

IBM WebSphere Adapters



WebSphere Adapter for Flat Files ユーザーズ・ガイド

バージョン 6.0

お願い

本書および本書で紹介する製品をご使用になる前に、67ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM WebSphere Adapter for Flat Files (製品番号 5724L78) バージョン 6、リリース 0、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。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

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： IBM WebSphere Adapters
WebSphere Adapter for Flat Files User Guide
Version 6.0

発 行： 日本アイ・ピー・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 2006.3

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、
平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 2005, 2006. All rights reserved.

© Copyright IBM Japan 2006

目次

WebSphere Adapter for Flat Files バージョン 6.0 ユーザーズ・ガイド	1
製品の概要	1
IBM WebSphere Adapter	1
対象読者	3
タスク・ロードマップ: IBM WebSphere Adapter for Flat Files	3
アダプター間の違い	4
エンタープライズ・サービス・ディスカバリー	5
アダプターの動作方法	5
ロケールとグローバリゼーションのサポート	8
ビジネス・オブジェクトの概要	10
ビジネス・オブジェクトの命名規則	10
ビジネス・オブジェクトの構造	10
属性プロパティ	11
サポートされる操作	12
アダプターのインストール	13
アダプター環境	13
アダプター固有のインストール情報	14
インストール済みファイルの構造	14
アダプターの配置 (Inbound 操作の場合)	15
アダプター・プロジェクトの作成 (Inbound 操作の場合)	16
サービス成果物の生成 (Inbound 操作の場合)	16
参照バインディングの生成 (Inbound 操作の場合)	21
EAR ファイルのエクスポート (Inbound 操作の場合)	24
アプリケーションのインストール (Inbound 操作の場合)	25
アプリケーションの実行 (Inbound 操作の場合)	26
アダプターの配置 (Outbound 操作の場合)	27
アダプター・プロジェクトの作成 (Outbound 操作の場合)	28
サービスの構成 (Outbound 操作の場合)	29
参照バインディングの生成 (Outbound 操作の場合)	33
EAR ファイルのエクスポート (Outbound 操作の場合)	35
アプリケーションのインストール (Outbound 操作の場合)	36
アプリケーションの実行 (Outbound 操作の場合)	38
アダプターの構成	39
プロパティの構成	39
WebSphere Adapter for Flat Files の構成プロパティ	41
トラブルシューティングおよびサポート	48
IBM ソフトウェア・サポートへの連絡	48
ロギング可能化	49
トレースの使用可能化	51
Common Event Infrastructure (CEI) の使用可能化	53
サンプル・アプリケーションの使用	54
アプリケーション・パッケージの構造	55
ビジネス・オブジェクトの構造	56
シナリオ 1 の Inbound 操作作用の配置および構成	56
シナリオ 1 の Outbound 操作作用の配置および構成	57
シナリオ 2 の Inbound 操作作用の配置および構成	58
シナリオ 2 の Outbound 操作作用の配置および構成	62
サンプル・アプリケーションの実行 (Inbound 操作の場合)	65
サンプル・アプリケーションの実行 (Outbound 操作の場合)	66
特記事項	67

プログラミング・インターフェース情報	69
商標	69

WebSphere Adapter for Flat Files バージョン 6.0 ユーザーズ・ガイド

IBM® WebSphere Adapter for Flat Files は、エンタープライズ情報システムのファイル・システムと J2EE ベースのプログラミング・モデル間のビジネス・オブジェクトの交換を円滑化します。

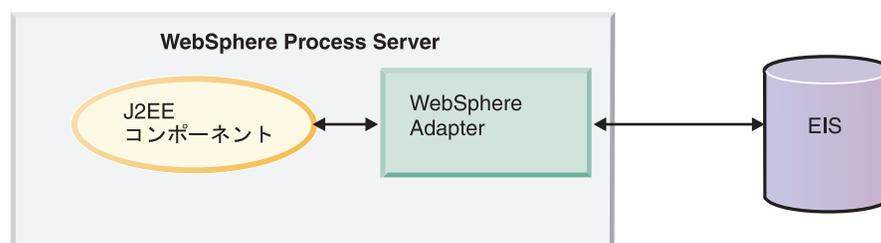
製品の概要

このトピックでは、WebSphere Adapter for Flat Files のリリース、製品の特長、およびシステム要件について概説します。

IBM WebSphere Adapter

IBM WebSphere Adapter には、Java 2 Enterprise Edition (J2EE) Connector Architecture (JCA) バージョン 1.5 が実装されています。WebSphere Adapter は、リソース・アダプターまたは JCA アダプターとも呼ばれ、WebSphere Process Server によってサポートされるエンタープライズ情報システム (EIS) と J2EE コンポーネント間の双方向の管理接続を可能にします。

WebSphere Adapter



IBM^(R) WebSphere^(R) Adapter のポートフォリオは、Java 2 Platform Enterprise Edition (J2EE) 標準に基いた新世代のアダプター群です。JCA とは、J2EE アプリケーションとエンタープライズ情報システムを統合するための標準アーキテクチャーのことです。これらの各システムには、呼び出す機能の識別、入力データの指定、出力データの処理用にそれぞれ固有の API があります。JCA の目的は、このような関数をコーディングするための独立した API の提供、データ共有の促進、および J2EE アプリケーションと既存の EIS やその他の EIS との統合です。JCA 標準では、アプリケーション・サーバー内での EIS と J2EE コンポーネント間の対話を制御する一連のコントラクトを定義することにより、上記の接続を実現します。

JCA 標準に完全に準拠している WebSphere Adapter は、WebSphere Process Server 上で稼働するように開発されました。WebSphere Adapter の機能は、以下のとおりです。

- WebSphere Process Server と統合する。
- WebSphere Process Server 上で稼働するアプリケーションと EIS とを接続する。
- アプリケーションと EIS の間のデータ交換を可能にする。

各 WebSphere Adapter は、次の構成要素から成ります。

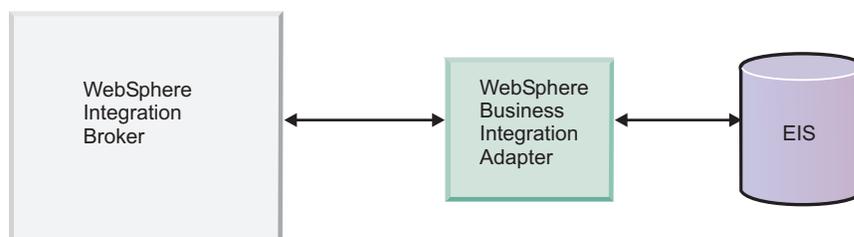
- WebSphere Process Server をサポートする (J2EE) Connector Architecture (JCA) バージョン 1.5 の実装環境。
- 標準のエンタープライズ・アプリケーション・アーカイブ (EAR) ファイルにコンパイルされているビジネス・オブジェクトやその他のサービス・コンポーネント・アーキテクチャー (SCA) 成果物を生成するための Enterprise Metadata Discovery コンポーネント (このコンポーネントをエンタープライズ・サービス・ディスカバリー・ウィザードで使用することで、EIS をイントロスペクトします)。

WebSphere Adapter は、データ・オブジェクトを表現するためにサービス・データ・オブジェクト (SDO) を使用します。

WebSphere Adapter と WebSphere Business Integration Adapter

WebSphere Adapter とは異なり、WebSphere Business Integration Adapter は JCA に対応していません。

WebSphere Business Integration Adapter



図に示すように、WebSphere Business Integration Adapter は分散型です。これらは、アプリケーション・サーバーの外側に存在します。このサーバー、つまり統合ブローカーは、このタイプのアダプターと Java Messaging Service (JMS) トランスポート層を介して通信します。

WebSphere Adapter と WebSphere Business Integration Adapter との違いには、このほかに以下のものがあります。

- **接続の管理:** WebSphere Adapter は、標準の JCA コントラクトに依存して、停止や開始などのライフ・サイクル・タスクを管理します。一方、WebSphere Business Integration Adapter は、WebSphere Adapter Framework に依存して接続性を管理します。
- **イベント通知:** WebSphere Adapter の Inbound イベント通知として知られています。
- **要求処理:** WebSphere Adapter での Outbound サポートとして知られています。
- **オブジェクト定義:** WebSphere Adapter では、Enterprise Metadata Discovery コンポーネントを使用して EIS を厳密に調べ、ビジネス・オブジェクトやその他の有益な成果物を作成します。この Enterprise Metadata Discovery コンポーネントは、WebSphere Adapter の一部になっています。WebSphere Business Integration Adapter は、個々の Object Discovery Agent (ODA) を使用して EIS を厳密に調べ、ビジネス・オブジェクト定義スキーマを生成します。

対象読者

このトピックでは、WebSphere Adapter 製品のユーザーを定義し、ユーザーに必要なスキルを示します。

アダプター・ユーザーズ・ガイドの対象読者には、各アプリケーション・コンポーネントを完全なソリューションとして組み立て、このソリューションをテストおよび配置用に準備する、データおよびアプリケーションの統合担当者が含まれています。これらのユーザーには、以下の一般的なスキルが必要です。

- ビジネス・ソリューションおよびビジネス環境をよく理解していること
- アプリケーションおよびソリューションのコンポーネントの実行時の効果的なコラボレーションを可能にするための、これらのコンポーネントに関する知識
- データベース、データ・アクセスの問題、トランザクション・モデル、および異種のリレーショナル・データベース、キュー、および Web サービス間の接続についての詳細な知識
- 統合ツールについての熟知

アプリケーションの統合担当者は、詳細なテスト作業についても責任があり、その他に以下のスキルも必要です。

- テストおよび配置に必要なスクリプト、ツール、およびテンプレートの作成
- 統合ワークスペースの作成とシステムおよびサブシステムの統合
- Enterprise Java Beans (EJBs)、ワークフロー、Web ページなどのエンティティー間の相互依存性の解決
- アプリケーションまたはソリューションの検証

データの統合担当者は、アプリケーション開発者が一連のデータ・ソースにアクセスできるようにする責任もあります。必要なスキルは以下のとおりです。

- 統合機能またはポイント・ツー・ポイント・ゲートウェイのインストールおよび構成
- データベース・アクセス・ロジックを効果的に使用するためのプロシージャの作成
- 外部データ・アクセス・ツール用のデータ・モデルの作成
- セキュリティ対策の実施

タスク・ロードマップ: IBM WebSphere Adapter for Flat Files

IBM WebSphere Adapter for Flat Files により、フラット・ファイルと J2EE ベースのプログラミング・モデルの間でビジネス・オブジェクトを容易に交換できます。

タスク・ロードマップにより、ユーザーは WebSphere Adapter for Flat Files のインストールから使用までのすべての全体像を把握することができます。

タスク	説明
ビジネス・オブジェクトについて	このトピックでは、WebSphere Adapter for Flat Files のビジネス・オブジェクトについて説明します。
アダプターのインストール	このトピックでは、WebSphere Adapter for Flat Files のインストール方法を説明します。

タスク	説明
アダプターの配置 (Inbound 操作の場合)	このトピックでは、Inbound 操作の場合の WebSphere Adapter for Flat Files の配置方法を説明します。
アダプターの配置 (Outbound 操作の場合)	このトピックでは、Outbound 操作の場合の WebSphere Adapter for Flat Files の配置方法を説明します。
プロパティの構成	このトピックでは、WebSphere Adapter for Flat Files のプロパティの構成方法を説明します。
アダプターのトラブルシューティング	このトピックでは、WebSphere Adapter for Flat Files のトラブルシューティング方法を説明します。
サンプル・アプリケーションの使用	このトピックでは、WebSphere Adapter for Flat Files のサンプル・アプリケーションの使用方法を説明します。

アダプター間の違い

このトピックでは、WebSphere® Business Integration Adapter for JText の前バージョンと、新しい WebSphere Adapter for Flat Files との違いについて説明します。

WebSphere Business Integration Adapter for JText の前バージョンと新しい WebSphere Business Integration Adapter for Flat Files との間には、大きな違いが 2 つあります。アーキテクチャー上の違いと機能上の違いを以下に示します。

• アーキテクチャー上の違い

- **プロトコルの分割:** WebSphere Business Integration Adapter for JText を使用した場合、アダプターはローカル・ファイル操作と FTP ファイル操作の両方の機能を兼ね備えていました。 WebSphere Adapter for Flat Files を使用した場合、アダプターが処理するのは、ローカル・ファイル操作のみになります。この機能上の分離により、アダプターは保守が容易になり、プロトコル固有の操作への特化傾向が強くなりました。
- **Inbound イベントの認識:** WebSphere Business Integration Adapter for JText の場合、イベント・ファイル内の個々のデータ・レコードはイベントとみなされており、アダプターはイベント・ファイルを解析してレコード内容を抽出していました。 WebSphere Adapter for Flat Files の場合は、ファイル全体がイベントとみなされ、アダプターはイベント・ファイルを解析しません。このアーキテクチャーが採用されたのは、ファイル処理とデータ形式変更の両方に関係するいくつかの異なるタスクを分離するためです。ファイル処理では、ファイル全体を処理しますが、ここでは、Inbound 処理のためのイベント・ファイルの到着の検出、ファイル全体の内容の読み取り、および Outbound 処理のためのファイル内容の書き込みが必要になります。一方、データ形式変更には、ファイルの解析と、ファイルからのデータ・レコードの抽出が必要です。このプロトコル処理とデータ形式変更の分割により、個々のコンポーネントは再利用しやすくなり、保守が容易になります。

• 機能上の違い

- **Outbound 処理機能の向上:** WebSphere Adapter for Flat Files は、Outbound 処理について、WebSphere Business Integration Adapter for JText より多くの操作をサポートします。WebSphere Business Integration Adapter for JText の要求操作は、適切なメタオブジェクト構成を介して Create、Append、Overwrite の各操作のみをサポートしていましたが、一方、WebSphere Adapter for Flat Files は、Create、Append、Overwrite、Retrieve、Delete、Exists、List の各操作をサポートします。
- **Inbound 処理で利用できるファイル分割機能:** WebSphere Adapter for Flat Files を使用した場合は、イベント・ファイルを指定のチャンク・サイズでエンドポイントまで配信できます。

エンタープライズ・サービス・ディスカバリー

エンタープライズ・サービス・ディスカバリー・ウィザードを使用すると、エンタープライズ情報システム (EIS) エンティティーやデータベース・エンティティー用のビジネス・オブジェクトを生成できます。

エンタープライズ・サービス・ディスカバリー・ウィザードを使用すると、ビジネス・オブジェクトの青写真を作成できます。このウィザードを操作すると、EIS またはデータベースのメタデータ情報を参照して、目的の成果物を選択し、配置可能なサービス・オブジェクトや記述を生成できます。メタデータのツリー構造からメタオブジェクト・ノードを選択することにより、EIS またはデータベース・エンティティーのビジネス・オブジェクトを生成することができます。メタデータは、ビジネス・グラフとビジネス・オブジェクトで構成されるサービス・データ・オブジェクトに変換されます。

エンタープライズ・サービス・ディスカバリー・ウィザードでは、以下の操作を実行できます。

- ビジネス・オブジェクトの生成
- ビジネス・オブジェクトでのアプリケーション固有情報の設定
- プロパティーでのアプリケーション固有情報の設定
- Inbound イベントおよび Outbound イベントのサービス記述の入力
- Inbound イベントおよび Outbound イベントの接続記述の入力

アダプターの動作方法

このトピックでは、WebSphere Adapter for Flat Files の動作について説明します。

IBM WebSphere Adapter for Flat Files は、JCA 1.5 仕様に準拠した J2EE コンポーネントで、この製品を使用すると、双方向接続が有効になり、エンタープライズ情報システム (EIS) ファイル・システム上のファイルにアクセスできるようになります。基本的にアダプターは、ファイルによってのみ通信するバックエンド EIS アプリケーションすべてに対するアクセスを、J2EE クライアントに許可します。EIS ファイル・システムは、アダプターが実行されているオペレーティング・システムにマウントされたファイル・システムの場合もあれば、ローカル・ファイル・システムからアクセスできるマップ済みドライブの場合もあります。

イベントおよび応答は、ファイル・システム上でファイルとしてキャプチャーされます。イベント・ファイルは、イベントを受信できるように構成されているエンド

ポイントに送信できます。要求は、任意の J2EE クライアントから開始することができ、応答は要求が処理された後にクライアントに戻されます。

アダプターには次の 2 つの動作モードがあります。

- Inbound イベント処理モード、および
- Outbound 要求処理モード

Inbound イベント処理

このトピックでは、WebSphere Adapter for Flat Files の Inbound イベント処理について説明します。

Inbound イベント処理

Inbound イベント処理は、非同期動作です。バックエンドのエンタープライズ情報システム (EIS) は、ファイルの形式でイベントを生成します。これらのファイルは、ユーザーが構成したイベント・ディレクトリーに保管されます。Inbound 処理モードの場合、アダプターは、ユーザーが構成したイベント・ディレクトリーからイベント・ファイルを一定の間隔でポーリングします。イベント・ファイルがイベント・ディレクトリーに到達すると、アダプターは、イベント・ファイル全体をバイトの集まりとして読み取り、ビジネス・オブジェクト内部にバイトをラップし、ビジネス・オブジェクトをサブスクライブされたエンドポイントに通知します。イベント・ファイルが通知された後、アダプターは、ポーリングされたイベント・ファイルをアーカイブ・ディレクトリーに保存するか、ユーザー構成に基いてイベント・ファイルを削除します。イベント・ディレクトリー、アーカイブ・ディレクトリー、ポーリング間隔、ポーリング数量 (1 回のポーリング・サイクルでポーリングするイベント・ファイルの数) は、すべて構成可能なパラメーターです。

ファイル分割

Inbound イベント処理モードでは、ファイル分割機能もサポートしており、イベント・ファイルが数個のチャンクに分割され、それぞれのチャンクはエンドポイントに対して別々に通知されます。これにより、イベント処理時のメモリー・ロードを削減します。クライアントは、アダプター構成においてファイル分割しきい値とチャンク・サイズを指定します。ファイル分割しきい値プロパティーとファイル・チャンク・サイズ・プロパティーは、`ActivationSpecification` で構成します。ファイル分割しきい値を超えるイベント・ファイルは、ファイル・チャンク・サイズに指定されたサイズでチャンクに分割されます。イベント・ファイルが分割されても、ポーリング数量の考え方に変更はありません。アダプターは、引き続きポーリング数量の限度までのイベント・ファイルをエンドポイントに送信します。ただし、しきい値サイズを超えるイベント・ファイルは、別々のチャンクとしてエンドポイントに送信されるようになります。ファイル分割しきい値が負の値である場合、チャンク機能は使用不可になり、ファイル・チャンク・サイズ値は考慮されません。このモードでは、WebSphere Process Server の管理コンソールからか、または WebSphere Integration Developer を使用してアダプターを構成できます。

チャンク機能が使用可能になると、各チャンクがビジネス・オブジェクトになります。これは、ポーリング数量の値と、エンドポイントが取得するビジネス・オブジェクトの数に違いが生じる可能性があることを意味します。

ファイルをマージする場合、アダプターはチャンク化データの再組み立て所有権を獲得しません。代わりに、外部アプリケーションがチャンクをマージすることができるよう、チャンクについての情報を提供します。チャンクの情報は、ビジネス・オブジェクトの `outputString` プロパティに組み込まれます。チャンク情報には、バイト単位でのチャンク・サイズやイベント ID があります。

イベント管理アーキテクチャーと制御フロー

イベント管理フレームワーク (EMF) には、アダプターからビジネス・オブジェクトを受信するエンドポイントについての情報が維持されています。EMF は、内部でイベント・データ・テーブル (EDT) を使用してイベントを追跡します。

`ActivationSpecification` プロパティである EDT データベース名、EDT ドライバー名、EDT テーブル名、EDT ユーザー名、および EDT ユーザー・パスワードにより、イベント管理フレームワークによって使用されるイベント・データ・テーブルの構成値が決まります。アダプターは、指定されたポーリング期間で、イベント・ディレクトリー内のファイルをポーリングします。これは、ユーザー構成のファイル・マスクとポーリング数量の仕様に一致します。アダプターは、Flat File イベント・テーブルとしてアドレスされるイベント・テーブルを内部で使用し、ポーリングされてはいてもまだエンドポイントに通知されていないイベントの状況をログに記録します。`ActivationSpecification` プロパティである FF イベント・テーブル名により、Flat Files イベント・テーブルの名前が決まります。FFEventTable は Cloudscape™ データベース上でのみ作成され、EDT テーブルで使用されるデータベースでは作成されません。イベントがいったんポーリングされると、アダプターは、Event ID を生成し、状況を「NEW」にして、イベント参照を Flat Files イベント・テーブルに保管します。次いでアダプターは、基本クラスのメソッド・コールバックがさらに進行するのを待ちます。基本クラス関数は、アダプターのメソッドをコールバックしてイベントを処理します。アダプターは、Flat Files イベント・テーブルのイベントの状況を「IN_PROGRESS」に変更します。さらに、ファイル内容をビジネス・オブジェクトでラップし、構成済みエンドポイント全体に同じ内容を通知します。次いで、イベント項目が Flat Files イベント・テーブルから削除されます。イベントは、構成に基づき、必要に応じてアーカイブされます。

Outbound 要求処理

このトピックでは、WebSphere Adapter for Flat Files の Outbound 要求処理について説明します。

Outbound 要求処理の場合、外部の J2EE クライアントは、Service Component Architecture (SCA) を介してアダプターを呼び出すことができます。クライアントが接続を要求すると、その接続要求はアダプターからエンタープライズ情報システム (EIS) に渡されます。Outbound サポートにより、クライアントはアダプターを呼び出して、EIS ファイル・システムで特定の操作を実行します。

Outbound 要求処理モードでサポートされている操作は、以下のとおりです。

- Create
- Append
- Delete
- Retrieve
- Overwrite

- Exist
- List

Outbound 要求処理は、以下の手順で構成されます。

1. Outbound サービス・クライアントがビジネス・オブジェクト・ファクトリー・サービスを検索します。
2. Outbound サービス・クライアントは、ビジネス・オブジェクト・ファクトリーからビジネス・オブジェクトを作成します。
3. Outbound サービス・クライアントは、アダプター・サービスを検索します。
4. Outbound サービス・クライアントは、関数名とビジネス・オブジェクトを渡すことにより、アダプター・サービス上の該当する関数を呼び出します。

注: アダプターは、同期 Outbound 通信のみをサポートし、非同期モードはサポートしません。

SCA フレームワークでのパラメーター渡し

サービス・クライアントを使用すると、ユーザーは、ディレクトリー・パスやファイル名などのプロトコル固有パラメーターを以下の方法で渡すことができます。

1. アダプター・サービスの呼び出し時にパラメーターとして渡されるビジネス・オブジェクトを介して行います。
2. パラメーターは、サービス成果物の生成時に WebSphere Integration Developer で設定することもできます。ビジネス・オブジェクトを介して渡されたパラメーターは、これらの値より優先されます。ディレクトリー・パスとファイル名も、これらのいずれかまたは両方の方法により、合わせて設定してください。

ロケールとグローバリゼーションのサポート

このアダプターは、1 バイト文字セットと 2 バイト文字セットをサポートして、指定された言語でメッセージ・テキストを送信できるように国際化されています。

このアダプターは、アラビア語やヘブライ語などの言語に対して、双方向スクリプト・データの処理をサポートします。双方向のキャパシティーを使用するためには、双方向プロパティーを構成する必要があります。このユーザー・ガイドでは、双方向プロパティーとは、双方向言語サポートの呼び出しを制御するプロパティーのことを指します。

ご使用のエンタープライズ情報システム (EIS) で使用される双方向形式が Windows の標準形式と異なる場合は、双方向サポートに関するすべてのプロパティーが、Windows の標準形式からターゲット EIS の双方向形式に変換されます。また、アダプターは、そのようなデータを WebSphere Process Server に渡す前にも、EIS の形式から Windows の標準形式にデータを変換します。

Java 仮想マシン (JVM) 内の JavaTM ランタイム環境では、データを Unicode 文字コード・セットで表します。Unicode には、知られているほとんどの文字コード・セット (1 バイトとマルチバイトの両方) の文字エンコードが含まれています。IBM WebSphere Business Integration システムのほとんどのコンポーネントは Java で書かれています。そのため、WebSphere Business Integration のシステム・コンポーネント間でデータを転送するときは、ほとんどの場合文字変換の必要はありません。

エラー・メッセージや情報メッセージを適切な言語や個々の国や地域に合った形でログに記録するために、アダプターは、稼働しているシステムのロケールを使用します。

WebSphere Process Server の双方向言語形式

WebSphere Process Server では、ILYNN (暗黙、左から右、オン、オフ、公称を表す英語の頭文字) という双方向言語形式が使用されます。この形式は、Windows の双方向言語形式でもあります。それ以外の双方向言語形式は、WebSphere Process Server に導入する前に変換する必要があります。

5 つの属性を適切な双方向言語形式に設定する必要があります。属性と設定を「双方向属性」という表に示します。

双方向属性

文字の位置	目的	値	説明	デフォルト設定
1	順序のスキーマ	I または V	暗黙 (論理的) または表示	I
2	方向	L R C D	左から右 右から左 コンテキスト上の左から右 コンテキスト上の右から左	L
3	対称スワッピング	Y または N	対称スワッピングのオン/オフ	Y
4	形状の指定	S N I M F B	テキストの形状を指定する テキストの形状を指定しない 初期形状指定 中間形状指定 最終形状指定 分離形状指定	N
5	数字の形状指定	H、C、または N	ヒンディ語、コンテキスト、または公称	N

アダプターには、データを WebSphere Process Server コンポーネントに送信する前に、論理的な左から右の形式に変換する役割があります。

注: ユーザー・インターフェース (ブラウザー) のロケール設定によって、双方向言語の表示形式と編集形式が定義されます。WebSphere Process Server のユーザー・インターフェースでは、ロケール固有の形式を WebSphere Process Server のデフォルトの形式に変換する必要があります。

双方向プロパティーのレベル

双方向プロパティーは、さまざまなレベルで設定できます。これらのプロパティーの詳細と、エンタープライズ・サービス・ディスカバリー・ウィザードを使用したプロパティーの設定方法の詳細については、アダプター・プロジェクトの作成とアダプターの構成のセクションを参照してください。

双方向プロパティの編集

WebSphere Integration Developer のビジネス・オブジェクト・エディターで注釈を使用して、ビジネス・オブジェクトとビジネス・オブジェクト属性の双方向プロパティを編集することができます。注釈は、ビジネス・オブジェクトに格納されています (*.xsd ファイル)。詳しくは、WebSphere Integration Developer の Web サイト (<http://www.ibm.com/software/integration/wid>) に置かれているビジネス・オブジェクト・エディターの資料を参照してください。

特定の双方向プロパティを WebSphere Integration Developer のアセンブリ・エディターを使用して定義しておく、そのプロパティを編集することもできます。双方向プロパティの実行時での使用について詳しくは、双方向言語サポートに関する一般的な技術文書やアダプターの技術文書を参照してください。アセンブリ・エディターについて詳しくは、WebSphere Integration Developer の Web サイト (<http://www.ibm.com/software/integration/wid>) に置かれているアセンブリ・エディターの資料を参照してください。

ビジネス・オブジェクトの概要

このトピック群では、WebSphere Adapter for Flat Files のビジネス・オブジェクトの命名規則、構造、およびサポートされる操作について説明します。

ビジネス・オブジェクトの命名規則

このトピックでは、WebSphere Adapter for Flat Files のビジネス・オブジェクトの命名規則について説明します。

Flat Files Adapter の場合は、FlatFile.xsd と FlatFileBG.xsd という 2 つの定義済みビジネス・オブジェクトがあります。ビジネス・オブジェクトの構造は、Inbound 操作の場合も Outbound 操作の場合も同じです。ビジネス・オブジェクト・スキーマである FlatFile.xsd は、以下の属性で構成されます。

- directoryPath
- fileName
- inputBytes
- outputBytes
- outputString

FlatFileBG.xsd は、前述の属性を含むビジネス・グラフです。

ビジネス・オブジェクトの構造

このトピックでは、WebSphere Adapter for Flat Files のビジネス・オブジェクトの構造について説明します。

Flat Files アダプターのビジネス・オブジェクト構造は、基本 XML スキーマとしてモデル化された汎用 WebSphere Business Integration ビジネス・オブジェクト構造を基にしています。アダプターには、Inbound と Outbound の両方のイベントで使用される同じビジネス・オブジェクト構造があります。

「Flat Files ビジネス・オブジェクト構造」の表には、エンタープライズ情報システムとの間でデータをやり取りするために、Inbound イベント処理と Outbound 要求処理の両方で使用されるビジネス・オブジェクト構造の定義を示します。

Flat Files ビジネス・オブジェクト構造

属性名	属性タイプ	説明
directoryPath	String	出力ファイルに対応するディレクトリーの絶対パス。
filename	String	拡張子付きイベント/出力ファイルの名前。
inputBytes	byte[]	現状のままのファイルの内容。Flat Files リソース・アダプターとの間でやり取りされる。
outputBytes	byte[]	操作の出力バイトを格納。例: Retrieve 操作
outputString	String	さまざまな Outbound 操作の出力を格納。例: List 操作

属性プロパティ

このトピックでは、WebSphere Adapter for Flat Files の属性プロパティについて説明します。

以下に示す「属性プロパティ」の表では、Flat Files アダプターの属性プロパティを定義します。

属性プロパティ

属性プロパティ	説明
Cardinality	単一の基数を持つフラット・ビジネス・オブジェクトだけがサポートされる。
Key および Foreign key	ビジネス・オブジェクト構造のキーは、要求と応答のどちら場合でも、DirectoryPath 属性と Filename 属性の組み合わせになる。キーはすべての操作で同じ。Foreign key はサポートされていない。
Name	ビジネス・オブジェクト属性の名前を示す。
Required	属性フィールドが必要フィールドであることを示す。キーは組み合わせキーなので、DirectoryPath 属性と Filename 属性は両方とも NULL にすることはできない。どちらか一方の値が NULL になっている場合は有効。ただし、NULL 以外の値が 1 つ必要になる。
Special	なし。

属性プロパティ	説明
Type	ビジネス・オブジェクト属性のタイプを示す。タイプには、ストリングまたは、統合オブジェクトを表す複合タイプがある。

サポートされる操作

WebSphere Adapter for Flat Files は、Inbound および Outbound 操作を実行します。このセクションでは、サポートされる操作、およびサポートされる操作のビジネス・オブジェクト構造をリストします。

以下に示す「サポートされる操作」の表では、Flat Files アダプターがサポートしている操作について説明しています。

注: WebSphere Adapter for Flat Files では、動詞はサポートされません。アダプターは、操作のみをサポートします。

サポートされる操作

操作	応答
Append	要求の内容がファイルの終わりに付加される。
Create	指定されたファイル名のファイルが、指定されたディレクトリーに、要求によって送信された内容で作成される。
Delete	要求で指定されたディレクトリーからファイルを削除する。
Exists	要求のファイルが指定のディレクトリーに存在する場合は <code>outputString</code> フィールドに <code>true</code> が戻り、そうでない場合は <code>outputString</code> フィールドに <code>false</code> が戻る。
List	操作により、要求で指定されたディレクトリー内のファイル名すべてを戻す。
Overwrite	ディレクトリーのファイルに要求で指定された内容を上書きする。
Retrieve	応答は、要求で指定されたファイル名のファイル内容を戻す。

以下に示す「Create の場合のビジネス・オブジェクト構造」の表では、Create 操作のビジネス・オブジェクト構造を定義します。

Create の場合のビジネス・オブジェクト構造

名前要求	値要求	名前応答	値応答
DirectoryPath	/home/user/ outputdir	DirectoryPath	/home/user/ outputdir
Filename	Myoutput.out	Filename	Myoutput.out

名前要求	値要求	名前応答	値応答
InputBytes	MSH ^~¥& . 199908180016 ADT^A04 ADT.1.1698593 P	InputBytes	MSH ^~¥& . 199908180016 ADT^A04 ADT.1.1698593 P
OutputBytes	NULL	OutputBytes	NULL
OutputString	NULL	OutputString	NULL

以下に示す「List の場合のビジネス・オブジェクト構造」の表では、List 操作のビジネス・オブジェクト構造を定義します。

List の場合のビジネス・オブジェクト構造

名前要求	値要求	名前応答	値応答
DirectoryPath	/home/user/ outputdir	DirectoryPath	/home/user/ outputdir
Filename	NULL	Filename	NULL
InputBytes	NULL	InputBytes	NULL
OutputBytes	NULL	OutputBytes	NULL
OutputString	NULL	OutputString	File01.out、 File02.out、 File03.out、 File04.out

以下に示す「Exists の場合のビジネス・オブジェクト構造」の表では、Exists 操作のビジネス・オブジェクト構造を定義します。

Exists の場合のビジネス・オブジェクト構造

名前要求	値要求	名前応答	値応答
DirectoryPath	/home/user/ outputdir	DirectoryPath	/home/user/ outputdir
Filename	File01.out	Filename	File01.out
InputBytes	NULL	InputBytes	NULL
OutputBytes	NULL	OutputBytes	NULL
OutputString	NULL	OutputString	True

アダプターのインストール

このトピックでは、WebSphere Adapter for Flat Files のインストール作業について概説します。

アダプターのインストールについては、「IBM WebSphere Adapters のインストール」を参照してください。

アダプター環境

このトピックでは、WebSphere Adapter for Flat Files の環境について説明します。

ハードウェア要件とソフトウェア要件

このアダプターのハードウェア要件とソフトウェア要件については、IBM WebSphere Adapters and IBM WebSphere Business Integration Adapters のハードウェア要件およびソフトウェア要件を参照してください。 WebSphere アダプターのリストからアダプターを選択してください。

アダプター固有のインストール情報

WebSphere Adapter for Flat Files をインストールする前に、追加のインストール要件がいくつかあります。

Inbound 操作

WebSphere Adapter for Flat Files をインストールする前に、Inbound 操作について以下の作業を完了していることを確認します。

1. 指定したイベント・ディレクトリーはファイル・システム内に存在する必要があり、ActivationSpecification 構成に指定されているディレクトリーと一致する必要があります。
2. アーカイブを使用可能にする場合は、指定したアーカイブ・ディレクトリーがファイル・システム内に存在する必要があります。
3. 指定したイベント・データベース・テーブルのデータベース・パラメーターは適切で、対応するデータベース・エンティティーが存在する必要があります。
4. イベント・データ・テーブル・データベースが、WebSphere Process Server に付属の Cloudscape データベースでない場合は、データベースのドライバーをアダプターのクラスパスに置く必要があります。
5. FFEventTable を作成したのがユーザー (アダプターではない) である場合、FFEventTable には、正しいスキーマが必要です。さらに、FFEventTable はアダプターが専用で使用する必要があり、その他のアプリケーションは使用しないようにする必要があります。

注: Flat Files リソース・アダプター固有のデータベースは、アダプターにより、Cloudscape 上にのみ作成されます。

Outbound 操作

アウトプット・ディレクトリーを指定する場合は、これがファイル・システム内に存在することを確認してください。

インストール済みファイルの構造

WebSphere Adapter for Flat Files のインストールでは、WebSphere Process Server に配置可能な Resource Adapter Archive (RAR) ファイルを提供します。RAR ファイルには、アダプターに付属するファイルが格納されています。

WebSphere Adapter for Flat Files の RAR ファイルは、インストール・プロセス中に配信されます。アダプター・パッケージに付属のファイルには、WebSphere Adapter for Flat Files に割り振られるコンポーネント ID である「CWYFF」が前に付きます。「パッケージ化構造」の表に、RAR ファイルに格納されているファイルの内容を示します。

パッケージ化構造

ファイル	説明
CWYBS_AdapterFoundation.jar	ファウンデーション・クラス JAR ファイル。
CWYFF_FlatFile.jar	このファイルには、Flat Files リソース・アダプター固有のサブクラスが格納されている。
icu4j_3_2.jar	グローバリゼーションに必要な ICU4J ライブラリー。
meta-inf¥Manifest.mf	
meta-inf¥discovery-service.xml	このファイルは、エンタープライズ・サービス・ディスカバリー・ウィザードによって使用される。
meta-inf/ra.xml	配置記述子。
Flatfile.xsd	このファイルは Dependencies の下にパッケージ化される。このファイルには、Flat Files リソース・アダプターが必要とする WebSphere ビジネス・オブジェクト構造のサンプルが格納されている。

アダプターの配置 (Inbound 操作の場合)

WebSphere Adapter for Flat Files は、インストールしたら配置する必要があります。配置は、プロジェクトの作成、サービス成果物の生成、参照インディキングの生成、アプリケーションのエクスポート、WebSphere Process Server 上での実行を目的とするアプリケーションのインストールで構成されます。

アダプターは、Resource Adapter Archive (RAR) ファイルとして配布されます。

1. RAR ファイルをインポートすることにより、アダプターを WebSphere Integration Developer にインストールします。
2. アダプターをインストールしたら、Enterprise Application Archive (EAR) ファイルを生成します。
3. 管理コンソールを使用して EAR ファイルを WebSphere Process Server に配置します。

注: WebSphere Integration Developer が稼働するプラットフォームは Windows[®] と Linux[®] のみですが、WebSphere Process Server は Windows、Linux、UNIX[®] の各プラットフォームで稼働します。

アダプターを配置することは、基本的には、ほかのコンポーネントを WebSphere Process Server に配置することと同じです。WebSphere Process Server へのコンポーネントの配置について詳しくは、IBM WebSphere Integration Developer バージョン 6.0 の資料を参照してください。

配置の前提条件

アダプターを配置する前に、以下の製品をインストールする必要があります。

- WebSphere Integration Developer V6.0 (WebSphere Integration Developer)
- WebSphere Adapter for Flat Files (WebSphere Integration Developer と同じマシンにインストールしたもの)
- WebSphere Process Server 管理コンソール

WebSphere Process Server のインストール手順説明については、IBM WebSphere Process Server for Multiplatforms バージョン 6.0 の資料を参照してください。

アダプター・プロジェクトの作成 (Inbound 操作の場合)

アダプターを配置するための最初のステップは、WebSphere Integration Developer へのインストール時にインストールされたアダプターの RAR ファイルをインポートすることです。RAR ファイルをインポートすると、自動的に新規プロジェクトが作成されます。

WebSphere Integration Developer を使用して、以下の手順を実行します。このツールの詳細については、IBM WebSphere Integration Developer バージョン 6.0 の資料を参照してください。

1. WebSphere Integration Developer を起動します。
2. J2EE パースペクティブに切り替えます。
3. WebSphere Integration Developer に RAR ファイルをインポートするには、J2EE パースペクティブのウィンドウ・フレームを右クリックし、ポップアップ・メニューから「ファイル」→「インポート」を選択します。
4. RAR ファイルを選択して、「次へ」をクリックします。「コネクター・インポート」ウィンドウが開きます。
5. RAR ファイルのインポート元のロケーション (インストール時にアダプター・ファイルをコピーしたロケーション) を選択して、プロジェクト名を指定します。
6. 「モジュールを EAR プロジェクトに追加」チェック・ボックスのチェックマークを外します。
7. 「終了」をクリックして、RAR ファイルをインポートします。これにより、ワークスペースに新しい J2EE コネクター・プロジェクトが作成されます。

これで、WebSphere Integration Developer に RAR ファイルをインポートすることによって新規アダプター・プロジェクトが作成されたため、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、サービス成果物を生成する準備ができました。

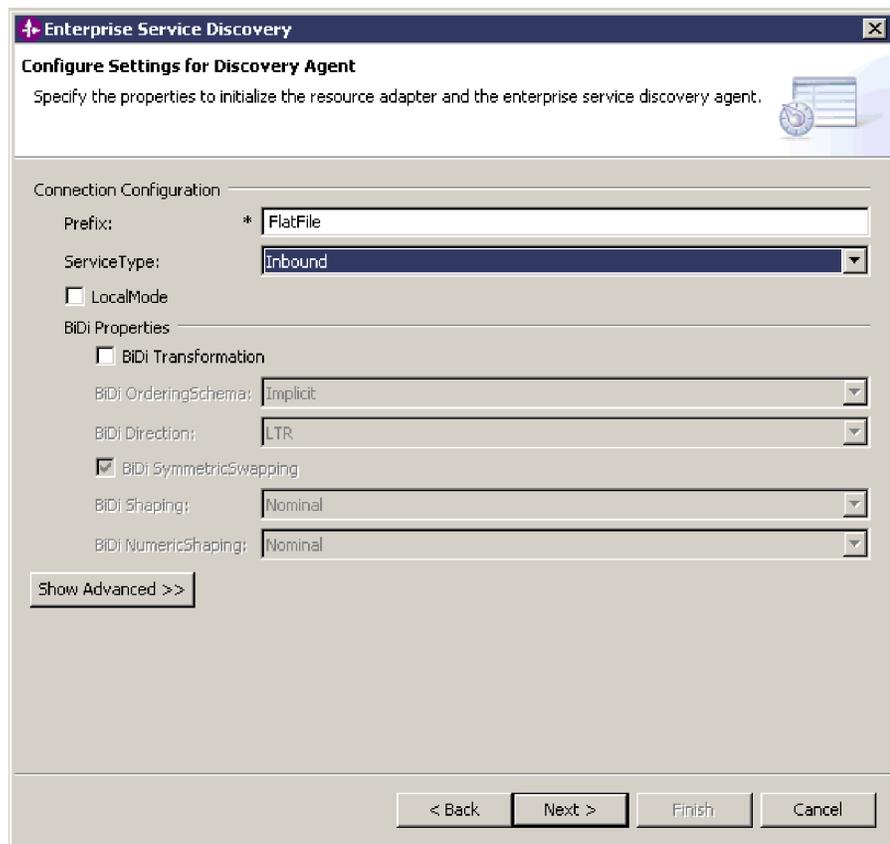
サービス成果物の生成 (Inbound 操作の場合)

Inbound 操作の場合のサービス成果物の生成は、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

Inbound 操作の場合のサービス成果物の生成プロセスを完了して、アダプターを初めて構成するときに必要なすべての情報を入力します。エンタープライズ・サービス・ディスカバリー・ウィザードからの出力は、ビジネス・インテグレーション・モジュールに保管します。ここでは、ビジネス・オブジェクト、インポート・ファイル (ConnectionFactory で定義されている Outbound 処理の構成が記述されてい

る)、エクスポート・ファイル (ActivationSpecification で定義されている Inbound イベント処理の構成が記述されている)、および Web サービス記述言語 (WSDL) ファイルが格納されています。

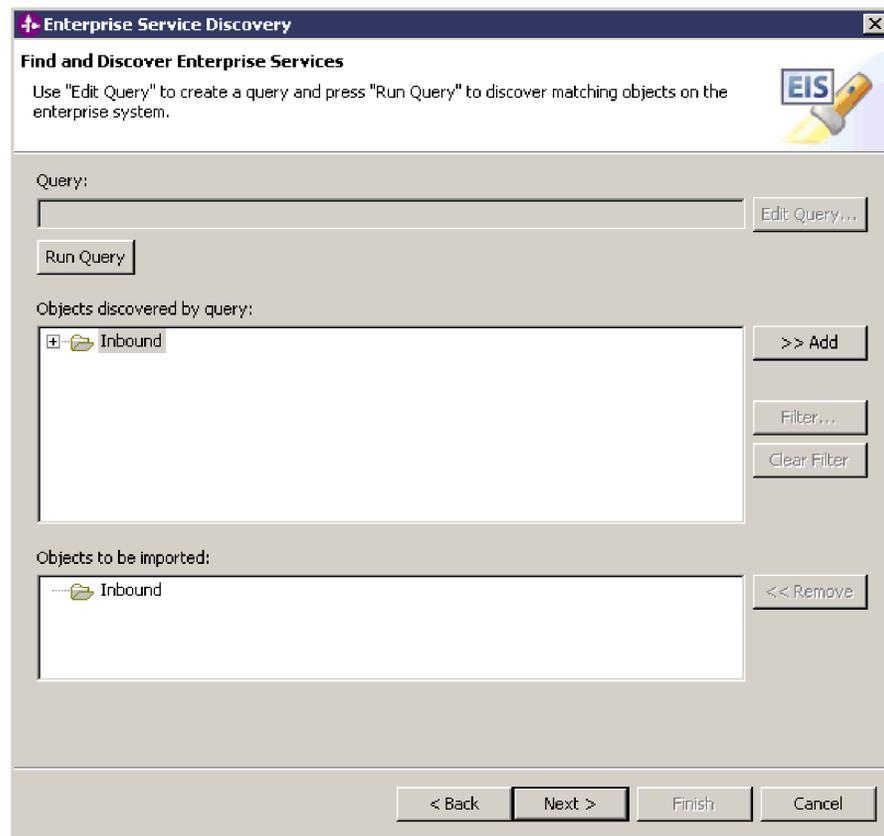
1. WebSphere Integration Developer ウィンドウで、Business Integration パースペクティブに切り替えるために、メニュー・バーから「ウィンドウ」 → 「パースペクティブを開く」 → 「その他 (Other)」を選択します。すべてのパースペクティブが表示されます。WebSphere Integration Developer で、Business Integration パースペクティブを選択します。
2. WebSphere Integration Developer の Business Integration パースペクティブで、フレームを右クリックし、ポップアップ・メニューから「新規」 → 「エンタープライズ・サービス・ディスカバリー (Enterprise Service Discovery)」を選択します。「エンタープライズ・サービス・リソース・アダプターの選択 (Select an Enterprise Service Resource Adapter)」ウィンドウが表示されます。
3. 「CWYFF_FlatFile」コネクタ・プロジェクトから「IBM WebSphere Adapter for Flat Files (バージョン 6.0.0) (IBM WebSphere Adapter for Flat Files (version 6.0.0))」を選択してから、「次へ」をクリックします。
4. 下の図に示すように、「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウで、「サービス・タイプ」に「Inbound」を選択して接続プロパティを設定し、「次へ」をクリックします。



「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウ

5. (オプション) BiDi のサポートを使用可能にするには、「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウで「**BiDi 変換**」チェック・ボックスを選択してから、該当する BiDi 形式を選択します。
6. (オプション) 「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウの下部で、「拡張を表示」ボタンをクリックします。ここでは、ロギング・レベルやトレース・レベルを設定できます。「次へ」をクリックします。
7. 「エンタープライズ・サービスの検索と発見 (Find and Discover Enterprise Services)」ウィンドウで、「**照会の実行 (Run Query)**」ボタンをクリックして、Flat Files アダプターのメタデータ・ツリーを表示します。
8. Inbound 操作の場合、「照会で発見したオブジェクト (Objects discovered by query)」の下で、メタデータ・ツリーの「Inbound」ルート・ノードを選択します。後出の図に示すように、インポートの対象として選択できるオブジェクトは、「照会で発見したオブジェクト (Objects discovered by query)」ボックスに表示されます。インポートするオブジェクトを強調表示してから、「**選択されているものを追加 (Add Selected)**」ボタンをクリックして、そのオブジェクトを「インポートするオブジェクト (Objects to be imported)」ボックスに追加します。「次へ」をクリックします。

注: 「照会で発見したオブジェクト (Objects Discovered by query)」テキスト・ボックスからオブジェクトを削除するには、削除するオブジェクトを強調表示して、「**選択されているものを除去**」ボタンをクリックします。

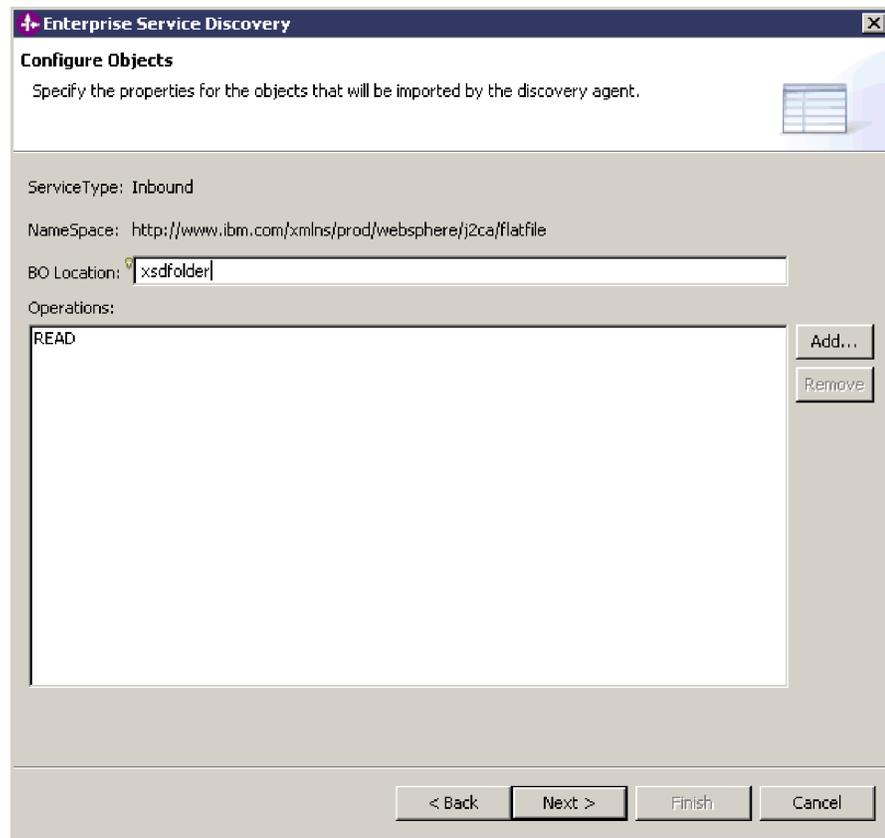


「エンタープライズ・サービスの検索と発見 (Find and Discover Enterprise Services)」ウィンドウ

9. 「オブジェクトの構成 (Configure Objects)」ウィンドウで、Discovery Agent によってインポートされるオブジェクトのプロパティを指定します。後出の図に示すように、BO ロケーション・プロパティの場合は、.xsd ファイルの生成場所となる Business Integration モジュール内部のフォルダー名を指定します。「次へ」をクリックします。

Inbound でサポートされる操作には、次のものがあります。

- Read



「オブジェクトの構成 (Configure Objects)」ウィンドウ

10. 「成果物の生成 (Generate Artifacts)」ウィンドウで、以下の操作を行います。
 - a. 「新規」をクリックして、新しいビジネス・インテグレーション・モジュールを作成します。
 - b. 「新規モジュール」ウィンドウで、モジュール名を入力して、「終了」をクリックします。
 - c. 「成果物の生成 (Generate Artifacts)」ウィンドウで、フォルダー名を入力します。
 - d. 後出の図に示すように、「実行時にエンタープライズ情報システムとの接続に使用される接続プロパティを指定する (Specify the connection properties which will be used to connect to the Enterprise Information System at

runtime)」の下で、「発見された接続プロパティを使用する (Use discovered connection properties)」をクリックします。

Enterprise Service Discovery

Generate Artifacts

JNDI Lookup Name: cannot be empty.

Properties for Interface

Module: FFInboundModule New...

Namespace: http://FFInboundModule/FFInboundFolder/FlatFile

Use Default Namespace

Folder: FFInboundFolder Browse...

Name: * FlatFileInboundInterface

Description:

Edit operation names...

Deploy connector with module

Specify the connection properties which will be used to connect to the Enterprise Information System at runtime:

Use connection properties specified on server

Use discovered connection properties

J2C Authentication Data Entry:

JNDI Lookup Name: *

< Back Next > Finish Cancel

「成果物の生成 (Generate Artifacts)」ウィンドウ

e. イベント配布データベース・プロパティを入力します。

注: アスタリスク (*) が付いているプロパティは必須です。

f. Flat Files アダプターの ActivationSpecification プロパティを入力します。

注: アスタリスク (*) が付いているプロパティは必須です。

g. ログ値とトレース値を入力します。

注: アスタリスク (*) が付いているプロパティは必須です。

h. (オプション) BiDi プロパティを入力します。

注: 「**BiDi をオフにする**」チェック・ボックスをクリアしてください。そうでなければ、BiDi サポートが呼び出されません。

11. プロパティ値を入力し終わったら、「終了」をクリックします。

これで、エンタープライズ・サービス・ディスカバリー・ウィザードで作成される SCA コンポーネントの参照バインディングを、WebSphere Integration Developer を使用して生成できるようになりました。

関連タスク

49 ページの『ロギング可能化』

WebSphere Adapter for Flat Files は、イベント処理の状況を判別するために表示できるログ・ファイルを維持しています。ログ・ファイル、および各ログ・エントリーの日付、時刻、イベントによって、アダプターに関連するすべてのイベントやエラーを追跡します。アダプターはエラーや警告の状態の検出時にエラー・メッセージを記録するため、ログ・ファイルは問題のトラブルシューティングを開始するための有効な情報源となります。

51 ページの『トレースの使用可能化』

トレースによって、アダプター・ログ・ファイルに収集するエラーや警告のレベルが決まります。トレース・レベルを定義することによって、アダプターの処理に関するメッセージをたどって調べることができます。

39 ページの『プロパティの構成』

WebSphere Process Server 管理コンソールを使用して、アダプターのプロパティを構成することができます。

関連資料

44 ページの『カスタム・プロパティ』

このトピックでは、WebSphere Adapter for Flat Files のデフォルトの構成プロパティについて説明します。

42 ページの『J2C アクティブ化仕様プロパティ』

このトピックでは、(メッセージ・エンドポイント・プロパティとしても知られる) J2C アクティブ化仕様プロパティについて説明します。この仕様は、J2EE Connector Architecture Specification の ActivationSpecification インターフェースに対応します。

46 ページの『リソース・アダプター・プロパティ』

アダプターを初めて構成するとき (および後で WebSphere Process Server の管理コンソールを介して構成する場合)、エンタープライズ・サービス・ディスカバリー・ウィザードを使用すると、リソース・アダプター・プロパティを構成できます。このカテゴリーのプロパティには、ロギングとトレースのプロパティおよびアダプター固有のプロパティがあります。

参照バインディングの生成 (Inbound 操作の場合)

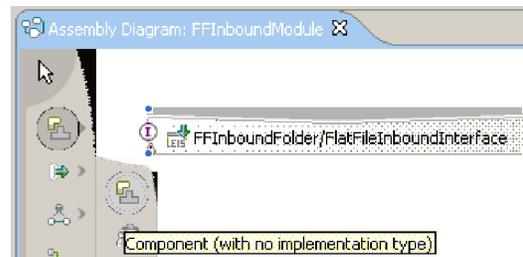
サービス成果物を生成した後に、WebSphere Integration Developer を使用して、参照バインディングを生成することができます。

前提条件: ワークスペース上にアダプター・プロジェクトを作成し、構成しておく必要があります。アダプター・プロジェクトを作成した後、参照バインディングを生成して、サービス・コンポーネントにバインドする必要があります。

参照バインディングは、WebSphere Business Integration のほかの SCA コンポーネントがアダプターへのアクセス時に使用します。アダプターをほかのサーバー・プロセスにリンクするために、プロジェクト・モジュールからアダプターへの参照を作成します。

1. WebSphere Integration Developer ウィンドウで、Business Integration パースペクティブに切り替えるために、メニュー・バーから「ウィンドウ」 → 「パースペクティブを開く」 → 「その他 (Other)」を選択します。すべてのパースペクティブが表示されます。WebSphere Integration Developer で、Business Integration パースペクティブを選択します。

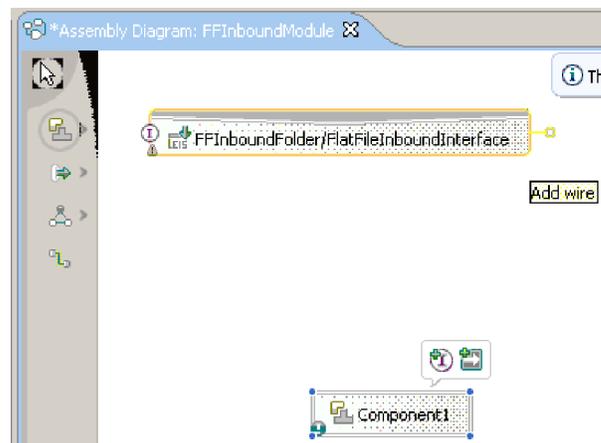
2. WebSphere Integration Developer の Business Integration パースペクティブで、モジュールを右クリックし、「アプリケーションから開く」 → 「アセンブリー・エディター」を選択します。「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウが表示され、モジュールのインポート・コンポーネントがビューに示されます。
3. 新規コンポーネントを作成するには、「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウの左側のフレーム (縦枠) の一番上のアイコンをクリックします。



「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウ

新しいアイコン・メニューが表示されます。上の図は、新しいアイコン・メニューを示しています。

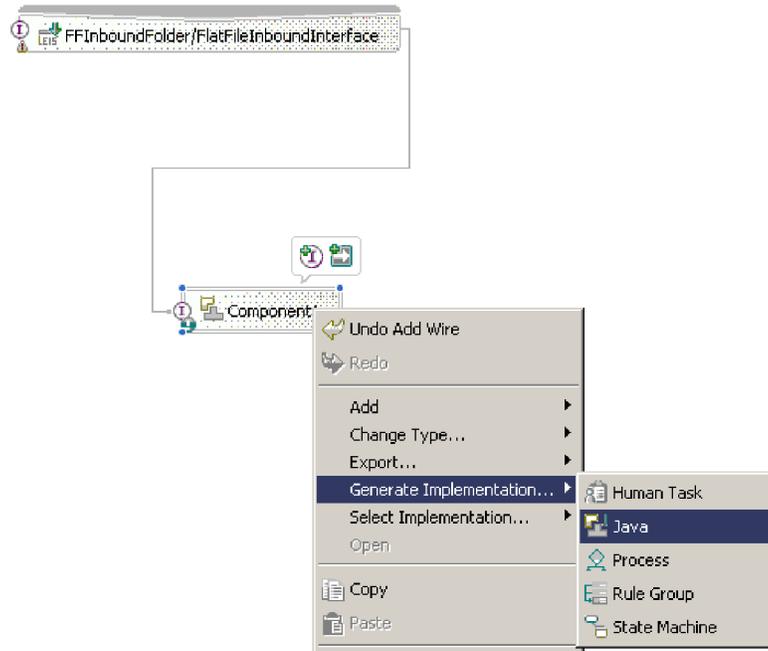
4. コンポーネントを実装なしで作成します。以下の図に示すように、「コンポーネントの作成 (Create a Component)」アイコンを「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウヘドラッグします。



「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウ

5. コンポーネントをインポート・モジュールに接続します。モジュールのインポート・コンポーネントをクリックして、新しいコンポーネントヘドラッグします。これにより、インポート・コンポーネントから新しいコンポーネントへのワイヤーが描画されます。
6. 「ワイヤーの追加 (Add Wire)」ダイアログ・ボックスで、「OK」をクリックします。「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウに新しいコンポーネントが、モジュールのインポート・コンポーネントに接続するワイヤーが付いた状態で表示されます。

7. エンドポイントのリスナーのような働きをする Java™ コンポーネントを作成します。 Inbound 操作のイベント配信の間に、アダプターはコンポーネントの実装の READ メソッドを呼び出し、Inbound ビジネス・オブジェクトをパラメーターとして渡します。
 - a. 以下の図に示すように、コンポーネントを右クリックして、「実装の生成 (Generate implementation)」 → 「Java」を選択します。



「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウ

- b. Java コードを作成するパッケージを選択して、「OK」をクリックします。
 - c. 以下の図に示すように、プロンプトが表示されたらパッケージ名を入力して、「OK」をクリックします。



「パッケージ名」ダイアログ・ボックス

- d. 「実装の生成 (Generate Implementation)」ウィンドウで、「OK」をクリックします。
 - e. 生成された Java 実装で、READ メソッドまでスクロールします。以下の図に示すように、READ メソッド内にカスタム・コードを追加して、配信されたビジネス・オブジェクトを特定のビジネス・ニーズに従って処理することができます。

```

public void READ(DataObject rEADInput) {
    System.out.println("Component1Impl>entered READ");
    DataObject dataObject = rEADInput.getDataObject("FlatFile");
    if(dataObject != null) {
        System.out.println("Component1Impl>directoryPath = " + dataObject.getString("directoryPath"));
        System.out.println("Component1Impl>fileName = " + dataObject.getString("fileName"));
        byte[] inputBytes = dataObject.getBytes("inputBytes");
        byte[] outputBytes = dataObject.getBytes("outputBytes");
        if(inputBytes != null) {
            System.out.println("Component1Impl>inputBytes = " + new String(inputBytes));
        }
        if(outputBytes != null) {
            System.out.println("Component1Impl>outputBytes = " + new String(outputBytes));
        }
        System.out.println("Component1Impl>outputString = " + dataObject.getString("outputString"));
    }
    else {
        System.out.println("Component1Impl>Data object = null");
    }
}
}

```

READ コードのサンプル

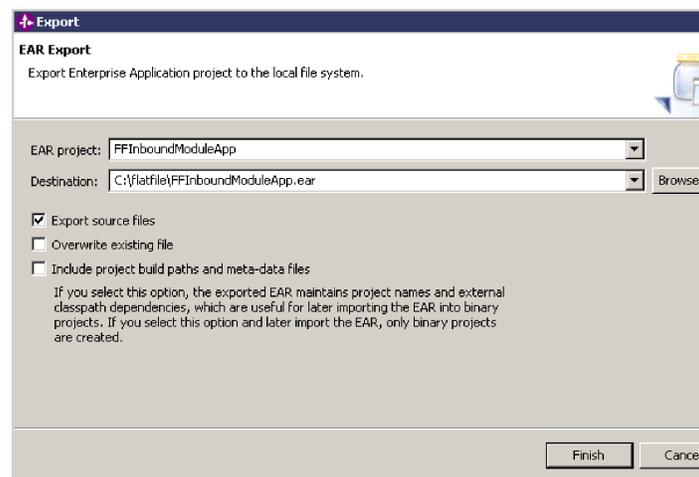
- 「ファイル」 → 「保管」をクリックして、ファイルを保管します。

これで、EAR ファイルをエクスポートする準備ができました。

EAR ファイルのエクスポート (Inbound 操作の場合)

プロジェクトを実行するためには、WebSphere Integration Developer を使用して、プロジェクトを EAR ファイルにエクスポートする必要があります。

- WebSphere Integration Developer ウィンドウで、J2EE パースペクティブに切り替えるために、メニュー・バーから「ウィンドウ」 → 「パースペクティブを開く」 → 「その他 (Other)」を選択します。すべてのパースペクティブが表示されます。WebSphere Integration Developer で、J2EE パースペクティブを選択します。
- WebSphere Integration Developer ツールの J2EE パースペクティブ・ウィンドウで、モジュールを右クリックし、ポップアップ・メニューから「エクスポート」を選択します。「エクスポート」ウィンドウが表示されます。
- 「エクスポート」 - 「選択」ウィンドウから、EAR ファイルを選択します。「エクスポート」 - 「EAR エクスポート」ウィンドウが表示されます。
- 以下の図に示すように、「EAR エクスポート」ウィンドウで、EAR プロジェクトと宛先ディレクトリー (プロジェクトのエクスポート先となる EAR ファイル名を含むディレクトリー) を選択します。



「EAR エクスポート」ウィンドウ

5. 「終了」をクリックします。

これで、プロジェクトの Enterprise Application Archive (.EAR) ファイルへのエクスポートが終了し、アプリケーションをインストールする準備ができました。

アプリケーションのインストール (Inbound 操作の場合)

アプリケーション・プロジェクト・モジュールのインストールは、配置プロセスの最後のステップです。

アプリケーションをインストールして実行すると、プロジェクト・モジュールに組み込まれているアダプターが、インストールしたアプリケーションの一部として稼働します。

1. WebSphere Process Server 管理コンソールで、「**アプリケーション (Applications)**」 → 「**新規アプリケーションのインストール (Install New Applications)**」をクリックします。
2. エンタープライズ・アプリケーションのリストから EAR ファイルを選択します。
3. 新しいアプリケーションへのパスの下に、EAR ファイルのパスを指定して、「**次へ**」をクリックします。
4. 「リソースへのリソース参照のマップ (Map resource reference to resources)」というステップのタイトルが付いたウィンドウが表示されるまで、さまざまな「ステップ」ウィンドウで「次へ」のクリックを続けます。
5. すべてのインストール・オプションの要約が表示されます。以下の図に示すように、「要約」ウィンドウで、すべてのオプションが意図するとおりに選択されて

いることを確認して、「終了」をクリックします。

Enterprise Applications

Specify options for installing enterprise applications and modules.

Step 1 Select installation options

Step 2 Map modules to servers

Step 3 Provide options to perform the EJB Deploy

Step 4 Provide listener bindings For message-driven beans

Step 5 Provide JNDI Names for Beans

Step 6 Map JCA resource references to resources

Step 7 Map virtual hosts for Web modules

Step 8 Ensure all unprotected 2.x methods have the correct level of protection

→ **Step 9: Summary**

Summary

Summary of installation options

Options	Values
Use Binary Configuration	No
Deploy EJB option - Class path	
Create MBeans for resources	Yes
Cell/Node/Server	Click here
Reload interval in seconds	
Enable class reloading	No
Deploy EJB option - Database type	DB2UDB_V81
Deploy EJB option - Database schema	
Process embedded configuration	Yes
Application name	FFInboundModuleApp
Deploy EJB option - RMIC	
Validate Input off/warn/fail	warn
Directory to install application	
Distribute application	Yes
Deploy Web services	No
Pre-compile JSP	No
Deploy enterprise beans	Yes

Previous Finish Cancel

WebSphere Process Server 管理コンソール

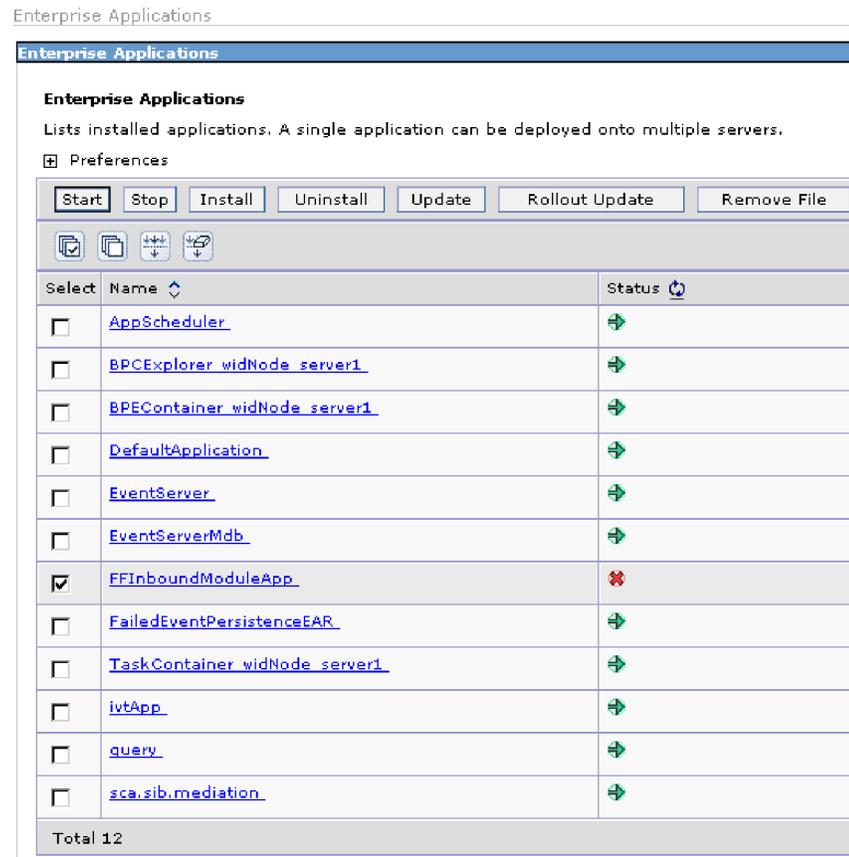
- 「次へ」をクリックします。すべてのオプションが意図するとおりに選択されていることを確認して、「終了」をクリックします。
- インストール・メッセージのリストが表示されます。リストの最後に、アプリケーションが正常にインストールされたことを示すメッセージが含まれていることを確認します。
- インストール・メッセージのリストの最後に表示されている「**マスター構成に保管 (Save to Master Configuration)**」リンクをクリックします。「エンタープライズ・アプリケーション」ウィンドウが表示されます。
- 「保管」をクリックして、アプリケーションを保管します。この時点で、アプリケーションが配置され、配置されたアプリケーション用の「エンタープライズ・アプリケーション」ウィンドウが表示されます。

これで、アプリケーションが配置され、正しく構成されました。次のステップは、アプリケーションの実行です。

アプリケーションの実行 (Inbound 操作の場合)

アプリケーションの配置が終了したら、このアプリケーションを実行することができます。アダプターはアプリケーション・プロジェクトに組み込まれているため、アプリケーションを実行すると、アダプターが起動され、実行を開始します。

1. WebSphere Process Server 管理コンソールで、「アプリケーション (Applications)」 → 「エンタープライズ・アプリケーション」を選択します。
2. 以下の図に示すように、アプリケーションのチェック・ボックスを選択して、「開始」ボタンをクリックします。アプリケーションが実行を開始します。



WebSphere Process Server 管理コンソール

アダプターの配置 (Outbound 操作の場合)

WebSphere Adapter for Flat Files は、インストールしたら配置する必要があります。配置は、プロジェクトの作成、作成したプロジェクトへの外部依存関係の追加、サービスの構成、WebSphere Process Server 上での実行を目的とするアプリケーションの配置で構成されます。

アダプターは、Resource Adapter Archive (RAR) ファイルとして配布されます。

1. RAR ファイルをインポートすることにより、アダプターを WebSphere Integration Developer にインストールします。
2. アダプターをインストールしたら、Enterprise Application Archive (EAR) ファイルを生成します。
3. 管理コンソールを使用して EAR ファイルを WebSphere Process Server に配置します。

注: WebSphere Integration Developer が稼働するプラットフォームは Windows と Linux のみですが、WebSphere Process Server は Windows、Linux、UNIX の各プラットフォームで稼働します。

アダプターを配置することは、基本的には、ほかのコンポーネントを WebSphere Process Server に配置することと同じです。WebSphere Process Server へのコンポーネントの配置について詳しくは、IBM WebSphere Integration Developer バージョン 6.0 の資料を参照してください。

配置の前提条件

アダプターを配置する前に、以下の製品をインストールする必要があります。

- WebSphere Integration Developer V6.0 (WebSphere Integration Developer)
- WebSphere Adapter for Flat Files (WebSphere Integration Developer と同じマシンにインストールしたもの)
- WebSphere Process Server 管理コンソール

WebSphere Process Server のインストール手順説明については、IBM WebSphere Process Server for Multiplatforms バージョン 6.0 の資料を参照してください。

アダプター・プロジェクトの作成 (Outbound 操作の場合)

アダプターを配置するための最初のステップは、WebSphere Integration Developer へのインストール時にインストールされたアダプターの RAR ファイルをインポートすることです。RAR ファイルをインポートすると、自動的に新規プロジェクトが作成されます。

WebSphere Integration Developer を使用して、以下の手順を実行します。このツールの詳細については、IBM WebSphere Integration Developer バージョン 6.0 の資料を参照してください。

1. WebSphere Integration Developer を起動します。
2. J2EE パースペクティブに切り替えます。
3. WebSphere Integration Developer に RAR ファイルをインポートするには、J2EE パースペクティブのウィンドウ・フレームを右クリックし、ポップアップ・メニューから「ファイル」→「インポート」を選択します。
4. RAR ファイルを選択して、「次へ」をクリックします。「コネクター・インポート」ウィンドウが開きます。
5. RAR ファイルのインポート元のロケーション (インストール時にアダプター・ファイルをコピーしたロケーション) を選択して、プロジェクト名を指定します。
6. 「モジュールを EAR プロジェクトに追加」チェック・ボックスのチェックマークを外します。
7. 「終了」をクリックして、RAR ファイルをインポートします。これにより、ワークスペースに新しい J2EE コネクター・プロジェクトが作成されます。

これで、WebSphere Integration Developer に RAR ファイルをインポートすることによって新規アダプター・プロジェクトが作成されたため、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、サービス成果物を生成する準備ができました。

サービスの構成 (Outbound 操作の場合)

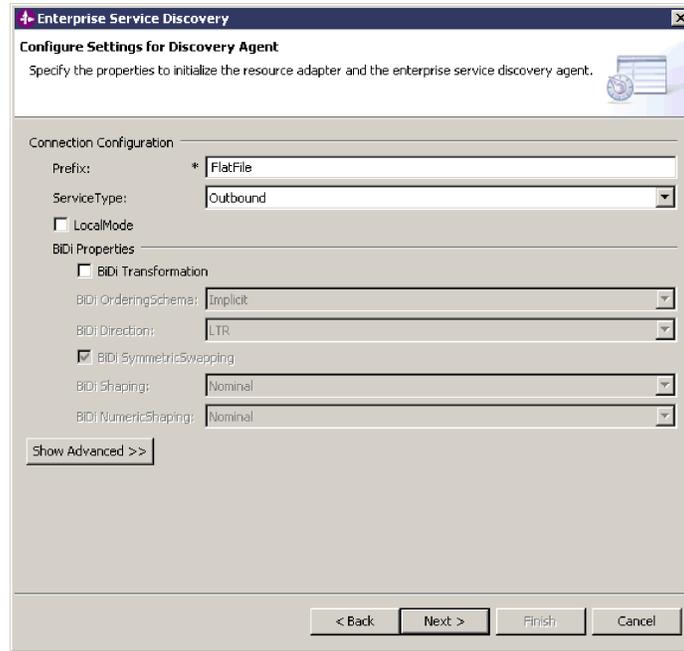
構成プロセスは、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

処理を完了するにつれ、アダプターを最初に構成するときに必要な情報すべてを入力することになります。エンタープライズ・サービス・ディスカバリー・ウィザードからの出力は、ビジネス・インテグレーション・モジュールに保管します。ここには、ビジネス・オブジェクト、インポート・ファイル (ActivationSpecification で定義されている Outbound 処理が記述されている)、エクスポート・ファイル (InteractionSpecification で定義されている Inbound イベント処理が記述されている)、および Web サービス記述言語 (WSDL) ファイルが格納されています。

重要: 配置作業中、サービスを初めて構成するときに J2C アクティブ化仕様プロパティを指定すると、これらのプロパティ設定は元の状態のまま維持されます (言い換えると、アプリケーションのインストール後は、WebSphere Process Server の管理コンソールを介してプロパティを後から更新することはできません。プロパティは引き続き更新できますが、これらのプロパティが配置時にすでに設定されている場合、アダプターは更新された値を認識しないためです)。何らかの理由で、アプリケーションのインストール後に管理コンソールを介して J2C アクティブ化仕様プロパティを設定する場合は、配置作業中の J2C アクティブ化仕様プロパティの設定は控える必要があります。なお、J2C 接続ファクトリー・プロパティの場合は、配置作業中に設定しても、アプリケーションのインストール後に管理コンソールを介して更新することができます。

1. WebSphere Integration Developer ウィンドウで、Business Integration パースペクティブに切り替えるために、メニュー・バーから「**ウィンドウ**」→「**パースペクティブを開く**」→「**その他 (Other)**」を選択します。すべてのパースペクティブが表示されます。WebSphere Integration Developer で、Business Integration パースペクティブを選択します。
2. WebSphere Integration Developer の Business Integration パースペクティブで、フレームを右クリックし、ポップアップ・メニューから「**新規**」→「**エンタープライズ・サービス・ディスカバリー (Enterprise Service Discovery)**」を選択します。「エンタープライズ・サービス・リソース・アダプターの選択 (Select an Enterprise Service Resource Adapter)」ウィンドウが表示されます。
3. 「CWYFF_FlatFile」コネクタ・プロジェクトから「**IBM WebSphere Adapter for Flat Files (バージョン 6.0.0) (IBM WebSphere Adapter for Flat Files (version 6.0.0))**」を選択してから、「**次へ**」をクリックします。「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウが表示されます。

4. 「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウで、後出の図に示すように「サービス・タイプ」に対して「Outbound」を選択して、接続プロパティを設定します。「Outbound」は、デフォルトのサービス・タイプです。

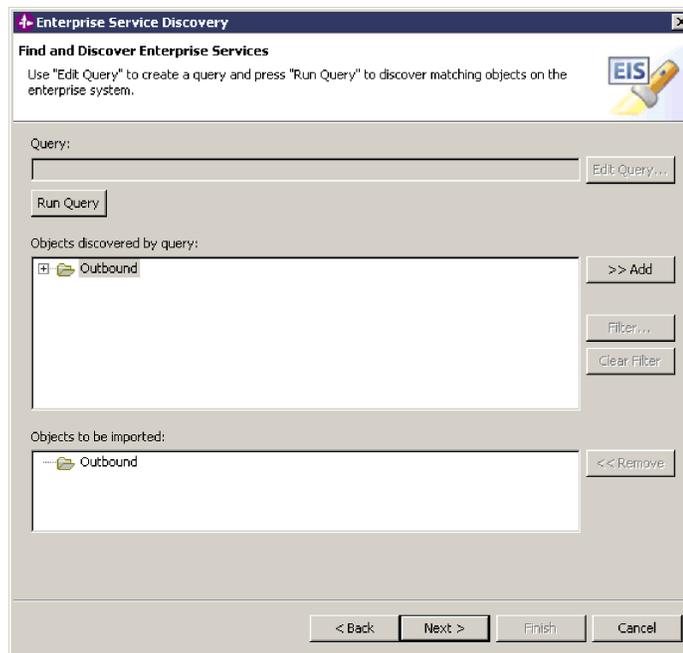


「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウ

5. (オプション) BiDi のサポートを使用可能にするには、「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウで「**BiDi 変換**」チェック・ボックスを選択してから、該当する BiDi 形式を選択します。
6. Discovery Agent の「プロパティの初期設定」ウィンドウの下部で、「拡張を表示」ボタンをクリックします。ここでは、ロギング・レベルやトレース・レベルを設定できます。「次へ」をクリックします。
7. 「エンタープライズ・サービスの検索と発見 (Find and Discover Enterprise Services)」ウィンドウで、「**照会の実行 (Run Query)**」ボタンをクリックして、Flat Files アダプターのメタデータ・ツリーを表示します。
8. 「発見したオブジェクト (Objects Discovered)」の下で、メタデータ・ツリーの「Inbound」ルート・ノードを選択します。後出の図に示すように、インポートの対象として選択できるオブジェクトは、「発見したオブジェクト (Objects Discovered)」ボックスに表示されます。インポートするオブジェクトを強調表示してから、「**選択されているものを追加 (Add Selected)**」ボタンをクリックして、そのオブジェクトを「インポートするオブジェクト (Objects to be imported)」ボックスに追加します。「次へ」をクリックします。

注: 「発見したオブジェクト (Objects discovered)」ボックスからオブジェクトを削除するには、削除するオブジェクトを強調表示して、「**選択されてい**

るものを除去」ボタンをクリックします。

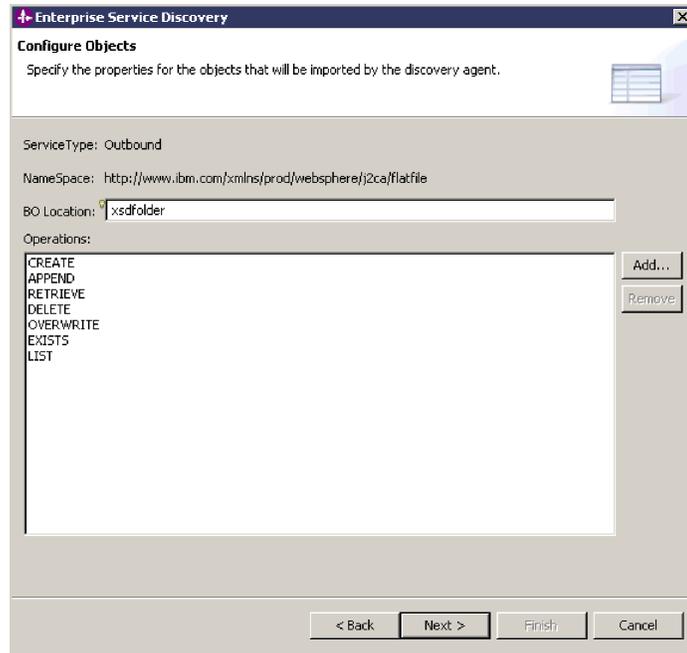


「エンタープライズ・サービスの検索と発見 (Find and Discover Enterprise Services)」ウィンドウ

9. 「オブジェクトの構成 (Configure Objects)」ウィンドウで、Discovery Agent によってインポートされるオブジェクトのプロパティを指定します。後出の図に示すように、BO ロケーション・プロパティの場合は、.xsd ファイルの生成場所となる Business Integration モジュール内部のフォルダー名を指定します。「次へ」をクリックします。

Outbound でサポートされる操作には、次のものがあります。

- Create
- Append
- Retrieve
- Delete
- Overwrite
- Exists
- List



「オブジェクトの構成 (Configure Objects)」ウィンドウ

10. 「成果物の生成 (Generate Artifacts)」ウィンドウで、以下の操作を行います。
- a. SCA 成果物 (ビジネス・オブジェクト、そのプロパティ、インポート・ファイル、エクスポート・ファイル、および WSDL) の保管先のモジュールの名前を指定し、「新規」をクリックして、新しいビジネス・インテグレーション・モジュールを作成します。

「新規モジュール」ウィンドウが表示されます。

- b. 「新規モジュール」ウィンドウで、モジュール名を入力して、「終了」をクリックします。
- c. 「成果物の生成 (Generate Artifacts)」ウィンドウで、フォルダー名を入力します。
- d. 「成果物の生成 (Generate Artifacts)」ウィンドウで、「発見された接続プロパティを使用する (Use discovered connection properties)」を選択して、「JNDI ルックアップ名 (JNDI Lookup Name)」を入力し、「次へ」をクリックします。
- e. 「成果物の生成 (Generate Artifacts)」ウィンドウで、Flat Files の ManagedConnectionFactory プロパティを指定します。

注: アスタリスク (*) が付いているプロパティは必須です。

- f. 「成果物の生成 (Generate Artifacts)」ウィンドウで、ResourceAdapter プロパティを指定します。ここでは、ロギング・ファイルおよびトレース・ファイルのサイズとファイル名を指定することもできます。

注: アスタリスク (*) が付いているプロパティは必須です。

- g. (オプション) BiDi プロパティを入力します。

注: 「BiDi をオフにする」チェック・ボックスをクリアしてください。そうでなければ、BiDi サポートが呼び出されません。

11. プロパティ値を入力し終わったら、「終了」をクリックします。

これで、エンタープライズ・サービス・ディスカバリー・ウィザードで作成される SCA コンポーネントの参照バインディングを、WebSphere Integration Developer を使用して生成できるようになりました。

関連タスク

49 ページの『ロギング可能化』

WebSphere Adapter for Flat Files は、イベント処理の状況を判別するために表示できるログ・ファイルを維持しています。ログ・ファイル、および各ログ・エントリの日付、時刻、イベントによって、アダプターに関連するすべてのイベントやエラーを追跡します。アダプターはエラーや警告の状態の検出時にエラー・メッセージを記録するため、ログ・ファイルは問題のトラブルシューティングを開始するための有効な情報源となります。

51 ページの『トレースの使用可能化』

トレースによって、アダプター・ログ・ファイルに収集するエラーや警告のレベルが決まります。トレース・レベルを定義することによって、アダプターの処理に関するメッセージをたどって調べることができます。

39 ページの『プロパティの構成』

WebSphere Process Server 管理コンソールを使用して、アダプターのプロパティを構成することができます。

関連資料

41 ページの『J2C 接続ファクトリー・プロパティ』

このトピックでは、ターゲットのエンタープライズ情報システム (EIS) インスタンスの構成に使用する J2C 接続ファクトリーとプロパティについて説明します。これらのプロパティは、Outbound 処理に影響を与え、J2EE Connector Architecture 仕様の ManagedConnectionFactory インターフェースに対応します。

46 ページの『リソース・アダプター・プロパティ』

アダプターを初めて構成するとき (および後で WebSphere Process Server の管理コンソールを介して構成する場合)、エンタープライズ・サービス・ディスカバリー・ウィザードを使用すると、リソース・アダプター・プロパティを構成できます。このカテゴリのプロパティには、ロギングとトレースのプロパティおよびアダプター固有のプロパティがあります。

参照バインディングの生成 (Outbound 操作の場合)

サービス成果物を生成した後に、WebSphere Integration Developer を使用して、オプションで参照バインディングを生成することができます。

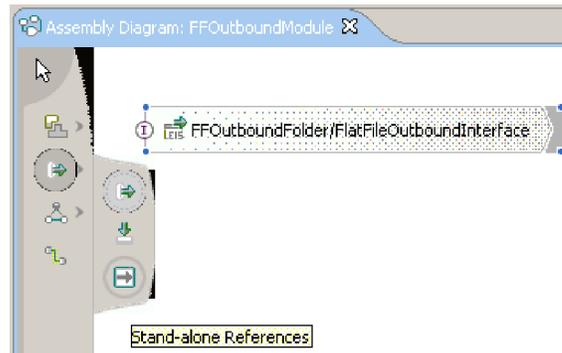
前提条件: ワークスペース上にアダプター・プロジェクトを作成し、構成しておく必要があります。アダプター・プロジェクトを作成した後、参照バインディングを生成して、サービス・コンポーネントにバインドする必要があります。

参照バインディングは、WebSphere Business Integration のほかの SCA コンポーネントがアダプターへのアクセス時に使用します。アダプターをほかのサーバー・プロセスにリンクするために、プロジェクト・モジュールからアダプターへの参照を作成します。

1. WebSphere Integration Developer ウィンドウで、Business Integration パースペクティブに切り替えるために、メニュー・バーから「ウィンドウ」→「パースペ

クティブを開く」 → 「その他 (Other)」 を選択します。すべてのパースペクティブが表示されます。 WebSphere Integration Developer で、Business Integration パースペクティブを選択します。

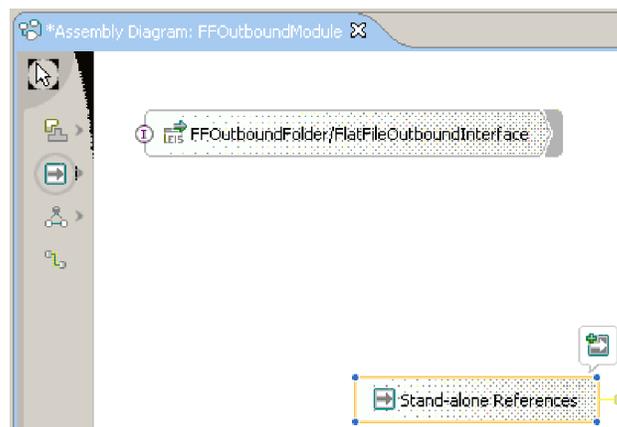
2. WebSphere Integration Developer の Business Integration パースペクティブで、モジュールを右クリックし、「アプリケーションから開く」 → 「アセンブリー・エディター」 を選択します。「アセンブリー・ダイアグラム (Assembly Diagram)」 ウィンドウが表示され、モジュールのインポート・コンポーネントがビューに示されます。
3. 独立の参照を作成するには、「アセンブリー・ダイアグラム (Assembly Diagram)」 ウィンドウの左側のフレーム (縦枠) の一番上にあるアイコンをクリックします。



「アセンブリー・ダイアグラム (Assembly Diagram)」 ウィンドウ

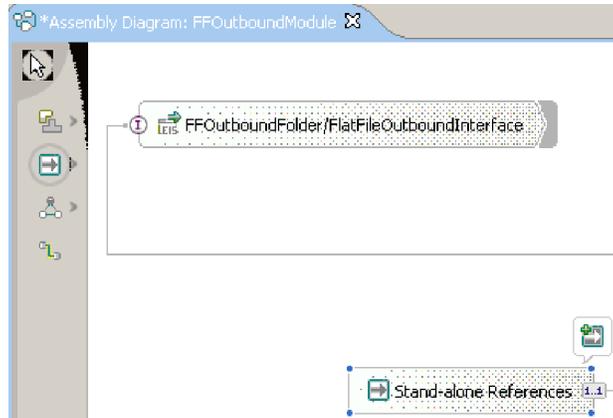
新しいアイコン・メニューが表示されます。上の図は、新しいアイコン・メニューを示しています。

4. 独立の参照を作成します。以下の図に示すように、「独立の参照 (Stand-alone Reference)」 アイコンを「アセンブリー・ダイアグラム (Assembly Diagram)」 ウィンドウへドラッグします。



「アセンブリー・ダイアグラム (Assembly Diagram)」 ウィンドウ

5. 独立の参照をインポート・モジュールに接続します。下の図に示すように、モジュールのインポート・コンポーネントをクリックして、新しいコンポーネントへドラッグします。これにより、インポート・コンポーネントから新しい独立の参照へのワイヤーが描画されます。



「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウ

「ワイヤーの追加 (Add Wire)」ダイアログ・ボックスが表示されます。

6. 「OK」をクリックします。「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウに、新しい独立の参照コンポーネントが、モジュールのインポート・コンポーネントにワイヤーで接続された状態で表示されます。
7. 「動的 Web プロジェクト」の下にある「Web プロジェクト」を右クリックして、「プロパティ」を選択します。Web プロジェクトには、サービス・クライアントが含まれています。
8. プロジェクトおよび Flat Files の RAR ファイルを選択してから、デフォルトの出力フォルダーを選択します。
9. 依存関係エディターを使用して、Web プロジェクトを従属する J2EE プロジェクトとして追加します。
 - a. モジュールを右クリックし、「依存関係エディターを開く (Open Dependency Editor)」を選択して、依存関係エディターを開きます。
 - b. 「J2EE プロジェクトの選択 (J2EE Project Selection)」ウィンドウで、プロジェクトを選択し、「OK」をクリックします。
10. 「ファイル」 → 「保管」をクリックして、ファイルを保管します。

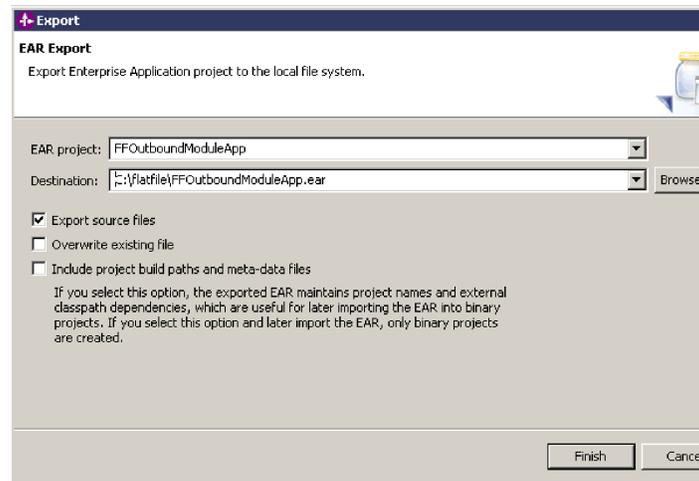
これで、EAR ファイルをエクスポートする準備ができました。

EAR ファイルのエクスポート (Outbound 操作の場合)

プロジェクトを実行するためには、WebSphere Integration Developer を使用して、プロジェクトを EAR ファイルにエクスポートする必要があります。

1. WebSphere Integration Developer ウィンドウで、J2EE パースペクティブに切り替えるために、メニュー・バーから「ウィンドウ」 → 「パースペクティブを開く」 → 「その他 (Other)」を選択します。すべてのパースペクティブが表示されます。WebSphere Integration Developer で、J2EE パースペクティブを選択します。
2. WebSphere Integration Developer ツールの J2EE パースペクティブ・ウィンドウで、モジュールを右クリックし、ポップアップ・メニューから「エクスポート」を選択します。「エクスポート」ウィンドウが表示されます。
3. 「エクスポート」 - 「選択」ウィンドウから、EAR ファイルを選択します。「エクスポート」 - 「EAR エクスポート」ウィンドウが表示されます。

- 以下の図に示すように、「EAR エクスポート」ウィンドウで、EAR プロジェクトと宛先ディレクトリー (プロジェクトのエクスポート先となる EAR ファイル名を含むディレクトリー) を選択します。



「EAR エクスポート」ウィンドウ

- 「終了」をクリックします。

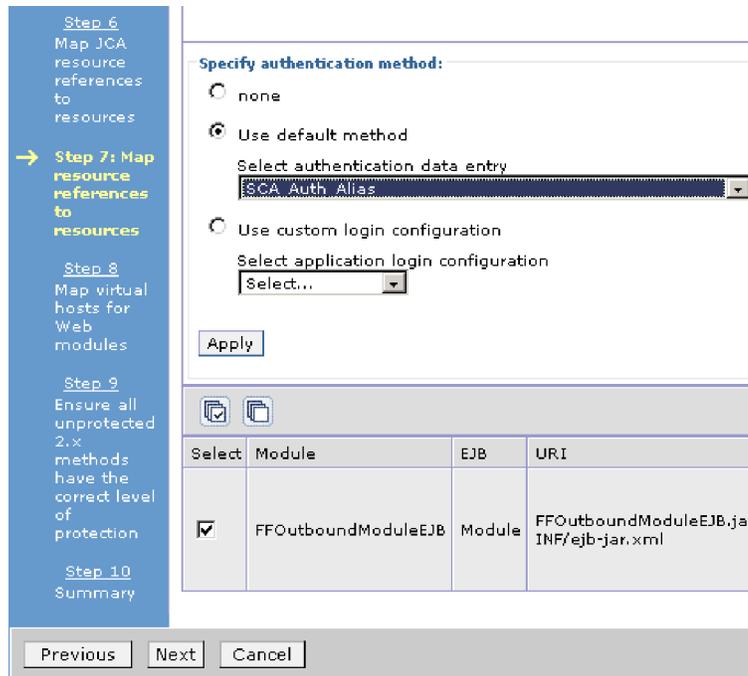
これで、プロジェクトの Enterprise Application Archive (EAR) ファイルへのエクスポートが終了し、アプリケーションをインストールする準備ができました。

アプリケーションのインストール (Outbound 操作の場合)

アプリケーション・プロジェクト・モジュールのインストールは、配置プロセスの最後のステップです。

アプリケーションをインストールして実行すると、プロジェクト・モジュールに組み込まれているアダプターが、インストールしたアプリケーションの一部として稼働します。

- WebSphere Process Server 管理コンソールで、「アプリケーション (Applications)」 → 「新規アプリケーションのインストール (Install New Applications)」をクリックします。
- エンタープライズ・アプリケーションのリストから EAR ファイルを選択します。
- 新しいアプリケーションへのパスの下に、EAR ファイルのパスを指定して、「次へ」をクリックします。
- 「ステップ 7: リソースへのリソース参照のマッピング (Step 7: Map resource reference to resources)」が表示されるまで「次へ」のクリックを続けます。
- 以下の図に示すように、「認証方式を指定する: (Specify authentication method:)」の下で以下の操作を行います。
 - ドロップダウン・メニューから、「SCA 認証の別名 (SCA Auth Alias)」を選択します。
 - モジュールのチェック・ボックスをクリックします。
 - 「適用」をクリックします。



WebSphere Process Server 管理コンソール

- 右側にスクロールして、JNDI 名が EJB プロジェクトの .xmi ファイルにある JNDI 名と一致することを確認します。

JNDI name	Login configuration
FFOutboundModule/FFOutb	Resource authorization: Container Authentication method: DefaultPrincipalMapping SCA_Auth_Alias

WebSphere Process Server 管理コンソール

- 「次へ」をクリックします。すべてのインストール・オプションの要約が表示されます。すべてのオプションが意図するとおりに選択されていることを確認して、「終了」をクリックします。

[Step 1](#) Select installation options

[Step 2](#) Map modules to servers

[Step 3](#) Provide options to perform the EJB Deploy

[Step 4](#) Provide listener bindings For message-driven beans

[Step 5](#) Provide JNDI Names for Beans

[Step 6](#) Map JCA resource references to resources

[Step 7](#) Map resource references to resources

[Step 8](#) Map virtual hosts for Web modules

[Step 9](#) Ensure all unprotected 2.x methods have the correct level of protection

→ Step 10: Summary

Summary

Summary of installation options

Options	Values
Use Binary Configuration	No
Deploy EJB option - Class path	
Create MBeans for resources	Yes
Cell/Node/Server	Click here
Reload interval in seconds	
Enable class reloading	No
Deploy EJB option - Database type	DB2UDB_V72
Deploy EJB option - Database schema	
Process embedded configuration	Yes
Application name	FFOutboundModuleApp
Deploy EJB option - RMIC	
Validate Input off/warn/fail	warn
Directory to install application	
Distribute application	Yes
Deploy Web services	No
Pre-compile JSP	No
Deploy enterprise beans	Yes

Previous
Finish
Cancel

WebSphere Process Server 管理コンソール

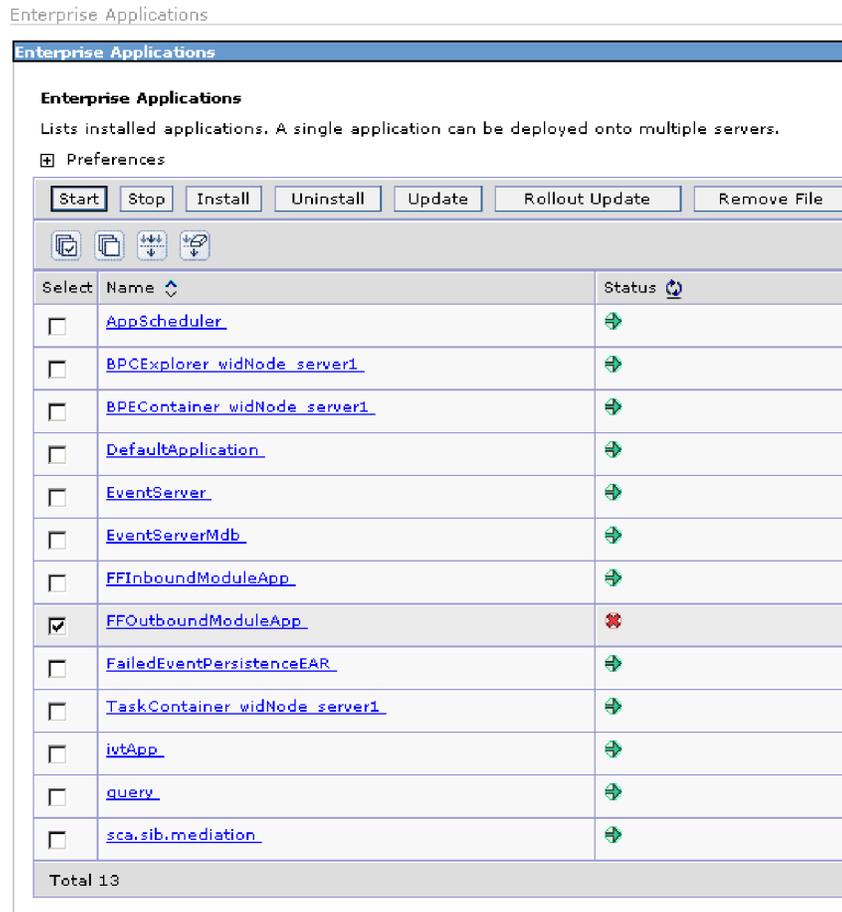
8. インストール・メッセージのリストが表示されます。リストの最後に、アプリケーションが正常にインストールされたことを示すメッセージが含まれていることを確認します。
9. インストール・メッセージのリストの最後に表示されている「**マスター構成に保管 (Save to Master Configuration)**」リンクをクリックします。「エンタープライズ・アプリケーション」ウィンドウが表示されます。
10. 「保管」をクリックして、アプリケーションを保管します。この時点で、アプリケーションが配置され、配置されたアプリケーション用の「エンタープライズ・アプリケーション」ウィンドウが表示されます。
11. 必要であれば、J2C アクティブ化仕様 (ActivationSpecification) プロパティを編集します。

これで、アプリケーションが配置され、正しく構成されました。次のステップは、アプリケーションの実行です。

アプリケーションの実行 (Outbound 操作の場合)

アプリケーションの配置が終了したら、このアプリケーションを実行することができます。アダプターはアプリケーション・プロジェクトに組み込まれているため、アプリケーションを実行すると、アダプターが起動され、実行を開始します。

1. WebSphere Process Server 管理コンソールで、「アプリケーション (Applications)」 → 「エンタープライズ・アプリケーション」をクリックします。
2. 以下の図に示すように、アプリケーションのチェック・ボックスを選択して、「開始」ボタンをクリックします。アプリケーションが実行を開始します。



WebSphere Process Server 管理コンソール

アダプターの構成

このトピックでは、アダプターを構成する作業について概説します。この作業は、構成プロパティを設定することによって実行します。

プロパティの構成

WebSphere Process Server 管理コンソールを使用して、アダプターのプロパティを構成することができます。

アダプターには、WebSphere Process Server 管理コンソールを使用して構成するプロパティのカテゴリが 4 つあります。

- J2C 接続ファクトリー・プロパティ (ManagedConnectionFactory インターフェースに対応)

- J2C アクティブ化仕様プロパティ (ActivationSpecification インターフェースに対応)
- カスタム・プロパティ (「デフォルト構成プロパティ」とも呼ぶ)
- リソース・アダプター・プロパティ

注: サービス成果物の生成中に、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Flat Files アダプターのプロパティを構成することもできます。

WebSphere Process Server 管理コンソールを使用してプロパティを構成するには、以下のステップを実行します。

1. コンソールを始動します。
2. 「リソース」の下で、「リソース・アダプター」を選択します。
3. 「リソース・アダプター」の下で、「WebSphere Adapter for Flat Files」を選択します。「一般プロパティ (General Properties)」ページが表示されます。
4. 「追加プロパティ (Additional Properties)」の下で、変更するプロパティのカテゴリを、以下から 1 つ選択します。
 - J2C 接続ファクトリー。ターゲット・エンタープライズ情報システムのインスタンスを構成する際に使用される ManagedConnectionSpec プロパティを構成する。
 - J2C アクティブ化仕様。メッセージ・エンドポイント・プロパティを構成する。
 - カスタム・プロパティ。すべての WebSphere アダプターによって共有されるデフォルト構成プロパティを構成する。
5. 以下のいずれかの操作を実行します。
 - J2C 接続ファクトリーを選択した場合は、構成する J2C 接続ファクトリーの名前を選択してから、構成する J2C 接続ファクトリーのプロパティに応じて、接続プール・プロパティ、拡張接続ファクトリー・プロパティ、またはカスタム・プロパティを選択します。カスタム・プロパティは、WebSphere Adapter for Flat Files に固有の J2C 接続ファクトリー・プロパティです。接続プールと拡張接続ファクトリーのプロパティは、独自のアダプターを開発する場合に構成するプロパティです。
 - J2C アクティブ化仕様を選択した場合は、構成する J2C アクティブ化仕様の名前を選択します。次に、構成するメッセージ・エンドポイント・プロパティの名前を選択し、必要な値に設定します。
 - カスタム・プロパティを選択した場合は、「カスタム・プロパティ」ページが表示されます。構成するデフォルト構成プロパティの名前を選択し、必要な値に設定します。

関連タスク

16 ページの『サービス成果物の生成 (Inbound 操作の場合)』

Inbound 操作の場合のサービス成果物の生成は、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

29 ページの『サービスの構成 (Outbound 操作の場合)』
 構成プロセスは、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

WebSphere Adapter for Flat Files の構成プロパティー

このトピックでは、WebSphere Adapter for Flat Files の構成プロパティーのリストを示します。

WebSphere Adapter for Flat Files には、複数のカテゴリーの構成プロパティーがあります。

以下の表に、アダプター構成プロパティーのカテゴリーを示します。

構成プロパティーのカテゴリー	説明
J2C 接続ファクトリー	Outbound 処理を構成するときに使用する
J2C アクティブ化仕様	Inbound 処理を構成するときに使用する
リソース・アダプター・プロパティー	ロギングやトレースなどの機能を構成するときに使用する
エンタープライズ・サービス・ディスカバリー接続プロパティー	初期のアダプター配置時に、Inbound 処理または Outbound 処理と双方向の使用可能化を構成するために使用する

J2C 接続ファクトリー・プロパティー

このトピックでは、ターゲットのエンタープライズ情報システム (EIS) インスタンスの構成に使用する J2C 接続ファクトリーとプロパティーについて説明します。これらのプロパティーは、Outbound 処理に影響を与え、J2EE Connector Architecture 仕様の ManagedConnectionFactory インターフェースに対応します。

J2C 接続ファクトリーは、接続プールを管理します。リソース・アダプターを介して Outbound 操作の構成情報を提供します。

「ManagedConnectionFactory 構成プロパティー」の表に、J2C 接続ファクトリーに
 関係する構成プロパティーの定義を示します。

ManagedConnectionFactory 構成プロパティー

プロパティー名	データ型	国際化されるかどうか	双方向変換サポート (はい/いいえ)	説明
アウトプット・ディレクトリー	String	はい	はい	Outbound 操作の出力ディレクトリー。
アウトプット・ファイル名	String	はい	はい	Outbound 操作の出力ファイル名。

関連タスク

29 ページの『サービスの構成 (Outbound 操作の場合)』
 構成プロセスは、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

62 ページの『シナリオ 2 の Outbound 操作の配置および構成』
 サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス

ス・ディスカバリー・ウィザードを使用して、Outbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

J2C アクティブ化仕様プロパティ

このトピックでは、(メッセージ・エンドポイント・プロパティとしても知られる) J2C アクティブ化仕様プロパティについて説明します。この仕様は、J2EE Connector Architecture Specification の ActivationSpecification インターフェースに対応します。

ActivationSpecification プロパティには、メッセージ・エンドポイントの Inbound イベント処理構成情報が保持されています。

「ActivationSpecification プロパティ」の表に、ActivationSpec インターフェースに関係する構成プロパティの定義を示します。

ActivationSpecification プロパティ

プロパティ名	データ型	国際化されるかどうか	双方向変換サポート (はい/いいえ)	説明
アーカイブ・ディレクトリー	String	はい	はい	処理されたイベント・ファイルがアダプターによってアーカイブされるディレクトリー。 必要な値である: はい (アーカイビング・プロセス = true の場合)
アーカイビング・プロセス	Boolean	はい	いいえ	処理されたイベントをアダプターによってアーカイブするかどうかを決定する boolean プロパティ。 デフォルト値: True 必要な値である: はい
イベント・ディレクトリー	String	はい	はい	バックエンド EIS によってイベント・ファイルが格納されるディレクトリー。 必要な値である: はい
イベント・ファイル・マスク	String	はい	はい	イベント・ファイルのフィルターを指定する。ファイル・フィルターは、ワイルドカード * を基にしたパターンのみと一致する。 デフォルト値: *.* 必要な値である: はい

プロパティ名	データ型	国際化されるかどうか	双方向変換サポート (はい/いいえ)	説明
FF データベース名	String	はい	はい	FFEventTable が存在する論理データベースの名前。このデータベースは、WebSphere Process Server が内蔵している Cloudscape DBMS 内に置かれている。 デフォルト値: FFDB 必要な値である: はい
FF イベント・テーブル名	String	はい	はい	イベント状況値を追跡するためにアダプターが使用するテーブルの名前。 デフォルト値: FFLOG 必要な値である: はい
ファイル・チャンク・サイズ	Integer	はい	いいえ	ファイルを複数のチャンクに分割する必要がある場合、各チャンクのサイズをバイト単位で指定する。ファイル分割しきい値プロパティの値が -1 の場合、このプロパティは非アクティブのままである。 デフォルト値: 8000
ファイル分割しきい値	Integer	はい	いいえ	ファイルを分割する際のファイル・サイズしきい値をバイト単位で指定する。このファイル・サイズを超えるすべてのファイルは、複数のチャンクに分割される。属性が空白の場合、分割は使用できない。

プロパティ名	データ型	国際化されるかどうか	双方向変換サポート (はい/いいえ)	説明
ソート・イベント・ファイル	String	はい	いいえ	<p>ポーリングされるイベント・ファイルのソート順を決定する。サポートされる値は、Filename (ファイル名で昇順にソート)、Timestamp (最終変更日時タイムスタンプで昇順にソート)、<空白> (ソートなし) など。グローバリゼーションをサポートする目的で、ファイル名のソート機能はシステム・ロケールに応じて提供される。ICU4J パッケージを使用してロケールおよびロケールに対応した規則を追跡する。</p> <p>デフォルト値: <空白> (= ソートなし)</p>

関連タスク

16 ページの『サービス成果物の生成 (Inbound 操作の場合)』

Inbound 操作の場合のサービス成果物の生成は、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

58 ページの『シナリオ 2 の Inbound 操作の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Inbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

カスタム・プロパティ

このトピックでは、WebSphere Adapter for Flat Files のデフォルトの構成プロパティについて説明します。

「カスタム・プロパティ」の表では、デフォルトの構成プロパティについて説明します。

カスタム・プロパティ

プロパティ名	データ型	国際化されるかどうか	双方向変換サポート (はい/いいえ)	説明
自動作成 EDT	Boolean	はい	いいえ	値を true に設定するとメモリー内のテーブルが作成され、イベント管理フレームワークはこのテーブルを使用してイベント配信を追跡する。 必要である: はい デフォルト値: True
送達タイプ	String	はい	いいえ	エンドポイントへの配信のタイプを ORDERED または RANDOM に指定する。 必要である: はい デフォルト値: ORDERED
EDT データベース名	String	はい	はい	イベント管理フレームワークが使用するデータベースの名前。
EDT ドライバー名	String	はい	いいえ	イベント・データ・テーブル・データベースのドライバーの名前。
EDT テーブル名	String	はい	はい	イベント管理フレームワークが使用するイベント・データ・テーブル・データベース上のテーブルの名前。
EDT ユーザー名	String	はい	はい	イベント・データ・テーブル・データベースにアクセスするためのユーザー ID 証明書。このプロパティは双方向言語の場合に使用可能。
EDT ユーザー・パスワード	String	はい	はい	イベント・データ・テーブル・データベースにアクセスするためのパスワード信用証明情報。
ポーリング間隔	Integer	はい		イベント・ディレクトリー内のイベントをポーリングする場合のミリ秒単位の間隔。 デフォルト値: 2000
ポーリング数量	Integer	はい	いいえ	各ポーリング時に処理するためにピックアップされるイベント・ファイルの数。 デフォルト値: 10

関連タスク

16 ページの『サービス成果物の生成 (Inbound 操作の場合)』

Inbound 操作の場合のサービス成果物の生成は、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

58 ページの『シナリオ 2 の Inbound 操作用の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Inbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

エンタープライズ・サービス・ディスカバリー接続プロパティ

接続プロパティには、エンタープライズ・サービス・ディスカバリー・ウィザードを介して初めてアダプターを配置したときに構成するカテゴリのプロパティがあります。これらのプロパティは、メタデータ・ディスカバリーの実行と双方向構成プロパティに必要な Outbound 接続プロパティと Inbound 接続プロパティです。

Inbound 接続の記述は、エクスポート・ファイルに書き込まれます。

Inbound 接続

プロパティ	説明	ファイル内の要素
ActivationSpec	Flat File リソース・アダプターのアクティブ化仕様を表すクラスのインスタンス。	接続エレメントのタイプ
Resource Adapter	Flat File リソース・アダプターを表すクラスのインスタンス。	resourceAdapter エlement

Outbound 接続

プロパティ	説明	ファイル内の要素
Managed Connection Factory	Flat File リソース・アダプターのアクティブ化仕様を表すクラスのインスタンス。	接続エレメントのタイプ
Resource Adapter	Flat File リソース・アダプターを表すクラスのインスタンス。	resourceAdapter エlement

リソース・アダプター・プロパティ

アダプターを初めて構成するとき (および後で WebSphere Process Server の管理コンソールを介して構成する場合)、エンタープライズ・サービス・ディスカバリー・ウィザードを使用すると、リソース・アダプター・プロパティを構成できます。このカテゴリのプロパティには、ロギングとトレースのプロパティおよびアダプター固有のプロパティがあります。

ロギングとトレースのプロパティ

以下の表に、Flat Files アダプターのロギングとトレースのプロパティを示します。

プロパティ名	データ型	国際化されるかどうか	説明
ログ・ファイル・サイズ	Integer	はい	ログ・ファイルのサイズ (KB)。値を指定しなかった場合、ファイルの最大サイズは設定されません。
ログ・ファイル名	String	はい	ログ・ファイルの絶対パス。
ログ・ファイル数	Integer	はい	使用するログ・ファイルの数。ログ・ファイルが最大サイズに到達すると、別のログ・ファイルの使用が開始されます。値を指定しなかった場合、値は 1 に設定されます。
トレース・ファイルのサイズ (TraceFileSizeMax)	Integer	はい	トレース・ファイルのサイズ (KB)。値を指定しなかった場合、ファイルの最大サイズは設定されません。
トレース・ファイル名	String	はい	トレース・ファイルの絶対パス。
トレース・ファイル数	Integer	はい	使用するトレース・ファイルの数。トレース・ファイルが最大サイズに到達すると、別のトレース・ファイルの使用が開始されます。値を指定しなかった場合、値は 1 に設定されます。

関連タスク

16 ページの『サービス成果物の生成 (Inbound 操作の場合)』

Inbound 操作の場合のサービス成果物の生成は、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

29 ページの『サービスの構成 (Outbound 操作の場合)』

構成プロセスは、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

58 ページの『シナリオ 2 の Inbound 操作の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Inbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

62 ページの『シナリオ 2 の Outbound 操作の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Outbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

トラブルシューティングおよびサポート

トラブルシューティングおよびサポート資料は、WebSphere Adapter for Flat Files で発生した問題やエラーの原因を見つけ出すのに役立ちます。問題の報告と解決のために、サポートおよび連絡先情報が提供されています。

以下のトピックでは、アダプターのトラブルシューティングとサポートについての情報を提供します。

IBM ソフトウェア・サポートへの連絡

資料およびお客様サポートについては営業担当員にお問い合わせください。

IBM ソフトウェア・サポートに連絡するには、会社に現在有効な IBM ソフトウェア保守契約があり、担当者に IBM への問題の送信が許可されている必要があります。必要なソフトウェア保守契約のタイプは、所有している製品のタイプによって異なります。

- IBM 分散ソフトウェア・プロダクト (Tivoli®、 Lotus®、 Rational® 製品、および Windows または UNIX オペレーティング・システムで動作する DB2® および WebSphere 製品など) の場合は、次のいずれかの方法でパスポート・アドバンテージ®に登録します。
 - **オンライン:** 米国のパスポート・アドバンテージ Web ページに移動し、「How to Enroll」をクリックします。
 - **電話:** 国別の連絡先電話番号は、 Web 上にある米国の IBM ソフトウェア・サポート・ハンドブックのお問い合わせ窓口ページへ移動し、該当する地域名をクリックします。
- IBM eServer™ ソフトウェア製品 (zSeries®、 pSeries®、 および iSeries™ 環境で稼働する DB2 および WebSphere 製品を含むが、これらに限定されない) については、IBM 営業担当員または IBM ビジネス・パートナーを通してソフトウェア・メンテナンス契約を購入できます。eServer ソフトウェア・プロダクトのサポートについての詳細は、米国の IBM テクニカル・サポート・アドバンテージ Web ページを参照してください。

必要なソフトウェア保守契約のタイプがわからない場合は、米国にお住まいの場合は 1-800-IBMSERV (1-800-426-7378) に電話するか、その他の国にお住まいの場合は Web 上にある米国の IBM ソフトウェア・サポート・ハンドブックのお問い合わせ窓口ページへ移動して、該当する地域名をクリックすると、その地域でサポートを提供している担当者の電話番号が表示されます。

IBM ソフトウェア・サポートに連絡する場合は、以下の手順に従ってください。

- 問題によるビジネス・インパクトの判別。
 - 問題の説明と背景情報の収集。
 - 問題を IBM ソフトウェア・サポートに送信します。
1. 問題のビジネス・インパクトを判別します。 IBM に問題を報告すると、重大度レベルを尋ねられます。そのため、報告する問題のビジネス・インパクトを理解し、評価する必要があります。次の基準を使用してください。

重大度	説明
重大度 1	重大なビジネス・インパクト。プログラムを使用できず、業務に重大な影響が及んでいる。この状態は、即時のソリューションを必要とする。
重大度 2	大きなビジネス・インパクト。プログラムは使用できるが、非常に制限されている。
重大度 3	いくらかのビジネス・インパクト。プログラムは使用でき、あまり重要でない機能（業務にとって重要でない）が使用できない。
重大度 4	最小のビジネス・インパクト。問題が業務に与える影響はほとんどなく、問題に対する回避策がとられている。

2. 問題の説明と背景情報の収集。 IBM に問題を説明する際は、できるだけ具体的に説明してください。 IBM ソフトウェア・サポート・スペシャリストが問題の解決を効果的に支援できるよう、関連するすべての背景情報を伝えてください。時間を節約するため、以下の質問に対する回答を用意しておいてください。
 - 問題が発生したときに実行していたソフトウェアのバージョンは何ですか。
 - 問題の徴候に関連するログ、トレース、メッセージはありますか。 IBM ソフトウェア・サポートは以下の情報を尋ねることがあります。
 - 問題は再現可能ですか。再現できる場合、どのステップを実行すると問題が発生しますか。
 - システムに変更を加えましたか (例えば、ハードウェア、オペレーティング・システム、ネットワーキング・ソフトウェアなど)。
 - この問題に対する解決方法を現在実行していますか。その場合は、問題を報告する際にそれを説明できるようにしておいてください。
3. 問題を IBM ソフトウェア・サポートに送信します。 問題を送信する 2 つの方法を次に示します。
 - **オンライン:** 米国の IBM ソフトウェア・サポート・サイトの「Submit and track problems」ページへ移動します。適切な問題送信ツールに情報を入力します。
 - **電話:** 国別の連絡先電話番号は、 Web 上にある米国の IBM ソフトウェア・サポート・ハンドブックのお問い合わせ窓口ページへ移動し、該当する地域名をクリックします。

送信した問題がソフトウェアの欠陥や資料の欠落または誤りである場合は、 IBM ソフトウェア・サポートによって APAR (プログラム診断依頼書) が作成されます。 APAR では、問題が詳細に記述されます。

可能であれば、 IBM ソフトウェア・サポートは、 APAR が解決されフィックスが送信されるまで、お客様が実行できる解決方法を提供します。 IBM は、同じ問題が発生した別のユーザーが同じ解決策を利用できるように、 IBM 製品サポート Web ページで常に、解決された APAR を公開しています。

ロギング可能化

WebSphere Adapter for Flat Files は、イベント処理の状況を判別するために表示できるログ・ファイルを維持しています。ログ・ファイル、および各ログ・エントリの日付、時刻、イベントによって、アダプターに関連するすべてのイベントやエ

ラーを追跡します。アダプターはエラーや警告の状態の検出時にエラー・メッセージを記録するため、ログ・ファイルは問題のトラブルシューティングを開始するための有効な情報源となります。

Flat Files Adapter のロギングは、WebSphere Process Server 管理コンソールから使用可能にします。ロギングを使用可能にするには、以下のステップを実行します。

1. WebSphere Process Server 管理コンソールを開始します。
2. 管理コンソールから、「トラブルシューティング」 → 「ログとトレース (Logs and Trace)」を選択します。
3. ご使用のサーバーに対応するサーバー・リンクをクリックします。
4. 「ログの詳細レベルの変更 (Change Log Detail levels)」をクリックします。
5. 「コンポーネント」をクリックして、個々のコンポーネントのログの詳細レベルを指定します。あるいは、「グループ (Groups)」をクリックして、あらかじめ定義されたコンポーネント・グループのログの詳細を指定します。
6. 必要なロギング・レベルを選択します。「ロギング・レベル」の表に、WebSphere Process Server 管理コンソールを使用して設定できる各種のロギング・レベルを示しています。

注: 詳細レベルより下のレベルのログ・イベントを表示するためには、診断トレース・サービスを使用可能に設定する必要があります。SystemOut ログ、IBM 保守ログ (使用可能な場合)、診断トレース・サービス (使用可能な場合) では、詳細レベルまたはそれより上のレベルのログ・イベントを表示することができます。

ロギング・レベル

レベル	標識	説明
監査	A	サーバー状態またはリソースに影響を与える重大なイベント。
構成	C	構成の変更または状況。
詳細	D	サブタスクの進行状況の詳細を示す一般情報。
致命的	F	タスクを継続不能。コンポーネントが動作不能。
情報	I	タスクの全体的な進行状況の概要を示す一般情報。
重大	E	タスクを継続不能。コンポーネントは動作可能。これには、差し迫った致命的エラーを示す (すなわち、リソースが枯渇寸前であることを強く示す) 状況も含まれる。
警告	W	潜在的なエラー、または差し迫ったエラー。これには、例えばリソース・リークの可能性など、進行性の障害を示す状況も含まれる。

7. 「適用」をクリックして、変更内容を保管します。

関連タスク

16 ページの『サービス成果物の生成 (Inbound 操作の場合)』

Inbound 操作の場合のサービス成果物の生成は、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

29 ページの『サービスの構成 (Outbound 操作の場合)』

構成プロセスは、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

58 ページの『シナリオ 2 の Inbound 操作の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Inbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

62 ページの『シナリオ 2 の Outbound 操作の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Outbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

トレースの使用可能化

トレースによって、アダプター・ログ・ファイルに収集するエラーや警告のレベルが決まります。トレース・レベルを定義することによって、アダプターの処理に関するメッセージをたどって調べることができます。

トレース・レベルは、WebSphere Process Server 管理コンソールで構成できます。トレース・レベルを設定するには、以下のステップを実行します。

1. WebSphere Process Server 管理コンソールを開始します。
2. 管理コンソールから、「トラブルシューティング」 → 「ログとトレース (Logs and Trace)」を選択します。
3. ご使用のサーバーに対応するサーバー・リンクをクリックします。
4. 「ログの詳細レベルの変更 (Change Log Detail levels)」をクリックします。
5. 「コンポーネント」をクリックして、個々のコンポーネントのログの詳細レベルを指定します。あるいは、「グループ (Groups)」をクリックして、あらかじめ定義されたコンポーネント・グループのログの詳細を指定します。
6. 必要なトレース・レベルを選択します。「トレース・レベル」の表に、WebSphere Process Server 管理コンソールを使用して設定できる各種のトレース・レベルを示しています。

トレース・レベル

レベル	標識	説明
詳細	1	一般トレース。アダプターによる一般的な動作。エンタープライズ情報システムへの接続の確立、エンタープライズ情報システムでのイベントのビジネス・オブジェクトへの変換 (キー値のみ)、ビジネス・オブジェクトの処理 (キー値のみ) などを含む。
より詳細	2	アダプターによって実行されているロジックについてのよりきめ細かい情報を提供する詳細なトレース。エンタープライズ情報システムに対して行われるさまざまなアプリケーション固有情報の呼び出しやパラメーターまたは戻り値などを含む。
極めて詳細	3	最も詳細なレベル。メソッドの入力値/終了値/戻り値を含む。完全なビジネス・オブジェクト・ダンプも含まれる。このレベルでは、デバッグに必要な詳細な情報すべてが提供される。

7. 「適用」をクリックして、変更内容を保管します。

関連タスク

16 ページの『サービス成果物の生成 (Inbound 操作の場合)』

Inbound 操作の場合のサービス成果物の生成は、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

29 ページの『サービスの構成 (Outbound 操作の場合)』

構成プロセスは、WebSphere Integration Developer のエンタープライズ・サービス・ディスカバリー・ウィザードを使用して実行します。

58 ページの『シナリオ 2 の Inbound 操作の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Inbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

62 ページの『シナリオ 2 の Outbound 操作の配置および構成』

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Outbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

Common Event Infrastructure (CEI) の使用可能化

このトピックでは、アダプター用に Common Event Infrastructure (CEI) を使用可能に設定する方法について説明します。

IBM WebSphere Adapters Event Definitions ファイルを CEI カタログにパブリッシュした後で、これらのイベント定義を設定する必要があります。この方法については、WebSphere Process Server の Web サイト (<http://www.ibm.com/software/integration/wps>) にある CEI の資料を参照してください。

1. WebSphere 管理コンソールを開始します。
2. 「トラブルシューティング」 → 「ログとトレース (Log and Trace)」を選択してから、<サーバー名> を選択します。
3. 「一般プロパティ (General Properties)」の多数のオプションがあります。「ログの詳細レベルの変更 (Change Log Detail Level)」を選択して、JCA コンポーネントに対して **com.ibm.j2ca.*** を選択します。このセクションの下に、以下のアダプター・タイプごとのサブコンポーネントがあります。
 - com.ibm.j2ca.flatfile.* (WebSphere Adapter for Flat Files)
 - com.ibm.j2ca.jdbc.* (WebSphere Adapter for JDBC)
 - com.ibm.j2ca.peoplesoft.* (WebSphere Adapter for PeopleSoft)
 - com.ibm.j2ca.sap.* (WebSphere Adapter for SAP)
 - com.ibm.j2ca.siebel.* (WebSphere Adapter for Siebel)
4. ご使用のアダプターと一致するコンポーネントを選択します。各アダプター・コンポーネントが、ロギング用と CEI 用の 2 つのサブコンポーネントを持ちます。形式は次のとおりです。
 - *subcomponent name.log.adapter id*
 - *subcomponent name.cei.adapter id*例えば、com.ibm.j2ca.siebel.cei.<AdapterID1> となります。配置されたアダプターのインスタンスごとに、別々の ID が表示されます。
5. 使用可能に設定する CEI アダプター ID を選択します。
6. ドロップダウン・メニューで、以下の中から項目を選択することができます。
 - 「オフ (off)」 - CEI をオフにする
 - 「詳細 (fine)」 - CEI をオンにして、イベントの内容を空に設定する
 - 「より詳細 (finer)」 - CEI をオンにして、イベントの内容をダイジェストに設定する
 - 「極めて詳細 (finest)」 - CEI をオンにして、イベントの内容をフルに設定する
 - 「すべて (all)」 - 「極めて詳細 (finest)」と同じ

各イベントの内容のレベル (空、ダイジェスト、フル) の意味の詳細について、また、共通ベース・イベント・モデルおよび Common Event Infrastructure の使用方法について詳しくは、WebSphere Process Server Web サイト (<http://www.ibm.com/software/integration/wps>) にある資料を参照してください。

サンプル・アプリケーションの使用

アダプターには、アプリケーション・パッケージの配置方法やアダプターによるビジネス・オブジェクト処理の仕組みを示すサンプル・アプリケーションが提供されています。サンプル・アプリケーションには、アダプターを使用する対象者に1つずつ、計2つのシナリオが用意されています。アダプターを使用する対象者は、アプリケーションの統合担当者とデータの統合担当者の2種類のユーザーで構成されます。

提供されているそれぞれのサンプルに対して、用意されている2つのシナリオは次のとおりです。

サンプル・アプリケーションのシナリオ

シナリオ	説明	対象読者
シナリオ 1	<ul style="list-style-type: none">生成済み成果物を提供し、アダプターによるビジネス・オブジェクト処理の仕組みを示します。このシナリオでは、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して成果物を生成する必要はありません。アプリケーションのコンポーネントをアセンブルしてソリューションを作成することと、このソリューションのテストと配置を準備することを担当する対象者をターゲットにしています。	アプリケーションの統合担当者
シナリオ 2	<ul style="list-style-type: none">エンタープライズ・サービス・ディスカバリー・ウィザードを使用してアプリケーションのコンポーネントを検出し、アダプターが処理するビジネス・オブジェクトを作成する方法を示します。アプリケーションの統合担当者と同じ責任を持つだけでなく、アプリケーション開発者が一連のデータ・ソースにアクセスできるようにする責任もある対象者をターゲットにしています。	データの統合担当者

アプリケーション・パッケージの構造

サンプル・アプリケーションのファイルは、アダプターをインストールしたときにインストールされます。これらのファイルは、samples フォルダにインストールされているアーカイブ・ファイル内に圧縮されています。

シナリオ 1: 包括的なサンプル・ファイル・セット

アプリケーションの統合担当者をターゲットにしているシナリオ 1 の場合、サンプル・アプリケーション・パッケージには、必要なすべての成果物が収められているため、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して成果物を取り込む必要はありません。アダプターの実際の配置では、シナリオ 2 に示すように、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、これらの成果物をすべて生成し、アダプターを構成することになります。

シナリオ 1 のファイルは、¥adapter¥flatfile¥samples¥nonemdsamples.zip ファイル内に保存されています。

1. samples フォルダに移動して、nonemdsamples.zip を解凍します。これにより、FlatFileInboundModuleApp.ear および FlatFileOutboundModuleApp.ear が解凍されます。
2. FlatFileInboundModuleApp.ear には、以下のファイルが収められています。
 - FlatFileInboundModule.jar
 - FlatFileInboundModuleEJB.jar
 - FlatFileInboundModuleEJBClient.jar
 - FlatFileInboundModuleWeb.war
 - CWYFF_FlatFile.rar
 - application.xml
 - deployment.xml
3. FlatFileOutboundModuleApp.ear には、以下のファイルが収められています。
 - FlatFileOutboundModule.jar
 - FlatFileOutboundModuleEJB.jar
 - FlatFileOutboundModuleEJBClient.jar
 - FlatFileOutboundModuleWeb.war
 - CWYFF_FlatFile.rar
 - applicaiton.xml
 - deployment.xml
 - MyWebProject.war

シナリオ 2: エンタープライズ・サービス・ディスカバリー・ウィザードの要件

データの統合担当者をターゲットにしたシナリオ 2 の場合は、サンプル・アプリケーション・パッケージの提供する機能により、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して SCA 成果物を作成し、アダプターを構成できます。ただし、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して作成したファイルが本当に正しいかどうかと、アプリケーションを正常に実行

するための前提は何かを確認する方法として、エンタープライズ・サービス・ディスクバリー・ウィザードを使用して最終的に生成したすべてのファイルのコピーにアクセスできます。

シナリオ 2 に必要なすべてのファイルは、¥adapter¥flatfile¥samples¥emdsamples.zip に格納されています。エンタープライズ・サービス・ディスクバリー・ウィザードを使用して生成するファイルのコピーは、¥adapter¥flatfile¥samples¥nonemdsamples.zip に格納されています。

1. samples フォルダーに移動して、emdsamples.zip ファイルを解凍します。これにより、CWYFF_FlatFile.rar ファイルが解凍されます。
2. エンタープライズ・サービス・ディスクバリー・ウィザードにより、この RAR ファイルがインポートされ、指定した構成ごとにサービス成果物が生成されま

す。

ビジネス・オブジェクトの構造

Flat Files Resource Adapter のビジネス・オブジェクト定義には、Flat Files アダプターの操作と Inbound 操作を実行するために必要な情報が格納されています。

Flat Files アダプターには、Inbound 操作と Outbound 操作の両方に対応する同じビジネス・オブジェクト構造があります。このビジネス・オブジェクトのスキーマは、以下の属性で構成されます。

- directoryPath
- fileName
- inputBytes
- outputBytes
- outputString

ビジネス・オブジェクト構造を内包するビジネス・グラフは、FlatFileBG.xsd ファイルです。これは、複合タイプ FlatFile のエレメントで構成されます。

シナリオ 1 の Inbound 操作の配置および構成

サンプル・アプリケーションのシナリオ 1 では、アダプターの構成済みインスタンスと、すべての必要な SCA 成果物が提供されます。必要な作業は、WebSphere Integration Developer を使用して、EAR ファイルをご使用の WebSphere Integration Developer ワークスペースにインポートし、WebSphere Business Integration Component Test Tool を使用して構成プロセスを完了させることです。

1. このシナリオでは、既に構成済みのアダプターのインスタンスが提供されているため、必要な作業は WebSphere Integration Developer を使用して、プロジェクトに EAR ファイルをインポートすることだけです。WebSphere Integration Developer で、「ファイル」→「インポート」を選択してから、CWYFF_FlatFile.rar を選択します。
2. Business Integration パースペクティブに切り替えて、新しいモジュール FlatFileInboundModule を作成します。新規モジュールを作成するには、「Business Integration」パネルを右クリックし、ポップアップ・メニューから「新規」→「モジュール」を選択します。
3. FlatFileInboundModule.jar の内容を抽出します。

4. WebSphere Integration Developer で、 Business Integration パースペクティブ内の FlatFileModule を更新します。 FlatFileInboundModule を右クリックし、ポップアップ・メニューから「更新」を選択します。
5. CWYFF_FlatFile.rar を、従属する J2EE プロジェクトとして追加します。 Business Integration パースペクティブ内の FlatFileInboundModule を右クリックして、「依存関係エディターを開く (Open Dependency Editor)」を選択します。
6. ActivationSpecification プロパティを編集します。 WebSphere Integration Developer で、 FlatFileInboundModule の下の最初のエレメントをダブルクリックして、アセンブリー・エディターを開きます。
 - a. 「プロパティ」 → 「バインディング (Binding)」 → 「接続」 → 「ActivationSpec プロパティ (ActivationSpec Properties)」を選択します。
 - b. 固有のシステム要件に従って、ActivationSpec プロパティを編集します。
 - c. ActivationSpec プロパティの編集が完了したら、「保管」をクリックします。
7. Business Integration パースペクティブで、 FlatFileInboundModule を右クリックします。
8. ポップアップ・メニューから、「テスト」 → 「モジュールのテスト (Test Module)」を選択します。「イベント」ウィンドウが表示されます。
9. 「イベント」ウィンドウで、サンプル・イベント・ファイルをイベント・ディレクトリーに格納します。
 - a. イベント・ファイルの拡張子が「.in」であることを確認します。
 - b. WebSphere Process Server を起動し、Inbound アプリケーションをインストールする場合は、「継続 (Continue)」をクリックします。

シナリオ 1 の Outbound 操作用の配置および構成

サンプル・アプリケーションのシナリオ 1 では、アダプターの構成済みインスタンスと、すべての必要な SCA 成果物が提供されます。必要な作業は、WebSphere Integration Developer を使用して、EAR ファイルをご使用の WebSphere Integration Developer ワークスペースにインポートし、WebSphere Business Integration Component Test Tool を使用して構成プロセスを完了させることです。

1. このシナリオでは、既に構成済みのアダプターのインスタンスが提供されているため、必要な作業は WebSphere Integration Developer を使用して、プロジェクトに EAR ファイルをインポートすることだけです。 WebSphere Integration Developer で、「ファイル」 → 「インポート」を選択してから、CWYFF_FlatFile.rar を選択します。
2. Business Integration パースペクティブに切り替えて、新しいモジュール FlatFileOutboundModule を作成します。新規モジュールを作成するには、「Business Integration」パネルを右クリックし、ポップアップ・メニューから「新規」 → 「モジュール」を選択します。
3. FlatFileOutboundModule.jar の内容を抽出します。
4. WebSphere Integration Developer で、 Business Integration パースペクティブ内の FlatFileModule を更新します。 FlatFileOutboundModule を右クリックし、ポップアップ・メニューから「更新」を選択します。

5. CWYFF_FlatFile.rar を、従属する J2EE プロジェクトとして追加します。
Business Integration パースペクティブ内の FlatFileInboundModule を右クリックして、「依存関係エディターを開く (Open Dependency Editor)」を選択します。
6. ManagedConnectionFactory プロパティを編集します。 WebSphere Integration Developer で、 FlatFileOutboundModule の下の最初のエレメントをダブルクリックして、アセンブリー・エディターを開きます。
 - a. 「プロパティ」 → 「バインディング (Binding)」 → 「接続」 → 「ManagedConnectionFactory プロパティ (ManagedConnectionFactory properties)」を選択します。
 - b. 固有のシステム要件に従って、ManagedConnectionFactory プロパティを編集します。
 - c. ActivationSpecification プロパティの編集が完了したら、「保管」をクリックします。
7. Business Integration パースペクティブで、 FlatFileOutboundModule を右クリックします。
8. ポップアップ・メニューから、「テスト」 → 「モジュールのテスト (Test Module)」を選択します。「イベント」ウィンドウが表示されます。
9. 「イベント」ウィンドウで、操作を選択します。
 - a. 実行する操作に基づいて、ビジネス・オブジェクトの値にデータを代入します。
 - b. WebSphere Process Server を始動し、Outbound アプリケーションをインストールする場合は、「継続 (Continue)」をクリックします。

シナリオ 2 の Inbound 操作の配置および構成

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Inbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

配置と構成を始める前に、プロジェクトに CWYFF_FlatFile.RAR ファイルをインポートしてください。この作業を行うには、WebSphere Integration Developer 内でエンタープライズ・サービス・ディスカバリーを実行して、アダプター構成プロパティを設定する必要があります。後でプロパティ値を変更するために、WebSphere Process Server 管理コンソールを使用します。

1. WebSphere Integration Developer ウィンドウで、Business Integration パースペクティブに切り替えるために、メニュー・バーから「ウィンドウ」 → 「パースペクティブを開く」 → 「その他 (Other)」を選択します。すべてのパースペクティブが表示されます。 WebSphere Integration Developer で、Business Integration パースペクティブを選択します。
2. WebSphere Integration Developer の Business Integration パースペクティブで、フレームを右クリックし、ポップアップ・メニューから「新規」 → 「エンタープライズ・サービス・ディスカバリー (Enterprise Service Discovery)」を選択します。「エンタープライズ・サービス・リソース・アダプターの選択 (Select an Enterprise Service Resource Adapter)」ウィンドウが表示されます。

3. 「CWYFF_FlatFile」コネクター・プロジェクトから「**IBM WebSphere Adapter for Flat Files (バージョン 6.0.0) (IBM WebSphere Adapter for Flat Files (version 6.0.0))**」を選択してから、「次へ」をクリックします。「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウが表示されます。
4. 「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウで、「サービス・タイプ」として「Inbound」を選択して接続プロパティを設定します。
5. オプション。「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウの下部で、「**拡張を表示**」ボタンをクリックします。ここでは、ロギング・レベルやトレース・レベルを設定できます。「次へ」をクリックします。
6. 「エンタープライズ・サービスの検索と発見 (Find and Discover Enterprise Services)」ウィンドウで、「**照会の実行 (Run Query)**」ボタンをクリックして、Flat Files アダプターのメタデータ・ツリーを表示します。
7. Inbound 操作の場合、「照会で発見したオブジェクト (Objects discovered by query)」の下で、メタデータ・ツリーの「Inbound」ルート・ノードを選択します。「照会で発見したオブジェクト (Objects discovered by query)」ボックスに、インポート用に選択できるオブジェクトが表示されます。インポートするオブジェクトを強調表示してから、「**選択されているものを追加 (Add Selected)**」ボタンをクリックして、そのオブジェクトを「インポートするオブジェクト (Objects to be imported)」ボックスに追加します。「次へ」をクリックします。

注: 「照会で発見したオブジェクト (Objects Discovered by query)」テキスト・ボックスからオブジェクトを削除するには、削除するオブジェクトを強調表示して、「**選択されているものを除去**」ボタンをクリックします。

8. 「オブジェクトの構成 (Configure Objects)」ウィンドウで、Discovery Agent によってインポートされるオブジェクトのプロパティを指定します。BO ロケーション・プロパティには、Business Integration モジュール内の .xsd ファイルが生成されるフォルダー名を指定します。「次へ」をクリックします。

Inbound でサポートされる操作には、次のものがあります。

- Read

9. 「成果物の生成 (Generate Artifacts)」ウィンドウで、以下の操作を行います。
 - a. 「**新規**」をクリックして、新しいビジネス・インテグレーション・モジュールを作成します。
 - b. 「新規モジュール」ウィンドウで、モジュール名 FlatFileInboundModule を入力して、「**終了**」をクリックします。
 - c. 「成果物の生成 (Generate Artifacts)」ウィンドウで、フォルダー名 FlatFileInboundFolder を入力します。
 - d. 「実行時にエンタープライズ情報システムとの接続に使用される接続プロパティを指定する (Specify the connection properties which will be used to connect to the Enterprise Information System at runtime)」の下で、「**発見された接続プロパティを使用 (Use discovered connection properties)**」を選択します。

- e. イベント配布データベース・プロパティを入力します。

注: 必須プロパティには、プロパティ・ボックスの横に * が付いています。
 - f. Flat Files アダプターの ActivationSpecification プロパティを入力します。

注: アスタリスク (*) が付いているプロパティは必須です。
 - g. ロギング値とトレース値を入力します。

注: 必須プロパティには、プロパティ・ボックスの横に * が付いています。
10. プロパティ値を入力し終わったら、「終了」をクリックします。 Business Integration パースペクティブに、新しいモジュールとそのすべての成果物が追加されます。
- これで、 WebSphere Integration Developer を使用して、エンタープライズ・サービス・ディスカバリー・ツールが作成する SCA コンポーネントの参照バインディングを生成することができます。参照バインディングは、 WebSphere Business Integration のほかの SCA コンポーネントがアダプターへのアクセス時に使用します。アダプターをほかのサーバー・プロセスにリンクするために、プロジェクト・モジュールからアダプターへの参照を作成します。
11. WebSphere Integration Developer の Business Integration パースペクティブで、モジュールを右クリックし、「アプリケーションから開く」 → 「アセンブリー・エディター」を選択します。「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウが表示され、モジュールのインポート・コンポーネントがビューに示されます。
12. 新規コンポーネントを作成するには、「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウの左側のフレーム (縦枠) の一番上のアイコンをクリックします。新しいアイコン・メニューが表示されます。
13. コンポーネントを実装なしで作成します。「コンポーネントの作成 (Create a Component)」アイコンを「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウへドラッグします。
14. FlatFileInboundInterface からこの新規コンポーネントへの接続を作成します。モジュールのインポート・コンポーネントをクリックして、新しいコンポーネントへドラッグします。これにより、インポート・コンポーネントから新しいコンポーネントへのワイヤーが描画されます。「ワイヤーの追加 (Add Wire)」ダイアログ・ボックスが表示されます。
15. 「OK」をクリックします。「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウに、新しい独立の参照コンポーネントが、モジュールのインポート・コンポーネントにワイヤーで接続された状態で表示されます。
16. エンドポイントのリスナーのような働きをする Java コンポーネントを作成します。 Inbound 操作のイベント配信の間に、アダプターはコンポーネントの実装の READ メソッドを呼び出し、Inbound ビジネス・オブジェクトをパラメーターとして渡します。
- a. コンポーネントを右クリックして、「実装の生成 (Generate implementation)」 → 「Java」を選択します。
 - b. Java コードを作成するパッケージを選択して、「OK」をクリックします。

- c. プロンプトが表示されたら、パッケージ名、リスナーを入力して、「OK」をクリックします。
 - d. 「実装の生成 (Generate Implementation)」ウィンドウで、「OK」をクリックします。
 - e. 生成された Java 実装で、READ メソッドまでスクロールします。READ メソッド内にカスタム・コードを追加して、配信されたビジネス・オブジェクトを特定のビジネス・ニーズに従って処理することができます。例えば、print ステートメントを追加して、イベント配信を検出することができます。
 - f. 「ファイル」 → 「保管」をクリックして、ファイルを保管します。
17. ActivationSpecifaciton プロパティを編集します。 WebSphere Integration Developer で、 FlatFileInboundModule の下の最初のエレメントをダブルクリックして、アセンブリー・エディターを開きます。
- a. 「プロパティ」 → 「バインディング (Binding)」 → 「接続」 → 「ActivationSpec プロパティ (ActivationSpec Properties)」を選択します。
 - b. 固有のシステム要件に従って、ActivationSpecification プロパティを編集します。
 - c. ActivationSpecification プロパティの編集が完了したら、「保管」をクリックします。
18. Business Integration パースペクティブで、 FlatFileInboundModule を右クリックします。
19. ポップアップ・メニューから、「テスト」 → 「モジュールのテスト (Test Module)」を選択します。「イベント」ウィンドウが表示されます。
20. 「イベント」ウィンドウで、サンプル・イベント・ファイルをイベント・ディレクトリに格納します。
- a. イベント・ファイルの拡張子が「.in」であることを確認します。
 - b. WebSphere Process Server を始動し、Inbound アプリケーションをインストールする場合は、「継続 (Continue)」をクリックします。

これで、すべての SCA 成果物を含むアダプター・アプリケーション・パッケージが構成され、配置されました。次に、サンプル・アプリケーションを実行することができます。

関連タスク

49 ページの『ロギング可能化』

WebSphere Adapter for Flat Files は、イベント処理の状況を判別するために表示できるログ・ファイルを維持しています。ログ・ファイル、および各ログ・エントリの日付、時刻、イベントによって、アダプターに関連するすべてのイベントやエラーを追跡します。アダプターはエラーや警告の状態の検出時にエラー・メッセージを記録するため、ログ・ファイルは問題のトラブルシューティングを開始するための有効な情報源となります。

51 ページの『トレースの使用可能化』

トレースによって、アダプター・ログ・ファイルに収集するエラーや警告のレベルが決まります。トレース・レベルを定義することによって、アダプターの処理に関するメッセージをたどって調べることができます。

関連資料

44 ページの『カスタム・プロパティー』

このトピックでは、WebSphere Adapter for Flat Files のデフォルトの構成プロパティーについて説明します。

42 ページの『J2C アクティブ化仕様プロパティー』

このトピックでは、(メッセージ・エンドポイント・プロパティーとしても知られる) J2C アクティブ化仕様プロパティーについて説明します。この仕様は、J2EE Connector Architecture Specification の ActivationSpecification インターフェースに対応します。

46 ページの『リソース・アダプター・プロパティー』

アダプターを初めて構成するとき (および後で WebSphere Process Server の管理コンソールを介して構成する場合)、エンタープライズ・サービス・ディスカバリー・ウィザードを使用すると、リソース・アダプター・プロパティーを構成できます。このカテゴリのプロパティーには、ロギングとトレースのプロパティーおよびアダプター固有のプロパティーがあります。

シナリオ 2 の Outbound 操作用の配置および構成

サンプル・アプリケーションのシナリオ 2 では、エンタープライズ・サービス・ディスカバリー・ウィザードを使用して、Outbound 処理用にアプリケーション・パッケージを配置し、アダプターを構成して、SCA 成果物を生成する必要があります。

配置と構成を始める前に、プロジェクトに CWYFF_FlatFile.RAR ファイルをインポートしてください。この作業を行うには、IBM WebSphere Integration Developer 内でエンタープライズ・サービス・ディスカバリーを実行して、アダプター構成プロパティーを設定する必要があります。後でプロパティー値を変更するために、WebSphere Process Server 管理コンソールを使用します。

1. WebSphere Integration Developer ウィンドウで、Business Integration パースペクティブに切り替えるために、メニュー・バーから「ウィンドウ」 → 「パースペクティブを開く」 → 「その他 (Other)」を選択します。すべてのパースペクティブが表示されます。WebSphere Integration Developer で、Business Integration パースペクティブを選択します。
2. WebSphere Integration Developer の Business Integration パースペクティブで、フレームを右クリックし、ポップアップ・メニューから「新規」 → 「エンタープライズ・サービス・ディスカバリー (Enterprise Service Discovery)」を選択します。「エンタープライズ・サービス・リソース・アダプターの選択 (Select an Enterprise Service Resource Adapter)」ウィンドウが表示されます。
3. 「CWYFF_FlatFile」コネクタ・プロジェクトから「IBM WebSphere Adapter for Flat Files (バージョン 6.0.0) (IBM WebSphere Adapter for Flat Files (version 6.0.0))」を選択してから、「次へ」をクリックします。「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウが表示されます。
4. 「Discovery Agent の設定の構成 (Configure Settings for Discovery Agent)」ウィンドウで、「サービス・タイプ」として「Outbound」を選択して接続プロパティーを設定します。「Outbound」は、デフォルトのサービス・タイプです。

5. Discovery Agent の「プロパティの初期設定」ウィンドウの下部で、「拡張を表示」ボタンをクリックします。ここでは、ロギング・レベルやトレース・レベルを設定できます。「次へ」をクリックします。
6. 「エンタープライズ・サービスの検索と発見 (Find and Discover Enterprise Services)」ウィンドウで、「照会の実行 (Run Query)」をクリックします。
7. 「発見したオブジェクト (Objects Discovered)」の下で、インポートするオブジェクトを選択します。「発見したオブジェクト (Objects Discovered)」ボックスに、インポート用に選択できるオブジェクトが表示されます。インポートするオブジェクトを強調表示してから、「**選択されているものを追加 (Add Selected)**」ボタンをクリックして、そのオブジェクトを「インポートするオブジェクト (Objects to be imported)」ボックスに追加します。「次へ」をクリックします。

注: 「発見したオブジェクト (Objects Discovered)」ボックスからオブジェクトを削除するには、削除するオブジェクトを強調表示して、「**選択されているものを除去**」ボタンをクリックします。

8. 「オブジェクトの構成 (Configure Objects)」ウィンドウで、Discovery Agent によってインポートされるオブジェクトのプロパティを指定します。BO ロケーション・プロパティには、Business Integration モジュール内の .xsd ファイルが生成されるフォルダー名を指定します。「次へ」をクリックします。

Outbound でサポートされる操作には、次のものがあります。

- Create
- Append
- Retrieve
- Delete
- Overwrite
- Exists
- List

9. 「成果物の生成 (Generate Artifacts)」ウィンドウで、以下の操作を行います。
 - a. 「成果物の生成 (Generate Artifacts)」ウィンドウで、SCA 成果物 (ビジネス・オブジェクト、そのプロパティ、インポート・ファイル、エクスポート・ファイル、および WSDL) の保管先のモジュールの名前を指定し、「**新規**」をクリックして、新しいビジネス・インテグレーション・モジュールを作成します。「新規モジュール」ウィンドウが表示されます。
 - b. 「新規モジュール」ウィンドウで、モジュール名を入力して、「**終了**」をクリックします。モジュール名は FlatFileOutboundModule とします。
 - c. 「成果物の生成 (Generate Artifacts)」ウィンドウで、フォルダー名を入力します。フォルダー名は FlatFileOutboundFolder とします。
 - d. 「実行時にエンタープライズ情報システムとの接続に使用される接続プロパティを指定する (Specify the connection properties which will be used to connect to the Enterprise Information System at runtime)」の下で、「**発見された接続プロパティを使用 (Use discovered connection properties)**」を選択します。

- e. 「成果物の生成 (Generate Artifacts)」 ウィンドウで、Flat Files の ManagedConnectionFactory プロパティを指定します。

注: アスタリスク (*) が付いているプロパティは必須です。

- f. 「成果物の生成 (Generate Artifacts)」 ウィンドウで、ResourceAdapter プロパティを指定します。ここでは、ロギング・ファイルおよびトレース・ファイルのサイズとファイル名を指定することもできます。

注: アスタリスク (*) が付いているプロパティは必須です。

- 10. プロパティ値を入力し終わったら、「終了」をクリックします。 Business Integration パースペクティブに、新しいモジュールとそのすべての成果物が追加されます。

これで、WebSphere Integration Developer を使用して、エンタープライズ・サービス・ディスカバリー・ツールが作成する SCA コンポーネントの参照バインディングを生成することができます。参照バインディングは、WebSphere Business Integration のほかの SCA コンポーネントがアダプターへのアクセス時に使用します。アダプターをほかのサーバー・プロセスにリンクするために、プロジェクト・モジュールからアダプターへの参照を作成します。

- 11. WebSphere Integration Developer の Business Integration パースペクティブで、モジュールを右クリックし、「アプリケーションから開く」 → 「アセンブリー・エディター」を選択します。「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウが表示され、モジュールのインポート・コンポーネントがビューに示されます。
- 12. 独立の参照を作成するには、「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウの左側のフレーム (縦枠) の一番上にあるアイコンをクリックします。新しいアイコン・メニューが表示されます。
- 13. 独立の参照を作成します。「独立の参照 (Stand-alone Reference)」アイコンを「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウへドラッグします。
- 14. 独立の参照から FlatFileOutboundInterface への接続を作成します。モジュールのインポート・コンポーネントをクリックして、独立の参照へドラッグします。これにより、インポート・コンポーネントから新しい独立の参照へのワイヤーが描画されます。「ワイヤーの追加 (Add Wire)」ダイアログ・ボックスが表示されます。
- 15. 「OK」をクリックします。「アセンブリー・ダイアグラム (Assembly Diagram)」ウィンドウに、新しい独立の参照コンポーネントが、モジュールのインポート・コンポーネントにワイヤーで接続された状態が表示されます。
- 16. ManagedConnectionFactory プロパティを編集します。WebSphere Integration Developer で、FlatFileOutboundModule の下の最初のエレメントをダブルクリックして、アセンブリー・エディターを開きます。
 - a. 「プロパティ」 → 「バインディング (Binding)」 → 「接続」 → 「ManagedConnectionFactory プロパティ (ManagedConnectionFactory properties)」を選択します。
 - b. 固有のシステム要件に従って、ManagedConnectionFactory プロパティを編集します。

- c. ManagedConnectionFactory プロパティの編集が完了したら、「保管」をクリックします。
17. Business Integration パースペクティブで、FlatFileOutboundModule を右クリックします。
18. ポップアップ・メニューから、「テスト」 → 「モジュールのテスト (Test Module)」を選択します。「イベント」ウィンドウが表示されます。
19. 「イベント」ウィンドウで、操作を選択します。
 - a. 実行する操作に基づいて、ビジネス・オブジェクトの値にデータを代入します。
 - b. WebSphere Process Server を始動し、Outbound アプリケーションをインストールする場合は、「継続 (Continue)」をクリックします。

これで、すべての SCA 成果物を含むアダプター・アプリケーション・パッケージが構成され、配置されました。次に、サンプル・アプリケーションを実行することができます。

関連タスク

49 ページの『ロギング可能化』

WebSphere Adapter for Flat Files は、イベント処理の状況を判別するために表示できるログ・ファイルを維持しています。ログ・ファイル、および各ログ・エントリーの日付、時刻、イベントによって、アダプターに関連するすべてのイベントやエラーを追跡します。アダプターはエラーや警告の状態の検出時にエラー・メッセージを記録するため、ログ・ファイルは問題のトラブルシューティングを開始するための有効な情報源となります。

51 ページの『トレースの使用可能化』

トレースによって、アダプター・ログ・ファイルに収集するエラーや警告のレベルが決まります。トレース・レベルを定義することによって、アダプターの処理に関するメッセージをたどって調べることができます。

関連資料

41 ページの『J2C 接続ファクトリー・プロパティ』

このトピックでは、ターゲットのエンタープライズ情報システム (EIS) インスタンスの構成に使用する J2C 接続ファクトリーとプロパティについて説明します。これらのプロパティは、Outbound 処理に影響を与え、J2EE Connector Architecture 仕様の ManagedConnectionFactory インターフェースに対応します。

46 ページの『リソース・アダプター・プロパティ』

アダプターを初めて構成するとき (および後で WebSphere Process Server の管理コンソールを介して構成する場合)、エンタープライズ・サービス・ディスカバリー・ウィザードを使用すると、リソース・アダプター・プロパティを構成できます。このカテゴリーのプロパティには、ロギングとトレースのプロパティおよびアダプター固有のプロパティがあります。

サンプル・アプリケーションの実行 (Inbound 操作の場合)

サンプル・アプリケーションのパッケージおよびアダプターの配置と構成が終了したら、アダプターがビジネス・オブジェクトの Outbound 処理をサポートする様子を示すこのアプリケーションを実行することができます。

1. WebSphere Process Server に FlatFileInboundModuleApp.ear を配置します。

2. WebSphere Process Server の管理コンソールを使用して、ActivationSpecification の構成を編集します。
3. WebSphere Process Server のクラスパスに、必要なデータベース・ドライバーを格納します。
4. イベント・ファイルをイベント・ディレクトリーに格納します。
5. Inbound アプリケーションを開始します。
6. イベント・ファイルがポーリングされたことを確認します。

注: システムにインストールされている RAR ファイルは、サンプルで提供されるファイルより新しいバージョンです。機能上の大きな不一致があった場合、サンプル・パッケージ内の RAR ファイルの代わりにインストールされている RAR を使用して、エンタープライズ・サービス・ディスカバリー・ウィザードで EAR を生成し、この EAR をサーバーに再配置してください。

サンプル・アプリケーションの実行 (Outbound 操作の場合)

サンプル・アプリケーションのパッケージおよびアダプターの配置と構成が終了したら、アダプターがビジネス・オブジェクトの Outbound 処理をサポートする様子を示すこのアプリケーションを実行することができます。

1. WebSphere Process Server に FlatFileOutboundModuleApp.ear を配置します。
2. WebSphere Process Server の管理コンソールを使用して、ConnectionFactory の構成を編集します。
3. ServiceClient を呼び出すために、Web ブラウザーを開き、サーブレット・クライアントを実行するための URL を入力します。例えば、動的 Web プロジェクトの名前が「MyWebProject」とするとします。この場合の URL は `http://localhost:9080/MyWebProject/ServiceClient` のようになります。
4. 以下の入力を行い、「**NewObject**」をクリックします。
 - 「名前空間」: .wsdl ファイル内のターゲット名前空間と同一である必要があります。
 - 「操作名」: テストする操作を入力します。例えば、「ビジネス・オブジェクト」を入力します。
 - 「IsWrappedStyle」: 新規オブジェクトを作成する場合は、True に設定します。
5. 選択した操作の必要な属性を入力し、IsWrappedType を false として設定します。「**OK**」をクリックします。
6. アダプターによるサービスの処理方法を確認するために、アプリケーションの出力を調べます。

注: システムにインストールされている RAR ファイルは、サンプルで提供されるファイルより新しいバージョンです。機能上の大きな不一致があった場合、サンプル・パッケージ内の RAR ファイルの代わりにインストールされている RAR を使用して、エンタープライズ・サービス・ディスカバリー・ウィザードで EAR を生成し、この EAR をサーバーに再配置してください。

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。

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。

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

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

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

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

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

IBM 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が、その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

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

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

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

著作権使用許諾:

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

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

以下は、IBM Corporation の商標です。

IBM
IBM ロゴ
AIX
AIX 5L
CICS
CrossWorlds
DB2
DB2 Universal Database
HelpNow
IMS
Informix
iSeries
Lotus
Lotus Domino
Lotus Notes
MQIntegrator
MQSeries
MVS
OS/400
Passport Advantage
pSeries
Redbooks
SupportPac
WebSphere
z/OS

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または登録商標です。

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

Intel、Intel (ロゴ)、Intel Inside、Intel Inside (ロゴ)、Pentium、Intel Centrino、Intel Centrino (ロゴ)、Celeron、Intel Xeon、Intel SpeedStep、および Itaniumは、Intel 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。



WebSphere Adapters バージョン 6.0



Printed in Japan