.xml (管理ウィジェットの場合同)
- WebSphere Process Server: wpsEndpoints.xml、 bpmAdministrationEndpoints.xml (管理ウィジェットの場合同)、 wesbWidgetEndpoints.xml (メディエーション・ポリシーの管理ウィジェット、サービス・ブラウザー・ウィジェット、およびプロキシ・ゲートウェイ・ウィジェットの場合同)、 HumanTaskManagementEndpoints.xml (ビジネス・プロセスおよびヒューマン・タスクの場合同)、 bspaceWFSEndpoints.xml (Lotus Webform Server をヒューマン・タスク管理ウィジェットとともに使用する場合同)
- すべての製品: wsumEndpoint.xml (ユーザー・メンバーシップの場合同)

管理者は、以下のステップを実行して、エンドポイントを登録し、ウィジェットを使用可能にすることができます。

手順

1. ウィジェットがインストールされたセルから、製品インストール時にBusiness Spaceが構成されたセルへ、ウィジェットをコピーします。ウィジェットは、`install_root¥BusinessSpace¥widgets` ディレクトリーにあり、一時フォルダーにコピーできます。
2. `installBusinessSpaceWidgets` コマンドを実行して、`install_root¥BusinessSpace¥widgets` ディレクトリーにある、指定されたウィジェットをインストール、デプロイ、および登録します。
 - a. ターゲット・サーバー (スタンドアロン・サーバー環境の場合) またはデプロイメント・マネージャー (Network Deployment 環境の場合) が稼働していることを確認し、そのプロファイルでコマンド・ウィンドウを開きます。

`wsadmin` コマンドは、`profiles¥profile_name¥bin` ディレクトリーに格納されています。

- b. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を開始します。
 - c. `installBusinessSpaceWidgets` コマンドを実行します。クラスター環境の場合、**-clusterName** パラメーターを指定します。スタンドアロン・サーバー環境の場合、**-serverName** および **-nodeName** パラメーターを指定します。**-widgets** パラメーターを、ウィジェットを含むディレクトリーまたはファイルの絶対パスと共に指定します。
3. `install_root¥BusinessSpace¥registryData¥endpoints` ディレクトリーで、エンドポイント・ファイルを見つけます。クラスターの場合、デプロイメント・マネージャー・プロファイル・ルートを使用します。ファイル名はすべて、`Endpoints.xml` または `Endpoint.xml` で終わっています。
 4. 構成するエンドポイント・ファイルごとに、バックアップ・コピーを作成します。
 5. 最初のセルのデプロイメント・マネージャー・プロファイル上のディレクトリー `profile_root¥BusinessSpace¥registryData¥` (`profile_root` は通常は `install_root¥profiles¥profile_name` または `install_root¥pf¥profile_name`) が存在しない場合はこのディレクトリーを作成し、エンドポイント登録ファイルをこのディレクトリーにコピーします。

6. 必要に応じて、エンドポイント・ファイルを編集してエンドポイントを構成します。エンドポイント・ファイル内の各エンドポイントは、`<tns:Endpoint>` ブロックにより指定されます。変更するブロックを特定します。

ヒント: 一部のエンドポイントをアクティブにしない場合は、混乱を防ぐためにファイルから除去してもかまいません。

エンドポイントで特定されたロケーションは、`<tns:url>` で指定されています。この値は、Web モジュール内のパスであり、絶対または相対 HTTP URL として指定されます。デフォルトにより、URL は相対です。絶対 URL パスに変更します。例えば、`https://virtualhost.com:virtualport/rest/bpm/htm` または `http://host1:9445/WBPublishingDRAFT/` (製品 Web モジュールへのアクセス方法をプロトコル、ホスト、およびポートで指定) とします。

サーバーのポート番号を調べるには、次のステップを実行します。

- 管理コンソールにログインします。
- 「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」をクリックします。
- ポート番号を調べる対象のサーバーをクリックし、「ポート」セクションを展開します。

すべてのアプリケーションは、**wc_defaulthost** (非セキュア・ホスト) パラメータまたは **wc_defaulthost_secure** (セキュア・ホスト) パラメータのいずれかに示されるように、同じポートを使用します。

注: HTTP サーバーを使用して Web モジュールにアクセスし、ロード・バランシングを行う場合は、HTTP サーバーのホスト名とポート設定を使用してください。

7. Business Space サーバーが構成されているセルで、エンドポイント XML ファイルを変更した後に、`updateBusinessSpaceWidgets` コマンドを使用してエンドポイント URL を更新します。
 - a. ご使用のプロファイルについて、コマンド・ウィンドウを開きます。
`wsadmin` コマンドは、`profiles¥profile_name¥bin` ディレクトリーに格納されています。クラスター環境の場合、`deployment_manager_profile_root¥bin` ディレクトリーからコマンドを実行します。スタンドアロン・サーバー環境の場合、`profile_root¥bin` ディレクトリーからコマンドを実行します。
 - b. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を開始します。
 - c. `updateBusinessSpaceWidgets` コマンドを実行します。クラスター環境の場合、**-clusterName** パラメーターを指定します。スタンドアロン・サーバー環境の場合、**-serverName** および **-nodeName** パラメーターを指定します。**-endpoints** パラメーターを、ウィジェット・エンドポイント・ファイルがあるディレクトリーの絶対パスまたは特定のエンドポイント・ファイルへの絶対パスと共に指定します。

例

以下のエンドポイント・ファイルの例は、WebSphere Business Monitor ウィジェットに対応するものです。

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- START NON-TRANSLATABLE -->
<tns:BusinessSpaceRegistry
  xmlns:tns="http://com.ibm.bspace/BusinessSpaceRegistry"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://com.ibm.bspace/BusinessSpaceRegistry
  BusinessSpaceRegistry.xsd ">

  <tns:Endpoint>
    <tns:id>{com.ibm.wbimonitor}monitorServiceRootId</tns:id>
    <tns:type>{com.ibm.wbimonitor}monitorServiceRootId</tns:type>
    <tns:version>1.0.0.0</tns:version>
    <tns:url>/rest/</tns:url>
    <tns:description>Location of backing services for Monitor widgets
```

```
</tns:description>
  </tns:Endpoint>

</tns:BusinessSpaceRegistry>
<!-- END NON-TRANSLATABLE -->
```

次のタスク

- `installBusinessSpaceWidgets` コマンドまたは `updateBusinessSpaceWidgets` コマンドを実行したら、手動のステップを実行して、**Business Space** のテンプレートとスペースを更新する必要があります。詳しくは、『ウィジェットのインストールまたは更新後の **Business Space** テンプレートおよびスペースの更新』を参照してください。
- サービス・エンドポイントの複数インスタンスの場合、例えば、2 つのクラスター上で処理を区画化して、各クラスターからのデータをウィジェットが表示するようにしたいときは、それぞれの追加クラスターに対して追加ウィジェットを手動で使用可能にする必要があります。ウィジェット・エンドポイント・ファイルとウィジェット・カタログ・ファイルの両方を編集する必要があります。詳しくは、『**Business Space** ウィジェットが複数のエンドポイントを処理できるようにする』を参照してください。
- ご使用の環境でセキュリティーを使用可能にしている場合は、**Business Space** を処理できるように正しくセットアップされていることを確認してください。

Business Space ウィジェットが複数のエンドポイントを処理できるようにする

既に **Business Space** ・インスタンスが構成してある状態で環境にサービス・エンドポイントの別のインスタンスを作成する必要がある場合は、ウィジェットが複数のサービス・エンドポイントからのデータを表示できるように **Business Space** を構成する必要があります。エンドポイントを **Business Space** に登録するためのエンドポイント・ファイルと、ウィジェットの定義を含むウィジェット・カタログ・ファイルの 2 つのファイルを編集する必要があります。

始める前に

このタスクを完了する前に、以下のタスクを完了しておく必要があります。

- 製品のインストール。
- **Business Space** 用のサーバーまたはクラスターの作成および構成。
- データベース表の構成 (リモート・データベースまたはデプロイメント環境を使用する場合)。

このタスクについて

デプロイメント環境では、作業区画を持つことができます。例えば、2 つのクラスターを使用して、一方のクラスターには会計データを処理させ、もう一方のクラスターには保険データを処理させることができます。ただし、サービス・エンドポイントは、1 つのクラスターにのみサービスを提供します。 **Business Space** から両方の作業区画にアクセスするには、2 つの別々のウィジェット (作業区画ごとに 1 つのウィジェット) を登録する必要があります。これにより、**Business Space** から両方の作業区画にアクセスすることができます。例えば、会計ヒューマン・タスク・

リスト・ウィジェットと保険タスク・リスト・ウィジェットをカタログ内に持つことができます (実際のヒューマン・タスク・リスト・コードは、両方とも同じです)。

エンドポイント・ファイルおよびウィジェット・カタログ・ファイルを手作業で編集する必要があります。

ウィジェット・エンドポイント・ファイルは各製品にバンドルされており、製品のインストール時に追加されます。インストールした製品、およびBusiness Spaceで使用しているウィジェットに基づいて、以下のエンドポイント・ファイルを 1 つ以上編集する必要があります。

- WebSphere Business Compass: pubserverEndpoints.xml および busLeaderWidgetEndpoints.xml。
- WebSphere Business Monitor: monitorEndpoints.xml
- WebSphere Business Monitor with Alphablox: monitorABXEndpoints.xml
- WebSphere Business Services Fabric: fabricEndpoints.xml
- WebSphere Enterprise Service Bus: wesbWidgetEndpoints.xml (メディエーション・ポリシーの管理ウィジェット、サービス・ブラウザー・ウィジェット、およびプロキシ・ゲートウェイ・ウィジェットの状況)、 bpmAdministrationEndpoints.xml (管理ウィジェットの状況)
- WebSphere Process Server: wpsEndpoints.xml、 bpmAdministrationEndpoints.xml (管理ウィジェットの状況)、 wesbWidgetEndpoints.xml (メディエーション・ポリシーの管理ウィジェット、サービス・ブラウザー・ウィジェット、およびプロキシ・ゲートウェイ・ウィジェットの状況)、 HumanTaskManagementEndpoints.xml (ビジネス・プロセスおよびヒューマン・タスクの状況)、 bspaceWFSEndpoints.xml (Lotus Webform Server をヒューマン・タスク管理ウィジェットとともに使用する状況)
- すべての製品: wsumEndpoint.xml (ユーザー・メンバーシップの状況)

ウィジェット・カタログ・ファイルには、製品のウィジェットの定義が含まれます。インストールした製品、およびBusiness Spaceで使用しているウィジェットに基づいて、以下のウィジェット・ファイルを 1 つ以上編集する必要があります。

- WebSphere Business Compass: catalog_pubserverWidgets.xml および catalog_busLeaderWidgets.xml
- WebSphere Business Monitor: catalog_WBMonitor.xml
- WebSphere Enterprise Service Bus: catalogProxyGateway.xml および catalog_ServiceAdmin.xml
- WebSphere Process Server: catalog_BPMAAdministration.xml、 catalog_BusinessRules.xml、 catalog_ServiceAdmin.xml、 および catalog_HumanTaskManagement.xml
- WebSphere Business Services Fabric: catalog_fabric.xml

エンドポイント・ファイルとウィジェット・カタログ・ファイルは、いずれも `install_root\BusinessSpace\registryData\` にあります。エンドポイント・ファイルは `endpoints` サブディレクトリーにあり、カタログ・ファイルは `catalogs` サブディレクトリーにあります。

ディレクトリー `install_root¥BusinessSpace¥registryData¥` には、製品のエンドポイントおよびウィジェットのカタログ・テンプレート・ファイルが含まれています。テンプレートとして使用する必要のあるファイルをコピーして、変更を追加することができます。

ウィジェットのインスタンスを追加で作成する場合は、以下のステップを実行します。

手順

1. エンドポイントをさらに追加するために、エンドポイント・ファイルを編集します。
 - `install_root¥BusinessSpace¥registryData¥endpoints` ディレクトリーで、エンドポイント・ファイルを見つけます。テンプレート・ファイルを使用する場合は、エンドポイント・テンプレート・ファイルをコピーします。変更しないエンドポイントをすべて除去して、追加のエンドポイントを新しいファイルに追加します。
 - `<tns:Endpoint>` で始まり、固有の ID (`<tns:id>`) および新しいエンドポイントの URL (`<tns:url>`) を持つが、バージョンは同じエンドポイントを追加します。タイプ (`<tns:type>`) は ID (`<tns:id>`) に一致していなければなりません。名前および説明 (「チームの保険タスク・リスト」など) は変更できません。
 - `<tns:id>`: ID には任意のストリングを使用できますが、登録されたすべてのエンドポイントで固有でなければなりません。エンドポイントをさらに追加するときは、この ID が固有であることを確認してください。
 - `<tns:url>`: URL が相対 URL の場合は、REST サービス・エンドポイントが Business Space・サーバーと同じ場所に配置されていることが想定されます。エンドポイントがリモート・システム上にある場合は、このフィールドを絶対 URL で更新します。
 - `<tns:description>`: このエンドポイントで操作するデータ・セットの性質を詳述した意味のある説明を入力します。この説明は、データ・セットを操作するクラスターに基づくものにもできますし、データ・セットの性質に基づくものにもできます (例: 保険請求ヒューマン・タスク、会計データ・ヒューマン・タスク)。
 - 変更を保存します。

以下のエンドポイントの例が `monitorEndpoints.xml` に入っています。

```
<tns:Endpoint>
  <tns:id>{com.ibm.wbimonitor}monitorServiceRootId</tns:id>
  <tns:type>{com.ibm.wbimonitor}monitorServiceRootId</tns:type>
  <tns:version>1.0.0.0</tns:version>
  <tns:url>/rest/</tns:url>
  <tns:description>Location of backing services for Monitor widgets
</tns:description>
</tns:Endpoint>
```

2. ウィジェット・カタログ・ファイルを編集して新しいウィジェット定義を追加します。
 - `install_root¥BusinessSpace¥registryData¥catalogs` ディレクトリーでウィジェット・カタログ・ファイルを探します。

- 固有の ID (id="{com.ibm.bspace.widget}widget_id など) および固有の名前 (unique-name="{com.ibm.bspace.widget}widget_name など) を持つ <entry> を追加します。その他の定義はすべて維持することができます。
- タイトルおよび説明を変更して、新しいエンドポイントの性質を説明する新しいウィジェットを、Business Spaceで別個のウィジェットとして使用できるようにします。例えば、<title> でウィジェットに「マイ・チームの保険タスク・リスト」という名前を付けることができます。ビジネス・ユーザーが適切なウィジェットを容易に選択できるようなタイトルを入力してください。選択したウィジェットの機能とデータの性質をビジネス・ユーザーが容易に理解できるような説明を入力してください。
- カタログ・ファイルの <metadata> で、endpoint:// がエンドポイント・ファイルのタイプおよび ID (<tns:type> および .<tns:id>) に一致していることを確認します。
- カタログ・ファイルの <metadata> で、"refVersion" : がエンドポイント・ファイルのバージョン (<tns:version>) に一致していることを確認します。
- 変更を保存します。

ウィジェット・カタログ・ファイルの例を以下に示します。

```
<entry id="{com.ibm.wbimonitor}instances"
unique-name="{com.ibm.wbimonitor}instances">
  <title>
    <!-- END NON-TRANSLATABLE -->
    <nls-string xml:lang="en">Instances</nls-string>
    <!-- START NON-TRANSLATABLE -->
  </title>
  <description>
    <!-- END NON-TRANSLATABLE -->
    <nls-string xml:lang="en">Instances</nls-string>
    <!-- START NON-TRANSLATABLE -->
  </description>
  <shortDescription>
    <!-- END NON-TRANSLATABLE -->
    <nls-string xml:lang="en">このウィジェットは、
個々のインスタンスまたはユーザー定義のコンテキスト・インスタンス・グループのいずれかで
使用可能なモニター・コンテキストを持つダッシュボードを表示します。</nls-string>
    <!-- START NON-TRANSLATABLE -->
  </shortDescription>
  <definition>endpoint://{com.ibm.wbimonitor}monitorWidgetRootId
/com/ibm/wbimonitor/common/iWidgets/instances_iWidget.xml</definition>
  <content>endpoint://{com.ibm.wbimonitor}monitorWidgetRootId/img/
thumb_instances.gif</content>
  <preview>endpoint://{com.ibm.wbimonitor}monitorWidgetRootId/img/
prev_instances.gif</preview>
  <previewThumbnail>endpoint://{com.ibm.wbimonitor}monitorWidgetRootId/
img/prev_instances.gif</previewThumbnail>
  <help>endpoint://{com.ibm.bspace}bSpaceWidgetHelpRootId/topic/
com.ibm.bspace.help.widg.mon.doc/dash/help_instance_whatIs.html</help>
  <icon>endpoint://{com.ibm.wbimonitor}monitorWidgetRootId/img/
icon_instances.gif</icon>
  <metadata name="com.ibm.mashups.builder.autoWiringEnabled">true
</metadata>
  <metadata name="com.ibm.bspace.version">7.0.0.0</metadata>
  <metadata name="com.ibm.bspace.owner">International Business
Machines Corp.</metadata>
  <metadata name="com.ibm.bspace.serviceEndpointRefs">
[{"name":"serviceUrlRoot", "required":"true",
"refId":"endpoint://{com.ibm.wbimonitor}monitorServiceRootId",
"refVersion":"1.0.0.0"}]</metadata>
</entry>
```

3. updateBusinessSpaceWidgets コマンドを実行します。

次のタスク

- installBusinessSpaceWidgets コマンドまたは updateBusinessSpaceWidgets コマンドを実行したら、手動のステップを実行して、Business Space のテンプレートとス

ペースを更新する必要があります。詳しくは、『ウィジェットのインストールまたは更新後の Business Space テンプレートおよびスペースの更新』を参照してください。

- REST サービスが実行されているのと異なるセルで Business Space が実行されている場合、エンドポイント・ファイルを手動で編集する必要があります。
- ご使用の環境でセキュリティーを使用可能にしている場合は、Business Space を処理できるように正しくセットアップされていることを確認してください。

複数の製品のウィジェットの構成

`installBusinessSpaceWidgets` コマンドを使用することによって、ある BPM 製品用の Business Space ウィジェットを、別の BPM 製品を使用して既に構成されている Business Space に構成または追加できます。

始める前に

このタスクを完了する前に、以下のタスクを完了しておく必要があります。

- BPM 製品をインストールおよび構成するためのすべてのステップが完了して、Business Space が構成されている。
- 追加の BPM 製品をインストールおよび構成するためのすべてのステップが完了している。

このタスクについて

Business Space と連携する複数の BPM 製品をインストールできます。2 番目の製品をインストールした後、両方の製品のウィジェットを構成できます。ただし、最初の製品用のウィジェットを使用して Business Space を構成した後、2 番目の BPM 製品をインストールする場合は、`installBusinessSpaceWidgets` コマンドを使用して、同じ Business Space と連携するように、2 番目の製品のウィジェットを追加して構成する必要があります。

スタンドアロン型の拡張では、ウィジェットは自動的にインストールされます。例えば、WebSphere Process Server スタンドアロン・プロファイルを作成し、Business Space 用のサーバーを構成し、WebSphere Business Monitor をインストールし、構成済みのサーバーを WebSphere Business Monitor に拡張すると、ウィジェットがインストールされます。しかし、Network Deployment 環境の場合、デプロイメント・マネージャーを別の製品に拡張しても、追加のウィジェットがインストールおよび構成されることはありません。

手順

1. デプロイメント・マネージャーのプロファイルが稼働していることを確認し、そのプロファイルでコマンド・ウィンドウを開きます。

`wsadmin` コマンドは `profiles/profile_name/bin` ディレクトリーに格納されています。

2. コマンド・プロンプトで、`wsadmin` コマンドを入力し、`wsadmin` 環境を開始します。
3. `installBusinessSpaceWidgets` コマンドを使用して、`install_root/BusinessSpace/widgets` ディレクトリーにある、指定されたウィジェットをインストール、デプロイ、および登録します。

例

以下の例では、Jython を使用して `installBusinessSpaceWidgets` を実行し、IBM WebSphere Process Server 用に構成済みの Business Space 環境で機能する、IBM WebSphere Business Monitor 用のウィジェットをインストールします。

```
AdminTask.installBusinessSpaceWidgets('[-nodeName node_name  
-serverName server_name -widgets  
install_root%BusinessSpace%widgets%WBM%Widgets_WBMonitor.zip]')
```

Jacl を使用した例を以下に示します。

```
$AdminTask installBusinessSpaceWidgets {-nodeName node_name  
-serverName server_name -widgets  
install_root%BusinessSpace%widgets%WBM%Widgets_WBMonitor.zip}
```

次のタスク

ランタイム環境で Business Space を使用可能にするには、ウィジェットを構成した後、に次の手順を実行する必要があります。

- `installBusinessSpaceWidgets` コマンドまたは `updateBusinessSpaceWidgets` コマンドの実行後に、Business Space のテンプレートおよびスペースを更新するための手動のステップを実行してください。詳しくは、『ウィジェットのインストールまたは更新後の Business Space テンプレートおよびスペースの更新』を参照してください。
- REST サービスを構成します。詳しくは、『REST サービスの構成』を参照してください。
- REST エンドポイントを登録します。詳しくは、『管理コンソールでの Business Space の構成および REST エンドポイントの登録』を参照してください。
- セキュリティーが、チームが使用する Business Space およびウィジェットに対して機能するように正しくセットアップされていることを確認します。詳しくは、『Business Space のセキュリティーのセットアップ』を参照してください。

Business Space 内で使用するための特定のウィジェットのセットアップ

製品に付属するウィジェットの中には、Business Space で使用する前に、追加の構成ステップが必要なものがあります。

このタスクについて

Business Process Management 製品にはいくつかのウィジェットが付属しており、これらの中には Business Space のソリューションと通信するために追加の構成が必要なものがあります。

サービス・モニターの構成

新しいサーバーを作成し、(Business Space 内で使用可能な) サービス・モニター・ウィジェットを使用して、SCA モジュールよって公開または呼び出されるサービスの応答時間および要求スループットを測定する場合、管理コンソールでサービス・モニターを構成して有効化します。

始める前に

このタスクに必要なセキュリティ・ロール: 管理セキュリティが使用可能になっている場合、このタスクを実行するには管理ロールでログインする必要があります。

このタスクについて

サービス・モニターではクライアント/サーバー・アーキテクチャーが使用されています。

- サービス・モニター・エージェント: 操作のスループットおよび応答時間を測定し、その測定値データをサービス・モニター・サーバーに送信します。
- サービス・モニター・サーバー: 稼働中のすべてのサービス・モニター・エージェントから応答時間およびスループットの測定値を収集および集約し、統計を算出して保管します。

デプロイメント環境では、サーバーがサポート・クラスターで稼働するのに対し、エージェントはモジュールがデプロイされたサーバー上のアプリケーション・クラスターで稼働します。スタンドアロン・サーバー環境では、サーバーとエージェントの両方がスタンドアロン・サーバーで稼働します。

重要: Business Space にアクセスするために外部 HTTP サーバーを使用している場合は、エンコードされたスラッシュを使用できるように HTTP サーバーを構成してください。詳しくは、HTTP サーバーの資料を参照してください。

手順

1. 管理コンソールに管理者特権でログインします。
2. サービス・モニター・サーバーを構成します。
 - a. コンソール内で、「サーバー」 → 「サーバー・タイプ (Server Types)」 → 「WebSphere アプリケーション・サーバー (WebSphere application servers)」 → 「servername」 → 「サービス・モニター」をクリックします。
 - b. 「サービス・モニター」ページで、「サービス・モニターの使用可能化」をクリックします。
 - c. サービス・モニターのバッファー・サイズおよび照会サイズの制限のデフォルト値を検査し、必要に応じて変更します。
 - d. サービス・モニターのターゲットを指定します。これらはデータ収集の対象となるサービス・モニター・エージェントです。

表9.

モニターのターゲット	実行の手順
稼働中のすべてのサービス・モニター・エージェントをモニターする	「使用可能なすべてのサービス・モニター・エージェント」オプションにチェック・マークを付けます。

表 9. (続き)

モニターのターゲット	実行の手順
稼働中のサービス・モニター・エージェントの特定のサブセットをモニターする	<ol style="list-style-type: none"> 1. 「使用可能なすべてのサービス・モニター・エージェント」オプションをクリアします。コレクション・テーブルが表示されます。これが新しい構成の場合、テーブルは空です。 2. 「追加」をクリックします。「デプロイメント・ターゲットのブラウズ」ページが開きます。 3. 「デプロイメント・ターゲットのブラウズ」ページのコレクション・テーブルから、モニターするエージェントを持つデプロイメント・ターゲットを選択します。 4. 「OK」をクリックして「サービス・モニター・サーバー」ページに戻ります。 5. モニターするすべてのエージェントが追加されるまで、手順 2 から手順 4 を繰り返します。

- e. 「サービス・モニター・サーバー」ページで、「OK」をクリックします。構成が保存され、すぐに有効になります。
3. サービス・モニター・エージェントを構成します。
 - a. コンソール内で、「サーバー」 → 「サーバー・タイプ (Server Types)」 → 「WebSphere アプリケーション・サーバー (WebSphere application servers)」 → 「servername」 → 「サービス・モニター・エージェント」をクリックします。
 - b. 「サービス・モニター・エージェント」ページで、「サービス・モニター・エージェントの使用可能化」をクリックします。
 - c. エージェント構成のデフォルト値を検査し、必要に応じて変更します。
 - d. 「OK」をクリックします。

Business Space でヒューマン・タスク管理ウィジェットを実行するためにフォームを使用可能にする

WebSphere Process Server を使用している場合は、追加のステップを実行して、Business Spaceでヒューマン・タスク管理ウィジェットを使用するためにフォームを使用可能にする必要があります。

このタスクについて

トピックの対象範囲: このトピックは、以下の製品を対象としています。

- WebSphere Business Compass
- WebSphere Business Monitor
- WebSphere Process Server
- WebSphere Business Services Fabric

Business Process Choreographer 以外のサーバー・インスタンスにBusiness Spaceがインストールされている場合は、別のエンタープライズ・アプリケーションにデプロイされているフォームを、ヒューマン・タスク管理ウィジェットに対して使用可能にするために追加のステップを実行する必要があります。これには、WebSphere Integration Developer で生成した HTML-Dojo フォーム、および IBM Lotus Forms が含まれます。

Business Space と WebSphere Process Server の両方を同じ WebSphere Network Deployment セルのデプロイメント・ターゲットで構成するのか、それとも別々のセルで構成するのかに応じて、以下のステップのいずれかを実行します。

手順

1. シングル・セルのセットアップの場合: プロセスまたはヒューマン・タスクおよびフォームが含まれるエンタープライズ・アプリケーションをデプロイするときには、そのフォームの HTML ファイルまたは Lotus フォーム定義が含まれている Web モジュールを、Business Spaceが構成されているのと同じデプロイメント・ターゲットにマップする必要があります。
2. クロス・セル環境のセットアップの場合: フォームの HTML ファイルまたは Lotus フォーム定義が含まれている Web モジュールを、リモート・セル内の Business Space をホストするデプロイメント・ターゲットにデプロイします。Web モジュールをデプロイするときは、WebSphere Integration Developer のヒューマン・タスク・エディターでこれらのフォーム用に定義されたコンテキスト・ルートを指定する必要があります。Business Spaceのサーバーまたはクラスター上の新規アプリケーションを開始します。

次のタスク

Lotus Webform Server をヒューマン・タスク管理ウィジェットとともに使用する場合は、Lotus Webform Server を Business Space 用に構成する必要があります。

Business Space 内のヒューマン・タスク管理ウィジェットのための Lotus Webform Server の構成:

WebSphere Process Server ヒューマン・タスク管理ウィジェットを使用しており、実行時に Lotus Webform Server を使用してフォームを操作する場合は、Lotus Webform Server を使用するようにBusiness Spaceを構成する必要があります。

始める前に

Lotus Webform Server をBusiness Space内のヒューマン・タスク管理ウィジェットとともに使用できるようにするには、Lotus Webform Server 3.5.1 とフィックスパック 1 以降をインストールしておく必要があります。

Webform Server は 32 ビット・アーキテクチャーのマシンでのみ稼働します。

Webform Server をインストールする場合は、インストール・ツールの「サーバー・コンポーネント」ページで、必ず「**Webform Server- Application Server**」と「**Webform Server - Translator Server**」の両方を選択してください。「オプション・デプロイメント設定 (Optional Deployment settings)」ページで、必ず「**Webform Server - Translator Server を WebSphere Application Server にデプロイ (Deploy Webform Server - Translator Server to WebSphere Application Server)**」を選択し

てください。「API を WebSphere Application Server または WebSphere Process Server にデプロイ (Deploy API to WebSphere Application Server or WebSphere Process Server)」を選択しないでください。

注: Derby データベースを使用する場合は、別個のプロファイルに Lotus Webform Server をインストールする必要があります。Business Space および WebSphere Process Server と同じプロファイルを使用することはできません。

このタスクについて

トピックの対象範囲: このトピックは、以下の製品を対象としています。

- WebSphere Business Compass
- WebSphere Business Monitor
- WebSphere Process Server
- WebSphere Business Services Fabric

環境に応じて、以下の 3 つのステップのいずれかを実行してください。

手順

1. シングル・サーバー環境を使用しており、Lotus Webform Server が WebSphere Process Server と同じシステムに既にインストールされている場合は、プロファイル管理ツールを使用して Lotus Webform Server を Business Space 用に構成します。それ以外の場合、ステップ 2 に進みます。
 - a. プロファイル管理ツールを開始して、スタンドアロン・サーバー・プロファイルを作成します。
 - b. 「プロファイル作成オプション」ページで、「**拡張**」オプションを選択します。
 - c. 「ビジネス・スペースの構成」ページで、「**Lotus Webform Server の構成 (Configure Lotus Webform Server)**」チェック・ボックスを選択し、Webform Server Translator およびインストールのルートを入力します。詳しくは、『**拡張スタンドアロン・サーバー・プロファイルの作成**』を参照してください。
2. WebSphere Process Server がインストールされているのと同じシステムに Lotus Webform Server がインストールされており、プロファイル管理ツールで Lotus Webform Server を構成しなかった場合は、以下のステップを実行します。それ以外の場合、ステップ 3 に進みます。
 - a. ご使用のプロファイルについて、コマンド・ウィンドウを開きます。
wsadmin コマンドは、`profiles¥profile_name¥bin` ディレクトリーに格納されています。クラスター環境の場合、`deployment_manager_profile_root¥bin` ディレクトリーからコマンドを実行します。スタンドアロン・サーバー環境の場合、`profile_root¥bin` ディレクトリーからコマンドを実行します。
 - b. コマンド・プロンプトで、wsadmin コマンドを入力し、wsadmin 環境を開始します。例えば、Windows プラットフォームの場合は `wsadmin.bat -conntype NONE` と入力します。
 - c. Webform Server が配置されているのと同じマシン上で、ローカル・ホストおよび場所を指定して `configureWebformServer` コマンドを実行します。
例えば、Jython を使用して以下のコマンドを実行します。

```
AdminTask.configureLotusWebformServer(['-nodeName', node_name,
'-serverName', server_name, '-translatorHTTPLocation',
'http://localhost:8085/translator', '-serverInstallRoot',
'C:/IBM/LotusWebForms/3.5/WebFormServer'])
```

```
AdminConfig.save()
```

または、Jacl を使用して以下のコマンドを実行します。

```
$AdminTask configureLotusWebformServer {-nodeName node_name -serverName
server_name -translatorHTTPLocation http://localhost:8085/translator
-serverInstallRoot C:/IBM/LotusWebForms/3.5/WebFormServer}
```

```
$AdminConfig save
```

3. WebSphere Process Server がインストールされているのとは異なるシステムに Lotus Webform Server がインストールされている場合、以下のステップを実行します。
 - a. BSpaceWebformsEnabler.ear を、*profile root/installableApps/BusinessSpace* ディレクトリーから Webform Server がインストールされているシステムにコピーします。この ear をリモート・アプリケーション・サーバーにデプロイします。
 - b. ローカルの Business Space プロファイル (bspaceWFSEndpoints.xml ファイル内) で、エンドポイント {com.ibm.bspace}bpaceWebformsProxyRootId が BSpaceWebformsEnabler.ear の完全修飾ロケーションを参照するように設定します。エンドポイント・ファイルの編集については、『Business Space ウィジェットをクロス・セル環境用に手動で使用可能にする』を参照してください。
 - c. Webform Server システムで、Lotus Webform Server を構成したプロファイルの管理コンソールを開きます。
 - d. 以下の変数を設定します。「環境」 → 「WebSphere 変数」をクリックし、使用中のサーバーを含むノードを選択し、次に「新規」をクリックしてそれぞれの新しい変数を設定します。
 - Webform Server インストール・ディレクトリー変数を設定するには、名前が LFS_DIR で、値が Webform Server のインストール場所 (c:¥Program Files¥Lotus Webform Server¥3.5¥WebformServer など) である変数を作成します。
 - LFS_API_DIR 変数を設定するには、名前が LFS_API_DIR で値が \$(LFS_DIR)¥Translator¥API の変数を作成します。
 - LFS_API_LIB_DIR 変数を設定するには、名前が LFS_API_LIB_DIR で値が \$(LFS_API_DIR)¥76¥java¥classes の変数を作成します。
 - LFS_DEP_DIR 変数を設定するには、名前が LFS_DEP_DIR で値が \$(LFS_DIR)¥redist の変数を作成します。
 - e. Java プロセス定義を設定します。
 - 「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「server_name」 → 「Java およびプロセス管理」 → 「プロセス定義」 → 「環境項目 (Environment Entries)」をクリックします。

- PUREEDGE_INI プロパティと、値
`${LFS_DIR}¥Translator¥PureEdgeAPI.ini` を追加します。
- f. 共有ライブラリー LFS_API_LIB および LFS_DEP_LIB を設定します。
- 「環境」 → 「共有ライブラリー (Shared Libraries)」をクリックします。
 - 使用中のサーバーを含むノードを選択してスコープを設定します。スコープは、環境変数の設定値と同じスコープである必要があります。
 - 「新規」をクリックします。
 - 「LFS_API_LIB」とクラスパスを名前として持つエントリーを作成します (1 行に 1 つずつ)。
 - `${LFS_API_LIB_DIR}/pe_api.jar`
 - `${LFS_API_LIB_DIR}/pe_api_native.jar`
 - `${LFS_API_LIB_DIR}/uwi_api.jar`
 - `${LFS_API_LIB_DIR}/uwi_api_native.jar`
 - `${LFS_API_LIB_DIR}/commons-codec.jar`
 - `${LFS_API_LIB_DIR}/xmlsec-1.4.1.jar`
 - 「OK」をクリックします。
 - 「新規」をクリックします。
 - 「LFS_DEP_LIB」とクラスパスを名前として持つエントリーを作成します (1 行に 1 つずつ)。
 - `${LFS_DEP_DIR}/commons-codec-1.3.jar`
 - `${LFS_DEP_DIR}/commons-httpclient-3.0.jar`
 - `${LFS_DEP_DIR}/ehcache-1.2.2.jar`
 - `${LFS_DEP_DIR}/log4j-1.2.8.jar`
 - `${LFS_DEP_DIR}/ws_common.jar`
 - `${LFS_DEP_DIR}/ws_framework.jar`
 - `${LFS_DEP_DIR}/ws_resourcestore.jar`
 - `${LFS_DEP_DIR}/ws_resourcebundle.jar`
 - 「OK」をクリックします。
- g. サーバー・クラス・ローダーを設定します。
- 「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「*server_name*」 → 「Java およびプロセス管理」 → 「クラス・ローダー (Classloader)」をクリックします。
 - アプリケーション・サーバーのクラス・ローダーが存在しない場合、クラス・ローダーを作成する必要があります。「新規」をクリックして、「親が最後」オプションを選択します。
 - アプリケーション・サーバーのクラス・ローダーを選択し、「共有ライブラリー参照 (Shared Library References)」をクリックします。
 - 「追加」をクリックします。
 - 「ライブラリー名 (Library Name)」リストから LFS_API_LIB を選択します。
 - ライブラリー LFS_DEP_LIB について同じ手順を繰り返します。

- 「OK」をクリックします。
- h. Webform Translator の場所を構成します。
 - BSpaceWebformsEnabler EAR がデプロイされていることを確認します。
 - 「アプリケーション」 → 「アプリケーション・タイプ」 → 「WebSphere エンタープライズ・アプリケーション」 → 「BSpaceWebformsEnabler」 → 「サープレットのパラメーターを初期化」をクリックします。
 - translatorLocation の値を Webform Server Translator の http アドレスに設定します。Translator が BSpaceWebFormsEnabler と同じマシン上で実行するように構成されている場合、デフォルト値の http://localhost:8085/translator のままにします。
- i. すべての変更をマスター構成に保存し、サーバーを再始動します。

ヒューマン・タスク管理ウィジェットでイメージを使用可能にする

ヒューマン・タスク管理ウィジェットを含むように Business Space を設定する場合は、これらのウィジェットでチーム・メンバーのイメージを使用するようにエンドポイント・ファイルを作成できます。ユーザー ID を表示し、このユーザー ID によってグループ化できるように構成されているすべてのウィジェットについて、イメージの表示を有効にすることができます。

始める前に

トピックの対象範囲: このトピックは、以下の製品を対象としています。

- WebSphere Business Compass
- WebSphere Business Monitor
- WebSphere Process Server
- WebSphere Business Services Fabric

このタスクについて

デフォルトでは、Business Space は、ヒューマン・タスク管理ウィジェットに対してイメージ・サーバーを指定せずに構成されますが、ビジネス・ユーザーにチーム・メンバーのイメージを表示する場合は、新しいウィジェット・エンドポイント・ファイルのイメージの取得を使用可能に設定することができます。

手順

1. `install_root%BusinessSpace%registryData%` に新規ファイルを作成します。例えば、`imageEndpoint.xml` という名前を付けます。
2. 以下のテンプレートの中にコピーします。

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:BusinessSpaceRegistry xmlns:tns="http://com.ibm.bspace/
BusinessSpaceRegistry" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://com.ibm.bspace/BusinessSpaceRegistry
BusinessSpaceRegistry.xsd ">
<tns:Endpoint>
<tns:id>{com.ibm.bspace.htm}bspaceUserImageServiceRootId</tns:id>
<tns:type>{com.ibm.bspace.htm}bspaceUserImageServiceRootId</tns:type>
<tns:version>1.0.0.0</tns:version>
<tns:url>URL</tns:url>
<tns:description>Location of user image services</tns:description>
</tns:Endpoint>
</tns:BusinessSpaceRegistry>
```

3. ユーザー・イメージに使用する適切なイメージ・サーバー・サーブレットを参照するように URL を更新します。

イメージ・サービス・エンドポイントは URL 接頭部に対する参照であり、これを使用して、ウィジェットは次の情報を連結することによりイメージを検出できます。

- 解決されたイメージ・サービス・エンドポイント・ストリング。
- 各ユーザーの固有 ID である Virtual Member Manager (VMM) 属性。
- .jpg ファイル拡張子。

例えば、エンドポイントの URL が `http://myserver:9080/UserImageWeb/UserImageServlet/` で、ユーザーの固有 ID が `id123456` である場合、ウィジェットは `http://myserver:9080/UserImageWeb/UserImageServlet/id123456.jpg` というリンクにある、そのユーザーのイメージを取得します。

4. `updateBusinessSpaceWidgets` コマンドを実行します。
 - a. ご使用のプロファイルについて、コマンド・ウィンドウを開きます。

`wsadmin` コマンドは `profiles/profile_name/bin` ディレクトリーに格納されています。
 - b. `updateBusinessSpaceWidgets` コマンドを使用して、指定したウィジェットのインストール、デプロイ、および登録を行います。

Business Spaceのセキュリティーのセットアップ

ご使用の環境で WebSphere が提供する Business Space を利用しようとする場合は、Business Space の成果物をチームがどのように処理するかについてのセキュリティー・オプションを検討する必要があります。Business Space のセキュリティーをオンにする場合は、アプリケーション・セキュリティーをセットアップし、ユーザー・リポジトリーを指定します。Business Space 管理者を定義するには、スーパーユーザー・ロールを割り当てます。

このタスクについて

最善の結果を得るため、Business Space を構成する前にセキュリティーを有効にします。管理コンソールの「グローバル・セキュリティー」管理ページで、管理セキュリティーとアプリケーション・セキュリティーの両方を有効にします。また、ユーザー・アカウント・リポジトリーも指定します。

ユーザー・アカウント・レジストリーを Business Space と共に使用する場合の考慮事項:

- 使用している LDAP 構成のタイプに応じて、設定が、Business Space に正しくアクセスできる能力に影響を及ぼすことがあります。ユーザー・フィルター、グループ・フィルター、およびマッピング設定が適切に構成されていることを確認してください。詳しくは、WebSphere Application Server 資料の『Lightweight Directory Access Protocol 検索フィルターの構成』を参照してください。
- 使用しているフェデレーテッド・リポジトリー構成のタイプに応じて、設定が、Business Space に正しくアクセスできる能力に影響を及ぼすことがあります。レ

ルムが適切に構成されていることを確認してください。詳しくは、WebSphere Application Server 資料の『フェデレーテッド・リポジトリ構成におけるレルムの管理』を参照してください。

- Business Spaceでの検索にログイン・プロパティ uid (ユーザー ID) を使用するように LDAP セキュリティーがデフォルトでセットアップされます。ログイン・プロパティに、mail (E メール・アドレス) など、別の固有の LDAP フィールドを使用するように LDAP セキュリティーを変更する場合は、Business Space で検索が機能するようにするため、ConfigServices.properties ファイルの `userIdKey` プロパティを変更する必要があります。ConfigServices.properties ファイルは、スタンドアロン・サーバーの場合は `profile_root%BusinessSpace%node_name%server_name%mm.runtime.prof%config%ConfigService.properties` に、クラスターの場合は `deployment_manager_profile_root%BusinessSpace%cluster_name%mm.runtime.prof%config%ConfigService.properties` にあります。LDAP セキュリティーのログイン・プロパティに合わせて、`userIdKey` 属性を `uid` から、例えば、`mail` に変更します。以下のパラメーターを指定して、`wsadmin` スクリプト・クライアントを使用して `updatePropertyConfig` コマンドを実行します。
-serverName および **-nodeName** (スタンドアロン・サーバーの場合) または **-clusterName** (クラスターの場合)、**-propertyFileName** (値は ConfigServices.properties ファイルのパス)、ならびに **-prefix** (値は Mashups_)。
- Microsoft SQL Server データベースおよび **Standalone LDAP** レジストリーを使用する場合は、必ずユーザー識別名 (ユーザー DN) を 131 文字より少ないようにしてください。いずれかのユーザー DN 項目が 131 文字以上である場合は、ユーザー・アカウント・リポジトリに「フェデレーテッド・リポジトリ」オプションを指定する必要があります。フェデレーテッド・リポジトリと他のレジストリーとの切り替えを行うと、すべての既存のスペースおよびページは Business Space でアクセスできなくなるので、再作成する必要があります。
- 「フェデレーテッド・リポジトリ」を使用すると、拡張検索機能などの、ウィジェットおよびフレームワークの機能が追加されます。スペースおよびページを共用するためにユーザーを検索をするときは、検索有効範囲に E メール、ユーザーのフルネーム、およびユーザー ID が含まれます。

IBM Tivoli Access Manager WebSEAL を使用しており、それを Business Space 環境と共に使用したい場合は、追加の構成ステップを実行する必要があります。外部 Java Authorization Contract for Containers (JACC) プロバイダーと一緒に Tivoli Access Manager のセキュリティを構成し、Tivoli Access Manager で WebSEAL を構成し、製品アプリケーション・サーバーで WebSEAL を構成し、ご使用の環境のホスト・ジャンクションを構成します。

Business Space 環境で管理者となるユーザーを設定するため、スクリプトを実行して Business Space スーパーユーザー・ロールを割り当てます。

Business Spaceのアプリケーション・セキュリティの設定

Business Spaceのセキュリティをオンにするには、アプリケーション・セキュリティと管理セキュリティの両方を使用可能にする必要があります。

始める前に

このタスクを完了する前に、以下のタスクを完了しておく必要があります。

- 製品のユーザー・レジストリーにユーザー ID が登録されていることの確認。

保護された環境を使用することを予定している場合は、必ず、Business Space を構成する前にセキュリティーを有効にしてください。Business Spaceの構成後にセキュリティーを有効化または除去する場合は、ConfigServices.properties ファイルの MashupAdminForOOBSpace プロパティーと noSecurityAdminInternalUserOnly プロパティーの両方を変更し、有効な管理者 ID として正しいユーザー ID を設定する必要があります。ConfigServices.properties ファイルは、スタンドアロン・サーバーの場合は `profile_root%BusinessSpace%node_name%server_name%mm.runtime.prof%config%ConfigService.properties` に、クラスターの場合は `deployment_manager_profile_root%BusinessSpace%cluster_name%mm.runtime.prof%config%ConfigService.properties` にあります。変更後のファイルを、システム上の空のフォルダーにコピーします。次に、wsadmin スクリプト・クライアントを使用し、以下のパラメーターを指定して updatePropertyConfig コマンドを実行します。

- **-serverName** および **-nodeName** (スタンドアロン・サーバーの場合) または **-clusterName** (クラスターの場合)
- **-propertyFileName** (値は ConfigServices.properties ファイルのパス)
- **-prefix** には値 Mashups_ を指定します

このタスクについて

Business Spaceは、アクセスの認証と許可を確実に行うように事前に構成されます。Business Space URL へのアクセス時には、ユーザーに認証のプロンプトが出されません。認証されないユーザーは、ログイン・ページにリダイレクトされます。Business Space は、HTTP と HTTPS のいずれかによってアクセスすることができます。ただし、ログイン・ページの場合は常に HTTPS にリダイレクトされます。したがって、IBM HTTP Server などの Web サーバーを使用する場合は、HTTPS をサポートするようにそのサーバーを構成する必要があります。

Business Spaceのスペースおよびページ内容への権限は、スペース管理の一部として Business Space内で処理されます。

Business Spaceへの認証済みアクセスを使用可能にするには、ユーザー・レジストリーを構成して、アプリケーション・セキュリティーを使用可能にする必要があります。

手順

1. セキュリティーの詳細な説明については、製品のセキュリティー・ドキュメンテーションを参照してください。
2. Business Space アプリケーションの場合は、「グローバル・セキュリティー」管理コンソール・ページで、「管理セキュリティーを使用可能にする」および「アプリケーション・セキュリティーを使用可能にする」の両方を選択します。
3. 同じ管理コンソール・ページ上の「ユーザー・アカウント・リポジトリー」で、「フェデレーテッド・リポジトリー」、「ローカル・オペレーティング・システム」、「スタンドアロン LDAP レジストリー」、または「スタンドアロン・カ

スタム・レジストリー」のいずれかを指定します。『Business Space のセキュリティのセットアップ』で、ユーザー・レジストリーの選択に関する考慮事項を確認してください。

4. Business Space が製品を実行している場所から離れている場合、および Business Space を実行中のノードと製品を実行中のノードが同じセル内にはない場合には、手動のステップによって確実にシングル・サインオン (SSO) を有効にする必要があります。例えば、複数の製品 (WebSphere Business Compass、WebSphere Business Monitor、WebSphere Enterprise Service Bus、または WebSphere Process Server) を使用し、サーバーをそれぞれ異なるノードに配置し、さらにそれらすべてが Business Space ・サーバーと連携できるようにする場合は、手動で SSO を構成する必要があります。SSO を有効にするには、以下のステップを実行します。
 - a. 各サーバーの管理コンソールで、「セキュリティ」 → 「グローバル・セキュリティ」をクリックして、「グローバル・セキュリティ」ページを開きます。「Web および SIP セキュリティ」を展開し、「シングル・サインオン (SSO)」をクリックして、「使用可能」チェック・ボックスが選択されていることを確認します。
 - b. すべてのノードが同じ「ユーザー・アカウント・リポジトリー」の情報 (ステップ 3 を参照) を使用することを確認します。
 - c. 最初のノードの管理コンソールで、「グローバル・セキュリティ」ページを開きます。「認証」の下にある「LTPA」をクリックします。
 - d. 「クロス・セル・シングル・サインオン」の下に、鍵ファイルのパスワード、および完全修飾鍵ファイル名 (つまり、鍵ファイルをエクスポートするロケーションとファイル名) を入力します。完全修飾鍵ファイル名は、サーバーが稼働しているシステム上での絶対パスです。
 - e. 「鍵のエクスポート」をクリックします。鍵ファイルは、サーバーが稼働しているシステムに保管されます。
 - f. 2 つのノードが同じシステム上にない場合は、鍵ファイルを物理的にその他のシステムにコピーします。
 - g. 同じ鍵ファイルを使用するその他すべてのノードに鍵ファイルをインポートするには、他のノードの管理コンソールにログオンして、「グローバル・セキュリティ」 > 「LTPA」ページに移動します。「クロス・セル・シングル・サインオン」の下に、鍵ファイルのパスワード、および完全修飾鍵ファイル名 (コピーしたエクスポート済み鍵ファイルと同じパスワードを使用) を入力し、「鍵のインポート」をクリックします。
 - h. 各システムに鍵をインポートした後、サーバーを再始動します。
5. エンドポイント・ファイルで HTTPS を使用していて、エンドポイントのロケーションが Business Space とは異なるノード上にあり、Secure Sockets Layer (SSL) 証明書が自己署名 SSL 証明書である場合には、その SSL 証明書をインポートする必要があります。
 - a. Business Space が含まれているサーバーの管理コンソールにログオンし、製品を実行しているリモート・ノードが使用する SSL 証明書をインポートします。
 - 1) 「セキュリティ」の下にある「SSL 証明書と鍵の管理」をクリックします。

- 2) 「SSL 証明書と鍵の管理」ページの「関連項目」で、「鍵ストアおよび証明書」をクリックします。
 - 3) 「鍵ストアおよび証明書」ページで、「NodeDefaultTrustStore」をクリックして truststore タイプを変更します。
 - 4) 「NodeDefaultTrustStore」ページの「追加プロパティ」の下で、「署名者証明書」をクリックします。
 - 5) NodeDefaultTrustStore の「署名者証明書」ページで、「ポートから取得」ボタンをクリックします。
 - 6) 「ポートから取得」ページの「一般プロパティ」に、製品を実行している場所でのホスト、ポート、および別名を入力します。「署名者情報の取得」ボタンをクリックして、「OK」をクリックします。
 - 7) 両方のサーバーを再始動します。
- b. 製品ノードの管理コンソールにログオンし、Business Space を実行しているノードが使用する SSL 証明書をインポートします。
- 1) ステップ a. の 1) から 5) を繰り返します。
 - 2) 「ポートから取得」ページの「一般プロパティ」に、Business Space を実行している場所でのホストおよびポートを入力します。「署名者情報の取得」ボタンをクリックして、「OK」をクリックします。
 - 3) 両方のサーバーを再始動します。

SSO および SSL については詳しくは、WebSphere Application Server インフォメーション・センターを参照してください。

次のタスク

- 管理セキュリティーおよびアプリケーション・セキュリティーをオンにした後は、Business Spaceにログオンすると、ユーザー ID およびパスワードを求めるプロンプトが表示されます。ログオンするためには、選択したユーザー・レジストリーから有効なユーザー ID およびパスワードを使用する必要があります。管理セキュリティーをオンにした後は、管理コンソールに戻るたびに管理権限を持つユーザー ID でログオンする必要があります。
- Business Space のページおよびスペースに対する権限を設定するには、Business Space のページおよびスペースを作成するときに権限を管理することができます。
- ユーザーおよびグループに基づいてウィジェット内のデータのセキュリティーをセットアップするには、REST サービス・ゲートウェイ・アプリケーションへのユーザーのマッピングを変更する必要があります。REST サービス・ゲートウェイ・アプリケーションを選択し、右のパネルの、「詳細プロパティ」の下の、「ユーザー/グループ・マッピングへのセキュリティー・ロール」を選択します。RestServicesUser ロールの場合、ユーザーおよびグループを追加して、すべての REST サービス・ウィジェットのデータへのアクセスを制御することができます。
- ユーザー・グループ・ロールに基づいてウィジェットのデータへのアクセスを制限したい場合は、管理グループ・ロールに割り当てたユーザーを変更することを検討してください。管理コンソールを開いて、「セキュリティー」→「管理、アプリケーション、およびインフラストラクチャーの保護」→「管理グループ・ロ

ール」 をクリックし、グループを選択することにより、ロール・リストを表示してこれらのロールに割り当てられているユーザーを確認することができます。

ビジネス・ルールやビジネス変数などの、ウィジェットの管理グループ・ロールに割り当てられたユーザーを変更することを検討したい場合があります。

例えば、システム正常性ウィジェットの場合、以下の管理ロールはすべてモニター権限を持ち、管理コンソールへのアクセスが可能であるため、これらのロールに割り当てられたユーザーはシステム正常性ウィジェットのデータにアクセスできます。

- モニター
- コンフィギュレーター
- オペレーター
- 管理者
- **Adminsecuritymanager**
- デプロイヤー
- **iscadmins**

これらの管理グループ・ロールにマップされたユーザーは、システム正常性ウィジェットのデータにアクセスできます。これらのロールにマップされていないユーザーは、システム正常性ウィジェットのデータにアクセスできません。

- さらに、ウィジェットの中には、ビジネス・ユーザーが作成した成果物へのロール・ベースのアクセスの追加層を持つものもあります。ソリューション管理の場合、セキュリティ・ロール・ウィジェットを使用すると、ビジネス・カレンダー・ウィジェットのタイムテーブルに対してメンバーが持つアクセス権限のレベルを決定するユーザーおよびグループのシステム・ロールまたはモジュール・ロールを割り当てることができます。レビューの場合、アクセス制御のレビュー・ウィジェットは、レビューを行いコメントを入力できるユーザーのアクセス権限を管理します。詳しくは、ご使用のウィジェットのオンライン・ヘルプを参照してください。

注:

SystemOut.log ファイルで次のエラーを見つけた場合は、処理できない余分な属性がユーザー・レジストリーに含まれている可能性があります。

```
00000046 SystemErr R Caused by: com.ibm.websphere.wim.exception.WIMSystemException: CWWIM1013E
The value of the property secretary is not valid for entity uid=xxx,c=us,ou=yyy,o=ibm.com.
00000046 SystemErr R at com.ibm.ws.wim.adapter.ldap.LdapAdapter.setPropertyValue(LdapAdapter.java:3338)
```

これらの属性をバイパスするには、ConfigServices.properties ファイルで以下の属性を設定します。

```
com.ibm.mashups.user.userProfile = LIMITED
com.ibm.mashups.user.groupProfile = LIMITED
```

ConfigServices.properties ファイルは、スタンドアロン・サーバーの場合は `profile_root%BusinessSpace%node_name%server_name`、`%mm.runtime.prof%config%ConfigService.properties` に、クラスターの場合は `deployment_manager_profile_root%BusinessSpace%cluster_name`、`%mm.runtime.prof%config%ConfigService.properties` にあります。

ConfigServices.properties ファイルに変更を加えた後、以下のパラメーターを指定して、wsadmin スクリプト・クライアントを使用して updatePropertyConfig コマンドを実行します。 **-serverName** および **-nodeName** (スタンドアロン・サーバーの場合) または **-clusterName** (クラスターの場合)、**-propertyFileName** (値は ConfigServices.properties ファイルのパス)、ならびに **-prefix** (値は Mashups_)。

注:

クラスターで Java 2 セキュリティーを有効にした場合は、Business Space のヘルプのロケーションに適用されるサーバー・ポリシーの入力設定を強化することを検討してください。

Business Space のヘルプのロケーション・ポリシーは次のようになっています。

```
grant codeBase      "file:${was.install.root}/profiles/profile_name/temp/
node_name/-" {

    permission java.security.AllPermission;

};
```

このポリシーを次のように変更して強化します。

```
grant codeBase      "file:${was.install.root}/profiles/profile_name/temp/
node_name/server_name/BusinessSpaceHelpEAR_node_name_server_name/
BusinessSpaceHelp.war/-" {

    permission java.security.AllPermission;

};
```

Business Space を操作できるように Tivoli Access Manager WebSEAL を構成する

Tivoli Access Manager WebSEAL を保有しており、それを Business Space と共に使用したい場合は、いくつかの追加構成ステップを実行する必要があります。

このタスクについて

Tivoli Access Manager WebSEAL を Business Space と共に使用する場合は、外部 Java Authorization Contract for Containers (JACC) プロバイダーと一緒に Tivoli Access Manager のセキュリティーを構成し、Tivoli Access Manager で WebSEAL を構成し、製品アプリケーション・サーバーで WebSEAL を構成し、ご使用の環境のホスト・ジャンクションを構成します。

手順

1. Tivoli Access Manager を JACC と一緒に構成します。
 - a. 管理コンソールと wsadmin コマンドのどちらを使用するかに応じて、以下のいずれかのステップを実行します。
 - 管理コンソールを使用して Tivoli Access Manager を JACC と一緒に構成する場合は、以下のステップを実行します。
 - 1) グローバル・セキュリティーを有効にします。

- a) 「セキュリティ」 → 「グローバル・セキュリティ」を選択します。
- b) Tivoli Access Manager と一緒に構成する LDAP サーバーの「管理セキュリティ」、「アプリケーション・セキュリティ」、および「Java 2 セキュリティ」を有効にします。
- c) 「グローバル・セキュリティ」 → 「LDAP」を選択し、以下の情報を入力して「OK」をクリックします。

名前	説明
サーバー・ユーザー ID	Tivoli Access Manager の設定で管理者 DN に入力したのと同じユーザー ID を入力します。例: user1
サーバー・ユーザー・パスワード	puser1
ホスト	Tivoli Access Manager と一緒に構成する LDAP
ポート	例: 389
基本識別名	例: o=ibm, c=us
バインド識別名	例: cn=SecurityMaster,secAuthority=Default
バインド・パスワード	SecurityMaster ユーザーのパスワード

- d) 構成を保存して、サーバーを再始動します。
- 2) Tivoli Access Manager および JACC の外部許可を有効にします。
 - a) 「セキュリティ」 → 「グローバル・セキュリティ」 → 「外部許可プロバイダー」を選択します。
 - b) 「許可プロバイダー」リストで、「外部 JACC プロバイダー」を選択し、「構成」をクリックします。Tivoli Access Manager のデフォルトのプロパティは正しく設定されているので、デフォルトの値は変更しないでください。
 - c) 「追加プロパティ」の下の「Tivoli Access Manager プロパティ」を選択します。「組み込み Tivoli Access Manager を使用可能にする」を選択し、以下の情報を入力して「OK」をクリックします。

名前	値
クライアント listen ポートの設定	デフォルト設定は 8900 から 8999 です。別のポートを使用する場合にのみ、変更してください。
ポリシー・サーバー (name:port)	該当する <i>policyserver:port</i> を指定します。例: windomain3.rtp.raleigh.ibm.com:7135
許可サーバーと優先順位 (name:port:priority)	該当する <i>authorizationserver:port:priority</i> を指定します。例: windomain3.rtp.raleigh.ibm.com:7136:1

名前	値
管理者ユーザー名	Tivoli Access Manager サーバーで別の管理者名を使用する場合を除き、ユーザー名は「 sec_master 」(デフォルト)のままにしてください。
管理者ユーザー・パスワード	domino123
ユーザー・レジストリー識別名の接尾部	アプリケーション・サーバーに使用する名前を入力します。例: o=ibm, c=us
セキュリティー・ドメイン	セキュリティー・ドメインの設定はデフォルトのままにします。Tivoli Access Manager サーバーでデフォルトのドメインを使用しない場合は、この設定を変更します。Tivoli Access Manager サーバーで複数のドメインを作成し、デフォルト以外のドメインに接続する場合またはそのようなドメインを使用する場合は、この設定を変更します。
管理者ユーザーの識別名	ユーザーの完全修飾名を入力します。例: cn=user1,o=ibm,c=us 注: このユーザーは、LDAP ユーザー・レジストリー・パネルで構成された「サーバー・ユーザー ID」と同じです。

このサーバーは、Tivoli Access Manager サーバーに接続して、いくつかのプロパティ・ファイルをアプリケーション・サーバーの下に作成します。このプロセスには数分かかることがあります。エラーが発生した場合は、system Out を調べて、問題を修正してください。

- wsadmin ユーティリティーを使用して Tivoli Access Manager を JACC と一緒に構成する場合は、以下のステップを実行します。以下の手順は、デプロイメント・マネージャー・サーバーで 1 回だけ実行します。同期化が実行されると、構成パラメーターが管理対象サーバー (ノード・エージェントを含む) に転送されます。構成の変更を有効にするには、管理対象サーバー自体を再始動する必要があります。
 - 1) すべての管理対象サーバー (ノード・エージェントを含む) が始動されたことを確認します。
 - 2) サーバーを始動します。
 - 3) `install_root/bin` ディレクトリーから `wsadmin` コマンドを実行して、コマンド行ユーティリティーを開始します。
 - 4) `wsadmin` のプロンプトで、以下の表内の該当する情報を指定して、`configureTAM` コマンドを実行します。

Jacl の例:

```
$AdminTask configureTAM -interactive
```

Jython の例:

AdminTask.configureTAM('-interactive')続けて、以下の情報を入力します。

名前	値
製品サーバーのノード名	ノードを 1 つだけ指定するか、アスタリスク (*) を入力してすべてのノードを選択します。
Tivoli Access Manager ポリシー・サーバー	Tivoli Access Manager ポリシー・サーバーの名前、および接続ポートを入力します。 <i>policy_server:port</i> という形式を使用します。 ポリシー・サーバーの通信ポートは Tivoli Access Manager の構成時に設定されます。デフォルトのポートは 7135 です。
Tivoli Access Manager 許可サーバー	Tivoli Access Manager 許可サーバーの名前を入力します。 <i>auth_server:port:priority</i> という形式を使用します。許可サーバーの通信ポートは Tivoli Access Manager の構成時に設定されます。デフォルトのポートは 7136 です。エントリーをコンマで区切ることにより、複数の許可サーバーを指定することができます。複数の許可サーバーを構成しておくことは、フェイルオーバーおよびパフォーマンスの観点で有効です。優先順位の値は許可サーバーの使用順序です。例えば、 <i>auth_server1:7136:1,auth_server2:7137:2</i> です。優先順位 1 は、許可サーバーを 1 つだけ構成する場合でも指定する必要があります。
製品サーバーの管理者の識別名	製品サーバーのセキュリティー管理者 ID の完全識別名を入力します。例えば、 <i>cn=wasadmin,o=organization,c=country</i> です。詳細については、関連リンクを参照してください。
Tivoli Access Manager ユーザー・レジストリーの識別名サフィックス	例: <i>o=organization, c=country</i>
Tivoli Access Manager 管理者のユーザー名	Tivoli Access Manager の構成時に作成される Tivoli Access Manager 管理者のユーザー ID を入力します。通常、この ID は <i>sec_master</i> です。
Tivoli Access Manager 管理者のユーザー・パスワード	Tivoli Access Manager 管理者のパスワードを入力します。
Tivoli Access Manager セキュリティー・ドメイン	ユーザーおよびグループを保管するために使用する Tivoli Access Manager セキュリティー・ドメインの名前を入力します。Tivoli Access Manager の構成時にセキュリティー・ドメインを確立していない場合は、「戻る」をクリックしてデフォルトを受け入れます。

名前	値
組み込み Tivoli Access Manager リスニング・ポート・セット	製品サーバーは、ポリシー・サーバーからの許可データベース・アップデートを TCP/IP ポートで listen します。特定のノードおよびマシンで複数のプロセスが実行されることがあるので、それらのプロセスのため、ポートのリストが必要です。Tivoli Access Manager クライアントがリスニング・ポートとして使用するポートを、コンマで区切って指定します。ポートの範囲を指定する場合は、最小値と最大値をコロンで区切って指定します。例えば、7999, 9990:9999 です。
据え置き	このオプションを yes に設定すると、次の再始動時まで、管理サーバーの構成が据え置かれます。no に設定すると、管理サーバーの構成が即座に有効になります。管理対象サーバーは次の再始動時に構成されます。

- 5) 必要な情報をすべて入力したら、「F」を選択して構成プロパティを保存します。または、「C」を選択して構成プロセスを取り消し、入力した情報を破棄します。

SVTM TAM60 サーバーの場合の例:

```
wsadmin>$AdminTask configureTAM -interactive
組み込み Tivoli Access Manager の構成
```

このコマンドは、指定された 1 つまたは複数の WebSphere Application Server ノードで組み込みの Tivoli Access Manager を構成します。

```
WebSphere Application Server ノード名 (nodeName): *
*Tivoli Access Manager Policy Server (policySvr):
  windomain3.rtp.raleigh.ibm.com:7135
*Tivoli Access Manager Authorization Server (authSvrs):
  windomain3.rtp.raleigh.ibm.com:7136:1
*WebSphere Application Server 管理者の識別名 (wasAdminDN):
  cn=was61admin,o=ibm,c=us
*Tivoli Access Manager ユーザー・レジストリー識別名の接尾部 (dnSuffix):
  o=ibm,c=us
Tivoli Access Manager 管理者のユーザー名 (adminUid):
  [sec_master]
*Tivoli Access Manager 管理者のユーザー・パスワード (adminPasswd):
  domino123
Tivoli Access Manager セキュリティ・ドメイン (secDomain): [Default]
Embedded Tivoli Access Manager listen ポート・セット (portSet): [9900:9999]
据え置き (defer): [no]
```

組み込み Tivoli Access Manager の構成

F (Finish)
C (Cancel)

Select [F, C]: [F] F

```
WASX7278I: 生成されたコマンド行: $AdminTask configureTAM {-policySvr
  windomain3.rtp.raleigh.ibm.com:7135 -authSvrs
  windomain3.rtp.raleigh.ibm.com:7136:1 -wasAdminDN cn=wa
組み込み Tivoli Access Manager 構成処置パラメーターが正常に
```

保存されました。

ターゲット・ノード上で実行されているすべての WebSphere Application Server インスタンスを再始動します。

wsadmin>

- 6) 管理コンソールで、「セキュリティ」 → 「グローバル・セキュリティ」 → 「外部許可プロバイダー」を選択します。次に、「JACC プロバイダーを使用する外部許可」を選択して「OK」をクリックします。
 - 7) メイン・セキュリティ画面に移動して、「OK」をクリックします。変更を保存して、同期化します。
 - 8) セル内のすべてのプロセスを再始動します。
- b. Tivoli Access Manager を有効にする前にアプリケーションをインストールした場合 (例えば、LDAP セキュリティーを有効にして、保護されたアプリケーションをいくつかインストールし、ユーザーおよびグループをセキュリティ・ロールにマップした場合など) は、セキュリティ・ロール・マッピング情報をデプロイメント記述子から Tivoli Access Manager ポリシー・サーバーに伝搬させます。管理コンソールと wsadmin コマンドのどちらを使用するかに応じて、以下のいずれかのステップを実行します。
- wsadmin コマンド propagatePolicyToJACCProvider を使用する場合は、『JACC プロバイダーへのインストール済みアプリケーションのセキュリティ・ポリシーの wsadmin スクリプトを使用した伝搬』を参照してください。
 - 管理コンソールを使用する場合は、『前にデプロイ済みのアプリケーションに対するセキュリティ・ポリシーとロールの伝搬』を参照してください。
2. Tivoli Access Manager で WebSEAL を構成します。
- a. WebSEAL がインストールされて適切に構成されていることを確認します。
 - b. TAI++ の場合は **-c iv_creds** オプションを使用して、TAI の場合は **-c iv_user** オプションを使用して、WebSEAL と製品アプリケーション・サーバーの間のジャンクションを作成します。ご使用の環境に応じた変数を使用して、以下のいずれかのコマンドを 1 行に入力します。

TAI++ の場合

```
server task webseald-server create -t tcp -b supply -c iv_creds
```

```
-h host_name -p websphere_app_port_number junction_name
```

- c. (TAI の構成に使用できるように) トラステッド・ユーザー・アカウントを Tivoli Access Manager で作成するには、以下のコマンドを発行します。

```
pdadmin -a sec_master -p domino123
```

```
pdadmin sec_master> user create -gsouser -no-password-policy taiuser  
"cn=taiuser,ou=websphere,o=ibm,c=us" taiuser taiuser ptaiuser
```

```
pdadmin sec_master> user modify taiuser password-valid yes
```

```
pdadmin sec_master> user modify taiuser account-valid yes
```

- d. WebSEAL 構成ファイル `webseal_install_directory/etc/webseald-default.conf` で、以下のパラメーターを設定します。

```
basicauth-dummy-passwd=webseal_userid_passwd
```

例えば、Tivoli Access Manager で `taiuser` または `ptaiuser` を設定する場合は、次のパラメーターを設定します。`basicauth-dummy-passwd = ptaiuser`

フォーム・ベースの認証を使用する場合は、以下のパラメーターを設定します。

```
forms-auth=both
```

```
ba-auth=none
```

3. 製品アプリケーション・サーバー上で TAI++ インターセプターを有効にして、そのサーバーで WebSEAL を構成します。
 - a. 管理コンソールで、「グローバル・セキュリティー」 → 「認証メカニズムと有効期限」を選択します。
 - b. 「Web および SIP セキュリティー」を展開し、「トラスト・アソシエーション」を選択します。チェック・ボックスを選択して、「適用」をクリックします。
 - c. 「インターセプター」 → 「TAMTrustAssociationInterceptorPlus」 → 「カスタム・プロパティー」を選択して、以下のプロパティーを追加します。

名前	値
<code>com.ibm.websphere.security.webseal.configURL</code>	<code>\${WAS_INSTALL_ROOT}/java/jre/PdPerm.properties</code>
<code>com.ibm.websphere.security.webseal.id</code>	<code>iv-creds</code>
<code>com.ibm.websphere.security.webseal.loginId</code>	<code>taiuser</code> (ユーザー <code>taiuser</code> または <code>ptaiuser</code> を Tivoli Access Manager で作成してある場合)

- d. セルを再始動します。
- e. クライアントにアクセスするため、`https://webseal_server_name:webseal_port/junction_name/web_uri_for_client` に移動します。
4. Business Space ウィジェットが表示されるように、ご使用の環境のホスト・ジャンクションを構成します。仮想ホスト・ジャンクションとトランスペアレント・ホスト・ジャンクションのどちらを使用するかに応じて、以下のいずれかのステップを実行します。
 - 仮想ホスト・ジャンクションを使用する場合は、それを作成します。仮想ホスト・ジャンクションを使用することにより、個別にジャンクションを作成しなくても済むようになります。
 - a. 仮想ホストが構成されていることを確認します。仮想ホスト・ジャンクションは、ホスト名、ポート番号、および転送アドレスをターゲット・ホストのものと突き合わせます。URL フィルタリングは一切実行されず、該当するすべての要求がターゲット・ホストに転送されます。
 - b. 同じ仮想ホストに対して以下のアプリケーションが有効になっていることを確認します。Business Space で使用している製品に応じて、これらのアプリケーションの一部を使用している場合と、全部を使用している場合が考えられます。

- BPMAdministrationWidgets_nodename_servername (WebSphere Enterprise Service Bus および WebSphere Process Server の場合)
- BusinessSpaceHelpEAR_nodename_servername (すべての製品の場合)
- BSpaceEAR_nodename_servername (すべての製品の場合)
- BSpaceWebformsEnabler_nodename_servername (すべての製品の場合)
- HumanTaskManagementWidgets_nodename_servername (WebSphere Process Server および WebSphere Business Monitor の場合)
- REST サービス・ゲートウェイ (すべての製品の場合)
- REST サービス・ゲートウェイ Dmgr (WebSphere Enterprise Service Bus および WebSphere Process Server の場合)
- mm.was_nodename_servername (すべての製品の場合)
- WBMDashboardWeb_nodename_servername (WebSphere Business Monitor の場合)
- wesbWidgets_nodename_servername (WebSphere Enterprise Service Bus の場合)
- widgets_busleader_nodename_servername (WebSphere Business Compass の場合)
- widgets_pubserver_nodename_servername (WebSphere Business Compass の場合)
- widgets_fabric_nodename_servername (WebSphere Business Services Fabric の場合)

注: このアプリケーション・リストに含まれているのは、Business Space に必要なアプリケーションのみです。状況によっては、Tivoli Access Manager WebSEAL を使用するが Business Space を使用しないシナリオ用に、他のアプリケーションをリストに追加することが必要な場合もあります。

- c. pdadmin を使用して、以下のコマンドを実行します。server task *webseal* server virtualhost create -t *transport* -h *target_host* [-p *port*] [-v *virtual_host_name*] *virtual_host_label*

次の情報を使用します。

- *webseal server* は、仮想ホスト・エントリーを作成する WebSEAL サーバーの名前です。
- *transport* はトランスポートのタイプです。有効なエントリーは、tcp、ssl、tcpproxy、および sslproxy です。
- *target_host* は、必要なアプリケーションのホストです。
- *virtual_host_name* は、仮想ホスト・ジャンクションに対する HTTP 要求の突き合わせに使用されます。何も値を入力しない場合は、デフォルトで、ターゲット・ホストとポートを組み合わせて値が設定されます。例えば、*virtual_host_name* を myvirthost.ibm.com:80 に設定した場合、WebSEAL は myvirthost.ibm.com:80 を含んでいる URL を、pdadmin コマンドで指定されているホストと突き合わせて経路指定します。
- *virtual_host_label* は、WebSEAL のエントリーを識別するためのラベルであり、固有のものでなければなりません。

期待したとおりに Business Space が実行されるようにするため、トランスポートのタイプとして `ssl` と `tcp` の両方のエントリーを作成する必要があります。Secure Sockets Layer (SSL) と伝送制御プロトコル (TCP) の両方を同じ仮想ホスト・ジャンクションでサポートする必要がある場合は、`-g vhost_label` オプションを使用する必要があります。`vhost_label` は構成を共有するためのオリジナルの仮想ホストのラベルです。このオプションを指定することにより、事前に作成されている仮想ホスト・ジャンクション (事前に作成された仮想ホスト・ジャンクションであり、`virtual_host_label` が `-g` オプションで指定されるラベルと一致するもの) が検出され、その構成が共有されるようになります。2 番目のエントリーについても、やはり、固有の `virtual_host_label` が必要ですが、ターゲット・ホスト、ポート、およびその他の値を共有することができます。この `-g` オプションを指定しない場合、WebSEAL がターゲット・ホストおよびポートが前に作成されたジャンクションのものと同じであるとみなし (そのような仮想ホストを作成することは許可されない)ので、2 番目の仮想ホストは作成されません。

- トランスペアレント・ホスト・ジャンクションを使用する場合は、それぞれの製品のウィジェット用の一連のトランスペアレント・パス・ジャンクションを作成します。
 - a. `pdadmin` を使用して、以下のコマンドを実行します。 `server task webseal server create -t transport type (ssl) or (tcp) -x -h hostname path`

例えば、次のように入力します。 `server task webseald-default create -t tcp -x -h monServer.ibm.com /BusinessSpace`
 - b. ご使用の製品用に以下のコンテキスト・ルートを作成します (『リバース・プロキシ・サーバーの Business Space URL のマッピング』を参照)。
- 5. ブラウザーの Cookie および仮想ホストに関する問題を解決するため、追加構成ステップを実行します。
 - a. Business Space の Cookie の名前変更を解決するため、以下の内容を WebSEAL 構成ファイルに追加します。

[preserve-cookie-names]

`name = com.ibm.bspace.UserName`

`name = com.ibm.wbimonitor.UserName`
 - b. オプション: デフォルト以外の仮想ホストをコンテキスト・ルートと一緒に使用する場合は、Business Space ページに関する問題が発生することがあります。その場合は、`-j` オプションをコンテキスト・ルートに追加して、Business Space ページの JavaScript™ がジャンクションで上書きされないようにすることが必要です。次のコマンドを実行します。 `server task default-webseald create -f -h hostname -p portnumber -t tcp -b supply -c iv-user,iv-creds,iv-groups -x -s -j -J trailer/root context`

Business Spaceのスーパーユーザー・ロールの割り当て

Business Spaceでは、スーパーユーザー (またはBusiness Space管理者) となるユーザーを割り当てることができます。スーパーユーザーは、すべてのスペースとページ

の表示、編集、削除のほか、テンプレートの管理と作成を行うことができます。また、所有者 ID を変更することでスペースの所有権を変更することもできます。

始める前に

Business Space を構成するときに管理セキュリティーを有効にする場合は、グループおよびスーパーユーザーに関する以下の情報について考慮してください。

- 特別なユーザー・グループ **administrators** に属するユーザーは、デフォルトでスーパーユーザー・ロールを持ちます。結果として、スーパーユーザー・ロールの割り当てはユーザー・グループのメンバーシップによって処理されます。
- シングル・サーバー環境の場合、Business Space・サーバーはデフォルトのユーザー・レジストリーに **administrators** ユーザー・グループを作成します。構成中に指定された管理者 ID は、自動的にこのグループのメンバーとして追加されません。
- ネットワーク・デプロイメント環境の場合、自動的に **administrators** ユーザー・グループが作成されることはありません。createSuperUser.py スクリプトを使用してユーザー・グループを作成し、デフォルトのユーザー・レジストリーでそのグループにメンバーを追加してください。
- デフォルトのユーザー・レジストリーではなく別のユーザー・レジストリー (例えば、LDAP など) を使用する場合、またはデフォルトのユーザー・レジストリーを使用するが **administrators** ユーザー・グループを使用したくない場合は、Business Space のスーパーユーザーに使用するユーザー・グループを特定する必要があります。必ず、ユーザー・レジストリーで認識される値を指定してください。例えば、LDAP の場合は、cn=administrators,dc=company,dc=com というように名前を指定します。このユーザー・グループを特定する方法については、「次の作業」で管理者グループの変更手順を参照してください。
- WebSphere Portal のBusiness Spaceの場合は、デフォルト・グループ **wpsadmins** もスーパーユーザー・ロールに使用されます。このグループのメンバーには、Business Space に対するスーパーユーザー・ロールが付与されます。

注: WebSphere Portal で Business Space を使用する場合は、セキュリティーを有効にする必要があります。

Business Space の構成時に管理セキュリティーを有効にしない場合、特別なユーザー ID **BPMAdministrator** のみに Business Space のスーパーユーザー・ロールが付与されます。

ネットワーク・デプロイメント環境を使用している場合は、createSuperUser.py スクリプトを実行してスーパーユーザー・ロールを割り当てる (ユーザー・グループを作成してメンバーを追加する) 必要があります。このスクリプトを実行する前に、以下のステップを実行します。

- デフォルトの **administrators** グループ名が変更されていないことを確認します。
- デフォルト・リポジトリーをユーザー・レジストリーに使用します。
- Business Spaceがインストールされているプロファイルについて、Business Space 環境用のサーバーまたはデプロイメント・マネージャーを始動します。

手順

1. ユーザーにスーパーユーザー・ロールを割り当てるためのスクリプト `install_root¥BusinessSpace¥scripts¥createSuperUser.py` を見つけます。
2. コマンド・プロンプトを開いてディレクトリー `profile_root¥bin` に移動します。この `profile_root` は、Business Space がインストールされているプロファイルのディレクトリーを表します。
3. 以下のコマンドを入力します。 `wsadmin -lang jython -f install_root¥BusinessSpace¥scripts¥createSuperUser.py user_short_name password` ここで、`user_short_name` は Virtual Member Manager (VMM) のユーザーの固有 ID、`password` はそのユーザーの VMM パスワードです。そのユーザーが VMM に存在する場合、そのユーザーが管理者グループに追加されます。

注: パスにスペースが含まれる場合 (例えば、`install_root` が `My install dir` である場合)、パス名を引用符で囲む必要があります。例えば、以下のコマンドを入力します。 `wsadmin -lang jython -f "¥My install dir¥BusinessSpace¥scripts¥createSuperUser.py" user_short_name_in_VMM`

次のタスク

Business Spaceを開くには、以下の URL を使用します。 `http://host:port/BusinessSpace` この `host` はサーバーが稼働しているホスト名で、`port` はサーバーのポート番号です。

デフォルトの特別なユーザー・グループ **administrators** は変更できます。現在のグループ名を確認したり別の名前に変更したりするには、以下の手順を実行します。

以下の構成ファイルで、測定基準 `com.ibm.mashups.adminGroupName` の値を検査します。

- スタンドアロン・サーバーの場合:
`profile_root¥BusinessSpace¥node_name¥server_name
¥mm.runtime.prof¥config¥ConfigService.properties`
- クラスターの場合:
`deployment_manager_profile_root¥BusinessSpace¥cluster_name
¥mm.runtime.prof¥config¥ConfigService.properties`

スタンドアロン・サーバー上で管理グループを変更する場合、以下のステップを実行します。

1. 構成ファイル `profile_root¥BusinessSpace¥node_name¥server_name
¥mm.runtime.prof¥config¥ConfigService.properties` の測定基準 `com.ibm.mashups.adminGroupName` を変更します。
2. プロファイルの `wsadmin` 環境で `updatePropertyConfig` コマンド `$AdminTask updatePropertyConfig {-serverName server_name -nodeName node_name -propertyFileName "profile_root¥BusinessSpace¥node_name¥server_name ¥mm.runtime.prof¥config¥ConfigService.properties" -prefix "Mashups_"}` を実行してから、`$AdminConfig save` を実行します。
3. サーバーを再始動します。

クラスター上で管理グループを変更する場合、以下のステップを実行します。

1. 構成ファイル `deployment_manager_profile_root¥BusinessSpace¥cluster_name ¥mm.runtime.prof¥config¥ConfigService.properties` の測定基準 `com.ibm.mashups.adminGroupName` を変更します。
2. デプロイメント環境プロファイルの `wsadmin` 環境で `updatePropertyConfig` コマンド `$AdminTask updatePropertyConfig {-clusterName cluster_name -propertyFileName "deployment_manager_profile_root¥BusinessSpace¥cluster_name ¥mm.runtime.prof¥config¥ConfigService.properties" -prefix "Mashups_"}` を実行してから、`$AdminConfig save` を実行します。
3. デプロイメント・マネージャーを再始動します。

スタンドアロン・サーバー上で、セキュリティが有効でないときにスーパーユーザーを変更する場合、以下のステップを使用します。

1. 構成ファイル `profile_root¥BusinessSpace¥node_name¥server_name ¥mm.runtime.prof¥config¥ConfigService.properties` の測定基準 `noSecurityAdminInternalUserOnly` を変更します。
2. プロファイルの `wsadmin` 環境で `updatePropertyConfig` コマンド `$AdminTask updatePropertyConfig {-serverName server_name -nodeName node_name -propertyFileName "profile_root¥BusinessSpace¥node_name¥server_name ¥mm.runtime.prof¥config¥ConfigService.properties" -prefix "Mashups_"}` を実行してから、`$AdminConfig save` を実行します。
3. サーバーを再始動します。

クラスター上で、セキュリティが有効でないときにスーパーユーザーを変更する場合、以下のステップを使用します。

1. 構成ファイル `deployment_manager_profile_root¥BusinessSpace¥cluster_name ¥mm.runtime.prof¥config¥ConfigService.properties` の測定基準 `noSecurityAdminInternalUserOnly` を変更します。
2. デプロイメント環境プロファイルの `wsadmin` 環境で `updatePropertyConfig` コマンド `$AdminTask updatePropertyConfig {-clusterName cluster_name -propertyFileName "deployment_manager_profile_root¥BusinessSpace¥cluster_name ¥mm.runtime.prof¥config¥ConfigService.properties" -prefix "Mashups_"}` を実行してから、`$AdminConfig save` を実行します。
3. デプロイメント・マネージャーを再始動します。

Business Space を構成するためのコマンド (wsadmin スクリプト)

スクリプト・オブジェクトまたはコマンド・クラスを検索して、そのコマンド構文の詳細を参照してください。

インフォメーション・センターの目次を開いて、この参照情報の場所を表示するに

は、インフォメーション・センターの枠にある「目次に表示」ボタン () をクリックします。

configureBusinessSpace コマンド

configureBusinessSpace コマンドを使用して、WebSphere が提供する Business Space のデータベースを構成します。

このコマンドは、Business Space のデータ・ソースを構成します。また、データベース表を作成および構成するスクリプトを生成します。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:
AdminConfig.save()
- Jacl の場合:
\$AdminConfig save

必須パラメーター

-serverName *server_name*

構成のサーバー名を指定するパラメーター。サーバーでBusiness Spaceを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-nodeName *node_name*

構成のノード名を指定するパラメーター。サーバーでBusiness Spaceを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-clusterName *cluster_name*

構成のクラスター名を指定するパラメーター。クラスターでBusiness Spaceを構成するには、**clusterName** を指定する必要があります。

オプション・パラメーター

-schemaName *schema_name*

Business Space データベースの構成に使用するデータベース・スキーマを指定するオプション・パラメーター。デフォルト値は **IBMBUSSP** です。

-tablespaceDir *table_space_path*

テーブル・スペースの物理的なロケーションとして使用されるファイルのディレクトリー・パスまたはファイル名接頭部を指定するオプション・パラメーター。デフォルト値は **BSP** です。DB2、Oracle、および SQL Server で有効です (その他の場合は無視されます)。SQL Server の場合、このパラメーターは 1 次データ・ファイルおよびログ・ファイルに適用されます。

-tablespaceNamePrefix *table_space_prefix*

名前を固有にするためにテーブル・スペース名の先頭に追加される接頭部ストリングを指定するオプション・パラメーター。デフォルト値は **BSP** です。テーブル・スペース名の接頭部が 4 文字より長い場合は、4 文字に切り捨てられます。DB2、DB2 z/OS V8、DB2 z/OS V9、および Oracle で有効です (その他の場合は無視されます)。

-dbLocationName *database_location_name*

z/OS 上のデータベース・ロケーション名を指定するオプション・パラメーター。デフォルト値は、**BSP** または製品データベース名です。DB2 z/OS V8 および V9 で有効です (その他の場合は無視されます)。

-storageGroup *storage_group*

Business Space に使用する z/OS 上のストレージ・グループを指定するオプション・パラメーター。z/OS を使用する場合は、生成されたデータベース・スクリプトを実行する前に、それらのスクリプトを更新する必要があります。スクリプトについて詳しくは、『Business Space のデータベース表の構成』を参照してください。

-bspacedbDesign *database_design_file_name*

スキーマおよびテーブル・スペース・ディレクトリーを含め、すべてのデータベース構成情報の定義に使用するデータベース設計ファイルを指定するオプション・パラメーター。 **-bspacedbDesign** パラメーターを使用してデータベース設計ファイルを指定する場合、特定のデータベース構成情報についてデータベース設計ファイルの内容を指定変更する必要がない限り、 **-schemaName** パラメーター、 **-tablespaceDir** パラメーター、および **-storageGroup** パラメーターを指定する必要はありません。

注: Business Spaceのデータ・ソースには JNDI 名として jdbc/mashupDS が常に使用されるため、データベース設計ファイル内の JNDI 名は使用されません。JNDI 名が jdbc/mashupDS のデータ・ソースが存在する場合、 **-replaceDatasource true** パラメーターを同時に指定した場合を除き、このコマンドはプロファイルを構成せずに停止します。

-productTypeForDatasource *product_database*

Business Spaceとともに使用するデータ・ソースの作成に使用するプロパティを指定するオプション・パラメーター。 **productTypeForDatasource** を指定すると、WebSphere Process Server、WebSphere Enterprise Service Bus、WebSphere Business Monitor、および WebSphere Business Compass などのインストール済み製品のデータ・ソースに基づいてモデル化した、JNDI 名が jdbc/mashupDS のBusiness Space用のデータソースが作成されます。有効な値は、WPS (WebSphere Process Server または WebSphere Enterprise Service Bus を指定する場合)、WPBS (WebSphere Business Compass を指定する場合)、および WBM (WebSphere Business Monitor を指定する場合) です。 **bspacedbDesign** パラメーターも指定した場合、 **productTypeForDatasource** はデータベース・タイプおよび JDBC プロバイダーを指定変更し、データベース設計ファイル内の JNDI 名は使用されません。

注: JNDI 名が jdbc/mashupDS のデータ・ソースが存在する場合、 **-replaceDatasource true** パラメーターを同時に指定した場合を除き、このコマンドはプロファイルを構成せずに停止します。

-replaceDatasource *true|false*

プロファイルが既に構成されている場合に **configureBusinessSpace** コマンドを実行するかどうかを指定するオプション・パラメーター。デフォルト値は **false** です。Business Space のプロファイルが構成される際には、JNDI 名が jdbc/mashupDS のデータ・ソースが作成されます。データ・ソースが存在し、 **-replaceDatasource true** を指定せずに **configureBusinessSpace** コマンドを実行した場合、このコマンドは構成を変更しません。 **true** を指定すると、このコマンドはデータ・ソースおよびその JDBC プロバイダーを削除し、これらを新しく作成して、新しい DDL スクリプトを作成します。

-save true/false

構成変更を保存することを示すパラメーター。デフォルト値は `false` です。

例

以下の例では、`configureBusinessSpace` コマンドを使用して、サーバーで `Business Space` データ・ソースを構成します。

- Jython の例:

```
AdminTask.configureBusinessSpace('[-nodeName myNode -serverName myServer]')
```

- Jacl の例:

```
$AdminTask configureBusinessSpace {-nodeName myNode -serverName myServer}
```

以下の例では、`configureBusinessSpace` を使用して、クラスター上で `Business Space` データ・ソースを構成して変更内容を保存します。

- Jython の例:

```
AdminTask.configureBusinessSpace('[-clusterName myCluster -save true]')
```

- Jacl の例:

```
$AdminTask configureBusinessSpace {-clusterName myCluster -save true}
```

以下の例では、`configureBusinessSpace` を使用して、`WebSphere Process Server` のスキーマ名と製品データ・ソースを指定して、クラスター上で `Business Space` データ・ソースを構成します。

- Jython の例:

```
AdminTask.configureBusinessSpace('[-clusterName myCluster -schemaName myCluster -productTypeForDatasource WPS -save true]')
```

- Jacl の例:

```
$AdminTask configureBusinessSpace {-clusterName myCluster -schemaName myCluster -productTypeForDatasource WPS -save true}
```

以下の例では、`configureBusinessSpace` を使用して、データベース設計ファイルに含まれるデータベース情報を使用して、クラスター上で `Business Space` データ・ソースを構成します。

- Jython の例:

```
AdminTask.configureBusinessSpace('[-clusterName myCluster -bspacedbDesign "C:%BSpace_dbDesign.properties" -save true]')
```

- Jacl の例:

```
$AdminTask configureBusinessSpace {-clusterName myCluster -bspacedbDesign "C:%BSpace_dbDesign.properties" -save true}
```

configureLotusWebformServer コマンド

`configureLotusWebformServer` コマンドを使用して、`IBM Lotus WebForm Server` を使用するよう `Business Space` を構成します。`Lotus Webform Server` はヒューマン・タスク管理ウィジェットを使用し、`WebSphere Process Server` サーバーおよびクラスター、ならびに `WebSphere Process Server` を含む `Business Process Management` 製品のあらゆるインストール済み環境に適用されます。

configureLotusWebformServer コマンドは、IBM Lotus WebForm Server がヒューマン・タスク管理ウィジェットを使用するよう Business Space を構成します。Webform Server は、スクリプトを実行するマシンと同じマシンにインストールされている必要があります。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:
AdminConfig.save()
- Jacl の場合:
\$AdminConfig save

必須パラメーター

-serverName *server_name*

構成のサーバー名を指定するパラメーター。サーバー上に Business Space ・ウィジェットを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-nodeName *node_name*

構成のノード名を指定するパラメーター。serverName、nodeName、または clusterName のいずれかが必須です。サーバーで構成するには、**serverName** および **nodeName** の両方を指定する必要があります。

-clusterName *cluster_name*

構成のクラスター名を指定するパラメーター。クラスターで Business Space を構成するには、**clusterName** を指定する必要があります。

-translatorHTTPLocation *URL*

Webform Server Translator の場所を指定するパラメーター。場所のデフォルト URL は `http://localhost:8085/translator` です。

-serverInstallRoot *Webform_Server_install_root*

Lotus Webform Server がインストールされている絶対パスを指定するパラメーター。例えば、Lotus Webform Server のインストール・ルートは、`C:/IBM/LotusWebForms/3.5/WebFormServer` である場合があります。

オプション・パラメーター

-save *true|false*

構成変更を保存することを示すパラメーター。デフォルト値は `true` です。

例

以下の例では、Business Space が Lotus Webform Server を使用してヒューマン・タスク管理ウィジェットと連携するように、configureLotusWebformServer を使用して Business Space を構成します。

- Jython の例:
AdminTask.configureLotusWebformServer(['-nodeName *node_name*
-serverName *server_name* -translatorHTTPLocation
`http://localhost:9080/translator` -serverInstallRoot
`C:/IBM/LotusWebForms/3.5/WebFormServer`'])
- Jacl の例:

```
$AdminTask configureLotusWebformServer {-nodeName node_name
-serverName server_name -translatorHTTPLocation
http://localhost:9080/translator
-serverInstallRoot C:/IBM/LotusWebForms/3.5/WebFormServer}
```

getBusinessSpaceDeployStatus コマンド

getBusinessSpaceDeployStatus コマンドを使用して、特定のデプロイメント・ターゲットで WebSphere が提供する Business Spaceが構成されているかどうかを確認します。

このコマンドは、指定したサーバー上、ノード上、またはクラスター上でBusiness Spaceが構成されているかどうかを検査します。パラメーターを設定しなかった場合は、Business Spaceがセルで構成されているかどうかを検査されます。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:
AdminConfig.save()
- Jacl の場合:
\$AdminConfig save

必須パラメーター

-serverName *server_name*

Business Spaceが構成されているかどうかを検査するサーバー名を指定するパラメーター。

-nodeName *node_name*

Business Spaceが構成されているかどうかを検査するノード名を指定するパラメーター。

-clusterName *cluster_name*

Business Spaceが構成されているかどうかを検査するクラスター名を指定するパラメーター。

例

以下の例では、getBusinessSpaceDeployStatus コマンドを使用して、サーバーで Business Space が構成されているかどうかを検査します。

- Jython の例:
AdminTask.getBusinessSpaceDeployStatus('[-nodeName myNode -serverName myServer]')
- Jacl の例:
\$AdminTask getBusinessSpaceDeployStatus {-nodeName myNode -serverName myServer}

以下の例では、getBusinessSpaceDeployStatus コマンドを使用して、クラスターで Business Space が構成されているかどうかを検査します。

- Jython の例:
AdminTask.getBusinessSpaceDeployStatus('[-clusterName myCluster]')
- Jacl の例:

```
$AdminTask getBusinessSpaceDeployStatus {-clusterName myCluster}
```

以下の例では、`getBusinessSpaceDeployStatus` コマンドを使用して、セル内の `Business Space` に対して構成されているすべてのデプロイメント・ターゲット (サーバーおよびクラスター) のリストを返します。

コマンドをプロファイル・ルートの `bin` ディレクトリーから実行すると、コマンドはセル内の `Business Space` に構成されたすべてのデプロイメント・ターゲット (サーバーおよびクラスター) のリストを返します。

コマンドをインストール・ルートの `bin` ディレクトリーから実行すると、コマンドは同じインストール・ルート・ディレクトリー内の `Business Space` に構成されたすべてのデプロイメント・ターゲット (サーバーおよびクラスター) のリストを返します。

- Jython の例:

```
AdminTask.getBusinessSpaceDeployStatus()
```

- Jacl の例:

```
$AdminTask getBusinessSpaceDeployStatus
```

installBusinessSpace コマンド

`installBusinessSpace` コマンドを使用して、ランタイム環境で WebSphere が提供する `Business Space` をセットアップします。

`installBusinessSpace` コマンドによって、`Business Space` のエンタープライズ・アーカイブ (EAR) ファイルがランタイム環境にインストールされます。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:

```
AdminConfig.save()
```

- Jacl の場合:

```
$AdminConfig save
```

必須パラメーター

-serverName *server_name*

構成のサーバー名を指定するパラメーター。サーバーで `Business Space` を構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-nodeName *node_name*

構成のノード名を指定するパラメーター。 `serverName`、`nodeName`、または `clusterName` のいずれかが必須です。サーバーで `Business Space` を構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-clusterName *cluster_name*

構成のクラスター名を指定するパラメーター。クラスターで `Business Space` を構成するには、**clusterName** を指定する必要があります。

オプション・パラメーター

-noWidgets true/false

オプション・パラメーターであり、true に設定すると、製品ウィジェットがデプロイメント・ターゲットにインストールされなくなります。そのため、ウィジェットをインストールしたい場合は、Business Space が正常に構成された後に `installBusinessSpaceWidgets` コマンドを使用する必要があります。デフォルト値は false です。

-save true/false

構成変更を保存することを示すオプション・パラメーター。デフォルト値は false です。

例

以下の例では、`installBusinessSpace` コマンドを使用して、サーバーに Business Space EAR ファイルをインストールします。

- Jython の例:

```
AdminTask.installBusinessSpace('[-nodeName myNode -serverName myServer -save true]')
```

- Jacl の例:

```
$AdminTask installBusinessSpace {-nodeName myNode -serverName myServer -save true}
```

以下の例では、`installBusinessSpace` を使用して、クラスターに Business Space EAR ファイルをインストールします。

- Jython の例:

```
AdminTask.installBusinessSpace('[-clusterName myCluster -save true]')
```

- Jacl の例:

```
$AdminTask installBusinessSpace {-clusterName myCluster -save true}
```

installBusinessSpaceWidgets コマンド

`installBusinessSpaceWidgets` コマンドを使用して、WebSphere が提供する Business Space で使用するウィジェットをインストール、デプロイ、および登録します。

`installBusinessSpaceWidgets` コマンドは、圧縮ファイルまたはエンタープライズ・アーカイブ (EAR) ファイルに含まれる指定のウィジェットをインストール、デプロイ、および登録します。ウィジェットが既にデプロイされている場合、`installBusinessSpaceWidgets` コマンドは、バイナリーおよび登録情報をリフレッシュします。

ウィジェット圧縮ファイルの構造には、以下の項目が含まれています。

- [ear¥widgets_name.ear] 1 つ以上の EAR ファイル。
- [catalog¥catalog_name.xml]
- [endpoints¥*.xml] ウィジェット・エンドポイント
- [templates¥*.zip] テンプレートは圧縮ファイルに格納され、IBM Lotus Mashups テンプレート形式に従う必要があります。
- [help¥eclipse¥plugins¥*]

すべてのフォルダーが含まれている必要はありません。空のフォルダーは有効です。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:
`AdminConfig.save()`
- Jacl の場合:
`$AdminConfig save`

必須パラメーター

-serverName *server_name*

構成のサーバー名を指定するパラメーター。サーバーでBusiness Spaceを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-nodeName *node_name*

構成のノード名を指定するパラメーター。 **serverName**、**nodeName**、または **clusterName** のいずれかが必須です。サーバー上にBusiness Space・ウィジェットを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-clusterName *cluster_name*

構成のクラスター名を指定するパラメーター。クラスターでBusiness Spaceのウィジェットを構成するには、**clusterName** を指定する必要があります。

-widgets *widgets_path*

以下のいずれかを指定するパラメーター。

- ウィジェットを含む圧縮ファイルまたは EAR ファイルが置かれているディレクトリーの絶対パス。ディレクトリーを指定すると、そのディレクトリー内のすべての圧縮ファイルおよび EAR ファイルに含まれているすべてのウィジェットがインストールされます。
- ウィジェットを含む個別の圧縮ファイルの絶対パス。
- ウィジェットを含む個別の EAR ファイルの絶対パス。

-save *true|false*

構成を保存することを示すパラメーター。デフォルト値は `true` です。

オプション・パラメーター

-save *true|false*

構成を保存することを示すパラメーター。デフォルト値は `true` です。

例

以下の例では、`installBusinessSpaceWidgets` を使用して、サーバーにウィジェットをインストール、デプロイ、および登録します。

- Jython の例:

```
AdminTask.installBusinessSpaceWidgets(['-nodeName node_name  
-serverName server_name -widgets  
install_root%BusinessSpace%widgets%MyWidget.zip'])
```
- Jacl の例:

```
$AdminTask installBusinessSpaceWidgets {-nodeName node_name
-serverName server_name -widgets
install_root%BusinessSpace%widgets%MyWidget.zip}
```

以下の例では、installBusinessSpaceWidgets を使用して、クラスターにウィジェットをインストール、デプロイ、および登録します。

- Jython の例:

```
AdminTask.installBusinessSpaceWidgets('[-clusterName cluster_name
-widgets X:%WPS%Temp]')
```

- Jacl の例:

```
$AdminTask installBusinessSpaceWidgets {-clusterName cluster_name
-widgets X:%WPS%Temp}
```

Business Space のテンプレートおよびスペースを更新するには、installBusinessSpaceWidgets または updateBusinessSpaceWidgets コマンドの実行後に手動ステップが必要です。詳しくは、『ウィジェットのインストールまたは更新後の Business Space テンプレートおよびスペースの更新』を参照してください。

registerRESTServiceEndpoint コマンド

registerRESTServiceEndpoint コマンドを使用して、構成されて使用可能な Representational State Transfer (REST) エンドポイントを登録して、チームが Business Space内でウィジェットを使用できるようにします。

このコマンドは REST サービス・エンドポイントを登録することで、Business Spaceが製品のウィジェットに正しく接続されるようにします。このコマンドは、Business Spaceと同じセル内にある REST サービスのエンドポイントを登録します。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:

```
AdminConfig.save()
```

- Jacl の場合:

```
$AdminConfig save
```

必須パラメーター

-clusterName *name_of_rest_services_cluster*

REST サービスのクラスター名を指定するパラメーター。クラスターの REST サービス・エンドポイントを登録するときには、**clusterName** を指定する必要があります。

-nodeName *name_of_rest_services_node*

REST サービスのノード名を指定するパラメーター。サーバーの REST サービス・エンドポイントを登録するときには、**serverName** および **nodeName** の両方を指定する必要があります。

-serverName *name_of_rest_services_server*

REST サービスのサーバー名を指定するパラメーター。サーバーの REST サービス・エンドポイントを登録するときには、**serverName** および **nodeName** の両方を指定する必要があります。

-type *name_of_service_type*

サービスのタイプ。このパラメーターはオプションです。このパラメーターを指定しない場合、指定されたデプロイメント・ターゲット上の指定された REST サービス・プロバイダーについて構成されているすべての固有の REST サービス・エンドポイントが登録されます。

-businessSpaceClusterName *name_of_business_space_cluster*

Business Space のクラスター名。Business Space がクラスター上に構成されている場合、**businessSpaceClusterName** を指定する必要があります。

-businessSpaceNodeName *name_of_business_space_node*

Business Space のノード名。Business Space がサーバー上に構成されている場合、**businessSpaceServerName** および **businessSpaceNodeName** の両方を指定する必要があります。

-businessSpaceServerName *name_of_business_space_server*

Business Space のサーバー名。Business Space がサーバー上に構成されている場合、**businessSpaceServerName** および **businessSpaceNodeName** の両方を指定する必要があります。

オプション・パラメーター

-appName *name_of_provider_application*

REST サービス・プロバイダーのアプリケーション名。

-webModuleName *name_of_web_module*

REST サービス・プロバイダーの Web モジュール名。

-version *name_of_version*

REST サービス・プロバイダーのバージョン。

例

以下の例では、registerRESTServiceEndpoint コマンドを使用します。ここでは、クラスター上で構成済みで使用可能なすべての REST サービスをBusiness Spaceに登録します。

- Jython の例:

```
AdminTask.registerRESTServiceEndpoint(['-clusterName  
name_of_rest_services_cluster -businessSpaceClusterName  
name_of_business_space_cluster'])
```

- Jacl の例:

```
$AdminTask registerRESTServiceEndpoint {-clusterName  
name_of_rest_services_cluster -businessSpaceClusterName  
name_of_business_space_cluster}
```

uninstallBusinessSpaceWidgets コマンド

uninstallBusinessSpaceWidgets コマンドを使用して、プロファイルからウィジェットおよびウィジェット定義を除去します。これには、個々のウィジェット資産 (アプリケーション、カタログ、エンドポイント、スペース、テンプレート、ヘルプ) の除去が含まれます。

`uninstallBusinessSpaceWidgets` コマンドは、指定された圧縮ファイル内またはエンタープライズ・アーカイブ (EAR) ファイル内のウィジェット・ファイルを除去します。ウィジェット圧縮ファイルの構造には、以下の項目が含まれています。

- `[ear¥widgets_name.ear]` 1 つ以上の EAR ファイル。
- `[catalog¥catalog_name.xml]`
- `[endpoints¥*.xml]` ウィジェット・エンドポイント
- `[templates¥*.zip]` テンプレートは圧縮ファイルに格納され、IBM Lotus Mashups テンプレート形式に従う必要があります。
- `[help¥eclipse¥plugins¥*]`

すべてのフォルダーが含まれている必要はありません。空のフォルダーは有効です。

注: `updateBusinessSpaceWidgets` コマンド以外の方法を使用して REST エンドポイント情報をカスタマイズした場合、エンドポイントに対するそれらの変更内容は、`uninstallBusinessSpaceWidgets` コマンドの実行後に失われます。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:
`AdminConfig.save()`
- Jacl の場合:
`$AdminConfig save`

必須パラメーター

-serverName *server_name*

構成のサーバー名を指定するパラメーター。サーバーで Business Space を構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-nodeName *node_name*

構成のノード名を指定するパラメーター。サーバーで Business Space を構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-clusterName *cluster_name*

構成のクラスター名を指定するパラメーター。クラスターで Business Space を構成するには、**clusterName** を指定する必要があります。

-widgets *widgets_path*

以下のいずれかを指定するパラメーター。

- ウィジェットを含む圧縮ファイルまたはウィジェット EAR ファイルが置かれているディレクトリーの絶対パス。ディレクトリーを指定すると、そのディレクトリー内のすべての圧縮ファイルおよび EAR ファイルに含まれているすべてのウィジェットがインストールされます。
- ウィジェットを含む個別の圧縮ファイルの絶対パス。
- ウィジェットを含む個別の EAR ファイルの絶対パス。

オプション・パラメーター

-save *true|false*

構成変更を保存することを示すパラメーター。デフォルト値は `true` です。

例

以下の例では、`uninstallBusinessSpaceWidgets` コマンドを使用して、クラスターからウィジェットを除去します。

注: これらの例は、説明のみを目的として記述されています。これらの例には変数値が含まれており、コード・スニペットとして再利用されることは想定していません。

- Jython の例:

```
AdminTask.uninstallBusinessSpaceWidgets('[-clusterName
cluster_name -widgets X:%WPS%Temp]')
```

- Jacl の例:

```
$AdminTask uninstallBusinessSpaceWidgets {-clusterName
cluster_name -widgets X:%WPS%Temp}
```

updateBusinessSpaceWidgets コマンド

`updateBusinessSpaceWidgets` コマンドを使用して、以前に構成した Business Space のウィジェットと、そのエンドポイント、カタログ、テンプレート、およびヘルプ・プラグインを更新します。

`updateBusinessSpaceWidgets` コマンドは、Business Space に以前インストールされて構成済みのウィジェットのウィジェット・バイナリー・ファイル、カタログ・ファイル、エンドポイント・ファイル、テンプレート、およびヘルプ・プラグインを更新します。

`updateBusinessSpaceWidgets` コマンドは、指定された圧縮ファイル内またはエンタープライズ・アーカイブ (EAR) ファイル内のウィジェット・ファイルを更新します。ウィジェット圧縮ファイルの構造には、以下の項目が含まれています。

- [ear%widgets_name.ear] 1 つ以上の EAR ファイル。
- [catalog%catalog_name.xml]
- [endpoints%*.xml] ウィジェット・エンドポイント
- [templates%*.zip] テンプレートは圧縮ファイルに格納され、IBM Lotus Mashups テンプレート形式に従う必要があります。
- [help%eclipse%plugins%*]

すべてのフォルダーが含まれている必要はありません。空のフォルダーは有効です。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:

```
AdminConfig.save()
```

- Jacl の場合:

```
$AdminConfig save
```

必須パラメーター

-serverName *server_name*

構成のサーバー名を指定するパラメーター。サーバー上にBusiness Space・ウィジェットを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-nodeName *node_name*

構成のノード名を指定するパラメーター。serverName、nodeName、またはclusterName のいずれかが必須です。サーバー上にBusiness Space・ウィジェットを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-clusterName *cluster_name*

構成のクラスター名を指定するパラメーター。クラスターでBusiness Spaceを構成するには、**clusterName** を指定する必要があります。

オプション・パラメーター

-widgets *widget_path*

ウィジェットのエンタープライズ・アーカイブ (EAR) ファイルまたはウィジェット圧縮ファイルが配置されているディレクトリーの絶対パス、または特定のEAR ファイルまたはウィジェット圧縮ファイルの絶対パスを指定するパラメーター。

-endpoints *endpoint_path*

ウィジェット・エンドポイント・ファイルが配置されているディレクトリーの絶対パス、または特定のエンドポイント・ファイルの絶対パスを指定するパラメーター。

-catalogs *catalog_path*

ウィジェット・カタログ・ファイルが格納されているディレクトリーの絶対パス、または特定のカタログ・ファイルの絶対パスを指定するパラメーター。

-templates *template_path*

ウィジェット・テンプレート・ファイルが格納されているディレクトリーの絶対パス、または特定のテンプレート・ファイルの絶対パスを指定するパラメーター。

-helpplugins *help_path*

ウィジェット・オンライン・ヘルプ・プラグイン・ファイルが格納されているディレクトリーの絶対パス、または特定のウィジェット・オンライン・ヘルプ・プラグイン・ファイルの絶対パスを指定するパラメーター。

-noWidgets *truelfalse*

ウィジェット圧縮ファイルに含まれているウィジェット EAR ファイルを更新しないことを指定します。

-noEndpoints *truelfalse*

ウィジェット圧縮ファイルに含まれている指定したエンドポイント・ファイルを更新しないことを指定します。

-noCatalogs *truelfalse*

ウィジェット圧縮ファイルに含まれているカタログ定義ファイルを更新しないことを指定します。

-noTemplates true/false

ウィジェット圧縮ファイルに含まれているテンプレートを更新しないことを指定します。

-noHelp true/false

ウィジェット圧縮ファイルに含まれているヘルプ・ファイルを更新しないことを指定します。

-save true/false

構成を保存することを示すパラメーター。デフォルト値は true です。

例

以下の例では、`updateBusinessSpaceWidgets` を使用して、クラスター上のウィジェットを更新します。

Jacl の例:

```
$AdminTask updateBusinessSpaceWidgets {-clusterName cluster_name
    -endpoints endpoint_path -widgets widget_path}
```

Jython の例:

```
AdminTask.updateBusinessSpaceWidgets(['-clusterName',
    cluster_name, '-endpoints', endpoint_path,
    '-widgets', widget_path])
```

以下の例では、`updateBusinessSpaceWidgets` を使用して、サーバー上のウィジェットを更新します。

Jacl の例:

```
$AdminTask updateBusinessSpaceWidgets {-nodeName node_name
    -serverName server_name -endpoints endpoint_path}
```

Jython の例:

```
AdminTask.updateBusinessSpaceWidgets(['-nodeName',
    node_name, '-serverName', server_name, '-endpoints',
    endpoint_path])
```

Business Space のテンプレートおよびスペースを更新するには、`installBusinessSpaceWidgets` または `updateBusinessSpaceWidgets` コマンドの実行後に手動ステップが必要です。詳しくは、『ウィジェットのインストールまたは更新後の Business Space テンプレートおよびスペースの更新』を参照してください。

updateRESTGatewayService コマンド

`updateRESTGatewayService` コマンドを使用して、Representational State Transfer (REST) サービスが構成されて使用可能になるように REST ゲートウェイ・サービスを更新します。

このコマンドは、REST サービスが構成されて使用可能になるように REST ゲートウェイ・サービスを更新します。REST サービスのデプロイメントは、スタンドアロン・サーバー・プロファイルで自動的に実行されます。他のタイプの構成では、「REST サービス」管理コンソール・ページまたは `updateRESTGatewayService` によって、Business Space内の製品のすべてのウィジェットに対して REST サービスを構成することができます。

注: WebSphere Process Server の場合、Business Process Choreographer および Human Task Manager の REST サービスは、Business Process Choreographer コンテナおよび Human Task Manager コンテナを構成するときに構成されます。

コマンドを使用した後、以下のいずれかのコマンドを実行して、変更をマスター構成に保存します。

- Jython の場合:
`AdminConfig.save()`
- Jacl の場合:
`$AdminConfig save`

必須パラメーター

-clusterName *cluster_name*

REST サービスのクラスター名を指定するパラメーター。クラスター上に REST サービスを構成するには、**clusterName** を指定する必要があります。

-nodeName *node_name*

REST サービスのノード名を指定するパラメーター。サーバー上で REST サービスを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-serverName *server_name*

REST サービスのサーバー名を指定するパラメーター。サーバー上で REST サービスを構成するには、**serverName** と **nodeName** の両方を指定する必要があります。

-enable **true** | **false**

REST サービスが使用可能かどうかを示します。有効な値には **true** または **false** があります。

オプション・パラメーター

-type *name_of_service_type*

REST サービスのタイプ。

-version *name_of_version*

REST サービスのバージョン。

例

以下の例では、`updateRESTGatewayService` コマンドを使用して、REST サービスが構成されて使用可能になるように REST ゲートウェイ・サービスを更新します。

- Jython の例:
`AdminTask.updateRESTGatewayService('[-nodeName node1 -serverName server1 -type "{com.ibm.bpm}TimeTable" -version 6.2.0.0 -enable true]')`
- Jacl の例:
`$AdminTask updateRESTGatewayService {-nodeName node1 -serverName server1 -type "{com.ibm.bpm}TimeTable" -version 6.2.0.0 -enable true}`

ウィジェットのインストールまたは更新後の Business Space テンプレートおよびスペースの更新

クラスター環境で `installBusinessSpaceWidgets` または `updateBusinessSpaceWidgets` コマンドを実行した後、Business Space のテンプレートおよびスペースを更新するための手動ステップを実行する必要があります。

始める前に

以前に `installBusinessSpaceWidgets` コマンドまたは `updateBusinessSpaceWidgets` コマンドを使用したことがある場合は、以下の追加ステップを実行する必要があります。

手順

1. Business Space がクラスター内で構成されている場合は、以下のステップを実行します。
 - a. 以下のようにして、`oobLoadedStatus.properties` ファイルのカスタム・プロファイルを識別します。
 - 1) デプロイメント・マネージャー・プロファイルで、
`deployment_manager_profile_root¥BusinessSpace¥cluster_name`
`¥mm.runtime.prof¥config¥ConfigService.properties` ファイルを開きます。
 - 2) `com.ibm.mashups.directory.templates` プロパティまたは `com.ibm.mashups.directory.spaces` プロパティでセル、ノード、およびサーバーの名前を探します。

例えば、`com.ibm.mashups.directory.templates = config/cells/Cell101/nodes/Node01/servers/Server1/mm/templates` の場合、カスタム・プロファイルは `Cell101` というセル名と `Node01` というノード名で探すことができます。
 - 3) セル、ノード、およびサーバーの名前を使用して、カスタム・プロファイルを探します。
 - b. カスタム・プロファイルで、
`custom_profile_root¥BusinessSpace¥cluster_name`
`¥mm.runtime.prof¥public¥oobLoadedStatus.properties` ファイルを開き、`importTemplates.txt` プロパティまたは `importSpaces.txt` プロパティを以下のように更新します。

```
importTemplates.txt=true
importSpaces.txt=true
```
 - c. カスタム・プロファイルを再同期します。
 - 1) 管理コンソールを開き、「システム管理」 → 「ノード」 をクリックします。
 - 2) 「完全な再同期」 をクリックします。
 - d. クラスターを再始動します。
2. Business Space が管理対象サーバー内で構成されている場合は、以下のステップを実行します。

- a. 管理対象サーバーが置かれているカスタム・プロファイルで、
`custom_profile_root¥BusinessSpace¥node_name¥server_name`
`¥mm.runtime.prof¥public¥oobLoadedStatus.properties` ファイルを開き、
`importTemplates.txt` プロパティまたは `importSpaces.txt` プロパティを以下のように更新します。

`importTemplates.txt=true`
`importSpaces.txt=true`
- b. カスタム・プロファイルを再同期します。
 - 1) 管理コンソールを開き、「システム管理」 → 「ノード」 をクリックします。
 - 2) 「完全な再同期」 をクリックします。
- c. サーバーを再始動します。

ビジネス・ルールおよびセレクトターの構成

ビジネス・ルールおよびセレクトターは、基準に基づいて処理の結果を変更できるため、ビジネス・プロセスに柔軟性が提供されます。ビジネス・ルールおよびセレクトターのコンポーネントを含むアプリケーションをインストールする前に、ビジネス・ルール動的リポジトリをインストールする必要があります。スタンドアロン・サーバー用または Network Deployment 用のビジネス・ルール動的リポジトリをインストールできます。

ビジネス・ルールとセレクトターの監査ログの構成

ビジネス・ルールとセレクトターの新規作成、変更、および削除を追跡するログの値として、デフォルト値とは異なる値を使用するように、サーバーを構成できます。構成を変更すると、サーバー上のリソースを効率的に使用できます。

始める前に

このタスクを実行するには、管理コンソールを使用している必要があります。

このタスクに必要なセキュリティ・ロール: セキュリティとロール・ベースの許可が有効になっている場合、このタスクを実行するには、コンフィギュレーターとしてログインする必要があります。

このタスクについて

サーバーを実稼働環境で一定期間実行した後で、ビジネス・ルールとセレクトターの監査ログのためにサーバーが使用しているデフォルト値を調整する必要があると判断することがあります。

ビジネス・ルールとセレクトターの監査ログを構成するには、以下のステップを実行します。

手順

1. 「サーバー」 > 「アプリケーション・サーバー」 > `servername` > 「ビジネス・ルール」 > 「ビジネス・ルールとセレクトター監査」 をクリックして、「ビジネス・ルールとセレクトター監査」 ページにナビゲートします。
2. 必要な変更のタイプに応じて、以下のいずれかを実行します。

変更のタイプ	アクション
即時	<ol style="list-style-type: none"> 1. 「ランタイム」タブを選択します。 2. 必要な変更を入力します。 3. オプション: 永続的に変更するには、「構成へのランタイム変更も保管」を選択して、変更内容をリポジトリにコピーします。 4. 変更を適用してから前のページに戻るには、「OK」をクリックします。変更の適用後もこのページを表示したままにするには、「適用」をクリックします。
遅延	<ol style="list-style-type: none"> 1. 「構成」タブを選択します。 2. 必要な変更を入力します。 3. 変更を適用してから前のページに戻るには、「OK」をクリックします。変更の適用後もこのページを表示したままにするには、「適用」をクリックします。 4. 変更内容を有効にするには、サーバーを再始動します。

タスクの結果

指定した属性が監査ログに適用されます。

注: WebSphere Application Server 6.1 でセキュリティーが有効に設定されている場合、サーバー・ユーザー ID の指定方法のために、ビジネス・ルールとセクター監査の構成の変更が必要なことがあります。サーバー・ユーザー ID にデフォルト値が使用されている場合は、ビジネス・ルールまたはセクターのインストール後にビジネス・ルールまたはセクターが含まれているアプリケーションが開始され、ビジネス・ルールまたはセクターを使用する監査アクションが実行されると、自動生成されたサーバー ID 値がユーザーの監査レコードに記録されます。インストール後のアプリケーションの開始、管理クライアント、または管理コンソールでのインポートまたはエクスポートによってビジネス・ルールまたはセクターの成果物に変更されると、監査可能なアクションが発生します。生成される値は、他の監査レコードで使用されているユーザー ID の形式と一致していないことがあり、より一貫性のある値が必要な場合があります。

サーバー ID を指定するには、「リポジトリに保管されたサーバー ID」を使用するオプションを選択します。これにより、ユーザー・リポジトリのユーザー ID が、サーバー・プロセスに関連付けられます。ビジネス・ルールまたはセクターの成果物がリポジトリにインストールされた後で、ビジネス・ルールまたはセクターを含むアプリケーションが開始され、ビジネス・ルールまたはセクターを使用する監査可能アクションが実行されると、監査レコードでこの ID が使用されます。

管理クライアント (ビジネス・ルール・マネージャーなど) による変更が行われる監査アクション、およびビジネス・ルール・グループのエクスポートやインポートな

どの管理アクションには、サーバー ID 値は影響しません。このようなアクションの場合、監査レコードでは認証済みユーザーが使用されます。

サーバー・ユーザー ID の変更についての詳細は、『アプリケーションと環境の保護』の各トピックと、WebSphere Application Server の WebSphere Application Server Network Deployment セキュリティーに関する資料を参照してください。

コマンドを使用したビジネス・ルールとセレクターの監査の構成

サーバーの実行中に特性を変更する必要がある場合には、コマンドを使用して、ビジネス・ルールとセレクターの監査を構成します。

始める前に

これらのコマンドは、サーバーのコマンド行環境から実行する必要があります。

このタスクに必要なセキュリティー・ロール: セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、コンフィギュレーターとしてログインする必要があります。

このタスクについて

ビジネス・ルールとセレクターを監査するサーバーの数を変更する必要があるにもかかわらず、関連するサーバーを再始動できない場合があります。コマンド行を使用して、サーバーの構成をバッチ・モードで自動化できます。以下のタスクは、コマンドを使用して 1 つのサーバーを構成する方法を示しています。

重要: これらの設定は、サーバーを再始動した場合には保存されません。これらのコマンドを入力した後で構成を保存するには、管理コンソールを使用する必要があります。「サーバー」>「サーバー・タイプ」>「WebSphere Application Server」>「*servername*」>「ビジネス・ルール」>「ビジネス・ルールとセレクター監査」>「ランタイム」または「サーバー」>「サーバー・タイプ」>「WebSphere Application Server」>「*servername*」>「セレクター」>「ビジネス・ルールとセレクター監査」>「ランタイム」を選択します。

コマンドを使用したビジネス・ルールとセレクターの監査を構成するには、以下のステップを実行します。

注: 以下のタスクでは、サーバー `server1` を構成します。サーバーの名前が `server1` でない場合は、以下の `server1` を、そのサーバーの名前で置き換えます。ステップ 3 以降のすべてのステップは、`jacl` スクリプトに組み込んで実行できます。

手順

1. 管理環境に入ります。

```
wsadmin
```

2. 監査ロギングを構成するか、既存の構成を変更するかを決定します。

タスク	コマンド
監査ロギングの構成	<pre>set mbean [\$AdminControl] queryNames *:* ,name=CustomizationAuditMBean,process=server1]</pre>

タスク	コマンド
監査ロギング構成の変更	set auditconfig [\$AdminConfig list AuditLog]

3. 該当するコマンドを入力します。

監査ロギングを構成または変更するコマンド

重要: 既存の構成を変更するコマンドを入力する場合は、変更内容を保存する必要があります。サーバーを再始動しないと、変更内容は有効になりません。

以下は、入力可能なコマンドです。

\$AdminControl invoke \$mbean getSeparateAuditLogEnabled

別個の監査ログへのロギングを実行するかどうかを判別するために使用します。

\$AdminControl invoke \$mbean setSystemOutAuditLogEnabled {boolean}

SystemOut.log ファイルへのロギングを使用可能または使用不可にするために使用します。Boolean には、true または false を指定できます。

\$AdminControl invoke \$mbean getSeparateAuditLogFileName

別個の監査ログのファイル名を判別するために使用します。

\$AdminControl invoke \$mbean setSeparateAuditLogFileName {filename}

新規のログ・ファイルの名前 (MyAudit.log など) を設定するために使用します。

\$AdminControl invoke \$mbean getSeparateAuditLogFileRolloverSize

監査ログのサイズを判別するために使用します。

\$AdminControl invoke \$mbean setSeparateAuditLogFileRolloverSize integer

システムが監査ログをヒストリー・ファイルへロール・オーバーする条件となる、監査ログのサイズを設定するために使用します。サイズはメガバイト単位です。

\$AdminControl invoke \$mbean getSeparateAuditLogFileMaxNumberOfBackupFiles

監査ログ・ヒストリー・ファイルの数を判別するために使用します。

\$AdminControl invoke setSeparateAuditLogFileMaxNumberOfBackupFiles integer

監査ログ・ヒストリー・ファイルの数を設定するために使用します。

\$AdminControl invoke \$mbean setSeparateAuditLogEnabled {boolean}

別個のログ・ファイルへのロギングを開始または停止するために使用します。Boolean には、true または false を指定できます。

\$AdminConfig showall \$auditconfig

現在の監査ログ構成を表示するために使用します。

\$AdminConfig modify \$auditconfig {{separateAuditLogEnabled true}}

別個の監査ログへのロギングを使用可能にするために使用します。

\$AdminConfig modify \$auditconfig {{systemOutAuditLogEnabled false}}

system.Out ファイルへの監査を使用不可にするために使用します。

```
$AdminConfig modify $auditconfig {{customAuditLog {{maxNumberOfBackupFiles 7} {rolloverSize 7}}}}
```

監査ログ・ヒストリー・ファイルの数と監査ログ・ファイルのサイズを変更するために使用します。

```
$AdminConfig modify $auditconfig {{customAuditLog {{fileName MyAudit.log}}}}
```

監査ログ・ファイルの名前を変更するために使用します。

```
$AdminConfig save
```

構成を保存するために使用します。

次のタスク

管理コンソールを開き、「サーバー」>「サーバー・タイプ」>「WebSphere Application Server」>「*servername*」>「ビジネス・ルール」>「ビジネス・ルールとセレクトター監査」>「ランタイム」、または「サーバー」>「サーバー・タイプ」>「WebSphere Application Server」>「*servername*」>「セレクトター」>「ビジネス・ルールとセレクトター監査」>「ランタイム」を選択して、変更を保存します。あるいは、`$AdminConfig save` と入力します。

注: WebSphere Application Server 6.1 でセキュリティーが有効に設定されている場合、サーバー・ユーザー ID の指定方法のために、ビジネス・ルールとセレクトター監査の構成の変更が必要なことがあります。サーバー・ユーザー ID にデフォルト値が使用されている場合は、ビジネス・ルールまたはセレクトターのインストール後にビジネス・ルールまたはセレクトターが含まれているアプリケーションが開始され、ビジネス・ルールまたはセレクトターを使用する監査アクションが実行されると、自動生成されたサーバー ID 値がユーザーの監査レコードに記録されます。インストール後のアプリケーションの開始、管理クライアント、または管理コンソールでのインポートまたはエクスポートによってビジネス・ルールまたはセレクトターの成果物に変更されると、監査可能なアクションが発生します。生成される値は、他の監査レコードで使用されているユーザー ID の形式と一致していないことがあり、より一貫性のある値が必要な場合があります。

サーバー ID を指定するには、「リポジトリに保管されたサーバー ID」を使用するオプションを選択します。これにより、ユーザー・リポジトリのユーザー ID が、サーバー・プロセスに関連付けられます。ビジネス・ルールまたはセレクトターの成果物がリポジトリにインストールされた後で、ビジネス・ルールまたはセレクトターを含むアプリケーションが開始され、ビジネス・ルールまたはセレクトターを使用する監査可能アクションが実行されると、監査レコードでこの ID が使用されます。

管理クライアント (ビジネス・ルール・マネージャーなど) による変更が行われる監査アクション、およびビジネス・ルール・グループのエクスポートやインポートなどの管理アクションには、サーバー ID 値は影響しません。このようなアクションの場合、監査レコードでは認証済みユーザーが使用されます。

サーバー・ユーザー ID の変更についての詳細は、『アプリケーションと環境の保護』の各トピックと、WebSphere Application Server の WebSphere Application Server Network Deployment セキュリティーに関する資料を参照してください。

ビジネス・ルール・マネージャーのインストールの考慮事項

ビジネス・ルール・マネージャーを分散環境で使用する場合、セル、ノード、クラスター概念、および実行時のパフォーマンスを最適にするビジネス・ルール・マネージャーのセットアップ方法を理解する必要があります。

アプリケーション・サーバーは、セル、ノード、およびサーバー概念に基づいて編成されます。スタンドアロン・サーバー構成では、1つのセルには1つのノードが含まれ、各ノードには1つのサーバーが含まれます。システム管理アプリケーションとユーザー・アプリケーションはすべて同じサーバーで稼働します。スタンドアロン・サーバー構成では、ビジネス・ルール・マネージャーを同じアプリケーション・サーバーにインストールできます。また、ビジネス・ルール・マネージャーにはデフォルトの URL からアクセスできます。

分散サーバー構成では、1つのセルに複数のノードが含まれ、各ノードに複数のアプリケーション・サーバーが含まれるように構成できます。各セルは、単一の管理ドメインを構成します。この構成では、ドメイン全体で中央管理、ワークロード管理、およびフェイルオーバー構成を使用できます。

分散サーバー構成でパフォーマンスを最適にするには、ビジネス・ルール・マネージャーを管理デプロイメント・ターゲット (ビジネス管理サービスが中央にホストされているセル内のアプリケーション・サーバー) にインストールします。通常、このサーバーは Common Event Infrastructure サービスをホストするサーバーです。

セル内では、すべてのサーバーが単一のビジネス・ルール・リポジトリを使用および共有します。ビジネス・ルール・リポジトリにアクセスすると、ビジネス・アプリケーションがどの場所にインストールされているかにかかわらず、すべての動的ビジネス・ルール成果物の定義にアクセスできます。

実行時にセル内のすべてのビジネス・ルールが使用できるこの中央ストレージがあることによって、セル内のどのアプリケーション・サーバーにもビジネス・ルール・マネージャーをデプロイできます。また、ビジネス・ルール・マネージャーは、セル内のすべてのビジネス・ルールの一貫性のあるビューを提供します。ただし、高可用性を考慮すると、ビジネス・ルール・マネージャーを管理デプロイメント・ターゲット (ビジネス管理サービスが中央でホストされているセル内の専用アプリケーション・サーバー) にデプロイすることを、システム管理者にお勧めします。管理デプロイメント・ターゲット・サーバーは、Common Event Infrastructure サービスやその他のビジネス管理アプリケーションがインストールされているのと同じサーバーです。この構成では、高可用性が必要な場合、管理デプロイメント・ターゲット・サーバーをクラスター化して、アプリケーション・ユーザーに拡張が容易なソリューションを提供できます。

管理コンソールを使用したビジネス・ルール・マネージャーのインストール

ビジネス・ルール・マネージャーをエンタープライズ・アプリケーションとして WebSphere Process Server にインストールすると、実行時にビジネス・ルールを管理できます。WebSphere Process Server 6.1 以降では、プロファイル管理ツールの「ビジネス・ルール・マネージャーの構成」ページのチェック・ボックスを選択して WebSphere Process Server プロファイルを作成する時点で、同時にビジネス・ルール・マネージャーをインストールできます。ビジネス・ルール・マネージャー

は、ほかの 3 つの方法を使用してインストールすることもできます。具体的には、管理コンソールの構成ページを使用する方法、オペレーティング・システムの JACL コマンドを使用する方法、Admin Tasks コマンドを使用する方法がありますが、Admin Tasks コマンドを使用する方法は、WebSphere Process Server 6.1 以降でのみ可能です。詳しくは、各インストール方法の個々のトピックを参照してください。

始める前に

このタスクに必要なセキュリティー・ロール: セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

このタスクについて

管理コンソールを使用してビジネス・ルール・マネージャーをインストールするには、以下のステップを実行します。

手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「サーバー」>「サーバー・タイプ」>「WebSphere Application Server」または「サーバー」>「クラスター」>「WebSphere Application Server クラスター」をクリックします。
3. サーバーまたはクラスター・ターゲットの名前を選択します。
4. 「構成」タブを含むページで、「Business Integration」の下にある「ビジネス・ルール」を展開して、「ビジネス・ルール・マネージャーの構成」をクリックします。
5. 「一般プロパティ」から、「ビジネス・ルール・マネージャーのインストール」チェック・ボックスを選択します。

注: ビジネス・ルール・マネージャーが既にインストールされている場合は、このチェック・ボックスがチェックされた状態になりますが、このページからビジネス・ルール・マネージャーをアンインストールすることはできないのでぼかし表示されます。ただし、アプリケーションのリストを表示し、このリストから手動でアンインストールできます。

6. 「コンテキスト・ルート」フィールドで、デフォルト・コンテキスト・ルートの /br を受け入れるか、またはビジネス・ルール・マネージャー URL のカスタム・コンテキスト・ルートを入力します。
7. 「OK」をクリックします。
8. 構成を保存します。

次のタスク

ナビゲーション・ペインで、「アプリケーション」>「アプリケーション・タイプ」>「WebSphere エンタープライズ・アプリケーション」をクリックし、「ビジネス・ルール・マネージャーの始動 (Start Business Rules Manager)」を選択します。

JACL コマンドを使用したビジネス・ルール・マネージャーのインストール

ビジネス・ルール・マネージャーをインストールするときに、管理コンソールの代わりとして JACL コマンドを使用できます。JACL コマンドを使用できるのは、WebSphere Process Server をインストールしてプロファイルを作成したときに、ビジネス・ルール・マネージャーをインストールしていなかった場合です。

始める前に

このタスクに必要なセキュリティ・ロール: セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターのロールに割り当てられているユーザー ID を使用して認証される必要があります。

このタスクについて

JACL コマンドを使用してビジネス・ルール・マネージャーをインストールするには、以下のステップを実行します。

手順

1. WebSphere Process Server が始動していることを確認します。
2. オペレーティング・システムのシェル環境またはコマンド・プロンプトを開き、`install_root/bin` ディレクトリーに移動します。
3. 以下のインストール・コマンドを実行します。 `wsadmin.sh -f ./installBRManager.jacl [-s servername -n nodename | -cl clustername] -ce cellname -r rootname`

ビジネス・ルール・マネージャーをインストールして、それを複数のターゲットにマップするには、以下のコマンドを実行します。 `wsadmin.sh -f installBRManager.jacl -m "{target1} {target2} ... {targetn}" -ce cellname -r rootname`

注: パラメーター「-m」（「複数 (multiple)」を意味する) を指定すると、ビジネス・ルール・マネージャーを複数のターゲットに一括でインストールしてマップできます。複数のターゲットを二重引用符で囲みます。

各部の意味は、次のとおりです。

servername

アプリケーション・サーバーの名前。

Network Deployment 構成内でクラスターが指定されていない場合には、引数のペア「-s servername」が必要です。欠落している場合、*servername* のデフォルト値は「server1」です。

nodename

インストール・ノードの名前。

Network Deployment 構成内でクラスターが指定されていない場合には、引数のペア「-n nodename」が必要です。

clustername

アプリケーションをインストールするクラスターの名前。

Network Deployment 構成内でサーバー名とノード名が指定されていない場合には、引数のペア「-cl *clustername*」が必要です。

注: ノードとサーバーを指定するか、またはクラスターを指定する必要があります。両方は指定しないでください。

cellname

インストール・セルの名前。

引数のペア「-ce *cellname*」はオプションです。

rootname

アプリケーション・ルート・ディレクトリの名前。

引数のペア「-r *rootname*」はオプションです。欠落している場合、*rootname* のデフォルト値は「/br」です。

target_i ビジネス・ルール・マネージャーをインストールしてマップするターゲット (*i* は 1、2、...、*n*)。

ターゲットは (-s *servername* および -n *nodename*) または -cl *clustername* のいずれかです。

重要: WebSphere Process Server が単一サーバー環境で構成されている場合、これらの引数のペアのすべてはオプションです。WebSphere Process Server が Network Deployment 環境用に構成されている場合、以下の引数ペアのいずれかが必要です。

- (-s *servername* および -n *nodename*)
- または -cl *clustername*
- または -m "{{target1} {target2} ... {targetn}}"

他の引数ペアはオプションです。

例

例: ビジネス・ルール・マネージャー・アプリケーションを以下のターゲットにマップするとします。

- クラスター「BofACluster」
- Web サーバー「RedirectorServer」およびノード「AIXNode01」
- アプリケーション・サーバー「LinuxServer」およびノード「LinuxNode02」

コンテキスト・ルートは「bofa/brm」です。

以下のようにコマンドを実行します。

```
install_root/bin/wsadmin -f installBRManager.jacl -m "{{-cl BofACluster}
{-n AIXNode01 -s RedirectorServer} {-s LinuxServer -n LinuxNode02}}" -r
bofa/brm
```

AdminTask コマンドを使用したビジネス・ルール・マネージャーのインストール

WebSphere Process Server 6.1 以降を使用して、Admin Task コマンドを使用してビジネス・ルール・マネージャーをインストールできます。管理コンソールや JACL コマンドを使用する場合と同様に WebSphere Process Server のインストール時にビジネス・ルール・マネージャーをインストールしなかった場合は、Admin Task コマンドを使用してください。

始める前に

このタスクに必要なセキュリティ・ロール: セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、管理者またはコンフィギュレーターとしてログインする必要があります。

このタスクについて

管理タスク・コマンドを使用してビジネス・ルール・マネージャーをインストールするには、以下のステップを実行します。

手順

1. WebSphere Process Server が始動していることを確認します。
2. コマンド・ウィンドウで WebSphere Process Server のホーム・ディレクトリーに移動して、ディレクトリー /bin に切り替えます。
3. wsadmin コマンドを実行して、wsadmin モードに入ります。
4. 以下のいずれかのコマンドを入力して、ビジネス・ルール・マネージャーをインストールします。

```
wsadmin> $AdminTask configBusinessRulesManager {-serverName <serverName>
-nodeName <nodeName> -contextRoot <contextRoot>}
```

または

```
wsadmin> $AdminTask configBusinessRulesManager {-clusterName
<clusterName> -contextRoot <contextRoot>}
```

ここで、

serverName

アプリケーション・サーバーの名前。

nodeName

インストール・ノードの名前。

clusterName

アプリケーションをインストールするクラスターの名前。

contextRoot

アプリケーションを起動するために使用するコンテキスト・ルート。デフォルト値は /br です。

5. wsadmin> \$AdminConfig save を実行して、構成を保存します。

ヒント: `$AdminTask help configBusinessRulesManager` を実行すると、そのパラメーターの詳細を知ることができます。

例

コンテキスト・ルート「br」を使用してサーバー「cvuServer」およびノード「cvuNode01」にビジネス・ルール・マネージャーをインストールするには、以下のコマンドを入力します。

```
wsadmin> $AdminTask configBusinessRulesManager {-serverName cvuServer  
-nodeName cvuNode01 -contextRoot br}
```

次に構成を保存するには、以下のように入力します。

```
wsadmin> $AdminConfig save
```

ビジネス・ルール・マネージャーのサーバー・セキュリティの構成

サーバーでセキュリティを使用する場合は、ビジネス・ルール・マネージャーを使用しているサーバーを構成する必要があります。サーバーでセキュリティを使用可能に設定しない場合は、追加の構成なしでビジネス・ルール・マネージャーを使用できます。

このタスクについて

異なるロールまたはユーザー ID がある場合、サーバーの構成時に管理セキュリティを設定する必要があります。サーバーのセキュリティを設定するには、以下のステップを実行します。

手順

1. ユーザー ID の作成時に各 ID にロールを割り当てることにより、ユーザー ID に管理セキュリティを設定します。各ユーザー ID を作成して、それをロール `BusinessRuleUser` にマップします。

ロールを設定するには、ビジネス・ルール・マネージャー・アプリケーションにナビゲートし（「アプリケーション」>「エンタープライズ・アプリケーション」）、ビジネス・ルール・マネージャー・アプリケーションを選択します。「ユーザー/グループ・マッピングへのセキュリティ・ロール」を選択し、`BusinessRuleUser` ロールを更新します。

`BusinessRuleUser` ロール以外に、`NoOne` と `AnyOne` という 2 つのロールが定義されています。`NoOne` は、直接アクセスすべきではないリソースを明示的に設定する場合に開発者が使用します。`AnyOne` は、WebSphere Process Server 環境の許可を取得する場合に Tivoli Access Manager が使用します。

注: 管理セキュリティが有効に設定された ND 環境において、ポート 908n (n は正の整数) 上でビジネス・ルール・マネージャーを実行する場合は、ポート「944(n+3) とホスト値 "*"」を構成する必要があります。該当するポートがない場合は、ビジネス・ルール・マネージャーを起動する前に手動で構成します。

2. Cookie を使用してセッションを追跡するように、セッション・トラッキング・メカニズムを設定します。
3. 最低でも、適切なセッション・タイムアウト値を必ず設定します。

ビジネス・ルール・マネージャーを正常に動作させるための Web ブラウザーの構成

サーバーは、ビジネス・ルール・マネージャーのインストール中に自動的にクライアントを構成しますが、ビジネス・ルール・マネージャーが正しく動作するように、Web ブラウザーが正しく構成されていることを確認する必要があります。

このタスクについて

ビジネス・ルール・マネージャー用の Web ブラウザーが正しく構成されていることを確認するには、以下のステップを実行します。

手順

1. Web ブラウザーでスクリプトが使用可能であることを確認します。

ビジネス・ルール・マネージャーが動作するには、スクリプトが機能する必要があります。

2. Cookie が使用可能であることを確認します。

ビジネス・ルール・マネージャーを使用する場合、必要に応じてセッションの追跡に Cookie が使用されます。このため、セッションを追跡できるように、ブラウザで Cookie を使用可能にします。Cookie を使用可能にする場合、必ずシステム管理者に連絡してください。

リレーションシップ・サービスの構成

製品をインストールした後で、リレーションシップ・サービス用の構成プロパティを設定する必要があります。

始める前に

このタスクに必要なセキュリティ・ロール: セキュリティーとロール・ベースの許可が有効になっている場合、このタスクを実行するには、コンフィギュレーターまたは管理者としてログインする必要があります。どの WebSphere セキュリティー・ロールでもこの構成を表示できます。

このタスクについて

リレーションシップ・サービスのデータ・ソース・プロパティと照会ブロック・サイズ・プロパティ (リレーションシップ・インスタンス・カウント) を設定するには、以下のステップを実行します。

手順

1. 管理コンソールが稼働していることを確認します。
2. ナビゲーション・ペインで、「統合アプリケーション」>「**Relationship Manager**」をクリックします。
3. 「リレーションシップ・サービス構成」をクリックします。

タブ付きの構成ページが表示され、現在インストールされているリレーションシップ・サービスの名前とバージョン (読み取り専用) が示されます。

4. 「照会ブロック・サイズ (リレーションシップ・インスタンスのカウンツ) (Query block size (relationship instance count))」フィールドに、リレーションシップ・サービスがリレーションシップ照会用に確保する最大キャッシュを指定します。この設定により、照会結果セットのサイズが決定します。デフォルトでは、5000 個のリレーションシップ・インスタンスが一度に読み取られます。このフィールドによってサーバー・サイズ・メモリー使用量が制御されるため、管理者は、任意の照会が消費できるメモリー・リソースの量を一定の範囲内で制御できるようになります。
5. 「データ・ソース」フィールドで、セル・レベルで定義されたデータ・ソースの Java Naming and Directory Interface (JNDI) 名を入力して、リレーションシップ・サービスのデフォルトのデータ・ソースを指定します。これは、リレーションシップ・サービス用のテーブルが保管される場所です。デフォルトでは、各リレーションシップ関連スキーマはこのデータ・ソース内に作成されます。
6. 以下のオプションがあります。
 - 変更内容を保存し、前のページに戻る場合は、「OK」をクリックします。
 - 変更内容をクリアし、現在構成されている値または最後に保存された値を復元する場合は、「リセット」をクリックします。
 - そのページで保管されていない変更内容をすべて破棄し、前のページに戻る場合は、「キャンセル」をクリックします。

リレーションシップ機能に必要な DB2 セットアップ

WebSphere Process Server リレーションシップ機能を使用可能にするには、以下の DB2 構成ステップが必要です。

リレーションシップ機能には、DB2 ストアード・プロシージャのカスタマイズが必要です。以下のステップは、この手順の概要です。

1. REXX が DB2 で使用可能であることを確認します。
2. DSNTPSMP を使用可能にします。
3. WLM 環境名を WLM に定義します。
4. WLM 環境名を DB2 に定義します。
5. WLM JCL を作成します。

これらのステップについて詳しくは、「DB2 for z/OS アプリケーション・プログラミングおよび SQL ガイド」の『SQL プロシージャ・プロセッサ (DSNTPSMP) 用の環境のセットアップ』を参照してください。

DSNTPSMP/WPS セットアップを実行する前に、使用している ID が DSNTPSMP を実行する権限を保有していることを確認してください。「DB2 for z/OS アプリケーション・プログラミングおよび SQL ガイド」の『アプリケーション内での SQL プロシージャ・プロセッサ (DSNTPSMP) の呼び出し』の概要を参照してください。

また、CFGTPSMP を使用して、DSNTPSMP を優先させることもできます。このファイルに設定できるすべてのオプションおよび設定方法について詳しくは、DSN.SDSNCLST(DSNTPSMP) の DSNTPSMP CLIST コメントを参照してください。

ヒント: CFGTPSMP を参照してください。CREATE PROCEDURE をデバッグする際は、DSNTPSMP_TRACELEVEL を MEDIUM に変更してください。

拡張メッセージング・リソースの構成

管理コンソールを使用して、拡張メッセージング・サービスおよびこのサービスを使用するアプリケーションに必要なリソースを構成します。拡張メッセージング・サービスを使用可能にして、遅延応答に対応するためリスナー・ポートの拡張を構成し、拡張メッセージングを使用するアプリケーションに対して入力および出力ポートを追加または変更します。

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

拡張メッセージングはコンテナ管理メッセージングを使用可能にします。この機能は、既存のコンテナ管理パーシスタンスおよびトランザクションの振る舞いを使用するように、基本の Java Message Service (JMS) サポート、Enterprise Java Bean (EJB) コンポーネント・モデル、EJB 2.0 メッセージ駆動型 Bean を拡張します。

拡張メッセージングは、Bean 管理メッセージング・インプリメンテーションを使用して JMS インターフェースを提供します。これにより、Bean 管理メッセージングと拡張メッセージングの両方で一貫した JMS サポートが使用されるようになります。JMS の使用法は簡単です。それは、JMS のサポートが拡張メッセージング・サービスによって管理されているためです。

拡張メッセージングについて詳しくは、WebSphere Business Integration Server Foundation インフォメーション・センターの以下の記事を参照してください。

- 拡張メッセージング: 概説
- アプリケーションにおける拡張メッセージングの使用

拡張メッセージング・サービスを使用可能にする

拡張メッセージング・サービスを使用可能にして、コンテナ管理メッセージング (拡張メッセージング) のランタイム・サポートを提供します。「拡張メッセージング・サービス」ページを使用して、アプリケーション・サーバーの始動時にこのサービスを自動的に開始するか、または手動で開始するかを指定します。

このタスクについて

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

このタスクに必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合は、このタスクを実行するために、管理者またはコンフィギュレーターとしてログインしている必要があります。

拡張メッセージング・サービスを使用可能にするには、以下のステップを実行します。

手順

1. 管理コンソールが稼働していることを確認します。
2. 「サーバー」 → 「サーバー・タイプ」 → 「WebSphere Application Server」 → 「*server_name*」 → 「拡張メッセージング・サービス」をクリックして、「拡張メッセージング・サービス」ページを表示します。
3. サーバー起動時に拡張メッセージング・サービスを自動的に開始できるようにする場合、「サーバー起動時にサービスを使用可能にする」チェック・ボックスを選択します。手動でサービスを開始する場合は、チェック・ボックスをクリアします。
4. 「OK」をクリックします。
5. プロンプトが表示されたら、コンソール・タスクバーの「保管」をクリックして、マスター・リポジトリに変更を保存します。
6. ご使用の構成で WebSphere MQSeries バインディング・トランスポートを使用している場合、MQ_INSTALL_ROOT 環境変数の値を次のように設定します。
 - a. 管理コンソールで、「環境」 → 「WebSphere 変数」をクリックします。
 - b. MQ_INSTALL_ROOT をクリックして、環境変数の構成ページを表示します。
 - c. 「値」フィールドで、デフォルト値 ($\${WAS_INSTALL_ROOT}/lib/WMQ$) を削除し、明示的なインストール・パス (Windows システムの場合は `D:/IBM/WebSphereMQ` など) に置換します。
 - d. 「OK」をクリックします。
7. 変更を有効にするには、アプリケーション・サーバーを停止して再始動します。

遅延応答を処理するためのリスナー・ポートの拡張の構成

リスナー・ポートに遅延応答を処理させるためには、拡張を構成して、ポートが応答をチェックする頻度、および応答を待つ長さを指定します。

このタスクについて

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

遅延応答は、送信側 Bean から送信されたメッセージに対する応答がメッセージング・インフラストラクチャーによって遅れた場合に発生します。その結果、アプリ

ケーションは応答を受け取ることができません。拡張メッセージングは、これらの遅延応答メッセージを取得して、アプリケーションが提供するメッセージ駆動型 Bean に渡し、遅延応答を処理します。

このタスクに必要なセキュリティ・ロール: セキュリティーおよびロール・ベースの許可が使用可能になっている場合は、このタスクを実行するために、管理者またはコンフィギュレーターとしてログインしている必要があります。

遅延応答を処理する、リスナー・ポートの拡張を作成し使用可能にするには、以下のステップを実行します。

手順

1. リスナー・ポートが定義済みおよび構成済みであること、および送信側 Bean を「**遅延応答を処理 (Handle late responses)**」オプションを使用可能にしてデプロイ済みであることを確認してください。

注: このオプションを使用可能にした送信側 Bean のデプロイについて詳しくは、WebSphere Business Integration Server Foundation インフォメーション・センターを参照してください。

2. 管理コンソールから、「**サーバー**」 → 「**サーバー・タイプ**」 → 「**WebSphere Application Server**」 → 「*server_name*」 → 「**拡張メッセージング・サービス**」 → 「**リスナー・ポートの拡張**」をクリックします。
3. 「リスナー・ポートの拡張」ページで、「**新規**」をクリックし、新規のリスナー・ポート拡張を作成します。
4. 「**新規リスナー・ポートの拡張 (New Listener Port Extension)**」ページで、「**使用可能**」チェック・ボックスを選択して、拡張および遅延応答の処理を使用可能にします。
5. 「**要求間隔**」フィールドで、デフォルト値を受け入れるか、または新規の値を指定して、リスナー・ポートが遅延応答をチェックする頻度を指定します。
6. 「**要求タイムアウト**」フィールドで、デフォルト値を受け入れるか、または新規の値を指定して、リスナー・ポートが遅延応答を待つ長さを指定します。リスナー・ポートは、指定されたタイムアウト値後に受信したあらゆる応答を廃棄します。
7. 「**リスナー・ポート**」ドロップダウン・メニューを使用して、拡張に使用するリスナー・ポートを指定します。
8. 「**OK**」をクリックします。
9. プロンプトが表示されたら、コンソール・タスクバーの「**保管**」をクリックして、マスター・リポジトリに変更を保存します。
10. 変更を有効にするには、アプリケーション・サーバーを停止して再始動します。

次のタスク

リスナー・ポートの拡張を作成した後、「リスナー・ポートの拡張」の設定ページを使用して、必要に応じて構成を変更できます。

拡張メッセージング・プロバイダーを選択する

管理する拡張メッセージング・プロバイダーを選択するには、「拡張メッセージング・プロバイダー」ページで適切なスコープをクリックします。拡張メッセージングを使用するアプリケーションを含む各スコープ (セル、ノード、およびサーバー) は、それぞれ独自の拡張メッセージング・プロバイダーを保有し、リソースを管理します。入力ポート、出力ポート、または各プロバイダーのその他のカスタム・プロパティを作成、変更または削除できます。

このタスクについて

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

このタスクに必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合は、このタスクを実行するために、管理者、オペレーター、コンフィギュレーター、またはモニターとしてログインしている必要があります。

管理する拡張メッセージング・プロバイダーを選択するには、以下のステップを実行します。

手順

1. 管理コンソールで、「リソース」>「拡張メッセージング・プロバイダー」をクリックします。
2. 「拡張メッセージング・プロバイダー」ページで、管理する拡張メッセージング・プロバイダーについて適切なスコープを選択します。
 - 「セル」: 最も汎用的なスコープです。セル・スコープに定義された拡張メッセージング・リソースは、オーバーライドされない限り、すべてのノードおよびサーバーから可視です。
 - 「ノード」: ノード・スコープに定義された拡張メッセージング・リソースは、セル・スコープに定義されたあらゆる重複する定義をオーバーライドします。そのノード上のサーバー・スコープでオーバーライドされていない限り、同じノードのすべてのサーバーから可視です。
 - 「サーバー」: サーバー・スコープに定義された拡張メッセージング・リソースは、セル・スコープ、または親ノード・スコープに定義されたあらゆる重複する定義をオーバーライドします。これらは、特定のサーバーからのみ可視です。

スコープについて詳しくは、WebSphere Application Server for z/OS インフォメーション・センターを参照してください。

3. 「適用」をクリックします。

タスクの結果

管理コンソールは、ページ下部の「スコープ」、「名前」、「説明」フィールドを更新し、選択したリソース・プロバイダーの値として反映させます。

次のタスク

入力ポート、出力ポート、または選択した拡張メッセージング・プロバイダーのその他のカスタム・プロパティを作成、変更または削除できます。

入力ポートの構成

管理コンソールを使用して、セッション Bean から作成された各受信側 Bean について、新規入力ポートを作成または既存入力ポートを変更します。入力ポートは、受信する Java Message Service (JMS) 宛先についてプロパティを定義し、メッセージの選択および処理方法を指定し、必要な応答宛先に関する詳細を提供します。

このタスクについて

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

メッセージ駆動型 Bean から作成された受信側 Bean について入力ポートを作成する必要はありません。必要な詳細は、デプロイ済みメッセージ駆動型 Bean およびメッセージ・リスナー・サービスに関連付けられています。

このタスクに必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合は、このタスクを実行するために、管理者またはコンフィギュレーターとしてログインしている必要があります。

入力ポートを追加または変更するには、以下のステップを実行します。

手順

1. 管理コンソールで、「リソース」>「拡張メッセージング・プロバイダー」をクリックします。
2. 「拡張メッセージング・プロバイダー」ページで、作業するリソース・プロバイダーについて適切な範囲を選択します。
3. 「適用」をクリックします。
4. 「追加プロパティ」テーブルから「入力ポート」をクリックします。
5. 「入力ポート」のコレクション・ページで、以下のいずれかを行います。
 - 新規の入力ポートを作成する場合は、「新規」をクリックします。
 - 既存の入力ポートを変更する場合は、ポート名をクリックします。
6. 「入力ポート」の設定ページで、入力ポートに適切なプロパティを指定します。
7. 「OK」をクリックします。

8. プロンプトが表示されたら、コンソール・タスクバーの「保管」をクリックして、マスター・リポジトリに変更を保存します。
9. 変更を有効にするには、アプリケーション・サーバーを停止して再始動します。

入力ポート設定:

新規の入力ポートを作成、または既存の入力ポートを変更する場合、特定のプロパティを指定する必要があります。このトピックの情報を使用して、オプションまたは必須のプロパティ、受け入れるデータ・タイプを決定してください。

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

入力ポートには以下の構成プロパティがあります。

有効範囲

拡張メッセージング・プロバイダーが定義される範囲。この値は、構成ファイルの場所を表します。管理コンソールは自動的にこのフィールドを取り込みます。値を編集することはできません。

名前 管理目的で使用される入力ポートの名前。このフィールドにはストリング値が必要です。

JNDI 名

入力ポートの Java Naming and Directory Interface (JNDI) 名。このフィールドにはストリング値が必要です。

説明 管理目的で使用される入力ポートの説明。このフィールドはオプションです。ストリング値が入力できます。

カテゴリ

リソースの分類またはグループ化を行うために使用するカテゴリ・ストリング。このフィールドはオプションです。ASCII 文字で最大 30 文字のストリング値が入力できます。

JMS 接続ファクトリー JNDI 名

入力ポートが使用している Java Message Service (JMS) 接続ファクトリーの JNDI 名。このフィールドにはストリング値 (jms/connFactory1 など) が必要です。

JMS 宛先 JNDI 名

入力ポートが使用している JMS 宛先の JNDI 名。このフィールドにはストリング値 (jms/destn1 など) が必要です。

JMS 肯定応答モード

メッセージの応答に使用する JMS モード。このフィールドは Bean 管理トランザクション区分を使用するメッセージ駆動型 Bean (言い換えると、トランザクション・タイプを Bean に設定) に必須です。

このフィールドの有効な値は以下のとおりです。

- **Auto Acknowledge:** 以下のいずれかの場合、セッションは自動的にメッセージに応答します。
 - セッションが、呼び出しから正常に戻ってメッセージを受信する場合
 - セッションがメッセージ・リスナーを呼び出してメッセージを処理し、そのリスナーから正常な応答を受け取った場合
- **Dups OK Acknowledge:** セッションはメッセージの送達にのみ応答します。このため、JMS が失敗すると、重複メッセージが送達されることとなります。

デフォルト・モードは **Auto Acknowledge** です。

宛先タイプ (Destination Type)

JMS リソース・タイプ。このフィールドには以下のいずれかの値が必要です。

- **Queue:** 受信側 Bean はキュー宛先からメッセージを受信します。
- **Topic:** 受信側 Bean はトピック宛先からメッセージを受信します。

デフォルト値は **Queue** です。

サブスクリプション永続性

JMS トピック・サブスクリプションに耐久性があるかどうかを指定します。このフィールドは、JMS 宛先タイプがトピックの場合、必須です。このフィールドの有効な値は以下のとおりです。

- **Durable:** サブスクライバーは、JMS によって保存される固有の ID を持つ永続サブスクリプションを登録します。ID が同じ後続のサブスクライバー・オブジェクトは、前のサブスクライバーが残した状態でサブスクリプションを再開します。永続サブスクリプションのアクティブ・サブスクライバーがない場合、JMS は、メッセージを受信するかメッセージが期限切れになるまで、サブスクリプションのメッセージを保存します。
- **NonDurable:** 非永続サブスクリプションは、所有するサブスクライバーの存続時間中は持続します。クライアントは、サブスクライバーがアクティブの間のみ、トピックで公開されたメッセージを確認します。サブスクライバーが非アクティブの場合、クライアントはそのトピックで公開されたメッセージを確認できません。

デフォルト値は **NonDurable** です。

応答 JMS 接続ファクトリー JNDI 名

応答で使用する JMS 接続ファクトリーの JNDI 名。このフィールドにはストリング値 (`.jms/connFactory1` など) が必要です。

応答 JMS 宛先 JNDI 名

応答で使用する JMS 宛先の JNDI 名。このフィールドにはストリング値 (`.jms/destn1` など) が必要です。

出力ポートの構成

管理コンソールを使用して、送信側 Bean について、新規出力ポートを作成または既存出力ポートを変更します。出力ポートは、送信されたメッセージの宛先を定義するために必要な送信側 Bean のプロパティを指定します。また、応答が期待される場合のオプションのプロパティも指定します。出力ポートは、デプロイメント時に送信側 Bean に関連付けられます。

このタスクについて

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

このタスクに必要なセキュリティ・ロール: セキュリティおよびロール・ベースの許可が使用可能になっている場合は、このタスクを実行するために、管理者またはコンフィギュレーターとしてログインしている必要があります。

出力ポートを追加または変更するには、以下のステップを実行します。

手順

1. 管理コンソールで、「リソース」>「拡張メッセージング・プロバイダー」をクリックします。
2. 「拡張メッセージング・プロバイダー」ページで、変更する拡張メッセージング・プロバイダーについて適切な範囲を選択します。
3. 「適用」をクリックします。
4. 「追加プロパティ」テーブルから「出力ポート」をクリックします。
5. 「出力ポート」のコレクション・ページで、以下のいずれかを行います。
 - 新規の出力ポートを追加する場合は、「新規」をクリックします。
 - 既存の出力ポートを変更する場合は、ポート名をクリックします。
6. 「出力ポート」の設定ページで、出力ポートに適切なプロパティを指定します。
7. 「OK」をクリックします。
8. プロンプトが表示されたら、コンソール・タスクバーの「保管」をクリックして、マスター・リポジトリに変更を保存します。
9. 変更を有効にするには、アプリケーション・サーバーを停止して再始動します。

出力ポート設定:

新規の出力ポートを作成、または既存の出力ポートを変更する場合、特定のプロパティを指定する必要があります。このトピックの情報を使用して、オプションまたは必須のプロパティ、受け入れるデータ・タイプを決定してください。

重要: 拡張メッセージング・サービス機能は、WebSphere Process Server 6.0.1 から非推奨になっています。WebSphere Process Server 6.2 以降は、アプリケーションで使用することはできません (マイグレーション中のセル内の 6.0.x ノードを管理する場合は除く)。拡張メッセージング・サービスに依存する既存のアプリケーションは、標準の JMS API を使用するアプリケーションに置き換えるか、同じ機能を備えたメッセージング製品に置き換えてください。

出力ポートには以下の構成プロパティがあります。

有効範囲

拡張メッセージング・プロバイダーの範囲。この値は、構成ファイルの場所を表します。管理コンソールは自動的にこのフィールドを取り込みます。値を編集することはできません。

名前 管理目的で使用される出力ポートの名前。このフィールドにはストリング値が必要です。

JNDI 名

出力ポートの Java Naming and Directory Interface (JNDI) 名。このフィールドにはストリング値が必要です。

説明 管理目的で使用される出力ポートの説明。このフィールドはオプションです。ストリング値が入力できます。

カテゴリ

リソースの分類またはグループ化を行うために使用するカテゴリ・ストリング。このフィールドはオプションです。ASCII 文字で最大 30 文字のストリング値が入力できます。

JMS 接続ファクトリー JNDI 名

出力ポートが使用している Java Message Service (JMS) 接続ファクトリーの JNDI 名。このフィールドにはストリング値 (jms/connFactory1 など) が必要です。

JMS 宛先 JNDI 名

出力ポートが使用している JMS 宛先の JNDI 名。このフィールドにはストリング値 (jms/destn1 など) が必要です。

JMS 送達モード

メッセージの送達に使用する JMS モード。このフィールドには以下の値のいずれかを選択する必要があります。

- Persistent: 宛先上のメッセージは永続的です。
- Nonpersistent: 宛先上のメッセージは永続的ではありません。

デフォルト値は Persistent です。

JMS 優先順位

キュー宛先のメッセージ優先順位。このフィールドでは、0 から 9 までの整数値が必要です。デフォルト値は 4 です。

JMS 存続時間

メッセージがキュー内に存続する時間 (ミリ秒)。指定された時間が経過すると、メッセージの有効期限が切れます。

このフィールドでは、0 から n までの整数値が必要です。

- 0: メッセージはタイムアウトになりません。
- n : メッセージは n ミリ秒後にタイムアウトになります。

デフォルト値は 0 です。

JMS 使用不可メッセージ ID

JMS メッセージ ID をシステムが生成するかどうかを指定。このフィールドは必須フィールドです。以下のいずれかの値を選択します。

- 選択: JMS メッセージ ID は生成されません。

- クリア: JMS メッセージ ID が自動的に生成されます。

デフォルトでは、JMS message ID は生成されます。

JMS メッセージ・タイム・スタンプ使用不可

JMS メッセージ・タイム・スタンプが生成されるかどうかを指定します。このフィールドは必須フィールドです。以下のいずれかの値を選択します。

- 選択: 送信メッセージにメッセージ・タイム・スタンプは追加されません。
- クリア: 送信メッセージにメッセージ・タイム・スタンプは自動的に追加されます。

デフォルトでは、送信メッセージにメッセージ・タイム・スタンプは追加されます。

応答 JMS 接続ファクトリー JNDI 名

出力ポートで処理する応答で使用する JMS 接続ファクトリーの JNDI 名。このフィールドにはストリング値 (jms/connFactory1 など) が必要です。

応答 JMS 宛先 JNDI 名

出力ポートで処理する応答で使用する JMS 宛先の JNDI 名。このフィールドにはストリング値 (jms/destn1 など) が必要です。

メッセージング・サーバー環境のセットアップ

XMS アプリケーション (XMS に付属のサンプル・アプリケーションを含む) を実行する前に、メッセージング・サーバー環境をセットアップする必要があります。

このタスクについて

メッセージング・サーバー環境をセットアップするために実行すべき手順は、アプリケーションが接続する成果物と、Message Service Client for .NET か Message Service Client for C/C++ のどちらを使用しているかによって異なります。これらの手順については、当該タイプのクライアントに関する資料で説明されています。

手順

- Message Service Client for .NET のセットアップ
- Message Service Client for C/C++ のセットアップ

次のタスク

メッセージ・サービス・クライアントに付属のサンプル・アプリケーションを使用すると、インストールおよびメッセージング・サーバーのセットアップを検証することができます。サンプル・アプリケーションの使用については、以下のトピックを参照してください。

- .NET のサンプル XMS アプリケーションの使用
- C/C++ のサンプル XMS アプリケーションの使用

JNDILookup Web Service の構成

Message Service Client for C/C++ および Message Service Client for .NET を備えた WebSphere Process Server で提供された管理 JMS オブジェクトを使用する場合、WebSphere Process Server が提供する JNDILookup Web Service を、非 Java クライアントが非 Java 環境から管理 JMS オブジェクトにアクセスできるように構成する必要があります。

始める前に

このタスクを開始する前に、JNDILookup Web サービス・アプリケーションがインストールされていることを確認してください。

このタスクについて

管理上定義された ConnectionFactory オブジェクトおよび Destination オブジェクトによって、JMS 実装と JMS インターフェースが分離されます。これにより、JMS クライアント・アプリケーションが JMS プロバイダーの実装詳細から隔離されるため、移植性が高まります。管理対象オブジェクトを使用すると、管理者は中央リポジトリからクライアント・アプリケーションの接続設定を管理できるようになります。例えば、アプリケーションが JNDI を通して取得する管理対象の Destination オブジェクトを変更することで、アプリケーションが使用する特定のキューを変更できます。

Message Service Client for C/C++ および Message Service Client for .NET などの非 Java クライアントも管理対象オブジェクトを使用できます。ただし、WebSphere Process Server によって提供される管理対象の JMS オブジェクトは、JNDI を通してアクセスされる直列化 Java オブジェクトなので、非 Java クライアントは JNDILookup Web サービスを使用しないとこれらのオブジェクトを正しく解釈できません。この Web サービスによって提供されるルックアップ・オペレーションを使用すると、Message Service Client for C/C++ および Message Service Client for .NET がオブジェクト名を指定することで JNDI オブジェクトの取得を要求できるようになります。管理対象オブジェクトのプロパティは、名前/値ペアのマップを使用してアプリケーションに返されます。

手順

Message Service Client for C/C++ または Message Service Client for .NET アプリケーション内で JNDILookup Web サービス URL を定義します。アプリケーション内で Web サービス URL を定義するには、InitialContext オブジェクトの XMSC_IC_URL プロパティを Web サービスのエンドポイント URL に設定します。このプロパティは、InitialContext オブジェクトの構成時に引数として指定することも可能です。

Common Event Infrastructure の構成

サーバー AdminTask オブジェクトを使用して、Common Event Infrastructure リソースを構成できるほか、既存のリソースを変更することもできます。

このタスクについて

Common Event Infrastructure (CEI) は、デフォルト構成を使用してインストールすることができます。このデフォルト構成は、スタンドアロン・サーバー構成で完全に機能する構成です。このタスクは、スタンドアロン・サーバーを作成する場合にのみ実行します。これ以外の場合 (Network Deployment 環境またはクラスターにインストールする場合など) は、管理コンソールを使用して CEI を構成し、システムに適した構成となるようにします。

また、wsadmin コマンドを使用して CEI を構成したり、このコマンドを使用して既存の CEI 構成を変更することもできます。いずれの場合も、サーバー AdminTask オブジェクトを使用して CEI の構成を変更し、管理コマンドを実行します。

CEI 構成の変更後に、サーバーまたはクラスターを再始動する必要があります。

Common Event Infrastructure コンポーネント

Common Event Infrastructure コンポーネントは、一連のアプリケーション、サービス、およびリソースとしてサーバー上にインストールされます。

Common Event Infrastructure を構成すると、いくつかのコンポーネントが作成され、サーバー上にデプロイされます。

Common Event Infrastructure サービス

サーバーにインストールされるサービスで、アプリケーションおよびクライアントが Common Event Infrastructure を使用できるようにします。

Common Event Infrastructure サービスの構成は、次のようにして管理コンソールに表示できます。

- サーバーの場合は、「サーバー」>「アプリケーション・サーバー」>「*server_name*」>「ビジネス・インテグレーション」>「Common Event Infrastructure」>「Common Event Infrastructure サービス」を選択します。
- クラスターの場合は、「サーバー」>「クラスター」>「*cluster_name*」>「ビジネス・インテグレーション」>「Common Event Infrastructure」>「Common Event Infrastructure サービス」を選択します。

「イベント・インフラストラクチャー・サーバーを使用可能に設定」というチェック・ボックスが選択されている場合、サービスはインストール済みで稼働しているか、サーバーまたはクラスターを再始動すると始動します。このチェック・ボックスがクリアされている場合、サービスはインストールされていないか、サーバーまたはクラスターを再始動するとアンインストールされます。

イベント・サービス設定

イベント・サービスが使用する一連のプロパティ。これらのプロパティにより、データ・ストアを使用して、イベントの配布および永続化が可能になります。通常、このリソースに構成は必要ありませんが、同じセルに複数のイベント・サービスをセットアップする場合は、追加のイベント・サービ

ス設定を作成することが必要な場合があります。イベント・サービス設定を表示するには、「サービス統合」>「イベント・サービス」>「イベント・サービス設定」をクリックします。

イベント・メッセージング構成

Java Messaging Service (JMS) を使用して、イベントをイベント・サービスに非同期的に転送することが可能なリソース。デフォルトのメッセージング構成では、サーバー組み込みメッセージが使用されます。必要な場合は、イベント・メッセージング用に外部の JMS プロバイダーを構成することもできます。

イベント・データベース

イベント・データベースは、イベント・サービスが受け取ったイベントを永続的に保管するために使用されます。Derby データベースは、サーバーの一部として組み込まれていますが、実稼働環境での使用には推奨されません。その代わりとして、DB2、Oracle、SQLServer、および Informix® 製品で外部イベント・データベースを構成できます。

イベント・フィルター・プラグイン

フィルター・プラグインは、XPath イベント・セレクターを使用して、ソースのイベントをフィルターに掛けるために使用されます。フィルター・プロパティを構成するには、「サービス統合」>「Common Event Infrastructure」>「イベント・エミッター・ファクトリー」>「イベント・フィルター設定」をクリックします。

エミッター・ファクトリー

エミッター・ファクトリーは、エミッターを作成するためにイベント・ソースで使用されるオブジェクトです。エミッターは、イベントをイベント・サービスに送信するために使用されます。エミッター・ファクトリーのプロパティは、そのエミッター・ファクトリーを使用して作成したすべてのエミッターの動作に影響を与えます。フィルター・プロパティを構成するには、「サービス統合」>「Common Event Infrastructure」>「イベント・エミッター・ファクトリー」をクリックします。

イベント・サービス伝送

イベント・サービス伝送は、エミッターが EJB 呼び出しを使用してイベント・サービスに同期的にアクセスする方法を決定するプロパティを定義するオブジェクトです。これらのプロパティは、エミッターを新規作成するときにエミッター・ファクトリーで使用されます。使用可能なイベント・サービス伝送は、エミッター・ファクトリー設定から表示または変更できません。

JMS 伝送

JMS 伝送は、エミッターが JMS キューを使用してイベント・サービスに非同期にアクセスする方法を決定するプロパティを定義するオブジェクトです。これらのプロパティは、エミッターを新規作成するときにエミッター・ファクトリーで使用されます。使用可能な JMS 伝送は、エミッター・ファクトリー設定から表示または変更できます。

イベント・グループ

イベント・グループは、イベントの論理的な集合であり、イベントをその内容に応じて分類するために使用されます。イベント・サービスからイベント

を照会する場合、またはイベント配布をサブスクライブする場合、イベント・コンシューマーは、イベント・グループを指定することにより、そのグループのイベントのみを取得できます。イベント・グループは、永続データ・ストアに保管するイベントを指定するために使用することもできます。使用可能なイベント・グループを管理コンソールで表示するには、「サービス統合」>「Common Event Infrastructure」>「イベント・サービス (Event service)」>「イベント・サービス (Event services)」>「event_service」>「イベント・グループ」をクリックします。

管理コンソールを使用した Common Event Infrastructure の構成

サーバー管理コンソールを使用して Common Event Infrastructure を構成します。

このタスクについて

管理コンソールの「Common Event Infrastructure サーバー」パネルを開きます。

サーバーを構成する場合は、「サーバー」>「サーバー・タイプ」>「WebSphere Application Server」> *server_name* >「ビジネス・インテグレーション」>「Common Event Infrastructure」>「Common Event Infrastructure サーバー」を選択します。

クラスターを構成する場合は、「サーバー」>「クラスター」>「WebSphere Application Server クラスター」> *cluster_name* >「ビジネス・インテグレーション」>「Common Event Infrastructure」>「Common Event Infrastructure サーバー」をクリックします。

手順

1. Common Event Infrastructure エンタープライズ・アプリケーションのデプロイメントを可能にするために、「イベント・インフラストラクチャー・サーバーを使用可能に設定」というチェック・ボックスを選択します。サーバーが既に構成済みである場合は、このチェック・ボックスを選択するかクリアすることによって、サーバーを使用可能または使用不可に設定できます。使用可能に設定するチェック・ボックスがクリアされている場合、Common Event Infrastructure はまだ構成されていないか、以前の構成が既に無効にされており、サーバーが再始動されていません。情報メッセージにより、このデプロイメント・ターゲットに Common Event Infrastructure が構成済みであるかどうかが表示されます。サーバーが既に構成済みである場合は、イベント・データベースかメッセージ・ストア、またはその両方のデータ・ソース設定を変更できます。

注: サーバーがまだ構成されていない場合に、チェック・ボックスを選択して Common Event Infrastructure サーバーを使用可能にすると、表示されたパラメーターを変更しない限り、それらのパラメーターを使用してサーバーが構成されます。

- 初めて構成を行う場合は、共通データベースにイベント・データ・ソース・テーブルが作成されます。既に Common Event Infrastructure サーバー構成がある場合は、データベースを作成する必要があります。
- 共通データベース内の固有スキーマの下にメッセージング・サービスが作成されます。

Common Event Infrastructure が構成されているサーバークラスターを再始動すると、新規の変更が有効になります。

2. イベント・データベースを構成 (または既存のイベント・データベース構成の現在の設定を変更) します。次のいずれかで、フィールドに適切な設定値を入力してください。
 - 「編集」をクリックして、このパネルよりも詳細なオプション・リストを持つデータベース構成パネルを表示します。
 - パネルの各フィールドを使用して、情報を入力します。
 - a. 「データベース名」 - イベントの保管に使用するデータベースの名前を入力します。
 - b. 「テーブルの作成」 - イベント・データベースにデータベース表を作成する場合は、このチェック・ボックスを選択します。

注: 別のサーバーにあるデータベースを使用するように Common Event Infrastructure を構成する場合は、このコントロールを使用してテーブルを作成することはできません。この構成をすべて設定した後に生成されるデータベース・スクリプトを使用する必要があります。その場合、「編集」をクリックしてデータ・ソース詳細パネルを表示すると、データベース作成スクリプトの場所が表示されます。

- c. 「ユーザー名」および「パスワード」 - イベント・データベースの認証時に使用されます。
- d. 「サーバー」 - イベント・データベースが存在するサーバーの名前を入力します。
- e. 「プロバイダー」 - データベースのプロバイダーをメニューから選択します。

注: 「スキーマ」フィールドは、iSeries® または z/OS プラットフォーム上で DB2 を使用してデータベースを作成する場合のみアクティブになります。それ以外の場合、「スキーマ」フィールドは使用できません。

重要: ターゲット・データベースにテーブルが存在する場合は、構成が失敗することがあります。

3. Common Event Infrastructure バスがサーバー上で「ローカル」として存在するか、あるいは「リモート」として他のサーバー上に存在するかを選択します。「リモート」を選択した場合は、リモートのロケーションをメニューから選択するか、あるいは「新規」をクリックして新規リモート・バスを作成します。
4. メッセージング用の Common Event Infrastructure サポートを構成します。
 - 「編集」をクリックして、このパネルよりも詳細なオプション・リストを持つデータベース構成パネルを表示します。
 - パネルの各フィールドを使用して、情報を入力します。
 - a. 「データベース名」 - メッセージの保管に使用するデータベースの名前を入力します。
 - b. 「スキーマ」 - スキーマの名前を入力します。表示されたデフォルト名をそのまま使用してもかまいません。
 - c. 「ユーザー名」および「パスワード」 - メッセージング・データベースの認証時に使用されます。

- d. 「サーバー」 - メッセージング・データベースが存在するサーバーの名前を入力します。
 - e. 「プロバイダー」 - データベースのプロバイダーをメニューから選択します。
5. 「OK」 または 「適用」 をクリックします。
 6. サーバーまたはクラスターを再始動します。

タスクの結果

これで、Common Event Infrastructure の主要な部分がすべて構成されて、サーバーまたはクラスターで稼働するようになります。この構成にはイベント・データ・ストア、メッセージング・エンジン、およびイベント・アプリケーションが含まれます。このパネルを使用することにより、多数のコマンドとステップを使用して Common Event Infrastructure を構成する必要がなくなります。

次のタスク

サーバーまたはクラスターの再始動後は、アプリケーションから発行されたサービス・コンポーネント・イベントを保管できるようになります。この後は、「**Common Event Infrastructure の宛先**」パネルを選択して、Common Event Infrastructure サーバーのランタイム・プロパティを変更することができます。また、始動時に Common Event Infrastructure サーバーを始動するかどうかを選択したり、イベントが送信されるエミッター・ファクトリー JNDI 名を指定することができます。

Common Event Infrastructure アプリケーションのデプロイ

Common Event Infrastructure を使用するには、まずイベント・サービスおよび関連するリソースをサーバー・ランタイム環境にデプロイしておく必要があります。

このタスクについて

Common Event Infrastructure エンタープライズ・アプリケーションには、イベント・サービスのランタイム・コンポーネントと、非同期イベント送信で使用されるデフォルトのメッセージング構成が含まれます。

イベント・サービスをデプロイするには、以下の手順を実行します。

手順

wsadmin ツールから **deployEventService** 管理コマンドをバッチ・モードまたは対話モードで実行します。 **deployEventService** 管理コマンドのパラメーターは、以下のとおりです。

nodeName

イベント・サービスをデプロイするノードの名前。このパラメーターは任意指定です。ノード名を指定しない場合、デフォルト値は現在のノードです。ノード名を指定する場合は、**serverName** パラメーターを使用してサーバー名も指定する必要があります。クラスターにイベント・サービスをデプロイする場合は、このパラメーターは無効です。

serverName

イベント・サービスをデプロイするサーバーの名前。このパラメーターは、ノードを指定する場合にのみ必要です。クラスターにイベント・サービスをデプロイする場合は、このパラメーターは無効です。

clusterName

イベント・サービスをデプロイするクラスターの名前。このパラメーターは任意指定です。ノードまたはサーバーの有効範囲にデプロイする場合は指定しないでください。

enable

サーバー始動時にイベント・サービスを自動的に開始するかどうかを示します。デフォルト値は true です。

タスクの結果

管理コマンドが完了すると、Common Event Infrastructure のイベント・サービスおよびデフォルトのメッセージング構成が、指定された有効範囲にデプロイされます。

次のタスク

WebSphere セキュリティーが使用可能な場合は、**setEventServiceJmsAuthAlias** 管理コマンドを使用して JMS 認証別名とパスワードを構成する必要もあります。

クラスターにイベント・サービスをデプロイする場合は、イベント・データベースを手動で構成する必要もあります。

Common Event Infrastructure のクラスターへのデプロイ

Common Event Infrastructure リソースは、いくつかの方法でクラスター環境にデプロイできます。

Common Event Infrastructure の既存のクラスターへのデプロイ:

既存のクラスターにイベント・サービス・アプリケーションをデプロイできます。

このタスクについて

イベント・サービス・アプリケーションをクラスターにデプロイすることは、アプリケーションをスタンドアロン・サーバーにデプロイすることと本質的には同じです。ただし、クラスター環境では、デフォルトのイベント・データベースは構成されません。

Common Event Infrastructure をクラスター環境にデプロイして構成するには、以下の手順を実行します。

手順

1. クラスターの名前を指定し、それ以外はスタンドアロン・サーバーの場合と同様に、**deployEventService** 管理コマンドを実行します。クラスターを指定するには、**clusterName** パラメーターを使用します。

2. デプロイメント・マネージャー・システムで、データベース構成管理コマンドを実行します。 `clusterName` パラメーターを使用して、クラスター名を指定します。このコマンドにより、データベース構成スクリプトが生成されます。
3. 生成されたデータベース構成スクリプトをデータベース・システムにコピーします。
4. データベース・システムに対してデータベース構成スクリプトを実行して、イベント・データベースを作成します。
5. デプロイメント・マネージャー・システムで `enableEventService` コマンドを実行して、イベント・サービスを使用可能にします。 `clusterName` パラメーターを使用して、クラスターの名前を指定します。

既存の Common Event Infrastructure サーバーの変換によるクラスターの作成:

Common Event Infrastructure と共に構成されている既存のスタンドアロン・サーバーを変換して、クラスターを作成することができます。

始める前に

既存のサーバーを変換する前に、そのサーバーが Common Event Infrastructure 用に完全に構成されていることを確認してください。この構成には、イベント・サービス・アプリケーションのデプロイとイベント・データベースの構成も含まれます。

このタスクについて

クラスターを作成するには、以下の手順を実行します。

手順

1. 通常の WebSphere プロセスに従って、スタンドアロン・サーバーを新しいクラスターの最初のメンバーに変換します。サーバーを変換するときに、以下のステップが実行されます。
 - サーバーの有効範囲にある使用可能な Common Event Infrastructure リソースが、新しいクラスターの有効範囲に移動されます。

デフォルト・データベース: 既存のサーバーがデフォルトの Derby データベースと共に構成されている場合、データベース・リソースは、クラスターの有効範囲に移動されずに除去されます。デフォルトのデータベース構成は、クラスターでサポートされません。この場合、デフォルトでは、クラスター内のイベント・サービスは使用不可になります。
 - デプロイするイベント・サービス・アプリケーションのターゲット・リストが変更され、変換されるサーバーが除去されて新しいクラスターが追加されます。
2. オプション: 変換されるサーバーがデフォルトの Derby データベースと共に構成されていた場合は、クラスターの新しいイベント・データベースを構成してから、イベント・サービスを使用可能にする必要があります。
 - a. デプロイメント・マネージャー・システムで、データベース構成管理コマンドを実行します。 `clusterName` パラメーターを使用して、クラスター名を指定します。このコマンドにより、データベース構成スクリプトが生成されます。

- b. 生成されたデータベース構成スクリプトをデータベース・システムにコピーします。
- c. データベース・システムに対してデータベース構成スクリプトを実行して、イベント・データベースを作成します。
- d. デプロイメント・マネージャー・システムで **enableEventService** コマンドを実行して、イベント・サービスを使用可能にします。 `clusterName` パラメーターを使用して、クラスターの名前を指定します。

既存の Common Event Infrastructure サーバーをテンプレートとして使用してクラスターを作成する:

既存の Common Event Infrastructure サーバーをテンプレートとして指定することにより、クラスターを作成できます。

始める前に

この方法を使用してクラスターを作成する前に、Common Event Infrastructure 用に完全に構成済みである既存のサーバーを用意する必要があります。この構成には、イベント・サービス・アプリケーションのデプロイとイベント・データベースの構成も含まれます。

このタスクについて

クラスターを作成するには、以下の手順を実行します。

手順

1. 通常の WebSphere プロセスに従い、既存の Common Event Infrastructure サーバーを最初のクラスター・メンバーのテンプレートとして使用して、新しいクラスターを作成します。最初のメンバーを作成すると、以下のステップが実行されます。
 - 既存のサーバーの有効範囲にある使用可能な Common Event Infrastructure リソースが、新しいクラスターの有効範囲にコピーされます。

デフォルト・データベース: 既存のサーバーがデフォルトの Derby データベースと共に構成されている場合、データベース・リソースは、クラスターの有効範囲にはコピーされません。デフォルトのデータベース構成は、クラスターではサポートされません。この場合、クラスター内のイベント・サービスはデフォルトで使用不可になります。
 - デプロイするイベント・サービス・アプリケーションのターゲット・リストが変更され、新しいクラスターが組み込まれます。
2. オプション: 既存のサーバーがデフォルトの Derby データベースと共に構成されていた場合は、クラスターの新しいイベント・データベースを構成してから、イベント・サービスを使用可能にする必要があります。
 - a. デプロイメント・マネージャー・システムで、データベース構成管理コマンドを実行します。 `clusterName` パラメーターを使用して、クラスター名を指定します。このコマンドにより、データベース構成スクリプトが生成されます。
 - b. 生成されたデータベース構成スクリプトをデータベース・システムにコピーします。

- c. データベース・システムに対してデータベース構成スクリプトを実行して、イベント・データベースを作成します。
- d. デプロイメント・マネージャー・システムで `enableEventService` コマンドを実行して、イベント・サービスを使用可能にします。 `clusterName` パラメーターを使用して、クラスターの名前を指定します。

イベント・メッセージングの構成

JMS でイベントをイベント・サービスに転送する場合に使用するメッセージングの構成を変更できます。

このタスクについて

管理コンソール・パネルを使用して、サーバー上に `Common Event Infrastructure` を構成するときは、`Common Event Infrastructure` のメッセージング・インフラストラクチャーを作成します。一般に、メッセージング構成は、イベント・サービスへのイベントの非同期伝送にデフォルトのメッセージング・プロバイダーを使用し、単一の JMS キューを作成します。必要であれば、このメッセージング構成を変更できます。

追加の JMS キューの構成

デフォルトのイベント・メッセージング構成を使用している場合は、イベントをイベント・サービスに転送するために JMS キューを追加できます。

このタスクについて

デフォルトのメッセージング構成を使用して追加の JMS キューを構成するには、サービス統合バスのキュー宛先に転送される複数の JMS キューをセットアップします。`Common Event Infrastructure` サービス統合バス のキュー宛先は、イベント・サービスがデプロイされる有効範囲によって決まります。

有効範囲	サービス統合バスのキュー宛先
サーバー	<code>node.server.CommonEventInfrastructureQueueDestination</code>
クラスター	<code>cluster.CommonEventInfrastructureQueueDestination</code>

サービス統合バスの構成の詳細については、資料を参照してください。

外部 JMS プロバイダーを使用したイベント・メッセージングの構成

デフォルトの組み込みメッセージング構成をイベント伝送で使わない場合は、非同期メッセージ・トランスポートを構成することにより、外部の Java Messaging Service (JMS) プロバイダーを使用できます。

始める前に

外部 JMS プロバイダーを使用してイベント・メッセージングを構成する前に、まず使用している JMS プロバイダーに適したインターフェースを使用して JMS キューおよび接続ファクトリーを作成する必要があります。リスナー・ポートまたはアクティベーション・スペックを作成する必要もあります。

このタスクについて

外部 JMS プロバイダーを使用してイベント・メッセージングを構成するには、以下の手順を実行します。

手順

wsadmin ツールから **deployEventServiceMdb** 管理コマンドをバッチ・モードまたは対話モードで実行します。 **deployEventServiceMdb** コマンドのパラメーターは、以下のとおりです。

applicationName

デプロイするイベント・サービスのメッセージ駆動型 Bean のアプリケーション名。このパラメーターは必須です。

nodeName

イベント・サービスのメッセージ駆動型 Bean をデプロイするノードの名前を指定します。ノード名を指定する場合は、サーバー名も指定する必要があります。ノード名はオプション・パラメーターで、デフォルト値は現在のノードです。クラスターにアプリケーションをデプロイする場合は、このパラメーターを指定しないでください。

serverName

イベント・サービスのメッセージ駆動型 Bean をデプロイするサーバーの名前を指定します。このパラメーターは、サーバーの有効範囲にアプリケーションをデプロイする場合、必須です。それ以外の場合は、任意指定です。クラスターにアプリケーションをデプロイする場合は、サーバー名を指定しないでください。

clusterName

イベント・サービスのメッセージ駆動型 Bean をデプロイするクラスターの名前を指定します。このパラメーターは、アプリケーションをクラスターにデプロイする場合にのみ指定します。

listenerPort

イベントの公開時にイベント・サービスのメッセージ駆動型 Bean が使用するリスナー・ポートの名前を指定します。指定するリスナー・ポートは存在している必要があります。リスナー・ポートとアクティベーション・スペックのいずれかを指定する必要がありますが、両方を指定しないでください。

activationSpec

イベントの公開時にイベント・サービスのメッセージ駆動型 Bean が使用するアクティベーション・スペックの JNDI 名を指定します。指定するアクティベーション・スペックは存在している必要があります。リスナー・ポートとアクティベーション・スペックのいずれかを指定する必要がありますが、両方を指定しないでください。

qcfJndiName

イベント・サービスのメッセージ駆動型 Bean が使用する JMS キュー接続ファクトリーの JNDI 名。このパラメーターは、アクティベーション・スペックを指定する場合、必須です。それ以外の場合は、任意指定です。キュー接続ファクトリーとリスナー・ポートを指定する場合、キュー接続ファクトリーは、そのリスナー・ポート用に構成されたものと一致する必要があります。

タスクの結果

deployEventServiceMdb 管理コマンドは、指定したリスナー・ポートまたはアクティベーション・スペック用に構成された、イベント・サービスのメッセージ駆動型 Bean をデプロイします。外部の JMS 構成を使用して、エミッター・ファクトリーと JMS 伝送の作成も行います。アプリケーションは、(デフォルトのメッセージング構成を使用するように構成された) デフォルトのエミッター・ファクトリー、または (外部の JMS プロバイダーを使用する) 新しいエミッター・ファクトリーを使用できます。

次のタスク

イベント・サービスに対して複数の JMS キューをセットアップする場合は、異なるエンタープライズ・アプリケーション名と JMS キューを指定して、このコマンドを複数回実行します。スクリプトを実行するたびに、追加のメッセージ駆動型 Bean がデプロイされ、新規リソースで指定された JMS キューを使用するように構成されます。

JMS 認証別名の構成

WebSphere セキュリティーが使用可能で、非同期 JMS メッセージングを使用してイベントをイベント・サービスに送信する場合は、JMS 認証別名を構成する必要があります。

このタスクについて

JMS 認証別名を構成するには、以下の手順を実行します。

手順

wsadmin ツールから **setEventServiceJmsAuthAlias** 管理コマンドをバッチ・モードまたは対話モードで実行します。 **setEventServiceJmsAuthAlias** コマンドのパラメーターは、以下のとおりです。

userName

JMS 認証別名で使用するユーザーの名前。このパラメーターは必須です。

password

JMS 認証別名で使用するユーザーのパスワード。このパラメーターは必須です。

nodeName

JMS 認証別名を更新または作成するノードの名前。ノード名を指定する場合は、サーバー名も指定する必要があります。クラスターの認証別名を構成する場合は、ノード名を指定しないでください。

serverName

JMS 認証別名を更新または作成するサーバーの名前。このパラメーターは、ノードを指定する場合にのみ必要です。クラスターで認証別名を構成する場合は、このパラメーターは無効です。

clusterName

JMS 認証別名を更新または作成するクラスターの名前。このパラメーターは、

クラスターで認証別名を構成する場合にのみ指定します。クラスター名を指定する場合は、ノード名またはサーバー名を指定しないでください。

タスクの結果

イベント・サービス・オブジェクトが使用する JMS 認証別名は、指定された有効範囲で更新されます。認証別名が存在しない場合は、指定された値を使用して認証別名が作成されます。

イベント・データベースへのデータの取り込み

イベントの永続性をサポートするには、イベント・データベースが必要になります。createDB.sh スクリプトが実行されていれば、イベント・データベースが作成され、データの取り込みも完了しています。createDB.sh スクリプトが実行されていない場合は、生成された DDL スクリプトを実行して、イベント・データベースにデータを取り込みます。

始める前に

作業を開始する前に、イベント・データベースを作成しておく必要があります。

『DButility.sh、SPUFI、または DSNTEP2 データベースおよびストレージ・グループの作成』を参照してください。

このタスクについて

管理コンソールで生成した DDL スクリプトを使用して、イベント・データベースにデータを取り込みます。

手順

- 以下に示す生成済み DDL スクリプト・ファイルを /WebSphere/V6R1M0/DeploymentManager/profiles/default/databases/event/*cluster_name*/dbscripts/db2zos/ddl から作業ディレクトリーにコピーします (この *cluster_name* は、Network Deployment セルが属しているクラスターの名前です)。
 - catalogSeed.ddl
 - cr_db.ddl
 - cr_db_catalog.ddl
 - cr_tbl.ddl
 - cr_tbl_catalog.ddl
 - ins_metadata.ddl
- 次のコマンドを指定して、各ファイルのコピーに適切なアクセス権を割り当てます。

```
chmod 755 catalogSeed.ddl
```
- 必要に応じてファイル内の値を編集します。命名要件に合わせてデータベース名およびストレージ・グループ名を変更します。ファイルで指定する名前は、構成スクリプトへの入力を提供した応答ファイルで入力した値と同じにする必要があります。

注: ファイルは ASCII フォーマットで提供されています。スクリプトの表示、編集、および実行に使用するツールで、スクリプトが EBCDIC 形式であることが必要な場合は、iconv コマンドを使用して、ファイルを EBCDIC に変換します。以下に例を示します。

```
iconv -t IBM-1047 -f ISO8859-1 catalogSeed.ddl >
catalogSeed_EBCDIC.ddl
```

ファイルを ASCII フォーマットから EBCDIC に変換するが、そのファイルの実行は ASCII フォーマットで行う必要がある場合、iconv を使用して、ファイルを ASCII に戻します。以下に例を示します。

```
iconv -t ISO8859-1 -f IBM-1047 catalogSeed_EBCDIC.ddl >
catalogSeed.ddl
```

4. 選択したツールを使用して、カスタマイズ済みのスクリプトを実行します。例えば、DBUtility.sh または SPUFI などです。

タスクの結果

これで、イベント・データベースにデータが取り込まれました。

以前のバージョンからの DB2 for z/OS イベント・データベースのアップグレード

z/OS システムに Common Event Infrastructure バージョン 5.1 の既存の DB2 イベント・データベースがある場合は、それを現行バージョンにアップグレードする必要があります。

このタスクについて

z/OS システムの DB2 イベント・データベースをアップグレードするには、以下の手順を実行します。

手順

1. 既存のイベント・データベースのバックアップ・コピーを作成します。
2. *profile_root/bin* ディレクトリーに移動します。
3. DB2 for z/OS アップグレード・スクリプトを以下のように実行します。
 - `eventUpgradeDB2ZOS.sh runUpgrade=[true|false] dbUser=user`
`[dbName=name] [dbPassword=pw]`
`[scriptDir=dir] storageGroup=group`
`bufferPool4K=4kbufpool bufferPool8k=8kbufpool`
`bufferPool16K=16kbufpool`

通常、必要なパラメーターは以下のとおりです。

runUpgrade

生成された DDL スクリプトをアップグレード・スクリプトで自動的に実行してデータベース・アップグレードを完了するかどうかを指定します。このパラメーターは必須です。データベースを後で、または別のシステム上で手動でアップグレードする場合は、false を指定します。

z/OS システム: このパラメーターは、ネイティブ z/OS システムでは無視されます。生成された DDL スクリプトの自動実行は、クライアント・システムでのみサポートされます。

dbUser

使用する DB2 ユーザー ID を指定します。このパラメーターは必須です。

dbName

DB2 データベース名を指定します。デフォルトのイベント・データベース名は event です。このパラメーターは、runUpgrade=true を指定した場合、必須です。

dbPassword

指定した DB2 ユーザー ID のパスワードを指定します。このパラメーターは任意指定です。パスワードを指定しない場合、DB2 は、パスワードの入力プロンプトを表示します。

scriptDir

生成された DDL スクリプトを含めるディレクトリーを指定します。このパラメーターは任意指定です。ディレクトリーを指定しない場合は、`¥eventDBUpgrade¥db2zos` ディレクトリーにスクリプトが保管されます。

storageGroup

ストレージ・グループの名前を指定します。このパラメーターは必須です。

bufferPool4K

4K バッファ・プールの名前を指定します。このパラメーターは必須です。

bufferPool8K

8K バッファ・プールの名前を指定します。このパラメーターは必須です。

bufferPool16K

16K バッファ・プールの名前を指定します。このパラメーターは必須です。

詳細なパラメーターのリストとその使用方法を確認するには、パラメーターを指定せずに `eventUpgradeDB2ZOS` スクリプトを実行します。

タスクの結果

アップグレード・スクリプトは、イベント・データベースをアップグレードするために必要な DDL スクリプトを生成します。クライアント・システムで `runUpgrade=true` を指定した場合は、DDL スクリプトが自動的に実行され、アップグレードが完了します。

次のタスク

ここで、生成された DDL スクリプトを、SQL Processor Using File Input (SPUFI) 機能を使用して手動で実行する必要があります。この手順によって、データベース・アップグレードが完了します。

WebSphere Business Monitor 用のクロスセル Common Event Infrastructure 構成

Common Event Infrastructure (CEI) イベントを生成するリモート・サーバーと、WebSphere Business Monitor サーバーの間に、接続を構成する必要があります。

このタスクについて

マルチサーバー環境で複数のセルにわたって CEI を構成する方法について詳しくは、IBM WebSphere Business Monitor インフォメーション・センターの『WebSphere Business Monitor を使用するようにリモート CEI サーバーを構成』というトピックを参照してください。

WebSphere Business Integration Adapter の構成

WebSphere Business Integration Adapter と WebSphere Process Server を連携させるには、インストールおよび構成の手順を実行する必要があります。

手順

1. アダプターをインストールします。
 - a. WebSphere Business Integration Adapters のインストール方法が説明されている『WebSphere Business Integration Adapters 製品のインストール』の手順に従います。
 - b. 『WebSphere Business Integration Adapters の資料』に移動し、「アダプター」の下にあるナビゲーションを展開して、ご使用のアダプターに固有の必須の追加手順があるかどうかを確認します。ご使用のアダプターについて追加のインストール・タスクが記載されている場合は、そのタスクを実行します。
2. 『WebSphere Business Integration Adapters の資料』に移動して、「アダプター」の下にあるナビゲーションを展開し、ご使用のアダプターの構成手順に従って、アダプターを構成します。構成手順を実行すると、必要な成果物が生成されます。
3. メディエーション・モジュールのデプロイの指示に従って、アプリケーション EAR ファイルをインストールします。

WebSphere Business Integration Adapter の管理のセットアップ

WebSphere Business Integration Adapter を管理できるようにするには、いくつかの管理機能を実行する必要があります。

始める前に

- 『WebSphere Business Integration Adapters 製品のインストール』に示されている手順を理解している必要があります。
- アプリケーション EAR ファイルをインストールして、このタスクを実行する前に WebSphere Business Integration Adapter に必要な成果物を作成する必要があります。

このタスクについて

WebSphere Business Integration Adapter を管理するには、以下の管理機能を実行します。

手順

1. キュー接続ファクトリーを作成します。

管理コンソールの最上位から、以下の手順を実行します。

- a. 「リソース」を展開します。
- b. 「JMS」を展開します。
- c. 「キュー接続ファクトリー (Queue connection factories)」を選択します。
- d. 管理入出力キューの有効範囲レベルに一致する有効範囲レベルを選択します。
- e. 「新規」をクリックして、新規 JMS キュー接続ファクトリーを作成します。
- f. JMS リソース・プロバイダーを選択します。「デフォルトのメッセージング・プロバイダー」を選択して「OK」をクリックします。
- g. 以下を例外として、すべてのデフォルト値を受け入れます。
 - 名前: QueueCF
 - JNDI 名: jms/QueueCF
 - バス名: 現在のバス名
- h. 「OK」をクリックして、新しい JMS キュー接続ファクトリーの作成を完了します。

「JMS キュー接続ファクトリー」パネルの上部にメッセージ・ウィンドウが表示されます。

- i. メッセージ・ウィンドウで「保管」をクリックして、ローカル構成レベルで行った変更をマスター構成に適用します。

2. WebSphere Business Integration Adapter リソースを作成します。

管理コンソールの最上位から、以下の手順を実行します。

- a. 「リソース」を展開します。
- b. 「WebSphere Business Integration Adapter」ページを開きます。

「WebSphere Business Integration Adapters」を選択します。
- c. 「新規」をクリックして、新しい WebSphere Business Integration Adapter を作成します。
- d. 以下を例外として、すべてのデフォルト値を受け入れます。
 - 名前: EISConnector
 - キュー接続ファクトリー JNDI 名: jms/QueueCF
 - 管理入力キュー JNDI 名: *connectorName/AdminInQueue*
 - 管理出力キュー JNDI 名: *connectorName/AdminOutQueue*
- e. 「OK」をクリックして、WebSphere Business Integration Adapter の作成を完了します。

「WebSphere Business Integration Adapters」パネルの上部にメッセージ・ウィンドウが表示されます。

- f. メッセージ・ウィンドウで「保管」をクリックして、ローカル構成レベルで行った変更をマスター構成に適用します。

3. WebSphere Business Integration Adapter サービスを使用可能にします。

管理コンソールの最上位から、以下の手順を実行します。

- a. 「サーバー」を展開します。
- b. 「サーバー・タイプ」を展開します。
- c. 「WebSphere Application Server」を選択します。
- d. サーバー・リストから、WebSphere Business Integration Adapter サービスを有効にするサーバーを選択します。

対象となるリソースをホストするサーバー名をクリックします。

- e. 「構成」タブの「Business Integration」リストから、「WebSphere Business Integration Adapter サービス」を選択します。
- f. 「サーバー始動時にサービスを使用可能にする」チェック・ボックスが選択されていることを確認します。
- g. 「OK」をクリックします。

「WebSphere Business Integration Adapters」 ページの上部にメッセージ・ウィンドウが表示されます。

- h. WebSphere Business Integration Adapter サービスを使用可能とする予定の各サーバーごとに、ステップ 3d から 3g を繰り返します。
- i. メッセージ・ウィンドウで「保管」をクリックして、ローカル構成レベルで行った変更をマスター構成に適用します。

注: WebSphere Business Integration Adapter サービスを使用可能または使用不可にする場合、変更を有効にするためにサーバーを再始動する必要があります。

Service Federation Management 用の WebSphere Process Server の構成

WebSphere Service Registry and Repository バージョン 7.0 で提供された Service Federation Management (SFM) コンソールによって管理できる接続サーバーとして、WebSphere Process Server を使用可能にすることができます。これによって、SFM コンソールは WebSphere Process Server で、SFM プロキシを構成できます。

このタスクについて

別のビジネス単位に、別のエンタープライズ・サービス・バス (ESB) を保有している場合があります。各 ESB および関連付けられたサービス・レジストリーは、接続されたサービス・アプリケーションの別のドメインを構成しています。この結果、ドメイン間で大量のアプリケーションの重複が発生する可能性があります。また、ドメインを横断するアプリケーション接続の実装に要する開発体力が増加する可能性があります。WebSphere Service Registry and Repository バージョン 7.0 で提供されている SFM を使用すれば、異なる ESB 間にブリッジが確立され、サービスおよびアプリケーションをドメイン間で共用することができます。

SFM プロバイダー:

- フェデレーション対応コンテンツを統一して表示できるフェデレーション・モデル。

- ドメインをサポートする、サービス接続およびレジストリー・コンポーネントにアクセスする、Service Connectivity Management プロトコル。
- サービス・ドメインを制御するコンソール。

SFM を使用することにより、コンソール・ユーザーは 1 つのドメインでサービスを構成し、そのサービスを使用して別のドメインの消費者にサービスを提供することができます。1 つのドメインのサービス・エンドポイントは、別のドメインのサービス・プロキシ・エンドポイントとして使用可能です。

Service Connectivity Management 接続サーバーの構成

Service Federation Management (SFM) コンソールは、Service Connectivity Management プロトコル (SCMP) を使用して、WebSphere Process Server と通信します。

このタスクについて

WebSphere Process Server は、SCM Connectivity Server という名前のシステム REST サービスとして、Atom ベースのプロトコルを公開します。このサービスは、スタンドアロン・サーバー用の REST サービス・プロバイダーと、Network Deployment 環境のデプロイメント・マネージャーでは、デフォルトで使用可能になっています。

手順

1. REST サービスを構成します。プロトコルによって戻される Atom 文書には、SFM コンソールによって保持される絶対 URL が含まれています。これらの絶対 URL で使用されるプロトコル、ホスト名、およびポート番号は、REST サービス構成から取得されます。SFM コンソール・サーバーと WebSphere Process Server の間のロード・バランシングおよびネットワーク・コンポーネントを検討することが重要です。
 - a. サービス・プロバイダーのトピックの『REST サービスの構成』で説明されているように、スタンドアロン・サーバーまたはデプロイメント・マネージャー REST サービス・プロバイダー用に、プロトコル、完全修飾ホスト名、およびポート番号を構成します。
2. SFM コンソール・ユーザーに、接続サーバーへのアクセスに必要な詳細を提供してください。
 - a. 接続サーバー用の Atom サービス文書の URL は、REST サービス・パネルで見つけることができます。サービスのタイプは、*SCM Connectivity Server* です。
 - b. WebSphere Process Server 管理セキュリティーが使用可能な場合、SFM コンソール・ユーザーも、サービス・エンドポイントへのアクセスには、ユーザー名およびパスワードが必要です。これらの資格情報は、Service Connectivity Architecture モジュールをインストールするための十分な管理権限を持つ、RestServicesUser グループ内のユーザーのものでなければなりません。

Service Connectivity Management 接続プロバイダーの構成

管理コンソールを使用して、ご使用の環境の Service Connectivity Management (SCM) 接続プロバイダーを構成できます。

このタスクについて

SCM 接続プロバイダーは、SCM プロトコルを使用して公開される ESB の論理区画です。その接続プロバイダー上で SCM グループ・プロキシを作成するときにプロキシ・ゲートウェイ・モジュールがデプロイされるターゲット (サーバーまたはクラスター) が定義されます。また、それらのグループ・プロキシ上で作成されたプロキシ・ターゲットで使用されるプロパティも定義されます。

手順

「サービス統合」>「SCM 接続プロバイダー」を選択します。「SCM 接続プロバイダー」ページが開き、現在の環境内の接続プロバイダーがすべて表示されます。

タスクの結果

SCM 接続プロバイダーは、このページから、追加、除去または使用できます。

接続プロバイダーの追加

管理コンソールを使用して、Service Connectivity Management (SCM) 接続プロバイダーとして、サーバーまたはクラスターを追加できます。

手順

1. 「サービス統合」>「SCM 接続プロバイダー」をクリックします。「SCM 接続プロバイダー」ページが開き、現在の環境内の接続プロバイダーがすべて表示されます。
2. 「追加」をクリックして、サーバーまたはクラスターを接続プロバイダーとして追加します。接続プロバイダーを追加するためのウィザードが開きます。
3. ウィザードで「ステップ 1 サーバーまたはクラスターの選択」を完了して、この接続プロバイダー用に SCM グループ・プロキシをデプロイする必要があるサーバーまたはクラスターを識別します。「次へ」をクリックします。
4. ウィザードで「ステップ 2 SCM 接続プロバイダー・プロパティの指定」を完了して、プロパティを指定します。

オプション	説明
名前	SCM 接続プロバイダーの名前。これはセル内で固有でなければなりません。名前が既に存在する場合は、例外がスローされます。名前、説明、連絡先、組織、およびロケーションは、Service Federation Management コンソールのユーザーに表示されます。
説明	SCM 接続プロバイダーの簡単な説明。これは任意指定であり、デフォルトは空ストリングです。名前、説明、連絡先、組織、およびロケーションは、Service Federation Management コンソールのユーザーに表示されます。

オプション	説明
連絡先	SCM 接続プロバイダーの連絡先担当者の名前。これは任意指定であり、デフォルトは空ストリングです。名前、説明、連絡先、組織、およびロケーションは、Service Federation Management コンソールのユーザーに表示されます。
組織	SCM 接続プロバイダーの所有組織の名前。これは任意指定であり、デフォルトは空ストリングです。名前、説明、連絡先、組織、およびロケーションは、Service Federation Management コンソールのユーザーに表示されます。
場所	SCM 接続プロバイダーのロケーション。これは任意指定であり、デフォルトは空ストリングです。名前、説明、連絡先、組織、およびロケーションは、Service Federation Management コンソールのユーザーに表示されます。
HTTP ホスト	保護されていないプロキシ・ターゲットのエンドポイントに対して返されるホスト名。これは、別のドメイン内の Web サービス・クライアントがプロキシへのアクセスに使用するホストであり、Web サーバーおよびその他のネットワーク・コンポーネントが考慮されます。
HTTP ポート	保護されていないプロキシ・ターゲットのエンドポイントに対して返されるポート。これは、別のドメイン内の Web サービス・クライアントがプロキシへのアクセスに使用するポートであり、Web サーバーおよびその他のネットワーク・コンポーネントが考慮されます。
HTTPS ホスト	保護されているプロキシ・ターゲットのエンドポイントに対して返されるホスト名。これは、別のドメイン内の Web サービス・クライアントがプロキシへのアクセスに使用するホストであり、Web サーバーおよびその他のネットワーク・コンポーネントが考慮されます。
HTTPS ポート	保護されているプロキシ・ターゲットのエンドポイントに対して返されるポート。これは、別のドメイン内の Web サービス・クライアントがプロキシへのアクセスに使用するポートであり、Web サーバーおよびその他のネットワーク・コンポーネントが考慮されます。

オプション	説明
認証別名	基本認証資格情報を提供する認証別名の名前。この資格情報は、SCM 接続プロバイダーのドメインに関連付けられたサービス・レジストリーから、HTTP を使用して WSDL 文書を取得するために使用されます。サービス・レジストリーに接続するために基本認証が必要とされない場合は、このパラメーターを指定する必要はありません。
SSL 構成	SSL 構成の名前。この構成は、SCM 接続プロバイダーのドメインに関連付けられた保護されているサービス・レジストリーから、HTTP を使用して WSDL 文書を取得するために使用されます。これは任意指定であり、指定しなかった場合は、サーバーのデフォルト SSL 構成が使用されます。

5. 「終了」をクリックします。SCM 接続プロバイダー・ページが開き、新規の接続プロバイダーがリストされます。
6. 「メッセージ」セクションを確認し、接続プロバイダーおよびそのプロパティーが完了していることを確認してください。
7. 「保存」をクリックして、接続プロバイダーをマスター構成に保存します。

接続プロバイダーの除去

管理コンソールを使用して、Service Connectivity Management (SCM) 接続プロバイダーとして、サーバーまたはクラスターを除去できます。

手順

1. 「サービス統合」>「SCM 接続プロバイダー」をクリックします。「SCM 接続プロバイダー」ページが開き、現在の環境内の接続プロバイダーがすべて表示されます。
2. 接続プロバイダーを選択します。「除去」をクリックして、サーバーまたはクラスターを接続プロバイダーから除去します。

接続プロバイダーの使用

管理コンソールを使用して、Service Connectivity Management (SCM) 接続プロバイダーをリスト、表示および変更できます。

手順

1. 「サービス統合」>「SCM 接続プロバイダー」をクリックします。「SCM 接続プロバイダー」ページが開き、現在の環境内の接続プロバイダーがすべて表示されます。
2. 接続プロバイダーを選択して、詳細ページを表示します。
3. このページでフィールドを変更できます。ただし、「名前」、「作成者」、「作成」、または「更新」フィールドは変更できません。
4. 変更を完了するには、「適用」、「OK」、「リセット」、「キャンセル」ボタンを使用してください。

Service Component Architecture モジュールの Service Connectivity Management の使用

Service Component Architecture モジュールは、Service Federation Management コンソールがグループ・プロキシを作成するごとに、インストールされます。これらの Service Component Architecture モジュールは、エンタープライズ・アプリケーション・ビューおよび管理コンソールの Service Component Architecture モジュール・リストで表示できます。

バージョン管理されている Service Component Architecture モジュールは、グループ・プロキシ用に使用されます。基本モジュール名は ScmGroupProxy で、バージョン番号は v1_0_0 です。セル ID は、接続プロバイダー名およびセル内のグループ・プロキシ用固有 ID から形成されます。

モジュール・リストに表示されるサービス・モジュールの名前は ScmGroupProxy (*ConnectivityProviderName_UniqueId*) で、サービス・アプリケーション名は、ScmGroupProxy_v1_0_0_ConnectivityProviderName_UniqueIDApp という形式です。また、同じ固有 ID が、SCM プロトコルを介してグループ・プロキシにアクセスするために使用される、URL および Atom ID の一部も形成しています。

生成された固有 ID *xot5* で接続プロバイダー *ExampleConnectivityProvider* 上に作成されたグループ・プロキシは、最終的に ScmGroupProxy (*ExampleConnectivityProvider_xot5*) という名前のモジュールになり、アプリケーション ScmGroupProxy_v1_0_0_ExampleConnectivityProvider_xot5App として、接続プロバイダーに関連付けられたサーバーまたはクラスターにデプロイされます。

グループ・プロキシ・リソースを表す、Atom 文書にアクセスするための URL は、次の形式になります。

```
/rest/scmp/connectivity-provider/ExampleConnectivityProvider-g0jk9fzm/mediation/group-proxy-type/group-proxy/xot5-g0jkja19
```

この文書の Atom ID は次の形式になります。

```
urn:wesb-scmp:cell/localhostNode01Cell/connectivity-provider/ExampleConnectivityProvider-g0jk9fzm/mediation/group-proxy-type/group-proxy/xot5-g0jkja19
```

注: SCM グループ・プロキシの属性は、モジュールのプロモートされたプロパティとして表示されます。これらは、管理コンソールを介して表示できますが、変更してはいけません。

Service Connectivity Management (SCM) のプロキシ・ゲートウェイへのマッピング

Service Connectivity Management (SCM) グループ・プロキシ・モジュールは、WebSphere Process Server 内にプロキシ・ゲートウェイとして実装されています。

グループ・プロキシの SCM プロキシ・ターゲットは、プロキシ・ゲートウェイの仮想サービスとして表示され、プロキシ・ゲートウェイ・ウィジェットを

介して、WebSphere が提供する Business Space で表示することができます。プロキシー・ターゲットのプロパティは、仮想サービスのプロパティとして表示されます。

注: SCM グループ・プロキシー・モジュールに関連付けられた仮想サービスを、プロキシー・ゲートウェイ・ウィジェットを介して、追加、除去、または変更しないでください。

構成のトラブルシューティング

WebSphere Process Server の構成に失敗したときに、問題を診断できます。

WebSphere Process Server のエラー

構成タスクのいずれかで問題が発生した場合は、その問題についての情報を次の 3 つの主な情報源から入手できます。

1. タスクによって発行されるエラー・メッセージ
2. WebSphere デプロイメント・マネージャーまたはアプリケーション・サーバーのジョブ・ログにあるエラー・メッセージ。ノードを統合している場合は、ノード・エージェントのジョブ・ログにもメッセージが記録されています。
3. UNIX ファイル・システムのログ・ファイル

可能な場合にはいつでも、それぞれの問題の原因と解決策も症状と一緒に文書化されます。ここで説明される問題は、WebSphere Process Server のインストール手順の完了後、サーバーを始動するときに発生したものです。エラー・メッセージの例では、改行位置を変更することによってメッセージを読みやすくしています。したがって、ご使用のシステムでこれらのエラーが発生する場合には、メッセージの配置が若干異なります。

関連資料

『T2 ネイティブ・ライブラリー db2jcct2zos のロードの失敗』

このエラーは、DB2 Universal Driver コネクターを使用する場合に発生する可能性があります。WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

T2 ネイティブ・ライブラリー db2jcct2zos のロードの失敗

このエラーは、DB2 Universal Driver コネクターを使用する場合に発生する可能性があります。WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOD2 からロードできません。

エラー・メッセージ: BBOO0220E:

```
error message: BBOO0220E:
[SCA.APPLICATION.mdcell.Bus:mdnodea.mdsr01a-SCA.APPLICATION.mdcell.Bus]
CWSIS0002E: The messaging engine encountered an exception while
starting.
Exception: com.ibm.ws.sib.msgstore.PersistenceException:
CWSIS1501E: The data source has produced an unexpected exception:
java.sql.SQLException: Failure in loading T2 native library
db2jcct2zos, reason: java.lang.UnsatisfiedLinkError:
/pp/db2v8/UK14852/jcc/lib/libdb2jcct2zos.so:
EDC5157I An internal error has occurred. (errno2=0x0BDF03B2)DSRA0010E:
SQL State = null, Error Code = -99,999DSRA0010E: SQL State = null,
Error Code = -99,999
com.ibm.ws.sib.utils.ras.SibMessage
com.ibm.ws.sib.utils.ras.SibMessage
```

libdb2jcct2zos.so をロードできなかった理由として、複数の原因が考えられます。

WAS サーバー・プロセスの STEPLIB に DB2 ライブラリーがないことが、一般的なエラーとして考えられます。このような障害は、アクセスしている DB2 システムで、DB2 Universal Driver が完全に構成されていないなどの、より大きな問題の症状であることも考えられます。

DB2 Universal Driver をインストールするためのすべての手順が、DB2 システムで実行されていることを確認してください。

DB2 Universal Driver のインストール手順は、DB2 インフォメーション・センター (IBM Data Server Driver for JDBC and SQLJ のインストール) にあります。

関連資料

269 ページの『WebSphere Process Server のエラー』

構成タスクのいずれかで問題が発生した場合は、その問題についての情報を次の 3 つの主な情報源から入手できます。

37 ページの『WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報』

WebSphere Process Server for z/OS のメッセージ参照情報には、インストール・スクリプトまたは構成スクリプトの実行時に表示されるメッセージ・コードをリストします。

286 ページの『ログ・ファイル』

製品のインストールおよび構成処理中に、さまざまなログ・ファイルが作成されます。

279 ページの『検査エラー』

インストールの検査時には、このセクションで説明する問題が発生する場合があります。

280 ページの『管理コンソールにリソースが表示されない』

インストールしたアプリケーションがシステムに存在することを確認する場合は、インストールしたアプリケーション・セクションの下にリストされない可能性があります。リストにアプリケーションが表示されない場合は、管理コンソールからログアウトし、もう一度ログインしてください。

281 ページの『Common Base Event ブラウザー検証のトラブルシューティング』

Network Deployment 構成で Common Base Event ブラウザーをテストする場合、次のようなさまざまな理由によってエラーが発生します。

277 ページの『bpeconfig.jacl: TaskContainer のインストール中に発生したエラー』

bpeconfig.jacl エラーは、通常、無効な値が入力された場合に発生します。グループ名の入力に誤りがあった場合、および訂正を行う際に Backspace キーではなく Delete キーを使用した場合には、以下のようなエラー・メッセージが表示されます。以下の例では、入力内容が MKHTSMG となっていますが、入力された値に無効な文字「MKSMG[D[D[D[D[CHTSMG」が含まれています。

データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

```
error message: BB000222I: DSRA8208I: JDBC driver type : 2
com.ibm.ws.exception.RuntimeWarning:
com.ibm.ws.runtime.component.binder.ResourceBindingException:
invalid configuration passed to resource binding logic. REASON: Invalid
Configuration!
The DataSource: DB2 Universal JDBC Driver DataSource has a null
RelationalResourceAdapter property.
```

以下の手順を実行して、冗長なデータ・ソースを削除してください。

1. WebSphere 管理コンソールにログインし、「リソース」→「JDBC プロバイダー」にナビゲートします。
2. スコープを「サーバー」に設定し、「適用」をクリックします。
3. 「DB2 Universal JDBC Driver Provider」という JDBC プロバイダーをクリックします。
4. 右側にある「データ・ソース」へのリンクをクリックします。
5. 3 つのデータ・ソースを示したリストが表示されます。「DB2 Universal JDBC Driver Datasource」の横にあるボックスにチェック・マークを付け、「削除」ボタンをクリックします。
6. 構成の変更を保存し、WebSphere サーバーを再始動します。

関連資料

269 ページの『WebSphere Process Server のエラー』

構成タスクのいずれかで問題が発生した場合は、その問題についての情報を次の 3 つの主な情報源から入手できます。

37 ページの『WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報』

WebSphere Process Server for z/OS のメッセージ参照情報には、インストール・スクリプトまたは構成スクリプトの実行時に表示されるメッセージ・コードをリストします。

286 ページの『ログ・ファイル』

製品のインストールおよび構成処理中に、さまざまなログ・ファイルが作成されます。

279 ページの『検査エラー』

インストールの検査時には、このセクションで説明する問題が発生する場合があります。

280 ページの『管理コンソールにリソースが表示されない』

インストールしたアプリケーションがシステムに存在することを確認する場合は、インストールしたアプリケーション・セクションの下にリストされない可能性があります。リストにアプリケーションが表示されない場合は、管理コンソールからログアウトし、もう一度ログインしてください。

281 ページの『Common Base Event ブラウザー検証のトラブルシューティング』

Network Deployment 構成で Common Base Event ブラウザーをテストする場合、次のようなさまざまな理由によってエラーが発生します。

277 ページの『bpeconfig.jacl: TaskContainer のインストール中に発生したエラー』

bpeconfig.jacl エラーは、通常、無効な値が入力された場合に発生します。グループ名の入力に誤りがあった場合、および訂正を行う際に Backspace キーではなく Delete キーを使用した場合には、以下のようなエラー・メッセージが表示されます。以下の例では、入力内容が MKHTSMG となっていますが、入力された値に無効な文字「MKSMG[D[D[D[D[CHTSMG」が含まれています。

SQLCODE = -471

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

SYSIBM.SYSROUTINES テーブルで、SYSIBM.SYSTABLES 内の WLM_ENVIRONMENT の WLM 名が、ストアード・プロシージャ・アドレス・スペース JCL で使用されている名前と一致しません。

DB2 Universal Driver のインストール手順については、DB2 インフォメーション・センターを参照してください。

```
ExtendedMessage: BB000220E:  
[CommonEventInfrastructure_Bus:mdnodea.mdsr01a-CommonEventInfrastructure_Bus]  
CWSIS0002E: The messaging engine encountered an exception while  
starting.  
Exception: com.ibm.ws.sib.msgstore.PersistenceException:  
CWSIS1501E: The data source has produced an unexpected exception:  
com.ibm.db2.jcc.t2zos.y:[IBM/DB2][T2zos/2.9.32]  
v.readExecuteCallInternal: nativeExecuteCall:5587:  
DB2 engine SQL error, SQLCODE = -471, SQLSTATE = 55023,  
error tokens = SYSIBM.SYLTABLES;00E7900C
```

関連資料

269 ページの『WebSphere Process Server のエラー』

構成タスクのいずれかで問題が発生した場合は、その問題についての情報を次の 3 つの主な情報源から入手できます。

37 ページの『WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報』

WebSphere Process Server for z/OS のメッセージ参照情報には、インストール・スクリプトまたは構成スクリプトの実行時に表示されるメッセージ・コードをリストします。

286 ページの『ログ・ファイル』

製品のインストールおよび構成処理中に、さまざまなログ・ファイルが作成されます。

279 ページの『検査エラー』

インストールの検査時には、このセクションで説明する問題が発生する場合があります。

280 ページの『管理コンソールにリソースが表示されない』

インストールしたアプリケーションがシステムに存在することを確認する場合は、インストールしたアプリケーション・セクションの下にリストされない可能性があります。リストにアプリケーションが表示されない場合は、管理コンソールからログアウトし、もう一度ログインしてください。

281 ページの『Common Base Event ブラウザー検証のトラブルシューティング』

Network Deployment 構成で Common Base Event ブラウザーをテストする場合、次のようなさまざまな理由によってエラーが発生します。

277 ページの『bpeconfig.jacl: TaskContainer のインストール中に発生したエラー』

bpeconfig.jacl エラーは、通常、無効な値が入力された場合に発生します。グループ名の入力に誤りがあった場合、および訂正を行う際に Backspace キーではなく Delete キーを使用した場合には、以下のようなエラー・メッセージが表示されます。以下の例では、入力内容が MKHTSMG となっていますが、入力された値に無効な文字「MKSMG[D[D[D[CHTSMG」が含まれています。

SQL コード -204 および -516

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

```
error message: BB000220E: SCHD0125E: Unexpected exception
while processing the acquireLease operation:
com.ibm.ws.leasemanager.LeaseException: SCHD0300E:
Error during Database operation,
localized message is
```

```
-:nativePrepareInto:1377:
DB2 engine SQL error, SQLCODE = -204, SQLSTATE = 42704,
error tokens = MDDBU.WSCH_LMGR,
Vendor Error Code is -204, ANSI-92 SQLState is 42704, cause:
[IBM/DB2][T2zos/2.9.32]T2zosPreparedStatement.readDescribeInput_
:nativeDescribeInput:2006:
DB2 engine SQL error, SQLCODE = -516, SQLSTATE = 26501,
error tokens =
```

..

```
..
com.ibm.db2.jcc.t2zos.y:
[IBM/DB2][T2zos/2.9.32]T2zosPreparedStatement.readDescribeInput:2006:
DB2 engine SQL error, SQLCODE = -516, SQLSTATE = 26501, ...
```

関連資料

269 ページの『WebSphere Process Server のエラー』
構成タスクのいずれかで問題が発生した場合は、その問題についての情報を次の 3 つの主な情報源から入手できます。

37 ページの『WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報』

WebSphere Process Server for z/OS のメッセージ参照情報には、インストール・スクリプトまたは構成スクリプトの実行時に表示されるメッセージ・コードをリストします。

286 ページの『ログ・ファイル』

製品のインストールおよび構成処理中に、さまざまなログ・ファイルが作成されます。

279 ページの『検査エラー』

インストールの検査時には、このセクションで説明する問題が発生する場合があります。

280 ページの『管理コンソールにリソースが表示されない』

インストールしたアプリケーションがシステムに存在することを確認する場合は、インストールしたアプリケーション・セクションの下にリストされない可能性があります。リストにアプリケーションが表示されない場合は、管理コンソールからログアウトし、もう一度ログインしてください。

281 ページの『Common Base Event ブラウザー検証のトラブルシューティング』

Network Deployment 構成で Common Base Event ブラウザーをテストする場合、次のようなさまざまな理由によってエラーが発生します。

277 ページの『bpeconfig.jacl: TaskContainer のインストール中に発生したエラー』
bpeconfig.jacl エラーは、通常、無効な値が入力された場合に発生します。グループ名の入力に誤りがあった場合、および訂正を行う際に Backspace キーではなく Delete キーを使用した場合には、以下のようなエラー・メッセージが表示されます。以下の例では、入力内容が MKHTSMG となっていますが、入力された値に無効な文字「MKSMG[D[D[D[D[CHTSMG」が含まれています。

ロックの獲得および損失についての SIB メッセージの繰り返し

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

```
ExtendedMessage: BB000222I:
[CommonEventInfrastructure_Bus:mdnodea.mdsr01a-CommonEventInfrastructure_Bus]
CWSIS1538I: The messaging engine, ME_UID=68E9550CE7780888,
INC_UID=5f244052b02f04b4,
is attempting to obtain an exclusive lock on the data store.
..
ExtendedMessage: BB000222I:
[CommonEventInfrastructure_Bus:mdnodea.mdsr01a-CommonEventInfrastructure_Bus]
CWSIS1546I: The messaging engine, ME_UID=68E9550CE7780888,
INC_UID=5f244052b02f04b4,
has lost an existing lock or failed to gain an initial lock on the database
```

これらのエラー・メッセージは、データ・ストアへのアクセスに問題があることを示します。データ・ソースで `fixWPSvars.jacl` が作成されていること (jdbc/MEdatasource) を確認してください。またデータ・ソースに、JCA 認証別名が関連付けられていることを確認してください。データ・ソースに関連付けられている JCA 認証別名がない場合、データベース・アクセスは、デフォルトのサーバント領域のユーザー ID を使用し、存在しない `MKASRU` というテーブルを見つけようとしています。

このエラーは、`fixWPSvars.jacl` の実行時に `-sibauth` オプションがコード化されていなかったことが原因で発生する可能性があります。この問題は多くの方法で修正できます。

- SIB で使用される JDBC データ・ソースを `WPSDBAlias` と呼ばれる JCA 認証別名に関連付ける。
- 新しい JCA 認証別名を作成して、JDBC と関連付ける。`-sibauth` オプションを使用して `fixWPSvars.jacl` を再実行し、`WPSDBAuth` を JCA 認証別名として指定するか、または WebSphere 管理コンソールを使用して変更を加え、`WPSDBAlias` を別名として指定します。

以下の手順は、DB2 にアクセスするために SIB で使用される新しい JCA 認証別名を作成する方法を説明しています。

1. WebSphere 管理コンソールを開き、「セキュリティ」→「グローバル・セキュリティ」にナビゲートします。
2. 「追加プロパティ」の下で「J2C 認証データ」へのリンクをクリックします。
3. 「新規」ボタンをクリックします。
4. 別名とする名前を入力し、その別名のユーザー ID とパスワードを入力します。
5. 「OK」をクリックします。
6. 「保管」をクリックします。

以下のステップは、サービス統合バスを、作成した認証別名に関連付ける方法を示しています。

1. WebSphere 管理コンソールを開き、「サービス統合」→「バス」にナビゲートします。
2. リストの先頭のバスをクリックします。
3. 次のパネルで、「メッセージング・エンジン」をクリックします。
4. メッセージング・エンジンへのハイパーリンクをクリックします。
5. 「追加プロパティ」の下で「データ・ストア」へのリンクをクリックします。
6. 「認証別名」フィールドのドロップダウン・リスト・ボックスを展開し、前に作成した別名を選択します。
7. 「OK」をクリックし、構成への変更を保存します。
8. ページの先頭のリンクをクリックし、ナビゲートしてバスのリストに戻ります。
9. リスト内の次のバスを選択し、同じ手順を繰り返します。残りのバスについても、同じく繰り返します。

10. 有効な JCA 認証別名を参照するようにすべてのバスを更新したら、サーバーを停止してから再始動します。

関連資料

269 ページの『WebSphere Process Server のエラー』

構成タスクのいずれかで問題が発生した場合は、その問題についての情報を次の 3 つの主な情報源から入手できます。

37 ページの『WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報』

WebSphere Process Server for z/OS のメッセージ参照情報には、インストール・スクリプトまたは構成スクリプトの実行時に表示されるメッセージ・コードをリストします。

286 ページの『ログ・ファイル』

製品のインストールおよび構成処理中に、さまざまなログ・ファイルが作成されます。

279 ページの『検査エラー』

インストールの検査時には、このセクションで説明する問題が発生する場合があります。

280 ページの『管理コンソールにリソースが表示されない』

インストールしたアプリケーションがシステムに存在することを確認する場合は、インストールしたアプリケーション・セクションの下にリストされない可能性があります。リストにアプリケーションが表示されない場合は、管理コンソールからログアウトし、もう一度ログインしてください。

281 ページの『Common Base Event ブラウザー検証のトラブルシューティング』

Network Deployment 構成で Common Base Event ブラウザーをテストする場合、次のようなさまざまな理由によってエラーが発生します。

『bpeconfig.jacl: TaskContainer のインストール中に発生したエラー』

bpeconfig.jacl エラーは、通常、無効な値が入力された場合に発生します。グループ名の入力に誤りがあった場合、および訂正を行う際に Backspace キーではなく Delete キーを使用した場合には、以下のようなエラー・メッセージが表示されます。以下の例では、入力内容が MKHTSMG となっていますが、入力された値に無効な文字「MKSMG[D[D[D[CHTSMG」が含まれています。

bpeconfig.jacl: TaskContainer のインストール中に発生したエラー

bpeconfig.jacl エラーは、通常、無効な値が入力された場合に発生します。グループ名の入力に誤りがあった場合、および訂正を行う際に Backspace キーではなく Delete キーを使用した場合には、以下のようなエラー・メッセージが表示されます。以下の例では、入力内容が MKHTSMG となっていますが、入力された値に無効な文字「MKSMG[D[D[D[CHTSMG」が含まれています。

```
..
..
[] Install the task container [Yes/no]? Yes
[adminHTMUsers] User(s) to add to role TaskSystemAdministrator
(separator is pipe,
'|') []: MKADMIN
[adminHTMGroups] Group(s) to add to role TaskSystemAdministrator
(separator is pipe,
'|') []: MKSMADMG|MKCFG
[monitorHTMUsers] User(s) to add to role TaskSystemMonitor
(separator is pipe, '|')
```

```

[]:
[monitorHTMGroups] Group(s) to add to role TaskSystemMonitor
(separator is pipe, '|')
[]: MKSMG[D[D[D[CHTSMG
[jmsHTMRunAsUser] Run-as UserId for role EscalationUser
[MKADMIN]: MKHTSM
[jmsHTMRunAsPwd] MKHTSM's password []: *****
[auto:mqType] Use WebSphere default messaging or
WebSphere MQ? WPM
task.ear install options: -appname "TaskContainer_mkc101"
-usedefaultbindings
-defaultbinding.ejbjndi.prefix ejb/htm -cluster
"mkc101"
-BindJndiForEJBMessageBinding {"TaskContainer" "HTMScheduler"
"taskejb.jar,META-INF/ejb-jar.xml" ""
"eis/HTMInternalActivationSpec"
"jms/HTMIntQueue" ""} -MapResRefToEJB {
{"TaskContainer"
"GenericHumanTaskManagerEJB" "taskejb.jar,META-INF/
ejb-jar.xml" "jdbc/BPEDB"
"javax.sql.DataSource" "jdbc/BPEDB_mkc101"} {"TaskContainer"
"TaskContainerStartupBean" "taskejb.jar,META-INF/ejb-jar.xml"
"jdbc/BPEDB"
"javax.sql.DataSource" "jdbc/BPEDB_mkc101"} -MapResEnvRefToRes
{ {"TaskContainer"
"TaskContainerStartupBean" "taskejb.jar,META-INF/ejb-jar.xml"
"jms/HTMHoldQueue"
"javax.jms.Queue" "jms/HTMHldQueue"} {"TaskContainer"
"TaskContainerStartupBean"
"taskejb.jar,META-INF/ejb-jar.xml" "scheduler/BPCScheduler"
"com.ibm.websphere.scheduler.Scheduler" "BPEScheduler"}
} -MapRolesToUsers
{"TaskSystemAdministrator" "AppDeploymentOption.No"
"AppDeploymentOption.No"
"MKADMIN" "MKSMADMG|MKCFG"} {"TaskSystemMonitor"
"AppDeploymentOption.No"
"AppDeploymentOption.No" "" "MKSMG[D[D[D[CHTSMG"}
{"EscalationUser"
"AppDeploymentOption.No" "AppDeploymentOption.Yes"
"" ""} -MapRunAsRolesToUsers
{"EscalationUser" "*****" "MKHTSM"}
An error occurred installing TaskContainer_mkc101:
..
..
com.ibm.ws.scripting.ScriptingException: WASX7132E: Application install for
/wasmkconfig/mkcell/mkdmnode/DeploymentManager/installableApps/task.ear failed: see
previous messages for details. Discarding changes.

```

デプロイメント・マネージャーのサーバント・ジョブのログを見れば、以下のよう
な関連するエラーも確認することができます。

```

error message: FFDC0010I: FFDC closed incident stream file
/wasmkconfig/mkcell/mkdmnode/DeploymentManager/profiles/default/logs/ff
dc/mkcell_mkdmnode_dmgr_STC12532_MKDMGRS_06.11.13_04.05.37_1.txt
com.ibm.etools.j2ee.commonarchivecore.exception.ResourceLoadException: IWAE0007E
Could not load resource "META-INF/ibm-application-bnd.xmi" in archive
"/wasmkconfig/mkcell/mkdmnode/DeploymentManager/profiles/default/temp/app35301.ear"
!Stack_trace_of_nested_exce!
com.ibm.etools.j2ee.exception.WrappedRuntimeException:
Exception occurred loading META-INF/ibm-application-bnd.xmi
!Stack_trace_of_nested_exce!
Wrapped exception
org.xml.sax.SAXParseException: An invalid XML character (Unicode: 0x1b) was found in
the value of attribute "name" and element is "groups".

```

関連資料

270 ページの『T2 ネイティブ・ライブラリー db2jcc2zos のロードの失敗』
このエラーは、DB2 Universal Driver コネクタを使用する場合に発生する可能性があり、WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOAD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

検査エラー

インストールの検査時には、このセクションで説明する問題が発生する場合があります。

関連資料

270 ページの『T2 ネイティブ・ライブラリー db2jcc2zos のロードの失敗』
このエラーは、DB2 Universal Driver コネクタを使用する場合に発生する可能性があります。WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOAD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

管理コンソールにリソースが表示されない

インストールしたアプリケーションがシステムに存在することを確認する場合は、インストールしたアプリケーション・セクションの下にリストされない可能性があります。リストにアプリケーションが表示されない場合は、管理コンソールからログアウトし、もう一度ログインしてください。

構成したサービス統合バスが表示されない場合は、管理コンソールからログアウトし、もう一度ログインしてください。

関連資料

270 ページの『T2 ネイティブ・ライブラリー db2jcc2zos のロードの失敗』
このエラーは、DB2 Universal Driver コネクタを使用する場合に発生する可能性があり、WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOAD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

Common Base Event ブラウザー検証のトラブルシューティング

Network Deployment 構成で Common Base Event ブラウザーをテストする場合、次のようなさまざまな理由によってエラーが発生します。

CWLCB0020E: NameNotFoundException により Common Event Infrastructure が使用不可 (Common Event Infrastructure is unavailable with NameNotFoundException)

- Common Base Event ブラウザーで完全修飾 JNDI 名が使用されなかった。
- CEI テーブルのスキーマ名が、event および eventcat データ・ソースで使用されている JCA 認証別名のユーザー ID と等しくない。

Network Deployment 構成で Common Base Event ブラウザーを初めて検査するときに表示される可能性があるメッセージの例を以下に示します。

WebSphere Application Server のサーバント・ログに表示される可能性があるメッセージの例を以下に示します。

```
javax.naming.NameNotFoundException:  
Context: mkcell/nodes/mkdmnode/servers/dmgr,  
name: ejb/com/ibm/events/access/EventAccess:  
First component in name com/ibm/events/access/EventAccess not found.  
Root exception is org.omg.CosNaming.NamingContextPackage.NotFound:  
IDL:omg.org/CosNaming/NamingContext/NotFound:1.0
```

CEI が Network Deployment 構成内で構成されている場合には、イベント・データ・ストア内に完全修飾 JNDI 名が必要です。この問題を解決するには、EventAccess EJB の完全修飾 JNDI 名を指定します。

CWLCB0020E: CEIDS0035E により Common Event Infrastructure が使用不可 (Common Event Infrastructure is unavailable with CEIDS0035E)

Network Deployment 構成で Common Base Event ブラウザーを初めて検査するときには、次のエラー・メッセージも表示される可能性があります。

```
error message: CEIDS0035E The implementation class that supports the
configured relational database system cannot be loaded.
Implementation class name:
com.ibm.events.datastore.impl.Db2UniversalDriverImpl
Relational database name: DB2
Database version: DSN08015
com.ibm.events.datastore.impl.DatabaseSpecificsFactory
handleCreateException(String, String, String, Exception)
```

これらのエラー・メッセージは、クラスのロードに問題があることを示しますが、原因は通常、データベースへのアクセスの問題です。場合によっては、問題の診断に役立つ -204 のような DB2 戻りコードと共にいくつかのエラー・メッセージが表示されることもあります。CEIDS0035E メッセージしか表示されない場合もあります。

解決策 1

FFDC ファイル内で -551 エラーを見つけた場合は、DB2 で適切な GRANT ステートメントを発行することによって、報告されている許可障害を訂正してください。

例えば、CEI データベースで作成された CEI テーブル、ビュー、および索引のすべてに対して、GRANT ALL ON TABLE MKCELL. TO MKDBU を発行することができます。その後、WebSphere Application Server を再始動します。ただし、Common Base Event ブラウザーではエラー・メッセージ CWLCB0020E が表示され、サーバント・ログではエラー・メッセージ CEIDCS0035E が依然として表示される場合があります。

以下の例は、FFDC ログに報告された別の DB2 エラーで、同一のエラーがサーバント・メッセージ・ログに報告されます。

```
Exception = com.ibm.db2.jcc.t2zos.y
Source = com.ibm.ws.rsadapter.jdbc.WSJdbcConnection.prepareStatement
probeid = 1584
Stack Dump = com.ibm.db2.jcc.t2zos.y:
[IBM/DB2][T2zos/2.9.32]T2zosPreparedStatement.readPrepareDescribeOutput
_:nativePrepareInto:1377:DB2 engine SQL error, SQLCODE = -204, SQLSTATE
= 42704, error tokens = MKDBU.CEI_T_Common Base Event_MAP
```

-204 コードは、リソースの割り振り失敗 (つまり MKDBU.CEI_T_Common Base Event_MAP が見つからなかったこと) を示します。MKCELL というスキーマ名で CEI テーブルが作成されましたが、この障害は、Common Base Event ブラウザーが MKDBU.CEI_T_Common Base Event_MAP にアクセスしようとしていることを示しています。このユーザー ID (MKDBU) は、event および eventcat データ・ソースで使用されている JCA 認証別名のユーザー ID です。

問題は、CEI コンポーネントがデータ・ソースに関連付けられている別名を調べ、その別名を完全修飾 SQL を発行するために使用することです。これは、データ・ソースの currentSchema カスタム・プロパティで設定するすべての値が無視されることを意味します。残念ながら現時点では、DB2 内のすべての CEI オブジェクト

トのスキーマが event および eventcat データ・ソースで使用される JCA 認証別名のユーザー ID と等しくなるように、CEI を構成する必要があります。

この問題は 2 つの方法で解決できます。

- データベース内のスキーマを、event データ・ソースと eventcat データ・ソースで使用されている JCA 認証別名ユーザー ID のユーザー ID と一致させる (解決策 2a)。
- 既存の CEI テーブルのスキーマと一致するユーザー ID を持つ event データ・ソースと eventcat データ・ソースの新しい JCA 認証別名を使用する (解決策 2b)。この方法は、CEI テーブルのスキーマを JCA 認証別名ユーザー ID と長期的に一致させたくない場合に使用できます。

解決策 2a

CEI データベースを除去してから再作成します。その後、CEI DDL を再実行 (メタデータの挿入およびカタログのシードも行われる) しますが、すべての CREATE ステートメントに JCA 認証別名ユーザー ID と等しいスキーマ名を指定してください。

解決策 2b

CEI テーブルを除去して再作成する方法を使用しない場合は、次のステップを実行します。

1. 現在使用しているスキーマ名と等しい RACF ユーザー ID を作成します。
 2. WebSphere 管理コンソールを使用して、新しい JCA 認証別名を定義し、その別名に RACF ユーザー ID とパスワードを設定します。作成する JCA 別名に CEI ユーザー ID およびパスワードのスキーマを設定します。
 3. 「リソース」→「JDBC プロバイダー」にナビゲートし、クラスターとサーバーのどちらに CEI をデプロイしたかに応じて、スコープを設定します。
 - CEI EventServer アプリケーションがクラスターにデプロイされている場合は、スコープをクラスターに設定します。
 - CEI EventServer アプリケーションがサーバーにデプロイされている場合は、スコープをサーバーに設定します。
- 「Event_DB2ZOS_JDBC_Provider」をクリックします。
4. 「追加プロパティ」の下で「データ・ソース」をクリックします。
 5. 「イベント」をクリックします。
 6. スクロールダウンし、「コンポーネント管理認証別名」フィールドのドロップダウン・リスト・ボックスで、作成した新しい JCA 認証別名を選択します。
 7. 「OK」をクリックします。
 8. eventcat データ・ソースにナビゲートし、同じ変更を行います。
 9. 構成の変更を保存し、サーバーまたはクラスターを再始動します。

この問題が発生したのは、JCA 認証別名ユーザー ID と一致するスキーマを持つテーブルを DB2 内に作成したくないという理由からであるため、2a ではなく 2b の解決策を実行するほうが利点があります。CEI テーブルには、使用するスキーマが既に存在しており、したがって、スキーマを変更しても意味がありません。データ・ソースで currentSchema プロパティを使用できるようにする CEI に対するフ

フィックスが提供されている場合は、データ・ソースを容易に切り替え、元の JCA 認証別名を使用する設定に戻すことができます。

もちろん、解決策 2b を実行した場合でも、CEI スキーマと等しい JCA 認証別名を一時的に使用することになります。CEI 用のフィックスがあり、元の JCA 認証別名 (スキーマと同じではない別名) に切り替える場合は、CEI スキーマと一時的に一致するよう作成した JCA 認証別名用の RACF ユーザー ID および別名を削除できます。

CWLCB0020E: CORBA NO MEMORY により Common Event Infrastructure が使用不可 (Common Event Infrastructure is unavailable with CORBA NO MEMORY)

サーバント・ログには、次のエラーが記録される場合があります。

```
java.rmi.RemoteException: CORBA NO_MEMORY exception
```

WebSphere Process Server をインストールしたばかりの時点では、この問題はほとんど発生しませんが、「検索するイベントの最大数」フィールドに大きい数値を指定したイベントが発生すると、メモリーについての問題が発生する可能性があるという点を認識しておくことは有益です。

この問題は、次の 2 つの方法のいずれかで解決できます。

- 検索するイベントの最大数を 500 から 100 に減らしてから、日時クエリーを使用してイベントの次のブロックを検索する。
- EventServer アプリケーションが実行されているサーバーで、サーバント領域の JVM が使用する最小および最大ヒープ・サイズを増やす。

注: 非常に大きな CEI イベントがある場合、イベントの数はそれほど多くないとしても、メモリー不足の例外が発生する可能性があります。この問題を解決するには、初期ヒープおよび最大ヒープのサイズを増やしてください。

関連資料

270 ページの『T2 ネイティブ・ライブラリー db2jcct2zos のロードの失敗』
このエラーは、DB2 Universal Driver コネクタを使用する場合に発生する可能性があります。WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

WebSphere Process Server for z/OS のインストールおよび構成のメッセージ参照情報

WebSphere Process Server for z/OS のメッセージ参照情報には、インストール・スクリプトまたは構成スクリプトの実行時に表示されるメッセージ・コードをリストします。

インストール・エラー・メッセージについて

WebSphere Process Server for z/OS メッセージ・コードのトラブルシューティングを行うときに、「説明」フィールドと「ユーザー応答」フィールドのデータを使用してください。

メッセージ・コードは CWPIZyyyyz という形式で表示されます。各部位には次のような意味があります。

- CWPIZ = WebSphere Process Server for z/OS メッセージ接頭語
- yyyy = 番号に割り当てられている数値 ID
- z = メッセージ・タイプの記述子 (E、I、または W)。各記述子は以下のタイプを示します。
 - E = エラー・メッセージ
 - I = 通知メッセージ
 - W = 警告メッセージ

WebSphere Process Server for z/OS インストール時のエラー・メッセージのリストについては、参照資料の『メッセージ (Messages)』の部分に記載されている CWPIZ を参照してください。

WebSphere Process Server for z/OS インストール・エラー・メッセージは、ランタイム・ディレクトリーの zSMPInstall.log ファイルに書き込まれます。このログ・ファイルの標準のデフォルトの場所は、/WebSphere/V7R0/AppServer/logs/wbi/zSMPInstall.log です。

WebSphere Process Server for z/OS 構成エラー・メッセージは、ランタイム・ディレクトリーの zWPSConfig.log ファイルおよび zWESBConfig.log ファイルに書き込まれます。これらのログ・ファイルの標準のデフォルトの場所は、それぞれ /WebSphere/V7R0/AppServer/logs/wbi/zWESBConfig.log および /WebSphere/V7R0/AppServer/logs/wbi/zWPSConfig.log です。

関連資料

270 ページの『T2 ネイティブ・ライブラリー db2jcc2zos のロードの失敗』
このエラーは、DB2 Universal Driver コネクタを使用する場合に発生する可能性があります。WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOAD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。

ログ・ファイル

製品のインストールおよび構成処理中に、さまざまなログ・ファイルが作成されます。

目的

製品のインストールおよび構成処理中に問題が発生した場合は、該当するログを参照してください。

ログ・ファイルにリダイレクトされる標準出力

標準出力メッセージは、コマンド行引数を検証するアクションの開始および完了などの高位のアクションを報告します。

デフォルトでは、これらのメッセージは、製品インストール・スクリプトを実行した画面に直接表示されます。ただし、コマンド行の末尾でリダイレクト記号とファイル名を使用して、これらのメッセージをリダイレクトすることができます。例えば、インストール・コマンドの末尾で `>>run.log` を指定すると、現在の作業ディレクトリーの `run.log` という名前のファイルに標準出力メッセージをリダイレクトします。

標準出力メッセージは、ログおよびトレース・ファイルがオープンされる前に発生した重大エラーも報告します。例えば、次のメッセージ・ブロックは、必要なキーワード (`-runtime`) がインストール・コマンドに含まれていなかった場合に表示されます。

```
parsing command arguments...
CWPIZ0101E -runtime keyword and value not specified on command line.
com.ibm.ws390.installer.InstallFailureException: -runtime keyword and value not specified
CWPIZ0017E install task failed.
```

ログ・ファイル

これらのメッセージには、標準出力に書き込まれたメッセージが含まれますが、インストーラー・プログラムが使用した追加情報および設定も提供します。

例えば、次のログの一部は、使用される応答プロパティおよびそれらの値を示します。また、シンボリック・リンクの作成時に使用されるソース・ディレクトリーおよびターゲット・ディレクトリーも示します。

```
response property: profilePath=/WebSphere/V7R0/AppServer/profiles/default
response property: nodeName=SY1
response property: scaSecurityPassword=ibmuser
response property: dbType=Derby
response property: ceiSampleJmsUser=ibmuser
response property: scaSecurityUserId=ibmuser
response property: configureScaSecurity=true
response property: mqUser=ibmuser
response property: serverName=server1
response property: adminBFMGGroups=ibmuser
response property: profileName=default
response property: dbCreateNew=true
response property: ceiSampleJmsPwd=ibmuser
response property: cellName=SY1
response property: dbLocation=/WebSphere/V7R0/AppServer/derby/databases/WBIDB
response property: mqPwd=ibmuser
response property: was.install.root=/WebSphere/V7R0/AppServer
response property: augment=
response property: ceiDbProduct=DERBY_V51_1
response property: wbi.install.root=/WebSphere/V7R0/AppServer
response property: ceiSampleServerName=server1
response property: templatePath=/WebSphere/V7R0/AppServer/profileTemplates/default.*
response property: dbName=WBIDB
構成のセットアップが完了しました
シンボリック・リンクの作成中...
Source=/usr/lpp/zWPS/V7R0

Target=/WebSphere/V7R0/AppServer
シンボリック・リンクの作成が完了しました
インストール後ファイル更新の実行中...
インストール後更新が完了しました
構成マネージャー更新の実行中...
構成マネージャー更新が完了しました
```

トレース・ファイル

これらのメッセージは、ランタイム・ディレクトリー内の **zWPSInstall.trace** ファイルに書き込まれます。

以下の例では、いくつかの予備的通知メッセージを示してから、ユーザーがインストール・スクリプト・コマンド行で指定した応答ファイル内で必要な **profileName** プロパティーが検出されなかった (あるいは **-Z** オーバーライドとしても提供されなかった) ことを示す **CWPIZ0322E** エラーを示します。

後続の **CWPIZ0017E** エラー・メッセージは、**zWPSInstall.sh** 実行の最終的な結果を示す一般メッセージです。

```
[8/16/05 17:00:45:380 EDT] 0000000a ManagerAdmin I BB000222I:
TRAS0017I: The startup trace state is *=info.

[8/16/05 17:00:48:230 EDT] 0000000a WPSInstaller I BB000222I:
CWPIZ0044I Begin install task.

[8/16/05 17:00:48:273 EDT] 0000000a WPSInstaller I BB000222I:
CWPIZ0117I WPS installer log data will be written to
/WebSphere/V7R0/AppServer/logs/wbi/zWPSInstall.log.

[8/16/05 17:00:48:282 EDT] 0000000a WPSInstaller I BB000222I:
CWPIZ0024I WPS installer trace data will be written to
/WebSphere/V7R0/AppServer/logs/wbi/zWPSInstall.trace.

[8/16/05 17:00:48:292 EDT] 0000000a WPSInstaller I BB000222I:
CWPIZ0014I Trace specification is "*=all=disabled".

[8/16/05 17:00:48:298 EDT] 0000000a WPSInstaller I BB000222I:
CWPIZ0045I WPS SMP/E root directory is /zrockuser/wbi/Install.

[8/16/05 17:00:48:302 EDT] 0000000a WPSInstaller I BB000222I:
CWPIZ0052I WAS SMP/E root directory is /web/usr/lpp/zWebSphere/V7R0.

[8/16/05 17:00:48:307 EDT] 0000000a WPSInstaller I BB000222I:
CWPIZ0046I Destination application server root directory is
/WebSphere/V7R0/AppServer.

[8/16/05 17:00:48:314 EDT] 0000000a WPSInstaller E BB000220E:
CWPIZ0322E profileName property not specified in Response File.

[8/16/05 17:00:48:318 EDT] 0000000a WPSInstaller E BB000220E:
CWPIZ0017E install task failed.
A trace file from a zWPSInstall.sh executed with the trace specification argument
set to "*=all=enabled" provides additional debugging information. It may contain
information that is meaningful only to a developer. The following is a partial
trace using "*=all=enabled":

***** Start Display Current Environment *****
Host Operating System is z/OS, version 01.04.00
Java version = J2RE 1.4.2 IBM z/OS Persistent Reusable VM build cm142-20050623
(JIT enabled: jitc), Java
Compiler = jitc, Java VM name = Classic VM
was.install.root = /WebSphere/V7R0/AppServer
user.install.root = /WebSphere/V7R0/AppServer/profiles/default
Java Home = /web/usr/lpp/zWebSphere/V7R0/java/J1.4
ws.ext.dirs = /WebSphere/V7R0/AppServer/java/lib:/WebSphere/V7R0/AppServer/java/lib/
ext:/WebSphere/V7R0/AppServer/classes:/WebSphere/V7R0/AppServer/lib:/WebSphere/V7R0/AppServer/
installedChannels:/WebSphere/V7R0/AppServer/lib/ext:/WebSphere/V7R0/AppServer/deploytool/itp
/plugins/com.ibm.etools.ejbdeploy/runtime:/WebSphere/V7R0/AppServer/MQSeries/pubsubroot/lib
Classpath = /zrockuser/bbzconfig.jar:/WebSphere/V7R0/AppServer/lib/admin.jar:/WebSphere/V7R0
/AppServer/lib/ant.jar:/WebSphere/V7R0/AppServer/lib/bootstraps390.jar:/WebSphere/V7R0
/AppServer/lib/bootstrap.jar:/WebSphere/V7R0/AppServer/lib/configmanager.jar:/WebSphere
/V7R0/AppServer/lib/emf.jar:/WebSphere/V7R0/AppServer/lib/ras.jar:/WebSphere/V7R0
/AppServer/lib/runtimefw.jar:/WebSphere/V7R0/AppServer/lib/utills.jar:/WebSphere/V7R0
/AppServer/lib/wasjmx.jar:/WebSphere/V7R0/AppServer/lib/wasproduct.jar:/WebSphere/V7R0
/AppServer/lib/wccm_base.jar:/WebSphere/V7R0/AppServer/lib/wjmxapp.jar:/WebSphere/V7R0
/AppServer/lib/wsanttasks.jar:/WebSphere/V7R0/AppServer/lib/wsexception.jar:/WebSphere
/V7R0/AppServer/lib/wsprofile.jar:/WebSphere/V7R0/AppServer/profiles/default/properties:
/WebSphere/V7R0/AppServer/properties:/WebSphere/V7R0/AppServer/lib/bootstrap.jar:/WebSphere
/V7R0/AppServer/lib/j2ee.jar:/WebSphere/V7R0/AppServer/lib/lmproxy.jar:/WebSphere/V7R0
/AppServer/lib/ur1protocols.jar:/WebSphere/V7R0/AppServer/lib/bootstraps390.jar
Java library path = /web/usr/lpp/zWebSphere/V7R0/java/J1.4/bin/classic/libjvm.so:/web/usr
/lpp/zWebSphere/V7R0/java/J1.4/bin/classic:/web/usr/lpp/zWebSphere/V7R0/java/J1.4/bin/:
/WebSphere/V7R0/AppServer/lib:/WebSphere/V7R0/AppServer/lib:/WebSphere/V7R0/AppServer
/MQSeries/pubsubroot/lib:/mqm/java/bin:/mqm/java/lib:/db2beta/db2710/lib:
/web/usr/lpp/WebSphere/lib:/lib:/usr/lib:/java/J1.3/bin:/java/J1.4/bin:/java/J5.0/bin:
/staf/lib:/WebSphere/V7R0/AppServer/lib:/usr/lib
Current trace specification = *=all
***** End Display Current Environment *****
```

```

[10/3/05 16:35:05:709 EDT] 0000000a ManagerAdmin I BB000222I: TRAS0017I:
The startup trace state is *=all.
[10/3/05 16:35:08:638 EDT] 0000000a WPSInstaller > setup Entry
/web/usr/wbi/zWebSphere/V7R0
APPSERVER
zWPSInstall.sh
-smproot
/web/usr/wbi/zWPS/V7R0
-runtime
/WebSphere/V7R0/AppServer
-response
/web/usr/wbi/zWPS/V7R0/zos.config/standAloneProfile.rsp
-prereonly
-trace
*=all=enabled
[10/3/05 16:35:08:640 EDT] 0000000a WPSInstaller 3 logFileDeleted
true
[10/3/05 16:35:08:660 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0044I:
Begin install task.
[10/3/05 16:35:08:702 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0117I:
WPS installer log data will be written to /WebSphere/V7R0/AppServer/logs/wbi/zWPSInstall.log.
[10/3/05 16:35:08:712 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0024I:
WPS installer trace data will be written to /WebSphere/V7R0/AppServer/logs/wbi/zWPSInstall.trace.
[10/3/05 16:35:08:722 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0014I:
Trace specification is " *=all=enabled".
[10/3/05 16:35:08:726 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0052I:
WAS SMP/E root directory is /web/usr/lpp/zWebSphere/V7R0.
[10/3/05 16:35:08:730 EDT] 0000000a WPSInstaller > checkPathName Entry
/web/usr/wbi/zWPS/V7R0
[10/3/05 16:35:08:731 EDT] 0000000a WPSInstaller < checkPathName Exit
[10/3/05 16:35:08:732 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0045I:
WPS SMP/E root directory is /web/usr/wbi/zWPS/V7R0.
[10/3/05 16:35:08:736 EDT] 0000000a Symlink > isSymlink Entry
/web/usr/wbi/zWPS/V7R0
[10/3/05 16:35:08:737 EDT] 0000000a Symlink 3 absolute path
/web/usr/wbi/zWPS/V7R0
[10/3/05 16:35:08:737 EDT] 0000000a Symlink 3 canonical path
/web/usr/wbi/zWPS/V7R0
[10/3/05 16:35:08:738 EDT] 0000000a Symlink < isSymlink Exit
false
[10/3/05 16:35:08:738 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0046I:
Destination application server root directory is /WebSphere/V7R0/AppServer.
[10/3/05 16:35:08:744 EDT] 0000000a WPSInstaller I BB000222I: CWPIZ0247I:
Response file is /web/usr/wbi/zWPS/V7R0/zos.config/sample.rsp.
[10/3/05 16:35:08:764 EDT] 0000000a WPSInstaller 3 response property
profilePath=/WebSphere/V7R0/AppServer/profiles/default
[10/3/05 16:35:08:765 EDT] 0000000a WPSInstaller 3 response property
nodeName=SY1

```

関連資料

270 ページの『T2 ネイティブ・ライブラリー db2jcc2zos のロードの失敗』

このエラーは、DB2 Universal Driver コネクタを使用する場合に発生する可能性があります。WebSphere Application Server は一部の外部 DB2 モジュールを SDSNLOAD または SDSNLOAD2 からロードできません。

271 ページの『データ・ソースに NULL 値の RelationalResourceAdapter プロパティがある』

以下の例に示すエラーは、拡張スクリプト zWPSConfig.sh の実行後に冗長なデータ・ソースが残されることが原因で発生します。このデータ・ソースは、WebSphere 管理コンソールを使用して安全に削除できます。よく似た名前の JDBC プロバイダーを削除しないように注意してください。

272 ページの『SQLCODE = -471』

このエラーは、DB2 システムでユニバーサル・ドライバーが正しく構成されなかった場合に発生します。

274 ページの『SQL コード -204 および -516』

このエラーは、作成したテーブルおよび索引のスキーマ名と currentSchema プロパティが一致しない場合に発生する可能性があります。エラー・メッセージには、使用されている JCA 認証別名が示されています。

275 ページの『ロックの獲得および損失についての SIB メッセージの繰り返し』

このエラーは、DB2 Universal Driver 構成を訂正し、サーバーを再始動した後に発生する可能性があります。エラー・メッセージは、付加属性領域で継続的に繰り返されます。



Printed in Japan