

バージョン 6.1.2



製品概説

バージョン 6.1.2



製品概説

お願い

本書に記載されている情報をご使用になる前に、本書末尾の特記事項セクションに記載されている情報をお読みください。

本書は、WebSphere Process Server for Multiplatforms バージョン 6、リリース 1、モディフィケーション 2 (製品番号 5724-L01)、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® Process Server for Multiplatforms
Version 6.1.2
Product Overview

発行： 日本アイ・ビー・エム株式会社

担当： ナショナル・ランゲージ・サポート

第 1 刷 2008.9

© Copyright International Business Machines Corporation 2005, 2008. All rights reserved.

目次

| | | | |
|---|----|---|----|
| 図 | v | 第 8 章 エンタープライズ・サービス・バスのメッセージング・インフラストラクチャー | 29 |
| 第 1 章 WebSphere Process Server の概要 | 1 | メッセージ・サービス・クライアント | 30 |
| 第 2 章 このリリースの新機能 | 3 | メディエーション・モジュール | 30 |
| 第 3 章 製品ファミリーの概要 | 5 | メディエーション・プリミティブ | 34 |
| 第 4 章 WebSphere Process Server のアーキテクチャーの概要 | 9 | サービス・メッセージ・オブジェクト | 38 |
| サービス指向アーキテクチャー・コア | 10 | 第 9 章 WebSphere Process Server におけるアプリケーションの管理 | 43 |
| Service Component Architecture | 10 | 第 10 章 WebSphere Process Server におけるアプリケーションの開発およびデプロイメント | 45 |
| サービス・データ・オブジェクトおよびビジネス・オブジェクト | 12 | 第 11 章 WebSphere Process Server のセキュリティ | 47 |
| WebSphere Process Server での Common Event Infrastructure | 14 | 第 12 章 WebSphere Process Server のモニター | 49 |
| サポート・サービス | 15 | 第 13 章 サンプル | 51 |
| メディエーション・フロー | 15 | サンプル・ギャラリーのインストールおよびアクセス | 51 |
| インターフェース・マップ | 16 | Business Process Choreographer サンプル | 54 |
| ビジネス・オブジェクト・マップ | 16 | 第 14 章 標準への準拠 | 55 |
| リレーションシップ | 16 | アクセシビリティ | 55 |
| セレクター | 17 | 連邦情報処理標準 | 56 |
| ビジネス・カレンダー | 18 | インターネット・プロトコル バージョン 6 | 57 |
| サービス・コンポーネント | 19 | 第 15 章 グローバリゼーション | 59 |
| ビジネス・プロセス | 20 | 特記事項 | 65 |
| ヒューマン・タスク | 21 | | |
| ビジネス・ステート・マシン | 21 | | |
| ビジネス・ルール | 21 | | |
| 第 5 章 インポートおよびエクスポートとアダプター | 23 | | |
| 第 6 章 WebSphere Process Server でのデプロイメント環境 | 25 | | |
| 第 7 章 WebSphere が提供するビジネス・スペース | 27 | | |



| | | | |
|---|----|--|----|
| 1. WebSphere Process Server コンポーネント・ベースのフレームワーク | 9 | 5. ビジネス・スペース および Business Process Management 製品. | 28 |
| 2. WebSphere Process Server コンポーネント・ベースのフレームワーク | 10 | 6. メディエーション・モジュールの簡単な例 | 32 |
| 3. WebSphere Process Server コンポーネント・ベースのフレームワーク | 15 | 7. メディエーション・モジュールが含まれる EAR ファイルの簡単な例. | 33 |
| 4. WebSphere Process Server コンポーネント・ベースのフレームワーク | 20 | 8. 3 つのメディエーション・プリミティブから成るメディエーション・モジュール | 34 |
| | | 9. SMO 構造の概要. | 40 |

第 1 章 WebSphere Process Server の概要

IBM® WebSphere® Process Server は、実証されたビジネス・インテグレーション概念、アプリケーション・サーバー・テクノロジー、および最新のオープン・スタンダードから発展した、ビジネス・プロセス統合サーバーです。WebSphere Process Server は、ビジネス・ゴールに合致するプロセスの形成に役立つ、高性能のビジネス・エンジンです。

WebSphere Process Server により、標準に基づくビジネス・インテグレーション・アプリケーションのサービス指向アーキテクチャー (SOA) へのデプロイメントが可能になります。SOA では、常用のビジネス・アプリケーションを取り出し、そのアプリケーションを個々のビジネス機能やビジネス・プロセスに細分して、サービスとして提供します。WebSphere Application Server によって提供される堅固な J2EE 1.4 インフラストラクチャーおよび関連するプラットフォーム・サービスに基づき、WebSphere Process Server は最新のビジネス・インテグレーションの課題への対処に役立ちます。例えば、ビジネス・プロセスの自動化などに役立ちます。

WebSphere Process Server により、人員、システム、アプリケーション、タスク、ルール、およびその全体の相互作用にまたがるプロセスのデプロイメントが可能になります。疎結合のビジネス・プロセスに対するトランザクション・ロールバックなどの機能を提供することにより、長期間のビジネス・プロセスと短期間のビジネス・プロセスの両方がサポートされます。

ハードウェアおよびソフトウェア要件

WebSphere Process Server でサポートされるハードウェアおよびソフトウェアに関する公式の説明を確認するには、WebSphere Process Server system requirements の Web サイトを参照してください。

情報ロードマップ

製品のインフォメーション・センターの内部または外部にある情報源のナビゲーションに役立つよう、ビジネス・プロセス・マネージメント情報ロードマップが、www.ibm.com/developerworks/websphere/zones/bpm/roadmaps/bpm_info_resources.htmlにある IBM developerWorks® でオンラインで使用可能です。

第 2 章 このリリースの新機能

このバージョンには、ビジネスの柔軟性を高め、アプリケーションをより短時間で効果的にデプロイすることを可能にする、いくつかの新機能が組み込まれています。

注: WebSphere Process Server バージョン 6.1.2 のインフォメーション・センターが更新されました。フィックスパックのインストール方法については、『フィックスの入手』を参照してください。

WebSphere Process Server バージョン 6.1.2 へようこそ。この製品には、以下の新機能が組み込まれています。

- プロセス対話のための新しいビジネス・ユーザー・グラフィカル・ユーザー・インターフェース。
- ビジネス・ユーザー向けの新規 Web 2.0 BPM クライアントによってビジネス・ユーザーの能力を高め、次に示す主要な生産性機能により短時間で利益を生み出します。
 - 導入後すぐに稼働可能な、ビジネス・ユーザー向けの新しい Web 2.0 BPM クライアント。あらかじめ組み込まれたヒューマン・タスクを使用する随時コラボレーションのサポートを含め、構成可能なワーク・リストと詳細な作業項目ビューを備えています。
 - タスクを表示するためのビジネス・ユーザー向けグラフィカル・プロセス・ビュー。
- ビジネス・プロセス機能を追加し、ビジネスの柔軟性を向上させています。
- 処理中のビジネス・プロセス・インスタンスと対話するための新しい拡張機能により、変化するビジネス条件により柔軟に対応できます。
 - 処理中のプロセス・インスタンスのフローを変更する場合の柔軟性を向上させ、アクティビティーをスキップする機能に加えて、アクティビティー間を前方および後方にジャンプして移動する機能を装備。
 - 処理中にプロセス関連のデータ変更を取り込み可能。
 - ヒューマン・タスクとのプログラマチックな対話を実現する汎用 REST 型インターフェース。
 - WebSphere Business Modeler または WebSphere Integration Developer で作成されたビジネス・カレンダーを利用し、ビジネス・プロセスおよびヒューマン・タスクにおいて特定の時間ベース設定 (例えば、期間やタイムアウト) を定義可能。
- 使いやすさを高めることにより、より短時間かつ効果的に、ビジネス・プロセス・ソリューションの開発、テスト、デプロイメント、実行、および管理を行うことができます。
 - Web Services Description Language (WSDL) および XML Schema Definition (XSD) のサポートが拡張されており、多くの業界標準 XML スキーマを使用可能。
 - アプリケーションの開始時間とサーバーの再始動時間を改善。

- アプリケーション・ロジックの問題判別と分析にテスト・クライアントを最大限活用することにより、WebSphere Integration Developer Integrated Test Client で後で表示できるよう、実行時にサーバーを通過するデータを収集するトレース機能を強化。
- 決定応答の追跡性を高めるために、WebSphere Business Monitor で利用可能な追加のイベントおよびイベント情報を生成。
- WebSphere Process Server をより簡単にインストールおよび構成できます。これは、SOA インフラストラクチャーを正常にデプロイするために重要な機能です。新しく拡張されたインストールと構成の機能により、WebSphere Process Server の使いやすさを高めます。
 - 拡張されたインストーラーにより、標準インストール・プロセスの一部として、デフォルトまたは指定されたディレクトリーから WebSphere Process Server および WebSphere iFixes をインストール可能。これは、i5/OS または z/OS ではサポートされていません。
 - 各フィックスパック・レベルでの完全なインストール・パッケージの可用性を確保することにより、簡素化されたインストール・プロセスを実現。
 - リモート・サーバーでデータベース・インスタンスを作成する場合に必要なデータベース・スクリプトをインストール・メディアから直接取得することが可能。これにより、WebSphere Process Server をインストールしてスクリプトにアクセスする必要を排除しています。
- WebSphere MQ の MQCIH メッセージ・ヘッダーを操作するためのサポート機能を追加しました。CICS ブリッジでは、この機能を利用して CICS アプリケーションとの対話を使用可能にすることができます。
- 次の追加メッセージ形式で新しい形式のサポートを追加しました。
 - コンマ区切り値 (CSV) の部分的または全部のサポート。
 - 固定幅形式。
 - JavaScript Object Notation (JSON)。

関連タスク



Lotus Forms Server API および Lotus Forms Viewer の使用

IBM Lotus® Forms Designer によって作成されたフォームは、ヒューマン・タスクおよびプロセス用のユーザー・インターフェースとして使用できます。

第 3 章 製品ファミリーの概要

WebSphere Process Server は、WebSphere ビジネス・プロセス・マネージメント・プラットフォームの一部で、その他の IBM 製品と共に動作します。

WebSphere Application Server Network Deployment

WebSphere Process Server は、堅固な J2EE インフラストラクチャーと、WebSphere Application Server によって提供される関連プラットフォーム・サービスをベースにしています。WebSphere Process Server for z/OS[®] は、WebSphere Application Server for z/OS を基礎にして構築されています。WebSphere Application Server Network Deployment オファリングについては、WebSphere Application Server 資料を参照してください。

WebSphere Process Server は、WebSphere Application Server からのインフラストラクチャーおよびプラットフォーム・サービスについても作業を行います。

WebSphere Application Server について詳しくは、WebSphere Application Server バージョン 6.1 インフォメーション・センターを参照してください。

IBM WebSphere Enterprise Service Bus

エンタープライズ・サービス・バス機能は、基盤となる WebSphere Process Server 機能の一部であり、これらの機能を使用するために、WebSphere Enterprise Service Bus の追加ライセンスは必要ありません。ただし、企業で購入した WebSphere Enterprise Service Bus の追加スタンドアロン・ライセンスをデプロイし、WebSphere Process Server によるプロセス統合ソリューションの接続範囲を拡張できます。例えば、WebSphere Enterprise Service Bus を SAP アプリケーションのさらに近くにインストールして、IBM WebSphere Adapter for SAP をホストしたり、SAP メッセージを変換してからネットワーク間でその情報を、WebSphere Process Server によって連携可能になったビジネス・プロセスに送信したりすることができます。

IBM WebSphere Partner Gateway

WebSphere Process Server と共に使用される IBM WebSphere Partner Gateway は、企業間アプリケーションをサポートします。WebSphere Partner Gateway の制限付きライセンスが WebSphere Process Server に付属しています。WebSphere Partner Gateway について詳しくは、WebSphere Partner Gateway の製品資料ライブラリーを参照してください。

IBM WebSphere Integration Developer

WebSphere Integration Developer は、WebSphere Process Server 用の開発環境です。これは、WebSphere Process Server、WebSphere エンタープライズ・サービス・バス、および WebSphere Adapters 全体でサービス指向アーキテクチャー (SOA) ベースの統合ソリューションを構築する共通のツールです。WebSphere Integration Developer について詳しくは、IBM WebSphere Business Process Management インフォメーション・センターを参照してください。

IBM WebSphere Adapters

IBM WebSphere Adapters は、既存のエンタープライズ情報システム・インフラストラクチャーと、WebSphere Process Server にデプロイされたアプリケーションの統合を可能にします。WebSphere Adapters により、エンタープライズ・リソース・プランニング、ヒューマン・リソース、カスタマー・リレーションシップ・マネジメント、およびサプライ・チェーンの各システム間で情報を交換する統合プロセスを素早く簡単に作成できるようになります。

アプリケーション・アダプターでは、業界横断および業界固有のパッケージ・アプリケーションからデータおよびトランザクション情報を抽出し、それを中央のハブに接続します。テクノロジー・アダプターでは、データ、テクノロジー、プロトコルにアクセスするコネクティビティーを提供し、統合インフラストラクチャーを強化します。Adapter Development Toolkit は、カスタム・アダプターの作成に使用できます。

一部の WebSphere Adapter は、WebSphere Integration Developer に組み込まれているコンポーネントです。

WebSphere Adapters について詳しくは、IBM WebSphere Business Process Management インフォメーション・センターの WebSphere Integration Developer の資料を参照してください。

IBM WebSphere Service Registry and Repository

WebSphere Service Registry and Repository (WSRR) は、正常なサービス指向アーキテクチャー (SOA) における各サービスの選択、呼び出し、管理、ガバナンス、および再利用に使用される、一般的にサービス・メタデータと呼ばれる情報の保管、アクセス、および管理のためのシステムです。つまり、WSRR は、既に使用中であるか、使用する計画であるか、または認識する必要がある、ユーザーのシステムまたは別の組織のシステムにおけるサービスに関する情報の保管場所です。例えば、アプリケーションではサービスを呼び出す前に WSRR を検査し、その機能およびパフォーマンスの要件に最も合致するサービス・インスタンスを見つけることができます。また WSRR は、SOA ライフ・サイクルの他のステージでの役割も務めます。WebSphere Service Registry and Repository について詳しくは、『WebSphere Service Registry and Repository』を参照してください。

IBM WebSphere Business Modeler および IBM WebSphere Business Monitor

WebSphere Process Server および WebSphere Integration Developer には、統合アプリケーションのモデル化、構築、デプロイ、インストール、構成、実行、モニター、および管理を可能にする追加機能が組み込まれています。WebSphere Integration Developer は、IBM WebSphere Business Modeler および IBM WebSphere Business Monitor を補足するものです。

これらの製品について詳しくは、WebSphere Business Modeler インフォメーション・センターおよび IBM WebSphere Business Process Management インフォメーション・センターの WebSphere Business Monitor の資料を参照してください。

IBM Rational® Application Developer および IBM Rational Software Architect

WebSphere Integration Developer は、IBM Rational Application Developer または IBM Rational Software Architect と共に使用して、固有で、統合された、強力な統合開発プラットフォームを作成することができます。

これらの製品について詳しくは、Rational Application Developer インフォメーション・センターおよび Rational Software Architect インフォメーション・センターを参照してください。

IBM CICS® Transaction Gateway および IBM WebSphere Host Access Transformation Services

エンタープライズ・プロセスで再利用するための既存のアプリケーションを、CICS Transaction Gateway および WebSphere Host Access Transformation Services を組み込んだ IBM エンタープライズ・モデル化ポートフォリオで拡張することができます。

これらの製品について詳しくは、CICS Transaction Gateway Library および WebSphere Host Access Transformation Services (HATS) Information Center を参照してください。

IBM WebSphere Portal

IBM WebSphere Portal は、さまざまな管理機能へのアクセスを提供し、ポータルでビジネス・プロセスおよび WebSphere Process Server にあるその他の Service Component Architecture サービスにアクセスできるようにします。

WebSphere Portal について詳しくは、WebSphere Portal の製品資料ライブラリーを参照してください。

IBM WebSphere Application Toolkit

WebSphere Application Server Toolkit は、WebSphere Process Server で使用する Web サービスをアセンブル、テスト、およびデプロイするためのツールのセットです。

詳しくは、WebSphere Application Server インフォメーション・センターにある WebSphere Application Server Toolkit 資料を参照してください。

IBM WebSphere Extended Deployment

WebSphere Extended Deployment は、WebSphere Process Server Network Deployment 環境を提供しており、この環境にポリシーとして定義した処理目標を満たすように環境内のクラスター間でリソースを調整する機能を備えています。アプリケーションのボリュームの盛衰のため、ピーク期間中の要求を満たすために使用できる処理能力が足りなくなる可能性があります。また、重要なアプリケーションが所要処理時間を確保できるようにするため、リソースを最適化するのが困難になる可能性があります。

このような場合の処理能力の動的再配分により、ビジネスの必要を満たすことができます。WebSphere Extended Deployment は、アプリケーション・ボリュームの小さいクラスターからリソースを動的に除去し、追加リソースが必要なアプリケーションにサービスを提供するクラスターにそれを追加します。処理優先順位は WebSphere Extended Deployment でポリシーとして指定されます。

WebSphere Extended Deployment について詳しくは、WebSphere Extended Deployment インフォメーション・センターを参照してください。

第 4 章 WebSphere Process Server のアーキテクチャーの概要

WebSphere Process Server は、WebSphere Application Server の全機能を使用して、完全集中型の標準に基づいたビジネス・プロセス・エンジンを持つ統合プラットフォームを提供します。

WebSphere Process Server は、均一の呼び出しプログラミング・モデルおよび均一なデータ表現モデルに基づいたサービス指向アーキテクチャー (SOA) 統合プラットフォームです。

WebSphere Process Server の基本ランタイム・インフラストラクチャーは WebSphere Application Server です。SOA コアの一部である Service Component Architecture およびビジネス・オブジェクトは、均一の呼び出しプログラミング・モデルおよびデータ表現プログラミング・モデルを提供します。SOA コアには、WebSphere Process Server 上で稼働するアプリケーションのモニターと管理のためのイベントを生成する Common Event Infrastructure が含まれています。サポート・サービスは、基本のビジネス・オブジェクトおよび変換フレームワークを WebSphere Process Server に提供します。サービス・コンポーネントは、複合アプリケーションに必要な機能コンポーネントを表します。

WebSphere Process Server の強力な基盤 (WebSphere Application Server および SOA コア) とサービス・コンポーネントの組み合わせによって、WebSphere Process Server で稼働する高度な複合アプリケーションを短時間で開発しデプロイすることができます。

1つのコンポーネント・ベースのフレームワークであらゆるスタイルの統合に対応します。

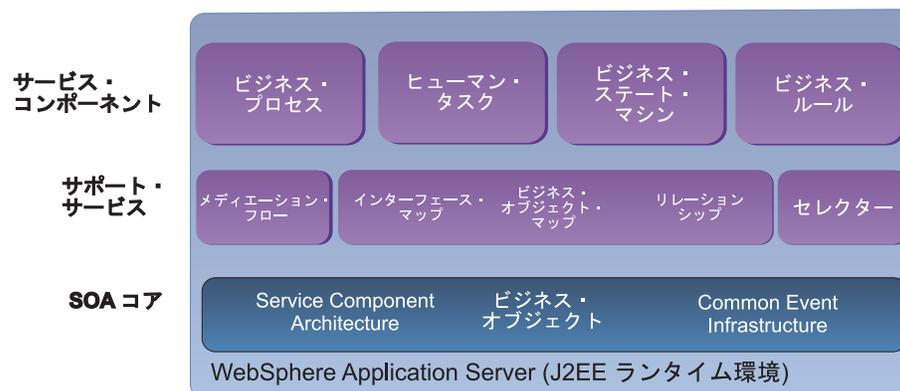


図 1. WebSphere Process Server コンポーネント・ベースのフレームワーク

サービス指向アーキテクチャー・コア

IBM WebSphere Process Server のサービス指向アーキテクチャー (SOA) コアは WebSphere Process Server で稼働するアプリケーションに対して、均一な呼び出しプログラミング・モデルとデータ表現プログラミング・モデルの両方を提供し、さらにはモニター機能および管理機能も提供します。

サービス指向アーキテクチャー (SOA) は、アプリケーションが提供するコンポーネントとサービスに関して、基礎にあるコンポーネント、サービス、およびコンポーネント間接続のインプリメンテーションに関係なく、ソフトウェア・システムの構造を概念的に記述するものです。WebSphere Process Server では、標準ベースのプロセス統合ソリューションを SOA でデプロイできます。つまり、コンポーネント用のビジネス・レベル・インターフェースの明確なセットを、下位テクノロジーの変更の影響を受けることなく作成および保守できます。SOA に基づく疎結合統合アプリケーションは、柔軟性と敏捷性を備えています。統合ソリューションは、プラットフォーム、プロトコル、および製品から独立してインプリメントできます。SOA について詳しくは、IBM Web サイトの Service Oriented Architecture (SOA) を参照してください。

SOA コアの一部である Service Component Architecture およびビジネス・オブジェクトは、WebSphere Process Server にデプロイされるアプリケーションに対して、均一の呼び出しプログラミング・モデルおよびデータ表現プログラミング・モデルを提供します。SOA コアには、WebSphere Process Server 上のアプリケーションをモニターおよび管理するためのイベントを生成する Common Event Infrastructure も含まれています。

1つのコンポーネント・ベースのフレームワークであらゆるスタイルの統合に対応します。

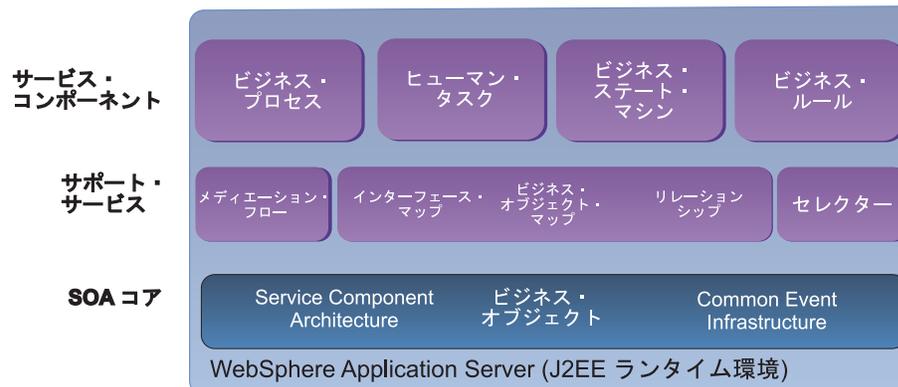


図2. WebSphere Process Server コンポーネント・ベースのフレームワーク

Service Component Architecture

Service Component Architecture では、WebSphere Process Server のランタイム環境におけるビジネス・トランザクションのすべてのエレメントをサービス指向の方法で表します。

Service Component Architecture (SCA) とは、Web サービス、エンタープライズ情報システム (EIS) のサービス・アセット、ビジネス・ルール、ワークフロー、および

データベースなどへのアクセスなどのビジネス・トランザクションのすべてのエレメントがサービス指向の方法で表されるアーキテクチャーです。

Service Component Architecture によりビジネス・ロジックがインプリメンテーションから切り離されるので、インプリメンテーションの詳細を知らなくても統合アプリケーションのアセンブルに集中できます。ビジネス・プロセスのインプリメンテーションは、サービス・コンポーネントに含まれます。

サービス・コンポーネントは、IBM WebSphere Integration Developer ツールを使用してグラフィカルにアセンブルできるので、インプリメンテーションを後で追加できます。Service Component Architecture プログラミング・モデルは、Java™ および J2EE、または特定のシナリオにおけるその他のインプリメンテーションについて開発者が知っておくべき知識の範囲を、今日、他のプログラム言語でビジネス・アプリケーションを開発しているすべての開発者になじみの深い、言語概念のコア・セットに狭めます。これにより、開発者は素早く容易にテクノロジーを統合できます。

従来のアプリケーション開発環境から切り替えた開発者は、非常に小さな学習曲線に向き合うこととなります。つまり、このプログラミング・モデルを使用することで、早く生産性を上げることができます。また、Service Component Architecture のプログラミング・モデルを使用すると、経験豊富な J2EE 開発者の生産性がさらに向上します。

Service Component Architecture は、いくつかの標準サービス・インプリメンテーション・タイプをサポートします。

- Java オブジェクト。Java クラスをインプリメントします。Java プログラム言語の場合のように、実行時の Java コンポーネントのインスタンスは Java オブジェクトと呼ばれます。
- ビジネス・プロセス・コンポーネント。ビジネス・プロセスをインプリメントします。そのインプリメンテーション言語は、Business Process Execution Language (BPEL) および IBM 拡張です。
- ヒューマン・タスク・コンポーネントは、ビジネス・プロセスまたは統合アプリケーションにおいて、通常は人が実行するタスクを表し、インプリメントします。
- ビジネス・ステート・マシン・コンポーネント。一連の状態を持つ成果物をアプリケーションが処理する場合に使用されます。状態マシンは、特定の時間に成果物が実行できることを定義します。
- ビジネス・ルール・コンポーネント。コンテキストに基づいてビジネス・プロセスの結果を判別するもので、if-then ルール、デシジョン・テーブル、または決定木として設計できます。ビジネス・プロセス内のビジネス・ルールにより、アプリケーションはビジネス条件の変化に迅速に対応できます。ルールはビジネス・プロセス自体から独立しているので、プロセスを再実行せずにもいつでも変更できます。

サービス修飾子は、WebSphere Process Server ランタイム環境のサービス・クライアントとサービス間の対話を管理します。サービス修飾子とは、伝送優先順位、経路の信頼性のレベル、トランザクション管理、およびセキュリティー・レベルのためにアプリケーションに必要な一連の通信特性を定義する、サービス品質仕様で

す。アプリケーションは、サービス修飾子を指定することにより、ランタイム環境へのサービス品質ニーズと通信します。サービス品質修飾子は、WebSphere Integration Developer 内のアセンブリー・エディターでコンポーネントを結合するときに指定できます。WebSphere Process Server 上で実行されている場合、これらの仕様は、ターゲット・コンポーネントとのクライアントの対話方法を決定します。指定された修飾子に応じて、追加の必要処理をランタイムで実行します。

Service Component Architecture 内の機能のインポートおよびエクスポートでは、サービス・モジュールの外部インターフェースまたは WebSphere Process Server のアクセス・ポイントを定義します。インポートおよびエクスポートは、同一アプリケーション内の他のモジュールに対しても、エンタープライズ情報システム (EIS) 上の他のアプリケーションに対しても実行できます。これにより、IBM WebSphere Adapters との連動が可能になります。詳しくは、『インポートおよびエクスポートとアダプター』を参照してください。

WebSphere Process Server ソリューションは、トランザクション、セキュリティ、およびワークロード管理用の、基礎となる WebSphere Application Server 機能に依存して、拡張が容易な統合環境を提供します。

ビジネス・プロセスの場合、WebSphere Process Server は 2 フェーズ・コミット・プロセスを使用して、複数のリソース・マネージャーを含むトランザクション用のサポートを提供し、原子性、一貫性、独立性および耐久性 (ACID) プロパティを保証します。この機能は、短期実行フロー (単一トランザクション) および長期実行フロー (複数トランザクション) の両方に使用できます。ビジネス・プロセス内の複数のステップを 1 つのトランザクションにグループ化するには、WebSphere Integration Developer でトランザクションの境界を変更します。

すべてのサービス起動で 2 フェーズ・コミット・トランザクションがサポートされるわけではないので、WebSphere Process Server にはリカバリー機能も組み込まれています。統合アプリケーションの実行途中で障害が発生した場合、サーバーは障害を検出し、管理者が失敗したイベントを障害のあるイベント・マネージャーから管理できるようにします。

サービス・データ・オブジェクトおよびビジネス・オブジェクト

サービス・データ・オブジェクトおよびビジネス・オブジェクトは、Service Component Architecture で定義されているコンポーネント間に流れるデータを定義します。

Service Data Object (SDO) テクノロジーは、アプリケーションが、異機種混合のデータ・ソースから得られたデータを一様な方法で処理できるようにするオープン・スタンダードです。SDO では J2EE パターンを採用していますが、J2EE データ・プログラミング・モデルを簡略化しています。

IBM WebSphere Application Server 機能の一部であり、WebSphere Process Server に組み込まれている SDO は、データ・アプリケーション開発用のフレームワークを提供します。このフレームワークは、J2EE データ・プログラミング・モデルを単純化します。

WebSphere Process Server には、データ・アクセス・テクノロジーに基づいて、拡張 SDO であるビジネス・オブジェクトが組み込まれています。SDO には、さま

さまざまなデータ (例えば、JDBC ResultSet や XML スキーマで記述されたデータ) を記述するための汎用手段が準備されています。ビジネス・オブジェクトには、統合ソリューションにとって非常に重要ないくつかの拡張機能があり、これらは Service Component Architecture サービス間で交換されるデータを詳細に記述するために使用されます。ビジネス・オブジェクトは、WebSphere Process Server のサービス指向アーキテクチャー (SOA) コアの一部です。

ビジネス・オブジェクト は、ビジネス・エンティティ (従業員など)、データに対するアクション (作成操作や更新操作)、およびデータの処理に対する指示を表す一連の属性です。統合アプリケーションのコンポーネントは、ビジネス・オブジェクトを使用して、情報を交換したりアクションを起動したりします。ビジネス・オブジェクトはさまざまな種類のデータを表すことができるので、柔軟性があります。例えば、従来の統合サーバーのデータ正規化モデルのサポートに加えて、同期 EJB セッション Bean ファサードまたは同期ビジネス・プロセスから戻されたデータを表し、それを IBM WebSphere Portal ポートレットや JSF コンポーネントに結合することができます。

ビジネス・オブジェクトは、ビジネス・エンティティを表す、つまりリテラル・メッセージ定義を文書化するための基本メカニズムであり、スカラー・プロパティを持つ単純な基本オブジェクトから、大規模で複雑なオブジェクトの階層またはグラフまでのすべてを使用可能にします。

WebSphere Process Server では、ビジネス・オブジェクトのフレームワークは、以下のエレメントで構成されます。

- ビジネス・オブジェクト定義
- ビジネス・グラフ定義
- ビジネス・オブジェクト・メタデータ定義
- ビジネス・オブジェクト・サービス (サービス API)

ビジネス・オブジェクト定義は、ビジネス・オブジェクトのタイプを指定する、名前、順序付けられた属性のセット、プロパティ、バージョン番号、およびアプリケーション固有のテキストです。ビジネス・グラフ定義は、ビジネス・グラフ内のビジネス・オブジェクトに関連する変更の要約情報やイベントの要約情報の送達などの追加機能を提供するために、単純なビジネス・オブジェクトまたはビジネス・オブジェクトの階層に追加されるラッパーです。ビジネス・オブジェクト・メタデータ定義は、WebSphere Process Server 上で実行されているときに、ビジネス・オブジェクト定義に追加して、値を拡張することができるメタデータです。このメタデータは、xs:annotation エレメントおよび xs:appinfo エレメントとしてよく知られている、ビジネス・オブジェクトの XML スキーマ定義に追加されます。ビジネス・オブジェクト・サービスは、サービス・データ・オブジェクトにより提供される基本機能のトップにある一連の機能です。例えば、作成、コピー、等価、直列化などのサービスです。

WebSphere Application Server のサービス・データ・オブジェクトについては、WebSphere Application Server Network Deployment の資料を参照してください。

関連概念

16 ページの『ビジネス・オブジェクト・マップ』
ビジネス・オブジェクト・マップとは、各ビジネス・オブジェクト間を関係付ける手段です。

WebSphere Process Server での Common Event Infrastructure

Common Event Infrastructure は、基本的なイベント管理サービスを提供する WebSphere Process Server 内の組み込みテクノロジーの 1 つです。

Common Event Infrastructure のインフラストラクチャーの部分は、基となる IBM WebSphere Application Server 機能の一部として WebSphere Process Server に組み込まれています。イベント発信機能は、WebSphere Process Server の追加機能です。

Common Event Infrastructure (CEI) は、一連の API を実装する、ビジネス、システム、およびネットワークの Common Base Event の作成、伝送、パーシスタンス、および配布のためのインフラストラクチャーです。 *Common Base Event* とは、イベント (ロギング・イベント、トレース・イベント、管理イベント、ビジネス・イベントなど) をビジネス・エンタープライズ・アプリケーションで管理するための仕組みを定義する、XML ベースの仕様のことです。

CEI は、複数の異機種混合ソースからのロー・イベントの統合と持続、およびそれらのイベントのイベント利用者への配布を含む基本のイベント管理サービスを提供します。サービス・コンポーネント・プロセスを表すイベントの生成、伝搬、パーシスタンス、および利用のための機能も提供します。標準の XML ベース・フォーマットの Common Base Event モデルにより、これらのイベントの構造を定義します。サーバーが使用するイベントの各タイプには、特定のタイプのイベントに固有のいくつかの標準のフィールドがあります。場合によっては、イベントには、サービス・コンポーネントによって、特定のイベント・ポイントで使用されるビジネス・オブジェクト・データのカプセル化が含まれていることがあります。

WebSphere Process Server は、CEI 内のイベントをほぼ排他的に使用して、サービス・コンポーネントのモニターを可能にします。イベント関連の機能を使用する場合は CEI サーバーを構成する必要がありますが、構成後に CEI を直接使用することはできません。代わりに、WebSphere Process Server の既存のサービスを使用してください。

WebSphere Process Server では、特別に構成された CEI サーバー (既存のプロセス・サーバーまたは別のサーバーの一部となっている場合もあります) が、あらゆるイベント関連のサービスに使用されます。ユーザーは最初に、イベント・データベース、メッセージング・エンジン、1 つ以上のエンタープライズ・アプリケーション、データベース・ドライバーなどの CEI サーバーが使用するいくつかの機能を作成しデプロイする必要があります。

関連情報

 [Common Event Infrastructure](#)

 [Common Base Event モデル](#)

サポート・サービス

IBM WebSphere Process Server のサポート・サービスは、コンポーネントおよび外部成果物を接続するためのいくつかの変換の問題に対処します。

メディエーション・フロー、インターフェース・マップ、ビジネス・オブジェクト・マップ、リレーションシップ、およびセレクターを使用して、サーバーで稼働するアプリケーションを統合することができます。WebSphere Process Server では、ビジネス・カレンダーを使用することもできます。

1つのコンポーネント・ベースのフレームワークであらゆるスタイルの統合に対応します。

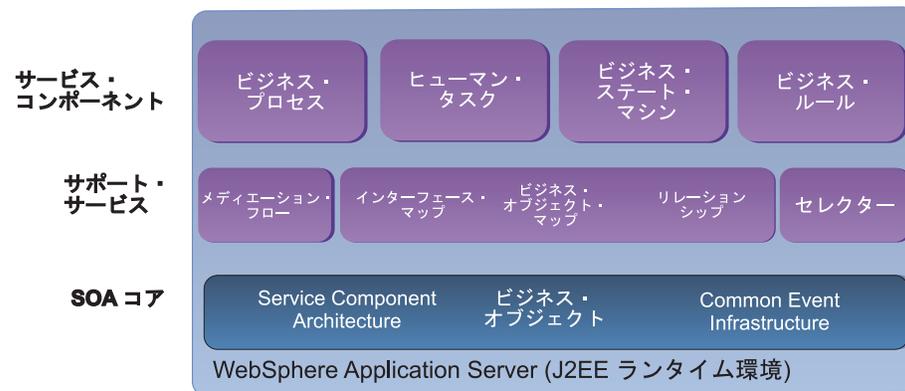


図3. WebSphere Process Server コンポーネント・ベースのフレームワーク

メディエーション・フロー

メディエーション・フローは、既存のサービス（プロバイダー）と、それらのサービスを使用するクライアント（要求側）との間で渡されるメッセージをインターセプトして変更します。

メディエーション・フローは、エクスポートとインポートの間に介在または介入して、メッセージ・ロギング、データ形式変更、ルーティングなどの機能を提供します。メディエーション・フローは、IBM WebSphere Integration Developer で作成され、WebSphere Process Server のメディエーション・モジュールの一部としてデプロイされます。

関連概念

29 ページの『第 8 章 エンタープライズ・サービス・バスのメッセージング・インフラストラクチャー』

WebSphere Process Server には、エンタープライズ・サービス・バス機能があります。WebSphere Process Server は、サービス指向、メッセージ指向、およびイベント・ドリブンの各テクノロジーの統合をサポートすることにより、統合化されたエンタープライズ・サービス・バスにおいて標準ベースのメッセージング・インフラストラクチャーを実現します。

インターフェース・マップ

インターフェース・マップは、異なるインターフェースを持つコンポーネント間の差異を調整します。

インターフェース・マップは、他の Service Component Architecture (SCA) コンポーネントのインターフェース間の差異を解決および調整して、コンポーネント同士が通信できるようにする、WebSphere Process Server のサポート・サービス・コンポーネントです。インターフェース・マップは、IBM WebSphere Integration Developer のモジュール設計者が変換や他の基本操作によって複数インターフェース間の差異を調整できるようにする first-class パターンを取り込みます。インターフェース・マップは、モジュール (SCA モジュールとも呼ばれる) の一部として WebSphere Process Server にデプロイされます。

ビジネス・オブジェクト・マップ

ビジネス・オブジェクト・マップとは、各ビジネス・オブジェクト間を関係付ける手段です。

ビジネス・オブジェクト・マップは、ソース・ビジネス・オブジェクトのサービス・コンポーネントの値に基づいてターゲット・ビジネス・オブジェクトのサービス・コンポーネントに値を割り当てる、IBM WebSphere Process Server のサポート・サービス・コンポーネントです。あるビジネス・オブジェクトがソースになり、別のビジネス・オブジェクトはターゲットになります。ビジネス・オブジェクト・マップでは、このソースとターゲットをマッピングします。ビジネス・オブジェクト・マップでは、各ビジネス・オブジェクト間の 1 対複数、複数対 1、および複数対複数のマッピングがサポートされています。これには、ビジネス・データ、および verb などのビジネス・オブジェクトに関連付けられた側面のマッピングも含まれています。

開発者は IBM WebSphere Integration Developer でビジネス・オブジェクト・マップを作成および編集します。実行時には、このマップによりソースとターゲットの各ビジネス・オブジェクト間でのデータの表現方法が解決されます。WebSphere Process Server では、実行時のマップ・イベントをモニターできます。

関連概念

12 ページの『サービス・データ・オブジェクトおよびビジネス・オブジェクト』

サービス・データ・オブジェクトおよびビジネス・オブジェクトは、Service Component Architecture で定義されているコンポーネント間に流れるデータを定義します。

リレーションシップ

リレーションシップとは、ビジネス・オブジェクトとその他のデータの間の関連のモデル化および保守に使用されるサービスです。

リレーションシップは、複数のデータ型のデータ間の関連を設定する、IBM WebSphere Process Server アプリケーションのサポート・サービスです。

リレーションシップとは、ビジネス・インテグレーション・システムにおける複数のデータ・エンティティー間の関連です。多くの場合、そのエンティティーはビジ

ネス・オブジェクトです。リレーションシップは、各ビジネス・オブジェクト全体で等価であるが、表現が異なるデータの変換に使用されます。

WebSphere Process Server のリレーションシップ・マネージャーは、手動でリレーションシップ・データを操作するためのツールです。このツールで、自動化されたリレーションシップ管理で検出されたエラーを訂正し、より完全なリレーションシップ情報を提供します。特に、このツールには、リレーションシップ・インスタンス・データの変更および検索の機能もあります。リレーションシップ・マネージャーによって、リレーションシップ・ランタイム・データ (参加者およびそのデータを含む) の構成、照会、表示、およびオペレーションの実行が可能になります。リレーションシップの定義は、Relationship Designer を使用して作成します。実行時、リレーションシップのインスタンスには、異なるアプリケーションからの情報を関連付けるデータが取り込まれます。

関連概念

リレーションシップの管理

リレーションシップ・マネージャーは、手動でリレーションシップ・データを制御および操作するためのツールです。このツールで、自動化されたリレーションシップ管理で検出されたエラーを訂正し、より完全なリレーションシップ情報を提供します。特に、このツールには、リレーションシップ・インスタンス・データの検索および変更を行う機能があります。

セレクター

セレクターは、実行時にサービス・コンポーネントの処理時点で柔軟性を提供します。

セレクター (セレクター・コンポーネントとも呼ばれる) とは、1 つの呼び出しで選択基準に基づくさまざまなターゲットの呼び出しを可能にする、IBM WebSphere Process Server の支援サービスです。

セレクター・コンポーネント とは、クライアント・アプリケーションと一連のターゲット・インプリメンテーションの間に動的選択メカニズムを介在させる方法を提供するコンポーネントです。

セレクターにより、ビジネス・ルールにいつもの柔軟性を追加できるようになります。ビジネスの基本的な部分であるビジネス・ルール に基づいて、アプリケーションの一般的な処理や、アプリケーションを介してデータを取得する特定のサービスの呼び出しが行われます。例えば、学校が始まる 2 週間前に、学校に関連する商品の新学期前の特別価格を提供するルールがあるとします。セレクターを使用すると、1 つの呼び出しによって、選択基準に基づいて異なるターゲットを呼び出すことができます。例えば、学校が始まる直前の場合、以前に新学期前に提供した価格が呼び出されます。ただし、シーズンが学校の終了直後である場合、夏休み準備用に提供した価格が呼び出されます。

アプリケーションはいつでも同じものを呼び出すため、移植可能です。ビジネス・ルールは常に変わりません。しかし、セレクターのため、実際の処理が異なります (別のサービス・コンポーネントを呼び出します)。

関連概念

セレクター・コンポーネントの概要

ビジネスの変更に応じて、ビジネスを駆動するビジネス・プロセスも変更する必要があります。プロセスの設計は変更することなく、特定のプロセスが元の設計とは異なる結果を戻すよう変更しなければならない場合もあります。セレクター・コンポーネントは、このような状況に柔軟に対応するためのフレームワークを提供します。

ビジネス・カレンダー

ビジネス・カレンダーは、事業年度の開始、長さ、および区分を定義する時間管理のシステムです。アプリケーションにおけるビジネス・コンポーネントの不連続時間を表します。ビジネス・ユーザーは、営業日、営業時間、および勤務外時間を定義できます。

例えば、組織はビジネス・カレンダーに次の基準を指定して、営業日および休日を定義できます。営業時間は午前 9 時から午後 5 時、営業日は月曜から金曜まで、元日、戦没者記念日、レイバー・デー、感謝祭、クリスマスは休日などです。

ビジネス・カレンダーは WebSphere Integration Developer および WebSphere Modeler で作成され、モジュールの一部として WebSphere Process Server にデプロイされ、そこで XML 成果物として保管されます。ビジネス・カレンダー・モジュールは、単体テスト環境を使用する WebSphere Integration Developer と管理コンソールのいずれかから、他のモジュールと同じ方法で (エンタープライズ・アーカイブ (EAR) ファイルとして) デプロイします。

ビジネス・プロセスでは、ビジネス・カレンダーを使用してタイムアウトをスケジュールできます。ヒューマン・タスクでは、ビジネス・カレンダーを使用して、タスクの有効期限や、削除されたり期限切れになったりする時期をスケジュールできます。

ビジネス・カレンダーには、WebSphere Process Server のビジネス・カレンダー・サービスによってアクセスすることができ、ビジネス・プロセスやヒューマン・タスクなどの WebSphere Process Server コンポーネントで使用できます。

実行時にビジネス・カレンダー・サービスでは、サービスのクライアントであるコンポーネントから指定されたビジネス・カレンダーをロードし、現在時刻とデルタを使用して、コンポーネントの時間を計算します。例えば、3 営業日以内に完了しなければ要求が期限切れと識別されるときに、祝祭日前の 5 月 16 日金曜日に要求が従業員に割り当てられた場合、そのプロセスは、6 日後の 5 月 22 日木曜日まで期限切れになりません。これは、オフィスが閉まっている土曜日、日曜日、および祝祭日の 3 日間が考慮されるからです。

ビジネス・カレンダー・スキーマには、数種類のカレンダーを許容できるだけの十分な柔軟性があります。フラット・カレンダー・モデルでは、メタデータすべてが 1 つのカレンダー・ファイル内に書き込まれます。階層モデルでは、それぞれ完結した複数の小さなカレンダーを作成し、その後他のカレンダーを参照する最上位レベルのカレンダーを作成できます。

ビジネス・カレンダーでは、グリニッジ標準時 (GMT) に基づいてオフセットされる日付を使用します。例えば、ニューヨークでの営業時間 (午前 9 時から午後 5 時) が指定されているカレンダーの場合は、GMT オフセット GMT-5 が設定され、カレンダー・モジュールがカリフォルニア州のサーバーに移された場合でも同じ営業時間が維持されます。フラット・カレンダーでは、すべての日付で同じオフセットが使用されます。他のカレンダーを参照する階層カレンダーでは、個々のカレンダーで別々の GMT オフセットを指定できます。

ビジネス・カレンダーのスコープは、カレンダーがデプロイされるモジュールです。

ビジネス・カレンダー・モデルは、Internet Engineering Task Force (IETF) の iCalendar 仕様 (RFC 2445) に基づいています。このモデルは iCalendar 標準の XML 準拠表現である xCal に基づいており、他のカレンダー・システムと連動できます。

ビジネス・カレンダーの作成について詳しくは、WebSphere Integration Developer 資料の『ビジネス・カレンダーでの作業』を参照してください。

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。

Service Component Architecture (SCA) では、サービス・コンポーネント (SCA コンポーネントとも呼ばれる) はサービス・インプリメンテーションを定義します。各サービス・コンポーネントはインターフェースを持ち、相互にワイヤリングすることにより、WebSphere Process Server にデプロイするモジュールを形成することができます。

これにより柔軟なランタイム環境が作成され、アプリケーションの一部を変更する際に他の部分に影響を与えずに済みます。例えば、承認を表すヒューマン・タスクを、ビジネス・プロセスやビジネス・プロセスの呼び出し元を変更せずに、自動承認を表すビジネス・ルールに置き換えることができます。このようにするには、アセンブリー・ダイアグラム内のサービス・コンポーネントを置き換えるだけで済みます。

サービス・コンポーネントは、以下のプログラミング構成体を使用して、既存のアプリケーションと対話します。

- Java Bean
- Enterprise Java Beans
- Web サービス
- JMS メッセージ

また、サービス・コンポーネントは IBM WebSphere Adapters を使用して、エンタープライズ情報システム (EIS) 上の他のアプリケーションと対話できます。

WebSphere Process Server は、サポート・サービスのランタイム・インフラストラクチャーおよびサービス指向アーキテクチャー・コアを基盤として、すぐに使用で

きる各種の SCA コンポーネントを提供します。これらのコンポーネントは統合アプリケーションで使用できます。

1つのコンポーネント・ベースのフレームワークであらゆるスタイルの統合に対応します。

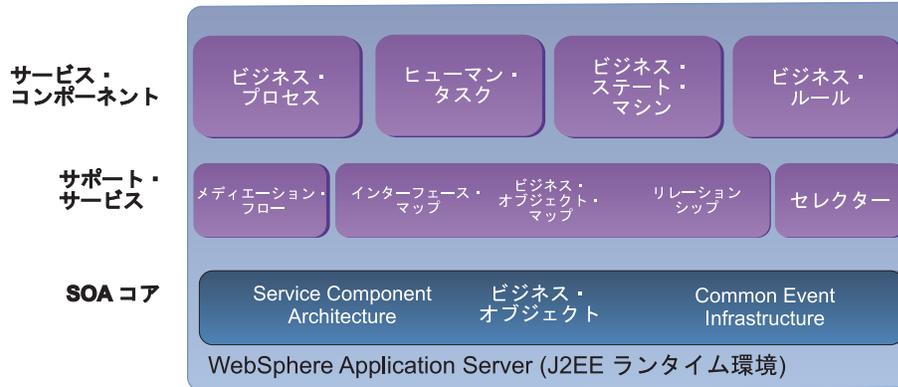


図 4. WebSphere Process Server コンポーネント・ベースのフレームワーク

ビジネス・プロセス

ビジネス・プロセスは、エンタープライズ・サービス統合の主な手段を提供するサービス・コンポーネントです。

ビジネス・プロセスは、より大きなビジネス・ゴールを達成するために組織が使用するシステムまたは手順です。ビジネス・プロセスを分解すると、実際は一連の個別タスクであり、各タスクは特定の順序で実行されることがわかります。IBM WebSphere Process Server で稼働するアプリケーションに不可欠な部分として、ビジネス・プロセスはエンタープライズ・サービス統合の主な手段を提供します。

ビジネス・プロセス・コンポーネントは、完全にサポートされる Web Services BPEL (Business Process Execution Language) エンジンを実装します。WebSphere Process Server には、WebSphere Application Server の上にあるビジネス・プロセス・コレオグラフィー・エンジンが含まれます。拡張性の高いインフラストラクチャーにおいて、長期実行および短期実行のビジネス・プロセスに対する高度なサポートを持つ単純な開発モデルで、複雑なビジネス・プロセスを開発およびデプロイすることができます。WebSphere Integration Developer で BPEL モデルを作成することも、WebSphere Business Modeler で作成したビジネス・モデルから BPEL モデルをインポートすることもできます。

Web Services Business Process Execution Language (BPEL) が使用され、ビジネス・プロセス・フローのコレオグラフィーを行います。ビジネス・プロセス統合サービスは BPEL4WS バージョン 1.1 を基盤に構築され、さらに来るべき WS-BPEL バージョン 2.0 仕様の主要機能が追加されています。

関連情報

 [ビジネス・プロセスについて](#)

ヒューマン・タスク

ヒューマン・タスクはサービス・コンポーネントであり、従業員への作業の割り当てやその他のサービスの起動に使用できます。

ヒューマン・タスクとは、人間によって実行される作業単位であり、他のサービスとの相互作用が関係することが多いため、大きなビジネス・ゴールでのタスクになります。

Human Task Manager (WebSphere Process Server で使用可能) は、実行時におけるタスクの作成および追跡をサポートします。ユーザー情報およびグループ情報へのアクセスには、(オペレーティング・システムのリポジトリおよび WebSphere ユーザー・レジストリーのほかに) 既存の LDAP ディレクトリーを使用できます。WebSphere Process Server では、E メール通知を含むヒューマン・タスクのマルチレベル・エスカレーションがサポートされます。また、ヒューマン・タスクを管理するための Web クライアントと、カスタム・クライアントの作成やヒューマン・タスク機能の他の Web アプリケーションへの組み込みに使用できる、一連の Java Server Faces (JSF) コンポーネントも組み込まれています。

ロール・ベースのタスク割り当て、起動、およびエスカレーションを可能にする、ヒューマン・タスク・サービス・コンポーネント。

関連情報

 ヒューマン・タスク

ビジネス・ステート・マシン

ビジネス・ステート・マシンとは、順次のビジネス・プロセス・モデルではなく、状態およびイベントに基づいてビジネス・プロセスを表すことができるようにするサービス・コンポーネントです。

ビジネス・ステート・マシンでは、各イベントに対応してオブジェクトや対話が遂行する一連の状態、応答、およびアクションを指定します。

ビジネス・ステート・マシンは、IBM WebSphere Integration Developer で作成して編集し、実行時には IBM WebSphere Process Server でモニターします。

関連資料

 ビジネス・ステート・マシン・イベント

ビジネス・ステート・マシン・コンポーネントで使用可能なイベント・タイプをリストします。

ビジネス・ルール

ビジネス・ルールは、ビジネス内で満たされなければならないポリシーまたは条件を宣言するサービス・コンポーネントです。

ビジネス・ルールは、ビジネス・アクティビティーに対してビジネス・ポリシーや手法を適用する方法を示しています。ビジネス・ルールでは、ビジネス手法での振

る舞いを制御したり、それに構造を適用したりします。ルールは、ビジネス・ポリシーの適用、組織内での共通のガイドラインの確立、ビジネス環境でのアクセスの制御を実行できます。

ビジネス・プロセスの柔軟性を高めるビジネス・ルール
ビジネス・ルールは、コンテキストに基づいてプロセスの結果を決定するので、ビジネス・プロセス内でビジネス・ルールを使用するとアプリケーションはビジネス条件の変化に迅速に対応できます。

ビジネス・ルールのオーサリングは、IBM WebSphere Integration Developer でサポートされています。IBM WebSphere Process Server には Business Rules Manager が含まれています。これは、ビジネス・アナリストがビジネス・ニーズに応じてビジネス・ルールを更新するための Web ベースのランタイム・ツールであり、ルールを更新しても、他のコンポーネントまたは Service Component Architecture (SCA) サービスに影響を及ぼしません。

関連概念



ビジネス・ルールの概要

ビジネス・ルールを使用して、ビジネス・プラクティスの振る舞いを制御します。

第 5 章 インポートおよびエクスポートとアダプター

Service Component Architecture 内の機能のインポートおよびエクスポートでは、サービス・モジュールの外部インターフェースまたは WebSphere Process Server のアクセス・ポイントを定義します。インポートおよびエクスポートは、同一アプリケーション内の他のモジュールに対しても、エンタープライズ情報システム (EIS) 上の他のアプリケーションに対しても実行できます。

インポート は、モジュールの外側のサービスを識別し、モジュール内からそれらのサービスを呼び出せるようにします。エクスポート は、モジュール内のコンポーネントが、サービスを外部のクライアントに提供できるようにします。

インポートまたはエクスポートでは、モジュールが他のモジュールにアクセスすることができ、アプリケーションが EIS システム上のアプリケーションに、ローカル・コンポーネントと同様にアクセスすることができます。これにより、IBM WebSphere Adapters との連動が可能になります。

WebSphere Adapters では、EIS の統合に対するサービス指向のアプローチを提供します。WebSphere Adapters は J2EE コネクター・アーキテクチャー (JCA 1.5) に準拠しています。JCA は、EIS の接続性のための J2EE 標準です。EIS のインポートおよび EIS のエクスポートでは、SCA コンポーネントに対して、モジュール外部のサービスの統一した表示を提供します。これにより、各コンポーネントが一貫性のある SCA プログラミング・モデルを使用して、さまざまな外部 EIS システムと通信することができます。インポートされた RAR ファイルを元に、WebSphere Integration Developer 内で WebSphere Adapters がアセンブルされ、その後エンタープライズ・アーカイブ (EAR) ファイルとしてエクスポートされて、WebSphere Process Server 上でデプロイされます。

WebSphere Adapter には以下のものが含まれます。

- IBM WebSphere Adapter for E-mail バージョン 6.1
- IBM WebSphere Adapter for FTP バージョン 6.1
- IBM WebSphere Adapter for Flat Files バージョン 6.1
- IBM WebSphere Adapter for JDBC バージョン 6.1
- IBM WebSphere Adapter for SAP Software バージョン 6.1
- IBM WebSphere Adapter for Siebel Business Applications ビジネス 6.1
- IBM WebSphere Adapter for Oracle E-Business Suite バージョン 6.1
- IBM WebSphere Adapter for JD Edwards EnterpriseOne バージョン 6.1

WebSphere Adapters は、WebSphere Integration Developer に組み込まれているコンポーネントです。

WebSphere Adapters について詳しくは、IBM WebSphere Business Process Management インフォメーション・センターの WebSphere Integration Developer の資料を参照してください。

インポートおよびエクスポートにはバインディング情報が必要であり、この情報によって、モジュールからのデータのトランスポート手段が指定されます。WebSphere Integration Developer のアセンブリー・エディターは、インポートおよびエクスポートのセットアップ、サポートされるバインディングのリスト、およびそれらの作成の単純化を行います。バインディング情報は、プロパティ・ビューに表示されます。

第 6 章 WebSphere Process Server でのデプロイメント環境

WebSphere Process Server により、Service Component Architecture (SCA) モジュールのデプロイメント環境をサーバーのコレクションの 1 つとして管理できます。WebSphere Process Server に組み込まれている WebSphere Application Server Network Deployment 機能により、このサーバーのコレクションのエレメントが提供されています。

WebSphere Process Server 環境には相互接続されるサーバーのレイアウト (つまりトポロジー) が含まれていて、Business Process Choreographer、ビジネス・ルール、メディエーション、およびリレーションシップなどの WebSphere Process Server SCA モジュールがサポートされます。このトポロジーは、1 つのコンピューター・システムで実行されている 1 つのサーバー・プロセスで構成されるか、または複数のコンピューター・システムで実行されている複数のサーバー・プロセスで構成されます。サーバー・プロセス とは、SCA モジュールとしてデプロイされるコンポーネントのランタイム環境です。WebSphere Process Server などの WebSphere 製品の場合、サーバー・プロセスとは Java 仮想マシン (JVM) です。

ご使用の環境が、1 つのシステム上の 1 つのサーバー・プロセスから構成される場合、セットアップされているそのサーバー・プロセスをスタンドアロン・サーバーと呼びます。スタンドアロン・サーバーには他のサーバー・プロセスとの相互接続がないため、容量がその 1 つのコンピューター・システム上のリソースに限られ、フェイルオーバー・サポートは組み込まれていません。また、セットアップが最も簡単な環境でもあります。

ユーザーの環境が複数のサーバー・プロセスで構成されている場合は、セル内にクラスター環境をセットアップすると考えられます。セルは分散コンピューティング環境の管理ドメインで、SCA モジュールと、SCA モジュールのサポートに必要なリソースで構成されます。デプロイメント環境 とは、各サーバー・プロセス (通常はさまざまな物理的コンピューター・システムにある) が総合して管理される環境です。1 つのデプロイメント・マネージャーで複数のデプロイメント環境を管理できます。

クラスターでデプロイメント環境を使用すると、以下の利点があります。

- 簡単な管理: SCA モジュールを構成する 1 つのビュー、その SCA モジュールをサポートするサーバー・プロセスのビュー、および始動、停止、作成、削除などの SCA モジュールのランタイム・アクションを制御する 1 つのポイントが提供されます。
- ワークロード・バランシング: クラスターでは、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、クラスター内のサーバー間でアプリケーションの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: アプリケーションをサポートするクラスター・メンバーとしてサーバー・ハードウェアを追加構成することにより、アプリケーションに処理能力を増すことができます。

- アプリケーションの可用性: サーバーに障害が起こった場合、アプリケーションがクラスター内の他のサーバーで引き続き作業を処理するため、アプリケーション・ユーザーに影響を及ぼすことなくリカバリー作業を進めることができます。
- 保守容易性: 計画的な保守のために、アプリケーションの処理を停止させることなくサーバーを停止することができます。
- 柔軟性: 管理コンソールを使用して、機能を必要に応じて加えたり除去したりすることができます。

第 7 章 WebSphere が提供するビジネス・スペース

WebSphere Process Server WebSphere が提供するビジネス・スペース が含まれています。これは IBM WebSphere Business Process Management ポートフォリオ横断の Web インターフェースを作成、管理および統合するための共通インターフェースをアプリケーション・ユーザーに提供します。

ビジネス・スペース は、ブラウザー・ベースのグラフィカル・ユーザー・インターフェースで、これを使用してアプリケーション・ユーザーは WebSphere Business Process Management ポートフォリオの製品のコンテンツをカスタマイズできます。アプリケーション・ユーザー (ビジネス・ユーザー) は、 WebSphere Process Server にデプロイされたアプリケーションのユーザーです。

ビジネス・スペースは、マッシュアップ技術を使用する非同期 JavaScript™ および XML (AJAX) インターフェースを備えており、これによりビジネス・ユーザーは、ヒューマン・タスク中心のユーザー・エクスペリエンスを作成およびカスタマイズできます。マッシュアップは、Web アプリケーション (ウィジェット) を組み合わせることによって作成される Web ページで、Web コンテンツを混合して新しいインターフェースを作成します。ヒューマン・タスク・ウィジェットは、JavaScript Object Notation (JSON) データ・フォーマットによる Representational State Transfer (REST) 対話を使用して、WebSphere Process Server ランタイムと通信します。

ビジネス・ユーザーは、ビジネス・スペースのウィジェットをカスタマイズし、それぞれの設定に従ってランタイム・ビジネス・データを表示できます。ビジネス・スペース管理者は、新規スペース (事前に構成されたページの集合) を作成し、ビジネス・スペース に付属の事前定義シナリオに加えてマッシュアップ・ページ・コンテンツを作成できます。例えばこれらのマッシュアップは、さまざまなタスクへの担当者の割り当てや、異なる結果に応じたビジネス・ルールの調整など、エンタープライズでの特定のビジネス・ニーズに応えます。

ビジネス・スペース 共通フレームワークは、 WebSphere Process Server、 WebSphere Business Monitor、および WebSphere Business Modeler Publishing Server に付属しています。事前定義シナリオ用のテンプレートは、各製品のインストール時に ビジネス・スペース に取り込まれます。ビジネス・スペース は、ビジネス・ユーザーが表示および変更できる WebSphere Service Fabric の情報も含んでいます。次の図は、ビジネス・スペース フレームワークと WebSphere Business Process Management ポートフォリオの製品との関係を表しています。

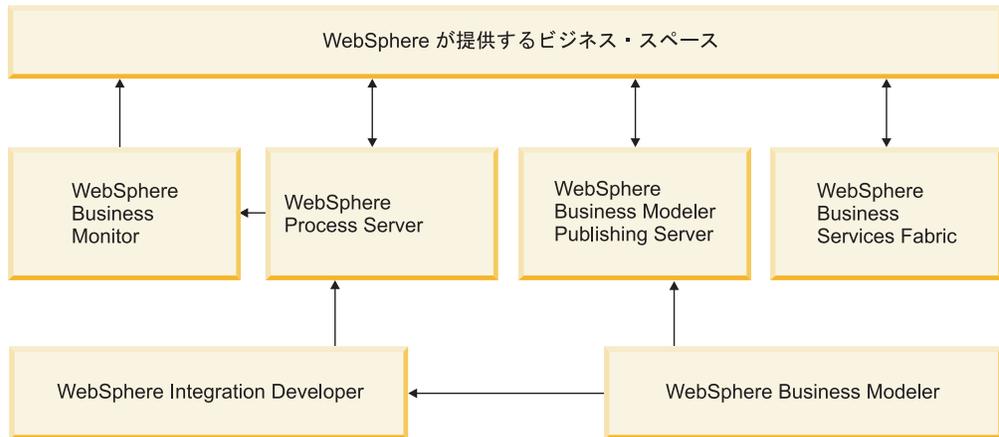


図5. ビジネス・スペース および Business Process Management 製品

WebSphere Process Server アプリケーションの場合、ビジネス・スペース は、ビジネス・ユーザーがヒューマン・タスクとビジネス・ルールを処理するためのページを提供します。ビジネス・ユーザーは、表示および変更可能なヒューマン・タスク・データとビジネス・ルールでマッシュアップを作成できます。

WebSphere Process Server によって提供されるスペースには、ビジネス・ユーザー自身のタスク、作成したタスク、所属するチームのメンバーのタスク、使用可能なタスク、およびタスク情報を表示する、ビジネス・ユーザーのためのヒューマン・ワークフローの各ウィジェットが組み込まれています。それらのウィジェットにより、タスクを作成し、同じビジネス・プロセスに属するタスクの状況を示すヒューマン・ワークフローの図を表示できます。ビジネス・スペース には、ビジネス・ルールを処理するウィジェットを 1 つ含むビジネス構成ページも組み込まれています。ビジネス・スペース には、WebSphere Process Server アプリケーションに関連するウィジェットの他に、Google ツールのウィジェットや、プレゼンテーション、文書、スプレッドシート、Web サイト、および RSS フィードを表示するためのウィジェットも組み込まれています。

ビジネス・スペース フレームワークは、WebSphere Process Server と共にインストールされ、ビジネス・スペース は、スタンドアロン・サーバー・プロファイル用の Profile Management Tool を使用して構成できます。ヒューマン・ワークフローのウィジェットを使用するには、Business Process Choreographer を構成する必要があります。詳しくは、WebSphere Process Server インフォメーション・センターの『Business Process Choreographer の構成』を参照してください。ビジネス・スペース プロファイルでデプロイメント環境を使用している場合、管理コンソールの ビジネス・スペース 構成ページを使用してビジネス・スペース を構成する必要があります。

第 8 章 エンタープライズ・サービス・バスのメッセージング・インフラストラクチャー

WebSphere Process Server には、エンタープライズ・サービス・バス機能があります。WebSphere Process Server は、サービス指向、メッセージ指向、およびイベント・ドリブンの各テクノロジーの統合をサポートすることにより、統合化されたエンタープライズ・サービス・バスにおいて標準ベースのメッセージング・インフラストラクチャーを実現します。

エンタープライズ・アプリケーションで使用可能なエンタープライズ・サービス機能では、サービスの対話を円滑にするために、トランスポート層だけでなく、メディエーション・サポートも提供されます。エンタープライズ・サービス・バスは、オープン・スタンダードおよびサービス指向アーキテクチャー (SOA) を中心に構築されます。この機能は、IBM WebSphere Application Server Network Deployment

で提供される堅固な J2EE 1.4 インフラストラクチャーおよび関連プラットフォーム・サービスに基づいています。WebSphere Process Server には、IBM WebSphere Enterprise Service Bus で利用可能なテクノロジーと同じテクノロジーが採用されています。この機能は、WebSphere Process Server の基盤機能であるため、この機能を利用するために WebSphere Enterprise Service Bus 用の追加ライセンスを取得する必要はありません。

ただし、WebSphere Enterprise Service Bus の追加スタンドアロン・ライセンスを企業にデプロイすることで、WebSphere Process Server によるプロセス統合ソリューションの接続性の範囲を拡大することができます。例えば、WebSphere Enterprise Service Bus を SAP アプリケーションのさらに近くにインストールして、IBM WebSphere Adapter for SAP をホストしたり、SAP メッセージを変換してからネットワーク間でその情報を、WebSphere Process Server によって連携可能になったビジネス・プロセスに送信したりすることができます。

関連概念

15 ページの『メディエーション・フロー』

メディエーション・フローは、既存のサービス (プロバイダー) と、それらのサービスを使用するクライアント (要求側) との間で渡されるメッセージをインターセプトして変更します。



メッセージ・サービス・クライアント

WebSphere Process Server は、Java 以外のアプリケーションを エンタープライズ・サービス・バス に接続できる Message Service Client for C/C++ および Message Service Client for .NET を備えています。

30 ページの『メディエーション・モジュール』

メディエーション・モジュールは、サービス要求のフォーマット、内容、またはターゲットを変更できる Service Component Architecture (SCA) モジュールです。

34 ページの『メディエーション・プリミティブ』

メディエーション・コンポーネントは、サービス・コンポーネント間のメッセージ・フローを操作します。メディエーション・コンポーネントの機能は、標準サービス・インプリメンテーション・タイプをインプリメントするメディエーション・プリミティブ によってインプリメントされます。

38 ページの『サービス・メッセージ・オブジェクト』

Service message objects (SMO) は、サービス間で交換されるメッセージを処理および操作するための抽象化層を提供します。

メッセージ・サービス・クライアント

WebSphere Process Server は、Java 以外のアプリケーションを エンタープライズ・サービス・バス に接続できる Message Service Client for C/C++ および Message Service Client for .NET を備えています。

Message Service Client for C/C++ および .NET は、Java Message Service (JMS) API と同じインターフェースのセットを持つ XMS という API を提供します。Message Service Client for C/C++ には 2 つの XMS インプリメンテーションがあり、一方は C アプリケーション用、もう一方は C++ アプリケーション用です。Message Service Client for .NET には、完全に管理された XMS インプリメンテーションがあり、これは任意の .NET 対応言語で使用できます。

WebSphere Application Server Network Deployment から J2EE クライアント・サポート (Web サービス・クライアント、EJB クライアント、および JMS クライアントを含む) をインストールして使用することもできます。

関連概念

エンタープライズ・サービス・バスのメッセージング・インフラストラクチャー WebSphere Process Server には、エンタープライズ・サービス・バス機能があります。WebSphere Process Server は、サービス指向、メッセージ指向、およびイベント・ドリブンの各テクノロジーの統合をサポートすることにより、統合化されたエンタープライズ・サービス・バスにおいて標準ベースのメッセージング・インフラストラクチャーを実現します。

メディエーション・モジュール

メディエーション・モジュールは、サービス要求のフォーマット、内容、またはターゲットを変更できる Service Component Architecture (SCA) モジュールです。

メディエーション・モジュールは、サービス要求元とサービス・プロバイダー間で伝送中のメッセージを操作します。メッセージを異なるサービス・プロバイダーにルーティングでき、またメッセージの内容または形式を修正することもできます。メディエーション・モジュールは、ユーザーの要件に合わせて調整したメッセージ・ロギングやエラー処理などの機能を提供できます。

メディエーション・モジュールの一定の側面を、モジュールを再デプロイすることなく、WebSphere Process Server 管理コンソールから動的に変更できます。

メディエーション・モジュールのコンポーネント

メディエーション・モジュールには、以下のアイテムが含まれています。

- **インポート:** SCA モジュールとサービス・プロバイダーの間の対話を定義します。これにより、SCA モジュールは、外部サービスをローカル・サービスのよう呼び出すことができます。WebSphere Process Server を使用してメディエーション・モジュール・インポートを表示し、バインディングを変更できます。
- **エクスポート:** SCA モジュールとサービス要求元の間の対話を定義します。これにより、SCA モジュールでサービスを提供し、SCA モジュールの外部インターフェース (アクセス・ポイント) を定義することが可能になります。メディエーション・モジュール・エクスポートは、WebSphere Process Server から表示できません。
- **SCA コンポーネント:** メディエーション・モジュールなどの SCA モジュールを構築するための基本単位です。SCA モジュールとコンポーネントは、WebSphere Integration Developer を使用してグラフィカルに作成およびカスタマイズできます。メディエーション・モジュールのデプロイ後に、WebSphere Process Server の管理コンソールから、モジュールを再デプロイせずにその属性をカスタマイズできます。

一般に、メディエーション・モジュールは、メディエーション・フロー・コンポーネント と呼ばれる特定のタイプの SCA コンポーネントを含みます。メディエーション・フロー・コンポーネントは、メディエーション・フローを定義します。1 つのメディエーション・モジュールに含めることができるメディエーション・フロー・コンポーネントは 1 つのみです。

メディエーション・フロー・コンポーネントには、メディエーション・プリミティブを 1 つまたは複数含めることができますが、まったく含めないことも可能です。WebSphere Process Server では、メッセージ・ルーティングおよび変換のための機能を提供する、あらかじめ用意された 1 組のメディエーション・プリミティブがサポートされます。メディエーション・プリミティブをより柔軟に使用する必要がある場合は、カスタム・メディエーション・プリミティブを使用して、カスタム・ロジックを呼び出すことができます。

メディエーション・フロー・コンポーネントが含まれていないメディエーション・モジュールには、サービス要求をあるプロトコルから別のプロトコルに変換するという目的があります。例えば、SOAP/JMS を使用して作成したサービス要求を、送信前に SOAP/HTTP に変換しなければならない場合があります。

注: WebSphere Process Server から、メディエーション・モジュールを表示して、一定の変更を加えることができます。ただし、WebSphere Process Server モジュールの内部から SCA コンポーネントを表示あるいは変更することはできません。SCA コンポーネントをカスタマイズするには、WebSphere Integration Developer を使用してください。

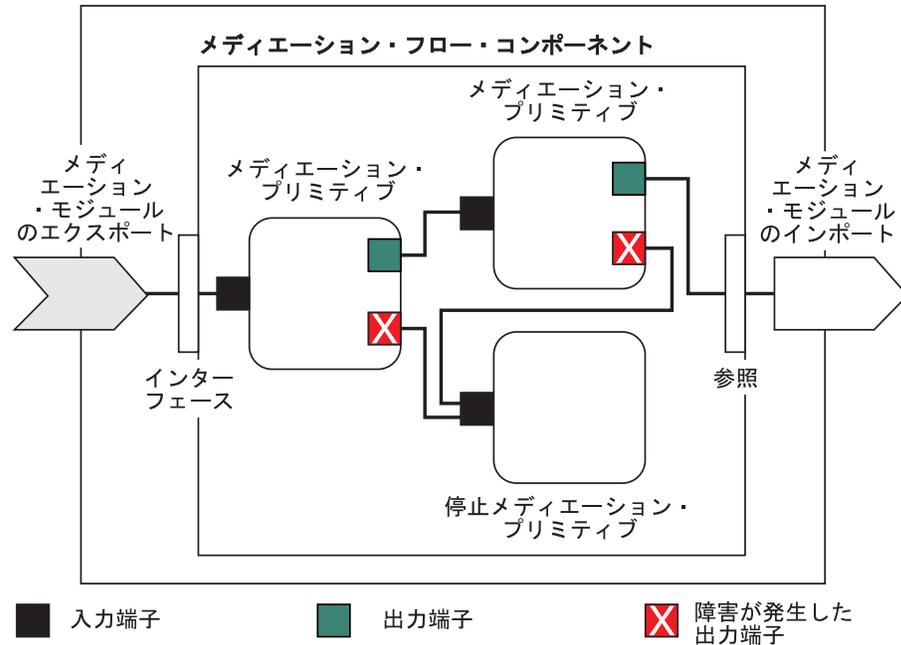


図6. メディエーション・モジュールの簡単な例： メディエーション・モジュールには、メディエーション・プリミティブが含まれた 1 つのメディエーション・フロー・コンポーネントが含まれています。

• プロパティー

メディエーション・プリミティブはプロパティーを持ちます。一部のプロパティーは、SCA モジュールの追加のプロパティーとして管理コンソールに表示させることができます。

メディエーション・プリミティブのプロパティーを WebSphere Process Server 管理コンソールで表示するには、統合開発者はこれらのプロパティーにプロモート済みというフラグを立てる必要があります。一部のプロパティーは管理上の構成に適しています。これらのプロパティーは統合サイクルから管理サイクルへプロモート可能であるため、WebSphere Integration Developer はこれらのプロパティーをプロモート可能なプロパティーとして記述します。他のプロパティーは、変更するとメディエーション・フローに影響が生じ、メディエーション・モジュールの再デプロイが必要になるおそれがあるため、管理構成には適しません。

WebSphere Integration Developer は、メディエーション・プリミティブのプロモートされるプロパティーの下に、プロモート対象として選択できるプロパティーのリストを示します。

WebSphere Process Server 管理コンソールを使用すると、メディエーション・モジュールの再デプロイや、サーバーまたはモジュールの再始動を行うことなく、プロモートされるプロパティーの値を変更できます。

プロパティーの変更がデプロイメント・マネージャー・セル内で行われた場合を除き、新たに呼び出されたメディエーション・フローでは、この変更内容がただちに使用されます。デプロイメント・マネージャー・セル内に変更が生じた場合、各ノードで同期化が行われるとそのノード上で変更が有効になります。処理中のメディエーション・フローでは、前の値が引き続き使用されます。

注: プロパティ値ではなく、メディエーション・プリミティブのプロパティ名およびタイプを変更する場合は、WebSphere Integration Developer を使用してください。

メディエーション・モジュールのデプロイ

メディエーション・モジュールは、WebSphere Integration Developer を使用して作成され、一般に WebSphere Process Server のエンタープライズ・アーカイブ (EAR) ファイル内にデプロイされます。

プロモートされるプロパティの値は、デプロイメント時に変更できます。

WebSphere Integration Developer からメディエーション・モジュールをエクスポートすると、WebSphere Integration Developer にメディエーション・モジュールを Java アーカイブ (JAR) ファイル内にパッケージ化させ、その JAR ファイルを EAR ファイル内にパッケージ化させることができます。この後、管理コンソールから新規アプリケーションをインストールすることにより、EAR ファイルをデプロイできます。

メディエーション・モジュールは 1 つのエンティティと考えられます。ただし、SCA モジュールは、1 つの JAR ファイルに保管されたいくつかの XML ファイルによって定義されます。

メディエーション・モジュールが含まれる EAR ファイルの例



図7. メディエーション・モジュールが含まれる EAR ファイルの簡単な例: EAR ファイルには JAR が含まれます。ユーティリティ JAR ファイルにはメディエーション・モジュールが含まれます。

関連概念

エンタープライズ・サービス・バスのメッセージング・インフラストラクチャー WebSphere Process Server には、エンタープライズ・サービス・バス機能があります。WebSphere Process Server は、サービス指向、メッセージ指向、およびイベント・ドリブンの各テクノロジーの統合をサポートすることにより、統合化されたエンタープライズ・サービス・バスにおいて標準ベースのメッセージング・インフラストラクチャーを実現します。

メディエーション・プリミティブ

メディエーション・コンポーネントは、サービス・コンポーネント間のメッセージ・フローを操作します。メディエーション・コンポーネントの機能は、標準サービス・インプリメンテーション・タイプをインプリメントするメディエーション・プリミティブによってインプリメントされます。

メディエーション・コンポーネントには、1 つ以上のフローがあります (例、要求用と応答用に 1 つずつ)。

WebSphere Process Server は、提供されたメディエーション・プリミティブのセットをサポートしており、このメディエーション・プリミティブ・セットは、WebSphere Process Server にデプロイされたメディエーション・モジュールの標準メディエーション機能をインプリメントします。特殊なメディエーション機能が必要な場合は、ユーザー独自のカスタム・メディエーション・プリミティブを開発できます。

メディエーション・プリミティブは、サービス・メッセージ・オブジェクト (SMO) で表されるメッセージを処理するかまたは取り扱う 1 つの「入力」操作を定義します。メディエーション・プリミティブは、メッセージを別のコンポーネントまたはモジュールに送信する「出力」操作も定義します。

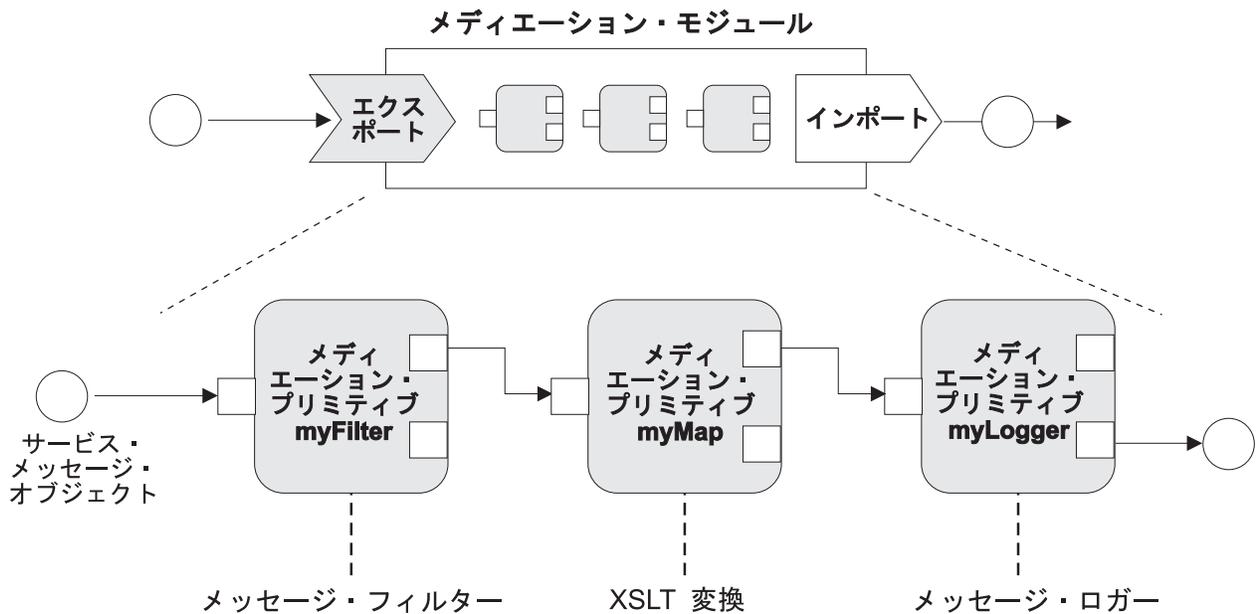


図 8. 3 つのメディエーション・プリミティブから成るメディエーション・モジュール

WebSphere Integration Developer を使用すると、メディエーション・プリミティブを構成して、そのプロパティを設定できます。これらのプロパティの一部は、プロモートを行うことにより、ランタイム管理者に対して表示することができます。

WebSphere Integration Developer を使用すると、メディエーション・プリミティブを基にしてメディエーション・コンポーネントをグラフィカルにモデル化して組み立てることと、メディエーション・コンポーネントを基にメディエーション・モジュールを組み立てることができます。

サポートされるメディエーション・プリミティブ

以下のメディエーション・プリミティブ・セットが WebSphere Process Server でサポートされています。

ビジネス・オブジェクト・マップ

メッセージを変換します。

- 再利用可能なビジネス・オブジェクト・マップを使用して、メッセージ変換を定義します。
- ビジネス・オブジェクト・マップ・エディターを使用して、メッセージ変換をグラフィカルに定義できます。
- メッセージの内容を変更できます。
- 入力メッセージ・タイプを別の出力メッセージ・タイプに変換できます。

カスタム・メディエーション

独自のメディエーション・ロジックを Java コードに実装できます。カスタム・メディエーション・プリミティブは、ユーザー定義メディエーション・プリミティブの柔軟性と、事前定義メディエーション・プリミティブの簡素性を兼ね備えています。以下の方法により、複雑な変換およびルーティング・パターンを作成できます。

- Java コードの作成。
- 独自プロパティの作成。
- 新規端末の追加。

カスタム・メディエーション・プリミティブからサービスを呼び出すことは可能ですが、サービス起動メディエーション・プリミティブはサービスを呼び出す目的で設計されており、再試行などの追加機能を提供します。

データベース・ルックアップ

ユーザーが提供するデータベースからの情報を使用して、メッセージを変更します。

- データベース・ルックアップ・メディエーション・プリミティブを使用できるように、データベース、データ・ソース、およびサーバー認証の設定をセットアップする必要があります。これを簡単に実行するには、管理コンソールを使用します。
- データベース・ルックアップ・メディエーション・プリミティブは、1 つのテーブルからのみ読み取ることができます。
- 指定されたキー列には、固有な値が含まれていなければなりません。
- 値列のデータは、Java プリミティブまたは Java ストリング (あるいは Java プリミティブまたは Java ストリングにキャスト可能) でなければなりません。

エンドポイント・ルックアップ

リポジトリ内でサービス・エンドポイントを検索することにより、要求の動的ルーティングを可能にします。

- サービス・エンドポイント情報は、ローカルまたはリモートの WebSphere Service Registry and Repository (WSRR) から取得されます。
- レジストリーの変更を WSRR 管理コンソールから行います。
- WebSphere Process Server は、どのレジストリーを使用するかを認識する必要があるため、WebSphere Process Server 管理コンソールを使用して WSRR アクセス定義を作成する必要があります。

イベント・エミッター

イベントをメディエーション・フロー・コンポーネント内から送信できるようにして、モニター機能を拡張します。

- この後、WebSphere Process Server 上の Common Base Events (CBE) ブラウザーを使用してイベント・エミッターのイベントを表示することができます。
- パフォーマンス上の理由から、メディエーション・フロー内の重要な点でのみイベントを送信するようにします。
- イベントに記録されるメッセージの一部分を定義できます。
- イベントは Common Base Event の形で、Common Event Infrastructure サーバーに送信されます。
- イベント・エミッター情報を十分活用するには、イベント・コンシューマーが Common Base Event の構造を理解しておく必要があります。Common Base Events には包括的なスキーマがありますが、このスキーマでは拡張データ・エレメントに含まれるアプリケーション固有のデータはモデル化されません。拡張データ・エレメントをモデル化するために、WebSphere Integration Developer のツール群により、構成済みのイベント・エミッター・メディエーション・プリミティブ ごとに Common Event Infrastructure イベント・カタログ定義ファイルが生成されます。イベント・カタログ定義ファイルは、ユーザーのサポート用に提供されるエクスポート成果物ですが、これらは WebSphere Integration Developer や WebSphere Process Server ランタイムでは使用されません。イベント・カタログ定義ファイルは、イベント・エミッターのイベントを利用するアプリケーションを作成するときに参照する必要があります。
- WebSphere Process Server から、その他のモニターを指定することができます。例えば、インポートおよびエクスポートから送信するイベントをモニターすることができます。

障害 フロー内の特定のパスを停止して、例外を生成します。

ファンイン

メッセージを集約 (結合) するのに役立ちます。

- ファンアウト・メディエーション・プリミティブと組み合わせた場合のみ使用できます。
- ファンアウト・メディエーション・プリミティブとファンイン・メディエーション・プリミティブを組み合わせて使用すると、データを 1 つの出力メッセージに集約できます。

- ファンイン・メディエーション・プリミティブは、決定点に到達するまで複数のメッセージを受信し、1つのメッセージを出力します。
- 集約データを保持するには、共用コンテキストを使用します。

ファンアウト

メッセージを分割して集約 (結合) するのに役立ちます。

- ファンアウト・メディエーション・プリミティブとファンイン・メディエーション・プリミティブを組み合わせて使用すると、データを1つの出力メッセージに集約できます。
- 繰り返しモードでは、ファンアウト・メディエーション・プリミティブにより、繰り返し要素を含む単一の入力メッセージを繰り返すことができます。繰り返し要素が出現するたびに、メッセージが送信されます。
- 集約データを保持するには、共用コンテキストを使用します。

メッセージ・エレメント・セッター

メッセージの内容を設定するための単純な機能を提供します。

- メッセージ・エレメントを変更、追加、または削除することができます。
- メッセージのタイプは変更されません。

メッセージ・フィルター

メッセージ内容に基づいて、さまざまなパスにメッセージを送付します。

メッセージ・ロガー

リレーショナル・データベースにメッセージを記録します。メッセージはXMLとして保管されるので、データはXML対応アプリケーションで後処理できます。

- データベース・スキーマ (テーブル構造) はIBMによって定義されます。
- デフォルトでは、メッセージ・ロガー・メディエーション・プリミティブが共通データベースを使用します。ランタイムは、`jdbc/mediation/messageLog`にあるデータ・ソースを共通データベースにマップします。

サービス起動

メディエーション・フローが終了するまで待ってからコールアウト・メカニズムを使用するのではなく、メディエーション・フローの内部からサービスを呼び出します。

- サービスから障害が返された場合は、同じサービスを再試行するか、別のサービスを呼び出すことができます。
- サービス起動メディエーション・プリミティブは、単純なサービス呼び出しに対して単独で使用できるだけでなく、複雑なメディエーションの場合には他のメディエーション・プリミティブと組み合わせて使用することもできる強力なメディエーション・プリミティブです。

メッセージ型の設定

統合開発時に、型付けの弱いメッセージ・フィールドを型付けの強いメッセージ・フィールドであるかのように処理できます。フィールドに複数のデータ型を入力できる場合、そのフィールドは型付けの弱いフィールドです。フィールドの型および内部構造が既知の場合、そのフィールドは型付けの強いフィールドです。

- メッセージ型の設定メディエーション・プリミティブを使用すると、予想したデータ型とメッセージの内容が一致していることを実行時に確認できます。

停止 例外を生成せずに、フロー内の特定のパスを停止します。

XSL 変換

メッセージを変換します。

- Extensible Stylesheet Language (XSL) 変換を実行できます。
- XSLT 1.0 変換を使用してメッセージを変換します。この変換は、メッセージの XML 直列化を処理します。

関連概念

エンタープライズ・サービス・バスのメッセージング・インフラストラクチャー WebSphere Process Server には、エンタープライズ・サービス・バス機能があります。WebSphere Process Server は、サービス指向、メッセージ指向、およびイベント・ドリブンの各テクノロジーの統合をサポートすることにより、統合化されたエンタープライズ・サービス・バスにおいて標準ベースのメッセージング・インフラストラクチャーを実現します。

サービス・メッセージ・オブジェクト

Service message objects (SMO) は、サービス間で交換されるメッセージを処理および操作するための抽象化層を提供します。

SMO モデル

メディエーション・プリミティブは、メッセージを SMO として処理します。SMO はサービス・データ・オブジェクト (SDO) の拡張版で、SMO モデルは SDO DataObject を使用してメッセージを表現するためのパターンです。SMO には、以下のデータ・グループの表現が含まれます。

- メッセージに関連付けられたヘッダー情報。例えば、JMS API を使用してメッセージが伝達された場合は Java Message Service (JMS) ヘッダーになり、WebSphere MQ によってメッセージが生成される場合は MQ ヘッダーになります。
- メッセージ・ペイロード。メッセージ・ペイロードは、サービス・エンドポイント間で交換されるアプリケーション・データです。
- コンテキスト情報 (メッセージ・ペイロード以外のデータ)。

この情報はすべて SDO DataObject としてアクセスされます。また、SMO の全体構造を指定するスキーマ宣言があります。スキーマは WebSphere Integration Developer によって生成されます。

SMO の内容

すべての SMO の基本構造は同じです。この構造は、ServiceMessageObject と呼ばれるルート・データ・オブジェクトで構成されており、ルート・データ・オブジェクト内にヘッダー、本体、およびコンテキスト・データを表す他のデータ・オブジェクトが格納されています。ヘッダー、本体、およびコンテキストの正確な構造

は、統合開発時のメディエーション・フローの定義方法に応じて異なります。メディエーション・フローは、サービス間に介在する目的で、実行時に使用されます。

SMO ヘッダーには特定のエクスポート・バインディングまたはインポート・バインディングを発信元とする情報が記録されています (バインディングは、メッセージ・フォーマットおよびプロトコルの詳細を指定します)。メッセージは複数のソースから送信される可能性があるため、SMO は異なる種類のメッセージ・ヘッダーを保持できなければなりません。処理されるメッセージ・ヘッダーの種類は以下のとおりです。

- Web サービスのメッセージ・ヘッダー
- Service Component Architecture (SCA) メッセージ・ヘッダー
- Java Message Service (JMS) メッセージ・ヘッダー
- WebSphere MQ メッセージ・ヘッダー
- WebSphere Adapter メッセージ・ヘッダー

通常、アプリケーション・データを保持している SMO 本体の構造は、メディエーション・フローの構成時に指定した Web サービス記述言語 (WSDL) メッセージによって決定されます。

SMO コンテキスト・オブジェクトは、ユーザー定義またはシステム定義です。ユーザー定義のコンテキスト・オブジェクトを使用すると、メディエーション・プリミティブが後ほどフロー内で使用できるプロパティを保管できます。ユーザー定義のコンテキスト・オブジェクトの構造をビジネス・オブジェクトで定義して、そのビジネス・オブジェクトを要求フローの入力ノードで使用します。関連コンテキスト、一時コンテキスト、および共有コンテキストは、ユーザー定義のコンテキスト・オブジェクトです。

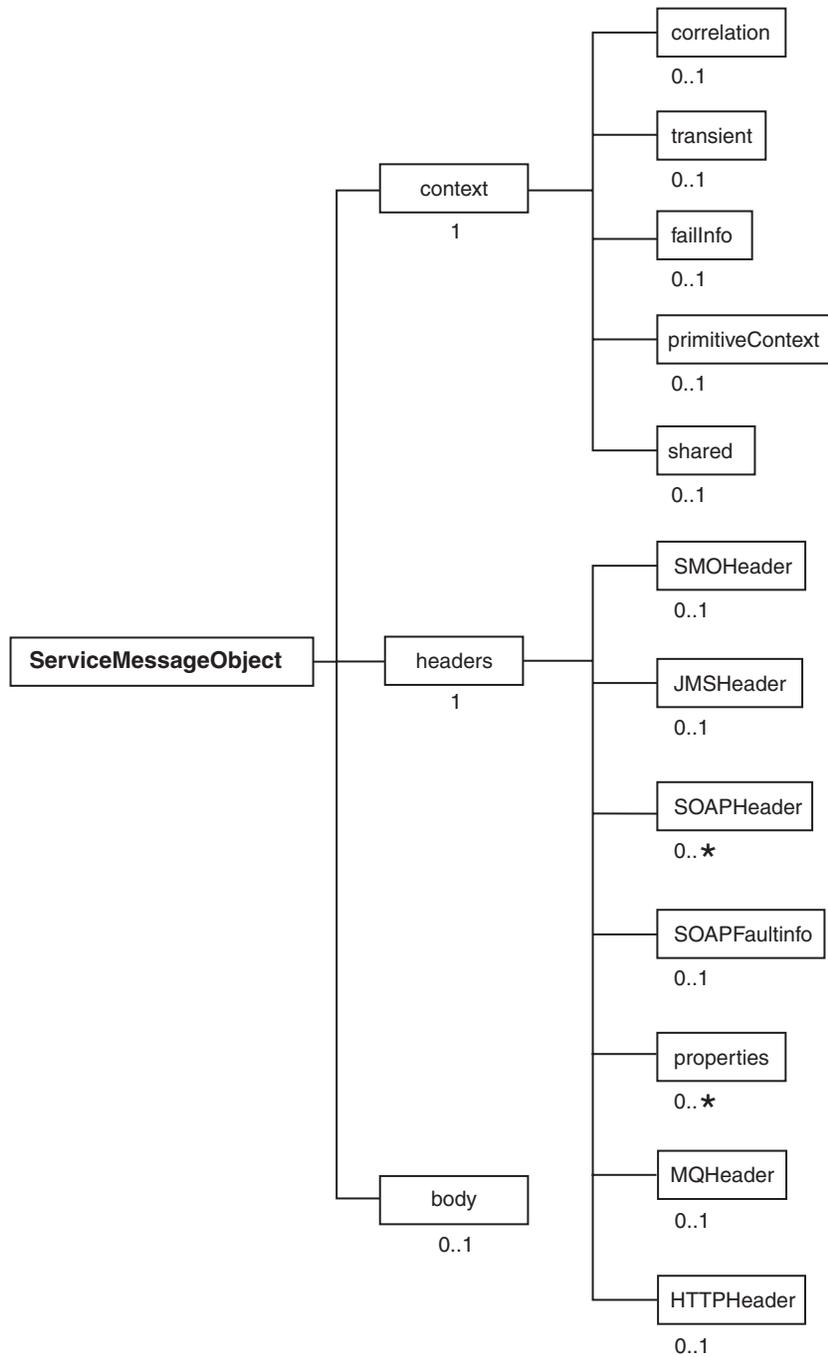


図9. SMO 構造の概要： ServiceMessageObject のコンテキスト、ヘッダー、および本文。

SMO は、メッセージ・ヘッダー、メッセージ・ペイロード、およびメッセージ・コンテキストへのアクセスまたはこれらの変更を行うためのインターフェースを提供します。

ランタイムによる SMO 使用の仕組み

ランタイムは、対話エンドポイント間で送信途中にあるメッセージを操作します。ランタイムは SMO オブジェクトを作成しますが、メディエーション・フローはこれを使用してメッセージを処理します。

メディエーション・フローを作成すると、WebSphere Integration Developer は、各端末 (入力、出力、または失敗) のメッセージ本体のタイプを指定し、必要に応じてコンテキスト情報のタイプも指定します。ランタイムはこの情報を使用してメッセージを変換し、指定されたタイプの SMO オブジェクトにします。

動的ルーティングを実現するには、WebSphere Service Registry and Repository (WSRR)、つまりデータベースを使用して、対話エンドポイントを検索します。WSRR 照会、つまりデータベース検索の結果は、SMO の特定の場所に保管できます。ランタイムは、そこから動的エンドポイントを取ります。

関連概念

エンタープライズ・サービス・バスのメッセージング・インフラストラクチャー WebSphere Process Server には、エンタープライズ・サービス・バス機能があります。WebSphere Process Server は、サービス指向、メッセージ指向、およびイベント・ドリブンの各テクノロジーの統合をサポートすることにより、統合化されたエンタープライズ・サービス・バスにおいて標準ベースのメッセージング・インフラストラクチャーを実現します。

第 9 章 WebSphere Process Server におけるアプリケーションの管理

IBM WebSphere Process Server の管理には、Service Component Architecture (SCA) モジュールがアプリケーションやリソースとしてデプロイされる先となる環境の準備、モニター、および変更と、アプリケーションやリソースでの作業が含まれます。

アプリケーションの管理について詳しくは、WebSphere Process Server の「管理」の PDF ファイルを参照してください。

WebSphere Process Server には、ランタイム環境の管理用として以下のインターフェースがあります。

- 管理コンソール

管理コンソールは、WebSphere Process Server で稼働する各種のアプリケーション、サービス、およびアプリケーションのリソースをモニター、更新、停止、および開始できる、ブラウザ・ベースのインターフェースです。また、管理コンソールは、リレーションシップの処理に使用したり、失敗した WebSphere Process Server イベントの検索および解決に使用したりできます。

管理コンソールには、WebSphere Application Server およびその他のユーザー定義製品の管理機能もあります。WebSphere Process Server 管理コンソールは、全体的に見れば Integrated Solutions Console フレームワークの一部であり、部分的に見れば WebSphere Application Server 管理コンソールの一部です。そのため、多数の管理タスク (例えば、セキュリティの設定、ログの表示、およびアプリケーションのインストール) は、WebSphere Process Server および WebSphere Application Server の両方で同一の内容になっています。

- コマンド行ツール

コマンド行ツールは、特定のタスクを実行するために、オペレーティング・システムのコマンド行プロンプトから実行する簡単なプログラムです。このツールを使用して、アプリケーション・サーバーの始動と停止、サーバー状況の確認、ノードの追加または除去、およびその他のタスクを実行できます。WebSphere Process Server コマンド行ツールには、serviceDeploy コマンドが含まれており、このコマンドは、WebSphere Integration Developer 環境からエクスポートされた .jar、.ear、.war および .rar ファイルを処理して、それらをプロダクション・サーバーにインストールするための準備をします。

- WebSphere 管理 (wsadmin) スクリプト・プログラム

wsadmin スクリプト・プログラムは、非グラフィカルなコマンド・インタープリター環境であり、スクリプト言語での管理オプションの実行、および実行用のスクリプト言語プログラムの実行依頼が可能です。管理コンソールと同じタスクがサポートされています。wsadmin ツールは、実稼働環境および不在操作向けです。

- 管理プログラム

Java Management Extensions (JMX) 仕様に基づく Java クラスおよびメソッドのセットを使用することにより、Service Component Architecture (SCA) およびビジネス・オブジェクトの管理がサポートされます。各プログラミング・インターフェースには、その目的の説明、インターフェースまたはクラスの使用法を示す例、および個々のメソッド説明への参照が含まれています。

- Business Process Choreographer Explorer

Business Process Choreographer Explorer は、ビジネス・プロセスやヒューマン・タスクを管理するための管理機能の基本セットを提供する、スタンドアロンの Web アプリケーションです。プロセス・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、およびそ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操作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、失敗アクティビティの修復と再開、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が可能です。

- Business Process Choreographer Observer

Business Process Choreographer Observer は、ビジネス・プロセスおよびヒューマン・タスクの実行時に発生するイベントについてのレポートを作成する Web アプリケーションです。これらのレポートを使用して、プロセスおよびアクティビティの実効性と信頼性を評価できます。

- ビジネス・ルール・マネージャー

ビジネス・ルール・マネージャーは、ビジネス・アナリストがビジネス・ルール値を表示および変更する際に使用する Web ベースのツールです。このツールは WebSphere Process Server のオプションであり、プロファイル作成時、またはサーバーの初期インストール後にインストールすることを選択できます。

関連概念



WebSphere Process Server の管理コンソール

管理コンソールは、アプリケーション、サービス、およびその他のリソースをセル、ノード、またはサーバーの有効範囲で管理するために使用されるブラウザ・ベースのインターフェースです。管理コンソールは、スタンドアロン・プロセス・サーバーから使用することもできますし、ネットワーク環境のセル内のすべてのサーバーを管理するデプロイメント・マネージャーから使用することもできます。

関連情報



WebSphere Process Server の管理

このセクションのトピックでは、環境にデプロイするアプリケーションやリソースなどの WebSphere Process Server ランタイム環境の管理方法について説明します。

第 10 章 WebSphere Process Server におけるアプリケーションの開発およびデプロイメント

WebSphere Process Server における統合アプリケーションの開発およびデプロイメント用オプションには、WebSphere Integration Developer の開発環境での作業、Service Component Architecture API を使用した作業、および WebSphere Process Server を使用したテスト環境または実稼働サーバー環境でのアプリケーションの使用可能化があります。

IBM WebSphere Integration Developer は、WebSphere Process Server の開発環境です。WebSphere Integration Developer での統合アプリケーションの開発について詳しくは、IBM WebSphere Business Process Management インフォメーション・センターを参照してください。

WebSphere Integration Developer 開発環境に加えて、開発者には Service Component Architecture API が公開されています。Service Component Architecture API について詳しくは、『モジュールの開発とデプロイ』を参照してください。

モジュール (WebSphere Process Server にデプロイされた場合は Service Component Architecture (SCA) モジュールとも呼ばれる) によって、ランタイム環境にデプロイされるエンタープライズ・アーカイブ (EAR) ファイルにパッケージされる成果物が決まります。

WebSphere Integration Developer 内では、アセンブリー・エディターを使用して、サービスをモジュールにグループ化したり、モジュールによって外部の利用者に公開されるサービスを指定したりできます。その後で、モジュールは接続され、完全な統合ソリューションが形成されます。統合ロジックはモジュール内にカプセル化し、変更されたモジュールのインターフェースが同じままである場合に、モジュール内のサービスへの変更がソリューション内の他のモジュールに影響しないようにします。

デプロイ とは、テスト環境または実稼働環境でアプリケーション (SCA モジュール) を使用可能にする動作のことです。どちらの環境でもデプロイは同じ意味ですが、それぞれの環境のデプロイメント・タスクにはいくつかの違いがあります。SCA モジュールへの変更は、実稼働環境にコミットする前にテスト・サーバーでテストするのが最も望ましいので、WebSphere Process Server を使用して SCA モジュールを実稼働環境にデプロイする前に、WebSphere Integration Developer を使用して SCA モジュールをテスト環境にデプロイします。

多数のアプリケーション・ファイルをデプロイする必要がある場合 (つまり、多数の SCA モジュールをインストールする場合) は、バッチ・ファイルを使用することもできます。バッチ・ファイルについて詳しくは、『実動サーバーへのモジュールのインストール』および『Apache Ant タスクを使用したアプリケーションのデプロイ』を参照してください。

関連タスク

実動サーバーへのモジュールのインストール

このトピックでは、テスト・サーバーからアプリケーションを取り出して実稼働環境にデプロイするステップについて説明します。

Apache Ant タスクを使用したアプリケーションのデプロイ

このトピックでは、Apache™ Ant タスクを使用して、WebSphere Process Server に対するアプリケーションのデプロイメントを自動化する方法について説明します。 Apache Ant タスクを使用すると、複数のアプリケーションのデプロイメントを定義し、サーバーへのそれらのアプリケーションの不在デプロイを実行できます。

関連情報

モジュールの開発とデプロイ

モジュールの開発とデプロイは、基本的な作業です。

第 11 章 WebSphere Process Server のセキュリティー

WebSphere Process Server は、IBM WebSphere Application Server のセキュリティーに基づいたランタイム・セキュリティーのインフラストラクチャーとメカニズムを備えています。

セキュリティーについては、『アプリケーションとその環境の保護』を参照してください。

第 12 章 WebSphere Process Server のモニター

問題判別の評価、パフォーマンスの調整、およびビジネス・プロセスの有効性の判定を行う目的で、WebSphere Process Server 内のイベントをモニターします。

WebSphere Process Server イベント・モニター機能には、パフォーマンス・モニターとサービス・コンポーネント・モニターが含まれます。

パフォーマンスのモニター: パフォーマンス測定は、サービス・コンポーネント・イベント・ポイントで使用可能で、Performance Monitoring Infrastructure (PMI) および Tivoli® Performance Viewer を介して処理されます。

イベントの呼び出し回数やそのイベントの開始から終了までに要した時間の長さなどの、特定のイベントに固有のパフォーマンス測定値をモニターすることができます。また、イベントをモニターして、ログ・ファイル内のイベントを表示するか、イベント・データベースに保管されているイベントを照会する方法で、後でイベントの内容を表示することもできます。どちらの場合も、アプリケーション・ロジックやシステム・パフォーマンスにおける問題を発見するために、モニターするイベント・ポイント (1 つまたは複数) を一時的に指定することができます。

サービス・コンポーネント・イベントのモニター: WebSphere Process Server モニターは、サービス・コンポーネント内のデータを特定のイベント・ポイントで収集できます。これらのイベントは、Common Base Event と呼ばれる規格にフォーマット設定されています。プロセス・サーバーを使用してこれらのイベントをロギング機能に公開することもできますし、Common Event Infrastructure サーバー・データベースのより用途の広いモニター機能を使用してこれらのイベントを保管および分析することもできます。

プロセス・サーバー上で稼働するアプリケーションの中には、アプリケーションがデプロイされた後に継続的にモニターされるイベント・ポイントを含むものがあります。ビジネス・アナリストがプロセス・サーバー上にデプロイしたアプリケーション内でモデル化およびインプリメントしたビジネス・プロセスの有効性を監視する場合に、これを設定することができます。これにより、IBM WebSphere Business Monitor などの製品を使用して、カスタマイズしたパネル、つまり「ダッシュボード」を作成し、ビジネス・プロセスの重要なメトリックを表示することができます。

関連情報

モニター

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は、IBM WebSphere Process Server の重要な機能です。

第 13 章 サンプル

サンプルは、WebSphere Process Server を使用して目標を達成する方法を習得するために役立ちます。

WebSphere Process Server サンプルは、製品と共にインストールできるサンプル・ギャラリーから入手できます。

Business Process Choreographer のサンプルは、次の Web ページから直接ダウンロードできます。 <http://publib.boulder.ibm.com/bpcsamp/index.html>

サンプル・ギャラリーのインストールおよびアクセス

統合アプリケーション成果物のサンプルはサンプル・ギャラリーに用意されています。このギャラリーは、本製品のインストール時にオプションでインストールできます。

このタスクについて

サンプル・ギャラリーには IBM WebSphere Integration Developer によって生成されて IBM WebSphere Process Server にデプロイされる成果物などの、単純な成果物のサンプルが含まれています。ビジネス・プロセスおよびヒューマン・タスクのサンプルは、次の Web ページから直接ダウンロードできます。

<http://publib.boulder.ibm.com/bpcsamp/index.html>

WebSphere Process Server サンプル・ギャラリーをインストールして表示させるには、以下のステップを実行します。

手順

1. WebSphere Process Server をインストールし、「フィーチャーの選択」パネルでサンプル・パッケージを選択して、製品のインストール作業の一環としてプロファイルを作成します。

注: WebSphere Process Server を WebSphere Application Server 上にインストールする場合は、サンプルを使用するために、ベースとなる WebSphere Application Server のサンプル・ギャラリーをインストールする必要があります。

サンプルは `install_root/samples` ディレクトリーにインストールされます。

2. サーバーを始動します。
3. ファースト・ステップ・コンソールで「**サンプル・ギャラリー**」を選択することによって、サンプル・ギャラリーを開始します。WebSphere Process Server のサンプルは、最初はサンプル・ギャラリーにインストール可能なサンプルとしてリストされます。「**インストール可能なサンプル (Installable Samples)**」を展開し、「**ビジネス・インテグレーション**」から、デプロイして実行したいサンプルを探すことができます。

WebSphere Process Server で実行されるアプリケーションは、ビジネス・オブジェクト、リレーションシップ定義、ビジネス・ルールなどの XML 成果物を持ちます。これらは、アプリケーションをインストールする前にデプロイする必要があります。WebSphere Process Server には、これらの成果物を作成およびデプロイするためのユーティリティー `serviceDeploy` が用意されています。各サンプル・アプリケーションの `install_root/samples/lib` にあるエンタープライズ・アーカイブ (EAR) ファイルには、これらの成果物が含まれています。sampleDeploy ユーティリティーは、各サンプルに必要な所定のパラメーターで `serviceDeploy` を起動します。sampleDeploy を実行すると、元の EAR ファイルと同じディレクトリに `sample_nameDeployed.ear` という別の EAR ファイルが作成されます。この新しい EAR ファイルには、元の EAR ファイルに存在した Web アーカイブ (WAR) ファイルのほかに、デプロイされた成果物を含む追加の Java アーカイブ (JAR) ファイルおよび WAR ファイルも含まれます。デプロイされた EAR ファイルは、WebSphere Process Server にエンタープライズ・アプリケーションとしてインストールできます。

4. WebSphere Process Server のインストール可能なサンプルがサンプル・ギャラリーに自動インストールされなかった場合は、手動でインストールし、デプロイする必要があります。
 - クラスター化されている分散 WebSphere Process Server デプロイメント環境にサンプルをインストールし、デプロイするには、以下のステップを完了してください。
 - a. 管理コンソールで「アプリケーション」を展開し、「新規アプリケーションのインストール」をクリックします。
 - b. 「参照」ボタンをクリックし、以下のディレクトリーにある SamplesGallery.ear ファイルを指定します。
 - Linux UNIX i5/OS **UNIX®、Linux®、および i5/OS® プラットフォームの場合:** `install_root/samples/lib/SamplesGallery`
 - Windows **Windows® プラットフォームの場合:**
`install_root\samples\lib\SamplesGallery`
 - c. EAR ファイルをインストールします。このとき、すべてのデフォルトを受け入れますが、「ターゲット・マッピング」パネルでは、サンプル・ギャラリーをインストールするサーバーまたはクラスターを指定できます。
 - d. SamplesGallery ディレクトリーにある WBISamplesGallery.ear ファイルについて、上記のステップを繰り返します。
 - e. インストールしたアプリケーションを開始します。
 - f. ブラウザーを開き、`http://host_name:host_port /WSsamples/index.jsp` にあるサンプル・ギャラリーにアクセスします。
 - g. サンプル・ギャラリーの説明に従って各サンプルをデプロイして実行します。ただし、`installwbi` コマンドではなく、管理コンソールの「新規アプリケーションのインストール」を使用します (そのコマンドがクラスターをサポートしないためです)。デプロイされた EAR ファイルは、各サンプルの以下のディレクトリーにあります。
 - Linux UNIX i5/OS **UNIX、Linux、および i5/OS プラットフォームの場合:** `install_root/samples/lib/sample_name`

– **Windows** Windows プラットフォームの場合:

`install_root%samples%lib%sample_name`

- クラスター化されていない分散 WebSphere Process Server デプロイメント環境にサンプルをインストールし、デプロイするには、以下のステップを実行してください。

- a. デプロイメント・マネージャー・ノードを持つマシン上で、次のコマンドを実行します。

– **Linux** **UNIX** **i5/OS** UNIX、Linux、および i5/OS プラットフォームの場合: `install_root/samples/bin/installwbi -node node_name -server server_name -samples SamplesGallery WBISamplesGallery`

– **Windows** Windows プラットフォームの場合:

`install_root%samples%bin%installwbi -node node_name -server server_name -samples SamplesGallery WBISamplesGallery`

注: 管理セキュリティが WebSphere Process Server プロファイル上で有効に設定されている場合は、`-samplepw` パラメーターを入力し、そのプロファイルの作成時に登録したパスワードも入力する必要があります。

- b. 管理コンソールで「アプリケーション」を展開し、「エンタープライズ・アプリケーション」をクリックし、SamplesGallery および WBISamplesGallery を開始します。
 - c. ブラウザーを開き、`http://host_name:host_port /WSsamples/index.jsp` にあるサンプル・ギャラリーにアクセスします。
 - d. サンプル・ギャラリーの説明に従って各サンプルをデプロイして実行します。このとき、必ず `installwbi` コマンドに `-node node_name -server server_name` パラメーターを指定してください。
5. サンプル・ギャラリーを開始し、「最新表示」をクリックします。ブラウザー・ウィンドウに表示される指示に従うと、インストール可能な任意のサンプルをデプロイすることができます。以前にデプロイしたサンプルは、ブラウザー・ウィンドウ内にインストール済みサンプルとしてリストに表示され、いずれかのオプションを選択することで実行できます。
 6. 画面の説明に従って、z/OS でデプロイされている各サンプルを実行します。

関連概念

ファースト・ステップ・コンソールのオプション

WebSphere Process Server をインストールしたら、ファースト・ステップ・コンソールを使用して、製品ツールの開始、製品資料へのアクセス、個別プロファイルに関連するサーバーおよび管理コンソールなどのエレメントへの指示を行います。汎用バージョンのコンソールと、インストール内のプロファイルごとのバージョンが使用可能です。各コンソールのオプションは、インストールした機能と、当該オペレーティング・システムで使用可能なエレメントに応じて動的に表示されます。オプションには、インストールの検証、サーバーまたはデプロイメント・マネージャーの始動と停止、管理コンソールへのアクセス、プロファイル

管理ツールの開始、サンプル・ギャラリーへのアクセス、製品資料へのアクセス、マイグレーション・ウィザードの開始などがあります。

Business Process Choreographer サンプル

Business Process Choreographer サンプルは、IBM WebSphere Integration Developer で開発され、IBM WebSphere Process Server にデプロイされた機能を説明します。さまざまな製品機能を使用して独自のアプリケーションを開発する上で役立ちます。

Business Process Choreographer のサンプルは、<http://publib.boulder.ibm.com/bpcsamp/index.html> から入手可能です。

第 14 章 標準への準拠

WebSphere Process Server はアクセシビリティの規格、情報処理の規格、ソフトウェア・ダウンロード・セキュリティの規格、およびインターネット・プロトコルの規格を含む、政府および業界の複数の規格に準拠しています。

アクセシビリティ

IBM は、年齢や能力を問わず、すべての人が便利に使用できる製品の提供に努めています。

この製品では、標準の Windows ナビゲーション・キーを使用します。

WebSphere Process Server のアクセシビリティ機能

アクセシビリティ機能は、運動障害または視覚障害など身体に障害を持つユーザーが情報技術プロダクトを快適に使用できるようにサポートします。

アクセシビリティ機能

以下のリストに、WebSphere Process Server の主なアクセシビリティ機能を示します。アクセシビリティ機能には、以下の機能があります。

- キーボード専用の操作
- スクリーン・リーダーで一般に使用されるインターフェース

WebSphere Process Server を使用する場合は、アクセシビリティをサポートするオペレーティング・システムの機能を使用できます。

ヒント: WebSphere Process Server インフォメーション・センターは、スクリーン・リーダー・ソフトウェア (IBM ホームページ・リーダーを含む) に対応しています。マウスの代わりにキーボードを使用して、すべての機能を実行できます。

キーボード・ナビゲーション

この製品は、Web ブラウザーの標準ナビゲーション・キーと、Installshield Multiplatform の標準ナビゲーション・キーを使用します。

(サポートされる Web ブラウザーについて詳しくは、<http://www.ibm.com/software/integration/wps/sysreqs/>にある WebSphere Process Server System Requirements を参照してください。)

インターフェース情報

- インストール

WebSphere Process Server は、グラフィカル形式とサイレント形式のいずれかでインストールできます。アクセス支援が必要な方には、サイレント・インストール・プログラムをお勧めします。

詳しくは、『製品へのサイレント・インストール製品へのサイレント・インストール』を参照してください。

注: WebSphere Process Server インストーラー・プログラムは、Installshield Multiplatform コンソール・モードをサポートしません。

- 管理

管理コンソールは、製品との対話するための基本インターフェースです。このコンソールは、標準の Web ブラウザーに表示されます。管理者は、Microsoft® Internet Explorer などの利用できる Web ブラウザーを使用して、以下のことを行うことができます。

- スクリーン・リーダー・ソフトウェアとデジタル・スピーチ・シンセサイザーの使用による、画面の表示内容の聴覚による認識。
- IBM ViaVoice® などの音声認識ソフトウェアの使用による、データ入力およびユーザー・インターフェースのナビゲート。
- マウスの代わりにキーボードを使用することによる機能の操作。

提供されているグラフィカル・インターフェースの代わりに、標準テキスト・エディターおよびスクリプト・インターフェースまたはコマンド行インターフェースを使用して、製品の機能を構成および管理することができます。

該当する場合は、特定の製品機能の資料には、その機能のアクセシビリティについての詳細が記載されています。

ベンダー・ソフトウェア

この製品には、IBM のご使用条件が適用されない、特定のサード・パーティー・ソフトウェアが含まれています。IBM は、これらの製品についての米国リハビリテーション法第 508 条への準拠状況について何ら表明するものではありません。各製品の米国リハビリテーション法第 508 条への準拠状況については、当該製品のベンダーにお問い合わせください。米国リハビリテーション法第 508 条の Voluntary Product Accessibility Template (VPAT) を入手する場合は、次の URL にある Product accessibility information の Web ページから請求してください。
www.ibm.com/able/product_accessibility

アクセシビリティの関連情報

IBM のアクセシビリティに対する取り組みについて詳しくは、IBM Accessibility Center を参照してください。

連邦情報処理標準

連邦情報処理標準 (FIPS) は、米国連邦情報・技術局 (NIST) が、連邦政府コンピューター・システムのために発行した標準およびガイドラインです。

WebSphere Process Server は、連邦情報処理標準に準拠しているすべての暗号機能を、IBM WebSphere Application Server に依存しています。

FIPS は、セキュリティおよびインターオペラビリティなど、標準に関する連邦政府の切実な要求がある一方で、実施可能な業界標準または解決方法が存在しない

場合のために開発されました。政府機関および金融機関はこれらの標準を使用し、製品が明記されたセキュリティー要件に適合していることを確認します。これらの標準について詳しくは、米国連邦情報・技術局 (<http://www.nist.gov/>) を参照してください。

WebSphere Application Server では、FIPS 140-2 認定を受けた Java Secure Socket Extension (JSSE) および Java Cryptography Extension (JCE) などの暗号モジュールが統合されています。WebSphere Application Server の資料では、FIPS 認証を受けた IBM JSSE モジュールおよび JCE モジュールは、それぞれ IBMJSSEFIPS および IBMJCEFIPS と呼ばれています。

詳しくは、WebSphere Application Server インフォメーション・センターの『連邦情報処理標準 (FIPS) Java セキュア・ソケット拡張機能ファイルの構成』を参照してください。FIPS を使用可能にすると、サーバーの一部のコンポーネント (暗号スイート、暗号プロバイダー、ロード・バランサー、キャッシング・プロキシ、高可用性マネージャー、データ複製サービスなど) が影響を受けます。

インターネット・プロトコル バージョン 6

WebSphere Process Server は、インターネット・プロトコル バージョン 6 のすべての互換性について WebSphere Application Server に依存しています。

IBM WebSphere Application Server Version 6.1 とその JavaMail コンポーネントは、インターネット・プロトコル バージョン 6 (IPv6) をサポートしています。

WebSphere Application Server でのこの互換性について詳しくは、WebSphere Application Server Network Deployment 資料の『IPv6 のサポート』を参照してください。

IPv6 について詳しくは、www.ipv6.org を参照してください。

第 15 章 グローバリゼーション

グローバル化された製品は、言語や文化の障壁なく使用することができ、特定ロケールに関して使用可能に設定できます。

WebSphere Process Server は、すべてのロケールの基本的な使用可能化サポートを提供します。この製品は、以下の各国語に翻訳されています。

- ブラジル・ポルトガル語
- チェコ語
- フランス語
- ドイツ語
- ハンガリー語
- イタリア語
- 日本語
- 韓国語
- ポーランド語
- ロシア語
- 中国語 (簡体字) (GB18030 に準拠)
- スペイン語
- 中国語 (繁体字)

さらに、以下の各国語については、WebSphere Process Server は部分的に翻訳されています。

- アラビア語
- ヘブライ語

アプリケーションのグローバル化および WebSphere Application Server を介して使用可能な国際化対応サービスについては詳しくは、WebSphere Application Server Network Deployment 資料にある『WebSphere 拡張機能』を参照してください。

双方向言語サポート

WebSphere Process Server、双方向使用可能化により双方向言語をサポートします。双方向使用可能化とは、WebSphere Process Server と共にバンドルされるコンポーネント (例えば、Common Base Event Browser またはビジネス・ルール・マネージャーなどの Web ベース・ツール)、またはそれによってサポートされるコンポーネント (例えば、サービス・コンポーネント) の内部で双方向スクリプト・データを正確に表示および処理するためのメカニズムです。

WebSphere Process Server では、Windows の標準双方向言語形式である、「左から右」のロジックのすべての双方向言語データを処理します。内部コンポーネントに受け渡されたデータを処理し、データを保存し、データをそのフォーマットで出力します。WebSphere Adapters およびその他のエンタープライズ情報システム (EIS) では、データをこの形式に変換してから、WebSphere Process Server によって処理

されるようデータを送信する必要があります。 WebSphere Process Server によって出力されるデータも「左から右」のロジックの形式であるため、受信側アプリケーションでは、外部 EIS で必要な正しい双方向形式にそのデータを変換する必要があります。

次の表では、Windows の標準双方向形式に一致しなければならない属性および設定を示しています。

表 1. 双方向言語形式のSTRING値

| 文字位置 | 目的 | 指定可能な値 | デフォルト値 | 意味 |
|------|----------|--------|--------|---------------|
| 1 | 順序スキーマ | I | I | 暗黙 |
| | | V | | 表示 |
| 2 | 方向 | L | L | 左から右 |
| | | R | | 右から左 |
| | | C | | コンテキスト (左から右) |
| | | D | | コンテキスト (右から左) |
| 3 | 対称スワッピング | Y | Y | 対称スワッピングがオン |
| | | N | | 対称スワッピングがオフ |
| 4 | 形状指定 | S | N | テキストの形状を指定する |
| | | N | | テキストの形状を指定しない |
| | | I | | 語頭形の指定 |
| | | M | | 語中形の指定 |
| | | F | | 語尾形の指定 |
| | | B | | 独立形の指定 |
| 5 | 数値 | H | N | ヒンディ語 (各国語) |
| | | C | | コンテキスト |
| | | N | | 公称 |

双方向言語サポートを実施しない外部コンポーネントからのデータ (双方向データを処理するために使用可能にされていない Web サービスまたはコネクタ) の場合、双方向 API の例 (IBM Java Development Kit (JDK) に基づく) を使用して、データを外部ソースからサポートされる双方向言語形式に変換する API、および WebSphere Process Server から外部 EIS に送信されたデータを特定の EIS によって使用される双方向形式に変換する API を作成できます。

STRING・オブジェクトを変換する API を作成するには、『ある双方向言語形式から別の双方向言語形式へのデータ・オブジェクトの変換』を参照してください。

データ・オブジェクトを変換する API を作成するには、『ある双方向言語形式から別の双方向言語形式へのデータ・オブジェクトの変換』を参照してください。

注: ユーザー・インターフェース (ブラウザー) のロケール設定で、双方向言語の表示および編集形式を定義します。

双方向言語について詳しくは、www.ibm.com/developerworks/websphere/library/techarticles/bidi/bidigen.html の IBM developerWorks にある技術記事を参照してください。

ストリング・オブジェクトの双方向言語形式の変換

外部のエンタープライズ情報システム (EIS) との間でやり取りするデータについて、ストリング・データをサポート対象の双方向言語形式に変換したり、WebSphere Process Server から外部の EIS に送信するデータをその特定の EIS で使用されている双方向言語形式に変換したりする API を作成できます。

始める前に

双方向言語サポートについて詳しくは、『グローバリゼーション』を参照してください。『グローバリゼーション』に記載されている表を参照して、ストリング・データの形式変換を実行するとき使用する入力ストリングまたは出力ストリングの正しい値を確認できます。

ストリング・オブジェクトの双方向言語形式を変換するための API を作成するには、以下の手順を実行します。

手順

1. 双方向エンジンのインプリメンテーションを含むすべての双方向クラスを組み込みます。例えば以下のように構成できます。

```
import com.ibm.bidiTools.bdlayout.*;
```

2. 変換するデータ・オブジェクトを格納するストリングと、入出力形式値を定義します。

入力形式は、ストリング・オブジェクトが現在格納されている双方向形式です。出力形式は、ストリング・オブジェクトを格納する双方向形式です。例えば以下のように構成できます。

```
String strIn = new String("Hello world");  
String formatIn = "ILYNN";  
String formatOut = "VLYNN";
```

3. `BidiStringTransformation` 関数を呼び出します。例えば以下のように構成できます。

```
String strOut = BidiStringTransformation(strIn, formatIn, formatOut);
```

```
String BidiStringTransformation(String strIn, String formatIn, String formatOut) {
```

- a. 入力ストリングが `NULL` かどうかをテストします。例えば以下のように構成できます。

```
    if (strIn == null) return null;
```

- b. 変換を実行します。例えば以下のように構成できます。

```
    BidiFlagSet flagsIn;  
    BidiFlagSet flagsOut;  
    formatIn = formatIn.toUpperCase();  
    formatOut = formatOut.toUpperCase();
```

```
    if (formatIn != null)  
        flagsIn = new BidiFlagSet(formatIn.toCharArray());  
    else  
        flagsIn = new BidiFlagSet();
```

```

    if (formatOut != null)
        flagsOut = new BidiFlagSet(formatOut.toCharArray());
    else
        flagsOut = new BidiFlagSet();

    if (flagsIn.equals(flagsOut)) return strIn;
    String strOut = BiDiStringTransformation(strIn, flagsIn, flagsOut);
    return strOut;
}

```

データ・オブジェクトの双方向言語形式の変換

外部のエンタープライズ情報システム (EIS) との間でやり取りするデータについて、サービス・データ・オブジェクトをサポート対象の双方向言語形式に変換したり、WebSphere Process Server から外部の EIS に送信するデータをその特定 EIS で使用されている双方向言語形式に変換したりする API を作成できます。

始める前に

双方向言語サポートについて詳しくは、『グローバリゼーション』を参照してください。『グローバリゼーション』に記載されている表を参照して、DataObject タイプのデータの形式変換を実行するとき使用する入力ストリングまたは出力ストリングの正しい値を確認できます。

データ・オブジェクトの双方向言語形式を変換するための API を作成するには、以下の手順を実行します。

手順

1. 双方向エンジンのインプリメンテーションを含むすべての双方向クラスを組み込みます。例えば以下のように構成できます。

```
import com.ibm.bidiTools.bdlayout.*;
```

2. DataObject 型オブジェクトを操作する必要のあるすべてのクラスを組み込みます。例えば以下のように構成できます。

```
import commonj.sdo.DataObject;
import commonj.sdo.Type;
import commonj.sdo.Property;
```

3. DataObject 型オブジェクトに含まれる各種のストリングを格納するストリング変数を定義します。このようにして、DataObject を再帰的に処理しながら、String タイプの属性をフィルターにかけます。例えば以下のように構成できます。

```
String STRING_STR_TYPE = "String";
String NORM_STRING_STR_TYPE = "normalizedString";
String TOKEN_STR_TYPE = "token";
String LANG_STR_TYPE = "language";
String NAME_STR_TYPE = "Name";
String NMTOKEN_STR_TYPE = "NMTOKEN";
String NCNAME_STR_TYPE = "NCName";
String ID_STR_TYPE = "ID";
String IDREF_STR_TYPE = "IDREF";
String IDREFS_STR_TYPE = "IDREFS";
String ENTITY_STR_TYPE = "ENTITY";
String ENTITIES_STR_TYPE = "ENTITIES";
```

4. プロパティの型が String であるかどうかを検査する関数を定義します。例えば以下のように構成できます。

```

private static boolean isStringFamilyType (Property property) {
    boolean rc = false;
    if ((property.getType().getName().equalsIgnoreCase(String_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase(NORM_STRING_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase(TOKEN_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase(LANG_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase(NAME_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase(NMTOKEN_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase(NCNAME_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase(ID_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase>IDREF_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase>IDREFS_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase>ENTITY_STR_TYPE)) ||
        (property.getType().getName().equalsIgnoreCase>ENTITIES_STR_TYPE)))
        rc = true;
    return rc;
}

```

5. `DataObject` 全体に双方向変換を適用する再帰的関数を定義します。

注: コード・ロジックには、以下の前提を含めます。

- `String` タイプのプロパティーのみに双方向変換を適用します。
- `DataObject` の `String` タイプのプロパティーを 1 つの双方向形式で格納します。

例えば以下のように構成できます。

```

DataObject BiDiDataObjTransformationB0(DataObject boIn, String formatIn, String formatOut){
    Type type;
    Property property;

    if (boIn == null) return null;

    type = boIn.getType();
    List propertyList = type.getProperties();
    for (int propertyNumber = 0; propertyNumber < propertyList.size(); propertyNumber++){
        property = (Property) propertyList.get(propertyNumber);
        String propertyName = property.getName();

```

- a. `string` でないプロパティーはすべてスキップします。 例えば以下のように構成できます。

```

        if (!isStringFamilyType(property))
            continue;

        if (property.isContainment()) {
            if (property.isMany()) {
                List childsList = boIn.getList(property);

```

- b. 子オブジェクトを処理する変換を再帰的に呼び出します。 例えば以下のように構成できます。

```

                for (int childNumber = 0; childNumber < childsList.size();
                    childNumber++){
                    BiDiDataObjTransformationB0(connectionContext,
                    ((DataObject)childsList.get(childNumber)),formatIn, formatOut);
                }
            } else {

```

- c. 組み込まれているビジネス・オブジェクトの子オブジェクトを処理する変換を再帰的に呼び出します。 例えば以下のように構成できます。

```

                    BiDiDataObjTransformationB0(connectionContext,
                    ((DataObject)boIn.get(property)),formatIn, formatOut);
                }
            } else {

```

- d. 単純な `string` 属性を変換します。 例えば以下のように構成できます。

```
        String str = BiDiStringTransformation(
            (boIn.getString(propertyName),formatIn, formatOut);
        boIn.setString(propertyName, str);
    }
}
return boIn;
}
```

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

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

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

〒106-8711
東京都港区六本木 3-2-12
日本アイ・ビー・エム株式会社
法務・知的財産
知的財産権ライセンス渉外

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

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

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

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

IBM Corporation 577 Airport Blvd., Suite 800
Burlingame, CA 94010
U.S.A.

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

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

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

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

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

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

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生的創作物にも、次のように、著作権表示を入れていただく必要があります。(c) (お客様の会社名) (西暦年)。このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。(C) Copyright IBM Corp. _年を入れる_。 All rights reserved.

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

プログラミング・インターフェース情報

プログラミング・インターフェース情報は、プログラムを使用してアプリケーション・ソフトウェアを作成する際に役立ちます。

一般使用プログラミング・インターフェースにより、お客様はこのプログラム・ツール・サービスを含むアプリケーション・ソフトウェアを書くことができます。

ただし、この情報には、診断、修正、および調整情報が含まれている場合があります。診断、修正、調整情報は、お客様のアプリケーション・ソフトウェアのデバッグ支援のために提供されています。

警告: 診断、修正、調整情報は、変更される場合がありますので、プログラミング・インターフェースとしては使用しないでください。

商標

IBM、IBM ロゴ、ibm.com は、International Business Machines Corporation の米国およびその他の国における商標または登録商標です。これらおよび他の IBM 商標に、この情報の最初に現れる個所で商標表示 (® または ™) が付されている場合、これらの表示は、この情報が公開された時点で、米国において、IBM が所有する登録商標またはコモン・ロー上の商標であることを示しています。このような商標は、その他の国においても登録商標またはコモン・ロー上の商標である可能性があります。現時点での IBM の商標リストについては、www.ibm.com/legal/copytrade.shtml の「Copyright and trademark information」をご覧ください。

Adobe は、Adobe Systems Incorporated の米国およびその他の国における登録商標または商標です。

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標です。

Microsoft および Windows は、Microsoft Corporation の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

この製品には、Eclipse Project (<http://www.eclipse.org>) により開発されたソフトウェアが含まれています。





Printed in Japan