

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

インストールの計画



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

インストールの計画



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
Planning the Installation

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 2007, 2010.

目次

第 1 章 計画の概要	1	計画のためのシナリオ	99
第 2 章 ソフトウェアのニーズの判別	7	デフォルトのスタンドアロン環境の計画	99
要件の評価	8	カスタム・スタンドアロン環境の計画	101
使用可能なリソースの明確化	9	提供されたパターンの 1 つに基づくデプロイメ ント環境の計画	103
開発およびデプロイメントのバージョン・レベル	10	カスタム・デプロイメント環境の計画	107
WebSphere Process Server と他の WebSphere Application Server 製品との間のインターオペラビリ ティーの計画	11	プロファイル	111
インストールする製品の決定	12	サーバー	112
Business Process Choreographer の構成計画	13	スタンドアロン・サーバー	113
トポロジー、セットアップ、および構成パスの計 画	14	Network Deployment	114
Business Process Choreographer の基本サンプル構 成の作成の計画	20	デプロイメント・マネージャー	115
サンプル組織を含む Business Process Choreographer のサンプル構成の作成の計画	21	管理対象ノード	116
非実動デプロイメント環境構成の計画	22	デプロイメント環境	117
管理コンソールのデプロイメント環境ウィザード を使用するための計画	24	デプロイメント環境パターンの選択	119
カスタム Business Process Choreographer 構成の 計画	29	第 6 章 デプロイメント環境実装の汎用 ステップ	121
Business Process Choreographer の概要	61	第 7 章 WebSphere Process Server の計画: 追加考慮事項	127
必要なセキュリティ権限の明確化	67	第 8 章 エラー防止およびリカバリーの 計画	129
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	68	エラー防止とリカバリーの概要	129
第 3 章 セル内での複数のプラットフォームの 使用	71	エラー防止の計画	130
第 4 章 トポロジーの理解	73	アプリケーション設計の一部としてのエラー防止	130
トポロジーおよびデプロイメントの環境パターン	73	開発の一部としてのエラー防止	136
単一クラスター・トポロジー	79	問題判別方法の文書	141
リモート・メッセージング・トポロジー	81	ソフトウェア適用状況	143
リモート・メッセージングおよびリモート・サポ ート・トポロジー	83	エラー処理方針とソリューション・リカバリー	143
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・トポロジー	86	安定した環境の保守	144
カスタム・トポロジー	88	リカバリー方法の計画	145
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	89	高可用性 (HA)	145
トポロジーを選択するための考慮事項	92	リカバリーの環境および目的	146
第 5 章 デプロイメント環境の計画	97	トランザクションのプロパティーとソリューショ ン・リカバリー	149
		ピア・リカバリー	152
		エクスポート・バインディング	154
		Failed Event Manager について	155
		第 9 章 共存	157
		共存のサポート	157

第 1 章 計画の概要

WebSphere® Process Server を BPM ソリューションの一部として実装する際の関連アクティビティーは、計画した使用法によって異なります。

計画した使用法が計画アクティビティーに与える影響

以下の表に、WebSphere Process Server の使用目的がユーザーの計画アクティビティーの量に与える影響を示します。

表 1. WebSphere Process Server の使用目的

使用目的	構成パスおよび計画アクティビティー
単一サーバー UTE	スタンドアロン・プロファイル構成パス - 計画はほとんど必要ない。
クラスター化されたテスト環境	デプロイメント環境の標準リモート・メッセージングのリモート・サポート・トポロジー - 計画はほとんど必要ない。
柔軟性の高い実稼働環境	デプロイメント環境の標準リモート・メッセージングのリモート・サポート・トポロジー - 計画はほとんど必要ない。
高度に最適化された実稼働環境	詳細な計画が必要 (『高度に最適化された実稼働環境の計画アクティビティー』を参照)。

高度に最適化された実稼働環境の計画アクティビティー

高度に最適化された実稼働環境の計画には、評価アクティビティーおよび設計アクティビティーが含まれます。これらは、製品をインストールおよび構成する方法と、環境を作成および生成する方法に影響を与えます。

計画を正常に行うには、以下に示すような複数の情報源から情報を入手する必要があります。

- ソリューション・アーキテクト

BPM ソリューションのアーキテクチャーを管理し、ビジネスの目標および目的をサポートするために必要な製品、および製品の最適な使用方法について推奨する担当者。

- ビジネス・アナリストおよびビジネス・リーダー

提案された BPM ソリューションのビジネス上の意図を明らかにして、その意図をビジネス・ケイパビリティ・マップおよびプロセス・マップに対応付ける担当者。ビジネス・アナリストおよびビジネス・リーダーは、ビジネス・プロセスに関するオプションを識別して優先順位を付けます。ビジネス・アナリストは、「現状の」プロセス (現行プロセス) および実行すべきプロセス (既存のプロセスに対する改善または変更) を収集して定義します。ビジネス・アナリストは、モックアップ・フォームを作成し、人的関係を検証して視覚化します。また、ソリ

ユーザー・アーキテクトと緊密に作業して、ビジネス・ニーズに対処できる BPM ソリューションをソリューション・アーキテクトが視覚化できるように、重要な情報を提供します。

- システム担当者

BPM ランタイム製品のインストール先マシンおよび SCA アプリケーションのデプロイ先マシンを保守する担当者。システム担当者は、IT 環境をセットアップして、実動用成果物の作成およびデプロイを行います。システム担当者は、BPM ソリューションの正常性をモニターして、プロセス保全性の確保も行います。

- データベース管理者

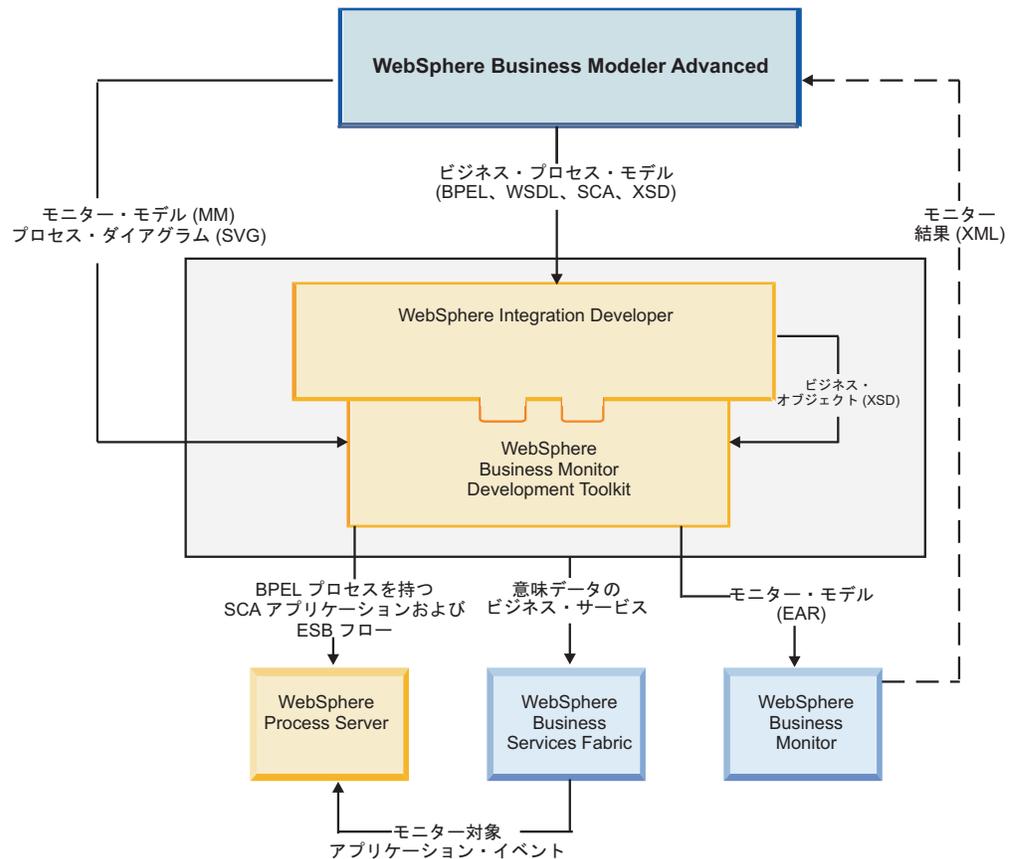
データベース・システムのインストールおよび保守の担当者。

- アプリケーション開発者

WebSphere Integration Developer、WebSphere Business Monitor Development Toolkit、WebSphere Business Modeler などの BPM アプリケーション開発ツールを使用してアプリケーションを作成する担当者。

- BPM ソリューションの専門知識を持っている IBM サービス担当員

以下の図に、IBM WebSphere 製品 (開発ツールおよび関連するランタイム) を使用してビジネス・プロセス・マネジメント (BPM) ソリューションを実装する方法を示します。



この図に示されている製品について詳しくは、『製品ファミリーの概要』を参照してください。

WebSphere Process Server トポロジー設計の重要な側面には、(分散環境での) 物理マシンの数、これらのマシン上のサーバーの数、ビジネスに必要な処理能力を実稼働環境に提供するために必要なクラスターの数が含まれます。さらに、実動トポロジーには、ユーザー・レジストリー (セキュリティー用)、1 つ以上の HTTP サーバー (Web コンテンツ用)、必要なファイアウォール、ロード・バランサーなどの、WebSphere Process Server 以外のその他のサポート・リソースが含まれます。

WebSphere Process Server 実動デプロイメント・トポロジーの計画は、そのトポロジーで実行する作業の要件を満たすことを目的として、慎重に行ってください。考慮すべき点はいくつかあります。以下に例を示します。

- 必要とする物理マシンおよびハードウェア・リソースの数
- ビジネスをサポートするために必要なクラスターおよびクラスター・メンバーの数
- 必要なデータベースの数
- 認証の役割およびセキュリティー上の考慮事項
- デプロイメント環境の実装に使用する方法

以下の図には、WebSphere Process Server の計画、インストール、および構成に関連したタスクの概要を示します。「計画」フェーズで行う決定は、「インストール」フェーズおよび「構成」フェーズの下に示されているタスクに影響を及ぼします。

構成フェーズは、製品構成タスクおよび環境構成タスクという 2 種類のタスクで構成されていることに注意してください。製品構成タスクは、製品プロファイルのセットアップとデータベースの構成に関係しています。環境構成タスクは、デプロイメント環境のセットアップおよび生成のタスクに関係しています。デプロイメント環境とは、ビジネスおよび必要なメッセージング・インフラストラクチャーをサポートする SCA モジュールおよび SCA アプリケーションをホストする、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。

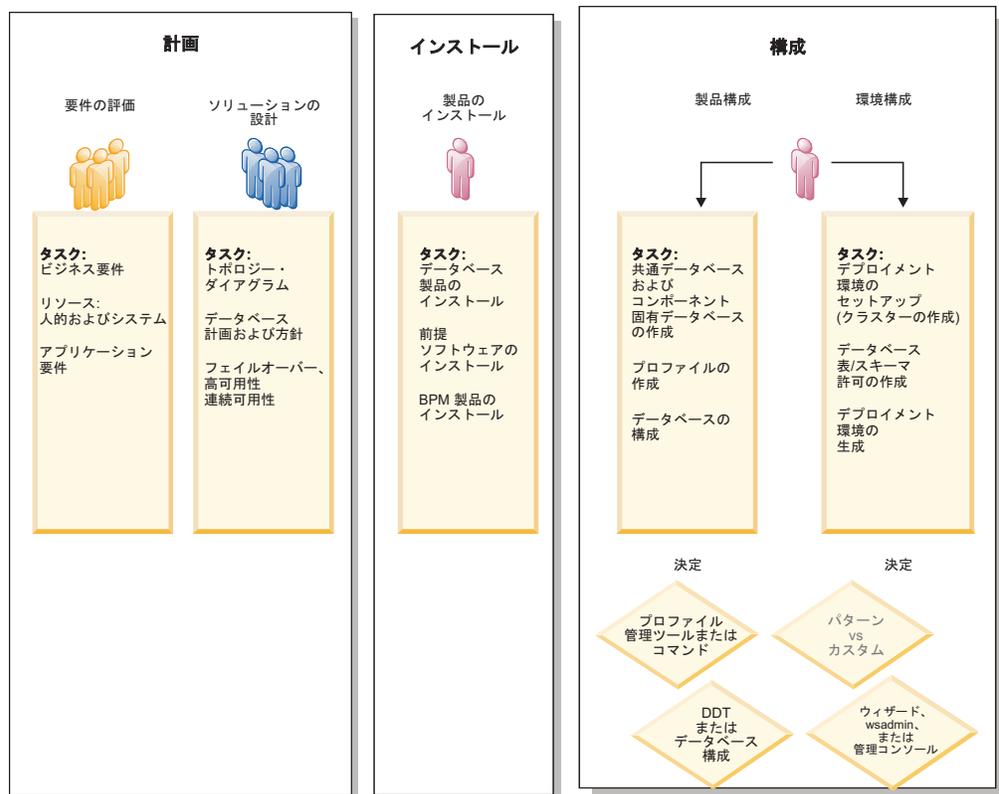
データベース製品のインストールは、製品のインストール・フェーズに含まれません。

データベースを WebSphere Process Server で使用できるようにする作業は、製品構成フェーズに含めることができます。

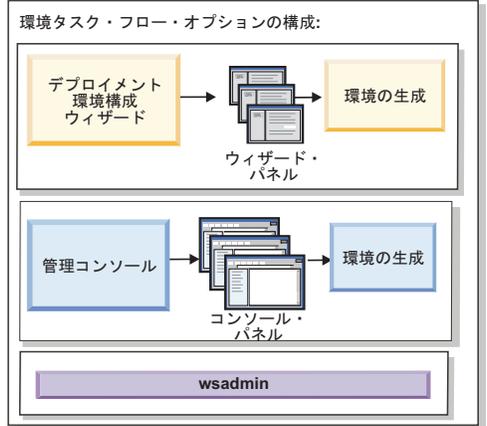
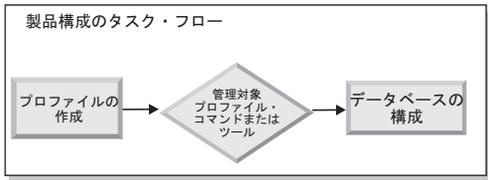
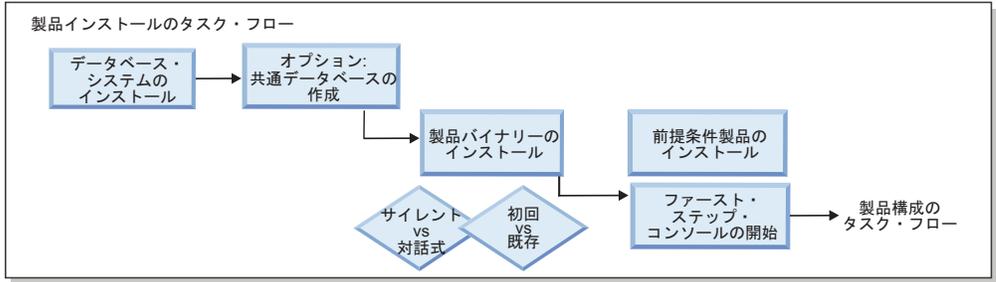
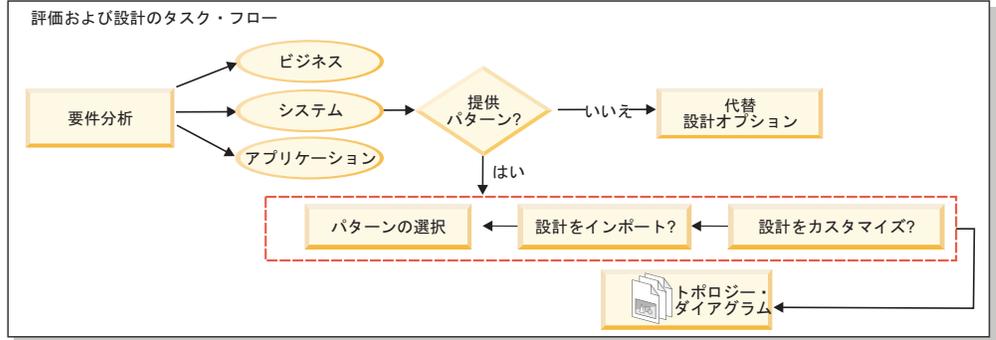
WebSphere Process Server データベースの作成時期については、多少柔軟性があります。データベースの作成は、WebSphere Process Server のインストール前、WebSphere Process Server のインストール後 (ただしプロファイルの作成前)、またはプロファイルの作成後に行うことができます。重要なことは、これらのデータベース (つまり、データベース、テーブル、スキーマなど) が、WebSphere Process Server サーバーによって使用される前に存在している必要があることです。

デプロイメント・マネージャー・プロセスは、共通データベースを使用します。デプロイメント・マネージャーはセル作成プロセスの「早い段階」で作成するため、デプロイメント・マネージャーに必要な共通データベースの作成は、製品構成 タスクと考えることができます。

デプロイメント環境のクラスターの作成は、環境構成 タスクに含まれます。デプロイメント環境内のクラスターには、そのクラスターの機能目的に応じた固有のテーブル、スキーマ、およびユーザー権限が必要です。例えば、デプロイメント環境にメッセージング・インフラストラクチャー・クラスターとアプリケーション・デプロイメント・ターゲット・クラスターの両方が含まれる場合があります。この場合、メッセージング・インフラストラクチャー・クラスターはメッセージング・エンジンのデータベース表にアクセスし、アプリケーション・デプロイメント・ターゲット・クラスターは Business Process Choreographer のデータベース表にアクセスします。



以下の図には、製品および環境の計画、インストール、および構成に関するタスク・フローを示します。



注: 製品データベースの作成は、製品インストールの前、製品インストールの後 (ただしプロファイル作成の前)、またはプロファイルの作成後に行うことができます。

関連概念

製品ファミリーの概要

Service Component Architecture

73 ページの『第 4 章 トポロジーの理解』

トポロジーの選択は、ビジネスの目標と目的、さらに環境にデプロイするアプリケーションの設計および目的によって異なります。

関連情報

ソフトウェアのニーズの判別

やり直しや障害を最小にするには、先に進んでインストールおよび構成の決定を行う前に、現在の環境の把握に時間を掛けてください。現在のビジネス・ニーズと設計、既にインストールされているハードウェアとソフトウェア、および現在の長所と短所の分析は、デプロイメント環境に最適な設計を判別するのに役立ちます。また、この計画は、現在のニーズに必要な財務投資を最小化するのにも役立つ場合があります。



[BPM Solution Implementation Guide](#)



[z/OS: WebSphere Business Process Management V6.2 Production Topologies](#)

第 2 章 ソフトウェアのニーズの判別

やり直しや障害を最小にするには、先に進んでインストールおよび構成の決定を行う前に、現在の環境の把握に時間を掛けてください。現在のビジネス・ニーズと設計、既にインストールされているハードウェアとソフトウェア、および現在の長所と短所の分析は、デプロイメント環境に最適な設計を判別するのに役立ちます。また、この計画は、現在のニーズに必要な財務投資を最小化するのにも役立つ場合があります。

ソフトウェアのニーズは、いくつかの要因に基づいて判別します。

それらの要因は、以下のカテゴリにまとめることができます。

- リソース・アベイラビリティおよび制約事項

このカテゴリには、製品のハードウェア要件およびソフトウェア要件の理解の他に、独自のシステム・リソース制約、およびシステムを管理して保守するためのリソースの可用性の理解も含まれます。

- ビジネス要件

これには、実行時環境にデプロイするアプリケーションの理解の他に、構成済み環境の所期の使用方法の理解も含まれます。

- ニーズを満たすためにインストールする製品 (および製品のバージョン)

これらすべての要因を考慮して、知識に基づいた選択を行うためには、以下の内容を理解する必要があります。

- 環境構成に関して使用される用語。
- インストール、構成、管理、および保守を行う製品 (または BPM 製品グループ) の管理アーキテクチャー。
- (提供パターンを介して) 使用できる構成オプション、および特定のパターンが製品の所期の使用方法に対応できるかどうかを判別する方法。
- サポートされている実装方法。製品をインストールして環境を構成するためのさまざまなタスク・フローの理解も含まれます。

このセクションの情報により、現在および将来のニーズを分析し、それらのニーズに合う環境を開発します。

注: プラットフォーム固有のディスク・スペース所要量、サポートされているオペレーティング・システム、サポートされているデータベースのバージョン、およびオペレーティング・システムを準拠させるためにインストールする必要があるオペレーティング・システムのフィックスおよびパッチについての最新情報は、<http://www.ibm.com/support/docview.wss?uid=swg27006205>にある WebSphere Process Server の詳細なシステム要件を参照し、ご使用のバージョンの WebSphere Process Server へのリンクを選択してください。

要件の評価

現在の要件は、ビジネス・コンポーネントの統合を合理化し、機能を向上させるための計画を立案する基礎になります。ビジネスの将来を見据えることにより、現在の業務を遂行するだけでなく、ビジネスの拡大も視野に入れた意思決定の指針を築くことができます。

始める前に

製品の製造および出荷、サービスの提供がどのように行われているかを把握する必要があります。

WebSphere Process Server には、実稼働環境とテスト環境の両方の要件に合うように設計されたデプロイメント環境パターンが用意されています。

このタスクについて

計画プロセスの一部として、業務の仕組みを分析する必要があります。ここに示す手順では、この分析のためのフレームワークを提供します。

手順

1. プロセスとアプリケーションが既存のサービスやバックエンド・システムとどう連携するかを検討します。
2. アプリケーションによるデータの処理方法と、システム内でのデータの流を検討して、具体的なビジネス・ニーズに対処します。

ソリューションを開発するときや、そのソリューションをサポートする環境を構成するときには、複数の検索、セッション、プロセス、およびその他の境界にわたって、データがどのように持続するかを理解することが重要になります。

環境にデプロイするアプリケーションに関する以下の項目を検討してください。

- アプリケーションの呼び出しパターン

非同期呼び出しがランタイムでどのように処理されるか、また、非同期呼び出しを実装するために、基盤となるメッセージ・システムが SCA ランタイムでどのように利用されるかを理解します。

アプリケーションに必要な機能は、アプリケーションごとに異なります。その決定要因としては、例えば、アプリケーション・モジュールのエクスポート・タイプ、コンポーネント・タイプ、コンポーネント間の対話、インポート・タイプ、データベースや JMS リソースなどの必要なリソース、ビジネス・イベントの必要性、ビジネス・イベントの伝送機構などがあります。

- 実装を計画しているビジネス・プロセスのタイプ (割り込み可能か割り込み不可能か)

割り込み不可能なビジネス・プロセス、つまり *microflow* は、1 回のトランザクションで実行されるかトランザクションなしで実行される、実行時間の短いビジネス・プロセスです。割り込み不可能なビジネス・プロセスは、パフォーマンスのオーバーヘッドがほとんどなく、高速です。このプロセス内のすべてのアクティビティは、単一スレッド内で処理されます。

割り込み可能なビジネス・プロセス、つまり *macroflow* は、一連のアクティビティが含まれている、実行時間の長いビジネス・プロセスです。このプロセス内のアクティビティは、それぞれ固有のトランザクションで実行されず。

割り込み可能なビジネス・プロセスには、人的介入またはリモート・システムの呼び出し、あるいはその両方が必要なアクティビティを組み込むことができます。ビジネス・プロセスに、人的介入 (スタッフ) を必要とする非同期アクティビティや、外部イベントの実行を必要とする非同期アクティビティが組み込まれている場合は、実行時の割り込みが可能なタイプのビジネス・プロセスになります。これらのアクティビティは、実行して完了するために数分、数時間、あるいは数日もの時間がかかる可能性があるためです。

3. 将来について計画します。

- プロジェクトや企業が拡張される場合、企業の IT 部門はその拡張にどう対応しますか?
- 競合他社を買収し、その製品およびサービスを自社に取り込む可能性はありますか? 新しい分野の製品またはサービスを開拓する可能性はありますか?

次のタスク

使用可能なリソースを明確化します。

関連概念

ビジネス・プロセスのタイプ

関連情報



SCA asynchronous invocation patterns in depth

使用可能なリソースの明確化

資産を明確化し、既に使用可能なリソースを最大限に利用するとともに、購買の決定の通知を受け取ります。

始める前に

現在のハードウェアおよびソフトウェアに精通している必要があります。以下のようにして使用可能な資産のリストを準備します。

このタスクについて

現在のエンタープライズ情報システムを評価し、業務上のニーズを満たすためにハードウェアまたはソフトウェアを追加する必要があるかを判別します。

手順

1. 使用する物理コンピューター・システムの数を決め、各物理ハードウェアの詳細を作成します。

以下の項目を調べてメモに記録します。

- 実装済みメモリーの量
- 実装済みマイクロプロセッサの数およびタイプ

- 外部メディア
 - 特定の装置がアップグレード可能かどうか
2. 現在インストールされているソフトウェアおよびデータベース・アプリケーションを列挙します。
 - 機能
 - 企業内での使用範囲
 - セキュリティー要件
 3. 現在の IT 担当者をリストし、WebSphere Process Server のインストールおよび保守の手段が適用できるか、およびデータベース管理に必要な専門家がいるかを確認します。すべての製品およびファイルを正常にインストールするための適切な権限を持つユーザー ID がすべての関係者に割り当てられていることを確認します。

関連情報

 ハードウェアおよびソフトウェア要件

開発およびデプロイメントのバージョン・レベル

ご使用の環境に必要な WebSphere Process Server のバージョン・レベルの決定は、アプリケーションが開発されたときのバージョン・レベルに依存します。一般に、前のバージョンの WebSphere Process Server にデプロイされたアプリケーションは、次に入手可能なバージョンの WebSphere Process Server 上で稼働します。

WebSphere Process Server バージョン 7.0 および WebSphere Integration Developer バージョン 7.0 は、以前のリリースとの間に以下に示す互換性があります。

- WebSphere Integration Developer バージョン 6.0.2、6.1.0、6.1.2、または 6.2.0 から WebSphere Process Server 7.0 へのデプロイメントがサポートされています。
 - WebSphere Integration Developer 6.0.2、6.1.0、6.1.2、または 6.2.0 を使用して作成および生成されたアプリケーションは、WebSphere Process Server 7.0 サーバーにパブリッシュできます。
 - WebSphere Integration Developer 6.0.2、6.1.0、6.1.2、または 6.2.0 で作成、生成、およびそこからエクスポートされたアプリケーションは、WebSphere Process Server 7.0 サーバーにインストールできます。

注: WebSphere Adapter (バージョン 6.1.0、6.1.2、および 6.2.0) の場合は、*Mandatory adapter fix for running 6.1 and 6.2 Adapters on WPS v7.0* というタイトルの暫定修正をインストールする必要があります。この暫定修正をソース環境に適用する必要があるのは、WebSphere Adapter をバージョン 7.0 レベルにアップデートせず、アプリケーションを WebSphere Adapter バージョン 6.1.0、6.1.2、または 6.2.0 で使用し続ける計画の場合です。

注: Websphere Adapter for SAP (バージョン 6.0.2、6.1.0、6.1.2、および 6.2.0) は、WebSphere Process Server バージョン 7.0 ではサポートされません。WebSphere Process Server バージョン 7.0 上で Websphere Adapter for SAP を使用するアプリケーションをデプロイする前に、Websphere Adapter for SAP をバー

ジョン 7.0 にアップデートする必要があります。 WebSphere Adapter for SAP に固有の詳細情報については、『*WebSphere Process Server の事後マイグレーション・タスク*』を参照してください。

- WebSphere Process Server 7.0 成果物を WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 で実行することは、サポートされていません。
 - WebSphere Integration Developer 7.0 で作成されたアプリケーションは、WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 (前のすべてのリリース) サーバーにパブリッシュまたはインストールすることはできません。このようなコンテンツは WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 で正しく稼働せず、コードの世代変更によってこれらのアプリケーションは WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 で正しく稼働しなくなります。
 - WebSphere Integration Developer 6.0.2、6.1.0、6.1.2、または 6.2.0 で作成され、WebSphere Integration Developer 7.0 で生成されたアプリケーションは、WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 サーバーにパブリッシュまたはインストールできません。コードの世代変更によってこれらのアプリケーションは WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 で正しく稼働しなくなります。
 - serviceDeploy を使用して WebSphere Process Server 7.0 サーバーから生成されたアプリケーションは、WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 サーバーにインストールできません。コードの世代変更によってこれらのアプリケーションは WebSphere Process Server 6.0.2、6.1.0、6.1.2、または 6.2.0 で正しく稼働しなくなります。

関連概念

『WebSphere Process Server と他の WebSphere Application Server 製品との間のインターオペラビリティの計画』

ソフトウェア環境を分析するときは、デプロイメント環境内に存在するさまざまなソフトウェア・レベル間で要求を受け渡すことができるかどうかを把握する必要があります。

関連情報

 [WebSphere Process Server へのマイグレーション](#)

WebSphere Process Server と他の WebSphere Application Server 製品との間のインターオペラビリティの計画

ソフトウェア環境を分析するときは、デプロイメント環境内に存在するさまざまなソフトウェア・レベル間で要求を受け渡すことができるかどうかを把握する必要があります。

最適なインターオペラビリティを維持するには、関連するすべての WebSphere Application Server サービスを適用し、該当するすべてのガイドラインに従った後、WebSphere Process Server の未解決のフィックスを適用してあることを確認してください。

関連概念

10 ページの『開発およびデプロイメントのバージョン・レベル』
ご使用の環境に必要な WebSphere Process Server のバージョン・レベルの決定は、アプリケーションが開発されたときのバージョン・レベルに依存します。一般に、前のバージョンの WebSphere Process Server にデプロイされたアプリケーションは、次に入手可能なバージョンの WebSphere Process Server 上で稼働します。

関連情報



Application Server の複数のバージョンの相互運用

インストールする製品の決定

デプロイメント環境の設計には、必要になる可能性があるソフトウェア製品の数とタイプの判断が含まれます。製品の要件は、お客様のニーズに基づき、その環境に関連するコンピューター LPAR システムによって異なる可能性があります。デプロイメント環境のすべてのサーバーにそれぞれ WebSphere Process Server ライセンスが必要なわけではありません。

始める前に

設計に関する以下の詳細情報が必要です。

- デプロイメント環境に組み込むクラスターおよびサーバー
- さまざまなサーバーを置く物理的ハードウェア
- 各クラスターがデプロイメント環境に提供する機能例えば、Web アプリケーション・コンポーネントのサポート、Java™ Platform Enterprise Edition コンポーネントのサポート、メディアエーション・モジュールのサポート、メッセージングのサポート、またはプロセス・サーバーのサポート。

このタスクについて

デプロイメント環境を設計した後、ソフトウェアを購入する前にこのタスクを使用して、ご使用のデプロイメント環境に必要な適切なソフトウェアを判断してください。

手順

1. デプロイメント環境のさまざまなコンポーネントをホストする専用の LPAR の数をカウントします。

重要: 同じ LPAR システム上で複数のサーバーを実行している場合、その LPAR システム上で動作するすべてのサーバーに必要な機能をもっと多数提供するソフトウェアをインストールする必要があります。

このカウントには以下のものを含まれます。

- 必要なデプロイメント・マネージャーの数。管理対象サーバー上で動作するソフトウェアにより、デプロイメント・マネージャーにインストールするソフトウェアが決まります。
- WebSphere Process Server インスタンス
- WebSphere ESB インスタンス: メディアエーションのみをホストするための専用のインスタンス

- まだカウントしていないメッセージング・エンジン: 固有の WebSphere Application Server インスタンスの数を表します
2. ソフトウェアのコストがプロジェクトの予算を超えるかどうかを判定します。
 3. オプション: 財務上の要件を満たすように設計を調整します。コストを下げるには、容量が最大の LPAR で複数のサーバーをホストする必要があります。
 - 別個の LPAR に別個のサーバー・インスタンスを作成するより、容量が大きい LPAR に同じタイプの複数のサーバー・インスタンスを作成すると、インスタンスの数は同じでも必要なソフトウェアの数が減ります。
 - メッセージング・エンジンをホストする専用の LPAR が必要かどうかを判断します。不要な場合は、それらを除去します。
 - 設計からアプリケーションを除去して、必要なアプリケーション・サーバー・インスタンスの数を減らします。

タスクの結果

以上で、設計の実装に必要なソフトウェアが判明しました。

次のタスク

必要なソフトウェアを発注します。

関連概念

117 ページの『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。

118 ページの『デプロイメント環境のクラスター』

クラスターを使用すると、アプリケーションの能力と可用性が単一サーバーの場合よりも高まります。

112 ページの『サーバー』

サーバーによって、WebSphere Process Server のコア機能が提供されます。プロセス・サーバーでは、アプリケーション・サーバーの機能が拡大または拡張されて、Service Component Architecture (SCA) モジュールが処理されます。他のサーバー (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) は、プロセス・サーバーの管理に使用されます。

関連情報

 [メッセージング・エンジン](#)

Business Process Choreographer の構成計画

Business Process Choreographer のセットアップと構成パラメーターを計画します。

手順

1. 14 ページの『トポロジー、セットアップ、および構成パスの計画』を実行します。

2. 選択した構成パスに応じて、以下のいずれかを実行します。
 - 『基本サンプル』の場合は、20ページの『Business Process Choreographer の基本サンプル構成の作成の計画』を実行してください。
 - 『組織付きサンプル』の場合は、21ページの『サンプル組織を含む Business Process Choreographer のサンプル構成の作成の計画』を実行してください。
 - 『非実動デプロイメント環境』の場合は、22ページの『非実動デプロイメント環境構成の計画』を実行してください。
 - 『実動デプロイメント環境』の場合は、24ページの『管理コンソールのデプロイメント環境ウィザードを使用するための計画』を実行してください。
 - 『柔軟なカスタム構成』の場合は、29ページの『カスタム Business Process Choreographer 構成の計画』を実行してください。

タスクの結果

これで、Business Process Choreographer を構成するために必要なすべての事項の計画が完了しました。

トポロジー、セットアップ、および構成パスの計画

選択するトポロジーとセットアップによって、使用できる Business Process Choreographer 構成パスが決まります。

このタスクについて

構成パスごとに、複雑さ、柔軟性、および各種トポロジーとデータベースのサポートが異なります。

手順

1. 次の 5 つの異なる構成パスから選択する必要がある点に注意してください。
 - 『基本サンプル』
 - 『組織付きサンプル』
 - 『非実動デプロイメント環境』
 - 『実動デプロイメント環境』
 - 『柔軟なカスタム構成』

ほとんどの構成パスでは、構成ツールを選択できます。

2. Business Process Choreographer の構成にはさまざまな構成ツールを使用できる点に注意してください。

インストーラー または プロファイル管理ツール

非実動システムを最も簡単に作成できます。また、計画しておく必要があることが最も少ないツールです。

- 『基本サンプル』構成には、次の Business Process Choreographer コンポーネントが含まれています。
 - Business Process Choreographer
 - レポート作成機能を持つ Business Process Choreographer Explorer
 - レポート作成機能のための Business Process Choreographer Event Collector

- 『組織付きサンプル』構成には、サンプル組織の 15 人のユーザーを使用して事前に構成されている担当者ディレクトリーも含まれており、代替とグループ作業項目が有効になっています。
- 『非実動デプロイメント環境』構成では、クラスターで Business Process Choreographer を容易に構成できますが、Business Process Choreographer に専用のデータベースを設定できません。代わりに、共通 WPRCSDB データベースが使用されます。

管理コンソールのデプロイメント環境ウィザード

このウィザードを使用して、デプロイメント環境パターンに基づいて『実動デプロイメント環境』の Business Process Choreographer 構成を作成できます。

管理コンソールの「Business Process Choreographer の構成」ページ

管理コンソールのこのページでは、サーバーまたはクラスター上の『柔軟なカスタム構成』の Business Process Choreographer 実動システムを構成できます。多数の構成パラメーターを設定できます。これには、詳細な計画が必要です。このページでは、Business Process Choreographer Explorer は構成されません。これは、管理コンソール内の独自の構成ページを使用するか、スクリプトを実行することによって構成できます。この構成パスは、実動システムの構築に最も適しています。

bpeconfig.jacl 構成スクリプト

このスクリプトを使用して、特定のサーバーまたはクラスター上で、『柔軟なカスタム構成』の Business Process Choreographer 実動システムと必要なすべてのリソースを構成できます。このスクリプトは対話式に実行できます。また、必要なパラメーターをすべて指定する場合は、バッチ・モードで実行して繰り返し可能な自動化を実現できます。ローカル・データベースおよび必要なメッセージング・リソースを作成することができます。また、オプションで Business Process Choreographer Explorer を構成することもできます。ここには、Business Process Choreographer Explorer レポート作成機能が組み込まれます。データベース・システムによっては、リモート・データベースも作成できます。この構成パスは、実動システムの構築に最も適しています。

clientconfig.jacl 構成スクリプト

このスクリプトを使用するだけで、Business Process Choreographer Explorer を構成でき、レポート作成機能を付けるかどうかをオプションで選択できます。

3. 一部の構成パスには、実動システムへの適合を限定する制約事項があります。以下に例を示します。
 - 1 つのサンプル構成を試験的に使用した場合、実動システムに適した構成を作成する前に、このサンプル構成を除去する必要があります。
 - Derby Embedded データベースまたは共通 WPRCSDB データベースを使用する構成を作成する場合、この構成はハイパフォーマンス・システムには適していません。別のハイパフォーマンス・データベースを使用する新規構成を作成する前に、この構成を除去する必要があります。
 - メッセージ・ストアがファイル・ストアまたは Derby 組み込みのデータ・ストアを使用する場合は、Network Deployment 環境にプロファイルを統合できません。プロファイルを統合するには、Business Process Choreographer 構成を

完全に除去してから、メッセージ・ストアとしてリモート操作でアクセス可能なデータベースを使用する新規構成を作成する必要があります。

- バージョン 6.1.2 までの Business Process Choreographer Observer を使用していた場合は、この機能が今では Business Process Choreographer Explorer に統合されていることに注意してください。これは今では Business Process Choreographer Explorer レポート作成機能と呼ばれ、Business Process Choreographer Explorer クライアントの「レポート」タブを使用してアクセスできます。レポート作成機能が使用する URL は、Business Process Choreographer Explorer の URL と同じです。

管理コンソールで Business Process Choreographer Explorer を構成する場合や、bpeconfig.jacl 構成スクリプトまたは clientconfig.jacl 構成スクリプトを使用する場合は、Business Process Choreographer Explorer レポート作成機能を構成することもできます。

既存の Business Process Choreographer 構成をマイグレーションしても、Business Process Choreographer Observer の構成はマイグレーションされません。Business Process Choreographer Explorer レポート作成機能を使用するには、Business Process Choreographer Explorer レポート作成機能のマイグレーション後の使用可能化の説明に従って、この機能を使用可能にする必要があります。

- 使用する構成パスを決定する上での主な基準を確認します。次の表で、選択できる構成パスと制約を確認してください。

表 2. 構成パスの選択基準

選択項目		制約事項		適切な構成パス	
実動システムを計画しているか	デプロイメント・ターゲットの形態	Business Process Choreographer 構成のタイプ	別の BPEDB データベースを使用できるか	メッセージング・エンジンでサポートされているメッセージ・ストア	構成パス名、ツール、およびオプション
いいえ	スタンドアロン・サーバー	基本サンプル (サンプル組織なし)	はい。ただし Derby Embedded のみ。	Derby Embedded のみ	『基本サンプル』 次のいずれかを使用: <ul style="list-style-type: none"> インストーラー プロファイル管理ツール オプションを選択する。 <ul style="list-style-type: none"> スタンドアロン・サーバーのプロファイル 標準的 管理セキュリティを有効にする
		15 名の担当者で構成される組織を含むサンプル (担当者の代替が使用可能)。 このサンプルは、WebSphere テスト環境を組み込むときに WebSphere Integration Developer で使用可能なサンプルと同じです。		Derby Embedded、ファイル・ストア、または WPRCSDB	『組織付きサンプル』 次のコンポーネントを使用: <ul style="list-style-type: none"> プロファイル管理ツール オプションを選択する。 <ul style="list-style-type: none"> スタンドアロン・サーバーのプロファイル 高度 開発テンプレートからサーバーを作成する (Create server from development template) 管理セキュリティを有効にする サンプル Business Process Choreographer の構成
	クラスター	選択できるデプロイメント環境パターン: <ul style="list-style-type: none"> リモート・メッセージングおよびリモート・サポート リモート・メッセージング 単一クラスター 	いいえ、WPRCSDB を共有しません。WPRCSDB は、Derby Embedded および Microsoft® SQL Server 以外の任意のデータベースです。	WPRCSDB を共有します。WPRCSDB は、ファイル・ストアおよび Derby Embedded 以外のサポートされている任意のデータベースです。	『非実動デプロイメント環境』 次のいずれかを使用: <ul style="list-style-type: none"> インストーラー プロファイル管理ツール 「デプロイメント環境」を選択する。

表 2. 構成パスの選択基準 (続き)

選択項目		制約事項		適切な構成パス
実動システムを計画しているか	デプロイメント・ターゲットの形態	Business Process Choreographer 構成のタイプ	別の BPEDB データベースを使用できるか	メッセンジャー・エンジンでサポートされているメッセージ・ストア 構成パス名、ツール、およびオプション
はい	クラスター	選択できるデプロイメント環境パターン: <ul style="list-style-type: none"> リモート・メッセンジャーおよびリモート・サポート リモート・メッセンジャー 単一クラスター カスタム 	はい。 Derby Embedded を除くサポートされているすべてのデータベース	『実動デプロイメント環境』 次のコンポーネントを使用: <ul style="list-style-type: none"> 管理コンソール 「デプロイメント環境」を選択する。
		柔軟なカスタム構成	はい。サポートされている任意のデータベース	『柔軟なカスタム構成』 次のいずれかを使用: <ul style="list-style-type: none"> bpeconfig.jacl スクリプト 管理コンソールの「Business Process Choreographer の構成」ページ
	スタンドアロン・サーバー		サポートされている任意のデータベース、またはファイル・ストア	

注: 実動システムに適していない構成を作成するときに、実動システムの作成向けに推奨されている構成パスを使用することもできます。

次のオプションを検討します。

- a. 実動システムを構成するかどうかを決定します。 一般に実動システムでは、ハイパフォーマンス、スケーラビリティ、およびセキュリティが必要で、Business Process Choreographer の場合、実動システムに専用の BPEDB データベース (Derby 以外) が必要です。
- b. Business Process Choreographer のデプロイメント・ターゲットとして、スタンドアロン・サーバーとクラスターのいずれを使用するかを決定します。
- c. 実動システムを構築しない場合は、スタンドアロン・サーバーのサンプル構成が要件に対応するかどうかを確認します。 対応する場合は、担当者割り当てと担当者の代替を使用可能にするためにサンプル担当者ディレクトリー (サンプル組織が取り込まれています) をサンプルに組み込むかどうかを決定します。

注: サンプル担当者ディレクトリーでは、フェデレーテッド・リポジトリー用に構成されたデフォルトのファイル・レジストリーが使用され、すべてのサンプル担当者のパスワードが「wid」に設定されています。WebSphere 管理ユーザー ID も、プロファイル作成時に指定されたパスワードを使用してこのディレクトリーに追加されます。サンプル構成が作成された後、管理コンソールを使用して「ユーザーおよびグループ」をクリックしてから「ユーザーの管理」または「グループの管理」をクリックすることで、使用可能なユーザーおよびグループを表示させることができます。

- d. クラスターで Business Process Choreographer を構成する場合は、パフォーマンスの要件に基づいて、メッセージング・エンジンとサポート・アプリケーション (Business Process Choreographer Explorer、 Common Event Infrastructure など) に専用のクラスターを設定するか、それともクラスターを共有するかを決定します。標準のデプロイメント環境パターンは以下のとおりです。

リモート・メッセージングおよびリモート・サポート

3 つのクラスターが使用されます。アプリケーション、メッセージング・エンジン、およびサポート・アプリケーション用にそれぞれクラスターが 1 つずつ使用されます。

リモート・メッセージング

アプリケーションとサポート関数に 1 つのクラスターが使用されます。2 番目のクラスターはメッセージング・エンジン用に使用されません。

単一クラスター

アプリケーション、メッセージング・エンジン、およびサポート・アプリケーションに対して 1 つのクラスターが使用されます。

カスタム

柔軟性の高いセットアップです。

- e. Business Process Choreographer に専用 BPEDB データベースを使用するかどうかを決定します。
- f. Business Process Choreographer は SCA によって使用されるものと同じ種類のメッセージ・ストアを使用します。
- SCA で FILESTORE が使用される場合、Business Process Choreographer でも FILESTORE が使用されます。
 - SCA で Derby Embedded データベースが使用される場合、Business Process Choreographer ではそれ独自の Derby Embedded データベースが使用されません。
 - SCA でそれら以外のデータベースが使用される場合、Business Process Choreographer では同じデータベース内でそれ独自のスキーマを使用しません。
6. Business Process Choreographer Explorer レポート作成機能 (Business Process Choreographer Explorer に統合) を使用する場合は、Business Process Choreographer 構成を作成する際に同時にこれを構成することも、後で作成することもできます。また、Business Process Choreographer Explorer レポート作成機能が BPEDB データベースを使用するか、あるいは専用の OBSRVDB データベースを使用するかどうかを決定します。Business Process Choreographer

Explorer レポート作成機能のコンポーネントのトポロジーについても計画します。詳細な計画を実行するには、56 ページの『Business Process Choreographer Explorer レポート作成機能の計画』を実行します。

7. WebSphere Portal Server または任意のカスタムの WebSphere Process Server クライアントから Business Process Choreographer にアクセスする場合は、60 ページの『リモート・クライアント・アプリケーションの計画』を実行します。
8. アプリケーション・セキュリティを有効にしている、リモート EJB メソッドを呼び出す長期実行のプロセスがある場合、Common Secure Interoperability Version 2 (CSIv2) のインバウンド認証を構成するときに CSIv2 の ID アサーションを有効にする必要があります。
9. ヒューマン・タスクを使用する場合は、WebSphere 管理セキュリティとアプリケーション・セキュリティの両方を有効にする必要があります。

タスクの結果

これでトポロジーの計画が完了し、使用する構成パスと構成ツールが決定しました。

Business Process Choreographer の基本サンプル構成の作成の計画

この基本サンプルはスタンドアロン・サーバー用であり、サンプル組織は含まれていません。

始める前に

14 ページの『トポロジー、セットアップ、および構成パスの計画』を実行し、『基本サンプル』構成パスを選択している。

手順

1. インストーラー またはプロファイル管理ツールのどちらかを使用してサンプルを作成するかを決定します。
2. プロファイル管理ツールを使用することを決定した場合、Business Process Choreographer メッセージング・エンジンに、ファイル・ストア、組み込み Derby データベース、または共通の WPRCSDB データベースのいずれかを使用するかを決定します。
3. Human Task Manager でエスカレーション E メールを送信できるようにするには、以下のことを計画します。
 - ローカルの Simple Mail Transfer Protocol (SMTP) メール・サーバーが利用できない場合、後でメール・セッションを変更して適切なメール・サーバーを指示することを計画します。
 - Eメールの送信者アドレスを変更することを計画します。変更しない場合、ダミーの送信者アドレスが使用されます。
4. このサンプル構成では、WebSphere 管理者のユーザー ID およびパスワードを、さまざまな Business Process Choreographer ユーザー ID として使用することに注意してください。

タスクの結果

これで、Business Process Choreographer の基本サンプル構成の作成計画が完了しました。

サンプル組織を含む Business Process Choreographer のサンプル構成の作成の計画

このサンプルには、15 人からなるサンプル組織が含まれていて、スタンドアロン・サーバーでの担当者の割り当ておよび代替を試験する場合に適しています。このサンプルは、WebSphere テスト環境を組み込むときに WebSphere Integration Developer で使用可能なサンプルと同じです。

始める前に

14 ページの『トポロジー、セットアップ、および構成パスの計画』を実行し、『組織付きサンプル』構成パスを選択している。

このタスクについて

この Business Process Choreographer のサンプル構成には、最小限の計画が必要です。

手順

1. Business Process Choreographer メッセージング・エンジンに、ファイル・ストア、組み込み Derby データベース、または共通の WPRCSDB データベースのいずれを使用するかを決定します。
2. このサンプルは、プロファイル管理ツールを使用することによってのみ作成できることに注意してください。このサンプルを取得するには、次のオプションを選択する必要があります。
 - スタンドアロン・サーバーのプロファイル
 - 高度
 - 開発テンプレートからサーバーを作成する (Create server from development template)
 - 管理セキュリティーを有効にする
 - サンプル Business Process Choreographer の構成

例えば、管理セキュリティーを使用可能にしない場合、Business Process Choreographer のサンプル構成は作成されません。

注: サンプル担当者ディレクトリーでは、フェデレーテッド・リポジトリー用に構成されたデフォルトのファイル・レジストリーが使用され、すべてのサンプル担当者のパスワードが「wid」に設定されています。WebSphere 管理ユーザー ID も、プロファイル作成時に指定されたパスワードを使用してこのディレクトリーに追加されます。サンプル構成が作成された後、管理コンソールを使用して「ユーザーおよびグループ」をクリックしてから「ユーザーの管理」または「グループの管理」をクリックすることで、使用可能なユーザーおよびグループを表示させることができます。

3. Human Task Manager でエスカレーション E メールを送信できるようにするには、以下のことを計画します。

- ローカルの Simple Mail Transfer Protocol (SMTP) メール・サーバーが利用できない場合、後でメール・セッションを変更して適切なメール・サーバーを指示することを計画します。
 - E メールを送信者アドレスを変更することを計画します。変更しない場合、ダミーの送信者アドレスが使用されます。
4. このサンプル構成では、WebSphere 管理者のユーザー ID およびパスワードを、さまざまな Business Process Choreographer ユーザー ID として使用することに注意してください。

タスクの結果

これで、サンプル組織を含む Business Process Choreographer のサンプル構成の作成の計画が完了しました。

非実動デプロイメント環境構成の計画

インストーラーまたは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パターンに基づく Business Process Choreographer 構成を作成することを計画します。

始める前に

14 ページの『トポロジー、セットアップ、および構成パスの計画』を実行し、『非実動デプロイメント環境』構成パスを選択している。

このタスクについて

デプロイメント環境ウィザードを使用するとき、デプロイメント環境パターンを選択する必要があります。その後、WBI_BPC コンポーネントのデフォルト・データベース・パラメーターおよび認証別名を変更し、Business Process Choreographer の他のパラメーターを入力することができます。

手順

1. 使用するデプロイメント環境パターンを以下から決定します。
 - リモート・メッセージングおよびリモート・サポート
 - リモート・メッセージング
 - 単一クラスター
2. セキュリティー・ステップで入力する Business Process Choreographer JMS 認証別名のユーザー名を計画します。
3. **Business Process Choreographer Explorer** のコンテキスト・ルートを計画します。これは、ブラウザーが Business Process Choreographer Explorer に到達するために使用する必要がある URL の一部を定義します。
4. Business Process Choreographer ステップ用のセキュリティー・パラメーターを計画します。次のユーザー ID およびグループは、Business Flow Manager および Human Task Manager に使用されます。

管理者ユーザーおよびグループ

ビジネス管理者ロールがマップされるユーザー ID のリストまたはグループのリスト、あるいはその両方を計画します。

モニター・ユーザーおよびグループ

ビジネス・モニター・ロールがマップされるユーザー ID のリストまたはグループのリスト、あるいはその両方を計画します。

JMS API 認証ユーザーおよびパスワード

Business Flow Manager のメッセージ・ドリブン Bean の Run-As ユーザー ID。

エスカレーション・ユーザーの認証ユーザーおよびパスワード

Human Task Manager のメッセージ・ドリブン Bean の Run-As ユーザー ID。

クリーンアップ・ユーザー認証のユーザーおよびパスワード

Business Flow Manager および Human Task Manager のクリーンアップ・サービスの Run-As ユーザー ID。このユーザーには、ビジネス管理者ロールが必要です。

5. Human Task Manager のエスカレーションのための E メール・セッションを構成する場合、Business Process Choreographer ステップ用に次のパラメーターを計画します。

メール・トランスポートのホスト

Simple Mail Transfer Protocol (SMTP) E メール・サービスが配置されている場所のホスト名または IP アドレス。

メール・トランスポートのユーザーおよびメール・トランスポートのパスワード

メール・サーバーで認証が不要な場合、これらのフィールドを空にしておくことができます。

Business Process Choreographer Explorer の URL

この URL は、生成された E メール内でリンクを指定するために使用されます。これにより、E メール通知を受信したビジネス管理者は、リンクをクリックして、関連したビジネス・プロセスまたはヒューマン・タスクを Web ブラウザーで表示することができます。

6. Business Process Choreographer Explorer、Business Space、または Representational State Transfer (REST) API あるいは JAX Web サービス API を使用するクライアントを使用する場合は、REST API および JAX Web Services API のコンテキスト・ルートを決定してください。
 - Business Flow Manager のデフォルトは /rest/bpm/bfm および /BFMJAXWSAPI です。
 - Human Task Manager のデフォルトは /rest/bpm/htm および /HTMJAXWSAPI です。
 - サーバー、単一クラスター、または異なる Web サーバーにマップされた複数のクラスターに構成されている場合は、デフォルト値を使用できます。
 - Network Deployment 環境内で、同じ Web サーバーにマップされた複数のデプロイメント・ターゲットに構成されている場合は、デフォルト値を使用しないでください。各 Business Process Choreographer 構成のコンテキスト・ルートは、ホスト名とポートの組み合わせごとに固有のものでなければなりません。これらの値は Business Process Choreographer の構成後に、管理コンソールを使用して手動で設定する必要があります。

7. 担当者の割り当てを使用する場合は、52 ページの『担当者ディレクトリー・プロバイダーの計画』を実行します。

タスクの結果

これで、非実動デプロイメント環境構成の作成の計画が完了しました。

管理コンソールのデプロイメント環境ウィザードを使用するための計画

実動システムの場合、別個のデータベースを含む Business Process Choreographer のすべての構成パラメーターを計画します。非実動システムの場合、共有データベースを使用することができます。

始める前に

14 ページの『トポロジー、セットアップ、および構成パスの計画』を実行し、『実動デプロイメント環境』構成パスを選択している。

このタスクについて

デプロイメント環境ウィザードを使用するとき、デプロイメント環境パターンを選択する必要があります。その後、WBI_BPC コンポーネントのデフォルト・データベース・パラメーターおよび認証別名を変更し、Business Process Choreographer の他のパラメーターを入力することができます。

手順

1. 構成全体を自力で作成するために必要な情報または権限がない場合、システムの他の部分の担当者と相談して計画を立てます。以下に例を示します。
 - 組織の LDAP サーバーに関する情報を要求することが必要な場合もあります。LDAP サーバーが認証を使用する場合は、ユーザー ID および権限を要求する必要があります。
 - データベースを作成する権限がない場合、データベースの計画にデータベース管理者 (DBA) が参加する必要があります。DBA は、カスタマイズして実行するためのデータベース・スクリプトのコピーが必要です。
2. セキュリティー、ユーザー ID、および許可の計画を実行します。
3. 使用するデプロイメント環境パターンを以下から決定します。
 - リモート・メッセージングおよびリモート・サポート
 - リモート・メッセージング
 - 単一クラスター
 - カスタム
4. カスタムのデプロイメント環境パターンを選択した場合、以下のようになります。
 - a. Business Process Choreographer Explorer をインストールするかどうかを決定します。インストールする場合、以下のことを計画します。
 - デプロイする場所。

- Business Process Choreographer Explorer レポート作成機能を使用する場合は、Business Process Choreographer Event Collector をデプロイする場所も計画します。
- b. SCA バインディング用のコンテキスト・ルートを計画します。
 - c. 状態監視者および監査ログを有効にするか無効にするかを計画します。
5. 次のデータベースについて、専用データベースを計画する場合があります。
- Business Process Choreographer の BPEDB データベース。これは、ウィザードでコンポーネント WBI_BPC についてのテーブル行で変更できます。
 - Business Process Choreographer メッセージング・エンジンの BPEME データベース。これは、ウィザードでコンポーネント WBI_BPC_ME についてのテーブル行で変更できます。
 - Business Process Choreographer Explorer レポート作成機能の OBSRVADB データベース。これは、ウィザードでコンポーネント WBI_BPCEventCollector についてのテーブル行で変更できます。

ウィザードのデータベース・ページで入力するために、各データベース用の次のパラメーターを計画します。

データベース名

データベースの名前。例えば、デフォルト値に代えて BPEDB、BPEME、または OBSRVADB など。デフォルト値 WPRCSDB を指定すると、共通データベースが共有されます。デフォルト値は、低いパフォーマンスのセットアップの場合にのみ適しています。

スキーマ

各データベースに使用されるスキーマ修飾子。

テーブルの作成

これを選択すると、データベースに最初にアクセスしたときにテーブルが自動的に作成されます。このオプションが機能するには、データベースが既に存在し、データ・ソースを作成するために指定したユーザー名が、データベース内でテーブルおよび索引を作成する権限を持つ必要があります。選択しない場合、テーブルは自動的に作成されず、スクリプトを実行してテーブルを手動で作成する必要があります。実動システムの場合、このオプションをクリアし、提供された SQL スクリプトを使用してデータベースをセットアップすることを計画します。

ユーザー名およびパスワード

データベースへの接続とデータの変更を行う権限を持つユーザー ID。データベース内にテーブルと索引を作成する権限のあるユーザー ID の場合、テーブルを自動的に作成するオプションを使用できます。また、サービス・パックまたはフィックスパックを適用した後、必要に応じてデータベース・スキーマが自動的に更新されます。

サーバー

データベース・サーバーのアドレス。ホスト名または IP アドレスのいずれかを指定します。

プロバイダー (Provider)

JDBC プロバイダー。

データベース固有の設定も計画します。これらの設定は、JDBC プロバイダーの「編集」ボタンを使用して設定できます。

表 3. データベース固有の設定

データベース / JDBC ドライバー・タイプ	データベース固有の設定
DB2 [®] UDB – Universal ドライバー	<ul style="list-style-type: none"> • ユーザー名 • パスワード • データベース名 • スキーマ名 • サーバー名 (Server name) • サーバー・ポート番号 • ドライバー・タイプ • 説明 • テーブル作成 (Create tables)
DB2 for i5/OS [®] – ツールボックス・ドライバー	<ul style="list-style-type: none"> • ユーザー名 • パスワード • データベース名 • 集合名 • サーバー名 (Server name) • 説明 • テーブル作成 (Create tables)
DB2 for z/OS [®] V8 および V9	<ul style="list-style-type: none"> • 実装タイプ – 接続プール・データ・ソースまたは XA データ・ソース • ユーザー名 • パスワード • データベース名 • スキーマ名 • サーバー名 (Server name) • サーバー・ポート番号 • ストレージ・グループ • 説明
Derby Network Server または Derby Network Server 40	<ul style="list-style-type: none"> • ユーザー名 • パスワード • 説明 • テーブル作成 (Create tables) • サーバー名 (Server name) • サーバー・ポート番号
Derby Embedded または Derby Embedded 40	<ul style="list-style-type: none"> • 説明 • テーブル作成 (Create tables)

表 3. データベース固有の設定 (続き)

データベース / JDBC ドライバー・タイプ	データベース固有の設定
Microsoft SQL Server – Datadirect ドライバー、および Microsoft ドライバー	<ul style="list-style-type: none"> • ユーザー名 • パスワード • データベース名 • サーバー名 (Server name) • サーバー・ポート番号 • 説明 • テーブル作成 (Create tables)
Informix® Dynamic Server – Universal ドライバーおよび DataServer ドライバー	<ul style="list-style-type: none"> • ユーザー名 • パスワード • サーバー名 (Server name) • サーバー・ポート番号 • 説明 • テーブル作成 (Create tables)
Oracle – oci ドライバー	<ul style="list-style-type: none"> • ユーザー名 • パスワード • データベース名 • スキーマ名 • ドライバー・タイプ – oci • 説明 • テーブル作成 (Create tables)
Oracle – thin ドライバー	<ul style="list-style-type: none"> • ユーザー名 • パスワード • データベース名 • スキーマ名 • サーバー名 (Server name) • サーバー・ポート番号 • ドライバー・タイプ – thin • 説明 • テーブル作成 (Create tables)

データベースの計画について詳しくは、38 ページの『Business Process Choreographer のデータベースの計画』を参照してください。

6. セキュリティー・ステップで入力する Business Process Choreographer JMS 認証別名のユーザー名を計画します。
7. **Business Process Choreographer Explorer** のコンテキスト・ルートを計画します。これは、ブラウザーが Business Process Choreographer Explorer に到達するために使用する必要がある URL の一部を定義します。
8. Business Process Choreographer ステップ用のセキュリティー・パラメーターを計画します。次のユーザー ID およびグループは、Business Flow Manager および Human Task Manager に使用されます。

管理者ユーザーおよびグループ

ビジネス管理者ロールがマップされるユーザー ID のリストまたはグループのリスト、あるいはその両方を計画します。

モニター・ユーザーおよびグループ

ビジネス・モニター・ロールがマップされるユーザー ID のリストまたはグループのリスト、あるいはその両方を計画します。

JMS API 認証ユーザーおよびパスワード

Business Flow Manager のメッセージ・ドリブン Bean の Run-As ユーザー ID。

エスカレーション・ユーザーの認証ユーザーおよびパスワード

Human Task Manager のメッセージ・ドリブン Bean の Run-As ユーザー ID。

クリーンアップ・ユーザー認証のユーザーおよびパスワード

Business Flow Manager および Human Task Manager のクリーンアップ・サービスの Run-As ユーザー ID。このユーザーには、ビジネス管理者ロールが必要です。

- Human Task Manager のエスカレーションのための E メール・セッションを構成する場合、Business Process Choreographer ステップ用に次のパラメーターを計画します。

メール・トランスポートのホスト

Simple Mail Transfer Protocol (SMTP) E メール・サービスが配置されている場所のホスト名または IP アドレス。

メール・トランスポートのユーザーおよびメール・トランスポートのパスワード
メール・サーバーで認証が不要な場合、これらのフィールドを空にしておくことができます。

Business Process Choreographer Explorer の URL

この URL は、生成された E メール内でリンクを指定するために使用されます。これにより、E メール通知を受信したビジネス管理者は、リンクをクリックして、関連したビジネス・プロセスまたはヒューマン・タスクを Web ブラウザーで表示することができます。

- Business Process Choreographer Explorer、Business Space、または Representational State Transfer (REST) API あるいは JAX Web サービス API を使用するクライアントを使用する場合は、REST API および JAX Web Services API のコンテキスト・ルートを決定してください。
 - Business Flow Manager のデフォルトは /rest/bpm/bfm および /BFMJAXWSAPI です。
 - Human Task Manager のデフォルトは /rest/bpm/htm および /HTMJAXWSAPI です。
 - サーバー、単一クラスター、または異なる Web サーバーにマップされた複数のクラスターに構成されている場合は、デフォルト値を使用できます。
 - Network Deployment 環境内で、同じ Web サーバーにマップされた複数のデプロイメント・ターゲットに構成されている場合は、デフォルト値を使用しないでください。各 Business Process Choreographer 構成のコンテキスト・ルートは、ホスト名とポートの組み合わせごとに固有のものでなければなりません。

せん。これらの値は Business Process Choreographer の構成後に、管理コンソールを使用して手動で設定する必要があります。

11. 担当者の割り当てを使用する場合は、52 ページの『担当者ディレクトリー・プロバイダーの計画』を実行します。

タスクの結果

これで管理コンソールのデプロイメント環境ウィザードを使用するための計画が完了しました。

カスタム Business Process Choreographer 構成の計画

管理コンソールの「Business Process Choreographer の構成」ページまたは bpeconfig.jacl 構成スクリプトのいずれかを使用してカスタム構成を作成するための構成パラメーターおよびオプションを計画します。

始める前に

14 ページの『トポロジー、セットアップ、および構成パスの計画』を実行し、『柔軟なカスタム構成』構成パスを選択している。

手順

1. Business Process Choreographer を構成するために以下のどちらを使用するかを確認します。
 - 管理コンソールの「Business Process Choreographer の構成」ページ
 - bpeconfig.jacl 構成スクリプト
2. 構成全体を自力で作成するために必要な情報または権限がない場合、システムの他の部分の担当者と相談して計画を立てます。以下に例を示します。
 - 組織の LDAP サーバーに関する情報を要求することが必要な場合もあります。LDAP サーバーが認証を使用する場合は、ユーザー ID および権限を要求する必要があります。
 - データベースを作成する権限がない場合、データベースの計画にデータベース管理者 (DBA) が参加する必要があります。DBA は、カスタマイズして実行するためのデータベース・スクリプトのコピーが必要です。
3. セキュリティー、ユーザー ID、および許可の計画
4. 38 ページの『Business Process Choreographer のデータベースの計画』
5. 51 ページの『Business Flow Manager および Human Task Manager の計画』
6. 52 ページの『担当者ディレクトリー・プロバイダーの計画』
7. 55 ページの『Business Process Choreographer Explorer の計画』
8. 管理コンソールの「Business Process Choreographer の構成」ページを使用する場合は、構成ページに入力するすべての値を計画するようにしてください。
9. bpeconfig.jacl 構成スクリプトを使用する場合:
 - a. コマンド行またはバッチ・ファイルに指定する必要があるすべてのオプションおよびパラメーター値を計画するようにしてください。オプションおよびパラメーターの要約は、『bpeconfig.jacl スクリプトによる Business Process Choreographer の構成』にあり、詳細は『bpeconfig.jacl スクリプト』に記述されています。

- b. バッチ・ファイルを使用して `bpeconfig.jacl` 構成スクリプトを実行する場合、バッチ・ファイルまたはシェル・スクリプトを作成します。

タスクの結果

Business Process Choreographer のカスタム構成を作成するために必要なものすべての計画が完了しました。

次のタスク

Business Process Choreographer の構成を実行します。

セキュリティー、ユーザー ID、および許可の計画

Business Process Choreographer を構成する場合のユーザー ID と許可を計画します。

このタスクについて

構成時には、さまざまなユーザー ID を使用する必要があり、実行時に使用される他のユーザー ID を指定する必要があります。Business Process Choreographer の構成を始める前に、必ずすべてのユーザー ID を計画および作成してください。

サンプル Business Process Choreographer 構成の場合:

必要な権限は、新規プロファイルを作成する権限のみです。プロファイル管理ツールで標準プロファイル作成オプションを使用し、管理セキュリティーを有効にすると、Business Process Choreographer サンプルも構成されます。他の計画やユーザー ID は不要なので、このタスクはスキップできます。

ハイ・セキュリティー構成の場合:

このタスクでの説明に従って、すべてのユーザー ID を詳細に計画する必要があります。

低セキュリティー構成の場合:

非実動システムなど、完全なセキュリティーが不要な場合は、使用されるユーザー ID の数を減らすことができます。すべてのユーザー ID を詳細に計画する必要がありますが、特定のユーザー ID を複数の目的で使用できます。例えば、データベース・スキーマを作成するために使用するデータベース・ユーザー ID は、実行時にデータベースに接続するためのデータ・ソース・ユーザー名としても使用できます。

`bpeconfig.jacl` スクリプトを使用して Business Process Choreographer を構成する場合:

`bpeconfig.jacl` スクリプトを実行するために使用されるユーザー ID には、スクリプトが実行する構成アクションで必要とされる権限が付与されている必要があります。そうでない場合は、必要な権限を持つユーザー ID をスクリプトのパラメーターとして指定する必要があります。そのためすべてのユーザー ID を詳細に計画する必要があります。`bpeconfig.jacl` スクリプトに対するパラメーターとして指定可能なユーザー ID の場合は、パラメーター名がテーブルに取り込まれます。プロファイルが既に作成されている必要があります。WebSphere 管理セキュリティーが有効になっている場合は、`wsadmin` ツールを起動するために使用できる `configurator` ロールの WebSphere 管理者ユーザー ID が必要です。

ヒューマン・タスクを使用する場合:

WebSphere 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する必要があります。

手順

1. このページのハードコピーを印刷し、計画した値を最後の列に書き込みます。
Business Process Choreographer を構成するときの参照用に保管し、将来的に参照できるように記録を残します。
2. Business Process Choreographer を構成するために、WebSphere Process Server で使用するユーザー ID を計画します。

表 4. WebSphere Process Server 用のユーザー ID の計画

ユーザー ID またはロール	ユーザー ID を使用する場合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画されるユーザー ID
Business Process Choreographer を構成するユーザー	構成	管理コンソールにログインして管理スクリプトを実行する。	WebSphere 管理セキュリティーが有効になっている場合は、WebSphere 管理者またはコンフィギュレーターのロール。	
		bpeconfig.jacl スクリプトを実行して Business Process Choreographer を構成する場合。	スクリプトを実行する場合は、選択するオプションの必須項目としてユーザー ID も入力する必要があります。詳しくは、bpeconfig.jacl スクリプトを参照してください。	

3. *install_root* のサブディレクトリーにアクセスする必要があるユーザーを計画します。セキュリティー・ポリシーにより、これらのユーザーにこの種のアクセス権限を付与することが許可されない場合は、ディレクトリー内のファイルのコピーを与える必要が生じます。

表 5. *install_root* のサブディレクトリーへのアクセスの計画

ユーザー ID またはロール	ユーザー ID を使用する場合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画されるユーザー ID
データベース管理者	構成	次のデータベースをセットアップするスクリプトの実行 BPEDB: Business Process Choreographer 用データベースのデフォルト名。 OBSRVDB: Business Process Choreographer Explorer レポート作成機能用データベースのデフォルト名。	bpeconfig.jacl スクリプトを使用して Business Process Choreographer を構成する場合: bpeconfig.jacl により次のディレクトリーのサブディレクトリーに生成される createSchema.sql スクリプト (またはそのコピー) への読み取りアクセス権限: <i>profile_root/dbscripts/ProcessChoreographer/</i> データベース・スクリプト・ファイルを確認する場合: ディレクトリー内に作成されたデータベース・スクリプト (またはそのディレクトリー内のファイルのコピー) への読み取りアクセス。ディレクトリーは以下のとおりです。 <i>install_root/dbscripts/ProcessChoreographer/database_type</i> ここで、 <i>database_type</i> は以下のいずれかになります。 <ul style="list-style-type: none"> DB2zOSV8 DB2zOSV9 	
		Integration Developer	カスタマイズ	Lightweight Directory Access Protocol (LDAP) または Virtual Member Manager (VMM) 担当者ディレクトリー・プロバイダーで担当者割り当てを使用するには、サンプル XSL 変換ファイルのコピーをカスタマイズする必要があります。

4. Business Process Choreographer が使用するデータベースの作成、構成、およびアクセスに使用するユーザー ID を計画します。

表 6. BPEDB データベースのユーザー ID の計画

ユーザー ID またはロール	ユーザー ID を使用する場合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画されるユーザー ID
データベース管理者	構成前	BPEDB データベースを作成します。Oracle の場合は、BPEDB データベースを作成します。	データベースを作成します。	

表 6. BPEDB データベースのユーザー ID の計画 (続き)

ユーザー ID またはロール	ユーザー ID を使用する場 合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画される ユーザー ID
bpeconfig.jacl スクリプトを 実行するデー タベース管理 者または管理 者	構成	ユーザーまたはデータベース 管理者は、組み込み Derby デ ータベースを使用するのでな い限り、Business Process Choreographer データベースの スクリプトを実行する必要が あります。	BPEDB データベースの場合: テーブル 変更、接続、テーブル挿入、および索 引、スキーマ、テーブル、テーブル・ス ペース、およびビューの作成を行いま す。	
データ・ソー スのユーザー 名 bpeconfig.jacl スクリプトを 使用する場 合、これは -dbUser パラ メーターで す。	構成	「 テーブルの作成 」オプショ ンを選択する場合は、このユ ーザー ID を使用してデー タベース表が作成されます。	「 テーブルの作成 」構成オプションを使 用するには、このユーザー ID に、次 に示す BPEDB データベースに対する 操作の実行権限も付与する必要がありま す: テーブル変更、接続、テーブル挿 入、および索引、テーブル、ビューの作 成。	
	実行時	Business Flow Manager と Human Task Manager は、こ のユーザー ID を使用して BPEDB データベースに接続 します。	このユーザー ID に、次に示す BPEDB データベースに対する操作の実行権限も 付与する必要があります: 接続、テー ブル削除、テーブル挿入、テーブルとビ ューの選択、テーブルの更新。	
	サービスまた はフィックス バックの適用 後	データベース・スキーマは、 必要に応じてサービスの適用 後に自動的に更新されます。 この仕組みは、このユーザー ID に必要なデータベース権 限が付与されている場合にの み機能します。権限が付与さ れていない場合は、スキーマ の更新を手動で実行する必要 があります。	このユーザー ID に、次に示す BPEDB データベースに対する操作の実行権限も 付与する必要があります。テーブルの変 更、作成、挿入、および選択、デー タベースへの接続、索引とビューの作成お よび除去。	

5. Business Process Choreographer Explorer レポート作成機能を構成する場合は、レポート・データベースの作成、構成、およびアクセスに使用するユーザー ID を計画します。

表 7. レポート・データベースのユーザー ID の計画

ユーザー ID またはロール	ユーザー ID を使用する場 合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画される ユーザー ID
データベース 管理者	構成前	レポート・データベースを作 成します。Oracle の場合は、 レポート・データベースを作 成します。	データベースを作成します。	

表7. レポート・データベースのユーザー ID の計画 (続き)

ユーザー ID またはロール	ユーザー ID を使用する場 合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画される ユーザー ID
データベース 管理者または 管理者	構成	setupEventCollector ツール、 またはスキーマを作成する SQL スクリプトを実行しま す。	レポート・データベースの場合: テー ブル変更、接続、プロシージャ ー作成、テーブル挿入、およ びテーブル、テーブル・ス ペース、およびビューの作 成。 ユーザー定義関数の Java 実 装を使用する場合、ユーザー ID には、JAR ファイルをイ ンストールする権限も必要で す。	
Event Collector デ ータ・ソース のユーザー名	実行時	レポート・データベースに接 続します。レポート・デー タベースを使用しており、そ のデータベースで BPEDB デ ータベースを使用する場 合は、Business Process Choreographer データ・ソ ースの場合と同じユーザー 名を使用します。	データベースに接続します。	

6. Business Process Choreographer のメッセージング・エンジンのメッセージ・ストア用に別のデータベース (Derby Embedded およびファイル・ストア以外) を使用する場合は、データベースへのアクセスに使用するユーザー ID を計画してください。

表8. 事前構成 BPEME メッセージング・エンジン・データベースのユーザー ID の計画

ユーザー ID (User ID)	ユーザー ID を 使用する場 合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画されるユ ーザー ID
バス・データ・ ソースのユー ザー名 bpeconfig.jacl ス クリプトを使 用する場合、こ れは -medbUser パ ラメーターで す。	構成および実行 時	このユーザー名は、 BPEME データベースへ の接続と、必要なテー ブルおよび索引の作成に 使われます。	このユーザー ID に、次に示す BPEME データベースに 対する操作の実行権限も 付与する必要があります: 接続、テーブル削除、 テーブル挿入、テー ブルとビューの選 択、テーブルの更 新。	

7. Java Message Service (JMS) で使用する Business Process Choreographer のユーザー ID を計画します。

表9. JMS のユーザー ID の計画

ユーザー ID (User ID)	ユーザー ID を使用する 場合	ユーザー ID の使用目的	ユーザー ID に必要な権限	計画されるユーザー ID
JMS 認証ユーザー	実行時	システム統合バスの認証別名。 Business Process Choreographer の構成時に指定する必要があります。 bpeconfig.jacl スクリプトを使用する場合、このユーザー ID とそのパスワードは、-mqUser パラメーターと -mqPwd パラメーターです。	WebSphere ユーザー・レジストリーに存在するユーザー名でなければなりません。Business Process Choreographer バスのバス・コネクタ・ロールに自動的に追加されます。	
JMS API 認証ユーザー	実行時	Business Flow Manager JMS API 要求はすべて、このユーザー ID を使用して処理されます。 bpeconfig.jacl スクリプトを使用する場合、このユーザー ID とそのパスワードは、-jmsBFMRunAsUser パラメーターと -jmsBFMRunAsPwd パラメーターです。	このユーザー名は、WebSphere ユーザー・レジストリーに存在するユーザー名でなければなりません。	
エスカレーションの認証ユーザー	実行時	Human Task Manager のエスカレーションはすべて、このユーザー ID を使用して処理されます。 bpeconfig.jacl スクリプトを使用する場合、このユーザー ID とそのパスワードは、-jmsHTMRunAsUser パラメーターと -jmsHTMRunAsPwd パラメーターです。	このユーザー名は、WebSphere ユーザー・レジストリーに存在するユーザー名でなければなりません。	

8. Business Flow Manager および Human Task Manager の Java EE ロールのマップ対象グループまたはユーザー ID を計画します。

表 10. Business Flow Manager および Human Task Manager のセキュリティのロールの計画

ユーザー ID またはロール	ユーザー ID を使用する場合	ユーザー ID の使用目的	ユーザー ID、グループ、またはこの両方の計画済みリスト
管理者ユーザー	実行時	Business Flow Manager と Human Task Manager の両方のセキュリティ・ロールであるシステム管理者またはシステム・モニターはそれぞれ、ユーザー ID のリスト、グループのリスト、またはこの両方のリストにマップされます。ここで定義される値により、必要とするアクセス権限をこのロールのユーザーに与えるマッピングが作成されます。 bpeconfig.jacl スクリプトを使用する場合、これらのユーザーおよびグループは次のパラメーターに対応します。 <ul style="list-style-type: none"> • -adminUsers • -adminGroups • -monitorUsers • -monitorGroups 	
管理者グループ	実行時		
モニター・ユーザー	実行時		
モニター・グループ	実行時		

9. Business Flow Manager および Human Task Manager のクリーンアップ・サービス、およびプロセス・インスタンス・マイグレーション・ツールなどの管理ジョブで Java EE run-as ロールとして使用するユーザー ID を計画します。このユーザー ID は、表 10 で計画した管理者ロールのユーザーまたはグループのメンバーでなければなりません。

表 11. 管理ジョブを実行するためのユーザー ID の計画

ユーザー ID (User ID)	ユーザー ID を使用する場合	ユーザー ID の使用目的	計画されるユーザー ID
管理ジョブ・ユーザー ID	ランタイム管理	このユーザー ID を使用して、管理ジョブを実行します。 bpeconfig.jacl スクリプトを使用する場合は、このユーザー ID とパスワードが -adminJobUser パラメーターと -adminJobPwd パラメーターに対応します。	

10. ヒューマン・タスクのエスカレーションで特定のビジネス・イベントについての通知 E メールを送信し、使用する Simple Mail Transfer Protocol (SMTP) サーバーで認証を必要とする場合は、E メール・サーバーに接続するために使用するユーザー ID を決定します。

表 12. E メール・サーバーのユーザー ID の計画

ユーザー ID またはロール	ユーザー ID を使用する場合	ユーザー ID の使用目的	ユーザー ID に必要な 権限	計画されるユ ーザー ID
メール・トラン スポートの ユーザー	実行時	Human Task Manager は、エスカレーション E メールを送信する構成済みメール・サー バーに対する認証でこのユーザー ID を使用 します。 bpeconfig.jacl スクリプトを使用する場合、 これは -mailUser パラメーターです。パスワ ードは -mailPwd パラメーターです。	E メールを送信する。	

11. ヒューマン・タスクで担当者割り当てを使用し、簡易認証を使用する Lightweight Directory Access Protocol (LDAP) 担当者ディレクトリー・プロバイダーを使用する場合は、LDAP サーバーへの接続に使用する Java 認証・承認サービス (JAAS) 別名および関連付けられたユーザー ID を計画します。LDAP サーバーで匿名認証を使用する場合、この別名およびユーザー ID は不要です。

表 13. LDAP サーバーの別名およびユーザー ID の計画

ユーザー ID またはロール	ユーザー ID を使用する場合	別名およびユーザー ID の使用目的	ユーザー ID に必要な権限	計画され る別名お よびユー ザー ID
LDAP プラグ イン・プロパテ ィー: 認証別名	実行時	この別名は、LDAP サーバーに接続 するときに使用するユーザー ID を 検索する目的で使用します。この別 名 ID は、LDAP プラグインのプロ パティをカスタマイズするときに 指定します。例えば、mycomputer/My LDAP Alias と指定します。	JAAS 別名は、LDAP ユーザー ID と関連付ける必要があります。	
LDAP ユーザ ー ID	実行時	このユーザー ID は、LDAP サーバ ーに接続するときに使用します。	LDAP サーバーで簡易認証を使用 する場合は、このユーザー ID で LDAP サーバーに接続できるよう にする必要があります。このユー ザー ID は、ショート・ネームま たは識別名 (DN) です。LDAP サ ーバーに識別名が必要な場合は、 ショート・ネームを使用すること はできません。	

12. 必要な権限を付与して、計画したユーザー ID を作成します。すべてを自分で作成する権限を持っていない場合は、適切な管理者に要求を実行依頼し、管理者によってユーザー用に作成されるユーザー ID の名前をこのテーブルに入力します。

タスクの結果

Business Process Choreographer の構成時に必要とされるユーザー ID がわかりま
す。

Business Process Choreographer のデータベースの計画

Business Process Choreographer のデータベースを計画します。セットアップによっては、最大 3 つのデータベースを作成するか、またはデータベースをまったく作成しないかを計画しておく必要があります。

このタスクについて

Business Process Choreographer は、他のプロセス・サーバー・コンポーネントや他の Business Process Choreographer 構成とデータベースを共用できます。BPEDB データベースは、Business Flow Manager と Human Task Manager で使用されます。実動システムでは、Business Process Choreographer が構成されているデプロイメント・ターゲットごとに専用データベースを設定することを計画します。

Business Process Choreographer の構成が複数ある場合は、それぞれの構成で独自のデータベースまたはデータベース・スキーマが必要になります。Business Process Choreographer の複数の構成で Business Process Choreographer データベース表を共用することはできません。

制約事項: Informix を使用している場合、複数の Business Process Choreographer 構成で同じデータベースを共用することはできません。

Business Process Choreographer Explorer レポート作成機能 (バージョン 6.1.2 までの Business Process Choreographer Observer) を使用する場合は、同じ BPEDB データベースを使用できますが、追加データベースを使用するとパフォーマンスが改善されます。レポート・データベースをセットアップするためのスクリプトの一部には、推奨される名前 OBSRVDB が既に含まれていますが、別の名前を選択することもできます。

Business Process Choreographer メッセージング・エンジンは、SCA メッセージング・エンジンとデータベースを共用するか、または専用の BPEMEDB データベースを使用できます。選択した構成パスでサポートされているデータベースの詳細については、17 ページの表 2 を参照してください。

手順

1. 実動システムの場合:

- a. パフォーマンスの重要性が高い場合は、40 ページの『BPEDB データベースの計画』の説明に従い、Business Process Choreographer に個別のデータベースを使用することを計画します。パフォーマンスの重要性が低い場合は、WPRCSDB 共通データベースを使用することを計画します。
- b. Business Process Choreographer Explorer レポート作成機能を使用する場合:
 - ビジネス・プロセスのパフォーマンスに及ぶ照会の影響を最小限に抑えるには、45 ページの『レポート・データベースの計画』の説明に従い、個別のデータベース (OBSRVDB) を使用するように計画します。
 - それ以外の場合は、BPEDB データベースを使用するように構成することを計画します。
- c. メッセージング・レートが非常に高い大規模クラスターなど、高負荷セットアップの場合、Business Process Choreographer メッセージング・エンジンに個別のデータベースを使用することでパフォーマンスを向上させることを検

討してください。これによって、データベース・ロギングが並列化され、ボトルネックになることを防ぐことができます。

- 管理コンソールを使用して Business Process Choreographer を構成するときに、Business Process Choreographer のメッセージング・エンジンで別個のデータベースを使用する場合は、49 ページの『メッセージング・エンジン・データベースの計画』を実行します。それ以外の場合は、Service Component Architecture (SCA) が使用するデフォルトのデータベースを使用するように計画します。
 - bpeconfig.jacl 構成スクリプトを使用して Business Process Choreographer を構成する場合、Business Process Choreographer は SCA によって使用されるものと同じ種類のメッセージ・ストアを使用します。
- d. オプション: データベース設計ツールを使用して、データベース設計ファイルおよび SQL スクリプト・ファイルを対話式で作成します。データベース管理者は、作成されたファイルを使用して、これまでのステップで計画した 3 つのすべてのデータベースを作成することができます。このツールを使用することには、以下のような大きなメリットがあります。
- ツールを必要に応じて随時実行して、データベース設計パラメーターを調整することができます。提供される SQL テンプレート・ファイルを手動で編集する方法とは異なり、データベース設計パラメーターを破損させるというリスクを回避できます。
 - データベース設計ファイルを使用している場合は、次回、新しいバージョンの WebSphere Process Server にマイグレーションするときに、スキーマ更新 SQL スクリプトを生成することができます。
 - テスト構成用のデータベース設計ファイルを作成する場合は、設計ファイルのコピーを作成してそれに若干の変更を加え、実動システムのデータベースに使用できるので便利です。
 - また、このツールを使用して、3 つのすべてのデータベースのデータ・ソースを定義することができます。ただし、レポート・データベース用のデータ・ソースは手動で構成する必要があります。

重要: データベース設計ツールを使用してデプロイメント環境を構築すると、共通データベースを構成した後に Business Process Choreographer が「完全」なものとして表示されます。これは、有効なデフォルトが存在し、その結果、Business Process Choreographer のテーブルが共通データベースに作成されるためです。ただし、このデフォルトは実動システムには適していません。実動システムでは、必ず、Business Process Choreographer が構成されるデプロイメント・ターゲットごとに専用データベースを構成してください。

2. パフォーマンスよりもセットアップの単純さの重要性の方が高い非実動システムの場合、選択するオプションは、選択した構成パスによって決まります。
- インストーラーまたはプロファイル管理ツールを使用して『基本サンプル』または『組織付きサンプル』の Business Process Choreographer 構成を作成する場合は、別の Derby Embedded BPEDB データベースが作成されます。このデータベースは、Business Process Choreographer Explorer レポート作成機能でも使用されます。Business Process Choreographer のメッセージング・エンジンの場合は、デフォルトで別個の Derby Embedded データベース (BPEME) が

作成されます。プロファイル管理ツールを使用する場合、「ファイル・ストア」を使用するか、または WPRCSDB データベースを共有するかを選択することもできます。

- インストーラーまたはプロファイル管理ツールを使用して、Business Process Choreographer 構成が含まれているデプロイメント環境を作成する場合、Business Process Choreographer、Business Process Choreographer Explorer レポート作成機能、および Business Process Choreographer メッセージング・エンジンはすべて WPRCSDB データベースを使用します。したがって、Business Process Choreographer のデータベースに関する計画は不要です。

タスクの結果

これで、Business Process Choreographer 構成のすべてのデータベースの計画が完了しました。

BPEDB データベースの計画:

Business Process Choreographer のデータベースについて計画を立てます。

このタスクについて

Business Process Choreographer にはデータベースが必要です。サポート対象のすべてのデータベース・システムには、データベース・スキーマを作成して管理するための SQL スクリプトが提供されています。データベースが配備されたら、データベースへの JDBC アクセスを Business Process Choreographer 用に構成する必要があります。データベース・システム、使用するトポロジー、インストールの目的、および使用する管理ツールによっては、データベースを作成して JDBC アクセスを構成するための一部のタスクまたはすべてのタスクを自動化することができます。実動システムの場合、Business Process Choreographer に独自のデータベースが必要ですが、パフォーマンスが重要でない場合は、他の WebSphere Process Server コンポーネントとデータベースを共有するように Business Process Choreographer を構成することもできます。

手順

1. 選択した BPEDB データベースと構成パスに互換性があることを確認します。以下のデータベースがサポートされています。

- Derby

注: Derby ではデータベース・アクセスがシリアライズされます。したがって、アクティビティの並列実行をサポートするようにモデル化されたフローであっても、アクティビティは常に順次実行されます。

- DB2 for z/OS

Business Process Choreographer を構成する方法を既に決定している場合、構成パスの選択が、データベースの作成方法に影響します。Business Process Choreographer の構成に使用する構成パスをまだ決定していない場合、データベース要件を識別することが、ニーズを満たさない構成パスを除外するのに役立ちます。それぞれの構成パスでサポートされているデータベースの詳細については、17 ページの表 2 を参照してください。

2. Business Process Choreographer では Derby データベースが使用されます。応答ファイルを使用する場合は、DB2 for z/OS が使用されます。
3. 実動システムに通常求められるパフォーマンス、スケーラビリティ、およびセキュリティを必要としない場合、データベース・オブジェクトは、WebSphere Process Server に対してローカルであるデータベース・サーバー上の単一のテーブル・スペース内に作成することができます。これにより、データベースの作成に必要な計画と労力を最小限に抑えることができますが、データベースへのアクセスに使用されるユーザー ID が、データベース管理権限を持つことが必要になります。計画する必要があるオプションは、以下のよう
に、選択する構成パスに依存します。
 - a. インストーラーまたはプロファイル管理ツールを使用して、Business Process Choreographer のサンプル構成を取得する場合、Business Process Choreographer 用に別個の Derby BPEDB データベースが作成されます。この場合、これ以上の計画は不要です。
 - b. 管理コンソールのデプロイメント環境ウィザードを使用して Business Process Choreographer を構成するとき、デフォルト・スキーマを単一テーブル・スペース内に作成すれば十分な場合、提供された SQL スクリプトのコピーを使用して BPEDB データベースを作成するように計画してください。
 - c. **bpeconfig.jacl** ツールを使用して Business Process Choreographer を構成する場合、次の中からどれが自身のケースに該当するかを計画します。
 - bpeconfig.jacl スクリプトを対話モードで実行する場合、既存のデータベース内に表を作成することを選択できます。
 - データベース・オブジェクトを作成する権限を持つユーザー ID を使用している場合、-createDB yes オプションが使用できます。このオプションによって bpeconfig.jacl スクリプトでは、デフォルトのテーブル・スペース内にデータベース・オブジェクトを作成する SQL ファイルが生成されて実行されます。この場合、サーバーを停止し、wsadmin ユーティリティで -conntype NONE オプションを使用することも計画します。
 - Derby データベースを使用している場合、bpeconfig.jacl はデータベース・インスタンスを作成します。DB2 for z/OS データベースを使用している場合、データベース・インスタンスが既に存在している必要があります。
 - データベースまたはオブジェクトの作成中にエラーが発生した場合、-createDB no オプションを使用すると同様の方法で、生成された SQL スクリプトを使用することができます。
 - データベース・オブジェクトを作成する権限を持つユーザー ID を使用しない場合、-createDB no オプションを使用する必要があります。このオプションによって bpeconfig.jacl スクリプトでは、デフォルトのテーブル・スペース内にデータベース・オブジェクトを作成する SQL ファイルが生成されますが、bpeconfig.jacl スクリプトは SQL ファイルのスクリプトを実行しません。この場合、スクリプトをカスタマイズして実行するようデータベース管理者に依頼することを計画します。
- d. 管理コンソールの「**Business Process Choreographer の構成**」ページを使用する場合は、以下のようになります。

このツールおよび他のデータベース・パラメーターについて詳しくは、bpeconfig.jacl スクリプトを参照してください。

- Business Process Choreographer データベース・オブジェクトを共通データベース WPRCSDB 内で作成する場合、デフォルト・データベースを Business Process Choreographer データ・ソースのターゲットとして使用することを計画します。
 - 既存のデータベースを再利用する場合、既存のデータベースを Business Process Choreographer データ・ソースのターゲットに指定することを計画します。
 - 「テーブルの作成」オプションを選択する場合、Business Process Choreographer では、データベースを最初に使用するときにデフォルトのテーブル・スペース内で必要とするデータベース・オブジェクトを作成します。このオプションは、DB2 on z/OS データベースあるいはリモート Oracle データベースの場合は使用できません。このオプションを DB2 UDB データベースで使用するには、データベースで AUTOMATIC STORAGE YES を使用可能にしておく必要があります。
 - スクリプトを使用してデータベースを作成する場合、「テーブルの作成」オプションを使用しないように計画します。
- e. ステップ 14 (44 ページ) にスキップします。
4. Business Process Choreographer に対して以下の特性を持つハイパフォーマンス・データベース・セットアップを適用する場合、以下のすべてのステップを実行します。
- データベースは Business Process Choreographer によってのみ使用される。
 - データベースは専用サーバー上に存在するのが理想的だが、WebSphere Process Server システムに対してローカルであっても構わない。
 - ディスクに対するテーブル・スペースの割り振りをカスタマイズしてパフォーマンスを高めることができる。
 - データベース管理用のユーザー ID とは別のユーザー ID を使用してデータベースにアクセスできる。
5. データベースのユーザー ID をまだ計画していない場合、32 ページの表 6 を実行します。
6. ディスクおよびテーブル・スペースの割り振りを計画します。データベース・ホストは、Network Attached Storage またストレージ・エリア・ネットワークなどの高速ストレージ・サブシステムを使用することが理想的です。実動システムの場合、開発およびシステム・テストから得た経験を考慮してください。データベースのサイズは多くの要因に依存します。microflow として実行されるプロセスは、ほとんどスペースを使用しませんが、各プロセス・テンプレートには数十キロバイトから数百キロバイトが必要な場合があります。

個別のディスクを使用する予定であり、ご使用のデータベース・システムでデータベース表を複数のディスクに割り振ることができる場合は、使用するディスクの数とディスクの割り振り方法を計画します。一般に、ハードウェア補助のディスク・アレイは単一ディスクよりも、パフォーマンスが優れています。

DB2 for z/OS の場合、テーブル・スペースが各表に作成され、LOB 列に追加のラージ・オブジェクト (LOB) テーブル・スペースが作成されます。これらはすべて、1 つの高性能 RAID アレイ上に存在させることができますが、並列アクセスを可能にするために、各テーブル・スペースは別のファイルに作成す

る必要があります。特定の数のディスクでは、RAID 構成を使用する方が、テーブル・スペースを別個のディスクに割り振るよりもパフォーマンスが高くなることに留意してください。例えば、N 個のプロセッサを搭載した専用サーバー上で DB2 データベースが稼働中の場合、次のガイドラインを適用することを検討してください。

- テーブル・スペースについては、プライマリー・ディスクが $2*N$ 個、ミラー・ディスクが $2*N$ 個の RAID-1 アレイを使用し、ストライプ・サイズは 256 KB とします。
 - データベース・トランザクション・ログについては、プライマリー・ディスクが $1.5*N$ 個、ミラー・ディスクが $1.5*N$ 個の RAID-1 アレイを使用し、ストライプ・サイズは 64 KB とします。
7. データベース・オブジェクトを作成する SQL スクリプトを実行する前に、ユーザー自身またはデータベース管理者がこのスクリプトをカスタマイズするように計画します。
- **bpeconfig.jacl** ツールを使用して Business Process Choreographer を構成する場合、`-createDB no` オプションを使用します。これにより、このツールで生成される SQL スクリプトが実行されません。生成される SQL ファイルは、データベース用に提供された元の SQL ファイルに基づいていますが、**bpeconfig.jacl** ツールに提供されたすべての構成パラメーターが、SQL ファイルに事前に入力されるため、必要なカスタマイズが最小限に抑えられます。
 - 管理コンソールの「**Business Process Choreographer 構成**」ページまたは**デプロイメント環境ウィザード**を使用して Business Process Choreographer を構成する場合、「テーブルの作成」オプションをクリアして、デフォルト・スキーマを取得しないようにします。生成される SQL ファイルは、データベース用に提供された元の SQL ファイルに基づいていますが、管理コンソールに入力したすべての構成パラメーターが、生成される SQL ファイルに事前に入力されるため、必要なカスタマイズが最小限に抑えられます。

生成される SQL スクリプトの使用については、生成済みの SQL スクリプトを使用した Business Process Choreographer 用のデータベース・スキーマの作成を参照してください。実施するカスタマイズについて計画できるように、データベースのオリジナルの SQL ファイルのプレビューを表示するには、ご使用のデータベースの SQL スクリプト `createSchema.sql` を見つけて表示します。ただし、このスクリプトを変更しないでください。元の SQL ファイルは、次のディレクトリーにあります。

`install_root/dbscripts/ProcessChoreographer/database_type`ここで、`database_type` は以下のいずれかになります。

- DB2zOSV8
 - DB2zOSV9
8. データベース・サーバーが WebSphere Process Server システムに対してリモートである場合、Java Database Connectivity (JDBC) ドライバーまたはデータベース・クライアントのいずれかを WebSphere Process Server システムにインストールするように計画します。
- タイプ 2 JDBC ドライバーの場合: インストールするデータベース・クライアントとインストールする場所を決定します。

- タイプ 4 JDBC ドライバーの場合: ドライバーの JAR ファイル (製品インストールの一部として提供されている) を見つけ、インストールする場所を決定します。
9. データベース・サーバーがプロセス・サーバーに対してローカルである場合、データベースへのアクセスに必要な JDBC JAR ファイルはデータベース・システムと共にインストールされています。これらの JAR ファイルを見つければ、場所をメモしておきます。
 10. DB2 for z/OS を使用する場合は、使用するサブシステムを決定します。スクリプト・ファイル createTablespace.sql および createSchema.sql 内のストレージ・グループ名、データベース名 (サブシステム名ではない)、およびスキーマ修飾子を置き換える値を計画します。
 11. DB2 for z/OS Universal JDBC ドライバーのプロバイダーおよびデータ・ソースについては、要件を調べてください。
 12. データベースをホストするサーバーを決定します。データベース・サーバーがリモートである場合、適切なデータベース・クライアントか、XA をサポートするタイプ 4 JDBC ドライバーが必要です。
 13. データベースに指定する必要がある次の構成パラメーターの値を決定します。
 - Java Database Connectivity (JDBC) プロバイダーは、タイプ 2 またはタイプ 4 が可能です。
 - サブシステム名。
 - ストレージ・グループ名。
 - データベース名。
 - スキーマ修飾子。

制約事項: Informix データベースを使用する場合は、スキーマ修飾子を使用できるようにするため、ANSI モードで作成する必要があります。現在は 1 つのスキーマのみサポートできます。

- スキーマを作成するユーザー名。
 - データベース・サーバーの名前または IP アドレス。
 - データベース・サーバーによって使用されるポート番号。これは、タイプ 4 JDBC ドライバーを使用する場合にのみ必須です。
 - 認証別名用のユーザー ID およびパスワード。これは、jdbc/BPEDB データ・ソースが実行時にデータベースにアクセスするために使用するユーザー ID です。これらは bpeconfig.jacl の -dbUser および -dbPwd パラメーターです。
14. 十分な数の並列 JDBC 接続をサポートするように計画します。
 - a. Business Process Choreographer BPEDB データベースに必要な、並列 JDBC 接続の最大数を推定します。これはビジネス・プロセスの性質とユーザー数に依存します。適切な推定値は、Business Process Choreographer API を介して同時に接続できるクライアントの最大数と、JMS アクティベーション・スペック BPEInternalActivationSpec および HTMInternalActivationSpec で定義される並行エンドポイントの数と、過負荷状態を許容するための 10% の安全マージンを加えた数です。
 - b. データベース・システムが、必要な数の並列 JDBC 接続をサポートできることを確認します。

- c. ご使用のデータベース・システムで予想される数の並列 JDBC 接続をサポートするためのベスト・プラクティスに基づき、最適な設定を計画します。
15. 実動システムの場合、次の管理タスクについて計画します。
- 標準的な実動データをデータベースに取り込んだ後、データベースを調整します。
 - 完了したプロセス・インスタンスおよびタスク・インスタンスを、データベースから定期的に削除します。利用可能なツールおよびスクリプトの概要については、『Business Process Choreographer のクリーンアップ手順』を参照してください。

タスクの結果

これで、Business Process Choreographer のデータベースの計画が完了しました。

レポート・データベースの計画:

Business Process Choreographer Explorer レポート作成機能のデータベースを計画します。

このタスクについて

Business Process Choreographer Explorer レポート作成機能は同じデータベースを使用することができますが、追加データベースを使用するとパフォーマンスが改善されます。BPEDB データベースを再利用しない場合、以下のことを実行してください。

手順

1. 複数の Event Collector インスタンスを使用することを計画していて、これらのインスタンスが同じデータベースを使用する場合は、Event Collector ごとに固有のスキーマ名を使用するよう計画します。パフォーマンスを改善するには、Event Collector ごとに個別のデータベースを使用することを計画します。
2. データベースに使用するデータベース・システムを以下から決定します。

注: Business Process Choreographer Explorer レポート作成機能では、Derby または DB2 のいずれかのデータベースが必要です。Derby ではデータベース・アクセスがシリアライズされます。したがって、アクティビティの並列実行をサポートするようにモデル化されたフローであっても、アクティビティは常に順次実行されます。

3. データベースをホストするサーバーを決定します。
4. データベースのユーザー ID をまだ計画していない場合、33 ページの表 7 を実行します。
5. レポート・データベースに Derby データベースを使用しない場合、SQL ベースまたは Java ベースのいずれのユーザー定義関数 (UDF) を使用するかを決定します。
 - Java UDF の方が正確ですが、これを使用するにはデータベースに JAR ファイルをインストールする必要があります。

- DB2 for z/OS データベースを使用し、SQL ベースの UDF でなく Java ベースの UDF を使用してデータベースを作成する場合、メニュー方式の管理ツール `setupEventCollector` を使用する以外に選択肢はありません。
- Derby データベースを使用する場合は、組み込み Derby データベースは SQL UDF をサポートしないため、Java ベースの UDF が使用されます。

UDF については、Business Process Choreographer Explorer レポート作成機能のユーザー定義関数を参照してください。

6. Business Process Choreographer Explorer レポート作成機能および Event Collector が BPEDB データベースを使用するように構成するために `bpeconfig.jacl` スクリプトを使用しない場合、レポート・データベースを作成する方法を決定します。

メニュー方式の管理ツール `setupEventCollector` の使用

このツールを使用すると、対話モードでデータベースを作成し、入力をランタイム環境に突き合わせて検証できます。このツールを使用する場合は、ツールで SQL ファイルを作成するがファイルを実行しないようにするかどうかを決定します。このオプションは、SQL を実行する前にカスタマイズする場合、または SQL ファイルのカスタマイズと実行をデータベース管理者に依頼する場合に使用します。このツールについて詳しくは、`setupEventCollector` ツールを参照してください。

他のデータベースの作成方法とは異なり、このツールでは、Java ベースのユーザー定義関数 (UDF) または SQL ベースの UDF のいずれかを作成することができます。また、このツールを使用して、これらの 2 つのオプションを切り替えたり、UDF をサポートするために必要な JAR ファイルをインストールおよび削除することができます。DB2 on z/OS データベースの場合、このツールは、Java ベースの UDF または SQL ベースの UDF のいずれかを使用したデータベースの作成をサポートしています。Derby データベースの場合、データベースの作成には、Java ベースの UDF のみが使用されます。

SQL スクリプトの実行

データベースにアクセスするツールの使用を許可されていない場合、SQL スクリプトを使用することが必要な場合もあります。`bpeconfig.jacl` スクリプトをバッチ・モードで使用するか、または管理コンソールを使用して Business Process Choreographer を構成した場合、必要なすべてのパラメーターが置換済みの SQL スクリプトが生成されています。その他の場合、データベース設計ツールを使用して、SQL スクリプトを対話式に生成することができます。

DB2 on z/OS データベースの場合、スクリプトは SQL ベースの UDF を使用するため、Java に対して有効なワークロード・マネージャー (WLM) 環境をセットアップする必要はありません。Derby データベースの場合、データベースの作成には、Java ベースの UDF のみが使用されます。

初回使用時にテーブルを自動的に作成する

管理コンソールの Business Process Choreographer Event Collector の構成ページで「**テーブルの作成**」オプションを選択することが、デフォルトのデータベース・スキーマを取得するための最も簡単な方法です。こ

のオプションは、ハイパフォーマンス・システムには適していません。
このオプションは、DB2 on z/OS データベースで使用できません。
Derby データベースの場合、データベースの作成には、Java ベースの
UDF のみが使用されます。

注: Derby Network Server データ・ソースを使用する場合、Derby
Network Server は *install_root/derby/bin/networkServer* ディレクトリ
から開始する必要があります。そうしない場合、テーブルの作成は
次のエラーで失敗します。CWWB04013E: bpcodbutil.jar ファイルが
Derby ネットワーク・サーバーで見つかりませんでした。

7. DB2 for z/OS データベースを使用する場合は、以下のことを計画します。

- サブシステムのロケーション名 (ネットワーク名)。
- ストレージ・グループ名。
- サブシステムによって認識されるデータベース名。デフォルト値は
OBSRVADB です。
- データベースへの接続に使用するユーザー ID。このユーザー ID のパスワードも
知っておく必要があります。
- データベース・オブジェクトが作成されるとき、基になるデータベース・スキーマ名
(SQLID)。
- 以下のテーブル・スペースの作成先となるストレージ・グループを計画します。
 - 通常のテーブル・スペース (OBSVR01、OBSVR02、OBSVR03、OBSVR04、OBSVR05、OBSVR06、OBSVR07、および OBSVR08 の場合)。
 - LOB テーブル・スペース (OS26201、OS26202、OS26203、および OS26204 の場合)。
- デフォルトの SQL ベースのユーザー定義関数 (UDF) でなく、Java ベースの UDF を
使用する場合は、関数の実行に使用する WLM 環境の名前を決定します。
- setupEventCollector ツールを使用してデータベースをセットアップする場合、
以下についても計画します。
 - 使用する JDBC ドライバーのタイプを以下から決定します。
 - タイプ 4 は JDBC 経由で直接接続します。この場合、以下についても知
っておく必要があります。
 - データベース・サーバーのホスト名または IP アドレス。デフォルトは
localhost です。
 - データベースに使用するポート番号。デフォルトは 446 です。
 - JDBC ドライバーの JAR ファイル db2jcc.jar および db2jcc_license_cisuz.jar
のディレクトリ。
 - タイプ 2 はネイティブのデータベース・クライアントを使用して接続
します。この場合、ローカル・カタログに含めるデータベース別名も計
画します。
- DB2 for z/OS Universal JDBC ドライバーのプロバイダーおよびデータ・ソース
については、要件を調べてください。

8. Derby データベースを使用する場合は、以下のことを計画します。
- データベース名。これはサーバーのファイル・システム上の完全修飾パスである必要があります。デフォルト値は `install_root/databases/BPEDB` です。
 - データベース・オブジェクトが作成される時、基になるデータベース・スキーマ名。デフォルト値は `APP` です。
 - `setupEventCollector` ツールを使用してデータベースをセットアップする場合、以下についても計画します。
 - Derby Network JDBC ドライバーを使用する場合は、データベースへの接続に使用するユーザー ID を計画します。このユーザー ID のパスワードも知っておく必要があります。
 - 使用する JDBC ドライバーのタイプを以下から決定します。
 - Embedded JDBC ドライバーまたは Embedded 40 JDBC ドライバー。この場合、JDBC ドライバーの JAR ファイル `derby.jar` のディレクトリも計画します。デフォルト・ロケーションは `install_root/derby/lib` です。
 - Network JDBC ドライバーまたは Network 40 JDBC ドライバー。この場合、以下についても知っておく必要があります。
 - JDBC ドライバーの JAR ファイル `derbyclient.jar` のディレクトリ。デフォルト・ロケーションは `install_root/derby/lib` です。
 - Derby Network Server を使用する場合、Derby Network Server 上の UDF JAR ファイル `bpcodbutil.jar` のロケーションを決定します。デフォルト・ロケーションは `install_root/derby/lib` です。
 - データベース・サーバーのホスト名。デフォルトは `localhost` です。
 - データベースに使用するポート番号。デフォルトは `1527` です。
9. `bpeconfig.jacl` ツールを `-createEventCollector yes` オプションを指定してバッチ・モードで使用する場合は、以下のいずれかを計画します。
- `-createDB yes` オプションを指定すると、`bpeconfig.jacl` によって生成された SQL スクリプトがツールによって実行されます。`-dbSchema` パラメーターを使用して、`BPEDB` データベースのスキーマ修飾子を指定することができます。また、`-reportSchemaName` および `-reportDataSource` パラメーターを使用して、Business Process Choreographer Explorer レポート作成機能で `BPEDB` データベースを使用せずに別のデータベースを使用することができます。
 - `-createDB no` オプションを指定すると、ツールで生成された SQL スクリプトがツールで実行されるのを回避します。生成される SQL ファイルは、データベース用に提供された標準の SQL ファイルに基づいていますが、`bpeconfig.jacl` ツールに提供されたすべての構成パラメーターが、SQL ファイルに事前に入力されるため、必要なカスタマイズが最小限に抑えられます。生成された SQL スクリプト (データベース・オブジェクト作成スクリプト) を実行する前に、ユーザー自身またはデータベース管理者がこのスクリプトをカスタマイズするように計画します。このツールおよび他のデータベース・パラメーターについて詳しくは、`bpeconfig.jacl` スクリプトによる Business Process Choreographer の構成を参照してください。

10. 管理コンソールの **Business Process Choreographer Event Collector** ページを使用してデータベース表を作成する場合、以下のいずれかを計画します。
 - Derby データベースの場合、「テーブルの作成」オプションを使用して、Business Process Choreographer が指定されたデータベースに最初にアクセスするときに、データベース内にデフォルト・スキーマをツールによって作成することができます。
 - SQL スクリプトを実行してデータベース表を作成する場合、「テーブルの作成」オプションを使用しないでください。データベース・オブジェクトを作成する SQL スクリプトのコピーを実行する前に、これを自身でカスタマイズするか、またはデータベース管理者にカスタマイズを依頼することを計画します。このオプションは、実動システムに最も適しています。
11. 実施するカスタマイズについて計画できるようにするためにデータベース用 SQL スクリプトのプレビューを表示するには、以下のようになります。データベース用の `createSchema_0bserver.sql` ファイルを見つけて表示します。ただし、ファイルを変更しないでください。SQL ファイルは、次の場所にあります。

`install_root/dbscripts/ProcessChoreographer/database_type`

ここで、`database_type` は以下のいずれかになります。

- DB2zOSV8
- DB2zOSV9
- Derby

注: `bpeconfig.jacl` ツールを使用して Business Process Choreographer を構成する場合、ツールが生成する SQL スクリプトを使用することを計画します。このスクリプトは、構成パラメーターのプレースホルダーを値に置き換えるように編集する必要はありません。生成されるスクリプトはツールを実行した後でのみ利用できますが、これらは上記のロケーションにあるスクリプトに基づいています。テーブル・スペースの割り振りをカスタマイズする場合は、生成されたスクリプト・ファイルを編集する必要があります。あるいは、データベース設計ツールを使用して、SQL スクリプトを生成することもできます。

タスクの結果

これで、レポート・データベースの計画が完了しました。

メッセージング・エンジン・データベースの計画:

データベース・ロギングがボトルネックになる可能性がある高負荷設定の場合、Business Process Choreographer バスのメッセージング・エンジンに別のデータベースを使用することで、パフォーマンスを改善できます。

このタスクについて

Service Component Architecture (SCA) システム・バスの各メッセージング・エンジン、SCA アプリケーション・バスの各メッセージング・エンジン、Common Event Infrastructure バスの各メッセージング・エンジン、および Business Process Choreographer バスの各メッセージング・エンジンに対して同じメッセージング・データベースを使用できます。このデータベースは、メッセージ・エンジンのフェイ

ルオーバー可用性を確保するために、メッセージ・エンジンのホストとして動作するクラスターのすべてのメンバーがアクセス可能である必要があります。パフォーマンスの重要性が高い場合は、SCA バスおよびアプリケーションに使用されるデフォルト MEDB を使用する代わりに、Business Process Choreographer メッセージング・エンジン専用のデータベースを使用するように計画します。

手順

1. インストーラーまたはプロファイル管理ツールを使用していずれかのサンプル Business Process Choreographer 構成を取得する場合は、Business Process Choreographer メッセージング・エンジンに Derby Embedded、ファイル・ストア、または WPRCSDB データベースを使用するかどうかを決定します。
2. Java Database Connectivity (JDBC) プロバイダー。Network Deployment 環境ではファイル・ストア・データベースと組み込み Derby データベースは使用できない点に注意してください。
3. WebSphere MQ を使用するには、bpeconfig.jacl 構成スクリプトを使用して Business Process Choreographer を構成する必要があります。WebSphere MQ を使用すべきではありません。
4. bpeconfig.jacl 構成スクリプトを使用して Business Process Choreographer を構成する場合、Business Process Choreographer は SCA によって使用されるものと同じ種類のメッセージ・ストアを使用します。
 - SCA で FILESTORE が使用される場合、Business Process Choreographer でも FILESTORE が使用されます。
 - SCA で Derby Embedded データベースが使用される場合、Business Process Choreographer ではそれ独自の Derby Embedded データベースが使用されます。
 - SCA でそれら以外のデータベースが使用される場合、Business Process Choreographer では同じデータベース内でそれ独自のスキーマを使用します。
5. 管理コンソールの Business Process Choreographer 構成ページを使用するとき、SCA メッセージ・ストア設定に基づくデフォルトの構成を使用する必要がある場合は「デフォルト構成の使用」チェック・ボックスを選択し、それ以外の場合は以下の構成パラメーターを計画します。
 - ローカル・バス・メンバーまたはリモート・バス・メンバーの場所。
 - データベースの名前。デフォルトは BPEME です。
 - スキーマ名。デフォルトは MEDBPM00 です。
6. ファイル・ストアまたは組み込み Derby JDBC プロバイダーを使用している場合は、メッセージ・ストアが自動的に作成されます。
7. ファイル・ストアまたは組み込み Derby JDBC プロバイダーを使用していない場合は、次の構成パラメーターを計画します。
 - a. Business Process Choreographer の開始前にデータベースを作成しておくように計画します。
 - b. データベース・サーバーのホスト名または IP アドレス、およびこのデータベース・サーバーが使用するポート番号。
 - c. データベースへの接続とスキーマの作成に使用するユーザー名。これは、34 ページの表 8 で計画したユーザー ID です。

タスクの結果

これで、Business Process Choreographer メッセージング・エンジンのデータベースの計画が完了しました。

Business Flow Manager および Human Task Manager の計画

Business Process Choreographer 構成の中核をなすのは、Business Flow Manager と Human Task Manager です。これらの構成パラメーターについて計画を立てる必要があります。

手順

1. Business Flow Manager のメッセージ・ドリブン Bean の Run-As ユーザー ID として使用される Java Message Service (JMS) プロバイダーのユーザー ID を確認してください。管理コンソールおよび 35 ページの表 9 では、「**JMS API 認証ユーザー**」と呼ばれています。
2. Human Task Manager のメッセージ・ドリブン Bean の Run-As ユーザー ID として使用される Java Message Service (JMS) プロバイダーのユーザー ID を確認してください。管理コンソールおよび 35 ページの表 9 では、「**エスカレーション・ユーザーの認証ユーザー**」と呼ばれています。
3. 管理者およびモニターのセキュリティー・ロールのマップ先のグループまたはユーザー ID を確認してください。詳しくは、36 ページの表 10を参照してください。
4. Human Task Manager でエスカレーション・イベントの E メール通知を送信する場合、Simple Mail Transfer Protocol (SMTP) E メール・サーバーが配置されている場所のホスト名または IP アドレスを識別します。E メール通知の送信者アドレスについて計画を立てます。E メール・サービスで認証が必要な場合、サービスへの接続に使用するユーザー ID およびパスワードを確認してください。
5. API の Web サービス・バインディングのコンテキスト・ルートを決定します。
 - サーバーでの構成の場合:
 - Business Flow Manager のデフォルトは `/BFMIF_nodeName_serverName` です。
 - Human Task Manager のデフォルトは `/HTMIF_nodeName_serverName` です。
 - クラスターでの構成の場合:
 - Business Flow Manager のデフォルトは `/BFMIF_clusterName` です。
 - Human Task Manager のデフォルトは `/HTMIF_clusterName` です。
6. Business Process Choreographer Explorer、Business Space、または Representational State Transfer (REST) API あるいは JAX Web サービス API を使用するクライアントを使用する場合は、REST API および JAX Web Services API のコンテキスト・ルートを決定してください。
 - Business Flow Manager のデフォルトは `/rest/bpm/bfm` および `/BFMJAXWSAPI` です。
 - Human Task Manager のデフォルトは `/rest/bpm/htm` および `/HTMJAXWSAPI` です。

- サーバー、単一クラスター、または異なる Web サーバーにマップされた複数のクラスターに構成されている場合は、デフォルト値を使用できます。
 - Network Deployment 環境内で、同じ Web サーバーにマップされた複数のデプロイメント・ターゲットに構成されている場合は、デフォルト値を使用しないでください。各 Business Process Choreographer 構成のコンテキスト・ルートは、ホスト名とポートの組み合わせごとに固有のものでなければなりません。これらの値は Business Process Choreographer の構成後に、管理コンソールを使用して手動で設定する必要があります。
7. 初期設定で Business Flow Manager または Human Task Manager (あるいはこの両方) の監査ロギングを使用可能にするかどうかを決定します。
 8. Business Process Choreographer Explorer レポート作成機能を使用する場合は、Common Event Infrastructure ロギング・イベントを生成するように Business Flow Manager を初期設定するかどうかを決定します。

タスクの結果

これで、Business Flow Manager および Human Task Manager のすべての初期構成パラメーターの計画が完了しました。これらの設定は、管理コンソールを使用して後でいつでも変更することができます。

担当者ディレクトリー・プロバイダーの計画

Business Process Choreographer の担当者ディレクトリー・プロバイダー、担当者の代替、virtual member manager、および Lightweight Directory Access Protocol (LDAP) の設定を計画します。

手順

1. ヒューマン・タスクを使用する場合は、使用する担当者ディレクトリー・プロバイダーを決定します。

Virtual Member Manager (VMM) 担当者ディレクトリー・プロバイダー

VMM 担当者ディレクトリー・プロバイダーは、すぐに利用できるフェデレーテッド・リポジトリー (virtual member manager と呼ばれる) であり、ファイル・リポジトリーを使用して WebSphere セキュリティー向けに事前構成されています。フェデレーテッド・リポジトリーと別のユーザー・リポジトリーを組み合わせる場合は、フェデレーテッド・リポジトリーを再構成する必要があります。VMM 担当者ディレクトリー・プロバイダーでは、代替を含む Business Process Choreographer の担当者割り当て機能がすべてサポートされています。これは、フェデレーテッド・リポジトリーで提供される機能 (LDAP、データベース、ファイル・ベース、およびプロパティー拡張リポジトリーなどの各種リポジトリー・タイプのサポートなど) を使用します。

VMM 担当者ディレクトリー・プロバイダーを使用するには、WebSphere Application Server セキュリティーに対してフェデレーテッド・リポジトリーを構成する必要があります。フェデレーテッド・リポジトリーは、ファイル、LDAP、またはデータベースに基づいて、1 つ以上のユーザー・リポジトリーに関連付けることができます。これについて詳しくは、『フェデレーテッド・リポジトリー構成におけるレルムの管理』を

参照してください。フェデレーテッド・リポジトリの使用法についての詳細は、『IBM® WebSphere Developer Technical Journal』を参照してください。

Lightweight Directory Access Protocol (LDAP) 担当者ディレクトリー・プロバイダー この担当者ディレクトリー・プロバイダーは、使用前に構成しておく必要があります。ステップ 2 で計画を行います。

システム担当者ディレクトリー・プロバイダー

この担当者ディレクトリー・プロバイダーは、構成せずに使用できます。実動システムではこのプロバイダーを使用しないでください。これはアプリケーション開発テストでのみ使用するプロバイダーとして提供されています。

ユーザー・レジストリーの担当者ディレクトリー・プロバイダー

この担当者ディレクトリー・プロバイダーは、構成せずに使用できます。WebSphere セキュリティー・レルム定義に応じて、ユーザー・レジストリーは以下のリポジトリのいずれかを使用します。

- フェデレーテッド・リポジトリ – 以下を使用できます。
 - ファイル・レジストリー
 - 1 つ以上の LDAP
 - 1 つ以上のデータベース
- スタンドアロン LDAP
- スタンドアロン・カスタム
- ローカル・オペレーティング・システム

2. Lightweight Directory Access Protocol (LDAP) を使用する場合は、以下を計画します。

- a. 場合によっては、使用する LDAPTransformation.xml ファイルをカスタマイズする必要があります。ファイルの場所と、カスタマイズする必要があるプロパティーのリストについては、LDAP 担当者ディレクトリー・プロバイダーの構成を参照してください。
- b. 以下の LDAP カスタム・プロパティーを計画します。

LDAP プラグイン・プロパティー	必須またはオプション	説明
AuthenticationAlias	オプション	mycomputer/My LDAP Alias など、LDAP への接続に使用される認証別名。「セキュリティ」→「セキュア管理、アプリケーション、およびインフラストラクチャー (Secure administration, applications, and infrastructure)」→「Java 認証・承認サービス (JAAS)」→「J2C 認証データ」をクリックして、管理コンソールでこの別名を定義する必要があります。この別名が設定されていないか、または AuthenticationType が simple に設定されていない場合、LDAP サーバーへの匿名ログオンが使用されます。
AuthenticationType	オプション	このプロパティーを simple に設定する場合は、AuthenticationAlias パラメーターを指定する必要があります。そうでない場合は、匿名認証が使用されます。

LDAP プラグイン・プロパティ	必須またはオプション	説明
BaseDN	必要	例えば o=mycompany, c=us など、すべての LDAP 検索操作の基本識別名 (DN)。ディレクトリー・ルートを指定するには、単一引用符を 2 個使用して空ストリング '' を指定します。
Casesentiveness ForObjectclasses	オプション	LDAP オブジェクト・クラスの名前がケース・センシティブかどうかを決定します。
ContextFactory	必要	例えば com.sun.jndi.ldap.LdapCtxFactory など、Java Naming and Directory Interface (JNDI) コンテキスト・ファクトリーを設定します。
ProviderURL	必要	この Web アドレスは、LDAP JNDI ディレクトリー・サーバーおよびポートを指す必要があります。フォーマットは、例えば ldap://localhost:389 など、通常の JNDI 構文である必要があります。SSL 接続の場合は LDAP の URL を使用します。 高可用性構成で、ミラーリングされたデータを 2 つ以上の LDAP サーバーが維持する場合、各 LDAP サーバーの URL を指定し、スペース文字を使用して URL を分離することを計画します。
SearchScope	必要	すべての検索操作のデフォルト検索スコープ。baseDN プロパティの下での検索の深さを決定します。objectScope、onelevelScope、または subtreeScope のいずれかの値を指定します。
additionalParameter Name1-5 および additionalParameter Value1-5	オプション	これらの名前と値のペアを使用し、最大 5 つの任意の接続用 JNDI プロパティを LDAP サーバーに対してセットアップします。

3. virtual member manager を使用する場合は、以下を計画します。
 - a. 場合によっては、使用する VMMTransformation.xml ファイルをカスタマイズする必要があります。ファイルの場所と、カスタマイズする必要があるプロパティのリストについては、Virtual Member Manager 担当者ディレクトリー・プロバイダーの構成を参照してください。
4. 担当者の代替を使用する場合は、以下の点を考慮に入れてください。
 - VMM 担当者ディレクトリー・プロバイダーを使用する必要があります。LDAP、システム、およびユーザー・レジストリーの担当者ディレクトリー・プロバイダーは、担当者の代替をサポートしていません。
 - 実稼働環境で担当者の代替を使用する場合、代替情報の格納先として VMM プロパティ拡張リポジトリーを使用するように計画します。プロパティ拡張リポジトリーはセル全体で固有であり、セル内からアクセス可能である必要があります。暗黙的には、選択されたデータベースも同様です。BPEDB データベースは必ずしもセル内で固有ではないため、BPEDB は使用できません。共通データベース WPSRCDB を使用してプロパティ拡張リポジトリーをホストできます。ただし実稼働環境では、他の WebSphere Process Server データベースから独立したデータベースを使用することをお勧めします。
 - 単一サーバーのテスト環境で担当者の代替を使用する場合は、フェデレーテッド・リポジトリー用に構成した内部ファイル・レジストリーに担当者代替情報を格納できます。

タスクの結果

これで、担当者ディレクトリー・プロバイダーと担当者割り当てオプションの計画が完了しました。

Business Process Choreographer Explorer の計画

Business Process Choreographer Explorer の構成オプションおよび構成パラメーターを計画します。

このタスクについて

Business Process Choreographer Explorer を使用する場合は、その構成を Business Process Choreographer の構成と同時に行うことも、後で行うこともできます。Business Process Choreographer Explorer レポート作成機能はオプションです。

手順

1. 構成する Business Process Choreographer Explorer インスタンスの数を決定します。最初のインスタンスは Business Process Choreographer の構成中に簡単に作成できます。以下の理由および考慮事項が考えられます。
 - Business Process Choreographer Explorer の各インスタンスが接続できる Business Process Choreographer 構成は 1 つだけであるため、ご使用の環境に複数の Business Process Choreographer 構成がある場合、それぞれの構成に対して Business Process Choreographer Explorer インスタンスをセットアップすることが妥当です。
 - 異なる複数のカスタマイズ・バージョンの Business Process Choreographer Explorer を同じ Business Process Choreographer 構成に接続することが必要な場合もあります。それぞれのバージョンを独立してカスタマイズすることができます。カスタマイズできる内容について詳しくは、Business Process Choreographer Explorer のカスタマイズを参照してください。
 - 各サーバーまたはクラスター上に複数の Business Process Choreographer Explorer インスタンスを構成できます。
 - Business Process Choreographer または Business Process Choreographer Event Collector の構成が存在する場所に関係なく、インスタンスはどのデプロイメント・ターゲットにも作成できます。
 - それぞれの Business Process Choreographer Explorer インスタンスのレポート作成機能が接続できる Business Process Choreographer Event Collector は 1 つだけであるため、レポート作成機能を持つ Business Process Choreographer Explorer インスタンスの数が、Business Process Choreographer Event Collector と同じになるように構成することを計画します。
2. 必要な Business Process Choreographer Explorer インスタンスごとに、以下を計画します。
 - a. Business Process Choreographer Explorer のコンテキスト・ルート。これはセル内で固有でなければなりません。デフォルトは /bpc です。
 - b. エスカレーション E メールに挿入される Business Process Choreographer Explorer の URL。
 - c. Business Flow Manager と Human Task Manager の Representational State Transfer (REST) API エンドポイントの URL。REST API で計画したコンテ

キスト・ルートの値と一致する必要があります。例えば、Human Task Manager Web サービスのコンテキスト・ルートが /rest/bpm/htm である場合、Human Task Manager REST API エンドポイントのエンドポイント URL は、http://hostname:port/rest/bpm/htm になります。

- d. 照会に対して返される結果の最大数。デフォルトは 10000 です。
- e. この Business Process Choreographer Explorer が管理する Business Process Choreographer インスタンスのデプロイメント・ターゲット (サーバーまたはクラスター)。
- f. オプション: Business Process Choreographer Explorer レポート作成機能を使用する場合は、『Business Process Choreographer Explorer レポート作成機能の計画』を実行します。計画と構成は後で行うこともできます。

タスクの結果

これで、Business Process Choreographer Explorer の構成オプションの計画が完了しました。

Business Process Choreographer Explorer レポート作成機能の計画:

Business Process Choreographer Explorer レポート作成機能と Event Collector の構成を計画します。

このタスクについて

Business Process Choreographer Explorer レポート作成機能を使用する場合は、その構成を Business Process Choreographer Explorer の構成時に行うことも、後で行うこともできます。

手順

1. セキュリティー・ロールは、Business Process Choreographer Explorer レポート作成機能へのアクセスを制限するためには使用されないため、Business Process Choreographer Explorer のすべてのユーザーがレポート作成機能にアクセスすることを望まない場合は、レポート作成機能用に別の Business Process Choreographer Explorer インスタンスを構成して、通常のユーザーからアクセス不能にするように計画します。
2. Business Process Choreographer Explorer レポート作成機能の各種トポロジー・エレメントの目的とエレメント間の関係を理解します。

Business Process Choreographer Explorer レポート作成機能

バージョン 6.2 より前は、この機能は Business Process Choreographer Observer として利用可能でした。バージョン 6.2 以降、この機能は Business Process Choreographer Explorer に統合され、「レポート」タブで使用できるようになりました。この機能を使用するには、まず Business Process Choreographer Explorer レポート作成機能を構成する必要があります。

Event Collector アプリケーション

このアプリケーションは、Common Event Infrastructure (CEI) サーバーが構成されているサーバーまたはクラスター上にデプロイする必要があります。各 CEI デプロイメント・ターゲットについて、Event Collector

を 1 つだけ持つことができます。Business Process Choreographer が構成されているターゲット上にデプロイする必要はありません。CEI からビジネス・プロセス・イベントを受信して変換し、変換したイベントをレポート・データベースに書き込みます。

レポート・データベース

Event Collector と Business Process Choreographer Explorer レポート作成機能は、同じデータベースを使用して通信します。非実動システムの場合、データベースを他のコンポーネントと共有できます。

セットアップされている Business Process Choreographer のトポロジーとは関係なく選択することができます。可能な構成については、63 ページの『Business Process Choreographer Explorer レポート作成機能の概要』を参照してください。

3. セットアップの目的、システムの要件、およびトポロジーの影響を確認します。

簡易セットアップ

構成と管理が単純であるが、パフォーマンスが低いセットアップでは、Business Process Choreographer Explorer および CEI が構成されているデプロイメント・ターゲットに Event Collector アプリケーションをデプロイし、ローカル・データベース・システムを使用します。

負荷が高い実動システム: Network Deployment

複数ノードで構成されるセルと複数のクラスターを使用します。Business Process Choreographer Explorer のインスタンスを、セル内の任意のデプロイメント・ターゲットにインストールします。Event Collector アプリケーションを、Common Event Infrastructure (CEI) を構成したクラスターにインストールします。個別のデータベース・サーバーを使用します。

4. Business Process Choreographer Explorer レポート作成機能のデータベースについてまだ計画していない場合は、45 ページの『レポート・データベースの計画』を行います。
5. 構成する Event Collector インスタンスごとに、以下を計画します。
 - a. インスタンスをインストールする場所を決定します。デプロイメント・ターゲットごとに Event Collector インスタンスを 1 つのみインストールできます。また、デプロイメント・ターゲットでは CEI が構成されている必要があります。
 - b. この Event Collector インスタンスの構成方法を決定します。
 - 管理コンソールのページを使用する。このオプションについては、管理コンソールを使用した Business Process Choreographer Event Collector の構成を参照してください。
 - 対話式 setupEventCollector ツールを使用する。このオプションについては、setupEventCollector ツールを使用した Business Process Choreographer Event Collector の構成を参照してください。
 - Business Process Choreographer 構成の作成時に、bpeconfig.jacl スクリプトを使用する。-createEventCollector オプションにはデフォルト値 yes が設定されています。

注: ハイパフォーマンス・システムについては、`bpeconfig.jacl` を使用して Business Process Choreographer Explorer レポート作成機能 を構成しないでください。この理由は、`bpeconfig.jacl` は Event Collector および Business Process Choreographer Explorer レポート作成機能 アプリケーションを、Business Process Choreographer 構成と同じデプロイメント・ターゲットに構成するためです。このオプションについて詳しくは、`bpeconfig.jacl` スクリプトによる Business Process Choreographer の構成を参照してください。

Event Collector を対話モードで構成する場合は、`bpeconfig.jacl` は使用できません。

- c. データ・ソースを計画します。
 - Business Process Choreographer Explorer レポート作成機能が Business Process Choreographer と同じ物理データベースを共有している場合は、レポート・データベースに別のデータ・ソースを使用することを計画し、その JNDI 名を計画します。
 - データベースに使用する認証別名を計画します。
 - 有効範囲がセルのデータ・ソースを作成するように計画します。
- d. Event Collector の構成時に必要な構成パラメーターを計画します。
 - レポート・データベースの JNDI データ・ソース名。
 - データベース・オブジェクトに使用するスキーマ。デフォルトは、データベースへの接続に使用するユーザー ID です。
 - データベースへの接続に使用するユーザー ID。デフォルトはデータベースによって異なります。DB2 の場合、デフォルトは `db2admin`、Oracle の場合、デフォルトは `system`、その他のデータベースの場合、デフォルトはログオン・ユーザーのユーザー ID です。
 - ユーザー ID のパスワード。
 - タイプ 4 JDBC 接続を使用する場合は、データベース・サーバーのホスト名または IP アドレスと、データベース・サーバーが使用するポート番号も収集します。
 - Event Collector をデプロイする場所を決定します。デプロイメント・ターゲットでは CEI が構成されている必要があります。したがって、CEI のための別のクラスターがある場合は、Event Collector をそのクラスターにデプロイするよう計画してください。
 - Network Deployment 環境に Event Collector をデプロイする場合は、CEI バスのメッセージング・エンジンが構成されているデプロイメント・ターゲットを確認しておいてください。
 - CEI バスのセキュリティーが有効な場合、CEI バスでの認証に使用する JMS ユーザー ID を計画します。
 - Event Collector の構成時に CEI イベントによるビジネス・イベントのログ記録を有効にするか、または後で管理コンソールを使用するかスクリプトを実行してログ記録を有効にするかを決定します。
- e. ランタイム構成値を計画します。場合によっては、Event Collector の構成後にニーズに合わせてランタイム構成値をカスタマイズする必要があります。
 - `BpcEventTransformerEventCount`

- BpcEventTransformerMaxWaitTime
- BpcEventTransformerToleranceTime
- ObserverCreateTables
- 認証別名ユーザー ID がデータベース・スキーマを所有しない場合は、ObserverSchemaName を計画します。

これらの値について詳しくは、Business Process Choreographer Explorer レポート作成機能の構成パラメーターの変更を参照してください。

6. 構成する Business Process Choreographer Explorer レポート作成機能ごとに、以下を計画します。

- このインスタンスの構成方法を決定します。
 - Business Process Choreographer Explorer の作成時に、Business Process Choreographer Explorer の管理コンソール・ページを使用する。このオプションについて詳しくは、管理コンソールによるBusiness Process Choreographer Explorer レポート作成機能の構成を参照してください。
 - Business Process Choreographer Explorer の作成時に、clientconfig.jacl スクリプトを使用する。
 - Business Process Choreographer 構成の作成時に、bpeconfig.jacl スクリプトを使用する。

注: ハイパフォーマンス・システムについては、bpeconfig.jacl を使用して Business Process Choreographer Explorer レポート作成機能を構成しないでください。この理由は、bpeconfig.jacl は Event Collector および Business Process Choreographer Explorer レポート作成機能 アプリケーションを、Business Process Choreographer 構成と同じデプロイメント・ターゲットに構成するためです。このオプションについて詳しくは、bpeconfig.jacl スクリプトによる Business Process Choreographer の構成を参照してください。

- レポート・データベースのスキーマ名。
 - レポート・データベースに接続するために Business Process Choreographer Explorer によって使用されるデータ・ソースの JNDI 名。
7. bpeconfig.jacl スクリプトを使用して Business Process Choreographer を構成する場合:
- スクリプトをバッチ・モードで実行する場合、デフォルトでは、Event Collector アプリケーションと Business Process Choreographer Explorer アプリケーションも、Business Process Choreographer 構成と同じデプロイメント・ターゲットで構成されます。
 - Event Collector および Business Process Choreographer Explorer レポート作成機能、あるいはそのいずれかを bpeconfig.jacl で構成したくない場合は、bpeconfig.jacl のオプション `-createEventCollector no` および `-reportFunction no`、あるいはそのいずれかを使用するように計画して、bpeconfig.jacl でこれらが構成されないようにします。

タスクの結果

これで、Business Process Choreographer Explorer レポート作成機能と Event Collector の構成オプションの計画が完了しました。

リモート・クライアント・アプリケーションの計画

Business Process Choreographer API を使用し、WebSphere Process Server クライアント・インストールで稼働するリモート Business Process Choreographer クライアント・アプリケーションを計画します。

このタスクについて

アプリケーションで Business Process Choreographer API を使用する場合は、WebSphere Process Server クライアント・インストールを使用して完全な WebSphere Process Server サーバー・インストールに対してリモート側からアプリケーションを実行できます。クライアントは、完全な WebSphere Process Server インストールよりも構成および管理が容易です。

WebSphere Process Server クライアント・インストールには WebSphere Process Server プロファイル・テンプレートは含まれていませんが、基盤となる WebSphere Application Server プロファイルを、Feature Pack for SCA バージョン 1.0 (SDO 2.1.1 適用) で拡張する必要があります。つまり、フェデレートされたプロファイルを持つ既存の WebSphere Application Server インストールの上に WebSphere Process Server クライアントをインストールでき、フェデレートされたこれらの WebSphere Application Server プロファイルでただちに WebSphere Process Server クライアント機能を利用できます。完全な WebSphere Process Server サーバーではこのシナリオは実現できません。WebSphere Process Server は、フェデレート済みプロファイルの拡張をサポートしていないためです。

手順

1. WebSphere Process Server クライアントのインストールを計画します。
 - WebSphere Portal Server から Business Process Choreographer にアクセスする場合は、互換性のある WebSphere Process Server クライアントをインストールする必要があります。

表 14. WebSphere Portal Server が Business Process Choreographer にアクセスするために使用できる WebSphere Process Server クライアントのバージョン

	WebSphere Process Server クライアントのバージョン			
WebSphere Portal Server のバージョン	6.1.0.1	6.1.2	6.2	7.0
6.1.0.1	はい	はい	いいえ	いいえ
6.1.0.2	はい	はい	はい	いいえ

- クライアント・インストールは基本プロファイルを拡張しないため、フェデレート済みプロファイルを含む既存のプロファイルは、いずれも WebSphere Process Server クライアントをただちに使用できます。
 - 既存の WebSphere Application Server インストールがない場合は、WebSphere Application Server Network Deployment インストールが作成されます。
2. 以下のどのタイプの Business Process Choreographer クライアント・アプリケーションを使用するかを決定します。
 - カスタム・クライアント・アプリケーション
 - Business Process Choreographer Explorer

注: カスタマイズした JavaServer Pages (JSP) を使用する場合は、タスクおよびプロセス・メッセージ用の JSP ページの開発で説明しているように、その場所を調べておきます。

3. Business Process Choreographer を使用するカスタム・クライアント・アプリケーションを開発する場合は、アプリケーションで使用するインターフェースを計画します。プロセスおよびタスクは、以下のいずれかを使用して処理できます。
 - Web サービス API、Java Messaging Service (JMS) API、または Representational State Transfer (REST) API – これらの API に基づくリモート・クライアント・アプリケーションは、WebSphere Process Server インストールを一切必要としません。
 - JavaServer Faces (JSF) コンポーネント
 - Enterprise JavaBeans™ (EJB) API

注: Business Process Choreographer の EJB API を使用するクライアント・アプリケーションを開発する場合は、セッション Bean のリモート・インターフェースにアクセスするで説明している方法に従ってパッケージ化する必要があります。

4. WebSphere Process Server クライアントをインストールするセルのタイプを決定するか識別します。
 - a. Business Process Choreographer を構成した管理対象サーバーまたはクラスターが存在するセルでは、リモート成果物ローダー (RAL) のデフォルト構成によってクライアントとサーバーの間で成果物の非セキュア伝送を実行できます。これを「単一セル」のシナリオと呼びます。
 - b. Business Process Choreographer が構成されている管理対象サーバーまたはクラスターを持たないセルには、さまざまなデプロイメント・マネージャーが存在します。これを「クロス・セル」のシナリオと呼びます。クライアント・アプリケーションが EJB API を使用する場合は、名前空間のバインディングを定義し、Business Process Choreographer が構成されているサーバーまたはクラスターをクライアント・アプリケーションが検出できるようにする必要があります。

タスクの結果

これで、リモート Business Process Choreographer クライアント・アプリケーションの計画が完了しました。

Business Process Choreographer の概要

Business Flow Manager と Human Task Manager によって提供される機能について説明します。

Business Process Choreographer は、WebSphere Application Server 環境でビジネス・プロセスとヒューマン・タスクの両方をサポートするエンタープライズ・ワークフロー・エンジンです。これらの構成体は、サービスのオーケストレーションと、ビジネス・プロセスの担当者が関与するアクティビティの統合に使用できます。

Business Process Choreographer は、ビジネス・プロセスのライフ・サイクルおよびヒューマン・タスクを管理し、関連したモデルをナビゲートして、該当するサービスを呼び出します。

Business Process Choreographer は、次の機能を提供します。

- ビジネス・プロセスおよびヒューマン・タスクのサポート。ビジネス・プロセスは、Web Services Business Process Execution Language (WS-BPEL、略記 BPEL) を使用してビジネス・プロセスをモデル化する標準的な方法を提供します。ヒューマン・タスクでは、Task Execution Language (TEL) を使用して、担当者が関与するアクティビティーをモデル化できます。ビジネス・プロセスおよびヒューマン・タスクは、サービス指向アーキテクチャー (SOA) または Service Component Architecture (SCA) でのサービスとして公開され、単純なデータ・オブジェクトおよびビジネス・オブジェクトもサポートします。
- ビジネス・プロセスおよびヒューマン・タスクとの対話用にカスタマイズされたアプリケーションを開発するためのアプリケーション・プログラミング・インターフェース。
- Business Process Choreographer Explorer。この Web アプリケーションで、ビジネス・プロセスとヒューマン・タスクを管理することができます。また、これにはオプションで Business Process Choreographer Explorer レポート作成機能 (これまでは Business Process Choreographer Observer と呼ばれていました) が組み込まれ、この機能によって実行中のプロセスの状態を監視することができます。
- Business Space の一部としてのヒューマン・ワークフロー・ウィジェット。これらのウィジェットにより、作業の管理、他のユーザーのためのタスクの作成、およびサービスとプロセスの開始が可能です。

Business Process Choreographer Explorer の概要

Business Process Choreographer Explorer は、ビジネス・プロセスおよびヒューマン・タスクと対話するための汎用 Web ユーザー・インターフェースを実装している Web アプリケーションです。

また、オプションのレポート作成機能が組み込まれていますが、この機能はこれまでは Business Process Choreographer Observer と呼ばれていました。

サーバーまたはクラスター上で Business Process Choreographer Explorer インスタンスを 1 つ以上構成できます。WebSphere Process Server プロファイルを持つ WebSphere Process Server のインストール済み環境、または WebSphere Process Server クライアントのインストール済み環境があれば十分です。Business Process Choreographer をサーバーまたはクラスター上に構成する必要はありません。クライアントを WebSphere Process Server に接続する必要がある唯一のインフラストラクチャーは WebSphere Process Server クライアント・インストールですが、これには Business Process Choreographer Explorer は含まれません。WebSphere Process Server クライアント・インストールの場合でも、デプロイメント・マネージャーを使用して Business Process Choreographer Explorer をサーバーにインストールしてください。

単一の Business Process Choreographer Explorer は、1 つの Business Process Choreographer 構成に接続できますが、ローカル構成に接続する必要はありません。ただし、Business Process Choreographer Explorer の複数のインスタンスを同じサーバーまたはクラスター上で構成して、各インスタンスを異なる Business Process Choreographer 構成に接続できます。

Business Process Choreographer Explorer を始動した場合、ユーザー・インターフェースに表示されるオブジェクトおよび実行できるアクションは、所属先のユーザー・グループとそのグループに与えられている権限に応じて異なります。例えば、ビジネス・プロセス管理者の場合は、デプロイされたビジネス・プロセスの処理を円滑に進める責任があります。したがって、プロセス・テンプレート、タスク・テンプレート、プロセス・インスタンス、タスク・インスタンス、およびこれらの関連オブジェクトに関する情報を表示できます。これらのオブジェクトを処理することもできます。例えば、新規プロセス・インスタンスの開始、タスクの作成および開始、失敗したアクティビティの修復および再開、作業項目の管理、完了したプロセス・インスタンスやタスク・インスタンスの削除などを実行できます。ただし、ユーザーの場合は、割り当てられているタスクについてのみ、表示や処理が可能になります。

Business Process Choreographer Explorer レポート作成機能の概要:

Business Process Choreographer Explorer レポート作成機能の概要。

Business Process Choreographer Explorer レポート作成機能を使用すると、完了しているプロセスに関するレポートを作成できます。これは、実行中のプロセスの状況を表示する場合にも使用できます。ここでは、アーキテクチャーと可能な構成パスについて説明します。

Business Process Choreographer Explorer レポート作成機能は、Common Event Infrastructure (CEI) を使用して、WebSphere Process Server が発行するイベントを収集します。いくつかの定義済みレポートを使用するか、または独自のレポートを定義して、多くのプロセス、アクティビティ、または他の集合データの概要を把握することができます。また、特定のプロセスまたはアクティビティについての情報を得ることもできます。

Business Process Choreographer Explorer レポート作成機能は、以下の図に示す 2 つの Java EE アプリケーションに基づいています。

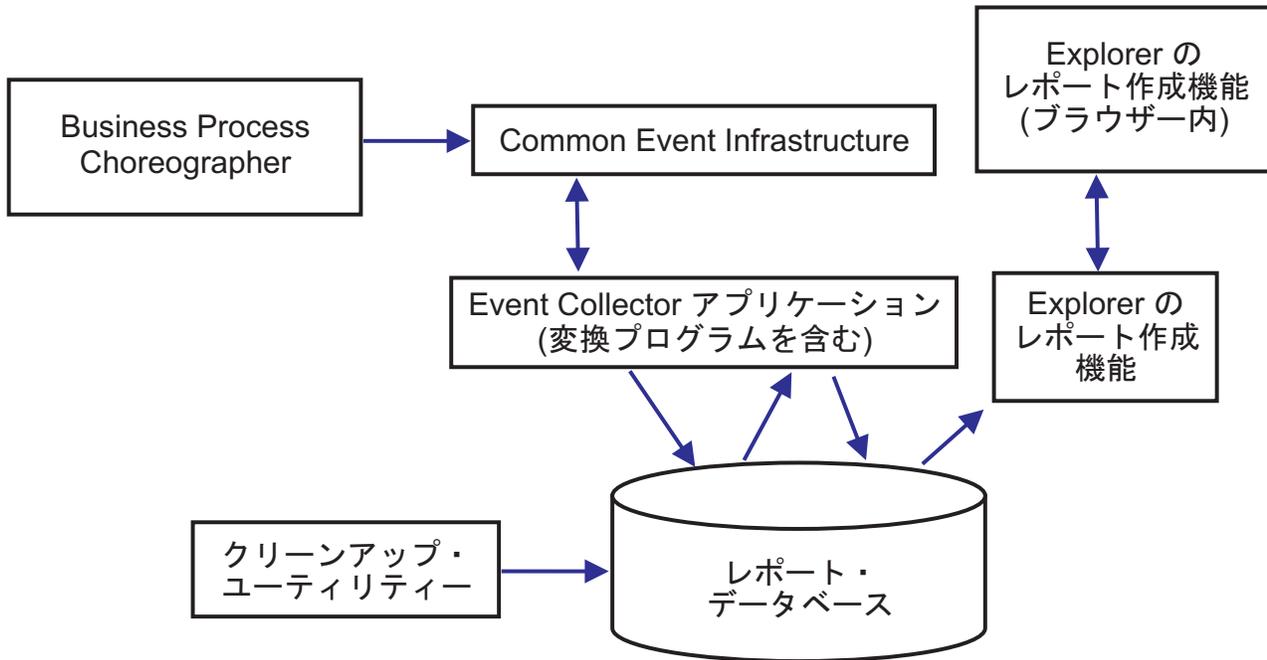


図1. アーキテクチャー

- Event Collector アプリケーションは、CEI バスからイベント情報を読み取り、レポート・データベースの Event Collector 表に保管します。
- レポート・データベースは、イベント・データを保管するデータベース表のセットです。
- イベント変換プログラムが定期的にトリガーされ、これにより未加工のイベント・データは、Business Process Choreographer Explorer レポート作成機能からの照会に適したフォーマットに変換されます。
- Business Process Choreographer Explorer レポート作成機能はレポートを生成し、ユーザーがグラフィカル・ユーザー・インターフェース (GUI) を使用して開始可能な他のアクションを実行します。
- GUI を使用して独自のレポートを生成することができます。また、自分で定義したレポートを保管および検索することも可能です。
- クリーンアップ・ユーティリティーを使用してデータベースからレコードを削除し、これによりパフォーマンスを改善することができます。

簡易構成

簡易構成では、パフォーマンスは重要な考慮事項ではありません。簡易構成を次の図に示します。

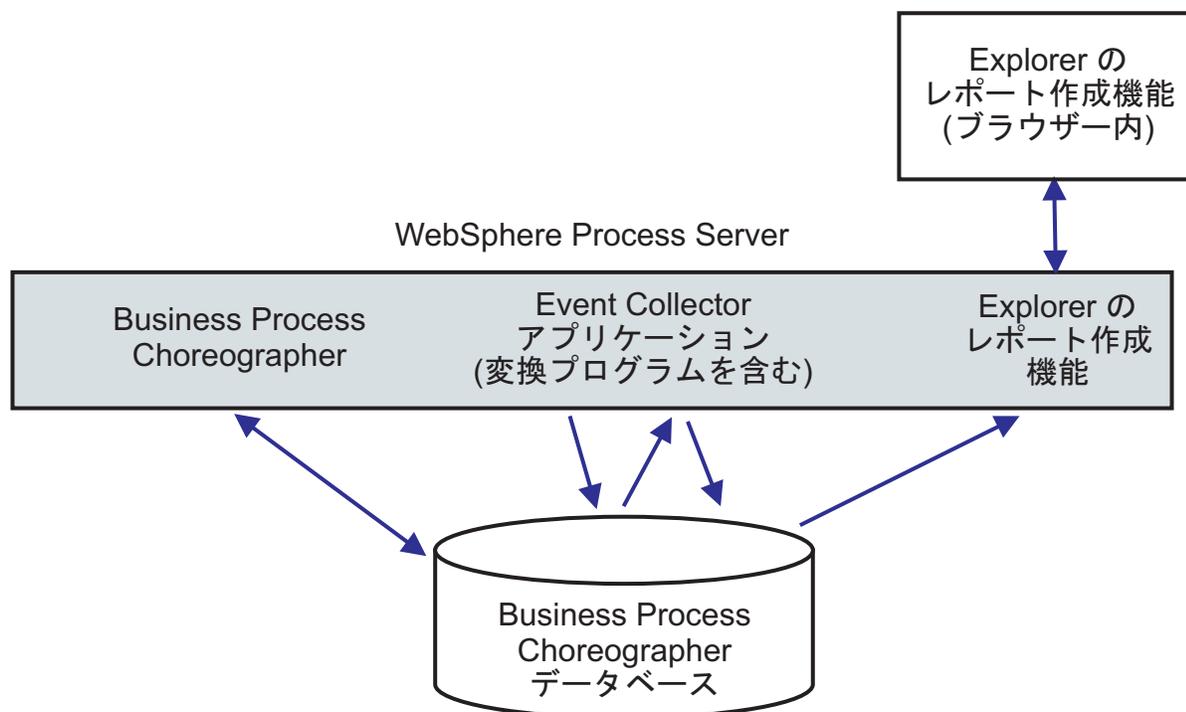


図2. スタンドアロン・セットアップ

すべての構成要素が単一システムにインストールされており、Business Process Choreographer と Business Process Choreographer Explorer レポート作成機能が同じデータベースを使用します。

サンプル Business Process Choreographer 構成を作成する場合、このような簡易構成が作成されます。また、bpeconfig.jacl ツールではデフォルトで、Business Process Choreographer 構成と同じデプロイメント・ターゲット上でこの種類のセットアップが構成されます。Common Event Infrastructure (CEI) ロギングが使用可能になり、必要なデータベース・スキーマが Business Process Choreographer の Derby データベース BPEDB に作成されます。この構成パスは、パフォーマンスが重要な考慮事項でない場合に最適です。

ハイパフォーマンス構成

提供されている対話式の構成ツールによって、Business Process Choreographer Explorer レポート作成機能のアーキテクチャの潜在能力を最大限に活用することができます。例えば、パフォーマンスを向上させる理想的な構成は、Business Process Choreographer 構成、CEI イベント・サーバー、および Business Process Choreographer Explorer (レポート作成機能を持つ) が別個のシステム上で稼働し、Business Process Choreographer および Business Process Choreographer Explorer レポート作成機能に独自のデータベースが用意されている構成です。

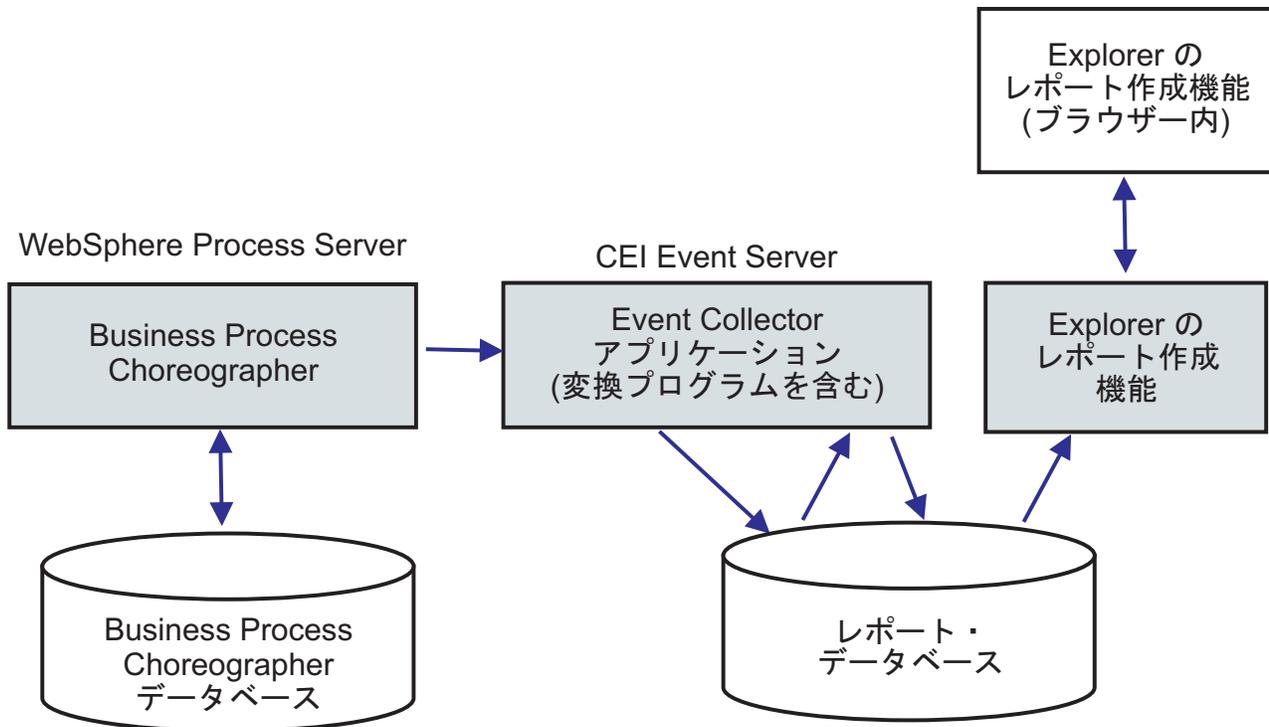


図3. 実動パフォーマンスを考慮した Business Process Choreographer Explorer レポート作成機能のセットアップ

Business Process Choreographer Explorer レポート作成機能に別個のデータベースを使用するか、Business Process Choreographer Explorer レポート作成機能を、クラスター構成のセットアップにおいて既存の Business Process Choreographer 構成に追加するか、またはより高度なデータベース・オプションを使用する場合は、Business Process Choreographer Explorer レポート作成機能および Event Collector の構成を実行します。

Network Deployment 環境の場合

以下の制約は、Business Process Choreographer Explorer レポート作成機能を Network Deployment 環境で構成する場合に適用されます。

- CEI は、使用するセル内で構成する必要があります。
- 前の図に示すとおり、Business Process Choreographer の Event Collector は、CEI イベント・サーバーが構成されているデプロイメント・ターゲットに構成する必要があります。CEI イベント・サーバーを Business Process Choreographer とは別のクラスターに構成する場合は、CEI イベント・サーバーが構成されているデプロイメント・ターゲットに Business Process Choreographer の Event Collector を構成する必要があります。Business Process Choreographer Explorer レポート作成機能アプリケーションを Event Collector と同じシステムにインストールする必要はありません。

必要なセキュリティ権限の明確化

サイトのセキュリティ・ポリシーによっては、設計を正常に実装するために、ファイルおよびフォルダーの作成、データベースへのアクセスなどのさまざまな作業を行えるユーザー ID およびパスワードが必要な場合があります。必要な権限を明確にすることで、サーバーが保護データにアクセスするときの問題を回避できます。

始める前に

- 設計を完了します。
- 使用する認証システム (例えば Lightweight Directory Access Protocol (LDAP)) を決定します。
- サイトのセキュリティ・ポリシーを確認し、適用されている制御のうち、どの制御が WebSphere Process Server のインストールに必要な権限に影響するかを判別します。
- 製品をインストールするシステムを識別します。

このタスクについて

サイトのセキュリティ・ポリシーではグローバル・セキュリティが有効になっており、ソフトウェアのインストール、データベースまたはテーブルの作成、およびデータベースへのアクセスを行うために所定の権限が必要になっています。正常に製品をインストールして運用するためには、ここに示すステップを行う必要があります。

手順

- システムにソフトウェアをインストールするために十分な権限を持つユーザー ID およびパスワードを獲得するか、セキュリティ管理者に提供します。

必ずファイルおよびフォルダーを作成するための権限を持つ ID を使用して WebSphere Process Server のインストール・ウィザードを実行してください。
- システムの日常の運用に必要なユーザー ID、パスワード、およびロールを獲得するか、またはこれらをセキュリティ管理者に提供します。これらには、以下のものがあります。
 - 管理コンソールのユーザー ID と能力を制限するロール。ロールの構成、管理、またはモニター用のユーザー ID を所有することができます。
 - システム通信の認証に使用するシステム・バスごとのユーザー ID。
 - Business Flow Manager と Human Task Manager での認証のための Business Process Choreographer Container ごとの管理およびモニター用のユーザー ID またはグループ。
 - Business Flow Manager と Human Task Manager での認証のための同期呼び出し用のユーザー ID またはグループ。
- オプション: インストール中に WebSphere がデータベースまたはデータベース表を作成するために使用するユーザー ID およびパスワードを獲得するか、データベース管理者に提供します。

注: サイト・ポリシーによっては、この権限がデータベース管理者に制限されている場合があります。その場合は、生成されたスクリプトを管理者に渡してデータベースまたはデータベース表を作成してください。

- 運用中に WebSphere がデータベース表にアクセスするために使用するユーザー ID およびパスワードを獲得するか、データベース管理者に提供します。

タスクの結果

WebSphere サーバーをセキュアな環境にインストールして運用できます。

関連情報

 セキュリティー、ユーザー ID、および許可の計画

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](#)

第 3 章 セル内での複数のプラットフォームの使用

綿密に計画することによって、分散 プラットフォーム、および z/OS オペレーティング・システムの各プラットフォームのノードが含まれたデプロイメント・マネージャー・セルを作成することができます。

例えば、z/OS ノード、Linux[®] ノード、UNIX[®] ノード、および Windows[®] ノードが含まれたデプロイメント・マネージャー・セルを作成できます。このような構成は、異機種混合 セルと呼ばれます。

異機種混合セルでは、しっかりと計画を立てることが必要です。異機種混合セルのセットアップでは、タスクの一部を自動化できないため、余計に時間がかかる場合もあります。「Heterogeneous Cells – cells with nodes on mixed operating system platforms」ホワイト・ペーパーに、異機種混合セルを作成するのに必要な計画およびシステムの考慮事項が概説されています。

管理コンソールを使用して新規サーバーを作成する場合は、サーバーの初期構成設定を提供するサーバー・テンプレート を選択します。サーバーを作成する管理対象ノードを選択した後、ユーザーは、そのノードのオペレーティング・システム・プラットフォームに使用できるテンプレートを管理コンソール上で選択できます。

重要: セルは異機種混合にすることができますが、z/OS ノードをサーバー・クラスター内の他のノードと混合することはできません。

関連概念

115 ページの『デプロイメント・マネージャー』

デプロイメント・マネージャーとは、他のサーバーの論理グループ (セル) の操作を管理するサーバーです。デプロイメント・マネージャーは、サーバーおよびクラスターを管理するための中央の場所になっています。

116 ページの『管理対象ノード』

管理対象ノード とは、デプロイメント・マネージャーにフェデレートされているノードのことです。このノードにはノード・エージェントが含まれ、さらに管理対象サーバーを含めることができます。管理対象ノードでは、管理対象サーバーを構成して実行できます。

関連情報



[Heterogeneous Cells - cells with nodes on mixed operating system platforms](#)

第 4 章 トポロジーの理解

トポロジーの選択は、ビジネスの目標と目的、さらに環境にデプロイするアプリケーションの設計および目的によって異なります。

製品をインストールおよび構成する前に、トポロジーとは何かを理解しておく必要があります。すべてのトポロジーで提供される一連の基本機能を知るとともに、各トポロジーの相違点を理解する必要があります。そのときに、コンポーネントおよびリソースの割り振り方法の違いだけでなく、どのようなシナリオにおいて特定のトポロジーが他のトポロジーよりも適しているかについても理解する必要があります。

このセクションの情報は、以下の目的のために使用します。

- トポロジーについて理解する
- トポロジーを選択するための考慮事項について理解する
- トポロジーの実装方法について理解する

関連概念

1 ページの『第 1 章 計画の概要』

WebSphere Process Server を BPM ソリューションの一部として実装する際の関連アクティビティは、計画した使用方法によって異なります。

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こと、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。

トポロジーの概要

トポロジーとは、キャパシティー、可用性、およびスケーラビリティに対するビジネス・ニーズを満たすために必要なデプロイメント環境の物理的レイアウトのことです。

トポロジーの設計方法および実装方法に影響を与える要因は、多岐にわたっています。例えば、トポロジーを選択するときや、そのトポロジーを設計して実装する方法を選択するときには、ビジネスおよびアプリケーションの要件、リソースの要件および制約、環境の所期の目的、製品をインストールして構成するオペレーティング・システムというすべての要因が、その選択に影響を与えます。トポロジーに影響を与える可能性があるビジネス要件およびアプリケーション要件の評価について詳しくは、『ソフトウェアのニーズの判別』を参照してください。

WebSphere Process Server には、以下のトポロジーのパターンが組み込まれています。これらのパターンを使用すると、概念検証 (POC) 設計から完全な機能を備えた実稼働環境に至るまでのさまざまなビジネス・シナリオに対処できます。

- 単一クラスター

- リモート・メッセージング
- リモート・メッセージングおよびリモート・サポート
- リモート・メッセージング、サポート、および Web

上記の各トポロジーは、特定のビジネス・ニーズに対処するかそのニーズを解決するための一定の設計特性を備えています。例えば、単一クラスター・トポロジーは、通常、テストや PoC のシナリオで使用します。各トポロジーの設計特性はパターンとして取り込まれ、構成テンプレートとして製品とともに提供されます。

IBM 提供のパターンを常に使用する必要はありません。実際のニーズに対処できるパターンがない場合は、カスタム・トポロジーを作成できます。

デプロイメント環境パターンの目的

デプロイメント環境パターンは、デプロイメント環境に含まれるコンポーネントおよびリソースの制約と要件を指定します。トポロジー・レイアウトごとに IBM 提供のパターンが用意されています。これらのパターンは、最も広く使用されている BPM トポロジーの特性である、コンポーネント間での対話のルールおよびガイドラインを提供します。IBM 提供のパターンは、テスト済みの既知の構成シナリオに基づいています。これを使用することで、ニーズに最も合ったデプロイメント環境を、自動化された方法で何度でも構築できます。各パターンは、関連付けられているトポロジーの構成要件およびビジネス・ニーズを満たすように設計されています。パターンを使用すると、デプロイメント環境を最も単純な方法で作成できます。

デプロイメント環境パターンは、共に動作するコンポーネント構成を含む既知のテスト済み推奨トポロジーを表しており、これらのパターンを使用すると、高品質で完全に機能するデプロイメント環境を構築できます。デプロイメント環境パターンの構成ルールを使用して、ファースト・パス構成を生成できます。これが可能であるのは、パターンには多数の設計上の決定 (例えば、どのコンポーネントを構成するか、どのデフォルト・パラメーターおよびリソースが必要か) が実装されているためです。

提供されているデプロイメント環境パターンのそれぞれが、特定の要件のまとまりに対応しています。ほとんどの要件セットは、これらのパターンのいずれかを使用することで対処できます。

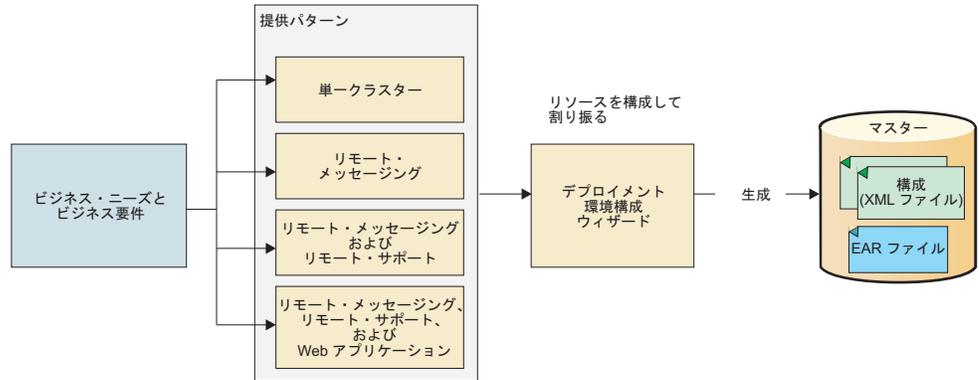
これらのパターンのいずれかに当てはまるデプロイメント環境を作成するには、インストーल、またはプロファイルの作成時、あるいは管理コンソールで選択を行います。

パターンを使用する際は、以下のことを行ってください。

- 作成するビジネス・ソリューションの要件を理解します
- 使用可能なパターンの機能および特性を確認して理解します
- 使用するパターンを決定します。

以下の図に、ビジネス・ニーズおよびビジネス要件に基づいてパターンを選択する方法と、選択したパターンに基づいてデプロイメント環境構成ウィザードで環境を生成する方法を示します。

注: デプロイメント環境構成ウィザードの使用は、デプロイメント環境を構成するための一つの方法であり、唯一の方法ではありません。(スクリプトまたは管理コンソールを使用して) クラスターを手動で構成しても、同じ結果を得ることができます。



実際のニーズに対処できる WebSphere Process Server パターンがない場合は、カスタム環境を作成できます。

IBM 提供のデプロイメント環境パターンの機能

どの WebSphere Process Server デプロイメント環境にも、基本的な一連の機能が含まれており、それらの機能が組み合わさることで完全な実稼働環境を形成しています。

堅固なデプロイメント環境を設計するためには、IBM 提供の特定のパターンまたはカスタム・デプロイメント環境で、各クラスターがどのような機能を提供できるかを理解しておく必要があります。特定のタイプの機能 (例えば、サポート・インフラストラクチャー機能) を特定のクラスターに割り振ることができます。この機能を理解しておくことは、ニーズに最も合ったデプロイメント環境パターンを正しく判別するのに役立ちます。

ネットワーク・デプロイメントの場合、クラスターは環境に特定の機能を提供するために共同することができます。お客様の要件に合わせて、デプロイメント環境内の各クラスターに特定の機能を割り当てて、パフォーマンス、フェイルオーバー、および容量を提供します。

デプロイメント環境で構成されたクラスターは、以下に示す機能を提供します。

これらの機能は、単一のクラスター内に存在している場合もあれば、複数のクラスターにまたがって分散している場合もあります。IBM 提供のトポロジー・パターンは、これらの機能をサポートするために、それぞれが異なる数のクラスターを作成します。デプロイメント環境内のクラスターの数、使用するトポロジー・パターンにより異なります。

アプリケーション・デプロイメントのターゲット

アプリケーション・デプロイメント・ターゲットとは、アプリケーション (ヒューマン・タスク、ビジネス・プロセス、メディエーションなど) のインストール先となる一連のサーバー (クラスター) です。選択したデプロイメ

ント環境パターンによっては、アプリケーション・デプロイメント・ターゲットによって、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の機能が提供される場合があります。デプロイするアプリケーションのタイプに基づいて適切な製品を選択します。

- アプリケーションにヒューマン・タスクまたはビジネス・プロセスの成果物が含まれる場合、WebSphere Process Server をインストールします。
- アプリケーションにメディエーション・モジュールのみが含まれる場合、WebSphere Enterprise Service Bus をインストールします。

単一のクラスター・パターンにおいて、アプリケーション・デプロイメントのターゲットはデプロイメント環境の機能全体を提供します。

サポート・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には、環境をサポートしてシステムを管理するために使用される Common Event Infrastructure (CEI) サーバーやその他のインフラストラクチャー・サービスが含まれます。各種インフラストラクチャー・サービスには、以下のものがあります。

- ビジネス・ルール
- セレクター
- ヒューマン・タスク
- ビジネス・プロセス

重要: このノードに対して、アプリケーション・デプロイメント・ターゲット・クラスターに対して使用した製品機能と同じ製品機能のカスタム・プロファイルを使用する必要があります。

実際のビジネス・ルール自体は、サポート・インフラストラクチャー・クラスターには結合されていないことに注意してください。実際、ビジネス・ルールは、セル内のすべての場所で存在し、機能することができます。サポート・インフラストラクチャー・クラスター (3 クラスター構成の場合)、または Web アプリケーション・インフラストラクチャー・クラスター (4 クラスター構成の場合) にデプロイできるのは、(ビジネス・ルール・マネージャーから実行される) ビジネス・ルール管理機能です。同じ原理がヒューマン・タスクとビジネス・プロセスに適用されます。ヒューマン・タスクとビジネス・プロセスは、実際にはアプリケーション・デプロイメント・ターゲット・クラスターで実行されます (このクラスターでヒューマン・タスク・コンテナとビジネス・プロセス・コンテナが構成されるため)。ただし、プロセスとタスクは、Business Process Choreographer Explorer から管理します。Business Process Choreographer Explorer は、3 クラスター構成の場合はサポート・インフラストラクチャー・クラスター上、4 クラスター構成の場合は Web アプリケーション・インフラストラクチャー・クラスター上に配置できます。

メッセージング・インフラストラクチャー

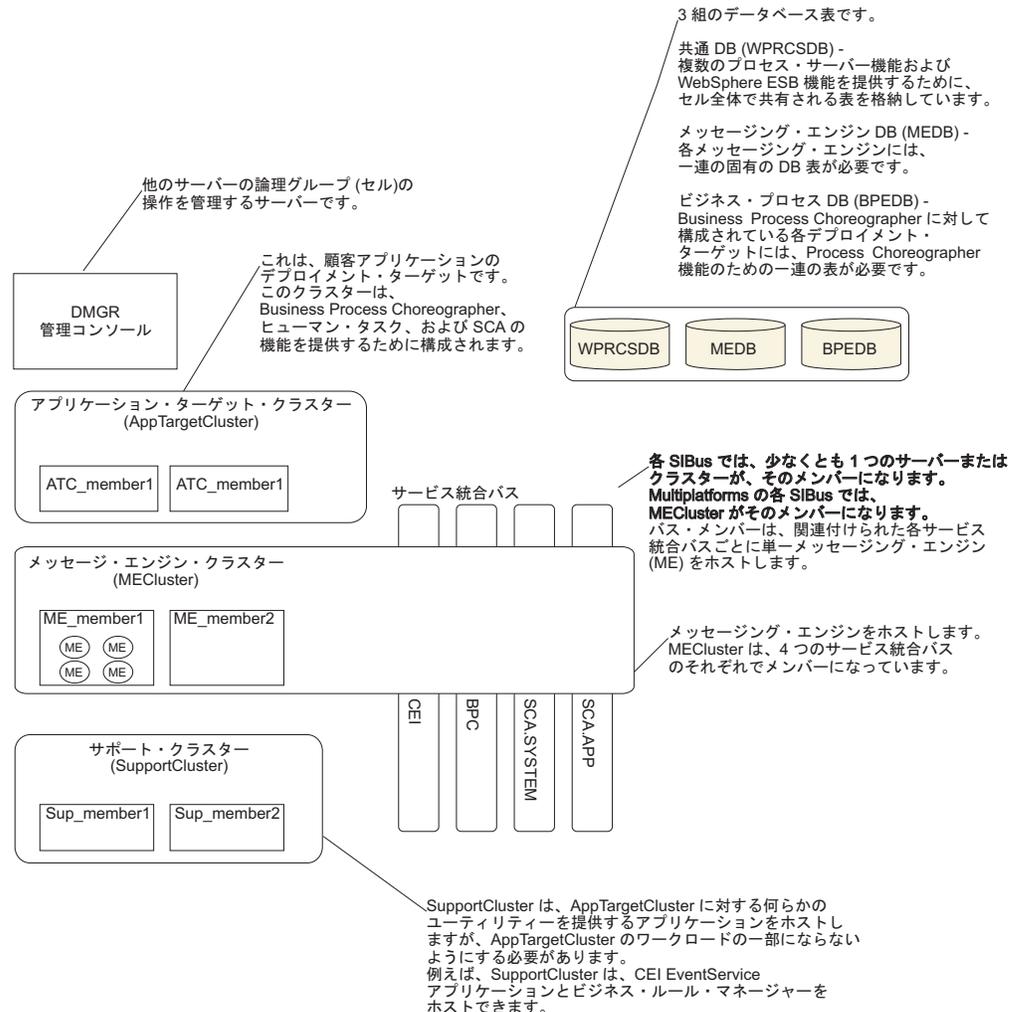
メッセージング・インフラストラクチャーは、メッセージング・エンジンが配置されている一連のサーバー (クラスター) で、WebSphere Process Server コンポーネントの内部メッセージング・ニーズに対して、およびご使用のアプリケーションに対して、非同期メッセージング・サポートを提供するために使用されます。メッセージング・エンジンによって、デプロイメント環境内のノード間の通信が可能になります。クラスターがメッセージング機能の

みを提供する場合は、このクラスターは WebSphere Process Server ではなく WebSphere Application Server で作成されたノード上のメンバーでも構成することができます。

Web アプリケーション・インフラストラクチャー

Web ベース・コンポーネントの Business Process Choreographer Explorer、Business Space およびビジネス・ルール・マネージャーが配置されている 1 つのクラスターで構成されます。

すべての環境内にあるトポロジーに対して、WebSphere Process Server および WebSphere ESB の基本的な部分は、常にほぼ同等です。どの WebSphere Process Server セルおよび WebSphere ESB セルでも、セル管理はデプロイメント・マネージャーが中心となって行います。以下の図に、リモート・メッセージングおよびリモート・サポート・トポロジー・パターンを使用して構成したデプロイメント環境の「注目点」を示します。クラスターの数と、クラスターによってサポートされている機能のタイプは、パターンごとに異なります。特定のトポロジー・パターンについては、そのパターンに適用されるトピックを参照してください。



カスタム・デプロイメント環境の機能

カスタム・デプロイメント環境では、更に多様なトポロジーが可能になります。アプリケーションでさらに処理能力を必要とするか、さらに多数のクラスターに対してサポート・インフラストラクチャー機能を拡大する必要があるか、いくつかのサーバーまたはクラスター用のサポート・インフラストラクチャーを 1 つのクラスターに統合する必要がある場合は、カスタム・デプロイメント環境を使用してこれらを実現することができます。

クラスター間で機能を分けるには、**コラボレーション単位** を使用します。コラボレーション単位により、実際のニーズに基づいて、1 つの単位として連携して動作する複数のクラスターやサーバーに機能を分散して、独立性、機能統合、スループット能力、およびフェイルオーバー機能を増強させることができます。

管理コンソールでは、コラボレーション単位を以下のようにグループ化します。

メッセージング

メッセージング単位では、IBM 提供のデプロイメント環境パターン用のメッセージング・インフラストラクチャーと同じサポートが提供されます。ローカルのメッセージング・エンジンが含まれたサーバーがクラスター内にあり、この単位内のその他のサーバーとクラスターは、そのメッセージング・エンジンをメッセージの宛先として使用します。

Common Event Infrastructure

Common Event Infrastructure 単位は、CEI サーバーをホスティングするサーバー、および CEI 機能をサポートする他のクラスターとサーバーから構成されます。単位内の各クラスターまたはサーバーで受信された Common Base Event は、CEI サーバーをホスティングするサーバーに送信されます。ご使用のデプロイメント環境で、さまざまなイベント・ソースからのイベントを分離するために、より多くの CEI サーバーをホスティングするのに必要な分だけのコラボレーション単位を使用します。

アプリケーション・サポート

アプリケーション・サポート単位は、IBM 提供のデプロイメント環境パターン用のサポート・インフラストラクチャーと似ています。これらの単位は、アプリケーションをデプロイするクラスターとサーバーをグループ化します。これらの単位は、コラボレーション単位をさらに定義することによって、複数のビジネス・コンテナまたは Service Component Architecture (SCA) サポート・クラスターをデプロイメント環境内に定義できるという点が異なります。1 つの単位により、その単位内の同一または異なるクラスターに 1 つのビジネス・プロセス・クラスター、および 1 つ以上の SCA サポート・クラスターとサポート・アプリケーションが定義されます。

関連概念

92 ページの『トポロジーを選択するための考慮事項』

使用するデプロイメント環境に適したトポロジーの選択は、いくつかの要因に基づいて行います。

114 ページの『Network Deployment』

Network Deployment という用語は、単一のデプロイメント・マネージャーによって管理されている、1 つ以上のマシンに存在するサーバーの論理グループから成る、WebSphere Process Server 環境構成のことを指します。

エラー防止とリカバリーの概要

エラー防止とリカバリーの情報では、システム障害を引き起こす問題を回避する方法について説明し、通常の状態と異常な状態の両方で発生する可能性があるシステム障害からリカバリーする方法について、情報を提供しています。

単一クラスター・トポロジー

提供されている BPM トポロジー・パターンの 1 つです。単一クラスター・トポロジーでは、WebSphere Process Server 環境のすべての機能が単一のクラスターにまとめられています。

これは、WebSphere Process Server for z/OS のデフォルト・パターンです。

単一クラスター・トポロジーは、ハードウェアが限られている場合に理想的です。すべてのコンポーネントが同じクラスターにインストールされるため、必要な物理マシンの数が少なく済みます。ただし、サポート・アプリケーションおよび統合アプリケーションを各サーバー・インスタンスで実行する必要があるため、個々の Java 仮想マシン (JVM) のメモリー所要量はかなり多くなります。さらに、クラスターの 1 つ以上のメンバーで、非同期対話に必要なメッセージング・エンジンも実行する必要があります。このため、単一クラスター・トポロジーは、通常、PoC (概念検証) 環境、開発環境、およびテスト環境の場合に使用されます。

z/OS 以外のプラットフォームでは、単一クラスター・トポロジーは一般にテスト、PoC (概念検証)、およびデモンストレーション環境に使用されます。しかし、z/OS では、単一のクラスター・トポロジーは、実行可能な実動トポロジーです。これは、WebSphere Process Server for z/OS のアーキテクチャーが異なるためです。z/OS での単一クラスター・トポロジーには、非 z/OS リモート・メッセージング・トポロジーと類似した特性があります。z/OS で単一クラスター・トポロジーを使用すると、zWLM 管理対象サーバント領域などの運用上の利点があります。

WebSphere Process Server 環境のすべての側面を単一クラスターに結合する場合には、メモリー所要量が増加する点に加えて、いくつかの点を考慮する必要があります。非同期対話 (JMS および MQ/JMS バインディングを含む)、ヒューマン・タスク、ステート・マシン、および長期実行ビジネス・プロセスは、メッセージング・インフラストラクチャーを頻繁に使用することがあるため、これらのコンポーネントを使用するアプリケーションには、単一クラスター環境は適していません。このパターンでは、メッセージング要件は最小に抑える必要があります (z/OS の場合は除く)。Service Component Architecture (SCA) の内部非同期呼び出し、Java Message Service (JMS) と MQ のメッセージングのバインディングは、同じクラスターで複数のメッセージング・エンジンをサポートしません。これらのいずれかがモジュールに必要な場合は、他のいずれかのパターン (メッセージング・インフラストラク

チャーターがアプリケーション・デプロイメント・ターゲットとは別のクラスターに存在する) を選択してください。このため、単一クラスター・パターンは、アプリケーションの実行および同期呼び出しに焦点を当てたシナリオに適しています。このトポロジは、Common Event Infrastructure (CEI) を頻繁に使用することを計画している場合にも適していません。CEI 関連のメッセージング・トラフィックおよびイベントを生成すると、クラスター・メンバーの負担が増します。

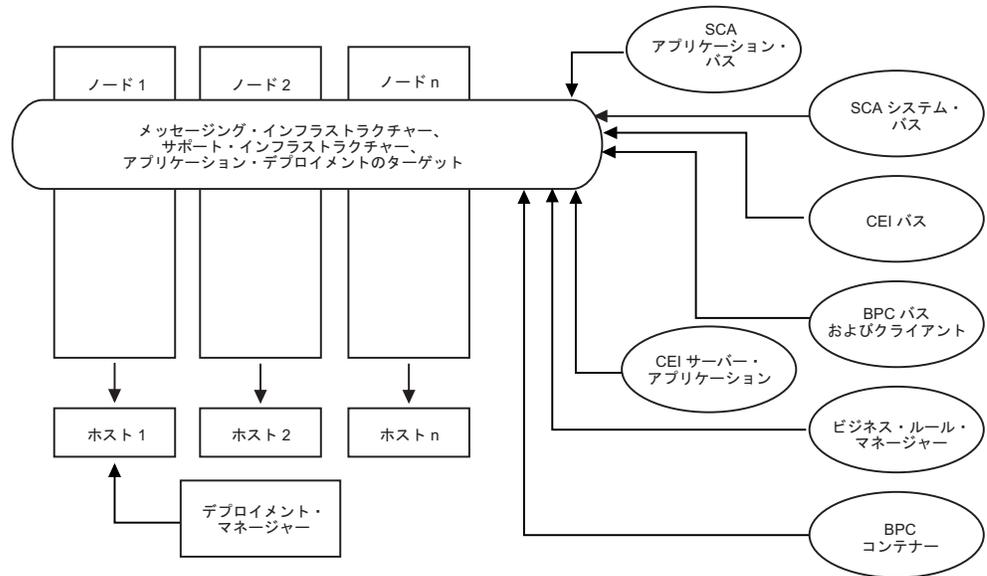
管理上の観点とスケーラビリティの観点からは、単一クラスター・トポロジが他のトポロジよりも優れています。各メンバーがすべての WebSphere Process Server コンポーネントを実行する単一クラスターは、管理するのが容易です。複数のクラスター内にいくつかのサーバー・インスタンスがある代わりに、少数のメンバーが単一のクラスターにまとめられています。環境のニーズが増大した場合でも、ノードおよびクラスター・メンバーを追加するだけで、インフラストラクチャーを拡張できます。このため、能力の増強は容易に行えますが、すべてのコンポーネントが同じ比率で拡張されます。例えば、クラスター・メンバーを追加するたびに、CEI 処理が必要かどうかに関係なく、CEI 処理が追加されます。ポリシーを使用して複数のサーバー・メンバーにわたってメッセージング・エンジンを分散した場合は、ポリシーの作成および保守を行うために、管理上の手間がある程度増すことがあります。

単一クラスター・パターンでは、以下に示すすべてのデプロイメント環境機能/コンポーネントが単一クラスター上で実行されます。

- Service Component Architecture (SCA) アプリケーション・バス・メンバー
- SCA システム・バス・メンバー
- Business Process Choreographer バス・メンバー
- Explorer などの Business Process Choreographer の各コンポーネント
- Business Process Choreographer Container
- Common Event Interface (CEI) バス・メンバー
- CEI サーバー
- ビジネス・ルール・マネージャー
- アプリケーション・デプロイメントのターゲット

アプリケーション・デプロイメント・ターゲットを構成して、SCA アプリケーションと Business Process Choreographer コンポーネントをサポートします。

以下に示す単一クラスター・トポロジのグラフィカル表現を参照してください。



関連概念

92 ページの『トポロジーを選択するための考慮事項』

使用するデプロイメント環境に適したトポロジーの選択は、いくつかの要因に基づいて行います。

同期呼び出し

サービス・コンポーネント・インターフェース (SCA) は、常に同期形式で定義されます。同期インターフェースごとに、1 つ以上の非同期インターフェースを生成できます。

リモート・メッセージング・トポロジー

提供されている BPM トポロジーの 1 つです。リモート・メッセージング・パターンでは、デプロイメント環境の機能が 2 つの独立したクラスター間で分割されます。

リモート・メッセージング・パターンでは、メッセージング機能用に独立したクラスターを使用します。このパターンは、負荷に対してクラスターを拡張できるため、非同期呼び出しを必要とするシナリオに適しています。各コンポーネントは、2 つのクラスター間で分割されます。

多数のヒューマン・タスク、長期実行ビジネス・プロセス、ステート・マシン、および非同期対話をサポートする必要がある環境では、リモート・メッセージング・トポロジーの方が単一クラスター・トポロジーよりも有利です。独立したクラスターにメッセージング・インフラストラクチャを分離することで、メッセージングのオーバーヘッドがアプリケーション・ターゲット・クラスターから取り除かれます。これによって、アプリケーション・ターゲット・クラスターのメンバーのメモリー所要量を削減できます。このトポロジーは、必要なハードウェアの観点からも単一クラスター・トポロジーとは異なっています。このトポロジーには、複数のクラスター・メンバーを持つ 2 つのクラスターがあるので、分散環境を構築するためにハードウェア要件が増しています。

管理上の観点からは、リモート・メッセージング・トポロジーの要件は、単一クラスター・トポロジーの要件よりも増えています。クラスターやクラスター・メンバーを追加すると、必要な管理上の手間が増えます。さらに、メッセージング・クラスターのメンバー全体にわたってメッセージング・エンジンを分散させるには、ポリシーの作成および保守が必要です。

サポートを行うアプリケーションおよび CEI コンポーネントは、リモート・メッセージング・トポロジーでも、アプリケーション・ターゲット・クラスターの一部となります。このため、CEI を頻繁に使用する環境には、リモート・メッセージング・トポロジーであっても対応できないことがあります。小規模から中規模のビジネスの場合や、厳しいモニター要件または監査要件が課されていないビジネスの場合は、通常、このトポロジーが適しています。

リモート・メッセージング・トポロジーのスケラビリティ・オプションは、単一クラスター・トポロジーのオプションと同じように単純です。メッセージング・エンジンが従うのは n 個のポリシーのうちのいずれか 1 つである (各メッセージング・エンジンは 1 つのサーバー上でのみアクティブになる) ため、メッセージング・クラスターにメンバーを追加しても、影響はほとんどありません。ポリシーを使用してサーバーのメンバー全体にわたってメッセージング・エンジンを分散させると、メッセージングの負担を最大で 3 つのサーバーに分割できます (SCA.SYSTEM エンジンおよび SCA.APPLICATION エンジンは、同じサーバー上でアクティブにする必要があります)。このため、3 つより多くのクラスター・メンバーをメッセージング・クラスターに追加しても、メッセージング・インフラストラクチャーの処理能力は増強されません。アプリケーション・ターゲット・クラスターの拡張は、比較的容易です。アプリケーションやサポート・インフラストラクチャーの処理能力を増強する必要がある場合は、ノードおよびメンバーをアプリケーション・ターゲット・クラスターに追加するだけで済みます。

リモート・メッセージング・クラスター:

- Service Component Architecture (SCA) アプリケーション・バス・メンバー
- SCA システム・バス・メンバー
- Business Process Choreographer (BPC) バス・メンバー
- Common Event Interface (CEI) バス・メンバー

サポート・インフラストラクチャーおよびアプリケーション・デプロイメント・ターゲット・クラスター:

- CEI サーバー・アプリケーション
- ビジネス・ルール・マネージャー
- Explorer などの Business Process Choreographer の各コンポーネント
- アプリケーション・デプロイメントのターゲット

アプリケーション・デプロイメント・ターゲットを構成して、SCA アプリケーションと Business Process Choreographer コンポーネントをサポートします。

以下に示すリモート・メッセージング・クラスター・トポロジーのグラフィカル表現を参照してください。

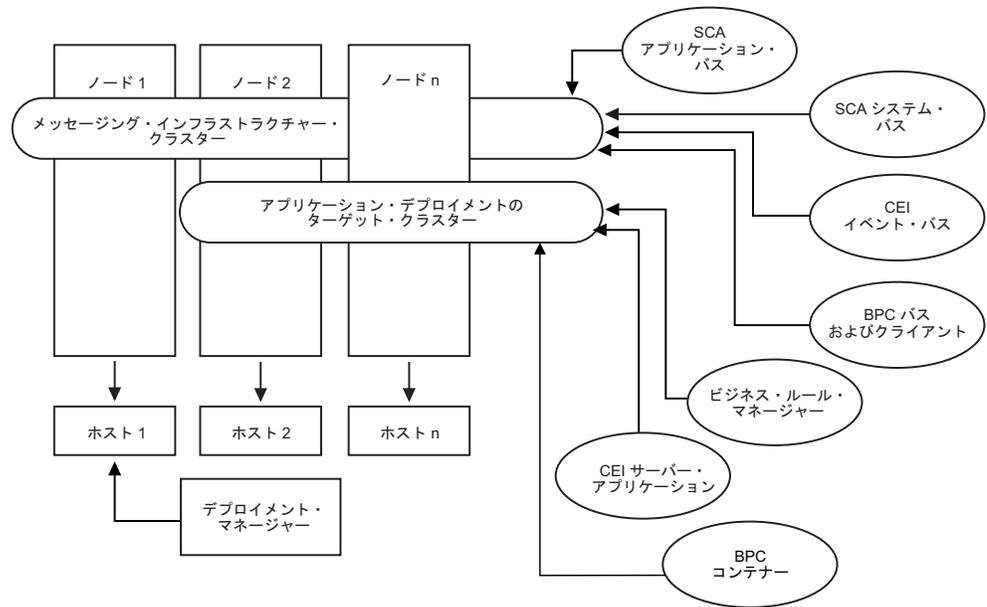


図4. リモート・メッセージング・パターン

関連概念

92 ページの『トポロジーを選択するための考慮事項』

使用するデプロイメント環境に適したトポロジーの選択は、いくつかの要因に基づいて行います。

非同期呼び出し

WebSphere Process Server は、非同期アプリケーションを開発するための強力なプログラミング・モデルを提供します。SCA で非同期呼び出しを行うとき、片方向、据え置き応答、およびコールバック付き要求という 3 つのタイプの非同期対話スタイルが使用可能です。3 つのすべてのタイプの非同期呼び出しで、`invokeAsync()` を呼び出すと、SCA ランタイムから即時にクライアントに制御が戻ります。

リモート・メッセージングおよびリモート・サポート・トポロジー

提供されている BPM トポロジー・パターンの 1 つです。リモート・メッセージングおよびリモート・サポート・パターンでは、デプロイメント環境機能は 3 つの独立したクラスター間で分割されます。

これは、WebSphere Process Server for Multiplatforms のデフォルト・パターンです。この 3 クラスター・パターンでは、最も高い負荷を処理するクラスターにリソースが割り振られます。このパターンは最も柔軟で用途が広く、ほとんどのユーザーの要望を満たすことができます (z/OS の場合を除く)。各コンポーネントは、3 つのクラスター間で分割されます。

大多数のお客様 (特に、大規模なコンピューティング・インフラストラクチャーを保有されているお客様) には、リモート・メッセージングおよびリモート・サポート・トポロジーの環境が適しています。分散プラットフォームのハードウェア要件

は増しますが、3 つ (以上) のクラスターと、特定の機能を実行する複数のメンバーを使用することにより、JVM のメモリー使用量をより柔軟に調整 (チューニング) できます。

それぞれが専用の機能およびアプリケーションを持つ 3 つのクラスターを作成することになるため、管理上の負担が増します。クラスターやクラスター・メンバーを追加するにつれて、パフォーマンス・チューニング計画の規模が大幅に拡大し、トラブルシューティングの負担が大幅に増すことがあります。メッセージング・クラスターのメンバー全体にわたってメッセージング・エンジンを分散させると、ポリシーの作成および保守に関連する管理上の負担も増します。

スケーラビリティの観点からは、リモート・メッセージングおよびリモート・サポート・トポロジーが最も高い柔軟性を提供します。WebSphere Process Server 内の個別の機能のそれぞれが 3 つのクラスター間で分割されるため、パフォーマンス上のボトルネックを正確に特定でき、クラスター・サイズもかなり容易に調整できます。追加の CEI 処理が必要な場合は、ノードおよびクラスター・メンバーをサポート・クラスターに追加すれば済みます。同様に、ビジネス・プロセスまたはヒューマン・タスクの処理能力を増強する必要がある場合は、ノードおよびメンバーをアプリケーション・ターゲット・クラスターに追加することができます。3 つより多くのクラスター・メンバーを追加してメッセージング・インフラストラクチャーを拡張しても処理能力は増強されないため、リモート・メッセージング・ポリシーに存在するスケーラビリティの制限事項は、リモート・メッセージングおよびリモート・サポート・トポロジーにも適用されます。

リモート・メッセージング・トポロジーの場合と同様に、リモート・メッセージングおよびリモート・サポート・トポロジーは、長期実行ビジネス・プロセス、ステート・マシン、ヒューマン・タスク、および非同期対話 (JMS および MQ/JMS バインディングを含む) のための理想的な環境を提供します。アプリケーション・ターゲット・クラスターの役割はビジネス・インテグレーション・アプリケーションの実行のみであるため、パフォーマンスの調整および診断は、アプリケーション・ターゲット・クラスターがその他の役割も担っている以前のトポロジーよりもはるかに単純です。リモート・メッセージングおよびリモート・サポート・トポロジーは、モニターおよび監査のために CEI を頻繁に使用する環境 (WebSphere Business Monitor を使用する環境を含む) でも理想的です。サポート・インフラストラクチャーが専用のクラスターに分割されることで、CEI およびサポートを行うアプリケーション (BPC エクスプローラーや Business Space など) 向けの、専用のクラスター・メンバー・セットが提供されます。

リモート・メッセージング・インフラストラクチャー・クラスター:

- Service Component Architecture (SCA) アプリケーション・バス・メンバー
- SCA システム・バス・メンバー
- Business Process Choreographer (BPC) バス・メンバー
- Common Event Interface (CEI) バス・メンバー

リモート・サポート・インフラストラクチャー・クラスター:

- CEI サーバー・アプリケーション
- ビジネス・ルール・マネージャー
- Explorer などの Business Process Choreographer の各コンポーネント

アプリケーション・デプロイメント・クラスター:

- アプリケーション・デプロイメントのターゲット
- Business Process Choreographer Container

アプリケーション・デプロイメント・ターゲットを構成して、SCA アプリケーションと Business Process Choreographer コンポーネントをサポートします。

以下に示すリモート・メッセージング・クラスター・トポロジーのグラフィカル表現を参照してください。

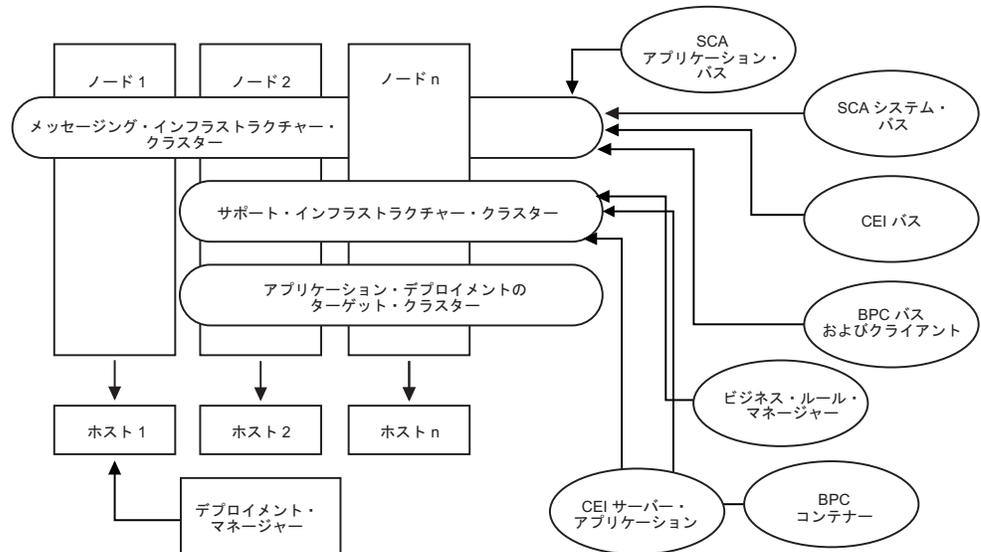


図5. リモート・メッセージングおよびリモート・サポート・パターン

リソース割り振りの例

次の図は、リソースがリモート・メッセージングおよびリモート・サポート・パターンを使用して割り振られる場合がある 1 つの例を示しています。この図は、3 つのホストを示しています。ホスト A にはサーバー 1 とサーバー 3 があります。ホスト B にはサーバー 2、サーバー 4、およびサーバー 5 があります。ホスト C にはサーバー 6 とサーバー 7 があります。このインストール済み環境に対する最も重い負荷はアプリケーションの使用によって発生するため、アプリケーション・デプロイメントのターゲット・クラスター (クラスター 3) には、(サーバー 1、サーバー 2、およびサーバー 6) のために他の機能よりも多くのリソースが割り振られています。

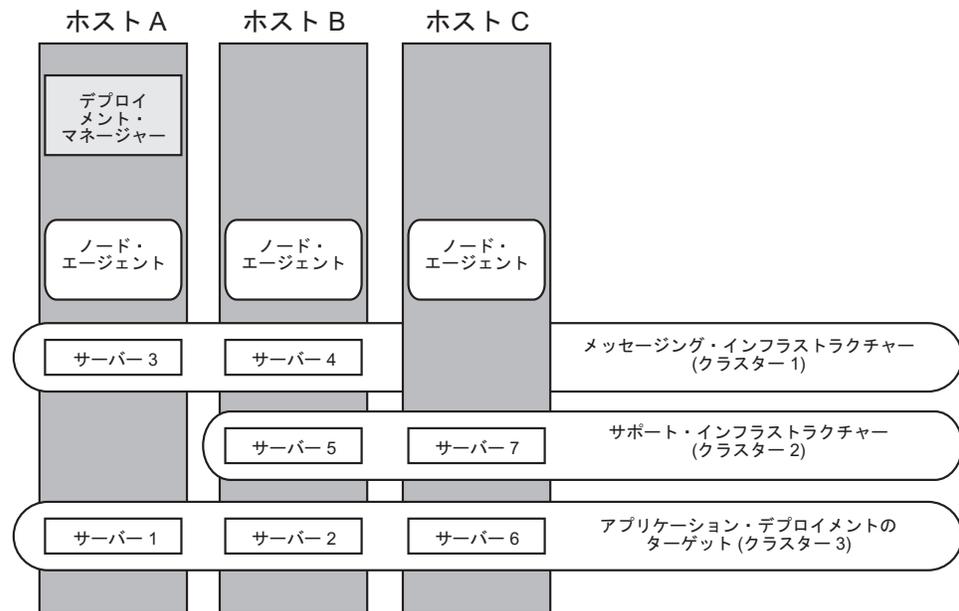


図6. リソース割り振りの例

関連概念

92 ページの『トポロジーを選択するための考慮事項』

使用するデプロイメント環境に適したトポロジーの選択は、いくつかの要因に基づいて行います。

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・トポロジー

パターン化された BPM トポロジーの 1 つです。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・パターンでは、デプロイメント環境の各機能が 4 つの独立したクラスター間で分割されます。

このパターンは、WebSphere Process Server のみのインストール済み環境およびデプロイメント・マネージャー構成には適用されません。これが適用されるのは、WebSphere Business Monitor を組み込むために拡張されたデプロイメント・マネージャー用のデプロイメント環境を作成する場合のみです。

4 つのクラスターから成るこのパターンは、サポートする Web アプリケーションが独自のクラスターに存在することを除いて、リモート・メッセージングおよびリモート・サポート・パターンと類似しています。

リモート・メッセージング・インフラストラクチャー・クラスター:

- Service Component Architecture (SCA) アプリケーション・バス・メンバー
- SCA システム・バス・メンバー
- Business Process Choreographer (BPC) バス・メンバー
- Common Event Interface (CEI) バス・メンバー

リモート・サポート・インフラストラクチャー・クラスター:

- CEI サーバー・アプリケーション

アプリケーション・デプロイメント・クラスター:

- アプリケーション・デプロイメントのターゲット
- Business Process Choreographer Container

リモート Web アプリケーション・インフラストラクチャー・クラスター

- ビジネス・ルール・マネージャー
- Business Process Choreographer Explorer
- Business Space

アプリケーション・デプロイメント・ターゲットを構成して、SCA アプリケーションと Business Process Choreographer コンポーネントをサポートします。

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・パターンでは、デプロイメント環境の各機能が 4 つの独立したクラスター間で分割されます (メッセージング機能に対して 1 つのクラスター、サポート機能に対して 1 つのクラスター、アプリケーションに対して 1 つのクラスター、Web ベースの機能に対して 1 つのクラスター)。

リモート・メッセージング、リモート・サポート、および Web トポロジー・パターンは、WebSphere Dynamic Process Edition から始める場合の推奨の開始トポロジーです。WebSphere Dynamic Process Edition では、Web UI コンポーネントが頻繁に使用されます。このパターンでは、以下の Web アプリケーションを収容するために、4 番目のクラスターを使用します。

- BPC ツール
- ビジネス・ルール・マネージャー
- Business Space
- REST API サービス
- データの多次元表示のための AlphaBlox

使用環境にデプロイされている個々のコンポーネントを詳細に制御できる点を除けば、このトポロジー・パターンの利点は、「リモート・メッセージングおよびリモート・サポート」トポロジーの利点と同様です。

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・トポロジーのグラフィカル表現を参照してください。

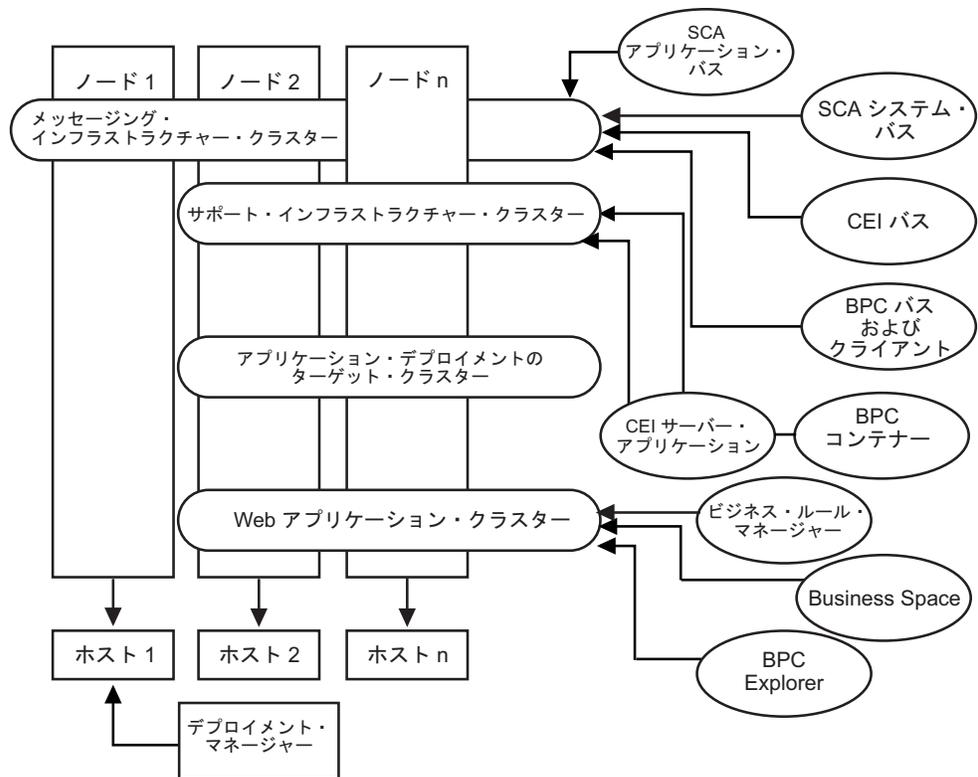


図7. リモート・メッセージング、サポート、および Web のパターン

関連概念

92 ページの『トポロジーを選択するための考慮事項』

使用するデプロイメント環境に適したトポロジーの選択は、いくつかの要因に基づいて行います。

カスタム・トポロジー

カスタム・トポロジーは、お客様の状態に固有の処理要件およびビジネス要件に対処します。これは、パターン化されたトポロジーではなく、具体的なニーズに合わせて作成および調整するトポロジーです。

カスタム・トポロジーでは、独自の環境を定義できるため、柔軟性が格段に高くなっています。前述したように、提供されているトポロジー・パターン（「単一クラスター」、「リモート・メッセージング」、「リモート・メッセージングおよびリモート・サポート」、および「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」）では、すべての WebSphere Process Server コンポーネントがデフォルトの場所にデプロイされます。これらのコンポーネントに関連する追加のオーバーヘッドは、必要な場合とそうでない場合があります。例えば、組織が CEI を必要としない場合は、CEI サポートおよび BPC オブザーバーを使用環境から取り除いたカスタム・トポロジーを作成できます。同様に、ビジネス・ルール・マネージャーの利用を許可しないガバナンス・ルールが組織に存在する場合は、デプロイメント環境からビジネス・ルール・マネージャーを取り除くことができます。

使用環境にデプロイされている個々のコンポーネントを詳細に制御できる点を除けば、カスタム・トポロジーの利点は、「リモート・メッセージングおよびリモート・サポート」トポロジーの利点と同様です。また、欠点も同様です。

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

トポロジー・パターンの可用性およびデフォルトの使用法は、BPM 製品ごとに異なります。

管理コンソールで「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場合、デプロイメント環境のベースにするトポロジー・パターンの可用性は、以下の条件および構成上の決定事項によって異なります。

- WebSphere Process Server をインストールしたプラットフォーム
- 1 次デプロイメント環境フィーチャーおよび補足フィーチャー

以下の表に、トポロジー・パターンと BPM 製品との間の関係を示します。

表 15. 使用可能な提供パターンおよびそれらのパターンとフィーチャーとの関係

トポロジー・パターン	クラスター数	説明	サポートされる BPM 製品とフィーチャー	デフォルト状況
単一クラスター	1	<p>単一クラスター・トポロジー・パターンは、すべての機能コンポーネント用に 1 つのクラスターを提供します。ユーザー・アプリケーション、メッセージング・インフラストラクチャー、CEI、およびサポート・アプリケーションのすべてが、同じクラスターで構成されます。ほとんどの企業では、このパターンを使用してデプロイメント環境をサポートできます。このパターンには、zWLM が管理するサーバント領域によって、ポリッシュド・シルバー・トポロジーとして知られる z/OS 上での単一クラスターを作成できるなど、運用上の利点があるからです。z/OS では、このトポロジーが推奨されます。</p> <p>これは、WebSphere Process Server 実稼働環境のデフォルト・パターンです。</p>	<p>以下の単一 BPM 製品インストール済み環境でサポートされます。</p> <ul style="list-style-type: none"> • WebSphere Process Server • WebSphere ESB • WebSphere Business Services Fabric • WebSphere Business Monitor <p>以下のマルチフィーチャー・インストール済み環境でサポートされます。</p> <ul style="list-style-type: none"> • WebSphere Business Monitor + WebSphere ESB • WebSphere Business Monitor + WebSphere Process Server • WebSphere Business Monitor + WebSphere Business Services Fabric 	これは、z/OS インストール済み環境でのデフォルト・パターンです。

表 15. 使用可能な提供パターンおよびそれらのパターンとフィーチャーとの関係 (続き)

トポロジー・パターン	クラスター数	説明	サポートされる BPM 製品とフィーチャー	デフォルト状況
リモート・メッセージング	2	リモート・メッセージング・パターンでは、メッセージング環境をアプリケーション・デプロイメント・ターゲットおよびアプリケーション・サポート機能から分離します。このパターンは、日常の運用でメッセージのスループットが重要な要件となる場合に使用します。	以下の単一 BPM 製品インストール済み環境でサポートされます。 <ul style="list-style-type: none"> • WebSphere Process Server • WebSphere ESB • WebSphere Business Services Fabric 	これはデフォルトのパターンではありません。
リモート・メッセージングおよびリモート・サポート	3	リモート・メッセージングおよびリモート・サポート・パターンでは、メッセージング、Common Event Infrastructure (CEI)、アプリケーション・デプロイメント・ターゲット、およびアプリケーション・サポート機能を別個のクラスターに分離します。ほとんどの業務では、このパターンを使用してデプロイメント環境をサポートできます。このパターンはパフォーマンスを重視し、トランザクション処理をメッセージングなどのサポート機能から分離して設計されているためです。	以下の単一 BPM 製品インストール済み環境でサポートされます。 <ul style="list-style-type: none"> • WebSphere Process Server • WebSphere ESB • WebSphere Business Services Fabric 	これは、以下のインストール済み環境でのデフォルト・パターンです。 <ul style="list-style-type: none"> • マルチプラットフォーム上の WebSphere Process Server • マルチプラットフォーム上の WebSphere Enterprise Service Bus • WebSphere Business Services Fabric

表 15. 使用可能な提供パターンおよびそれらのパターンとフィーチャーとの関係 (続き)

トポロジー・パターン	クラスター数	説明	サポートされる BPM 製品とフィーチャー	デフォルト状況
リモート・メッセージング、リモート・サポート、および Web アプリケーション	4	<p>このパターンは、WebSphere Process Server のみのインストール済み環境およびデプロイメント・マネージャー構成には適用されません。これが適用されるのは、WebSphere Business Monitor を組み込むために拡張されたデプロイメント・マネージャー用のデプロイメント環境を作成する場合のみです。</p> <p>このパターンは、アプリケーション・デプロイメント用に 1 つのクラスター、メッセージング・インフラストラクチャー用に 1 つのリモート・クラスター、サポート・アプリケーション用に 1 つのリモート・クラスター、および Web アプリケーション・デプロイメント (Business Process Choreographer Explorer、Business Space、およびビジネス・ルール・マネージャー) 用に 1 つのリモート・クラスターを定義します。</p>	<p>以下の単一 BPM 製品インストール済み環境でサポートされます。</p> <ul style="list-style-type: none"> • WebSphere Business Monitor <p>以下のマルチフィーチャー・インストール済み環境でサポートされます。</p> <ul style="list-style-type: none"> • WebSphere Business Monitor + WebSphere ESB • WebSphere Business Monitor + WebSphere Process Server • WebSphere Business Monitor + WebSphere Business Services Fabric 	これは、WebSphere Business Monitor インストール済み環境でのデフォルト・パターンです。

トポロジーを選択するための考慮事項

使用するデプロイメント環境に適したトポロジーの選択は、いくつかの要因に基づいて行います。

それらの要因には以下の項目が含まれますが、これらだけに限定されるわけではありません。

- 使用可能なハードウェア・リソース
- アプリケーションの呼び出しパターン
- 実装を計画しているビジネス・プロセスのタイプ (割り込み可能か割り込み不可能か)
- CEI の予定使用頻度
- 個別のスケーラビリティ要件
- 必要になる管理作業

トポロジー選択基準の要約

以下の表に記載されている情報を考慮してください。これは、使用する実動トポロジーを選択する場合の簡単な手引きです。この表には、各トポロジー・パターンの利点と欠点の要約リストを示します。

提供トポロジー・パターンをサポートする BPM 製品については、『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。

表 16. デプロイメント環境に合わせてトポロジーを選択するための考慮事項

考慮事項	トポロジー・パターン			
	単一クラスター	リモート・メッセージング	リモート・メッセージングおよびリモート・サポート	リモート・メッセージング、リモート・サポート、および Web
保守するクラスターの数	すべてのコンポーネントに対して 1 クラスター	アプリケーションおよびサポート・インフラストラクチャーに対して 1 クラスター メッセージングに対して 1 クラスター	アプリケーションに対して 1 クラスター サポート・インフラストラクチャーに対して 1 クラスター サポート・インフラストラクチャーに対して 1 クラスター	アプリケーションに対して 1 クラスター Web インターフェースに対して 1 クラスター サポート・インフラストラクチャーに対して 1 クラスター メッセージングに対して 1 クラスター
ハードウェア要件	限られた数のハードウェアで実装できる	分散環境ではより多くのハードウェアが必要	分散環境ではより多くのハードウェアが必要	必要なハードウェアが最も多い
非同期対話	最低限の使用頻度に抑えることが望ましい	使用頻度はリソース・アベイラビリティとのバランスをとる必要がある	非同期対話に理想的な環境	非同期対話に理想的な環境

表 16. デプロイメント環境に合わせてトポロジーを選択するための考慮事項 (続き)

考慮事項	トポロジー・パターン			
	単一クラスター	リモート・メッセージング	リモート・メッセージングおよびリモート・サポート	リモート・メッセージング、リモート・サポート、および Web
長期実行プロセス、ステート・マシン、およびヒューマン・タスク	最低限の使用頻度に抑えることが望ましい	使用頻度はリソース・アベイラビリティとのバランスをとる必要がある	割り込み可能プロセス、ステート・マシン、およびヒューマン・タスクに理想的な環境	割り込み可能プロセス、ステート・マシン、およびヒューマン・タスクに理想的な環境
非常に高い CEI 活動の頻度	非推奨 (CEI の使用頻度を低くしてリソース使用量とのバランスをとることが望ましい)	非推奨 (CEI の使用頻度を低くしてリソース使用量とのバランスをとることが望ましい)	CEI 使用頻度が高い場合に理想的な環境	CEI 使用頻度が高い場合に理想的な環境
管理上の負担	比較的少ない	手間が増す	管理の手間が増す	管理に最も多くの手間がかかる
スケーラビリティ	すべてのコンポーネントは同じ比率で拡張される	メッセージング・クラスターのスケーラビリティは制限されている (サーバーの数を 3 つより多くしてもメリットがない) それ以外のすべてのコンポーネントは同じ比率で拡張される	拡張が容易 すべての機能が独立している メッセージング・クラスターのスケーラビリティは引き続き制限されている (サーバーの数を 3 つより多くしてもメリットがない)	スケールの変更が最も容易 すべての機能が独立している メッセージング・クラスターのスケーラビリティは引き続き制限されている (他の BPM 製品を導入するとメリットが出てくる)

関連概念

73 ページの『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。

呼び出しスタイル

ビジネス・プロセスのタイプ

79 ページの『単一クラスター・トポロジー』

提供されている BPM トポロジー・パターンの 1 つです。単一クラスター・トポロジーでは、WebSphere Process Server 環境のすべての機能が単一のクラスターにまとめられています。

81 ページの『リモート・メッセージング・トポロジー』

提供されている BPM トポロジーの 1 つです。リモート・メッセージング・パターンでは、デプロイメント環境の機能が 2 つの独立したクラスター間で分割されます。

83 ページの『リモート・メッセージングおよびリモート・サポート・トポロジー』

提供されている BPM トポロジー・パターンの 1 つです。リモート・メッセージングおよびリモート・サポート・パターンでは、デプロイメント環境機能は 3 つの独立したクラスター間で分割されます。

86 ページの『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・トポロジー』

パターン化された BPM トポロジーの 1 つです。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・パターンでは、デプロイメント環境の各機能が 4 つの独立したクラスター間で分割されます。

第 5 章 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。

始める前に

デプロイメント環境を計画する前に、以下の作業を実行します。

- データベース・タイプの選択
- 使用可能なリソースの識別
- 必要なセキュリティ権限の明確化

このタスクについて

相互接続サーバーのレイアウトを計画する際には、いくつかの決定を下す必要があります。こうした決定は、使用可能なハードウェアと物理接続の間で行われるトレードオフ、管理および構成の複雑さ、およびパフォーマンス、可用性、スケーラビリティ、分離機能、セキュリティ、安定度などの要件に影響を与えます。

手順

1. デプロイメント環境の機能要件を明確化します。
 - a. デプロイメント環境のフィーチャーおよびランタイム機能を明確化します。

デプロイメント環境は WebSphere Process Server に加えて BPM 製品もサポートしますか。
 - b. デプロイするコンポーネント・タイプを決定します。

コンポーネント・タイプとコンポーネント間の対話を要件の一部として検討します。
 - c. インポートおよびエクスポートの実装タイプとトランスポートを決定します。

データベースまたは Java Message Service (JMS) リソースに必要なリソース、およびビジネス・イベントとそれらの伝送手段に必要な事柄について考慮します。
 - d. アプリケーションに関連しないすべての機能要件も明確化します。

セキュリティ・サーバー、ルーター、およびビジネス・イベントを処理するための他のすべてのハードウェア要件またはソフトウェア要件を検討します。
2. ご使用の環境に対する容量とパフォーマンスの要件を明確化します。
3. ご使用の環境に対する冗長度の要件を明確化します。
 - a. フェイルオーバーに必要なサーバーの数を決定します。
4. デプロイメント環境を設計します。

パターンを決定します。 WebSphere Process Server の場合、以下の 3 つの確立されたクラスター・パターンの中から選択できます。

- 単一クラスター
- リモート・メッセージング
- リモート・メッセージングおよびリモート・サポート

必要を満たすパターンがこれらにない場合、独自のカスタム・デプロイメント環境を作成できます。

注: WebSphere Process Server に加えて、これと互換性のある BPM 製品もサポートしている構成の場合は、デプロイメント環境の作成時に、これらの製品のパターンを使用することができます。例えば、「**リモート・メッセージング、リモート・サポート、および Web**」パターンを WebSphere Business Monitor で使用できます。

各パターンとそれらの相違点について詳しくは、『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。

5. デプロイメント環境の構成に使用できる方法を理解します。

以下の方法を使用して、 WebSphere Process Server のデプロイメント環境を構成できます。

- 管理コンソールから「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

管理コンソールから「デプロイメント環境構成」ウィザードを使用して、単一クラスター、リモート・メッセージング、リモート・メッセージングとリモート・サポート、および (該当する場合) リモート・メッセージング、サポート・クラスターと Web クラスターを作成できます。また、カスタム・デプロイメント環境は、管理コンソールの「デプロイメント環境構成」ウィザードを使用するか、または管理コンソールによってその環境を自分で作成して、作成することもできます。

- wsadmin を使用したデプロイメント環境の作成
- プロファイル作成時にプロファイル管理ツール (PMT) を使用したデプロイメント環境の作成
- プロファイル作成時に manageprofiles コマンド行ユーティリティを使用したデプロイメント環境の作成

注: インストーラーからのデプロイメント環境の作成はサポートされていません。

次のタスク

お客様の状況に最適な計画のシナリオを選択して、それに従ってください。

関連概念

113 ページの『スタンドアロン・サーバー』

スタンドアロン・サーバーは、Service Component Architecture (SCA) モジュールを 1 つのサーバー・プロセスにデプロイするための環境を提供します。このサーバー・プロセスには、管理コンソール、デプロイメント・ターゲット、メッセージング・サポート、ビジネス・ルール・マネージャー、および Common Event Infrastructure サーバーが含まれます (ほかのものが含まれる場合もあります)。

117 ページの『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。

73 ページの『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。

118 ページの『デプロイメント環境のクラスター』

クラスターを使用すると、アプリケーションの能力と可用性が単一サーバーの場合よりも高まります。



WebSphere Process Server 用のサービス統合バス

サービス統合バスとは、同期および非同期メッセージングによってサービス統合をサポートする、管理された通信メカニズムです。バスは、バス・リソースを管理する相互接続メッセージング・エンジンで構成されます。サービス統合バスは、WebSphere Process Server の基盤となる WebSphere Application Server テクノロジーの 1 つです。



サービス・コンポーネント

IBM WebSphere Process Server で実行されるすべての統合成果物 (例えば、ビジネス・プロセス、ビジネス・ルール、およびヒューマン・タスク) は、明確に定義されたインターフェースを持つコンポーネントとして表されます。

計画のためのシナリオ

デプロイメント環境を計画する方法は、デプロイメント環境を使用する方法によって決まります。以下の各シナリオをすべて読んで、デプロイメント環境を使用する方法に最もよく一致するシナリオを見つけてください。

デフォルトのスタンドアロン環境の計画

このシナリオは、デプロイメント環境を他の環境から分離する必要がある場合に使用します。この環境で実行されるアプリケーションは、必要なものをそれ自体が完備し、Web サービス SOAP/HTTP などの限られたインポート・プロトコルを使用する必要があります。このシナリオは、インストールおよびセットアップの容易さが高可用性に対する要件よりも重要な場合にも使用します。

始める前に

- デプロイメント環境を設計します。
- すべてのビジネス要件を単一サーバーで満たすことができることを確認します。
- スタンドアロン・プロファイルの概念についてよく理解してください。

このタスクについて

ニーズを満たすためにデフォルトの単一のサーバー環境をインストールする必要があります。

手順

1. 設計をサポートするために必要なハードウェアおよびソフトウェアを決定します。
2. インストールを完了するために必要な権限を持つすべてのユーザー ID を特定または作成します。
3. オプション: サイト・ポリシーによってデータベースの作成と中央の部門へのアクセスが制限される場合は、担当のデータベース管理者に連絡してください。

重要: 将来の計画に、この環境をデプロイメント・マネージャー・セルに統合することが含まれている場合は、リモート・アクセスをサポートするデータベースとデータベース・ドライバーを使用していることを必ず確認してください。これらのタイプの製品の例としては、Derby Network と Java Toolbox JDBC があります。

4. WebSphere Integration Developer および WebSphere Process Server のインストールを、開発コミュニティへの影響を最小限に抑えるように調整してスケジュールします。

WebSphere Integration Developer のインストールについては、IBM WebSphere Integration Developer 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

ソフトウェアをインストールします。

関連概念

113 ページの『スタンドアロン・サーバー』
スタンドアロン・サーバーは、Service Component Architecture (SCA) モジュールを 1 つのサーバー・プロセスにデプロイするための環境を提供します。このサーバー・プロセスには、管理コンソール、デプロイメント・ターゲット、メッセージング・サポート、ビジネス・ルール・マネージャー、および Common Event Infrastructure サーバーが含まれます (ほかのものが含まれる場合もあります)。

112 ページの『サーバー』
サーバーによって、WebSphere Process Server のコア機能が提供されます。プロセス・サーバーでは、アプリケーション・サーバーの機能が拡大または拡張されて、Service Component Architecture (SCA) モジュールが処理されます。他のサーバー (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) は、プロセス・サーバーの管理に使用されます。

zPMT ツール

関連タスク

zPMT ツールを使用した応答ファイルの作成

WebSphere Process Server for z/OS の構成

JCA 認証別名ユーザー ID への表の特権の付与

関連情報

 Business Process Choreographer の構成

カスタム・スタンドアロン環境の計画

このシナリオは、分離された環境を必要としているが、ビジネス要件のためにデフォルトの単一サーバー環境を使用できない場合に利用します。

始める前に

- デプロイメント環境を設計します。
- すべてのビジネス要件を単一サーバーで満たすことができることを確認します。
- スタンドアロン・プロファイルの概念についてよく理解してください。

このタスクについて

ニーズを満たすためにデフォルトの単一のサーバー環境をインストールする必要がある設計があります。

手順

1. デプロイメント環境をサポートするためのデータベース製品を選択します。

DB2 データベース (DB2 for i5/OS、DB2 for IBMi、および DB2 for z/OS を含む) を使用するシステムの場合、メッセージング・エンジンと Common Event Infrastructure (CEI) 用のデータベースとテーブルを自動的に作成することはできません。これらのシステム用のデータベースを作成する際には、データベース定義スクリプトを正常に実行するために必要な十分な権限があることを確認します。

重要: 将来の計画に、この環境をデプロイメント・マネージャー・セルに統合することが含まれている場合は、リモート・アクセスをサポートするデータベースとデータベース・ドライバーを使用していることを必ず確認してください。これらのタイプの製品の例としては、Derby Network と Java Toolbox JDBC があります。

2. データベース表の作成方法を決定します。

製品のインストール時にテーブルを作成するか、製品のインストール・プロセスにテーブルを作成するためのスクリプトを作成させるか、またはこの手順を実行するスクリプトを自分で作成します。

3. クライアントにデプロイメント環境内のアプリケーションにアクセスさせる方法を決定します。

アクセスの方法には、ユーザーのニーズに基づき、Web サービス (SOAP/HTTP と SOAP/JMS)、同期または非同期の Service Component Architecture (SCA) 要求、Java Message Service (JMS)、MQ (JMS またはネイティブ)、またはアダプターを介してなど多数の方法があります。これらの選択は、インストールする必要のある他のソフトウェアとリソースに影響を与えます。

4. アプリケーションで必要となるリソースにアプリケーションがアクセスする方法を決定します。

アクセスの方法には、ユーザーのニーズに基づき、Web サービス (SOAP/HTTP と SOAP/JMS)、同期または非同期の Service Component Architecture (SCA) 要求、Java Message Service (JMS)、MQ (JMS またはネイティブ)、またはアダプターを介してなど多数の方法があります。これらの選択は、インストールする必要のある他のソフトウェアとリソースに影響を与えます。

5. ソフトウェアのインストール方法、サーバーの作成および構成方法を決定します。

ソフトウェアのインストール時にサーバーを作成して構成でき、またプロファイル管理ツールを使用してサーバーを作成して構成することもできます。さらに、管理コンソールを使用してサーバーを作成して構成することもできます。経験を積んだインストール担当者は、これらのタスクを処理するためにスクリプトを使用することもできます。すべての方法の利点と欠点を理解してから、選択を行うようにしてください。

6. インストールを完了するために必要な権限を持つすべてのユーザー ID を特定または作成します。
7. オプション: サイト・ポリシーによってデータベースの作成と中央の部門へのアクセスが制限される場合は、担当のデータベース管理者に連絡してください。

重要: 将来の計画に、この環境をデプロイメント・マネージャー・セルに統合することが含まれている場合は、リモート・アクセスをサポートするデータベースとデータベース・ドライバーを使用していることを必ず確認してください。これらのタイプの製品の例としては、Derby Network と Java Toolbox JDBC があります。

8. WebSphere Integration Developer および WebSphere Process Server のインストールを、開発コミュニティへの影響を最小限に抑えるように調整してスケジュールします。

WebSphere Integration Developer のインストールについて詳しくは、IBM WebSphere Integration Developer 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

ソフトウェアをインストールします。

関連概念

113 ページの『スタンドアロン・サーバー』

スタンドアロン・サーバーは、Service Component Architecture (SCA) モジュールを 1 つのサーバー・プロセスにデプロイするための環境を提供します。このサーバー・プロセスには、管理コンソール、デプロイメント・ターゲット、メッセージング・サポート、ビジネス・ルール・マネージャー、および Common Event Infrastructure サーバーが含まれます (ほかのものが含まれる場合もあります)。

112 ページの『サーバー』

サーバーによって、WebSphere Process Server のコア機能が提供されます。プロセス・サーバーでは、アプリケーション・サーバーの機能が拡大または拡張されて、Service Component Architecture (SCA) モジュールが処理されます。他のサーバー (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) は、プロセス・サーバーの管理に使用されます。

zPMT ツール

DB2 の決定

関連タスク

12 ページの『インストールする製品の決定』

デプロイメント環境の設計には、必要になる可能性があるソフトウェア製品の数とタイプの判断が含まれます。製品の要件は、お客様のニーズに基づき、その環境に関連するコンピューター LPAR システムによって異なる可能性があります。デプロイメント環境のすべてのサーバーにそれぞれ WebSphere Process Server ライセンスが必要なわけではありません。

9 ページの『使用可能なリソースの明確化』

資産を明確化し、既に使用可能なリソースを最大限に利用するとともに、購買の決定の通知を受け取ります。

zPMT ツールを使用した応答ファイルの作成

WebSphere Process Server for z/OS の構成

JCA 認証別名ユーザー ID への表の特権の付与

関連情報

 Business Process Choreographer の構成

提供されたパターンの 1 つに基づくデプロイメント環境の計画

このシナリオは、Service Component Architecture (SCA) アプリケーションに対するスケーラビリティ、可用性、およびサービス品質の要件がある場合で、IBM 提供のパターンの 1 つでそれらの要件を満たすことができる場合に使用します。

始める前に

これらのトピックと関連したトピックに関する情報についてよく理解してください (まだ理解していない場合)。

- サーバー
- クラスター
- プロファイル
- データベースの選択
- デプロイメント環境
- デプロイメント環境の機能
- デプロイメント環境パターン

ご使用のデプロイメント環境で使用しているハードウェアの図表を作成し、各機器がホストするサーバーを示します。また、サーバーをクラスター化する方法についてより明確な認識を持てるように、デプロイメント環境機能を提供するサーバーも明確化します。

このタスクについて

ビジネス・ニーズの分析を完了していて、ニーズを満たすには単一サーバーでは不十分であることが判明しています。高可用性およびフェイルオーバーを提供するために、複数のサーバーを必要としています。お客様の設計が IBM 提供のデプロイメント環境パターンの 1 つに一致しています。

手順

1. 設計をサポートするために必要なハードウェアおよびソフトウェアを決定します。
2. デプロイメント環境をサポートするためのデータベース製品を選択します。

DB2 データベース (DB2 for i5/OS、DB2 for IBMi、および DB2 for z/OS を含む) を使用するシステムの場合、メッセージング・エンジンと Common Event Infrastructure (CEI) 用のデータベースとテーブルを自動的に作成することはできません。これらのシステム用のデータベースを作成する際には、データベース定義スクリプトを正常に実行するために必要な十分な権限があることを確認します。

重要: 将来の計画に、この環境をデプロイメント・マネージャー・セルに統合することが含まれている場合は、リモート・アクセスをサポートするデータベースとデータベース・ドライバーを使用していることを必ず確認してください。これらのタイプの製品の例としては、Derby Network と Java Toolbox JDBC があります。

3. データベース表の作成方法を決定します。

製品のインストール時にテーブルを作成するか、製品のインストール・プロセスにテーブルを作成するためのスクリプトを作成させるか、またはこの手順を実行するスクリプトを自分で作成します。

4. IBM 提供のいずれのパターンが設計に最適かを判断します。

5. 各サーバーを、設計で指定した機能を提供するクラスターのメンバーとしてマップします。

選択したパターンにより、ノードがクラスターにマップされ、メンバー数とそれらの配分が決まります。

6. クライアントにデプロイメント環境内のアプリケーションにアクセスさせる方法を決定します。

アクセスの方法には、ユーザーのニーズに基づき、Web サービス (SOAP/HTTP と SOAP/JMS)、同期または非同期の Service Component Architecture (SCA) 要求、Java Message Service (JMS)、MQ (JMS またはネイティブ)、またはアダプターを介してなど多数の方法があります。これらの選択は、インストールする必要のある他のソフトウェアとリソースに影響を与えます。

7. アプリケーションで必要となるリソースにアプリケーションがアクセスする方法を決定します。

アクセスの方法には、ユーザーのニーズに基づき、Web サービス (SOAP/HTTP と SOAP/JMS)、同期または非同期の Service Component Architecture (SCA) 要求、Java Message Service (JMS)、MQ (JMS またはネイティブ)、またはアダプターを介してなど多数の方法があります。これらの選択は、インストールする必要のある他のソフトウェアとリソースに影響を与えます。

8. ソフトウェアのインストール方法、サーバーの作成方法、および作成したサーバーの構成方法を決定します。

ソフトウェアのインストール中に**開発スタンドアロン・サーバー・プロファイル**を作成できます。また、プロファイル管理ツールを使用して、サーバーを作成および構成することもできます。さらに、管理コンソールまたはスクリプトを使用してサーバーを作成して構成することもできます。すべての方法の利点と欠点を理解してから、選択を行うようにしてください。

注: 開発スタンドアロン・サーバー・プロファイルはテストのみを目的としており、実稼働環境では使用できません。

9. 同じハードウェア上に作成するすべてのサーバーがそのシステム上のリソースを共有する方法を決定します。

ソフトウェアを別々の場所にインストールすることも、複数の異なるプロファイルを使用することもできます。また、z/OS の場合は、複数の異なるロジカル・パーティションを使用してリソースを共有することもできます。

WebSphere Process Server for z/OS の区分データ・セットの計画については、WebSphere Process Server for z/OS の資料を参照してください。

10. インストールを完了するために必要な権限を持つすべてのユーザー ID を特定または作成します。

次のタスク

デプロイメント環境をインストールします。

関連概念

112 ページの『サーバー』

サーバーによって、WebSphere Process Server のコア機能が提供されます。プロセス・サーバーでは、アプリケーション・サーバーの機能が拡大または拡張されて、Service Component Architecture (SCA) モジュールが処理されます。他のサーバー (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) は、プロセス・サーバーの管理に使用されます。

115 ページの『デプロイメント・マネージャー』

デプロイメント・マネージャーとは、他のサーバーの論理グループ (セル) の操作を管理するサーバーです。デプロイメント・マネージャーは、サーバーおよびクラスターを管理するための中央の場所になっています。

118 ページの『管理対象サーバー』

管理対象サーバーとは、管理対象ノード内に構成されるサーバーのことです。管理対象サーバーにより、アプリケーションが実行されるデプロイメント環境内に特定のリソースが提供されます。

118 ページの『デプロイメント環境のクラスター』

クラスターを使用すると、アプリケーションの能力と可用性が単一サーバーの場合よりも高まります。

73 ページの『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。



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。

11 ページの『WebSphere Process Server と他の WebSphere Application Server 製品との間のインターオペラビリティの計画』

ソフトウェア環境を分析するときは、デプロイメント環境内に存在するさまざまなソフトウェア・レベル間で要求を受け渡すことができるかどうかを把握する必要があります。

143 ページの『エラー処理方針とソリューション・リカバリー』

WebSphere Process Serverには、リカバリーのために使用できるエラー処理機能とツールが組み込まれています。

146 ページの『実稼働環境でのリカバリー』

実稼働環境での目標は、整然とした一貫性のある方法でシステムに入力された要求すべてを処理することです。この環境ではデータを保持する必要があり、システムを使用できなったり、データを損失したりする状況を最小限に抑えるための手段をすべて実施する必要があります。

関連タスク

97 ページの『第 5 章 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。

9 ページの『使用可能なリソースの明確化』

資産を明確化し、既に使用可能なリソースを最大限に利用するとともに、購買の決定の通知を受け取ります。

12 ページの『インストールする製品の決定』

デプロイメント環境の設計には、必要になる可能性があるソフトウェア製品の数とタイプの判断が含まれます。製品の要件は、お客様のニーズに基づき、その環境に関連するコンピューター LPAR システムによって異なる可能性があります。デプロイメント環境のすべてのサーバーにそれぞれ WebSphere Process Server ライセンスが必要なわけではありません。

JCA 認証別名ユーザー ID への表の特権の付与

関連情報

 Network Deployment のインストールの計画

 概要: クラスタ

 Business Process Choreographer の構成

カスタム・デプロイメント環境の計画

このシナリオは、サービス品質要件があるか、または IBM 提供のパターンで定義されたデプロイメント環境よりも複雑なデプロイメント環境が必要な場合に使用します。

始める前に

重要: カスタム・デプロイメント環境のインストールは、デフォルトのデプロイメント環境のインストールよりも複雑であり、Network Deployment、クラスタ化、および他の WebSphere Process Server の機能の理解が必要になります。IBM では、デプロイメント環境の各部分を個々に計画して、段階的に実装することをお勧めします。

これらのトピックと関連したトピックに関する情報についてよく理解してください (まだ理解していない場合)。

- サーバー
- クラスタ
- プロファイル
- カスタム・デプロイメント環境とそれらの機能
- Business Process Choreographer コンポーネントおよび構成

ご使用のデプロイメント環境で使用しているハードウェアの図表を作成し、各機器がホストするサーバーを示します。また、サーバーをクラスタ化する方法についてより明確な認識を持てるように、デプロイメント環境機能を提供するサーバーも明確化します。

設計では、どのクラスターがメッセージング、Common Event Infrastructure、およびアプリケーション・サポートをデプロイメント環境に提供するかを指定する必要があります。

このタスクについて

以下のステップは、お客様の設計が IBM 提供のパターンのいずれにも一致しないか、または既存のデプロイメント環境を拡張する場合に使用します。すべての複雑さを最小限に抑えるために、一度にデプロイメント環境の 1 つの部分の追加、構成、および検証を行うだけになるように、反復手法の使用を検討してください。

手順

1. デプロイメント環境をサポートするためのデータベース製品を選択します。

DB2 データベース (DB2 for i5/OS、DB2 for IBMi、および DB2 for z/OS を含む) を使用するシステムの場合、メッセージング・エンジンと Common Event Infrastructure (CEI) 用のデータベースとテーブルを自動的に作成することはできません。これらのシステム用のデータベースを作成するには、データベース定義スクリプトを正常に実行するために必要な十分な権限があることを確認します。

重要: 将来の計画に、この環境をデプロイメント・マネージャー・セルに統合することが含まれている場合は、リモート・アクセスをサポートするデータベースとデータベース・ドライバーを使用していることを必ず確認してください。これらのタイプの製品の例としては、Derby Network と Java Toolbox JDBC があります。

2. データベース表の作成方法を決定します。

製品のインストール時にテーブルを作成するか、製品のインストール・プロセスにテーブルを作成するためのスクリプトを作成させるか、またはこの手順を実行するスクリプトを自分で作成します。

3. このデプロイメント環境にデプロイするアプリケーションを分析して、それらのアプリケーションのサポートに必要なクラスターを決定します。
4. デプロイメント環境の物理レイアウトを設計します。
5. 各サーバーを、設計で指定した機能を提供するクラスターのメンバーとしてマップします。

デプロイメント環境により提供される機能、およびどのノードが各クラスターに関連するかを決定します。

6. クライアントにデプロイメント環境内のアプリケーションにアクセスさせる方法を決定します。

アクセスの方法には、ユーザーのニーズに基づき、Web サービス (SOAP/HTTP と SOAP/JMS)、同期または非同期の Service Component Architecture (SCA) 要求、Java Message Service (JMS)、MQ (JMS またはネイティブ)、またはアダプターを介してなど多数の方法があります。これらの選択は、インストールする必要のある他のソフトウェアとリソースに影響を与えます。

7. アプリケーションで必要となるリソースにアプリケーションがアクセスする方法を決定します。

アクセスの方法には、ユーザーのニーズに基づき、Web サービス (SOAP/HTTP と SOAP/JMS)、同期または非同期の Service Component Architecture (SCA) 要求、Java Message Service (JMS)、MQ (JMS またはネイティブ)、またはアダプターを介してなど多数の方法があります。これらの選択は、インストールする必要のある他のソフトウェアとリソースに影響を与えます。

8. ソフトウェアのインストール方法、サーバーの作成方法、および作成したサーバーの構成方法を決定します。

制約事項: 単一のセル内のカスタム・デプロイメント環境の場合、インストーラーまたはプロファイル管理ツールを使用してサーバーを作成することはできません。

9. インストールを完了するために必要な権限を持つすべてのユーザー ID を特定または作成します。
10. オプション: サイト・ポリシーによってデータベースの作成と中央の部門へのアクセスが制限される場合は、担当のデータベース管理者に連絡してください。

重要: 将来の計画に、この環境をデプロイメント・マネージャー・セルに統合することが含まれている場合は、リモート・アクセスをサポートするデータベースとデータベース・ドライバーを使用していることを必ず確認してください。これらのタイプの製品の例としては、Derby Network と Java Toolbox JDBC があります。

11. WebSphere Integration Developer および WebSphere Process Server のインストーラーを、開発コミュニティへの影響を最小限に抑えるように調整してスケジュールします。

WebSphere Integration Developer のインストールについては、IBM WebSphere Integration Developer 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

デプロイメント環境をインストールします。

関連概念

112 ページの『サーバー』

サーバーによって、WebSphere Process Server のコア機能が提供されます。プロセス・サーバーでは、アプリケーション・サーバーの機能が拡大または拡張されて、Service Component Architecture (SCA) モジュールが処理されます。他のサーバー (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) は、プロセス・サーバーの管理に使用されます。

115 ページの『デプロイメント・マネージャー』

デプロイメント・マネージャーとは、他のサーバーの論理グループ (セル) の操作を管理するサーバーです。デプロイメント・マネージャーは、サーバーおよびクラスターを管理するための中央の場所になっています。

118 ページの『管理対象サーバー』

管理対象サーバーとは、管理対象ノード内に構成されるサーバーのことです。管理対象サーバーにより、アプリケーションが実行されるデプロイメント環境内に特定のリソースが提供されます。

118 ページの『デプロイメント環境のクラスター』

クラスターを使用すると、アプリケーションの能力と可用性が単一サーバーの場合よりも高まります。



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。

11 ページの『WebSphere Process Server と他の WebSphere Application Server 製品との間のインターオペラビリティの計画』

ソフトウェア環境を分析するときは、デプロイメント環境内に存在するさまざまなソフトウェア・レベル間で要求を受け渡すことができるかどうかを把握する必要があります。

関連タスク

97 ページの『第 5 章 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。

9 ページの『使用可能なリソースの明確化』

資産を明確化し、既に使用可能なリソースを最大限に利用するとともに、購買の決定の通知を受け取ります。

12 ページの『インストールする製品の決定』

デプロイメント環境の設計には、必要になる可能性があるソフトウェア製品の数とタイプの判断が含まれます。製品の要件は、お客様のニーズに基づき、その環境に関連するコンピューター LPAR システムによって異なる可能性があります。デプロイメント環境のすべてのサーバーにそれぞれ WebSphere Process Server ライセンスが必要なわけではありません。

JCA 認証別名ユーザー ID への表の特権の付与

関連情報

 Network Deployment のインストールの計画

 概要: クラスタ

 Business Process Choreographer の構成

プロファイル

プロファイルでは、個別のコマンド・ファイル、構成ファイル、ログ・ファイルを持つ固有のランタイム環境を定義します。プロファイルでは、WebSphere Process Server システムにスタンドアロン・サーバー、デプロイメント・マネージャー、および管理対象ノードの 3 つのタイプの環境を定義します。

プロファイルを使用すると、WebSphere Process Server バイナリー・ファイルの複数のコピーをインストールしなくても、1 つのシステムに複数のランタイム環境を保持することができます。

WebSphere Application Server for z/OS をインストールする場合は、プロファイルを作成する WebSphere Customization Tools アプリケーションを使用してこの製品を構成します。WebSphere Process Server のインストール後、プロファイルを拡張して、ノードを WebSphere Process Server のデプロイメント・マネージャーまたはスタンドアロン・サーバーのいずれかとして定義する必要があります。デプロイメント・マネージャーを作成したら、次に 1 つ以上の管理対象ノードを作成できます。これにより、各ノードのランタイム環境を定義するプロファイルが自動的に作成されます。

注: 分散プラットフォームでは、各プロファイルには固有の名前があります。z/OS プラットフォームでは、すべてのプロファイルに「default」という名前が付けられます。z/OS ではプロファイルを名前変更、編集、コピー、または削除することはできません。

プロファイル・ディレクトリー

システム内の各プロファイルには、そのプロファイルのすべてのファイルを格納する固有のディレクトリーがあります。プロファイルの作成時に、プロファイル・ディレクトリーの場所を指定します。デフォルトでは、この場所は WebSphere Process Server がインストールされたディレクトリーの profiles ディレクトリーになります。以下に例を示します。installation_file_system_root/AppServer/profiles/default

関連概念

113 ページの『スタンドアロン・サーバー』

スタンドアロン・サーバーは、Service Component Architecture (SCA) モジュールを 1 つのサーバー・プロセスにデプロイするための環境を提供します。このサーバー・プロセスには、管理コンソール、デプロイメント・ターゲット、メッセージング・サポート、ビジネス・ルール・マネージャー、および Common Event Infrastructure サーバーが含まれます (ほかのものが含まれる場合もあります)。

115 ページの『デプロイメント・マネージャー』

デプロイメント・マネージャーとは、他のサーバーの論理グループ (セル) の操作を管理するサーバーです。デプロイメント・マネージャーは、サーバーおよびクラスターを管理するための中央の場所になっています。

116 ページの『管理対象ノード』

管理対象ノードとは、デプロイメント・マネージャーにフェデレートされているノードのことです。このノードにはノード・エージェントが含まれ、さらに管理対象サーバーを含めることができます。管理対象ノードでは、管理対象サーバーを構成して実行できます。

サーバー

サーバーによって、WebSphere Process Server のコア機能が提供されます。プロセス・サーバーでは、アプリケーション・サーバーの機能が拡大または拡張されて、Service Component Architecture (SCA) モジュールが処理されます。他のサーバー (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) は、プロセス・サーバーの管理に使用されます。

プロセス・サーバーは、スタンドアロン・サーバー または管理対象サーバー のいずれかです。管理対象サーバーは、必要に応じてクラスター のメンバーになることが可能です。管理対象サーバー、サーバーのクラスター、および他のミドルウェアの集合を、デプロイメント環境 と呼びます。デプロイメント環境では、管理対象サーバーまたはクラスターが、それぞれデプロイメント環境内の特定の機能 (例えば、宛先ホスト、アプリケーション・モジュール・ホスト、または Common Event Infrastructure サーバー) に構成されます。スタンドアロン・サーバーは、必要なすべての機能を提供するように構成されます。

サーバーによって、Service Component Architecture (SCA) モジュール、それらのモジュールで使用されるリソース (データ・ソース、アクティベーション・スペック、および JMS 宛先)、および IBM 提供のリソース (メッセージ宛先、Business Process Choreographer Container、および Common Event Infrastructure サーバー) にランタイム環境が提供されます。

ノード・エージェント とは、システムに対するノードを表し、そのノード上のサーバーを管理する管理エージェントのことです。ノード・エージェントによって、ホスト・システム上のサーバーがモニターされ、管理要求がサーバーに送付されます。ノード・エージェントは、ノードがデプロイメント・マネージャーに統合されると作成されます。

デプロイメント・マネージャー とは、複数のサーバーとクラスターに一元管理ビューを提供する管理エージェントのことです。

スタンドアロン・サーバーはスタンドアロン・プロファイルによって定義され、デプロイメント・マネージャーはデプロイメント・マネージャー・プロファイルによって定義され、管理対象サーバーは管理対象ノード内に作成され、管理対象ノードはカスタム・プロファイルによって定義されます。

スタンドアロン・サーバー

スタンドアロン・サーバーは、Service Component Architecture (SCA) モジュールを 1 つのサーバー・プロセスにデプロイするための環境を提供します。このサーバー・プロセスには、管理コンソール、デプロイメント・ターゲット、メッセージング・サポート、ビジネス・ルール・マネージャー、および Common Event Infrastructure サーバーが含まれます (ほかのものが含まれる場合もあります)。

ユーザー独自のソリュ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は可能ですが、スタンドアロン・サーバーでは、実稼働環境で一般的に必要とされる容量、スケーラビリティ、または頑強性を提供できません。実稼働環境では、Network Deployment 環境を使用する方が適切です。

最初はスタンドアロン・サーバーから始めて、後でそれを Network Deployment 環境に取り込むことは可能です。このためには、スタンドアロン・サーバーをデプロイメント・マネージャー・セルに統合しますが、このセルにその他のノードが統合されていないことが前提になります。1 つのセルに複数のスタンドアロン・サーバーを統合することはできません。スタンドアロン・サーバーを統合するには、デプロイメント・マネージャーの管理コンソールまたは addNode コマンドを使用します。addNode コマンドを使用してスタンドアロン・サーバーを統合する場合は、スタンドアロン・サーバーが稼働してはなりません。

スタンドアロン・サーバーは、スタンドアロン・サーバー・プロファイルで定義されています。

関連概念

111 ページの『プロファイル』

プロファイルでは、個別のコマンド・ファイル、構成ファイル、ログ・ファイルを持つ固有のランタイム環境を定義します。プロファイルでは、WebSphere Process Server システムにスタンドアロン・サーバー、デプロイメント・マネージャー、および管理対象ノードの 3 つのタイプの環境を定義します。

 メッセージング・エンジン

 データ・ソース

 WebSphere Process Server 用のサービス統合バス

サービス統合バスとは、同期および非同期メッセージングによってサービス統合をサポートする、管理された通信メカニズムです。バスは、バス・リソースを管理する相互接続メッセージング・エンジンで構成されます。サービス統合バスは、WebSphere Process Server の基盤となる WebSphere Application Server テクノロジーの 1 つです。

Network Deployment

Network Deployment という用語は、単一のデプロイメント・マネージャーによって管理されている、1 つ以上のマシンに存在するサーバーの論理グループから成る、WebSphere Process Server 環境構成のことを指します。

Network Deployment では、容量、スケーラビリティ、および一般に実稼働環境に要求される頑強性が提供されます。Network Deployment では、サーバーのグループが共同作業を行うことにより、ワークロード・バランスングおよびフェイルオーバーの機能を提供できます。各サーバーは、単一の管理コンソールを使用して一元管理されます。

WebSphere Process Server 内の Network Deployment は、WebSphere Application Server Network Deployment に実装された Network Deployment 機能をベースにしています。WebSphere Application Server Network Deployment での Network Deployment に詳しい場合は、その概念は同じなので理解しやすいはずですが。WebSphere Process Server では、デプロイメント環境の概念が Network Deployment に加わります。

Network Deployment に関して読んでおく必要があることは、WebSphere Application Server Network Deployment をアップグレードするのか、それとも WebSphere Application Server Network Deployment に関する経験がない状態で WebSphere Process Server を実装するのかによって異なります。

WebSphere Application Server Network Deployment のアップグレード

WebSphere Application Server Network Deployment では、その名前が示すように、アプリケーションの Network Deployment がサポートされます。WebSphere Process Server でアップグレードする WebSphere Application Server Network Deployment のインストール済み環境が既に存在する場合は、Network Deployment の概念に詳しいことでしょう。それぞれにデプロイメント・マネージャーと管理対象ノードを持つ

Network Deployment セルが、おそらく 1 つ以上存在していることでしょう。WebSphere Process Server のプロファイル管理ツールを使用して、それらのプロファイルを拡張して、WebSphere Process Server をサポートすることができます。拡張の後、サーバーは引き続きアプリケーション・サーバーとして機能しますが、それらはService Component Architecture (SCA) モジュールをサポートすることも可能です。

WebSphere Process Server の Network Deployment の実装

Network Deployment では、WebSphere Process Server を 1 つ以上のホスト・システムにインストールした後、デプロイメント環境を作成します。IBM では、クラスター、サーバー、およびService Component Architecture (SCA) モジュールをホストするために必要なミドルウェアを構成するために役立ついくつかのデプロイメント環境パターンを提供しています。

関連概念

73 ページの『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。

関連情報

 [WebSphere Application Server Network Deployment と単一サーバー \(すべてのオペレーティング・システム\) のインフォメーション・センター](#)

デプロイメント・マネージャー

デプロイメント・マネージャーとは、他のサーバーの論理グループ (セル) の操作を管理するサーバーです。デプロイメント・マネージャーは、サーバーおよびクラスターを管理するための中央の場所になっています。

セル内のサーバーおよびクラスターを管理するには、デプロイメント・マネージャーの管理コンソールを使用します。管理作業には、サーバーおよびクラスターの構成、クラスターへのサーバーの追加、サーバーおよびクラスターの開始および停止、およびサーバーおよびクラスターへの Service Component Architecture (SCA) モジュールのデプロイが含まれます。

デプロイメント・マネージャーは一種のサーバーですが、デプロイメント・マネージャー自体にモジュールをデプロイすることはできません。

関連概念

71 ページの『第 3 章 セル内での複数のプラットフォームの使用』
綿密に計画することによって、分散 プラットフォーム、および z/OS オペレーティング・システムの各プラットフォームのノードが含まれたデプロイメント・マネージャー・セルを作成することができます。

111 ページの『プロファイル』
プロファイルでは、個別のコマンド・ファイル、構成ファイル、ログ・ファイルを持つ固有のランタイム環境を定義します。プロファイルでは、WebSphere Process Server システムにスタンドアロン・サーバー、デプロイメント・マネージャー、および管理対象ノードの 3 つのタイプの環境を定義します。

管理対象ノード

管理対象ノード とは、デプロイメント・マネージャーにフェデレートされているノードのことです。このノードにはノード・エージェントが含まれ、さらに管理対象サーバーを含めることができます。管理対象ノードでは、管理対象サーバーを構成して実行できます。

管理対象ノード上で構成されているサーバーが、デプロイメント環境のリソースを形成しています。これらのサーバーの作成、構成、開始、停止、管理、および削除は、デプロイメント・マネージャーの管理コンソールを使用して行われます。

管理対象ノードには、ノード上のすべてのサーバーを管理するノード・エージェントがあります。

ノードがフェデレートされると、ノード・エージェント・プロセスが自動的に作成されます。プロファイルの構成を管理できるようにするには、このノード・エージェントが稼働していなければなりません。例えば、以下の作業を実行する場合などです。

- サーバー・プロセスの開始および停止
- デプロイメント・マネージャー上の構成データとノード上のコピーとの同期化

ただし、アプリケーションがノード内のリソースを実行または構成するようにしたい場合は、ノード・エージェントが稼働している必要はありません。

管理対象ノードには 1 つ以上のサーバーを配置できます。それらのサーバーは、デプロイメント・マネージャーによって管理されます。管理対象ノード内のサーバーにソリューションをデプロイできますが、その管理対象ノードにはサンプル・アプリケーション・ギャラリーは含まれません。

関連概念

71 ページの『第 3 章 セル内での複数のプラットフォームの使用』
綿密に計画することによって、分散 プラットフォーム、および z/OS オペレーティング・システムの各プラットフォームのノードが含まれたデプロイメント・マネージャー・セルを作成することができます。

111 ページの『プロファイル』

プロファイルでは、個別のコマンド・ファイル、構成ファイル、ログ・ファイルを持つ固有のランタイム環境を定義します。プロファイルでは、WebSphere Process Server システムにスタンドアロン・サーバー、デプロイメント・マネージャー、および管理対象ノードの 3 つのタイプの環境を定義します。

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。

デプロイメント環境の計画では、容量、可用性、スケーラビリティ、およびフェイルオーバー・サポートに対するビジネス・ニーズを満たすことができるように、デプロイメント環境の物理的なレイアウト (トポロジー) を設計することが必要になります。設計の重要な部分としては、デプロイメント環境を構成するハードウェア上のサーバーの数と相対的な配置があります。

スタンドアロン環境

Service Component Architecture (SCA) モジュールをスタンドアロン・サーバー にデプロイすることができます。この環境はセットアップが最も容易ですが、スタンドアロン・サーバーは他のサーバーへ接続されないため、その容量は同一コンピューター・システム上のリソースに限られ、フェイルオーバー・サポートも組み込まれません。

スタンドアロン・サーバーで提供される容量、スケーラビリティ、可用性、またはフェイルオーバー・サポートを超えるものが必要な場合は、相互接続サーバーのデプロイメント環境を検討する必要があります。

相互接続サーバー

デプロイメント環境とは、以下のような WebSphere Process Server のアプリケーション・コンポーネントをサポートする相互接続サーバーの集合のことです。

- Business Process Choreographer
- ビジネス・ルール
- メディエーション
- リレーションシップ

この環境では、WebSphere Enterprise Service Bus と WebSphere Application Server ベースのサーバーもサポートされます。

デプロイメント環境内のサーバーは、1 つ以上のホスト・システム上で稼働することができます。複数のサーバーを、ロード・バランシングとフェイルオーバーをサポートするためにクラスター にグループ化することができます。

スタンドアロン・サーバーでは提供できないパフォーマンス、可用性、スケーラビリティ、分離機能、セキュリティ、および安定度の特性に加えて、相互接続サーバーまたはクラスターのデプロイメント環境には、集中化されたデプロイメント・マネージャー からすべてのサーバーまたはクラスターを管理できるというさらなる利点もあります。

デプロイメント環境を作成するタイミング

使用するホスト・システム上にソフトウェアをインストールします。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を使用して、デプロイメント・マネージャーとカスタム・プロファイルを作成します。次に、デプロイメント・マネージャーの管理コンソールを使用して、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いくつかの面は、管理コンソールを使用して引き続き管理することができます。(例えば、デプロイメント環境にノードを追加できます)。

関連タスク



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。

管理対象サーバー

管理対象サーバーとは、管理対象ノード内に構成されるサーバーのことです。管理対象サーバーにより、アプリケーションが実行されるデプロイメント環境内に特定のリソースが提供されます。

管理対象サーバーは、必要に応じてクラスターのメンバーになることが可能です。堅固で実動規模のプロセス・サーバーを提供するには、管理対象サーバーのクラスターを含むデプロイメント環境を構成します。

サーバーとクラスターの構成および管理には、デプロイメント・マネージャーの管理コンソールを使用します。

デプロイメント環境のクラスター

クラスターを使用すると、アプリケーションの能力と可用性が単一サーバーの場合よりも高まります。

クラスター とは、高可用性およびワークロード・バランシングをアプリケーションに提供する管理対象サーバーの集合のことです。クラスターのメンバーは、各種ホスト上にあるサーバーの場合と、同じホスト (同じノード) 上にあるサーバー場合があります。高可用性とワークロード・バランシングを最大限に達成するには、各クラスター・メンバーを異なるホスト・マシンに配置します。

クラスター環境には、以下の利点があります。

- ワークロード・バランシング: 複数のサーバー上でアプリケーション・イメージを実行することにより、クラスターはクラスター内のサーバー全体のアプリケーション・ワークロードのバランスを取ります。
- アプリケーションの処理能力: サーバーのハードウェアをアプリケーションをサポートするクラスター・メンバーとして構成することにより、アプリケーションの処理能力を増強できます。
- アプリケーションの可用性: サーバーに障害が発生した場合、アプリケーションの処理はクラスター内の他のサーバー上で続行されます。これにより、リカバリー作業をアプリケーション・ユーザーに影響を与えることなく進めることができます。
- 保守容易性: アプリケーションの処理を停止することなく、計画された保守のためにサーバーを停止できます。
- 柔軟性: デプロイメント・マネージャーの管理コンソールを使用することにより、必要に応じて容量を追加または除去することができます。

z/OS 環境は、元々クラスター化されており、分離クラスター・サーバーが持つ複雑さはまったくなく高度に拡張が容易な環境を提供できます。分離クラスター・サーバーは、マルチシステム・ノードの可用性を高めるために使用できます。

デプロイメント環境パターンの選択

IBM 提供のいずれかのトポロジー・パターンを選択するか、独自のカスタム・デプロイメント環境を作成することによって、デプロイメント環境を構成できます。このトピック・セクションでは、IBM 提供の使用可能なトポロジー・パターンのリストを示して説明すると共に、トポロジーを選択する場合の考慮事項を示します。

始める前に

以下のための情報に精通している必要があります。

- ビジネス要件の評価
- 使用可能なリソースの明確化
- トポロジーを選択するための考慮事項 の検討

このタスクについて

デプロイメント環境の設計を完了し、さまざまな製品ウィザードでサポートされる IBM 提供のパターンがニーズを満たすかどうかを判別する必要があります。

重要: デプロイメント環境で z/OS システムまたはクラスターを使用する場合は、サーバーまたはクラスターが提供する機能を判別してください。同じクラスターで z/OS システムと他のシステムを混在させることはできないため、設計時にはこの点を考慮に入れる必要があります。

手順

1. ビジネスのニーズに最適な IBM 提供のパターンを判別します

使用可能なデプロイメント環境パターンについては、『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。

2. オプション: メディエーション・サービスのみを提供する必要がある場合は、WebSphere Process Server の代わりに Enterprise Service Bus をインストールします。
3. IBM 提供のいずれのパターンもビジネスのニーズを満たさない場合は、カスタム・デプロイメント環境を実装できます。

注: カスタム・パターンを実装するには、デプロイメント環境の動作に関する詳細な知識を持ち、サーバーおよびクラスターの正しい構成方法を理解している必要があります。

次のタスク

製品をインストールおよび構成します。

関連概念

73 ページの『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こと、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。

92 ページの『トポロジーを選択するための考慮事項』

使用するデプロイメント環境に適したトポロジーの選択は、いくつかの要因に基づいて行います。

143 ページの『エラー処理方針とソリューション・リカバリー』

WebSphere Process Serverには、リカバリーのために使用できるエラー処理機能とツールが組み込まれています。

第 6 章 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。

始める前に

- トポロジーを計画し、以下に関する決定事項を記録します。
 - 必要なサーバーおよびクラスター
 - 必要なデータベースの数
 - どのデータベース表がどのデータベースに属するか
 - 必要なユーザー ID および認証ロール
 - デプロイメント環境に関係する各クラスターが提供する機能
 - デプロイメント環境の実装に使用する方法
- 製品のインストール先システムがハードウェア要件およびソフトウェア要件を満たすことを確認します。
- インストールできるようにオペレーティング・システムを準備します。
- トポロジーに関係するすべてのサーバーが、IP アドレスおよびドメイン・ネーム・サーバー (DNS) 名で位置指定できることを確認します。
- すべてのシステムでディレクトリーおよびファイルを作成できる適切な権限を持つユーザー ID を用意します。
- 他の製品と共存して所定の冗長度を確保するために必要な準備を行います。

このタスクについて

以上でデプロイメント環境の計画が完了し、前提条件タスクをすべて実行しました。この後は、設計で必要になるサーバーおよびクラスターをインストールして構成します。どのような方法でデプロイメント環境を実装するかにかかわらず、以下のステップに従って設計の単一のセルを作成します。

注: ここに示す手順には、デプロイメント環境の実装に必要なすべてのステップを記載しています。インストール方法によっては、手順が多少前後する場合があります。

手順

1. デプロイメント環境に関連するすべてのシステムに製品バイナリーをインストールし、ソフトウェアが正常にインストールされたことを確認します。
2. オプション: データベース構成を設計します。

データベース構成を設計する方法として、データベース設計ツール (DDT) の使用を選択できます。プロファイル作成時より前にデータベース構成を設計すると、プロファイル作成プロセスが簡単になる場合があります。構成プロセスの早い時点でデータベース構成を設計する場合、プロファイル作成時にデータベース設計文書をインポートできます。

データベース構成を作成する方法として DDT を使用するかどうかによって設計手法が決まるため、ソリューション実施チームのメンバーと協力して検討する必要があります。

3. デプロイメント・マネージャーを作成します。

WebSphere Process Server は、デプロイメント・マネージャーを作成するためのいくつかの方法 (プロファイル管理ツールの使用や `manageprofiles` コマンド行ユーティリティの使用など) を提供します。デプロイメント・マネージャー・プロファイルの作成にどの方法を選択するかは、好みの問題です。それぞれの方法については、『プロファイルの作成』のセクションを参照してください。

4. デプロイメント・マネージャーを始動します。
5. 必要な数の管理対象ノードを作成します。
6. ステップ 5 のノードを、ステップ 3 で作成したデプロイメント・マネージャーに統合します。
7. セルを構成します。

「デプロイメント環境構成」ウィザードを使用してセルを構成できます。このウィザードを使用すると、パターンに基づいてデプロイメント環境を作成できます。デプロイメント環境パターンは、最も一般的に使用されるビジネス・インテグレーション・トポロジーのルール・ベースの構成です。パターンは環境構成用のテンプレートを提供します。デプロイメント環境パターンは、共に動作するコンポーネント構成を含む既知のテスト済み推奨トポロジーを表しており、これらのパターンを使用すると、信頼できるデプロイメント環境機能が確実に得られます。

重要: スクリプトを使用してデプロイメント環境を作成する場合、デプロイメント環境によっては構成処理に時間がかかることがあります。プロセスがタイムアウトになるのを防ぐため、デプロイメント・マネージャーの SOAP 要求のタイムアウト値を大きい値 (例えば 1800 秒) に設定します。SOAP 要求タイムアウトの詳細については、WebSphere Application Server インフォメーション・センターの『Java Management Extensions コネクタ・プロパティ』を参照してください。デフォルトのタイムアウト値を変更するには、任意の ASCII テキスト・エディターでファイル `$WAS_HOME/profiles/<profile name>/properties/soap.client.props` を開き、次の行を見つけます (ここでは、デフォルト値が 180 秒になっています)。

```
com.ibm.SOAP.requestTimeout=180
```

デフォルトを変更する必要がある場合は、この行を編集して、タイムアウトをより状況に適した値に設定することができます。上記の値を 0 に設定すると、タイムアウト・チェックが完全に無効になります。

セルを構成する場合、設計で定義した機能を実行するクラスターの作成後、そのクラスターにメンバーを追加する必要があります。

パターンに基づくデプロイメント環境を実装する設計の場合は、必要なすべてのクラスターの作成とクラスター・メンバーの定義が自動的に行われ、必要な機能がすべて提供されます。この機能には、選択したデプロイメント環境パタ

ーンに応じ、アプリケーション・デプロイメント、メッセージング・サポート、およびインフラストラクチャー・サポートのための各クラスターが含まれます。

カスタムのデプロイメント環境を実装する設計の場合は、必要な機能を提供するためのクラスターをすべて独自に作成する必要があります。この機能には、アプリケーション・デプロイメントのメッセージング・サポート、アプリケーション・サポート、および Common Event Infrastructure サポートがあります。

8. テーブルの作成延期を選択した場合は、トポロジで必要になるデータベースまたはデータベース表を構成します。

構成作業は、延期オプションの選択によって生成された各種スクリプトを実行することで進めます。

- a. 共通のデータベース表を構成します。これらのデータベース表は、共通データベース内の表です。詳しくは、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・グループの作成および Ddl2Pds.sh スクリプトを参照してください。
- b. メッセージング・エンジンのデータベース表を構成します。これらのデータベース表は、共通データベース内の表です。
- c. オプション: Business Process Choreographer のデータベース表を構成します。

システムでビジネス・プロセスもヒューマン・タスクも使用しない場合は、このステップを省略してください。このテーブルは、Business Process Choreographer で使用するよう構成したデータベース (デフォルトの名前は BPEDB) に配置するものです。

Business Process Choreographer Explorer レポート機能を使用する場合は、Business Process Choreographer Explorer レポート・データベース (OBSRVDB) も構成する必要があります。

- d. エンタープライズ・サービス・バスのロギング・メディエーション・データベース表を作成します。これらのデータベース表は、共通データベース内の表です。
 - e. Common Event Infrastructure データベースを構成します。
9. WebSphere Application Server にプロキシ・サーバーを作成します。プロキシ・サーバーは、作業を実行するコンテンツ・サーバーに対して HTTP 要求をルーティングします。

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。プロキシ・サーバーを使用する利点は、WebSphere Application Server と統合されているため、使用と保守が簡単なことです。

重要: 複数のクラスター・メンバー間で HTTP 要求のロード・バランシングを行う場合は、プロキシ・サーバー (または代替のルーティング・サーバー) が必要です。このサーバーによって、クライアントはこのトポロジ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。

10. テスト・アプリケーションをインストールして実行し、インストールを検証します。

次のタスク

- 必要に応じて別のセルを作成します。
- このデプロイメント環境で実行するアプリケーションをデプロイします。

関連概念

73 ページの『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。

117 ページの『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。

118 ページの『デプロイメント環境のクラスター』

クラスターを使用すると、アプリケーションの能力と可用性が単一サーバーの場合よりも高まります。

112 ページの『サーバー』

サーバーによって、WebSphere Process Server のコア機能が提供されます。プロセス・サーバーでは、アプリケーション・サーバーの機能が拡大または拡張されて、Service Component Architecture (SCA) モジュールが処理されます。他のサーバー (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) は、プロセス・サーバーの管理に使用されます。

115 ページの『デプロイメント・マネージャー』

デプロイメント・マネージャーとは、他のサーバーの論理グループ (セル) の操作を管理するサーバーです。デプロイメント・マネージャーは、サーバーおよびクラスターを管理するための中央の場所になっています。

関連タスク



デプロイメント・マネージャーの停止と再始動

デプロイメント・マネージャーに対して任意の構成変更を行った場合は、デプロイメント・マネージャーを停止後に再始動して、それらの変更を有効にする必要があります。



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

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。



データベースの作成および構成



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。

関連情報



Business Process Choreographer の構成

- ☞ プロキシ・サーバーの作成
- ☞ Web サーバーとの通信
- ☞ IBM HTTP Server のインストール
- ☞ メッセージング・エンジン
- ☞ wsadmin スクリプト・ツールの使用
- ☞ ノード・エージェントの管理
- ☞ クラスターの開始
- ☞ クラスターの停止
- ☞ Java Management Extensions コネクタ・プロパティ

第 7 章 WebSphere Process Server の計画: 追加考慮事項

WebSphere Process Server などのミドルウェアでは、製品をインストールする前に、エンタープライズ情報システム (EIS) の多くの側面 (キャパシティーやセキュリティなど) を評価する必要があります。また、計画性を持ってエラー防止やリカバリーなどの WebSphere Process Server の機能を活用する必要があります。

以下の各質問への回答が、お客様のニーズに合わせてデプロイメント環境を設計するために役立ちます。

- ビジネス目標は何ですか。また、その目標を達成するために、このソフトウェアはどのように役立ちますか。
- どのようなアプリケーションを統合する必要がありますか。
- 重複した情報を除去しますか。
- システム応答時間と可用性の要件は何ですか。
- インストールを完了するためにどのような財務、ハードウェア、ソフトウェア、および人材の各資源が使用可能ですか。
- 他の部門のサービスが必要ですか。
- どのような作業を行う必要がありますか。それらの作業を行うのは誰でしょうか。
- どのような既存のハードウェアがインストールに必要ですか。
- ビジネス要件を達成するために追加のハードウェアが必要ですか。
- 既存のデータベースを使用できますか。それとも新規のデータベースが必要ですか。
- 既存のユーザー ID を WebSphere Process Server コンポーネントで使用できますか。それとも新規 ID が必要ですか。どのような権限が新規 ID に必要ですか。
- 購入可能な製品ライセンスの数を制限する財務上の考慮事項はありますか。
- お客様のシステムの増大度合いはどの程度ですか。例えば、今後、負荷の増加、およびより多数の同時ユーザーに対応することが必要になりますか。追加の要求を満たすために、リソースを今後さらに追加することが必要になりますか。
- お客様のシステムで、日々の要求の変動に対応するために動的にリソースを追加または除去することが必要になりますか。
- お客様のシステムで、負荷または同時ユーザー数の変動に定期的に対応する必要がありますか。

また、お客様の以下の現在の目標についても考慮します。テスト環境または実稼働環境を計画していますか。その環境は小規模ですか。それとも大規模ですか。使用する環境をデフォルト値で素早くセットアップしたいですか。それともカスタマイズしたいですか。このセクションの最後に、お客様が達成しようとするに基づき、さまざまなシナリオの計画のための提案があります。

第 8 章 エラー防止およびリカバリーの計画

システムおよびアプリケーションのエラーの影響を最小化するために、エラー防止およびリカバリーの方法を作成することができます。

『エラー防止およびリカバリーの計画』の各トピックには、インフォメーション・センター・トピック、技術記事および IBM Redbooks® などのさまざまなリソースへのリンクが含まれています。これらには WebSphere のシステム・リカバリー機能を活用するように設計された開発プロセスおよびシステム構成パターンに関する詳細な情報が提供されています。

エラー防止とリカバリーの概要

エラー防止とリカバリーの情報では、システム障害を引き起こす問題を回避する方法について説明し、通常の状態と異常な状態の両方で発生する可能性があるシステム障害からリカバリーする方法について、情報を提供しています。

WebSphere Process Server は、ビジネス・プロセス管理 (BPM) ソリューションとサービス指向アーキテクチャー (SOA) ソリューションの実行および管理を可能にするために最適化されたミドルウェア・サーバーです。WebSphere Process Server は WebSphere Application Server の基盤となる機能に基づいて構築されています。

ミドルウェア・システムはさまざまな条件下で実行されますが、従来、それらのすべてが「良好なパス (good path)」条件になっているわけではありません。WebSphere Process Server の主要な機能の多くは、普通に見える動作から発生する可能性がある、不確実さに対処するためのものです。

想定および予期

『エラー防止およびリカバリーの計画』セクションの記載に従ってシステム障害およびリカバリーの情報を使用する前に、以下の想定リストを読んでください。

- WebSphere Process Server およびその構築に使用されている基本のアーキテクチャー原則、ならびにそれが実行する基本的な種類のアプリケーションについて熟知している。
- 統合プロジェクトの計画および実施方法などの、統合プロジェクトについての基礎知識がある。
- 特に指定しない限り、システム障害およびリカバリーについての情報は、WebSphere Process Server のバージョン 6.1.0 以降に関するものである。

注: 『エラー防止およびリカバリーの計画』セクションに含まれている情報は、リモート・メッセージングおよびリモート・サポートのパターンを想定しており、WebSphere Process Server 用に 1 つ、メッセージング・エンジンおよび CEI イベント・サーバー用にそれぞれ 1 つずつの 3 つの別個のクラスターで構成されています。

関連概念

73 ページの『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。

関連資料

152 ページの『ピア・リカバリー』

ピア・リカバリーは、同じクラスター内の別のメンバーによって実行されるリカバリーであり、手動でも自動でも開始できます。ピア・リカバリー処理 (自動ピア・リカバリーまたは手動ピア・リカバリー) は、WebSphere の高可用性環境と緊密に連携しています。

エラー防止の計画

あらゆる IT 処理の場合と同様、極端なシチュエーションに対して計画を行って実行すれば、正常にリカバリーする確率は高くなります。

システムおよびアプリケーションのリカバリーを準備することに関しては、必須の考慮事項が数多くあります。これらの考慮事項は、次の 2 つのカテゴリーに分類できます。

- アプリケーション設計の一部としてのエラー防止手段
- 開発プロセスの一部としてのエラー防止手段

アプリケーション設計の一部としてのエラー防止

アプリケーション設計の一部としてエラー防止手段を組み込むことは、特定の設計技法を実装し、製品の機能を使用して、システムおよびアプリケーションのエラーを防止することを意味します。

アプリケーションを正しく構築するためには、アーキテクチャーおよび設計のガイドラインと適切な標準と、その組み合わせとなるレビューおよびチェックポイントが完備された強力なシステム・ガバナンスが不可欠です。

アプリケーション設計の一部としてのエラー防止手段には、以下の要素が含まれています。

- 例外および障害に対する設計上の考慮事項の実装
- 既存の WebSphere Process Server エラー処理機能とツールを使用するエラー処理方針の実装
- 接続グループの作成とモジュール・アプリケーション設計技法の使用

接続グループ

接続グループは、SCA モジュール内に見られる特定の動作パターンを表します。

システムで考えられる要求ソースを表す接続グループを作成します。

接続グループでは以下を行います。

- インバウンド・データを取得するためのすべてのロジックを 1 つのモジュールに配置します。

これはアウトバウンド・データが、外部システムまたはレガシー・システムに送信される場合にも該当します。

- データを接続および変換するためのすべてのロジックを 1 つのモジュールに配置します。

他のすべてのモジュールもインターフェースの標準セットを使用できるようになったため、さらなる変換を考慮する必要はありません。

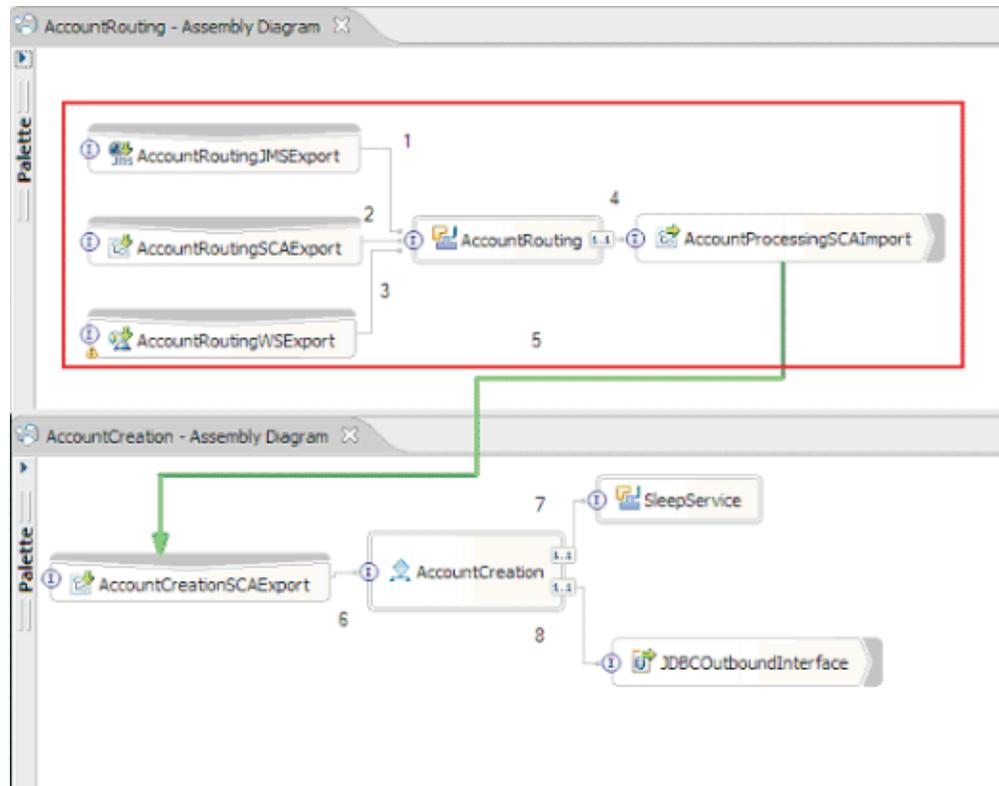
接続グループには、長期実行ビジネス・プロセスやビジネス・ステート・マシンのようなステートフル・コンポーネント・タイプが含まれません。これらの接続グループにより、特定のエンドポイントの統合要件をカプセル化して分離できます。一般に、WebSphere ESB メディエーション・モジュールがこの目的のために使用されるのは、「インフラストラクチャー」関連タスクを実装する便利な方法だからです。

接続グループの概念は、リカバリーの必要がある場合に、システムを休止状態にするための便利な方法として使用することもできます。接続グループ・モジュールはステートレスであるため、モジュールを一時的に停止して、システムが保持するイベントの処理を完了させる間、新規イベントのインバウンド・フローを遮断することができます。

注: インバウンド・イベントのフローを停止する場合、接続モジュールはインバウンドおよびアウトバウンドを同じモジュール内でサポートしてはいけません (同じ EIS システムがインバウンドとアウトバウンドの両方を持っている場合でも該当します)。インバウンドおよびアウトバウンドが同じモジュールでサポートされている場合、アウトバウンドはインバウンドと一緒にオフになります。このため、内部処理の実行が停止することがあります。このケースでは、インバウンドとアウトバウンドを分離することを検討してください。

システムがリカバリーされ、新規の作業を処理できるようになったら、これらのモジュールを再開できます。

以下の画面取りに外観が示されているモジュールは、接続グループの一部と見なされます。



接続グループは、SAP または CICS® などの外部ソースまたは既存のシステムからの入力用に使用できます。または、Web ブラウザー・ベースのクライアントからの新規作業のためにも使用できます。

関連概念

ユース・ケース: 失敗イベントからのデータのリカバリー
 ユース・ケースは、リカバリー・シナリオでのコンテキストとして使用されます。このユース・ケースでのビジネスには、新規アカウントを作成する要求を受け取るアプリケーションがあります。

関連資料

154 ページの『エクスポート・バインディング』
 システムを完全に停止するには、使用可能なエクスポート・バインディングによってサポートされるさまざまなタイプの要求呼び出しを考慮します。

例外および障害に対するアプリケーション設計上の考慮事項

アプリケーション設計では、WebSphere Process Server のエラー処理機能と障害処理機能の利点を活用することを考慮する必要があります。

包括的なエラー処理方法を作成するには、ソリューション設計者は、WebSphere Process Server および WebSphere ESB が宣言済みまたは宣言済みでない例外をどのように表すかを理解する必要があります。

SCA プログラミング・モデルには以下の 2 つのタイプの例外が提供されています。

- サービス・ビジネス例外

サービス・ビジネス例外は、ビジネス・メソッドの機能シグニチャー (WSDL 障害または Java throw) で宣言されるチェック例外です。サービス・ビジネス例外では、アプリケーションまたはサービスによって予期されるエラー状態を識別します。これらの例外は、「チェック例外」と呼ばれることもあります。

例えば、株価サービスでの `InvalidSymbolException` があります。そのような例外は `ServiceBusinessException` によってラップされ、クライアントに戻されます。

- サービス・ランタイム例外

「システム例外」としても知られています。サービス・ランタイム例外は、メソッド・シグニチャーでは宣言されません。一般にこの例外は、Java コンポーネントでの `NullPointerException` などのように、アプリケーションによって予期されていないエラー状態を表します。

これらの例外は、`ServiceRuntimeException` によってラップされ、クライアントに戻されます。クライアントは `ServiceRuntimeException` について問い合わせる原因を判別できます。

注: SCA レベルで動作するとき、これらの例外は障害と呼ばれることもあります。ただし、Java コードを使用するときは、これらは通常は例外と呼ばれます。

コンポーネントから `ServiceRuntimeException` が throw されると、現在のトランザクションがロールバックされます。

サービス・ビジネス例外の処理:

サービス・ビジネス例外は、アプリケーションまたはサービスによって予期される既知および宣言済みの例外を表します。

サービス・ビジネス例外は、サービス・インターフェースで定義されます。

コンポーネント開発者は、throw される可能性がある例外を宣言することに注意を払い、消費側のサービスでそれらの例外を処理できるようにしてください。例えば、銀行用アプリケーションのビジネス障害には、「無効な口座番号」または「資金不足」がビジネス例外として含まれる場合があります。したがって、サービスを呼び出すアプリケーションには、無効な口座番号が渡された場合や、\$100 を振り替えようとして口座に \$50 しかなかった場合などの状況を処理するロジックを組み込んでおく必要があります。これらは、呼び出し側のアプリケーションが処理するように設計されたタイプのビジネス・エラーです。WebSphere Process Server のビジネス例外は、catch して適切に処理するクライアントに戻されます。

ビジネス・サービス例外を処理する場合、サービス・コンシューマーは、宣言済みのビジネス例外の場合に以下のアクションのいずれかを実行するようにクライアントを実装します。

1. 例外を catch して、呼び出し側のアプリケーションに適したサービス・ビジネス例外を作成します。

これは、元の例外を新規例外に含める (ラップする) ことを意味する場合があります。これが最も頻繁に行われるのは、呼び出し側のモジュールが、呼び出し先

のサービスと同じビジネス例外を持たない場合です。例外を catch して、呼び出し側のアプリケーションのためにサービス・ビジネス例外を作成するフローの例を以下に示します。

- a. Module A は SBE 「MoneyTransferFailed」を持つ
- b. Module B は SBE 「InsufficientFunds」を持つ
- c. Module A は Module B を呼び出し、「InsufficientFunds」例外を取得する
- d. Module A は、資金の不足という元のエラーを定義するストリングを格納できる場所を持つ、新規例外 「MoneyTransferFailed」を作成する必要がある。

2. 例外を catch して代替ロジックを実行します。

関連概念



ユース・ケース: 失敗イベントからのデータのリカバリー

ユース・ケースは、リカバリー・シナリオでのコンテキストとして使用されます。このユース・ケースでのビジネスには、新規アカウントを作成する要求を受け取るアプリケーションがあります。

サービス・ランタイム例外の処理:

サービス・ランタイム例外は、未宣言の例外です。一般にこの例外は、アプリケーションによって予期されていないエラー状態を表します。

サービス・ランタイム例外を使用して、実行時の予期せぬ状態を知らせます。

コンポーネント開発者は、次の方法でサービス・ランタイム例外を処理できます。

1. それらの例外を catch し、何らかの代替ロジックを実行します。

例えば、あるパートナーが要求を処理できなくても、別のパートナーでは処理できることもあります。

2. 例外を catch し、クライアントに「再 throw」します。
3. その例外をビジネス例外に再マップします。

例えば、パートナーがタイムアウトになると、ビジネス例外が生成される可能性があります。その例外では、ほとんどの要求が処理されたものの、完了していない 1 つの要求が残されており、その要求を後で再試行するか、異なるパラメーターを指定して試行すべきことが示されます。

例外が catch されない場合、例外は現行コンポーネントを呼び出したコンポーネントに渡されます。この呼び出しチェーンは、チェーンの最初の呼び出し元に戻るまで続きます。例えば、Module A が Module B を呼び出し、Module B が Module C を呼び出して、Module C が例外を throw すると、Module B は例外を catch する場合も、またはしない場合もあります。Module B が例外を catch しない場合、例外は Module A まで戻されます。

コンポーネントから `ServiceRuntimeException` が throw されると、現在のトランザクションがロールバックされます。このタイプの例外処理は、チェーン内のすべてのコンポーネントに対して繰り返されます。例えば、Module C から `ServiceRuntimeException` が throw された場合は、そのトランザクションに、ロールバック対象としてのマークが付けられます。次にその例外は Module B に throw され、Module B で例外が catch されず、別のトランザクションが存在する場合

は、そのトランザクションもロールバックされます。コンポーネント開発者は、呼び出しが現行トランザクションと新規トランザクションのどちらで行われるかを制御するサービス品質 (QoS) 修飾子を使用できます。例えば、Module A が Module B を呼び出し、Module B が新しいトランザクションの一部である場合、Module A は Module B からの `ServiceRuntimeException` を "catch" することで、Module A のトランザクションをロールバックすることなく、処理を続行できます。

注: ランタイム例外はインターフェースの一部として宣言されていないため、コンポーネント開発者は、例外の解決を試みて、クライアントがユーザー・インターフェースである場合にランタイム例外がクライアントまで不意に伝搬されるのを防止してください。

ロールバックされるトランザクションの内容は、トランザクションの性質によって異なることに注意してください。例えば、長期実行の BPEL プロセスが、多数の小規模なトランザクションにセグメント化することができます。非同期要求および応答の呼び出しは、自動的にトランザクションから切り離されます (そうしないと、呼び出し側のアプリケーションが長時間にわたって応答を待たなければならなくなります)。

1 つのトランザクションが、(1 つの大きなトランザクションとして扱われるのではなく) 複数の非同期呼び出しに分割される場合、トランザクションの最初の作業は `ServiceRuntimeException` の発生時にロールバックします。ただし、非同期呼び出しの応答が別のトランザクションから送信された場合、その非同期呼び出しの応答の行き先がなくなるため、Failed Event Manager (FEM) でイベントが作成されます。

以下に、`ServiceRuntimeException` の現行サブクラス 4 つを示します。

1. `ServiceExpirationRuntimeException`

この例外を使用して、非同期 SCA メッセージの有効期限が切れたことを示します。有効期限は、サービス参照で `RequestExpiration` 修飾子を使用して設定できます。

2. `ServiceTimeoutRuntimeException`

この例外を使用して、非同期要求への応答を構成された期間内に受信しなかったことを示します。有効期限は、サービス参照で `ResponseExpiration` 修飾子を使用して設定できます。

3. `ServiceUnavailableException`

この例外を使用して、インポートを介して外部サービスを読み出し中に throw された例外が存在することを示します。

4. `ServiceUnwiredReferenceRuntimeException`

この例外を使用して、コンポーネントでのサービス参照が正しくワイヤーされていないことを示します。

関連概念

 ユース・ケース: 失敗イベントからのデータのリカバリー
ユース・ケースは、リカバリー・シナリオでのコンテキストとして使用されます。このユース・ケースでのビジネスには、新規アカウントを作成する要求を受け取るアプリケーションがあります。

関連情報

 修飾子とトランザクションの設定

開発の一部としてのエラー防止

開発プロセスの一部として、エラー防止手段を含めることができます。

開発プロセスの一部としてのエラー防止手段では、プロジェクトをロールアウトするために設置されるガバナンスと開発プロセスに焦点を合わせており、主に、テスト、チューニング、測定、および再テストのアクティビティーが関係しています。

開発プロセスの一部としてのエラー防止手段には、以下の要素が含まれています。

- 総合的なテストによる問題の防止
- 継続的かつ定期的にスケジュールされた環境調整
- インフラストラクチャー・モニター

エラー防止: 総合的なテスト

総合的な機能およびシステム・テスト計画を実装することにより、リカバリーが必要になる問題の発生を防止できます。

一般に、デプロイされたソリューションのテストは、以下のように分類できます。

- 機能テスト

機能テストは、アプリケーションに実装された機能が、指定のビジネス要件を満たすか確認します。機能テストは、ビジネス・ユーザーとアプリケーション設計者によって作成されます。

- システム・テスト

システム・テストは、パフォーマンス、高可用性、およびリカバリーのサービス・レベル・アグリーメントを検証するように設計されています。

システム・テストでは、パフォーマンス・テストと高可用性テストのような要素を組み合わせ、極端な実稼働状況でのシステムのリカバリーを評価することが重要です。

機能テストとシステム・テストの両方に関して、自動化することを強くお勧めします。自動化テストは、組織にとって回帰バグが入り込むのを阻止する有効な方法です。

関連概念

 [リカバリー: ファースト・ステップ](#)

管理者は一般的な手法のファースト・ステップ・チェックリストに従うことによって、ソリューション・リカバリー・プロセスを容易に行うことができます。

関連情報

 [Problem determination in WebSphere Process Server](#)

エラー防止: 環境のチューニング

チューニング演習は、システム開発ライフ・サイクルの正規の部分です。大規模なアプリケーションのデプロイメントごとに、パフォーマンス評価をスケジュールに入れてください。

実稼働環境へのソリューションのデプロイに対する前提条件として、実動前環境でソリューションの評価とテストを実行してください。これにより、既存のアプリケーションおよび現在のシステム・パラメーターとリソースに対する新規ソリューションの影響を測定できます。実動前環境でのソリューションの評価およびテストを怠ると、そのソリューションがリカバリーに関して問題を持つようになる確率が高くなります。

パフォーマンス・テスト計画のプロセスおよび実行について説明した多くのリソースが、一般に入手可能になっています。そのような資料を調べ、ご使用のアプリケーションとトポロジーに適切なテスト計画を作成してください。

WebSphere Process Server のパフォーマンスとチューニングについての情報が掲載されている IBM Redbooks、および WebSphere Process Server のパフォーマンスとチューニングに関するテクニカル・ホワイト・ペーパーを参照してください。また、Business Process Management (BPM) の新規リリースと IBM の Connectivity 製品すべてに付属している、パフォーマンス・レポートも参照してください。

関連情報

チューニング

 [IBM WebSphere Business Process Management Performance Tuning](#)

 [Endurance testing with WebSphere Process Server](#)

 [WebSphere Business Integration V6.0.2 Performance Tuning](#)

 [Performance Tuning Automatic Business Processes for Production Scenarios with DB2](#)

 [WebSphere Process Server V6 – Business Process Choreographer Performance Tuning of Human Workflows Using Materialized Views](#)

エラー防止: インフラストラクチャーのモニター

インフラストラクチャーのモニター、およびインフラストラクチャー・モニター・ツールを使用することは、実動システムの要件の 1 つです。

ITCAM for SOA および Tivoli® Performance Viewer のようなモニター・ツールにより、システム管理者は、重要なシステムの振る舞いをモニターして、停止状態を引き起こしかねない問題を検出することができます。

実動システムでの基本的なレベルの IT モニターは、可用性サービス・レベル・アグリーメントに適合するために欠かせません。

サービス・コンポーネント・イベントのパフォーマンスおよびビジネス・プロセスのモニターについては詳しくは、WebSphere Process Server インフォメーション・センターの『モニター』のセクションを参照してください。

関連情報

モニター

IBM Tivoli Composite Application Manager Family for SOA:

IBM Tivoli Composite Application Manager Family (ITCAM) for SOA (Service-Oriented Architecture) を使用して、WebSphere Process Server をモニターできます。また、ITCAM for SOA を使用して、問題メディエーションを自動化し、ソリューションの構成およびデプロイメントを管理することもできます。

ITCAM for SOA には以下の機能が組み込まれています。

SOA サービスの管理

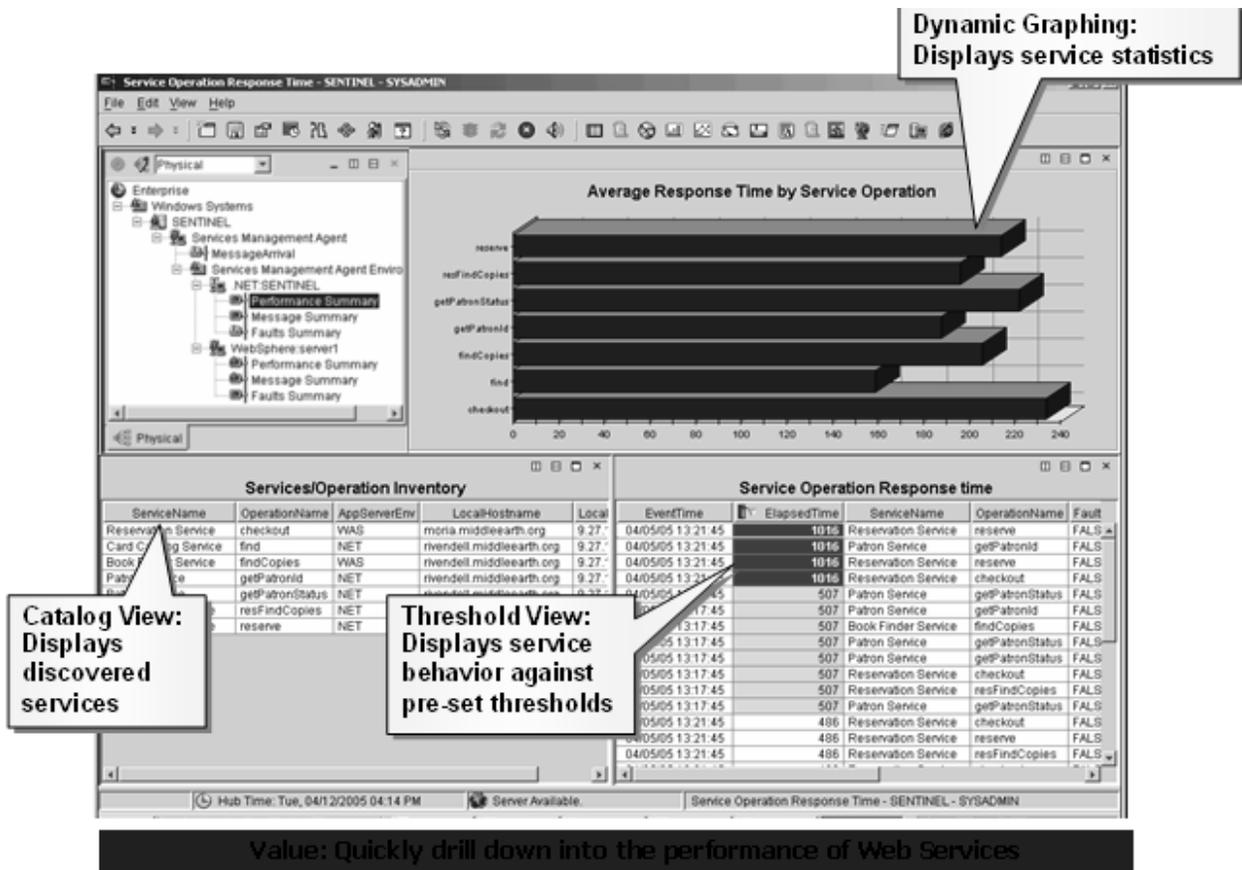
- SOA サービス対話内の可視性
- メッセージ・コンテンツ・パターンとトランザクション・フロー・パターン内の可視性
- テクノロジーとプラットフォームの境界を越えてパフォーマンスのボトルネックを識別して分離する能力
- 軽量で、業界標準のアプリケーション応答測定 (ARM) をベースとしたパフォーマンス計測
- ポリシーの高いパフォーマンスと柔軟な施行
- 統合が容易な標準ベースの計測

ビジネス・プロセスのモニター

- 未完了プロセスの管理
- アクティブ・プロセスのビジネス・パフォーマンスのモニター
- ビジネス・シチュエーションの検出およびアクションの実行
- 収集されたプロセス・データからのビジネス・インテリジェンスの収集
- 停止した、または動作速度の遅いアプリケーションを識別して素早く修正する総合的な deep-dive モニター
- リアルタイム・メトリックおよびヒストリカル・データ分析

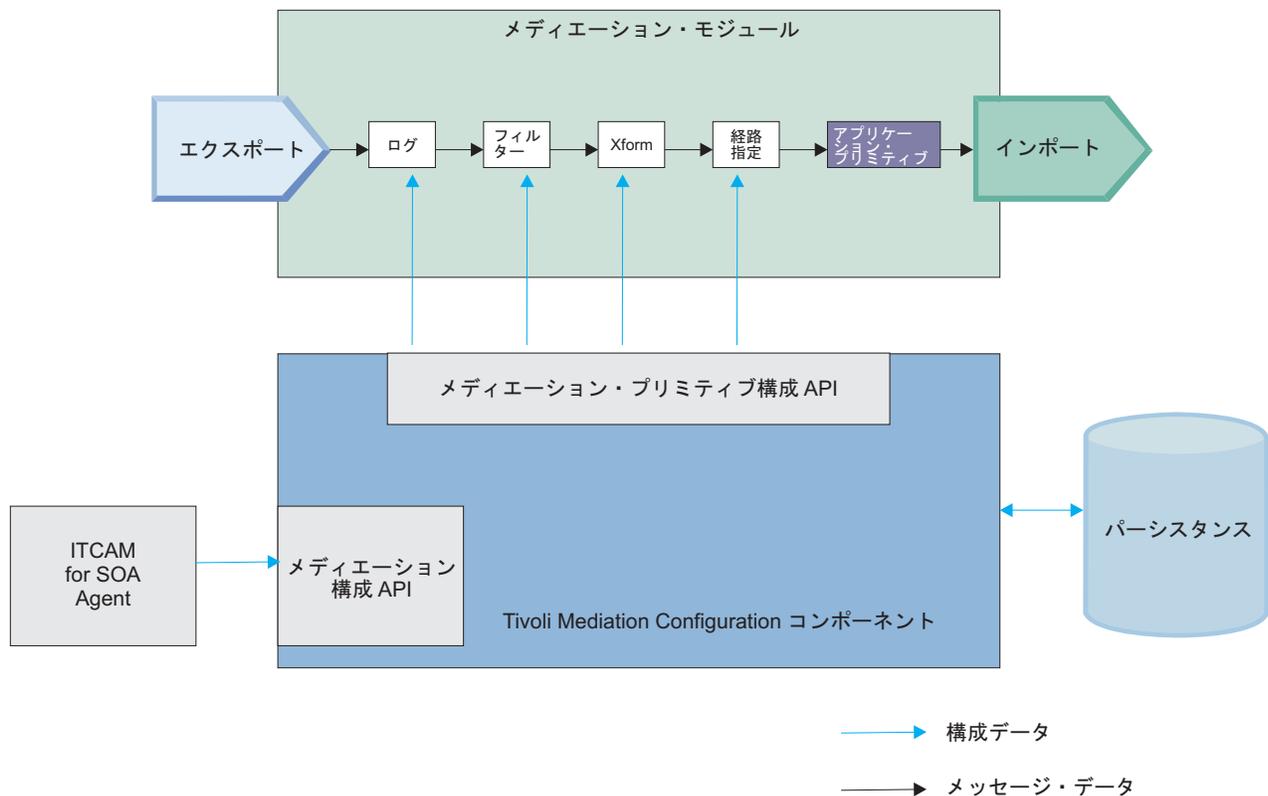
IBM Tivoli Composite Application Manager Family (ITCAM) for SOA の例

以下の例は、IBM Tivoli Composite Application Manager Family(ITCAM) for SOA による、サービス、応答時間、メッセージ数、およびメッセージ・サイズのモニター方法を示しています。



前述のとおり、IBM Tivoli Composite Application Manager Family (ITCAM) for SOA には、WebSphere ESB と連携してメディエーション・フロー構成を動的に変更する特別な機能が組み込まれています。

以下の図は、IBM Tivoli Composite Application Manager Family (ITCAM) for SOA のメディエーション・フロー構成機能を示しています。



関連情報

 [IBM Tivoli Composite Application Manager Family Installation, Configuration, and Basic Usage](#)

問題判別方法の文書

実稼働環境にデプロイするソリューションについて、明確で分かりやすい問題判別の方法論を確立します。

これは、問題判別方法の文書を維持し、文書化された方法を一貫して実行することを意味します。

操作マニュアルにおいて、ソリューション固有の問題判別方法論を文書化することをお勧めします。この操作マニュアルには、ソリューション固有の問題判別に関係した以下の種類の情報を含めてください。

- 問題判別中に観察結果を記録するための確立された形式

確立された形式を使用することにより、観察結果を記録する際の一貫性を保つことができます。Excel 形式のスプレッドシートは、一般的な「観察レポート作成ツール」です。

- トレース情報のリスト

ソリューション固有の問題判別用に、以下のトレース情報を含めてください。

- 有効にするトレースのリスト
- トレースを有効にするサーバーのリスト

- トレースを有効にするときの条件の記述

トレースを実装する前に、トレースによって問題が悪化しないことを確認してください。「すべてを有効にする」ことは適切ではありません。トレースの仕様は観察された状況に対して適切なものになっている必要があるため、トレースを有効にする場合は慎重に行ってください。インテリジェント・シチュエーション分析を使用して正しい診断情報を収集します。適切なトレース・レベルを実装する方法が不明確な場合は、IBM サポートにお問い合わせください。

- 冗長なガーベッジ・コレクション (verbosgc) を有効にする

冗長なガーベッジ・コレクション (GC) データには、特定のアプリケーションでの GC の実行方法に関する詳細が含まれています。この情報は、パフォーマンス問題の分析や、アプリケーションの GC 設定を調整する場合に役立ちます。

- ヒープ・ダンプの生成

ヒープ・ダンプ機能は、Java ヒープにあるすべてのオブジェクトのレコードをテキスト・ファイルに出力する IBM JVM の機能です。

各オブジェクトのサイズおよびアドレスが記録されるとともに、そのオブジェクトが参照するすべてのオブジェクトのアドレスが記録されます。この情報により、メモリーの大きな部分を占めているオブジェクトを把握することができます。

- Java.core の作成

javacore ファイルを分析して問題判別を実行することは、IBM Java 仮想マシン (JVM) で発生している可能性があるエラー状態の根本原因を判別するのに効果的です。

- 問題管理レコード (PMR) を開く前に収集する必要がある、ログの場所と種類。IBM の「must gather」スクリプトの正しい使用法を定義します。
- すべての保守パッケージ情報を含むバージョン情報 (versionInfo) の収集
- さまざまな問題が発生するときにデータベースによって記録されるログおよび情報を収集するための、データベース固有手順

作成するソリューション固有の問題判別文書は頻繁に更新される文書として扱い、機能およびシステム・テストによる観察で新しい手法が見つかるたびに、毎回保守および更新してください。

注: 問題判別および問題報告で活用可能な IBM Support Assistant を始めとする他のツールに精通し、使用してください。上に述べた情報を収集することは、その情報を含めることで PMR のサイクル・タイムが著しく短縮されるので、すべての新規 PMR を開く場合の前提条件としてください。

関連タスク

 管理コンソールを使用したデータ・スナップショット付きクロス・コンポーネント・トレースの有効化
データ・スナップショット付きでクロス・コンポーネント・トレースを有効に設定して、Service Component Architecture (SCA) 処理に関連付けられたデータと、WebSphere Process Server および WebSphere Enterprise Service Bus のモジュールおよびコンポーネントに関連付けられた呼び出しチェーン・データを収集します。

関連情報

 ヒープ・ダンプの検出および分析
製品のバージョン情報および履歴情報

 WebSphere Application Server で冗長なガーベッジ・コレクション (verbosegc) を有効にする

IBM Support Assistant

ソフトウェア適用状況

ソフトウェア適用状況は、デプロイされているソリューションの最新ソフトウェアを維持するための手法です。

デプロイされているソリューションのソフトウェア適用状況を維持することは重要です。

IBM は、製品ベースで存在するプログラム診断依頼書 (APAR) の適用を支援するフィックスパックを定期的に作成しています。このサービス・パッケージには、必須のコード変更が必ず含まれています。詳しくは、公開されている APAR フィックスのリストを参照してください。

関連情報

お客様サポートとの連絡
フィックスの入手

 WebSphere Process Server サポート

エラー処理方針とソリューション・リカバリー

WebSphere Process Serverには、リカバリーのために使用できるエラー処理機能とツールが組み込まれています。

ソリューションを構築するアーキテクチャー・チームは、エラー処理とリカバリーにおいて WebSphere Process Server のツールと機能を使用する方法を理解する必要があります。

アーキテクチャー・チームには、アプリケーション開発チームが遵守しなければならないエラー処理標準を作成する責任があります。

プロジェクトのエラー処理方針には、以下の点を含める必要があります。

- 作業単位の適切な使用法 (トランザクションおよびアクティビティ・セッション)

- 障害および ServiceBusinessExceptions の宣言および使用法
- 特に BPEL やメディエーション・フローのコンポーネントなど、すべてのコンポーネント・タイプで一貫した障害処理
- 再試行論理および「エラーの継続」 Business Process Choreographer 機能の使用法
- 完了したプロセス・インスタンス削除の適切な設定
- 同期および非同期の呼び出しパターンの正しい使用法
- インポート・タイプとエクスポート・タイプの適切な使用法
- メディエーション・フローでの再試行機能の正しい使用法

アーキテクチャー・チームは、上の点に加えて、WebSphere Process Server の組み込みリカバリー機能 (Failed Event Manager によるリカバリーなど) が適切に利用される設計パターンを作成する必要があります。

関連概念



ビジネス・プロセスでの障害処理

プロセスで障害が発生すると、ナビゲーションは障害ハンドラーまたは障害リンクに進みます。

関連タスク

119 ページの『デプロイメント環境パターンの選択』

IBM 提供のいずれかのトポロジー・パターンを選択するか、独自のカスタム・デプロイメント環境を作成することによって、デプロイメント環境を構成できます。このトピック・セクションでは、IBM 提供の使用可能なトポロジー・パターンのリストを示して説明すると共に、トポロジーを選択する場合の考慮事項を示します。

103 ページの『提供されたパターンの 1 つに基づくデプロイメント環境の計画』

このシナリオは、Service Component Architecture (SCA) アプリケーションに対するスケーラビリティ、可用性、およびサービス品質の要件がある場合で、IBM 提供のパターンの 1 つでそれらの要件を満たすことができる場合に使用します。

関連情報



ビジネス・プロセスでの障害処理および補正処理

安定した環境の保守

安定した環境を実現し、システムおよびアプリケーションの障害が発生する確率を低く抑えるために実行できる、いくつかの追加ステップがあります。

以降のセクションでは、ソリューションの安定性とシステムのリカバリーに影響する手動プロセスの数を減らすために、インフラストラクチャー・チームが活用できる手法について説明します。

環境作成の自動化

スクリプト・フレームワークは、環境の作成時の整合性に寄与します。

管理コンソールから実行可能なすべてのアクションは、スクリプトを使用して実行することもできます。特定のニーズに応じて使用およびカスタマイズすべき既存の IBM サービス・アセットがあります。これらのスクリプトは、調整を実行するたびに保守することができます。テスト環境で作業する場合、環境の再作成が必要にな

ることがよくあります。テスト環境の作成などの反復アクションを実装するには、スクリプトがもっとも効率的な方法です。テスト・システムのスクリプトは後で変更して、実動システムの作成に使用できます。

IBM Software Services for WebSphere (ISSW) の担当者と自動化デプロイメントについて話し合うか、ご使用の WebSphere Application Server 実稼働環境で活用されているのと同様の手順で構築してください。

関連情報

 [スクリプトの使用 \(wsadmin\)](#)

コマンドおよびスクリプト

アプリケーション・デプロイメントの自動化

自動化スクリプトを使用して、適切な環境へのアプリケーションまたはソリューション・グループのデプロイメントを支援します。

優れた設計の「ビルド、パッケージ、およびデプロイ」モデルには数多くの利点があり、例えば、開発者の生産性向上、ビルドおよびコード修正にかかるターンアラウンド・タイムの短縮、アプリケーション・コードにおける整合性の向上、および開発ポリシーの強化などが挙げられます。

アプリケーションまたはソリューション・グループのデプロイで使用される自動化スクリプトは、ご使用の環境を作成する際の自動化プロセスを補完します。

スクリプトを使用するアプリケーション・デプロイメントの自動化では、環境への手操作による介入が減り、再デプロイメントやリカバリーにヒューマン・エラーが入り込む余地も少なくなります。

IBM Software Services for WebSphere (ISSW) の担当者と自動化デプロイメントについて話し合うか、ご使用の WebSphere Application Server 実稼働環境で活用されているのと同様の手順で構築してください。

関連情報

 [wsadmin スクリプトによるアプリケーションの管理](#)

 [Jython スクリプト・ライブラリー](#)

リカバリー方法の計画

リカバリー方法を計画すれば、正常にリカバリーする確率が高くなります。

高可用性 (HA)

高可用性 (HA) とは、すべての障害に耐えて、事前定義されたサービス・レベルに従って処理能力を提供し続ける IT サービスの能力のことです。

ソリューション・リカバリーを促進するために行うことができる重要な作業の 1 つは、高可用性 (HA) を付与してシステムを構成することです。対象となる障害には、保守およびバックアップなどの計画イベントと、ソフトウェア障害、ハードウェア障害、電源障害、および災害などの未計画イベントの両方が含まれます。クラ

スター環境は、本質的に高可用性を備えています。これは、クラスター化されたシステムは、ノードまたはデーモンの障害が発生すると再構成され、作業負荷をクラスター内の残りのノードに再配分することができるからです。

高可用性ソリューションは、ハードウェア、ソフトウェア、およびサービスを組み合わせられており、リカバリー・プロセスが完全に自動化され、ユーザー・アクティビティーを中断させません。HA ソリューションは、リカバリー時間が最短になる直前のリカバリー・ポイントを見つける必要があります。

高可用性ソリューションでは、アプリケーション・サーバーが問題を検出すると、そのトランザクションと関連するデータが別のサーバー (同じデータ・センター内、または災害の場合は別の地理的位置にあるサーバー) に自動的に移動されます。トランザクションおよび関連データを別のサーバーに移すことを、ピア・リカバリー と呼びます。

関連資料

152 ページの『ピア・リカバリー』

ピア・リカバリーは、同じクラスター内の別のメンバーによって実行されるリカバリーであり、手動でも自動でも開始できます。ピア・リカバリー処理 (自動ピア・リカバリーまたは手動ピア・リカバリー) は、WebSphere の高可用性環境と緊密に連携しています。

関連情報

 [高可用性およびワークロード共用についての学習](#)

リカバリーの環境および目的

リカバリー・スペクトルの範囲は、テスト環境と実稼働環境、および異なるリカバリー目的 (システム・リカバリーとアプリケーション・リカバリー) に及びます。リカバリーの目標および目的は、リカバリー元となる環境に応じて異なります。

関連概念

149 ページの『トランザクションのプロパティとソリューション・リカバリー』
WebSphere Process Server は WebSphere Application Server をベースとしているため、ビジネス・トランザクションを実行するトランザクション・モデル をサポートしています。

関連情報

 [トランザクションの高可用性](#)

 [Asynchronous replication of WebSphere Process Server and WebSphere Enterprise Service Bus for disaster recovery environments](#)

実稼働環境でのリカバリー

実稼働環境での目標は、整然とした一貫性のある方法でシステムに入力された要求すべてを処理することです。この環境ではデータを保持する必要があり、システムを使用できなかつたり、データを損失したりする状況を最小限に抑えるための手段をすべて実施する必要があります。

実稼働環境に関する次の側面を考慮してください。

- トポロジー・タイプ

ご使用の実稼働環境に最適なトポロジー・タイプを理解しておく必要があります。ご使用の環境に適切なトポロジーを見つけるには、アプリケーションのプロパティと機能に無関係の要件を分析する必要があります。

トポロジー・タイプについて詳しくは、WebSphere Process Server インフォメーション・センターのトピック『提供されたパターンの 1 つに基づくデプロイメント環境の計画』を参照してください。

- リカバリーする必要がある条件についての理解と洞察

例えば、クラスターに複数のクラスター・メンバーがある場合は、リカバリーする必要がある唯一のコンポーネントが単一クラスター・メンバーであり、ワークロード管理機能によって処理が既に「実行中のサーバー」に転送済みになっている可能性があります。このような場合は、サーバーの再始動によってリカバリーが強制され、そのサーバーはクラスターに再び参加します。

一部の高可用性 (HA) 構成では、別のところにある 1 つのサーバーから失敗したトランザクションをリカバリーできます (ピア・リカバリー)。

実動データのリカバリーでは、システムおよびアプリケーションの 2 つのレベルで成功する必要があります。

関連タスク

103 ページの『提供されたパターンの 1 つに基づくデプロイメント環境の計画』このシナリオは、Service Component Architecture (SCA) アプリケーションに対するスケーラビリティ、可用性、およびサービス品質の要件がある場合で、IBM 提供のパターンの 1 つでそれらの要件を満たすことができる場合に使用します。

テスト環境でのリカバリー

テスト環境の目標および前提の数は、実稼働環境の目標や前提と異なります。

テスト環境での目標は、新規テストをできるだけすぐに実施できるような方法でシステムをリカバリーすることです。データの保持は不要で、システム内のすべての要求は破棄可能という前提があります。

注: これは、「リカバリー」テストとは異なります。リカバリー・テストでは、実動シナリオとして提供される推奨事項を利用し、プロジェクトのシステム・テスト・フェーズ中に実行されます。

システム・リカバリー

システム・リカバリーとは、ソリューションのインフラストラクチャーへの悪影響を修正するために (手動または自動で) 実行されるオペレーションのことです。

WebSphere Process Server のソリューションは、基本的なインフラストラクチャーの要件に依存しています。WebSphere Process Server は、以下の条件のいずれによっても悪影響を受ける可能性があります。

- 電源異常
- ネットワークの破損
- データベース障害

- ハードウェア障害

これらのタイプのいずれかの中断が発生した場合は、WebSphere Process Server システムのリカバリー前に、それに対処して修正する必要があります。

基本的なインフラストラクチャー要件における障害または中断に対処するとすぐ、WebSphere Process Server は、WebSphere からの継承機能に基づいてアプリケーション・リカバリーを開始します。

関連概念



リカバリー: ファースト・ステップ

管理者は一般的な手法のファースト・ステップ・チェックリストに従うことによって、ソリューション・リカバリー・プロセスを容易に行うことができます。

アプリケーション・リカバリー

アプリケーション・リカバリーとは、未完了ビジネス・トランザクションのリカバリーと解決を意味します。

システムで障害 (電源異常など) が発生すると、アクティブなトランザクションの多くが影響を受けます。それらのトランザクションすべては、プロセス・フローのさまざまな段階にあります。システムのリカバリー・プロセスの一環としてシステムがそれらのトランザクションを処理します。

アプリケーション・リカバリーを最後まで完全に実行するには、それらのアプリケーション自体が、設定されている防止手段を遵守する必要があります。

リカバリーおよびトランザクションの適用範囲を念頭に、ベスト・プラクティスに従ってアプリケーションが開発されていない場合、アプリケーション・リカバリーが最後まで完全に実行される確率は低くなります。

設計に問題がある、または「未調整」のシステムまたはアプリケーションでは、それ以外のアプリケーションが新規イベントの処理を開始した後も未解決のまま残る、未完了のトランザクションまたはプロセスの割合が高くなります。これは、WebSphere Process Server だけでなく、すべての Java EE アプリケーションとアプリケーション・サーバーの場合にも当てはまります。

注: 「未調整」という語句は、パフォーマンスの考慮事項やエラー処理手法について考慮せずに、すべてのコンポーネントでデフォルト設定を使用するソリューションを指します。

未解決イベントは、実行状態のままのプロセスや再実行依頼できない失敗イベントなど、異なる形態で送信されてくる可能性があります。完全リカバリーのためにアプリケーション内部で必要とされる変更を判別するには、これらのイベントのリカバリー後分析を行うことが必要になります。そのような変更は、総合的な機能およびシステム・テスト計画の実行時に発見できます。

関連概念

リカバリー: ファースト・ステップ

管理者は一般的な手法のファースト・ステップ・チェックリストに従うことによって、ソリューション・リカバリー・プロセスを容易に行うことができます。

関連情報

失敗イベントの管理

トランザクションのプロパティとソリューション・リカバリー

WebSphere Process Server は WebSphere Application Server をベースとしているため、ビジネス・トランザクションを実行するトランザクション・モデル をサポートしています。

WebSphere Process Server はこのトランザクション・モデルに基づいて構築されており、疎結合の SOA アプリケーションおよび BPM アプリケーションを提供します。

これは、技術的に 2 つの事柄を意味します。

1. WebSphere Process Server は、トランザクション・アプリケーション実行パターンを実現するためにデータベース・システムおよびメッセージング・システムに依存しています。
2. トランザクションは、メッセージング・システムおよびデータベース・システムにおいて重要な役割を担っています。

トランザクションは、ACID プロパティに対応しています。トランザクションは、原子性、一貫性、独立性、および耐久性を持つときに、ACID 準拠と見なされます。

WebSphere Process Server ではデータベース・システムおよびメッセージング・システムを使用して、「疎結合された」パターンを実現します。WebSphere Process Server はデータベースを更新してメッセージを送信します。データベースの更新およびメッセージの処理は同じトランザクションでコミットされます。

「疎結合」パターンの別の特徴は、メッセージング・システムからメッセージを取り出して、データベースを更新することです。この処理中に障害が発生すると、イベントはあたかも未読であったかのようにメッセージ・キューに戻ります。WebSphere Process Server には再試行メカニズムが存在し、5 回試行した後でイベントは Failed Event Manager に渡されます。「疎結合」という句は、すべての処理が 1 つの大きなトランザクション内で発生しなくてもよいことを表しています。

システム障害イベントでのデータ損失の回避

利用可能なリソース・マネージャーを適切に調整および構成しておけば、システムのある部分に障害があってもデータは失われません。ロールバックおよびリカバリーのメカニズムなどのトランザクションの保全性は、障害が発生してもデータが失われないようにする、WebSphere のキー・コンポーネントです。

WebSphere のロールバックおよびリカバリー・メカニズムを機能させるには、リソース・マネージャー (データベースおよびメッセージング) を正しくセットアップする必要があります。例えば、データベースのロック・タイムアウトを適切に設定し、サーバーのリカバリー時に、ロック状態になることなく、データベースでコミットまたはロールバックのどちらかを完了できるようにする必要があります。

WebSphere Process Server は、WebSphere Application Server の機能を拡張する機能を追加することで、予期しない障害からデータを回復する完全なソリューションを提供します。

リカバリー機能の使用可能化の概要

WebSphere Process Server のコア・リカバリー・モデルの基本は作業単位です。システムは、システム操作中に発生する障害を、実行される単一の作業単位に注目して処理および回復できるため、中断することなくサービスを提供できます。このタイプのリカバリーは、一連の再試行メカニズムとエラー・キューによって行われます。アプリケーション設計の一部に、システム・エラーをアプリケーション・エラーと区別する機能を組み込んでください。システム・エラーは、呼び出し側コンポーネントをサポートするインフラストラクチャーに戻されます。そこでは、追加のシステム・レベル・リカバリーが試みられたり、より一般的なビジネス例外への変換が行われたりします。自動的に実行されるさまざまな再試行メカニズムを構成できます。また WebSphere Process Server では、必要な場合に人間の詳細な介入を可能にする一連のコンソールおよび対応するプログラミング・インターフェースが提供されています。これらの機能およびこれらが扱う障害の多くは、この作業を実行するサーバーが新しい要求の処理を継続している間も活用できます。

使用不可サーバー - 概要

障害によって、高い可用性を持つ WebSphere クラスター内の 1 つまたは複数のサーバーが使用不可になった場合、システム内の追加のリカバリー機能が以下のように呼び出されます。

1. インバウンド処理を経路指定して障害システムから引き離す

これは、基盤となる WebSphere Application Server のワークロード管理機能を使用して実行されます。この機能は、プロトコル、トポロジー、および構成によって異なります。

2. 管理者がアクションを開始する

システムは、全体としてアクティブで使用可能な状態を続けますが、管理者はリカバリー操作を実行できます。

管理者のアクションの目的は、基本的な優先順位を決定し、停止しているサーバーを再始動することです。この再始動によってトランザクション・ログが再生され、ほとんどのサーバー・ダウンの状況がクリーンアップされるはずですが。

完全リカバリーを管理するには、WebSphere Process Server で提供されているエラー処理メカニズムを使用することが必要な場合があります。

使用不可クラスター - 概要

サーバー・クラスター全体が使用できないか、または応答しない場合、より複雑な一連のリカバリー・アクションが必要です。例えば、データベースなどの共用リソースが利用できない場合は、クラスター内のどのサーバーでも一様に処理の完了が難しくなります。

共用リソース・リカバリーを処理する手順は、障害が発生している共用リソースによって決まります。さまざまな WebSphere の技法を適用して、全体的なダウン時間を最小化し、停止した処理を再開することができます。

壊滅的な障害 - 概要

壊滅的な状況では、マシン全体が利用できないか、またはサーバーがリカバリー可能ではありません。このような事例では、WebSphere の拡張機能を使用して、サーバー障害のリカバリーを同じクラスター内の別のサーバー上で実行することができます。この機能では、ログを共用するためのネットワーク接続されたストレージまたはその他のメカニズムを用意するという前提条件を満たすことで、この種類のリカバリーも可能になります。同じクラスターの別のメンバーによる障害サーバーのリカバリーの詳細については、152 ページの『ピア・リカバリー』を参照してください。

関連概念

 ユース・ケース: 失敗イベントからのデータのリカバリー

ユース・ケースは、リカバリー・シナリオでのコンテキストとして使用されます。このユース・ケースでのビジネスには、新規アカウントを作成する要求を受け取るアプリケーションがあります。

サーバーの管理

146 ページの『リカバリーの環境および目的』

リカバリー・スペクトルの範囲は、テスト環境と実稼働環境、および異なるリカバリー目的 (システム・リカバリーとアプリケーション・リカバリー) に及びます。リカバリーの目標および目的は、リカバリー元となる環境に応じて異なります。

関連資料

『ピア・リカバリー』

ピア・リカバリーは、同じクラスター内の別のメンバーによって実行されるリカバリーであり、手動でも自動でも開始できます。ピア・リカバリー処理 (自動ピア・リカバリーまたは手動ピア・リカバリー) は、WebSphere の高可用性環境と緊密に連携しています。

関連情報

イベントの使用

 ビジネス・プロセスのトランザクションの振る舞い

 ビジネス・プロセスでの補正処理 (Compensation handling in business processes)

ピア・リカバリー

ピア・リカバリーは、同じクラスター内の別のメンバーによって実行されるリカバリーであり、手動でも自動でも開始できます。ピア・リカバリー処理 (自動ピア・リカバリーまたは手動ピア・リカバリー) は、WebSphere の高可用性環境と緊密に連携しています。

HA マネージャー

WebSphere では、HA マネージャー・コンポーネントを使用して、アプリケーション・サーバーにより提供されるサービスをモニターします。これらのサービスとしては、メッセージング、トランザクション・マネージャー、ワークロード管理コントローラー、およびクラスター内の他のアプリケーション・サーバーがあります。さらに HA マネージャー・コンポーネントでは、Network Attached Storage (NAS) デバイスを使用して、クラスター内の各アプリケーション・サーバーからのトランザクション・ログを保管します。

HA マネージャーは、定義された HA クラスターで障害が発生したサーバーに関して、未確定 トランザクションと未完了 トランザクションの両方の自動ピア・リカバリーを実行します。未確定トランザクションとは、例えばノードの除去によりメッセージング・エンジンが破壊されるなどの例外的な状況が原因で、いつまでも未確定状態のまま動かなくなったトランザクションです。未確定トランザクション 状態が発生するのは、データベースがフェーズ 1 のコミット処理を終えてからフェーズ 2 を開始するまでの間です。未完了トランザクションとは、コミット・プロセス

の「準備フェーズ」をまだ完了していないトランザクションです。この場合、トランザクションまたはメッセージはどこかに残っており、その場所でトランザクションまたはメッセージをリカバリーすることができます。HA マネージャーによって実行される自動リカバリー機能により、クラスターは 1 つ以上のクラスター・メンバーに障害が発生した場合にバランスを取り戻すことができます。

自動ピア・リカバリーと手動ピア・リカバリー

自動ピア・リカバリー は、デフォルトのピア・リカバリー開始方式です。アプリケーション・サーバーに障害が発生した場合、WebSphere Application Server は、そのアプリケーション・サーバーの代わりにピア・リカバリー処理を実行するサーバーを自動的に選択します。高可用性を有効にしたり、各クラスター・メンバーのリカバリー・ログ・ロケーションを構成したりするほかには、このモデルを使用するために追加の WebSphere Application Server 構成ステップを行う必要はありません。

手動ピア・リカバリー は、明示的に構成する必要がある特定のピア・リカバリー方式です。アプリケーション・サーバーに障害が発生した場合、オペレーターは管理コンソールを使用して、そのアプリケーション・サーバーの代わりにリカバリー処理を実行するサーバーを選択することができます。

ピア・リカバリー参照情報

『IBM WebSphere 開発者用技術ジャーナル: WebSphere Application Server V6 におけるトランザクションの高可用性とデプロイメントに関する考慮事項 (IBM WebSphere Developer Technical Journal: Transactional high availability and deployment considerations in WebSphere Application Server V6)』という表題の記事で、自動および手動によるピア・リカバリーの要件、セットアップ、管理について説明しています。

追加の文書については、WebSphere Application Server インフォメーション・センターおよび『WebSphere Application Server V6 Scalability and Performance Handbook』を参照してください。

- WebSphere Application Server V6 Scalability and Performance Handbook
- WebSphere Application Server インフォメーション・センターの『トランザクション・プロパティの、ピア・リカバリー用の構成』
- WebSphere Application Server インフォメーション・センターの『トランザクション・サービスの手動ピア・リカバリーの管理』

関連概念

リカバリーのトリガー

ソリューション・リカバリーは、さまざまなトリガーの結果として必要になります。

149 ページの『トランザクションのプロパティとソリューション・リカバリー』
WebSphere Process Server は WebSphere Application Server をベースとしているため、ビジネス・トランザクションを実行するトランザクション・モデル をサポートしています。

エラー防止とリカバリーの概要

エラー防止とリカバリーの情報では、システム障害を引き起こす問題を回避する方法について説明し、通常の状態と異常な状態の両方で発生する可能性があるシステム障害からリカバリーする方法について、情報を提供しています。

リカバリーのトリガー

ソリューション・リカバリーは、さまざまなトリガーの結果として必要になります。

高可用性 (HA)

高可用性 (HA) とは、すべての障害に耐えて、事前定義されたサービス・レベルに従って処理能力を提供し続ける IT サービスの能力のことです。

エクスポート・バインディング

システムを完全に停止するには、使用可能なエクスポート・バインディングによってサポートされるさまざまなタイプの要求呼び出しを考慮します。

SCA 呼び出しパターン

次の表は、各種エクスポート・バインディングに使用される SCA 呼び出しパターンのタイプを示したものです。

表 17. EIS エクスポート・バインディングと関連する呼び出しパターン

エクスポート・バインディング	操作のタイプ	パフォーマンス属性と対話スタイル	呼び出しスタイル
EIS	片方向	非同期	非同期 (デフォルト)
		同期	同期
	要求応答	任意の値	同期

表 18. エクスポート・バインディングおよび関連する操作タイプと呼び出しスタイル

エクスポート・バインディング	操作のタイプ	呼び出しスタイル
EIS	片方向または要求応答	同期
MQ または MQ JMS	片方向	非同期
SCA JMS	片方向	非同期
	要求応答	コールバック付き非同期

表 18. エクスポート・バインディングおよび関連する操作タイプと呼び出しスタイル (続き)

エクスポート・バインディング	操作のタイプ	呼び出しスタイル
Web サービス (soap/http) または (soap/jms)	片方向または要求応答	同期

使用するアプリケーションやトポロジーに応じて、さまざまな方法で同期通信を停止することができます。同期通信の停止方法を定義する場合は、使用するエクスポートやトポロジーの固有の特性に基づいて定義することを強くお勧めします。

関連概念

130 ページの『接続グループ』

接続グループは、SCA モジュール内に見られる特定の動作パターンを表します。

関連情報

エクスポートとエクスポート・バインディング

エクスポートの使用

バインディング

Failed Event Manager について

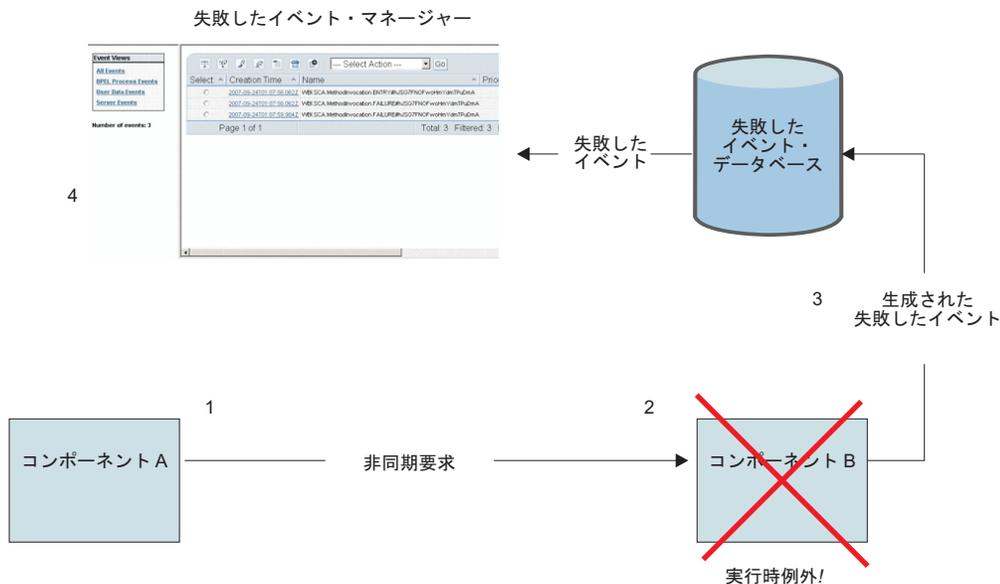
Failed Event Manager は、呼び出しの失敗を処理および再サブミットするための Web ベースのクライアントです。

Failed Event Manager は、インテグレーション・アプリケーションで、管理コンソールで使用できます。

このアプリケーションによって、失敗イベントの数が表示され、多数の検索機能が提供されます。

失敗イベントを、日付、最後の成功イベントまたは失敗イベントなどさまざまな基準を使用して、例外テキストによって、またはこれらを組み合わせて、照会できます。

以下の図は、WebSphere Process Server 例外処理の高位での説明、および Failed Event Manager との関係を示しています。図の後に、番号付きでステップの説明があります。



1. コンポーネント A は非同期でコンポーネント B を呼び出しています。
2. コンポーネント B にランタイム例外が発生し、失敗イベント・レコードが生成されました。
3. 障害リカバリー・サービスはこの障害を捕捉し、失敗イベントのデータベースに格納します。
4. システム管理者は Failed Event Manager を開き、問題を調査します。

Failed Event Manager を使用した失敗イベントの再サブミットについて

Failed Event Manager で入力した検索基準と合致するイベントが表示されます。単一または複数の失敗イベントを再サブミットできます。再サブミット中、ペイロードを変更することもできます。例えば、何らかの不適切なデータを渡したことが原因で、障害が発生したとします。この場合、ペイロードは Failed Event Manager 内から更新され、再サブミットできます。メモリーに格納されているデータのみが更新されるため、データのオリジナル・ソースは訂正されません。再サブミットされたイベントが失敗した場合、Failed Event Manager で新規の失敗イベントとして表示されます。また、単一または複数イベントを削除することが可能で、障害発生時点以降データは無効になっているため、これは多くの場合適切なアクションとなります。

関連概念



ユース・ケース: 失敗イベントからのデータのリカバリー

ユース・ケースは、リカバリー・シナリオでのコンテキストとして使用されます。このユース・ケースでのビジネスには、新規アカウントを作成する要求を受け取るアプリケーションがあります。

関連情報

失敗イベントの管理

失敗イベントの再サブミット

第 9 章 共存

共存とは、同一のシステムまたはネットワークで複数のエンティティを使用できるようにする機能です。

共存は、これらのエンティティのインターオペラビリティを意味するものではありません。ここでいう共存とは、複数の WebSphere サーバー構成 (同一製品の異なるバージョンが含まれることもあります) が単一の環境で共存することです。

共存のサポート

共存のサポートにより、ご使用のシステムで複数の製品をインストール、構成、実行、および管理することができます。

ご使用のシステムに以下の製品を一緒にインストールすることができます。

- WebSphere Application Server for z/OS
- WebSphere Process Server for z/OS
- ESB のみのサーバーとして構成された WebSphere Process Server

共存用にシステムをセットアップするときは、以下の点に注意してください。

- **WebSphere Application Server for z/OS の依存関係**

WebSphere Process Server for z/OS は、WebSphere Application Server for z/OS 上にインストールされます。したがって、最初に各サーバーを WebSphere Application Server for z/OS で構成して、WebSphere Application Server for z/OS 共存に準拠させ、その後に WebSphere Process Server for z/OS をインストールして構成する必要があります。詳しくは、WebSphere Application Server for z/OS のインフォメーション・センターの『共存サポート』を参照してください。

- **バージョンの共存**

同じシステムに旧バージョンの製品を置くことができます。この製品の旧バージョンは、同じ LPAR 内に置くことも、別個の LPAR に置くこともできます。

WebSphere Process Server for z/OS を WebSphere Business Integration Server Foundation for z/OS サーバーの上にインストールすることはできません。

- **共存の場合のインストール**

共存の場合、以下のインストール・シナリオがサポートされます。

- WebSphere Application Server for z/OS、WebSphere Process Server、および WebSphere ESB の共存 (まだいずれの製品もインストールされていない場合)
- WebSphere Process Server for z/OS と WebSphere Application Server for z/OS の共存 (WebSphere Application Server for z/OS が既に存在する場合)
- 既存の WebSphere Application Server for z/OS における WebSphere Process Server for z/OS と、ESB のみのサーバーとして構成された WebSphere Process Server for z/OS の共存



Printed in Japan