





**WebSphere Adapter for JD Edwards EnterpriseOne
ユーザーズ・ガイド
バージョン 6 リリース 2**

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、165 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 JD Edwards EnterpriseOne バージョン 6、リリース 2、モディフィケーション 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。

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® Adapters
Version 6 Release 2
WebSphere Adapter for JD Edwards EnterpriseOne User Guide
Version 6 Release 2

発行： 日本アイ・ビー・エム株式会社

担当： トランスレーション・サービス・センター

第1刷 2009.2

© Copyright International Business Machines Corporation 2006, 2008.

目次

第 1 章 WebSphere Adapter for JD

Edwards EnterpriseOne の概要 1

ハードウェア要件とソフトウェア要件	1
Adapter for JD Edwards EnterpriseOne の技術概要	2
Outbound 処理	2
Inbound 処理	3
ビジネス・オブジェクト	5
外部サービス・ウィザード	11

第 2 章 アダプター実装の計画 15

作業を始める前に	15
セキュリティー	15
ログ・ファイルおよびトレース・ファイル内の機密性の高いユーザー・データ保護のサポート	15
ユーザー認証	16
デプロイメント・オプション	18
クラスター化された環境の WebSphere Adapters	20
バージョン 6.2 へのマイグレーション	22
マイグレーションに関する考慮事項	22
マイグレーションの実行	24
マイグレーションしない場合のバージョン 6.0.2.x またはバージョン 6.1.x プロジェクトの更新	26

第 3 章 サンプルおよびチュートリアル 29

第 4 章 デプロイメントのためのモジュールの構成 31

モジュールの構成のためのロードマップ	31
認証別名の作成	33
プロジェクトの作成	34
外部ソフトウェア依存関係の追加	37
外部依存関係ファイルの編集	41
外部サービス・ウィザードの接続プロパティの設定	42
Outbound 処理のモジュールの構成	44
外部サービス・ディスカバリーを使用したビジネス関数の生成	44
外部サービス・ディスカバリーを使用した XML List の生成	57
Inbound 処理のモジュールの構成	68
ビジネス・オブジェクトおよびサービスの選択	68
選択されたオブジェクトの構成	73
デプロイメント・プロパティの設定およびサービスの生成	73

第 5 章 アセンブリー・エディターによる対話仕様プロパティの変更 79

第 6 章 モジュールのデプロイ 81

デプロイメント環境	81
---------------------	----

テスト用のモジュールのデプロイ	81
外部依存関係の追加	81
Inbound 処理をテストするためのターゲット・コンポーネントの生成および接続	84
サーバーへのモジュールの追加	86
テスト・クライアントを使用した Outbound 処理用モジュールのテスト	86
実稼働用のモジュールのデプロイ	87
サーバー・ランタイム環境への外部ソフトウェア依存関係の追加	87
RAR ファイルのインストール (スタンドアロン・アダプターを使用するモジュールの場合のみ)	89
EAR ファイルとしてのモジュールのエクスポート	91
EAR ファイルのインストール	92

第 7 章 アダプター・モジュールの管理 95

組み込みアダプターの構成プロパティの変更	95
組み込みアダプターのリソース・アダプター・プロパティの設定	95
組み込みアダプターの Managed (J2C) Connection Factory プロパティの設定	97
組み込みアダプターの活動化仕様プロパティの設定	99
スタンドアロン・アダプターの構成プロパティの変更	101
スタンドアロン・アダプターのリソース・アダプター・プロパティの設定	101
スタンドアロン・アダプターの Managed (J2C) Connection Factory プロパティの設定	102
スタンドアロン・アダプターの活動化仕様プロパティの設定	104
アダプターを使用するアプリケーションの開始	106
アダプターを使用するアプリケーションの停止	106
Performance Monitoring Infrastructure を使用したパフォーマンスのモニター	107
Performance Monitoring Infrastructure の構成	107
パフォーマンスに関する統計の表示	110
Common Event Infrastructure (CEI) を使用したトレースの使用可能化	111

第 8 章 トラブルシューティングおよびサポート 113

Log and Trace Analyzer のサポート	113
ロギングおよびトレースの構成	114
ロギング・プロパティの構成	114
ログ・ファイル名およびトレース・ファイル名の変更	116
First Failure Data Capture (FFDC) サポート	117
ビジネス・フォールト	117
XAResourceNotAvailableException	119
セルフ・ヘルプ・リソース	120

第 9 章 参照情報	123
ビジネス・オブジェクト情報	123
アプリケーション固有情報	123
サポートされる操作	125
命名規則	126
Outbound 構成プロパティ	126
外部サービス・ウィザードの接続プロパティ	128
リソース・アダプター・プロパティ	132
Managed Connection Factory プロパティ	136
対話仕様プロパティ	141
Inbound 構成プロパティ	142
外部サービス・ウィザードの接続プロパティ	143

リソース・アダプター・プロパティ	147
活動化仕様プロパティ	151
グローバリゼーション	159
グローバリゼーションおよび双方向データ変換	159
双方向データ変換で使用可能なプロパティ	161
関連情報	162

特記事項	165
プログラミング・インターフェース情報	167
商標	167
索引	169

第 1 章 WebSphere Adapter for JD Edwards EnterpriseOne の概要

WebSphere Adapter for JD Edwards EnterpriseOne を使用すると、JD Edwards EnterpriseOne サーバー との情報交換が組み込まれた統合プロセスを、特別なコーディングなしで作成できます。

アダプターには、標準インターフェースがあるため、コンポーネントがアプリケーションの下位の実装詳細またはデータ構造を把握する必要はありません。このアダプターを使用すると、コンポーネント (特定のビジネス関数を実行するプログラムまたはコード断片) は、JD Edwards EnterpriseOne サーバー に要求 (例えば JD Edwards EnterpriseOne テーブルの顧客レコードの照会や注文文書の更新などの要求) を送信することができるようになります。

WebSphere Adapter for JD Edwards EnterpriseOne は、Java™ 2 Platform, Enterprise Edition (J2EE) Connector Architecture (JCA) に準拠しています。JCA によって、アプリケーション・コンポーネント、アプリケーション・サーバー、および JD Edwards EnterpriseOne サーバー などのエンタープライズ情報システムが相互に対話するための方法が標準化されます。WebSphere Adapter for JD Edwards EnterpriseOne を使用すれば、JCA に準拠したアプリケーション・サーバーが JD Edwards EnterpriseOne サーバーに接続して対話することが可能になります。JCA に準拠したサーバーで稼働するクライアントは、標準的な方法で (ビジネス・オブジェクトや JavaBeans™ を使用して) JD Edwards EnterpriseOne サーバー と通信できるようになります。

以下の例は、WebSphere Integration Developer を使用してアダプターをセットアップし、そのアダプターを組み込んだモジュールを WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイすることを前提としています。

JD Edwards EnterpriseOne を使用して、ほとんどのビジネス・オペレーションを行う中規模の小売企業があるとします。JD Edwards EnterpriseOne には、その企業の全米 100 店舗のリアルタイムの在庫リストを返すことのできるビジネス関数が組み込まれています。アプリケーション・コンポーネントでは、ビジネス・プロセス全体の一部としてこのビジネス関数を使用できる場合があります。例えば、小売企業の従業員は、調達可能な在庫のリアルタイムのリストにアクセスし、正確なリアルタイム情報を顧客に提供することができます。

ハードウェア要件とソフトウェア要件

WebSphere Adapters のハードウェア要件とソフトウェア要件は、IBM® Support Web サイトに記載されています。

WebSphere Adapters<http://www.ibm.com/support/docview.wss?uid=swg27006249> を参照してください。

追加情報

以下のリンク先には、アダプターの構成およびデプロイを行う際に必要となる可能性のある追加情報が記載されています。

- WebSphere Business Integration Adapters および WebSphere Adapters の互換性マトリックスによって、ご使用のアダプターで必要となるソフトウェアのサポート対象バージョンが識別されます。この資料を表示するには、WebSphere Adapters のサポート・ページ (<http://www.ibm.com/software/integration/wbiadapters/support/>) にアクセスして、「**Additional support links**」セクションの「**Related**」の見出しの下にある「**Compatibility Matrix**」をクリックしてください。
- WebSphere Adapters の技術情報には、製品文書には記載されていない回避策および追加情報が記載されています。ご使用のアダプターの技術情報を表示するには、<http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm> にアクセスして、「**Product category**」リストの中からアダプター名を選択し、検索アイコンをクリックしてください。

Adapter for JD Edwards EnterpriseOne の技術概要

IBM WebSphere Adapter for JD Edwards EnterpriseOne は、アプリケーションが JD Edwards EnterpriseOne サーバー のデータと相互作用を行う手段を提供します。Outbound 処理 (アダプターを介したアプリケーションから JD Edwards EnterpriseOne サーバー への要求処理) がサポートされています。

このアダプターでは、ビジネス関数および XML List という 2 つのタイプのビジネス・オブジェクトのいずれかを使用して要求を処理します。ビジネス関数は、単一トランザクションとして処理できる 1 つまたは多数のビジネス・オブジェクトを組み込むことができるビジネス・オブジェクト・コンテナです。XML List は、テーブルを照会し、複数のレコードを返すことができる単一のビジネス・オブジェクトです。

ビジネス・オブジェクトは、外部サービス・ウィザードという、WebSphere Integration Developer から起動されるツールを使用して作成します。外部サービス・ウィザードによって生成されたビジネス・オブジェクトには、事前定義のビジネス・オブジェクト定義が設定されています。生成済みのビジネス・オブジェクトに対して機能を追加または削除したい場合、ビジネス・オブジェクト・エディターという、やはり WebSphere Integration Developer から起動されるツールを使用して、このビジネス・オブジェクトの定義のプロパティーを変更することができます。また、WebSphere Integration Developer の管理コンソールでも、ビジネス・オブジェクトの構成プロパティーを変更または設定できます。

Outbound 処理

Adapter for JD Edwards EnterpriseOne は、同期的な Outbound 要求の処理をサポートしています。これは、アダプターはモジュールから要求をビジネス・オブジェクトの形式で受信すると、要求を処理し、適用できる場合は結果をビジネス・オブジェクトの形で返すことを示しています。

アダプターが要求を受信すると、JD Edwards EnterpriseOne Dynamic Java コネクタ
ーを使用して要求を処理し、ビジネス関数または XML List を起動します。

ビジネス関数は、以下のタイプの操作をサポートしています。

- create
- delete
- execute
- retrieve
- update

XML List は、以下の操作をサポートしています。

- retrieveAll

Inbound 処理

WebSphere Adapter for JD Edwards EnterpriseOne は、非同期の Inbound 処理をサポートしています。これは、アダプターが、イベントに対して指定された間隔で、JD Edwards EnterpriseOne サーバー をポーリングすることを示しています。アダプターは、イベントを受信すると、イベント・データをビジネス・オブジェクトに変換して、それをコンポーネントに送信します。

WebSphere Adapter for JD Edwards EnterpriseOne は、リアルタイム・イベントをサポートしています。リアルタイム・イベントは、サード・パーティー・システムとの相互運用時に使用可能な、JD Edwards EnterpriseOne サーバー からの情報を提供するビジネス・トランザクションです。リアルタイム・イベントは、HTML、WIN32、およびエンタープライズ・サーバーなど、ビジネス関数が実行される場所ならば、どのような場所にも生成することができます。リアルタイム・イベントは、リアルタイムで通知を生成する場合に便利です。アダプターは、単一およびコンテナのリアルタイム・イベントをサポートしています。

アダプターは JD Edwards EnterpriseOne Dynamic Java Connector API を起動して、JD Edwards EnterpriseOne トランザクション・サーバーからリアルタイム・イベントを取得するときに、その内容を解析してビジネス・オブジェクトに変換します。さらに、アダプターは、そのビジネス・オブジェクトをイベント・エンドポイントに送信します。例えば、ある会社が更新された場合、JD Edwards EnterpriseOne サーバーが、この変更を即座に取り込み、JD Edwards EnterpriseOne トランザクション・サーバーによって、1 つのリアルタイム・イベントが生成されます。次にアダプターが、JD Edwards EnterpriseOne トランザクション・サーバーと通信して、このリアルタイム・イベントを取得し、これを処理します。ビジネス・オブジェクトに変換後、アダプターは、変換されたビジネス・オブジェクトをイベント・エンドポイントに送信します。

WebSphere Adapter for JD Edwards EnterpriseOne では、次のようにイベントを処理します。

1. アダプターは、JD Edwards EnterpriseOne Dynamic Java Connector API を起動し、リアルタイム・イベントを取得します。
2. アダプターは、このリアルタイム・イベントの内容を解析します。

3. アダプターは、このリアルタイム・イベントのペイロードから取得した値を、関連するビジネス・オブジェクトに設定します。
4. アダプターは、生成されたビジネス・オブジェクトを登録済みの各アプリケーションに送信します。

注: Inbound 処理が行われるためには、あらかじめリアルタイム・イベントをサポートするように JD Edwards EnterpriseOneサーバーを構成する必要があります。

イベント・パーシスタンス

アダプターでは、予期しない強制終了に備えて Inbound 処理のイベント・パーシスタンスをサポートしています。イベント・パーシスタンス (あるいは、1 回送達保証) は、障害時においてもエンドポイントへのイベントの送達が重複せず、1 回のみ行われることを保証する方法です。イベント処理中、アダプターは、データ・ソースにあるイベント・ストアにイベントの状態を保持します。イベント・ストアを作成するには、あらかじめ、WebSphere Process Server または WebSphere Enterprise Service Bus を使用して、このデータ・ソースをセットアップする必要があります。WebSphere Process Server または WebSphere Enterprise Service Bus で提供されるリカバリー機能を使用するには、活動化仕様の AssuredOnceDelivery プロパティを True に設定します。このリカバリー機能は、デフォルトで True に設定されています。

また、アダプターでは、イベント・ストアのメモリー内表記を使用するイベント・パーシスタンスも提供しています。この機能を使用すると、JNDI データ・ソースまたは外部イベント・ストアを作成する必要がなく、イベント処理が速まります。ただし、この機能を使用した場合、イベント・リカバリーのサポートはなくなります。サーバー障害が発生すると、メモリー内のイベント・ストアは失われます。

サーバー障害時のイベント喪失を防ぐために推奨される方法は、データベース・イベント・ストアを使用することです。アダプターのメモリー内イベント・パーシスタンス機能を使用する場合、EP_DataSource_JNDIName プロパティを設定してはなりません。

イベントが失敗し、ファイルをディスクに書き込めない場合、JDE アダプターは、逸失しないように、この失敗したイベントの内容を JDE トレース・ファイルに出力します。この内容は、2 つの "#####" 間にあり、トレース・レベルは INFO になります。この内容は、新規のファイルにコピーし、イベント ID をファイル名にしてディレクトリー、"FailedEventFolder" に保存することができます。失敗したイベント・レコードの状況がイベント・テーブルで -1 から 0 に変わると、そのイベントは再度処理されます。

イベント・ストア

イベント・ストアは、ポーリング・アダプターがイベント・レコードを処理できるまでイベント・レコードが保存されるパーシスタント・キャッシュです。アダプターは、Inbound イベントがシステム内を進行するときに、イベント・ストアを使用して Inbound イベントを追跡します。リアルタイム・イベントが受信されるたびに、アダプターはイベント・ストア内のイベントの状況を更新します。各イベントの状況は、イベントがエンドポイントに渡されるまで、リカバリーの目的のために、アダプターによって継続的に更新されます。

アダプターが、WebSphere Process Server または WebSphere Enterprise Service Bus 内に Inbound モジュールのイベント・ストアがないことを検出すると、アダプターでは、アプリケーションがランタイムにデプロイされる時に自動的にイベント・ストアを作成します。アダプターによって作成された各イベント・ストアは、特定の Inbound モジュールに関連付けられます。アダプターは、同じイベント・ストアを指す複数のアダプター・モジュールをサポートしていません。

アダプターは、JD Edwards EnterpriseOne トランザクション・サーバーをポーリングしてリアルタイム・イベントを受信すると、活動化仕様プロパティに指定された検索基準に一致する各イベントについて、イベント・ストアにエントリーを作成します。アダプターは、新しいエントリーの状況を NEW として記録します。

リアルタイム・イベントが正常に送達されると、対応するイベント・ストアのエントリーが削除されます。失敗したイベントについては、エントリーはイベント・ストアに残ります。

1 回送達保証

JD Edwards EnterpriseOne トランザクション・サーバーでは、イベント送達サービスの品質保証を提供しています。これは、アダプターが登録するすべてのリアルタイム・イベントが、失われることなくアダプターに送達されることを示しています。この時、JD Edwards EnterpriseOne トランザクション・サーバーがリアルタイム・イベントを重複してアダプターに送信する可能性があるため、アダプターでは、イベントの 1 回送達保証を指定できます。この場合、各イベントは必ず 1 回のみ配信されることとなります。1 回送達保証を有効にするには、AssuredOnceDelivery 活動化仕様プロパティを True に設定します。

注: AssuredOnceDelivery 活動化仕様プロパティを True に設定した場合、AutoAcknowledge 活動化仕様プロパティ False に設定する必要があります。そうでないと、1 回送達保証機能が作動しなくなります。

AssuredOnceDelivery 活動化仕様プロパティを True に設定した場合、アダプターは、イベント・ストアに各イベントの XID (トランザクション ID) 値を保管します。

リアルタイム・イベントが取得されると、次のように処理されます。

1. イベントの XID 値が、イベント・ストア内で更新されます。
2. イベントが、対応するエンドポイントに送信されます。
3. イベント・エントリーが、イベント・ストアから削除されます。
4. 肯定応答が、JD Edwards EnterpriseOne トランザクション・サーバーに送出されます。

ビジネス・オブジェクト

ビジネス・オブジェクトとは、データ、データに対して実行されるアクション、およびデータを処理するための追加の指示 (存在する場合) で構成される構造体のことです。データは、請求書または従業員記録などのビジネス・エンティティ、または、未構造のテキストのいずれかを表します。アダプターは、ビジネス・オブジェクトを使用して、JD Edwards EnterpriseOne サーバー との間でデータ送信やデータ取得を行います。

アダプターによるビジネス・オブジェクトの使用法

アダプターは、JD Edwards EnterpriseOne Dynamic Java Connector API を使用して、JD Edwards EnterpriseOne アプリケーションと通信を行います。アダプターでは、ビジネス関数、XML List の呼び出し、およびリアルタイム・イベント機構によって EnterpriseOne と情報を交換します。

以下の図は、Adapter for JD Edwards EnterpriseOne が、ビジネス・オブジェクトを使用して、Outbound 処理、Inbound 処理を行う状況を示しています。

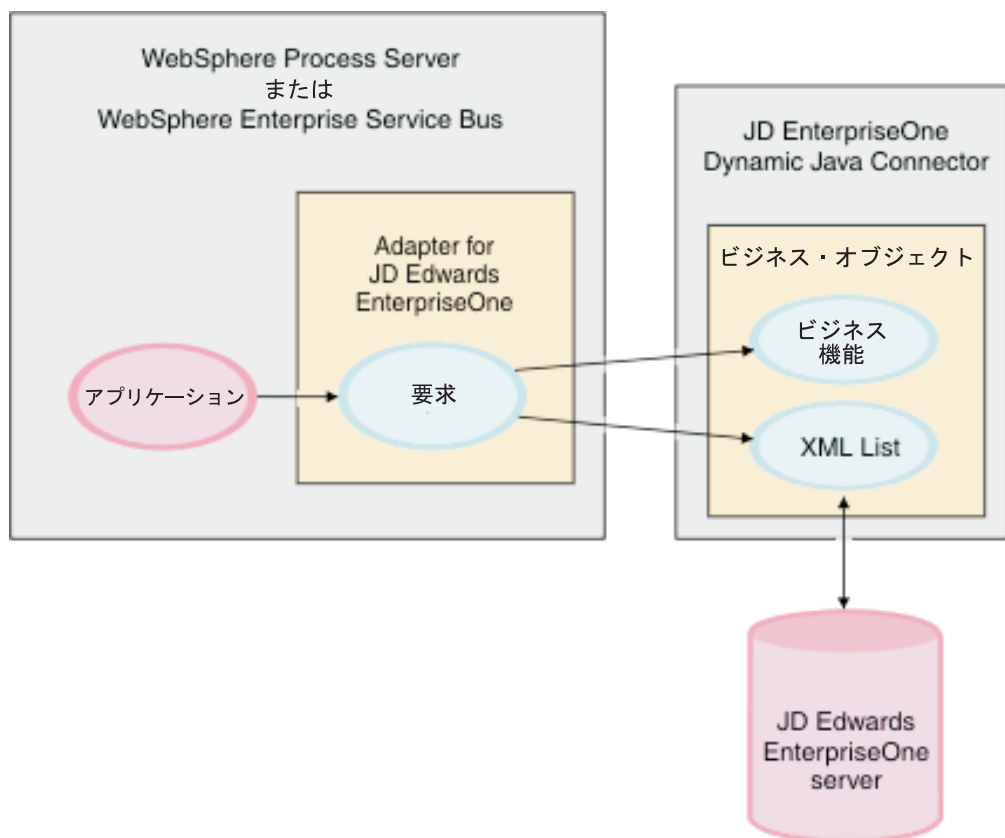


図1. Outbound 処理時の Adapter for JD Edwards EnterpriseOne によるビジネス・オブジェクトの使用

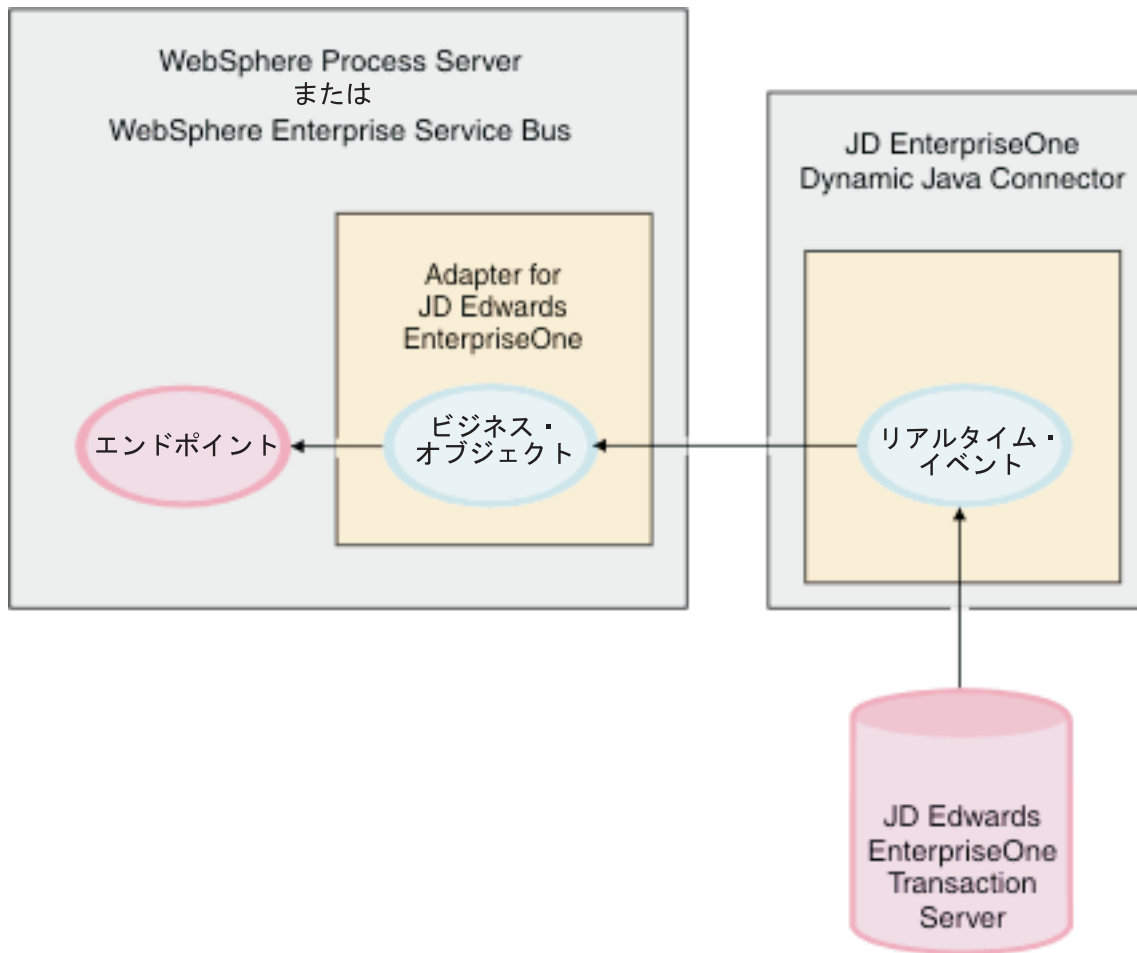


図2. Inbound 処理時の Adapter for JD Edwards EnterpriseOne によるビジネス・オブジェクトの使用

ビジネス・オブジェクトが作成される仕組み

ビジネス・オブジェクトは、WebSphere Integration Developer から起動される、外部サービス・ウィザードを使用して作成します。外部サービス・ウィザードは、アプリケーションに接続して、アプリケーションのデータ構造をディスカバーし、そのデータ構造を表すビジネス・オブジェクトを生成します。さらに、アダプターで必要とされるその他の成果物も生成します。

以下の図は、外部サービス・ウィザードを使用してビジネス・オブジェクトを作成する時のステップの概要フローを示しています。

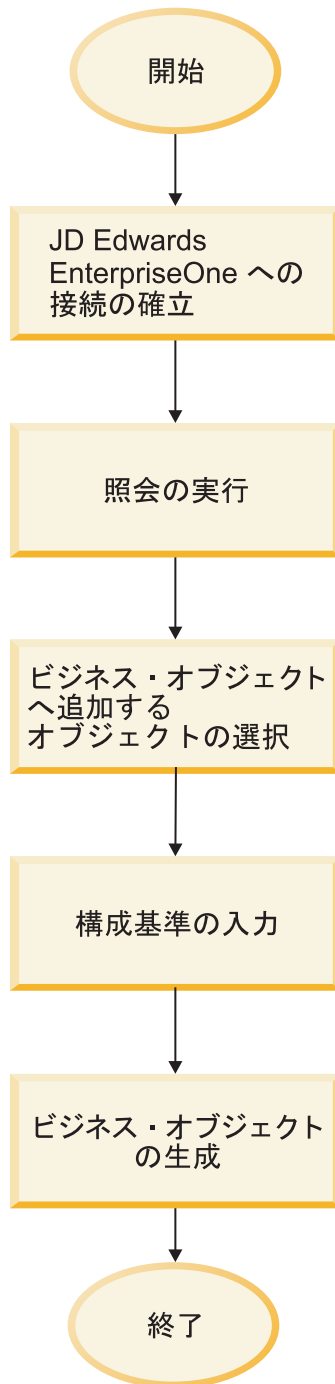


図3. 外部サービス・ウィザードを使用したビジネス・オブジェクトの作成

ビジネス・オブジェクトの構造

アダプターは、階層型のビジネス・オブジェクトの処理をサポートしています。ビジネス・グラフの下に位置する最上位のビジネス・オブジェクトは、ラッパー・コンテナ・ビジネス・オブジェクトです。コンテナ・ビジネス・オブジェクトは、JD Edwards EnterpriseOne 操作を表し、1 つまたは複数の子ビジネス関数オブジェクト (単純ビジネス関数オブジェクトとも言います) を含むラッパー・オブジェ

クトです。各ビジネス関数オブジェクトは、JD Edwards EnterpriseOne アプリケーション内の特定の関数呼び出しを表しています。

以下の図は、ビジネス・グラフ、ビジネス・オブジェクト・コンテナ、単純 (子) ビジネス・オブジェクト、コンテナ・リアルタイム・イベントのコンテナ・ビジネス・オブジェクト、および単一リアルタイム・イベントのフラット・ビジネス・オブジェクトのダイアグラム・ビューを示しています。

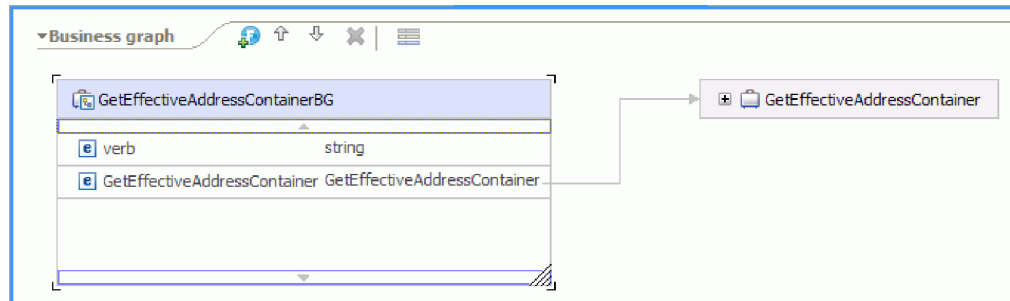


図 4. WebSphere Integration Developer のダイアグラム・ビューに示されたビジネス・オブジェクト・グラフの構造

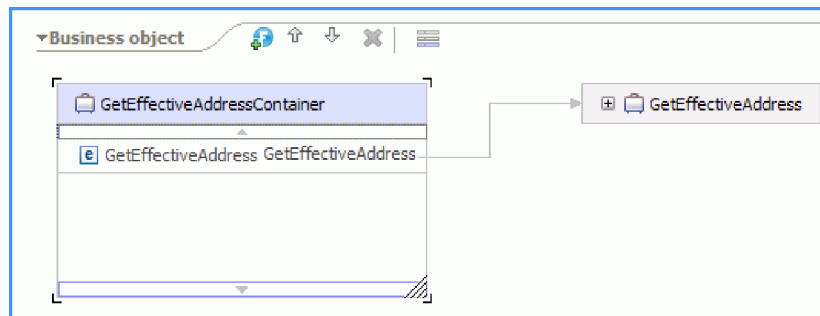


図 5. WebSphere Integration Developer のダイアグラム・ビューに示されたビジネス・オブジェクト・コンテナの構造

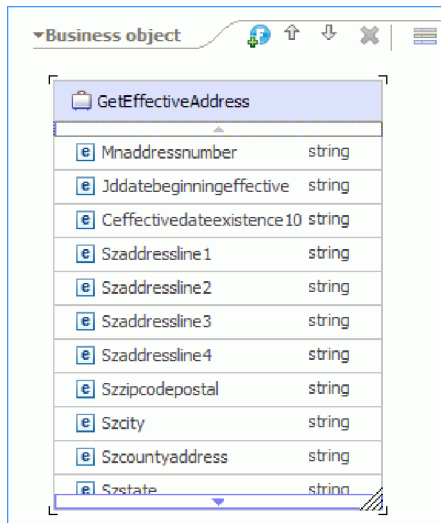


図6. WebSphere Integration Developer のダイアグラム・ビューに示された子ビジネス・オブジェクトの構造

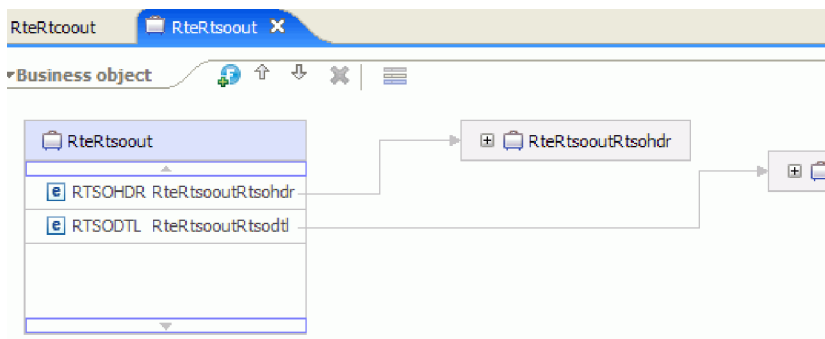


図7. コンテナ・リアルタイム・イベントのコンテナ・ビジネス・オブジェクト

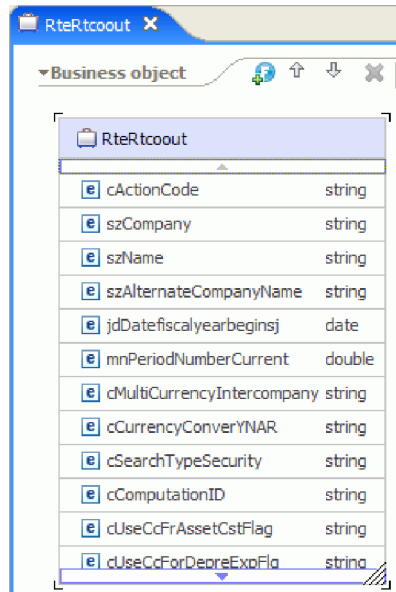


図8. 単一リアルタイム・イベントのフラット・ビジネス・オブジェクト

アダプターの構成時に、ビジネス・グラフを生成するオプションを選択することもできます。バージョン 6.0.2 では、最上位レベルの各ビジネス・オブジェクトがビジネス・グラフに含まれていますが、このビジネス・オブジェクトには、実行する操作に関する追加情報を指定するために、バージョン 6.0.2 でアプリケーションが使用できる動詞が組み込まれています。バージョン 6.1.x では、ビジネス・グラフはオプションです。ビジネス・グラフが必要になるのは、バージョン 6.1.x より前のバージョンの WebSphere Integration Developer を使用して作成したモジュールにビジネス・オブジェクトを追加する場合に限られます。ビジネス・グラフが存在する場合、ビジネス・グラフは処理されますが、動詞は無視されます。

外部サービス・ウィザード

外部サービス・ウィザードは、アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイする前に、そのアダプターを構成する場合に使用するツールです。外部サービス・ウィザードは、JD Edwards EnterpriseOne サーバー への接続を確立して、指定した検索基準に基づいてサービスをディスカバーし、ディスカバーされたサービスに基づいてビジネス・オブジェクトおよびインターフェースを生成します。

WebSphere Integration Developer を使用して、JD Edwards EnterpriseOne サーバーへの接続を確立し、JD Edwards EnterpriseOne サーバー のデータベース情報を参照します。

以下の図に示すように、サーバーへのアクセスに必要なユーザー名やパスワードなどの接続情報を指定します。

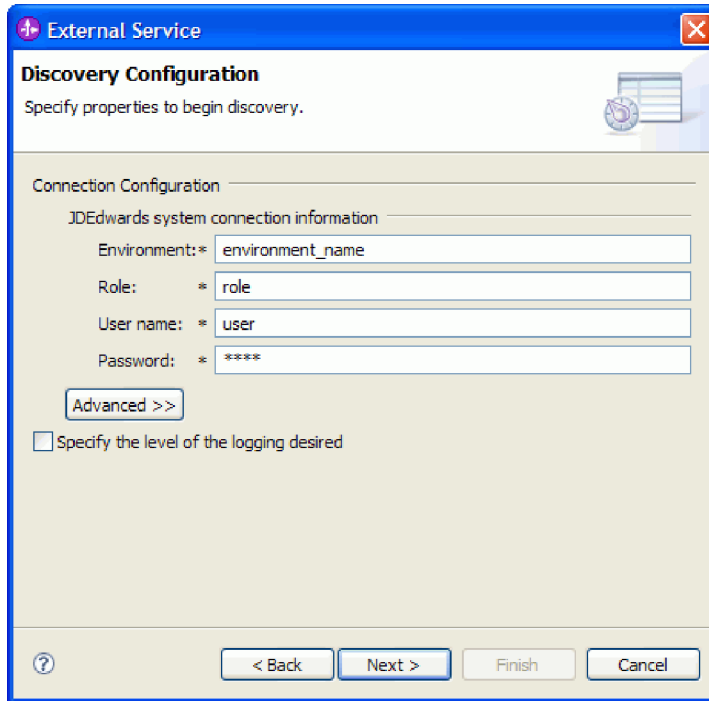


図9. 「ディスカバリー・エージェントの設定構成」ウィンドウ

外部サービス・ウィザードの実行の結果として、アダプターと共にインターフェースとビジネス・オブジェクトを含むモジュールが生成されます。このモジュールを WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイします。

例えば、外部サービス・ウィザードを実行し、BSFNSample というモジュールを作成すると、「データ・タイプ」の下に、生成されたビジネス・オブジェクトのリストが表示されます。リストには、処理中に生成された障害に関連付けられたオブジェクトが含まれる場合もあります。

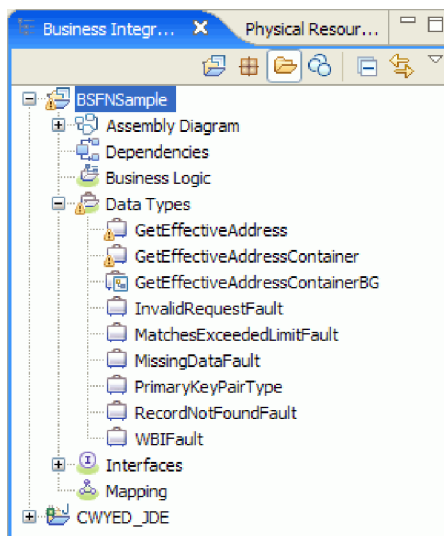


図10. 外部サービス・ウィザードにより生成された BSFNSample モジュールの例

第 2 章 アダプター実装の計画

WebSphere Adapter for JD Edwards EnterpriseOne を構成する前に、使用するデプロイメント・オプションも含めた、環境に必要なすべてのセキュリティー・オプションおよびパフォーマンス・オプションを把握する必要があります。また、前のバージョンの WebSphere Adapter for JD Edwards EnterpriseOne からマイグレーションする場合は、マイグレーション・タスクを実行してください。

作業を始める前に

アダプターのセットアップおよび使用を開始する前に、ビジネス・インテグレーションの概念、使用する統合開発ツールおよびランタイム環境の機能と要件、およびこのソリューションを構築して使用する場所となる JD Edwards EnterpriseOne 環境について十分に理解しておく必要があります。

WebSphere Adapter for JD Edwards EnterpriseOne を構成して使用するには、以下の概念、ツール、および作業に関する知識と経験が必要です。

- 構築するソリューションの業務要件。
- Service Component Architecture (SCA) プログラミング・モデルなどのビジネス・インテグレーションの概念およびモデル。
- ソリューションの作成のために使用する統合開発ツールに用意されている機能。これらのツールを使用した、モジュールの作成方法、コンポーネントのテスト方法、その他の統合作業の実行方法を理解しておく必要があります。
- 統合ソリューションのために使用するランタイム環境の機能と要件。ホスト・サーバーの構成方法と管理方法だけでなく、管理コンソールを使用して、プロパティ定義の設定と変更、接続の構成、およびイベントの管理を行う方法についても理解しておく必要があります。

セキュリティー

アダプターは、Java 2 セキュリティーの J2C 認証データ入力 (認証別名) 機能を使用して、ユーザー名およびパスワードの安全な認証機能を提供します。セキュリティー機能については詳しくは、WebSphere Process Server または WebSphere Enterprise Service Bus の資料を参照してください。

ログ・ファイルおよびトレース・ファイル内の機密性の高いユーザー・データ保護のサポート

アダプターでは、ログ・ファイルおよびトレース・ファイル内の機密性の高いデータが許可のないユーザーに表示されないようにする機能を提供しています。

アダプターのログ・ファイルおよびトレース・ファイルには、JD Edwards EnterpriseOne サーバーからのデータが含まれる場合があり、これには機密情報が含まれている可能性があります。場合によっては、これらのファイルを機密データを閲覧する許可のないユーザーが見なければならぬこともあります。例えば、サポ

ート・スペシャリストは、問題のトラブルシューティングを行うため、ログおよびトレース・ファイルを使用する必要があります。

このような場合にデータを保護するため、アダプターでは、機密性の高いユーザー・データをアダプターのログ・ファイルおよびトレース・ファイルに表示しないようにするかどうかを指定することができます。このオプションは、外部サービス・ウィザードで選択する、または、HideConfidentialTrace プロパティーを変更することで指定できます。このプロパティーを有効にすると、アダプターは、機密データを「XXX」という文字列で置き換えます。

このオプション・プロパティーの詳細については、136 ページの『Managed Connection Factory プロパティー』を参照してください。

以下の情報タイプが、潜在的に機密データとみなされ、秘匿の対象となります。

- ビジネス・オブジェクトの内容
- イベント・レコードのオブジェクト・キーの内容
- ユーザー名、パスワード、環境、およびロール
- JD Edwards EnterpriseOne サーバー への接続に使用される URL

以下の情報タイプは、ユーザー・データとはみなされず、秘匿されません。

- イベント・レコードのオブジェクト・キー部分以外のイベント・レコード内容。XID、イベント ID、ビジネス・オブジェクト名、イベント状況などがこれに該当します。
- ビジネス・オブジェクト・スキーマ
- トランザクション ID
- 呼び出しシーケンス

ユーザー認証

アダプターでは、JD Edwards EnterpriseOne サーバーへの接続に必要なユーザー名およびパスワードを指定する方法がいくつかサポートされています。それぞれの方法の特徴および制限を理解した上で、ご使用のアプリケーションにとって適切なセキュリティ・レベルであり、かつ都合のよい方法を選択してください。

アダプターをアプリケーションに統合するには、以下の場合にユーザー名およびパスワードが必要になります。

- ユーザーがアダプターでアクセスできるオブジェクトおよびサービスに関する情報を抽出、つまりディスカバリーするために外部サービス・ウィザードがJD Edwards EnterpriseOne サーバーに接続するとき。
- WebSphere Process Server または WebSphere Enterprise Service Bus での実行時に、アダプターが Outbound 要求を処理するためにJD Edwards EnterpriseOne サーバーに接続するとき。

ウィザードでの認証

外部サービス・ウィザードでは、ディスカバリー・プロセスの接続情報を要求し、その接続情報を、ランタイムに使用される接続情報を指定するアダプター・プロパティーのデフォルト値として再使用します。ウィザードの実行中に使用するユーザー

一名およびパスワードは、アプリケーションをサーバーにデプロイするときとは別のものを使用できます。別の JD Edwards EnterpriseOne サーバーに接続することもできます。ただし、2 つのデータベースのスキーマ名が同じである必要があります。例えば、Adapter for JD Edwards EnterpriseOne を使用するアプリケーションの開発および統合中は、実動データベースを使用しないことがあります。テスト・データベースを使用し、同じデータ形式で、より少ない数の模擬レコードを使用することにより、実動データベースのパフォーマンスに影響を与えることなく、また顧客データのプライバシー要件に起因する制限が生じることなく、アプリケーションを開発および統合できます。

ウィザードは、ディスカバリー・プロセス用に指定されたユーザー名およびパスワードをディスカバリー・プロセスでのみ使用します。これらは、ウィザードの完了後はアクセス不能になります。

実行時の認証

実行時、アダプターは、JD Edwards EnterpriseOne サーバーに接続するためにユーザー名およびパスワードを提供する必要があります。ユーザー介入なしに接続するためには、アダプターは保管されているユーザー情報のコピーにアクセスしなければなりません。サーバー環境では、ユーザー情報の保管方法はいくつかあります。外部サービス・ウィザードでは、アダプターが次のいずれかの方法でユーザー情報を取得するように構成できます。

- アダプター・プロパティ
- データ・ソース
- J2C 認証別名

アダプター・プロパティへのユーザー名およびパスワードの保管は、実行時にこの情報を提供するための直接的な方法です。外部サービス・ウィザードを使用してモジュールを構成するときに、このユーザー名およびパスワードを指定します。ユーザー名とパスワードを直接指定する方法は最も簡単のように見えますが、この方法には重要な制限があります。アダプター・プロパティは暗号化されません。パスワードは、サーバー上で他のユーザーがアクセスできるフィールドに平文で保管されます。さらに、パスワードが変更された場合は、アプリケーション EAR ファイルに組み込まれたアダプターおよびサーバーに個別にインストールされたアダプターを含む、その JD Edwards EnterpriseOne サーバーにアクセスするすべてのアダプター・インスタンスのパスワードを更新しなければなりません。

データ・ソースを使用すれば、別のアプリケーション用に既に確立された接続を使用することができます。例えば、複数のアプリケーションが同じユーザー名およびパスワードを使用して同じデータベースにアクセスする場合は、同じデータ・ソースを使用してそれらのアプリケーションをデプロイできます。ユーザー名およびパスワードを知るユーザーを、そのデータ・ソースにアプリケーションをデプロイする最初のユーザー、またはデータ・ソースを個別に定義する最初のユーザーのみに限定できます。

Java 2 セキュリティーの Java Authentication and Authorization Service (JAAS) 機能で作成された J2C 認証データ入力 (あるいは認証別名) を使用する方法は、堅固でセキュアなアプリケーション・デプロイ方法です。管理者は、システムにアクセスする必要がある 1 つ以上のアプリケーションで使用される認証別名を作成します。

ユーザー名およびパスワードを知るユーザーを、その管理者のみに限定できます。管理者は、変更が必要な場合は単一の場所でパスワードを変更できます。

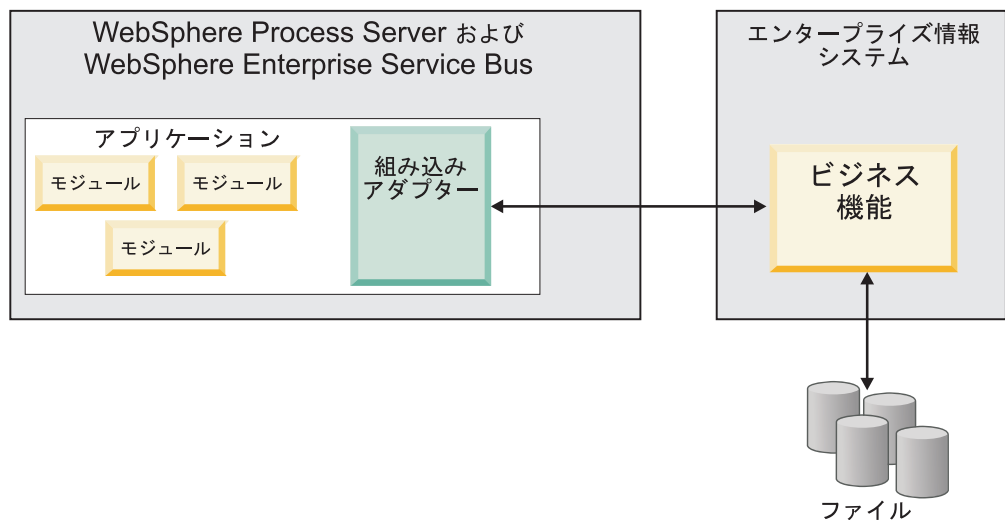
デプロイメント・オプション

アダプターをデプロイする方法は、以下の 2 つの方法があります。デプロイされたアプリケーションの一部としてアダプターを組み込むか、アダプターをスタンドアロン RAR ファイルとしてデプロイできます。ご利用の環境の要件によって、選択するデプロイメント・オプションのタイプが異なります。

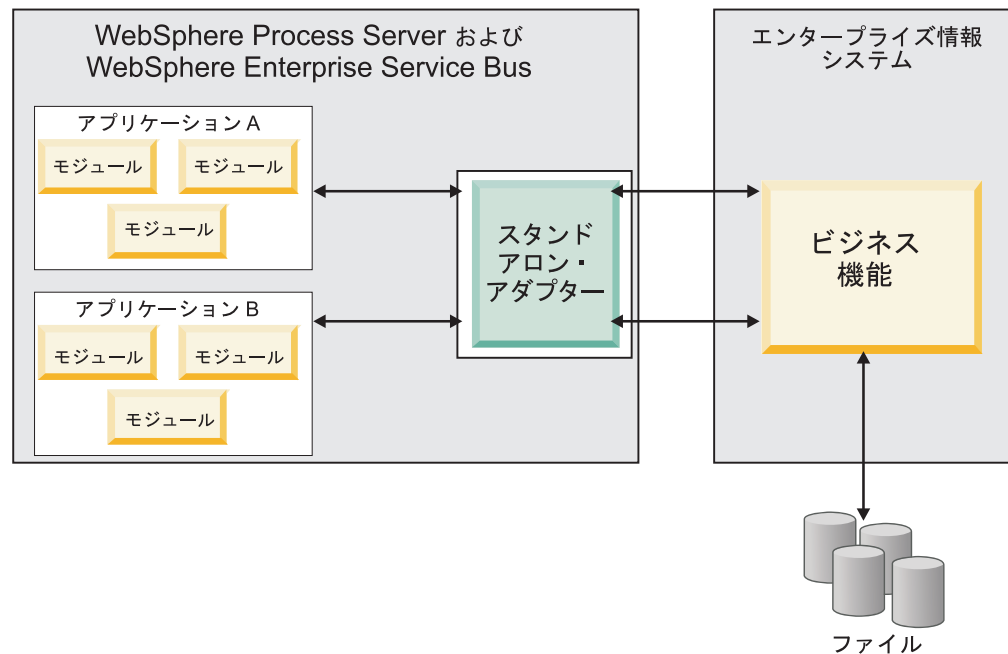
デプロイメント・オプションについて以下に説明します。

- 「**単一アプリケーションが使用するモジュールとともにデプロイする**」。アダプター・ファイルをモジュール内に組み込むと、モジュールをあらゆるアプリケーション・サーバーにデプロイすることができます。組み込みアダプターを使用するのは、組み込みアダプターを使用するモジュールが 1 つある場合か、複数のモジュールでバージョンの異なるアダプターを実行する必要がある場合です。組み込みアダプターを使用すると、他のモジュールのアダプター・バージョンを変更することで、それらのモジュールを不安定にするリスクを生じることなく、1 つのモジュール内でアダプターをアップグレードできます。
- 「**複数アプリケーションが使用するサーバー上**」。モジュール内にアダプター・ファイルを組み込まない場合は、このモジュールを実行するアプリケーション・サーバーごとにモジュールをスタンドアロン・アダプターとしてインストールする必要があります。複数のモジュールが同じバージョンのアダプターを使用可能で、アダプターを中央の場所で管理する場合は、スタンドアロン・アダプターを使用します。スタンドアロン・アダプターの場合も、複数のモジュールに対して単一のアダプター・インスタンスを実行することにより、必要なリソースが軽減されます。

エンタープライズ・アーカイブ (EAR) ファイル内には、組み込みアダプターがバンドルされています。この組み込みアダプターは、一緒にパッケージされ、デプロイされたアプリケーションでのみ使用することができます。



スタンドアロン・アダプターを表すのは、スタンドアロンのリソース・アダプター・アーカイブ (RAR) ファイルです。これは、デプロイされた後、サーバー・インスタンス内のすべてのデプロイ済みアプリケーションから使用することができます。



ご使用のアプリケーションのプロジェクトを WebSphere Integration Developer を使用して作成する場合は、アダプターのパッケージ方法 (EAR ファイルによるバンドルまたはスタンドアロン RAR ファイルとして) を選択できます。この選択に応じて、アダプターをランタイム環境で使用方法、および管理コンソールでのアダプターのプロパティの表示の仕方が異なります。

アダプターをアプリケーションに組み込むか、スタンドアロン・モジュールとしてデプロイするかのどちらを選択するかは、アダプターの管理の仕方によって決まります。アダプターの 1 つのコピーのみを保持して、アダプターのアップグレード時に複数のアプリケーションが中断してもかまわない場合は、アダプターをスタンドアロン・モジュールとしてデプロイすることが多くなります。

複数のバージョンを稼働させる計画があるため、アダプターのアップグレード時に起こる可能性のある中断により配慮する場合は、アダプターをアプリケーションに組み込むことになります。アダプターをアプリケーションに組み込む場合、アダプターのバージョンをアプリケーションのバージョンに関連付けて、単一のモジュールとして管理することができます。

アダプターのアプリケーションへの組み込みに関する考慮事項

アダプターをアプリケーションに組み込む計画がある場合は、以下の点を考慮してください。

- 組み込みアダプターには、クラス・ローダーの独立性があります。

クラス・ローダーは、アプリケーションのパッケージ化、およびランタイム環境にデプロイされたパッケージ済みアプリケーションの動作に影響を与えます。ク

ラス・ローダーの分離とは、アダプターは別のアプリケーションまたはモジュールからクラスを読み込むことができないという意味です。クラス・ローダーの分離機能により、異なるアプリケーションで、類似した名前の付いた 2 つのクラスによる相互干渉が防止されます。

- アダプターが組み込まれた各アプリケーションを、別々に管理する必要があります。

スタンドアロン・アダプターを使用する際の考慮事項

スタンドアロン・アダプターを使用する場合は、以下の点を考慮してください。

- スタンドアロン・アダプターには、クラス・ローダーの独立性がありません。

スタンドアロン・アダプターにはクラス・ローダーの分離が存在しないため、ある特定の Java 成果物の 1 つのバージョンのみが実行され、その成果物のバージョンや順序は特定されません。例えば、スタンドアロン・アダプターを使用する場合は、1 つのリソース・アダプター・バージョン、1 つのアダプター・ファウンデーション・クラス (AFC) バージョン、または 1 つのサード・パーティー JAR バージョンのみが存在します。スタンドアロン・アダプターとしてデプロイされたアダプターはすべて、単一の AFC バージョンを共有し、1 つのアダプターのすべてのインスタンスは同じコードのバージョンを共有します。1 つのサード・パーティー・ライブラリーを使用するアダプター・インスタンスはすべて、そのライブラリーを共有しなければなりません。

- これらの共有成果物のいずれかを更新する場合、その成果物を使用するすべてのアプリケーションが影響を受けることになります。

例えば、サーバー・バージョン X で動作しているアダプターを使用しているときに、クライアント・アプリケーションのバージョンをバージョン Y に更新すると、元のアプリケーションが動作しなくなることがあります。

- アダプター・ファウンデーション・クラス (AFC) には前のバージョンとの互換性がありますが、単体でデプロイされる各 RAR ファイルには、最新バージョンの AFC を入れておく必要があります。

スタンドアロン・アダプターのクラスパス内に JAR ファイルの複数のコピーがある場合、使用される JAR ファイルはランダムになります。このため、すべての JAR ファイルを最新バージョンにしておく必要があります。

クラスター化された環境の WebSphere Adapters

モジュールをクラスター化されたサーバー環境にデプロイすることで、アダプターのパフォーマンスおよび可用性を向上させることができます。スタンドアロン・アダプター、または組み込みアダプターのどちらかを使用してモジュールをデプロイする場合も、モジュールは、クラスター内のすべてのサーバー内に複製されます。

WebSphere Process Server または WebSphere Enterprise Service Bus、WebSphere Application Server Network Deployment、および WebSphere Extended Deployment では、クラスター化された環境がサポートされます。クラスターとは、ワークロードの平衡を取り、高可用性とスケラビリティを提供するために、一緒に管理されるサーバー・グループのことです。サーバー・クラスターをセットアップするときには、デプロイメント・マネージャー・プロファイルを作成してください。デプロ

イメント・マネージャーのサブコンポーネントである HAManager により、アダプター・インスタンスを活動状態にするよう Java 2 Platform, Enterprise Edition (J2EE) Connector Architecture (JCA) コンテナに通知されます。JCA コンテナにより、アダプター・インスタンスのランタイム環境が提供されます。クラスター環境の作成について詳しくは、リンク http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm_cluster_v61.html を参照してください。

必要に応じて、WebSphere Extended Deployment を使用して、クラスター環境内のアダプター・インスタンスのパフォーマンスを向上させることができます。

WebSphere Extended Deployment は、WebSphere Application Server Network Deployment で使用される静的作業負荷マネージャーの代わりに、動的作業負荷マネージャーを使用することにより、WebSphere Application Server Network Deployment の機能を拡張します。動的作業負荷マネージャーは、要求による負荷の平衡化を動的に行うことによって、クラスター内のアダプター・インスタンスのパフォーマンスを最適化できます。これは、負荷の変動に応じて、アプリケーション・サーバー・インスタンスを自動的に停止したり始動したりできることを意味します。これにより、能力や構成が異なる複数のマシンが負荷の変動に一樣に対処できるようになります。WebSphere Extended Deployment の利点について詳しくは、<http://publib.boulder.ibm.com/infocenter/wxinfo/v6r1/index.jsp> のリンクを参照してください。

クラスター化された環境では、WebSphere Adapter for JD Edwards EnterpriseOne のアダプター・インスタンスが処理できるのは Outbound 処理のみです。

Inbound 処理の高可用性

Inbound 処理は、JD Edwards EnterpriseOne サーバーのデータを更新した結果、起動するイベントに基づいています。WebSphere Adapter for JD Edwards EnterpriseOne は、イベント・テーブルをポーリングすることで更新を検出するよう構成されます。その後、アダプターはイベントをそのエンドポイントにパブリッシュします。

モジュールをクラスターにデプロイすると、JCA コンテナにより、enableHASupport リソース・アダプター・プロパティが検査されます。enableHASupport プロパティの値が真である場合 (デフォルトの設定)、すべてのアダプター・インスタンスはポリシー N のうちの 1 つを持つ HAManager に登録されます。このポリシーは、アダプター・インスタンスのうちの 1 つのみがイベントのポーリングを開始することを意味します。クラスター内のその他のアダプター・インスタンスが開始していても、それらのインスタンスは、アクティブなアダプター・インスタンスがイベントの処理を完了するまで、アクティブ・イベントに関して休止のままとなります。ポーリング・スレッドが開始しているサーバーが何らかの理由でシャットダウンした場合は、バックアップ・サーバーのいずれかで稼働しているアダプター・インスタンスが活動状態になります。

重要: enableHASupport プロパティの設定は変更しないでください。

Outbound 処理の高可用性

クラスター化された環境では、Outbound 処理要求の実行に、複数のアダプター・インスタンスが使用可能です。そのため、Outbound 要求について WebSphere Adapter

for JD Edwards EnterpriseOne と対話するアプリケーションが、ご使用の環境に複数存在する場合は、クラスター化された環境にモジュールをデプロイすることにより、パフォーマンスが向上することがあります。クラスター化された環境では、複数の Outbound 要求が同じレコードを処理しようとしないう限り、複数の Outbound 要求を同時に処理することができます。

複数の Outbound 要求が、顧客の住所などの同じレコードを処理しようとした場合、WebSphere Application Server Network Deployment のワークロード管理機能により、その要求は、受信された順に使用可能なアダプター・インスタンスの間で分配されます。このため、クラスター化された環境では、この種の Outbound 要求は、単一サーバー環境内と同じように処理されます。つまり、1 つのアダプター・インスタンスが一度に処理するのは、1 つの Outbound 要求のみです。ワークロード管理について詳しくは、リンク http://publib.boulder.ibm.com/infocenter/wasinfo/v6r1/index.jsp?topic=/com.ibm.websphere.nd.doc/info/ae/ae/trun_wlm.html を参照してください。

バージョン 6.2 へのマイグレーション

WebSphere Adapter for JD Edwards EnterpriseOne のバージョン 6.2 へのマイグレーションを行うことにより、アダプターの前のバージョンから自動的にアップグレードします。さらに、アダプターの前のバージョンを組み込んだアプリケーションをマイグレーションできるため、このアプリケーションは、バージョン 6.2 が備えている機能や処理能力を活用できます。

マイグレーションに関する考慮事項

WebSphere Adapter for JD Edwards EnterpriseOne バージョン 6.2 には、既存のアダプター・アプリケーションに影響を与える可能性のある更新が含まれています。WebSphere Adapter for JD Edwards EnterpriseOne を使用するアプリケーションのマイグレーションを行う前に、既存のアプリケーションに影響を与える可能性のある要因について考慮する必要があります。

前のバージョンとの互換性

WebSphere Adapter for JD Edwards EnterpriseOne バージョン 6.2 は、バージョン 6.0.2.x および バージョン 6.1.x のアダプターと完全互換で、カスタム・ビジネス・オブジェクト (XSD ファイル) およびデータ・バインディングを扱うことができます。

WebSphere Adapter for JD Edwards EnterpriseOne バージョン 6.2 は、バージョン 6.0.2.x およびバージョン 6.1.x と完全に互換性があるため、WebSphere Adapter for JD Edwards EnterpriseOne バージョン 6.0.2.x またはバージョン 6.1.0.x を使用していたアプリケーションは、バージョン 6.2 へのアップグレード後も変わらず稼働します。ただし、バージョン 6.2 のアダプターの機能をアプリケーションで使用する場合は、マイグレーション・ウィザードを実行してください。

マイグレーション・ウィザードは、バージョン 6.0.2.x またはバージョン 6.1.x のアダプターをバージョン 6.2 で置き換え (アップグレードし)、バージョン 6.2 のフィチャーと機能をアプリケーションで使用できるようにします。

注: マイグレーション・ウィザードは、バージョン 6.2 のアダプターで処理するためのマッパーやメディエーターなどのコンポーネントを新規に作成したり、既存のコンポーネントを変更したりすることはありません。バージョン 6.2 にアップグレードする場合で、アプリケーションにバージョン 6.1.x 以前のアダプターが組み込まれている際に、バージョン 6.2 の機能をアプリケーションで利用する場合は、これらのアプリケーションを変更しなければならない場合があります。

単一モジュール内でのバージョン管理に関して成果物の整合性が取れていない場合、このモジュール全体にマークが付けられ、マイグレーションの対象として選択可能になりません。バージョンの不整合はプロジェクトの破損を示す可能性があるため、ワークスペースのログに記録されます。

WebSphere Integration Developerのバージョンが、WebSphere Adapter for JD Edwards EnterpriseOne のバージョンに一致しない場合は、これらの製品を同時に実行することはできません。

以下の製品の組み合わせが使用される場合、外部サービス・ウィザードは実行できません。

表 1. サポートされない製品の組み合わせ

WebSphere Integration Developer	WebSphere Adapter for JD Edwards EnterpriseOne
バージョン 6.2	バージョン 6.1.x
バージョン 6.1.x	バージョン 6.0.2.x
バージョン 6.0.2.x	バージョン 6.2
バージョン 6.0.2.x	バージョン 6.1.x

アップグレードかアップグレード後にマイグレーションかの決定

デフォルトでは、マイグレーション・ウィザードは、アダプターのアップグレードを実行してから、アプリケーション成果物をマイグレーションし、アプリケーションがバージョン 6.2 のアダプターの機能を使用できるようにします。プロジェクトを選択してアダプターをアップグレードするようにすると、それに関連付けられている成果物がマイグレーションの対象としてウィザードにより自動的に選択されます。

アダプターをバージョン 6.0.2.x またはバージョン 6.1.x からバージョン 6.2 にアップグレードする場合で、アダプター成果物をマイグレーションしない場合は、マイグレーション・ウィザードの該当する領域でアダプター成果物の選択を解除してください。

アダプター成果物を選択しないでマイグレーション・ウィザードを実行すると、アダプターのインストールおよびアップグレードが行われますが、成果物はマイグレーションされないため、アプリケーションはバージョン 6.2 のアダプターの機能を利用できません。

テスト環境でのマイグレーション・ウィザードの初期実行

アダプターのマイグレーションでは、WebSphere Adapter for JD Edwards EnterpriseOne のバージョン 6.2 を使用するアプリケーションを変更しなければなら

ない場合があるため、アプリケーションを実稼働環境にデプロイする前に、まずマイグレーションを開発環境で実行して、アプリケーションをテストする必要があります。

マイグレーション・ウィザードは、開発環境に完全に統合されています。

非推奨機能

非推奨機能は、サポートされてはいても、使用することが推奨されず、廃止される可能性がある機能です。WebSphere Adapter for JD Edwards EnterpriseOne の以前のバージョンの機能のうち、バージョン 6.1.x で非推奨になった機能を以下にまとめます。

- バージョン 6.1.x では、タイムアウト・プロパティーが、リソース・アダプター・プロパティーから Managed Connection Factory (J2C) プロパティーに移りました。タイムアウト・プロパティーをリソース・アダプター・プロパティーとして設定して生成された成果物 (バージョン 6.0.2.x) の後方互換性を確保するため、Adapter for JD Edwards EnterpriseOne バージョン 6.1.x では、次のように動作します。
 1. アダプターは、Managed Connection Factory (J2C) プロパティーでタイムアウト・プロパティー設定を検索します。
 2. タイムアウト・プロパティーが Managed Connection Factory (J2C) プロパティーとして設定されていない場合、アダプターは、リソース・アダプター・プロパティーでタイムアウト・プロパティーを検索します。
 3. タイムアウト・プロパティーが Managed Connection Factory プロパティー、リソース・アダプター・プロパティーのいずれとして設定されていない場合、アダプターは、30 秒のタイムアウト・プロパティーのデフォルト設定を割り当てます。

タイムアウト・プロパティーの詳細については、136 ページの『Managed Connection Factory プロパティー』を参照してください。

マイグレーションの実行

プロジェクトまたは EAR ファイルは、アダプター・マイグレーション・ウィザードを使用すれば、バージョン 6.2 にマイグレーションできます。ツールが終了したらマイグレーションは完了するため、プロジェクトで作業したり、モジュールをデプロイしたりできます。

始める前に

『マイグレーションに関する考慮事項』の情報を見直します。

このタスクを実行する理由および時期

WebSphere Integration Developer でマイグレーションを実行するには、以下のステップを完了してください。

注: マイグレーションが完了すると、このモジュールは以前のバージョンの WebSphere Process Server または WebSphere Enterprise Service Bus、または WebSphere Integration Developer とは互換性がなくなります。

注: 以下の手順では、WebSphere Integration Developer の Java EE パースペクティブでコネクタ・プロジェクトのコンテキスト・メニューからアダプター・マイグレーション・ウィザードを実行する方法について説明します。

このタスクの手順

1. 既存のプロジェクトの場合は PI (プロジェクト交換) ファイルを、デプロイ済みアプリケーションの場合は EAR (エンタープライズ・アーカイブ) ファイルを、それぞれワークスペースにインポートします。
2. Java EE パースペクティブに切り替えます。
3. モジュールを右クリックし、「更新」 → 「コネクタ・プロジェクトの更新」を選択します。

注: また、以下のようにしてアダプター・マイグレーション・ウィザードを起動できます。

- Java EE パースペクティブでプロジェクトを右クリックし、「更新」 → 「アダプター成果物のマイグレーション」をクリックします。
- 問題ビューで、マイグレーション固有のメッセージを右クリックし、「クイック・フィックス」を選択して問題を解消します。

Java EE パースペクティブでコネクタ・プロジェクト・コンテキスト・メニューからマイグレーション・ウィザードを起動した場合、デフォルトでは、すべての依存関係のある成果物プロジェクトが選択されます。依存関係のある成果物プロジェクトを選択解除すると、そのプロジェクトはマイグレーションされません。選択解除したプロジェクトは、後からマイグレーションすることができます。以前にマイグレーション済みのプロジェクト、現行バージョンのプロジェクト、エラーのあるプロジェクトはマイグレーションの対象外であり、選択されません。

4. 「ようこそ」ページに表示されたタスクおよび警告を確認して、「次へ」をクリックします。
5. 警告ウィンドウが開き、「このバージョンのターゲット・アダプターでサポートされないプロパティは、マイグレーション中に除去されます。」というメッセージが表示されます。「OK」をクリックし、次のステップに進みます。
6. 「変更内容の確認」ウィンドウで、+ 記号をクリックすれば、マイグレーションする各成果物で行われるマイグレーションの変更内容を確認することができます。
7. 「終了」をクリックして、マイグレーションを実行します。

ウィザードでは、マイグレーション・プロセスの実行前に、マイグレーションによって影響を受けるすべてのプロジェクトがバックアップされます。プロジェクトのバックアップ先は、ワークスペース内の一時フォルダーになります。何らかの理由で万が一マイグレーションが失敗した場合、または、マイグレーションの完了前にマイグレーションを取り消す場合、ウィザードでは、変更したプロジェクトを削除し、この一時フォルダーに保管されていたプロジェクトを復元します。

マイグレーションが正常に完了すると、バックアップされていたすべてのプロジェクトが削除されます。

8. 問題ビューを参照して、マイグレーション・ウィザードからのメッセージ (先頭に CWPAD というストリングがあるメッセージ) の有無を確認します。

9. EAR ファイルをマイグレーションしている場合は、マイグレーション済みアダプターおよび成果物のある新規の EAR ファイルを作成して、WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイすることもできます。EAR ファイルのエクスポートおよびデプロイについて詳しくは、この資料で EAR ファイルについて説明しているトピックを参照してください。

結果

プロジェクトまたは EAR ファイルは、バージョン 6.2 へマイグレーションされます。アダプター・マイグレーション・ウィザードの終了後に外部サービス・ウィザードを実行する必要はありません。

マイグレーションしない場合のバージョン 6.0.2.x またはバージョン 6.1.x プロジェクトの更新

アダプターをバージョン 6.0.2.x またはバージョン 6.1.x からバージョン 6.2 にアップグレードする一方で、アダプター・プロジェクトの成果物をマイグレーションしない選択を行うことができます。

このタスクを実行する理由および時期

アダプターの内部名はバージョン 6.2 で変更されたため、WebSphere Integration Developerバージョン 6.2 でアダプター・ウィザードを使用するには、その前にバージョン 6.0.2.x またはバージョン 6.1.x プロジェクトの成果物を更新して、新しい名前を使用する必要があります。バージョン 6.0.2.x またはバージョン 6.1.x プロジェクトを更新するには、マイグレーション・ウィザードを使用します。次に、WebSphere Integration Developer のクイック・フィックス機能を使用して、プロジェクト成果物内のアダプター名を変更します。

このタスクの手順

1. プロジェクト交換 (PI) ファイルをワークスペースにインポートします。
2. Java EE パースペクティブで、プロジェクト名を右クリックして、「更新」 → 「コネクター・プロジェクトの更新」をクリックします。アダプター・マイグレーション・ウィザードが開きます。
3. 「ようこそ」ページで、「次へ」をクリックします。
4. 「プロジェクトの選択」ウィンドウで、依存関係のある成果物プロジェクトの選択を解除し、「終了」をクリックします。
5. 警告ウィンドウが開き、「このバージョンのターゲット・アダプターでサポートされないプロパティは、マイグレーション中に除去されます。」というメッセージが表示されます。「OK」をクリックし、次のステップに進みます。
6. 「クイック・フィックス」ウィンドウで、「参照しているアダプターの名前を変更する (Rename the referenced adapter)」というフィックスが選択されていることを確認し、「OK」をクリックします。
7. エラーが表示されたままの場合は、「プロジェクト」 → 「クリーン」をクリックし、更新直後のプロジェクトを選択して「OK」をクリックします。

結果

これで、プロジェクトを WebSphere Adapter for JD Edwards EnterpriseOne バージョン 6.2 で使用できるようになりました。

第 3 章 サンプルおよびチュートリアル

ユーザーが、WebSphere Adapters を円滑に使用できるようにするため、「Business Process Management Samples and Tutorials」Web サイトからサンプルおよびチュートリアルが利用できます。

サンプルおよびチュートリアルには、以下のいずれかの方法でアクセスできます。

- WebSphere Integration Developer を始動すると表示される「ようこそ」ページ。
WebSphere Adapter for JD Edwards EnterpriseOne のサンプルおよびチュートリアルを表示するには、「取得」をクリックします。表示されたカテゴリーをブラウザして、選択を行います。
- Web 上の以下のロケーション: Business Process Management Samples and Tutorials <http://publib.boulder.ibm.com/bpcsamp/index.html>

第 4 章 デプロイメントのためのモジュールの構成

アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus 上にデプロイできるように構成するには、WebSphere Integration Developer を使用して、アダプターをデプロイするときに EAR ファイルとしてエクスポートされるモジュールを作成します。次に、ディスカバーの対象となるビジネス・オブジェクトと、そのディスカバーを行うシステムを指定します。

モジュールの構成のためのロードマップ

ランタイム環境で WebSphere Adapter for JD Edwards EnterpriseOne を使用できるようにするには、まずモジュールを構成する必要があります。このタスクの概要を理解すれば、タスクを達成するのに必要な手順を実行できるようになります。

WebSphere Integration Developer を使ってアダプターのモジュールを構成して、使用できるようにします。以下の図は、構成作業の流れを示しています。また、図の後に示す手順で、この作業の概要を説明します。これらの各ステップの実行方法の詳細については、このロードマップの後に記載するトピックを参照してください。

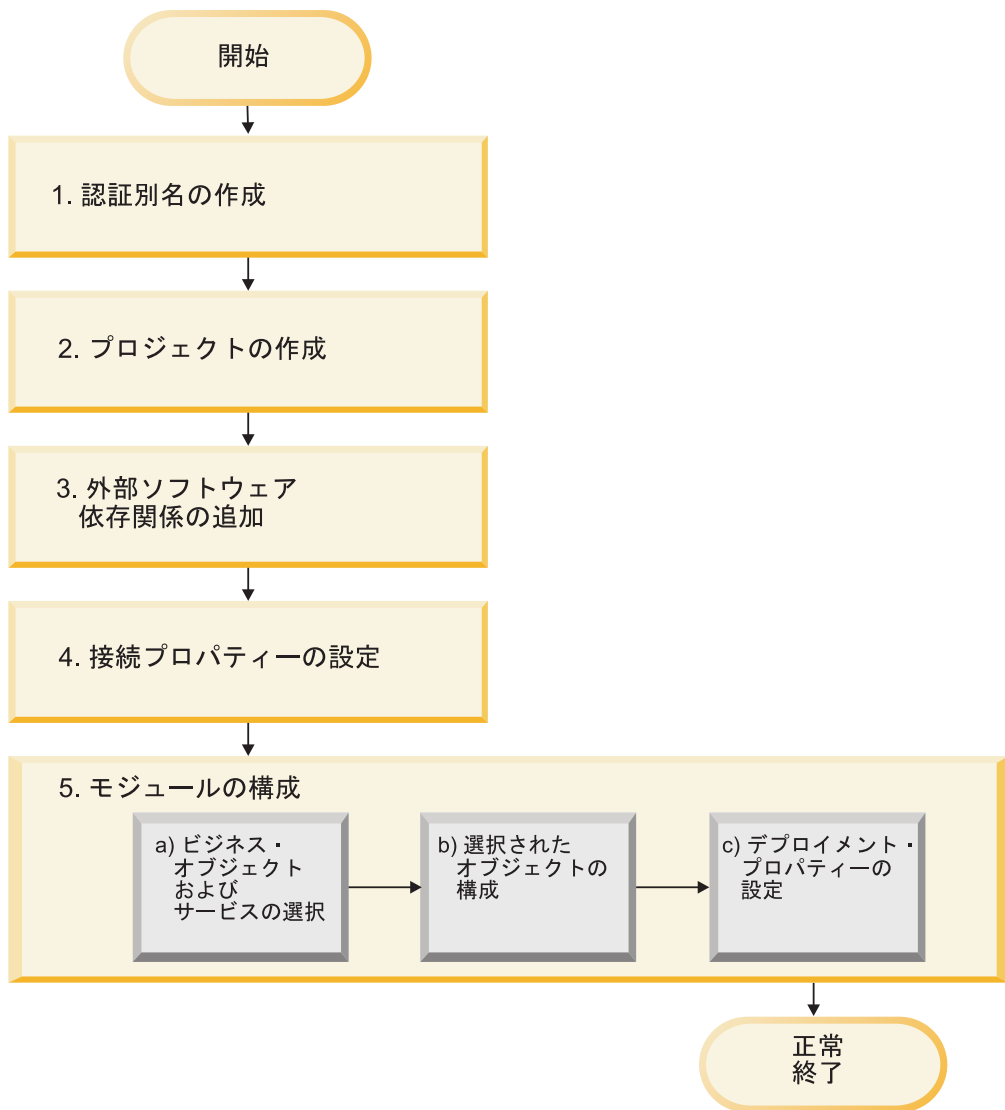


図 11. モジュールの構成のためのロードマップ

デプロイメント用のモジュールの構成

この作業は、次のステップから成ります。

1. 暗号化パスワードで JD Edwards EnterpriseOne サーバーにアクセスするための認証別名を作成します。この手順は、オプションです。パスワードおよび ID の扱いに関するポリシーに応じて実行してください。サーバーを使用して、この手順を実行してください。
2. プロジェクトを作成します。最初に外部サービス・ウィザードを WebSphere Integration Developer で始動して、モジュールの作成およびデプロイのプロセスを開始してください。このウィザードによって、プロジェクトが作成されます。これは、モジュールに関連付けられたファイルを編成するために使用されます。
3. WebSphere Adapter for JD Edwards EnterpriseOne で必要となる外部ソフトウェア依存関係を、プロジェクトに追加します。モジュールを EAR ファイルとしてエクスポートし、その EAR ファイルをサーバーにデプロイする際にも、これらの依存性が必要になります。

4. 接続プロパティを設定します。これは、外部サービス・ウィザードが JD Edwards EnterpriseOne サーバー と接続して、オブジェクトおよびサービスをディスカバリーする際に必要となります。
5. Inbound 処理または Outbound 処理用のモジュールを外部サービス・ウィザードを使って構成し、ビジネス・オブジェクトおよびサービスを JD Edwards EnterpriseOne サーバーから探して選択し、ビジネス・オブジェクト定義および関連成果物を生成します。
 - a. ビジネス・オブジェクトおよびサービス (Inbound 処理または Outbound 処理用) を、外部サービス・ウィザードによってディスカバリーされたビジネス・インテグレーション・コンポーネントの中から選択します。
 - b. ビジネス・オブジェクトすべてに適用される操作およびその他のプロパティを指定して、選択したオブジェクトを構成します。
 - c. デプロイメント・プロパティを設定します。アダプターは、実行時にこれを使用して、JD Edwards EnterpriseOne サーバー に接続します。次に、サービスを生成します。外部サービス・ウィザードを使用して、新規モジュールを保存してください。ここには、構成済みのビジネス・オブジェクト、インポート・ファイルまたはエクスポート・ファイル、およびサービス・インターフェースが含まれています。

認証別名の作成

認証別名は、アダプターが JD Edwards EnterpriseOne サーバーへのアクセスに使用するパスワードを暗号化する機能です。アダプターは、アダプター・プロパティに保管されたユーザー ID とパスワードを使用する代わりに、この認証別名を使用して JD Edwards EnterpriseOne サーバー に接続することができます。

始める前に

認証別名を作成するには、管理コンソール へのアクセス権が必要です。以下の手順は、WebSphere Integration Developer を介して管理コンソールへのアクセス権を取得する方法を示しています。

以下の手順は、WebSphere Integration Developer を介して 管理コンソール のアクセス権を取得する方法を示しています。管理コンソール を (WebSphere Integration Developer を介さずに) 直接使用する場合は、管理コンソール にログインして、ステップ 2 (34 ページ) に進みます。

このタスクを実行する理由および時期

認証別名を使用すると、アダプター構成プロパティに平文でパスワードを保管する (この場合、他人が表示できる可能性があります) 必要がなくなります。

認証別名を作成するには、以下の手順に従います。

このタスクの手順

1. 管理コンソール を開始します。

WebSphere Integration Developer によって 管理コンソール を開始するには、以下の手順を実行します。

- a. WebSphere Integration Developer の Business Integration パースペクティブで、「サーバー」タブをクリックします。
 - b. サーバーで「開始済み」という状況が表示されない場合は、サーバーの名前(例えば、「WebSphere Process V6.2 Server」)を右クリックして、「開始」をクリックします。
 - c. サーバーの名前を右クリックし、「管理コンソールの実行」をクリックします。
 - d. 管理コンソール にログオンします。管理コンソール にユーザー ID およびパスワードが必要な場合は、ID およびパスワードを入力して、「ログイン」をクリックします。ユーザー ID およびパスワードが不要な場合は、「ログイン」をクリックします。
2. 管理コンソール で、「セキュリティ」 → 「管理、アプリケーション、およびインフラストラクチャーの保護」をクリックします。
 3. 「認証」の下で、「Java 認証・承認サービス」を展開し、「J2C 認証データ」をクリックします。
 4. 認証別名を作成します。
 - a. 表示された J2C 認証別名のリストで、「新規作成」をクリックします。
 - b. 「構成」タブで、「別名」フィールドに認証別名の名前を入力します。
 - c. JD Edwards EnterpriseOne サーバーへの接続の確立に必要なユーザー ID およびパスワードを入力します。
 - d. オプション: 別名の説明を入力します。
 - e. 「OK」をクリックします。

新規に作成された別名が表示されます。

別名のフルネームは、ノード名および指定した認証別名で構成されます。例えば、ノード widNode に ProductionServerAlias という名前で別名を作成する場合、フルネームは、widNode/ProductionServerAlias となります。このフルネームは、後続の構成ウィンドウで使用する名前です。

- f. 「保管」をクリックした後、再度「保管」をクリックします。
5. 「新規作成」をクリックします。

結果

アダプター・プロパティを構成する際に使用する認証別名が作成されました。

プロジェクトの作成

モジュールの作成とデプロイのプロセスを開始するには、WebSphere Integration Developer の外部サービス・ウィザードを開始します。このウィザードによって、プロジェクトが作成されます。これは、モジュールに関連付けられたファイルを編成するために使用されます。

始める前に

JD Edwards EnterpriseOne サーバーへの接続の確立に必要な情報を収集済みであることを確認します。例えば、JD Edwards EnterpriseOne サーバー 上の JD Edwards

EnterpriseOne 環境のロール名および環境名が必要となります。また、JD Edwards EnterpriseOne サーバー にアクセスするためのユーザー ID とパスワードも必要です。

このタスクを実行する理由および時期

WebSphere Integration Developer で、外部サービス・ウィザードを開始し、アダプター用のプロジェクトを作成します。既存プロジェクトが存在する場合、ウィザードで新規に作成せずに、そのプロジェクトを選択することができます。

外部サービス・ウィザードを開始し、プロジェクトを作成するには、次の手順を実行します。

このタスクの手順

1. 外部サービス・ウィザードを開始するには、WebSphere Integration Developer の Business Integration パースペクティブに進み、「ファイル」→「新規」→「外部サービス」の順にクリックします。
2. 「新規の外部サービス (New external service)」ウィンドウで、「アダプター (Adapters)」が選択済みであることを確認して、「次へ」をクリックします。

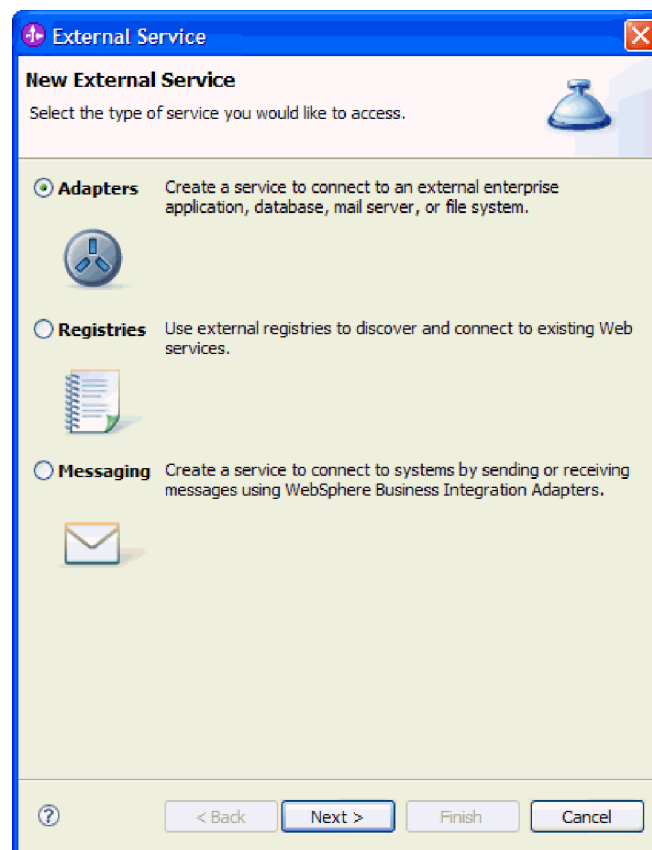


図 12. 「新規の外部サービス (New external service)」ウィンドウ

3. 「アダプターを選択 (Select an Adapter)」ウィンドウから、プロジェクトを作成するか、または既存プロジェクトを選択できます。

- 新規のプロジェクトを作成するには、以下の手順を実行します。
 - a. 次の図に示されているように、「**IBM WebSphere Adapter for JD Edwards EnterpriseOne (IBM : 6.2.0)**」を選択します。

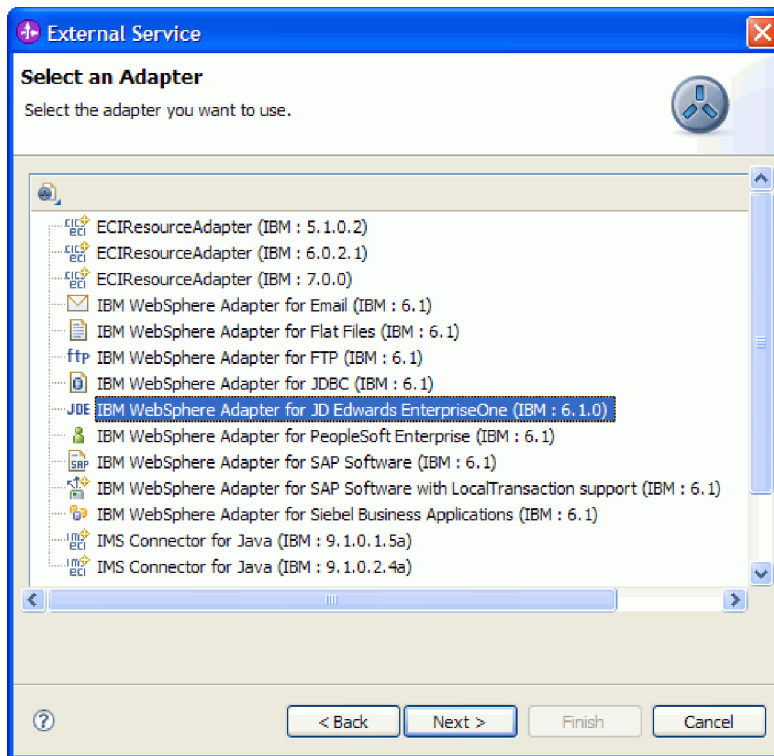


図 13. 「アダプターを選択 (Select an Adapter)」ウィンドウ

- b. 「次へ」をクリックします。
 - c. 「アダプターのインポート」ウィンドウで、プロジェクトのデフォルト名 (CWYED_JDE) を受け入れるか、「コネクター・プロジェクト」フィールドに別の名前を指定し、サーバーを「ターゲット・ランタイム」リストから選択して (例: **WebSphere Process Server v6.2**)、「次へ」をクリックします。
- 既存のプロジェクトを選択するには、以下の手順を実行します。
 - a. 「**IBM WebSphere Adapter for JD Edwards EnterpriseOne (IBM : 6.2.0)**」を展開します。
 - b. プロジェクトを選択します。

例えば、CWYED_JDE という名前の既存のプロジェクトがある場合、「**IBM WebSphere Adapter for JD Edwards (IBM : 6.2.0)**」「**CWYED_JDE**」と展開し、このプロジェクトの作成に使用した接続情報を選択することができます。
 - c. 「次へ」をクリックします。

結果

- 新規プロジェクトを作成した場合は、プロジェクトが作成され、「ビジネス・インテグレーション」ウィンドウにリストされます。

- 既存のプロジェクトを選択した場合は、プロジェクトが選択された状態になります。

次のタスク

- 新規のプロジェクトを作成した場合は、外部サービス・ウィザードから、JD Edwards EnterpriseOne サーバー への接続に必要な外部ソフトウェア依存関係ファイルの場所の指定を求めるプロンプトが出されます。
- 既存のプロジェクトを選択した場合で、そのプロジェクトに必須外部ソフトウェア依存関係ファイルが関連付けられていない場合は、外部サービス・ウィザードから、JD Edwards EnterpriseOne サーバー への接続に必要な外部ソフトウェア依存関係ファイルの場所の指定を求めるプロンプトが出されます。
- 既存のプロジェクトを選択した場合で、そのプロジェクトに必須外部ソフトウェア依存関係ファイルが既に関連付けられている場合は、外部サービス・ウィザードから、外部サービス・ウィザードと JD Edwards EnterpriseOne サーバー との接続を確立するために必要な情報の指定を求めるプロンプトが出されます。

外部ソフトウェア依存関係の追加

JD Edwards EnterpriseOne アプリケーションでは、プロジェクトに外部ソフトウェア依存関係を追加する必要があります。これらのソフトウェア依存関係によって、外部サービス・ウィザードが JD Edwards EnterpriseOne 環境と通信できるようになります。

始める前に

プロジェクトを作成するか、既存のプロジェクトを選択します。

このタスクを実行する理由および時期

必要なソフトウェア依存関係ファイルを取得し、その場所を指定するには、次の手順を実行します。

このタスクの手順

1. JD Edwards EnterpriseOne ソフトウェア依存関係ファイルを JD Edwards EnterpriseOne 管理者から入手します。必要なファイルは次の表にリストされています。

注: ソフトウェア依存関係は、使用する JD Edwards EnterpriseOne Tools のバージョンによって異なります。

表 2. JD Edwards EnterpriseOne Tools で必要とされる外部ソフトウェア依存関係ファイル

JD Edwards EnterpriseOne Tools、バージョン 8.9 (SP1、SP2)、 8.93	JD Edwards EnterpriseOne Tools、バージョン 8.94	JD Edwards EnterpriseOne Tools、バージョン 8.95、 8.96
connector.jar	Common_Jar.jar	ApplicationAPIs_JAR.jar
database.jar	Connector.jar	ApplicationLogic_JAR.jar
jdeinterop.ini	database.jar	Base_JAR.jar
jdeLog.properties	EventProcessor_EJB.jar	BizLogicContainer_JAR.jar
kernel.jar	jdeutil.jar	BizLogicContainerClient_JAR.jar

表 2. JD Edwards EnterpriseOne Tools で必要とされる外部ソフトウェア依存関係ファイル (続き)

JD Edwards EnterpriseOne Tools、バージョン 8.9 (SP1、SP2)、 8.93	JD Edwards EnterpriseOne Tools、バージョン 8.94	JD Edwards EnterpriseOne Tools、バージョン 8.95、 8.96
log4j.jar	jdbj.ini	bootstrap.jar
owra.jar	jdeinterop.ini	castor.jar
xalan.jar	jdelog.properties	Connector.jar
xerces.jar	kernel.jar	ecutils.jar
JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar 	log4j.jar	EventProcessor_JAR.jar
	xalan.jar	EventProcessor_EJB.jar
	xerces.jar	jdbj.ini
	JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar 	JdbjBase_JAR.jar
		JdbjInterfaces_JAR.jar
		jdeinterop.ini
		jdelog.properties
		JdeNet_JAR.jar
		Improxy.jar
		log4j.jar
		messagingClient.jar
		naming.jar
		PMApi_JAR.jar
		Spec_JAR.jar
		System_JAR.jar
		urlprotocols.jar
		xalan.jar
		xerces.jar

表 2. JD Edwards EnterpriseOne Tools で必要とされる外部ソフトウェア依存関係ファイル (続き)

JD Edwards EnterpriseOne Tools、バージョン 8.9 (SP1、SP2)、 8.93	JD Edwards EnterpriseOne Tools、バージョン 8.94	JD Edwards EnterpriseOne Tools、バージョン 8.95、 8.96
		JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar

- 外部依存関係ファイルを一時ロケーションにコピーします。例えば、C:\temp\JDE_dependencies にコピーします。

ヒント: JDBC ドライバー・ファイルを独自のフォルダーに入れると、それらのファイルを外部サービス・ウィザードから参照しやすくなります。例えば、Oracle データベース・サーバーを使用している場合、tnsnames.ora および ojdbc14.jar ファイルを「C:\temp\JDE_dependencies\jdbc_driver」に入れます。

- 「必要なファイルおよびライブラリー」ウィンドウで、次の手順で外部依存関係ファイルをプロジェクトに追加します。次の図は、JD Edwards EnterpriseOne Tools バージョン 8.96 用にすべての必要なファイルを選択した状態の「必要なファイルおよびライブラリー」ウィンドウを示しています。

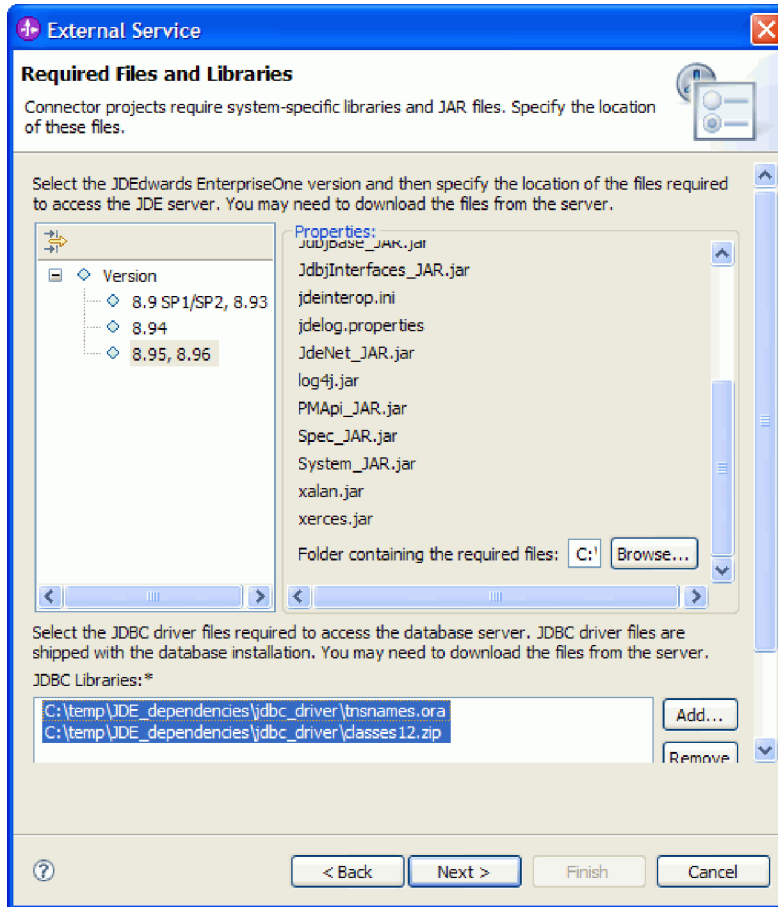


図 14. JD Edwards EnterpriseOne Tools バージョン 8.96 用に必要なファイルを選択した状態の「必要なファイルおよびライブラリー」ウィンドウ

- a. 左側ペインから、JD Edwards EnterpriseOne Tools のバージョンを選択します。選択したバージョンに必要なファイルが右側ペインに表示されます。
 - b. 必要な JAR、INI、および PROPERTIES ファイルの場所を確認し、それらを選択するには、「参照」をクリックし、リストされたファイルを含むフォルダーにナビゲートし、「OK」をクリックします。
 - c. JDBC ドライバー・ファイルを追加するには、「追加」をクリックして、JDBC ドライバー・ファイルの場所にナビゲートし、tnsnames.ora と ojdbc14.jar ファイルを選択してから、「OK」をクリックします。
4. 「次へ」をクリックします。

結果

アダプターが、JD Edwards EnterpriseOne サーバー との通信に必要なファイル名で構成されました。

一部の外部ソフトウェア依存関係ファイルの接続情報を確認または編集し、外部サービス・ウィザードが確実に JD Edwards EnterpriseOne サーバー と通信できるようにします。

外部依存関係ファイルの編集

一部の JD Edwards EnterpriseOne の外部依存関係ファイルには、IBM i サーバー WebSphere Integration Developer ワークスペースの場所、JD Edwards EnterpriseOne サーバー 名およびポート番号などの編集可能な情報が含まれています。外部サービス・ウィザードと JD Edwards EnterpriseOne サーバー との間の接続を確立するためには、このような情報をあらかじめ編集する必要がある場合があります

始める前に

外部依存関係ファイルをプロジェクトに追加したこと、および WebSphere Integration Developer ワークスペースの場所を知っていることを確認してください。

このタスクを実行する理由および時期

外部依存関係ファイルの情報が、WebSphere Integration Developer ワークスペースの場所、および JD Edwards EnterpriseOne 環境に必要な接続情報を正しく反映していることを確認します。そのためには、以下の手順を使用します。

外部ソフトウェア依存関係ファイルの構成に関する詳細については、ご使用のバージョンの JD Edwards EnterpriseOne の *JD Edwards EnterpriseOne Tools Connectors* の資料を参照してください。

このタスクの手順

1. 外部サービス・ウィザードが開いている場合、「取り消し」をクリックしてこれを閉じます。
2. 「ビジネス・インテグレーション」ペインで、プロジェクト名を展開し、続けて「**connecorModule**」を展開します。
3. `jdbj.ini` ファイル内の情報を確認します。
 - a. 「**jdbj.ini**」をダブルクリックして、これを開きます。ファイルがメモ帳に開きます。
 - b. **Ctrl+F** を押して、ファイル内を検索します。
 - c. 「検索する文字列」フィールドに、「`tns`」と入力します。
 - d. `tnsnames.ora` ファイルについてリストされている場所が、このプロジェクトの WID ワークスペースの正しい場所であることを確認します。例:
`tns=C:\IBM\wid6.1\workspace\CWYED_JDE\connectorModule\tnsnames.ora`
 - e. 「ファイル」 → 「保管」とクリックし、ファイルに対して行った変更を保存します。
4. `jdeinterop.ini` ファイルの情報を確認します。
 - a. 「**jdeinterop.ini**」をダブルクリックして、これを開きます。ファイルがメモ帳に開きます。
 - b. サーバー名とポート番号が正しいことを確認します。この情報は、JD Edwards EnterpriseOne 管理者に問い合わせれば入手できます。
 - c. 「ファイル」 → 「保管」とクリックし、ファイルに対して行った変更を保存します。
5. `jdelog.properties` ファイル内の情報を確認します。

- a. 「**jdelog.properties**」をダブルクリックして、これを開きます。ファイルは、WebSphere Integration Developer の右ペインに開きます。
- b. このファイル内の情報が正しいことを確認します。このファイルは、ログ・レベルとログ・ファイルの場所を指定しています。この情報は、JD Edwards EnterpriseOne 管理者に問い合わせれば入手できます。
- c. 「ファイル」 → 「保管」とクリックし、ファイルに対して行った変更を保存します。

結果

JD Edwards EnterpriseOne サーバー との接続に必要なファイルが、ウィザードに設定されたこととなります。

「ディスカバリー構成」ウィンドウで、外部サービス・ウィザードが JD Edwards EnterpriseOne 環境との通信に必要な接続プロパティを指定します。

外部サービス・ウィザードの接続プロパティの設定

外部サービス・ウィザードによる JD Edwards EnterpriseOne サーバー へのアクセスが可能になるように、このウィザードに接続プロパティを設定するため、サーバーへのアクセスに使用するユーザー名とパスワード、および JD Edwards EnterpriseOne 環境で必要とされる環境名およびロール名などの情報を指定します。

始める前に

外部ソフトウェア依存関係ファイルを正常に追加したこと、および依存関係ファイルの接続情報を編集したことを確認してください。

このタスクを実行する理由および時期

外部サービス・ウィザードが、JD Edwards EnterpriseOne 環境に接続し、ビジネス・オブジェクトおよびサービスをディスカバーする際に必要な接続プロパティを指定します。

注: オプションで、接続プロパティを設定するウィンドウと同じ「外部サービス・ウィザード」ウィンドウで、双方向プロパティおよびロギング・プロパティを設定できます。

必要な接続プロパティ、およびオプションの双方向プロパティ、ロギング・プロパティを指定するには、以下の手順を使用します。

このタスクの手順

1. 「ディスカバリー構成 (Discovery Configuration)」ウィンドウで、以下のようにして構成プロパティを指定します。
 - a. 「環境」フィールドで、JD Edwards EnterpriseOne 環境の名前を入力します。
 - b. 「ロール」フィールドで、JD Edwards EnterpriseOne 環境にアクセスする場合に使用するロール名を入力します。
 - c. 「ユーザー名」フィールドで、JD Edwards EnterpriseOne サーバーにアクセスするために必要なユーザー名を入力します。

- d. 「パスワード」フィールドで、JD Edwards EnterpriseOne サーバーにアクセスするために使用するパスワードを入力します。
2. オプション: 双方向プロパティを設定する必要がある場合は、以下の手順を実行します。
 - a. 「**拡張**」をクリックします。
 - b. 「**BiDi プロパティ**」セクションで、「**BiDi 変換**」を選択します。
 - c. ご使用の環境のプロパティを設定します。これらのプロパティの詳細については、128 ページの『外部サービス・ウィザードの接続プロパティ』を参照してください。

以下の図は、「拡張」ボタンと「BiDi プロパティ」ボックスが選択されている「ディスカバリー構成 (Discovery Configuration)」ウィンドウの例を示しています。

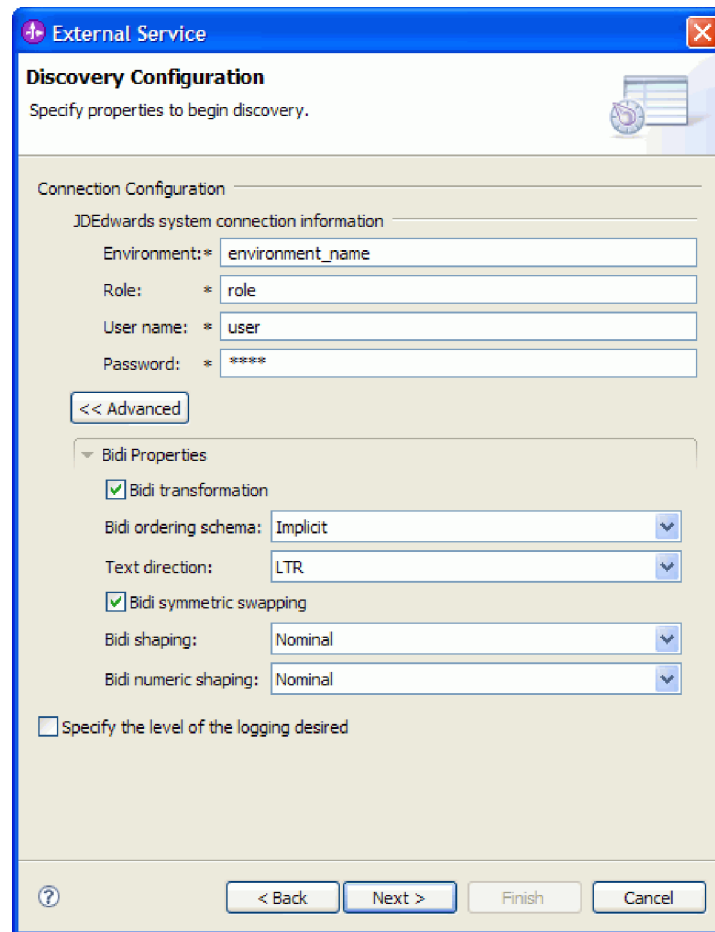


図 15. オプションの BiDi プロパティを示す「ディスカバリー構成 (Discovery Configuration)」ウィンドウ

3. オプション: 外部サービス・ウィザード用のロギング・プロパティを設定するには、以下の手順を実行します。
 - a. 「**必要なロギングのレベルの指定**」を選択します。
 - b. 「**参照**」をクリックして別の場所を選択することにより、ログ・ファイルの出力場所を変更します。

- c. 「**ロギング・レベル (Logging Level)**」を設定します。

テスト環境では、「**極めて詳細 (FINEST)**」(最高レベルのロギングを実行します)を選択します。実稼働環境では、FINEST よりも低いレベルを選択し、ロギング・プロセスを最適化してください。

注: このログは、外部サービス・ウィザードだけを対象としており、アダプターの操作は対象外です。

4. 「**次へ**」をクリックします。

結果

外部サービス・ウィザードは、ログイン用に指定した情報(ユーザー名やパスワードなど)を使用して、JD Edwards EnterpriseOne サーバーに接続します。「オブジェクトのディスカバリーおよび選択」ウィンドウが表示されます。

外部サービス・ウィザードによって JD Edwards EnterpriseOne サーバー上の関数またはデータのディスカバーに使用される検索基準を指定します。

Outbound 処理のモジュールの構成

アダプターを Outbound 処理に使用するようにモジュールを構成するには、WebSphere Integration Developer 内で外部サービス・ウィザードを使用して、JD Edwards EnterpriseOne サーバーからビジネス・オブジェクトおよびサービスを検出して選択し、ビジネス・オブジェクト定義および関連する成果物を生成します。

外部サービス・ディスカバリーを使用したビジネス関数の生成

ビジネス関数を使用して WebSphere Adapter for JD Edwards EnterpriseOne を構成するには、WebSphere Integration Developer で外部サービス・ウィザードを使用して、JD Edwards EnterpriseOne サーバー 内にあるビジネス関数を選択し、Outbound 処理のビジネス・オブジェクト定義および関連成果物を生成します。

ビジネス・オブジェクトおよびサービスの選択

呼び出すビジネス関数と処理するデータを指定するため、外部サービス・ウィザードに情報を入力します。

始める前に

外部サービス・ウィザードの接続プロパティを設定したことを確認してください。

このタスクを実行する理由および時期

外部サービス・ウィザードが JD Edwards EnterpriseOne サーバー でビジネス関数を検出するために使用する検索基準を指定します。外部サービス・ウィザードは、その検索基準に合致したビジネス関数のリストを戻します。

検索基準を指定してビジネス関数を選択するには、以下の手順を使用します。

このタスクの手順

1. 「オブジェクトのディスカバリーおよび選択」ウィンドウで、JD Edwards EnterpriseOne サーバー でビジネス関数を検索する照会が設定されていることを確認します。「照会」フィールドで、照会基準: **Business Functions=true** が設定されていることを確認します。「**Business Functions=false**」となっている場合は、以下の手順に従って Business Functions を「true」に設定します。
 - a. 「照会の編集」をクリックします。
 - b. 「照会プロパティ (Query Properties)」ウィンドウで、XML List のチェック・ボックスをクリアし、照会でビジネス関数のみが検索されるようにします。「照会プロパティ (Query Properties)」ウィンドウの「追加」ボタンがぼかし表示になります。
 - c. 「OK」をクリックします。
2. 「照会の実行」をクリックします。
3. 「ディスカバリーされたオブジェクト」リストで、処理したいビジネス関数を指定します。
 - a. 「**Business Function**」を展開します。これで、フィルター・ボタンが使用可能になります。

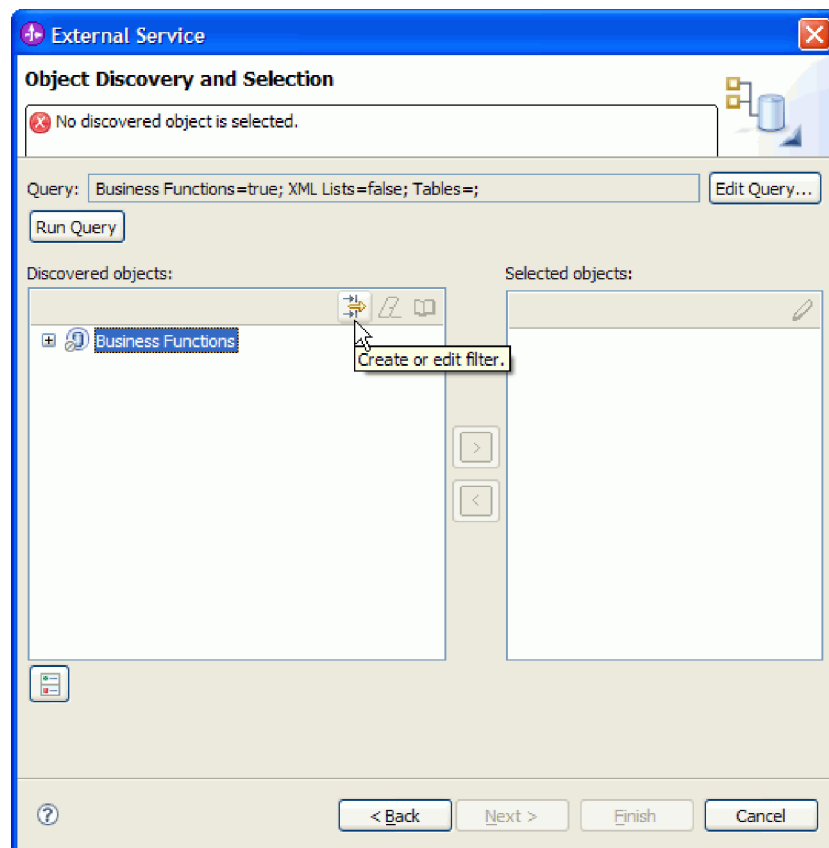


図 16. フィルター・ボタンが使用可能になっている「オブジェクトのディスカバリーと選択 (Object Discovery and Selection)」ウィンドウ

- b. 次のいずれかの方法により、ビジネス関数を検索します。

- 必要なビジネス関数のライブラリー名 (例: **CFIN**)、C ファイル名 (例: **B0100033**)、およびビジネス関数名 (例 : **GetEffectiveAddress**) がわかっている場合は、フィルター・ボタンをクリックします。
- ライブラリー名、C ファイル名、またはビジネス関数名がわからない場合は、「**Business Function**」、ライブラリー名、C ファイル名を展開し、ビジネス関数を選択します。

ヒント: 以下の 2 つの図に示されているように、マウスオーバー・テキストは、各ライブラリー名と C ファイル名の目的を示しています。

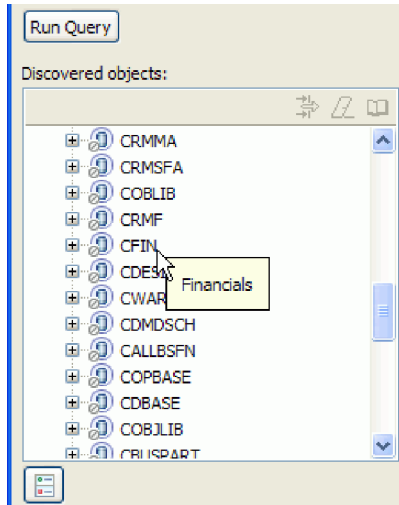


図 17. 検出されたビジネス関数に **CFIN** ライブラリーのマウスオーバー・テキストが表示された「オブジェクトのディスカバリーと選択 (Object Discovery and Selection)」ウィンドウ

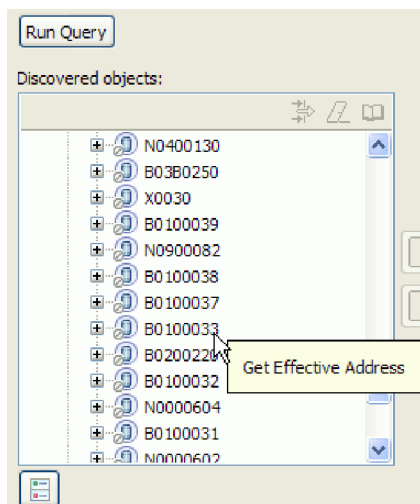


図 18. 検出されたビジネス関数に **B0100033** C ファイルのマウスオーバー・テキストが表示された「オブジェクトのディスカバリーと選択 (Object Discovery and Selection)」ウィンドウ

4. ビジネス関数を選択します。例えば、「**Business Function**」 → 「**CFIN**」 → 「**B0100033**」とナビゲートした場合、「**GetEffectiveAddress**」を選択します。

5. 矢印ボタンをクリックして、ビジネス関数を「選択済みオブジェクト」リストに追加します。

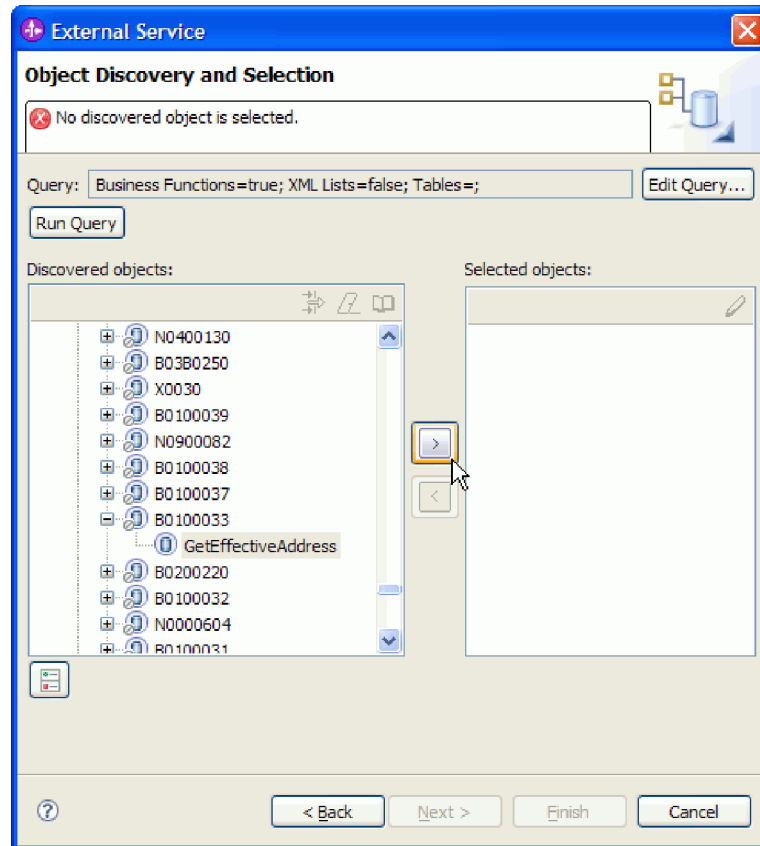


図 19. 矢印をクリックしてビジネス関数を「選択済みオブジェクト」リストに追加する

6. 「構成プロパティ (Configuration Properties)」ウィンドウで、デフォルトのビジネス・オブジェクト名をそのまま使用するか、別の名前を入力します。デフォルト名は、ビジネス関数の名前になります。

注: このビジネス・オブジェクト名は、意味のある値になっていないので、わかりやすい名前を設定できます。設定した名前は、ビジネス・オブジェクトの機能には影響しません。

7. 「OK」をクリックして、インポートするビジネス・オブジェクトのリストにビジネス関数を追加します。
8. 「次へ」をクリックします。

結果

以上で、作業を行うビジネス関数を選択し、その名前を選択しました。

「複合プロパティの構成 (Configure Composite Properties)」ウィンドウから、ビジネス・オブジェクト・コンテナ名および関連操作を指定します。オプションで、生成されたビジネス・オブジェクトが保管される名前空間とディレクトリー、およびビジネス・グラフを生成するかどうかを指定します。

選択されたオブジェクトの構成

ビジネス関数を構成するため、オブジェクトの名前、オブジェクトに関連付けられている操作など、そのオブジェクトに関する情報を指定します。

始める前に

ビジネス関数を選択してインポートしたことを確認してください。

このタスクを実行する理由および時期

ビジネス関数を構成するには、以下の手順を使用します。

このタスクの手順

1. オプション: 「複合プロパティの構成 (Configure Composite Properties)」ウィンドウで、以下の情報を入力します。
 - a. ビジネス関数をビジネス・グラフ内に囲む場合は、「**ビジネス・グラフ**」を選択したままにします。それ以外の場合は、チェック・マークを外します。
 - b. 「**ビジネス・オブジェクト Namespace**」フィールドでは、次の場合を除いてデフォルトの名前空間 (<http://www.ibm.com/xmlns/prod/websphere/j2ca/jde>) を使用します。すなわち、既存のモジュールにビジネス・オブジェクトを追加するときに、そのモジュールに対して外部サービス・ウィザードを実行したことがあり、ビジネス・オブジェクトが既に取り込まれているという場合に限っては、名前空間の値を変更します。例えば、名前空間を <http://www.ibm.com/xmlns/prod/websphere/j2ca/jde1> に変更します。
 - c. 「**相対パス**」フィールドで、ビジネス・オブジェクトを保管するディレクトリーを指定します。
 - d. 「**レコードの最大数**」フィールドに、RetrieveAll 操作の処理時に取り出すレコードの最大数を指定します。デフォルト値は 100 です。
 - e. 「**タイムアウト (ミリ秒)**」フィールドで、ミリ秒単位のタイムアウト値を指定します。値を設定しないと、デフォルト値は 30,000 ミリ秒 (30 秒) になります。
2. 必須: ビジネス関数にビジネス・オブジェクト・コンテナを追加し、そのコンテナに名前を割り当てます。すべてのビジネス関数には、ビジネス・オブジェクト・コンテナの追加が必要です。

注: 1 つのビジネス・オブジェクト・コンテナには、1 つまたは多数のビジネス関数を追加できます。

- a. 「**追加**」ボタンをクリックします。

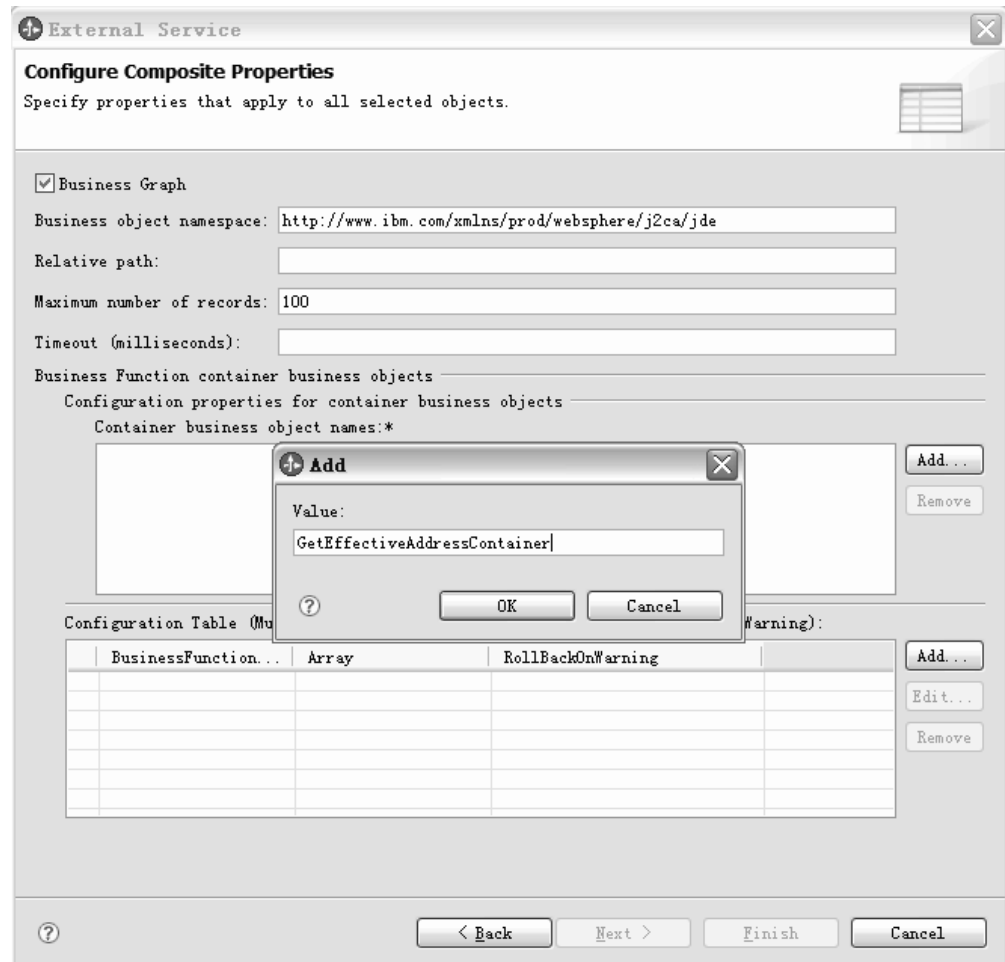


図 20. 「追加」ウィンドウを表示する「複合プロパティの構成 (Configure Composite Properties)」ウィンドウ

- b. 「追加」ウィンドウで、「値」フィールドに名前を入力します。わかりやすい名前なら、入力する名前はどのような名前でもかまいません。例えば、ビジネス関数の名前が `GetEffectiveAddress` ならば、「**GetEffectiveAddressContainer**」などを入力します。
3. 必須: ビジネス・オブジェクトに関連付ける操作を指定します。
 - a. 「追加」ボタンをクリックします。

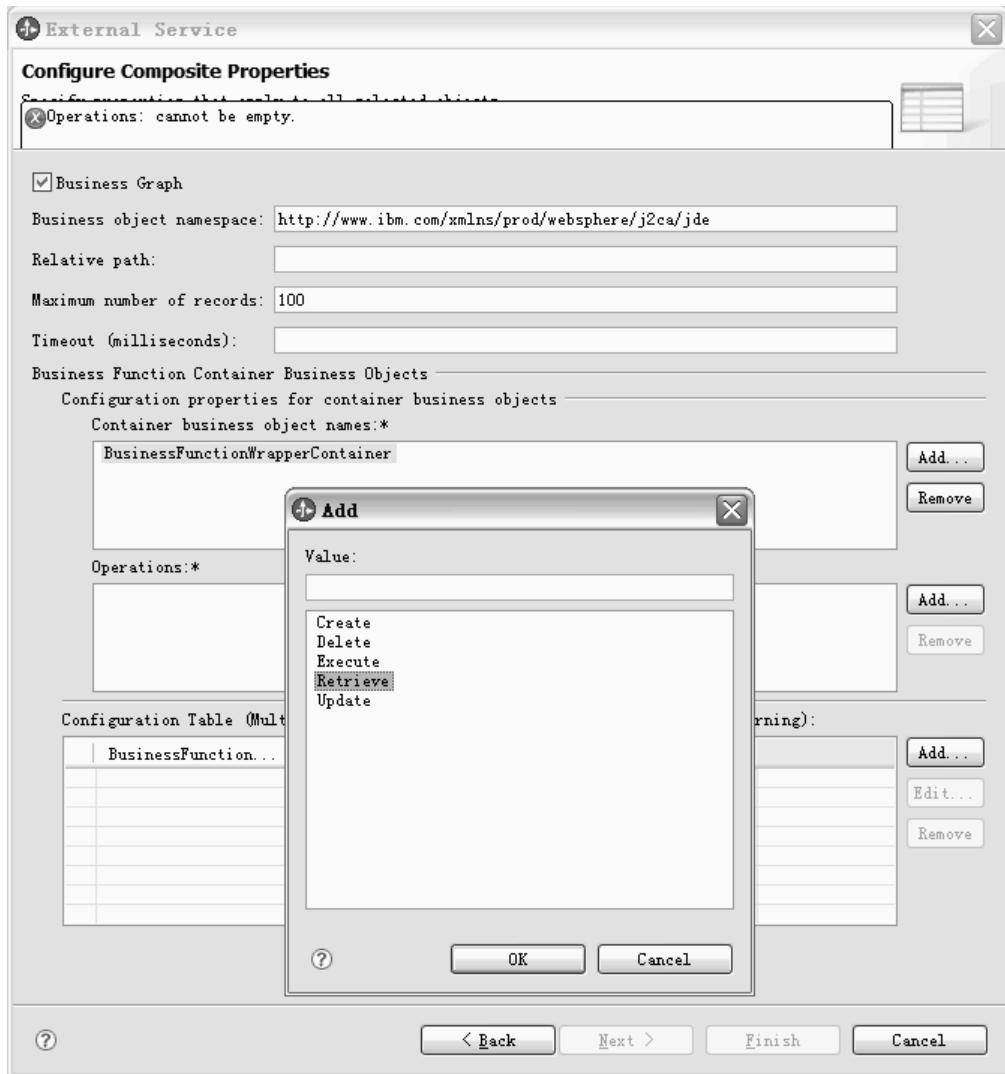


図 21. 「追加」ウィンドウに使用可能な操作が表示された「複合プロパティの構成 (Configure Composite Properties)」ウィンドウ

- b. 「追加」ウィンドウで、インポートする操作を選択し、「OK」をクリックします。
4. 必須: ビジネス関数を操作に関連付けます。
 - a. 「追加」ボタンをクリックし、ビジネス関数を作成します。

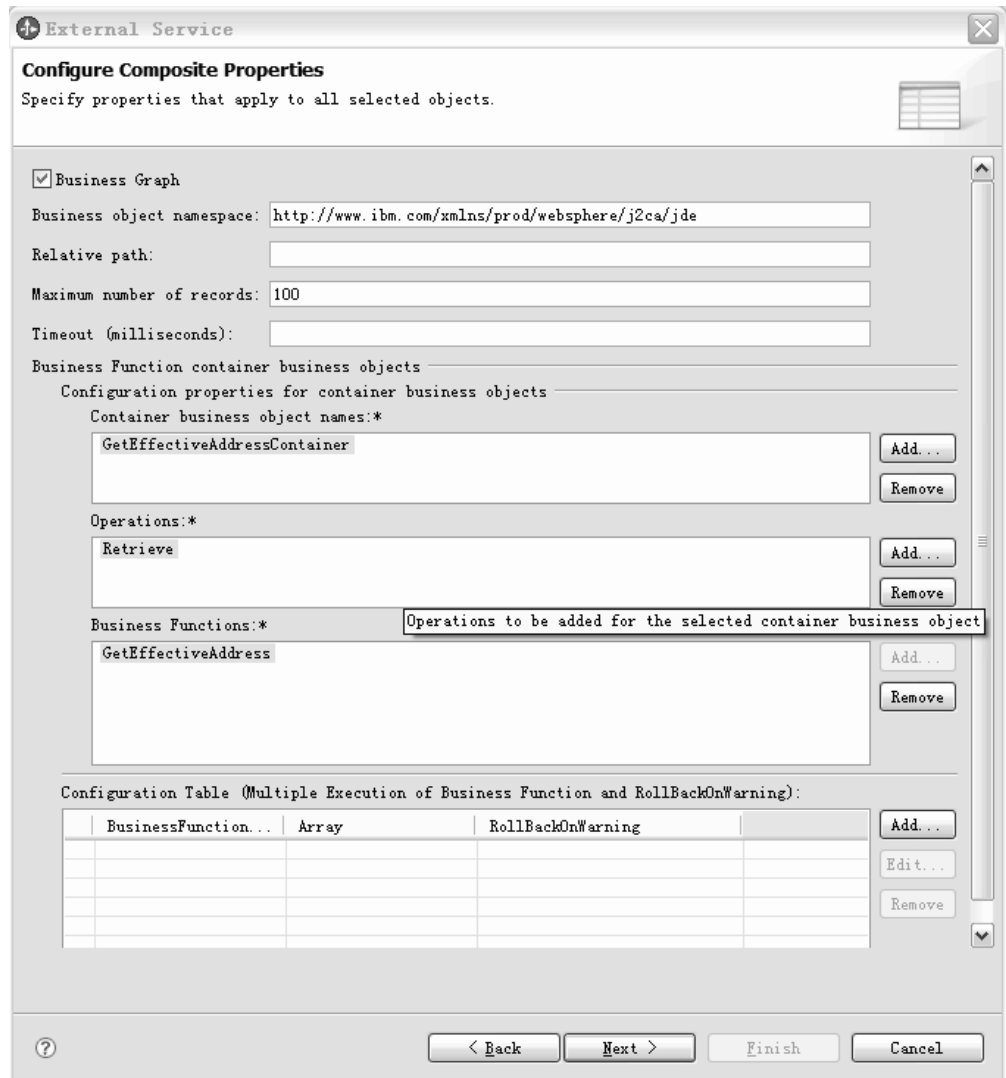


図 22. 「複合プロパティの構成 (Configure Composite Properties)」ウィンドウ

- b. 「追加」ウィンドウで、該当するビジネス関数を選択し、「OK」をクリックします。
5. オプション: 特定のビジネス関数を複数回実行するようにします。
 - a. 該当するビジネス関数を選択します。
 - b. 「追加」ボタンをクリックし、そのビジネス関数のプロパティを指定します。

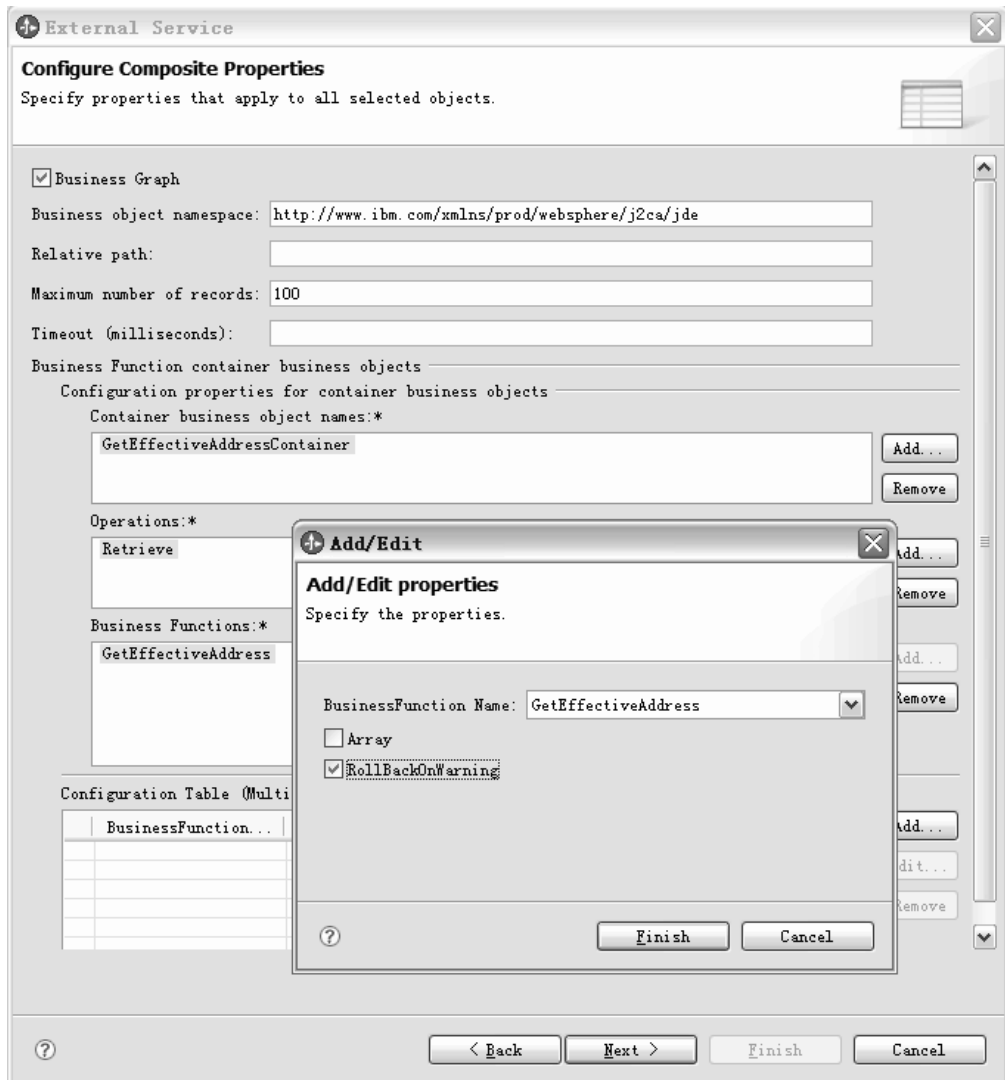


図 23. 「プロパティの追加/編集 (Add/Edit properties)」ウィンドウを表示する「複合プロパティの構成 (Configure Composite Properties)」ウィンドウ

- c. ビジネス関数を複数回実行するようにする場合は、「配列」チェック・ボックスを選択します。
 - d. ビジネス関数が警告を返した後でもそのビジネス関数を実行するようにする場合は、「警告時ロールバック」チェック・ボックスにチェックを入れます。
6. 「次へ」をクリックします。

結果

ビジネス・オブジェクト・コンテナの名前を指定し、ビジネス関数の操作が選択されました。「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウが表示されます。

次のタスク

アダプターとビジネス・オブジェクトを組み込む配置可能なモジュールを生成します。

デプロイメント・プロパティの設定およびサービスの生成

モジュール (WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイされる成果物) を生成するため、モジュールを作成し、モジュールにアダプターを関連付けます。

始める前に

ビジネス関数を構成したことを確認してください。「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウが表示されるはずです。

このタスクを実行する理由および時期

アダプターと構成済みのビジネス・オブジェクトを組み込むモジュールを生成します。モジュールとは、サーバーに配置する成果物です。

モジュールを生成するには、以下の手順を使用します。

このタスクの手順

1. オプション: デフォルトの操作名を変更するには、以下の手順を使用します。
 - a. 「**操作の編集**」をクリックします。
 - b. 「**操作名の編集**」ウィンドウで、編集する操作を選択し、「**編集**」をクリックします。
 - c. 「**プロパティの追加/編集 (Add/Edit properties)**」ウィンドウで、新規の操作名とオプションの説明を入力し、「**終了**」をクリックします。
2. JD Edwards EnterpriseOne サーバー への接続を確立するために、認証別名を (ユーザー ID とパスワードを入力する代わりに) 使用するかどうかを示します。
 - 認証別名を指定する場合は、「**Java Authentication and Authorization Service (JAAS) の別名セキュリティ・クリデンシャルの指定 (Specify a Java Authentication and Authorization Services (JAAS) alias security credential)**」を選択したままにします。次に「**J2C 認証データ項目**」フィールドに、管理コンソールの「**セキュリティ**」セクションで指定されている名前を入力します。
 - 認証別名を使用しない場合は、「**Java Authentication and Authorization Service (JAAS) の別名セキュリティ・クリデンシャルの指定 (Specify a Java Authentication and Authorization Services (JAAS) alias security credential)**」をクリアします。
3. 「**単一アプリケーションが使用するモジュールで (With module for use by single application)**」を選択して、アプリケーション・サーバーにデプロイされるモジュールにアダプター・ファイルを組み込むか、「**複数アプリケーションが使用するサーバー上 (On server for use by multiple applications)**」を選択して、スタンドアロン・アダプターとしてアダプター・ファイルをインストールします。

- 「**単一アプリケーションが使用するモジュールとともにデプロイする**」。アダプター・ファイルをモジュール内に組み込むと、モジュールをあらゆるアプリケーション・サーバーにデプロイすることができます。組み込みアダプターを使用するのは、組み込みアダプターを使用するモジュールが 1 つある場合か、複数のモジュールでバージョンの異なるアダプターを実行する必要がある場合です。組み込みアダプターを使用すると、他のモジュールのアダプター・バージョンを変更することで、それらのモジュールを不安定にするリスクを生じることなく、1 つのモジュール内でアダプターをアップグレードできます。
 - 「**複数アプリケーションが使用するサーバー上**」。モジュール内にアダプター・ファイルを組み込まない場合は、このモジュールを実行するアプリケーション・サーバーごとにモジュールをスタンドアロン・アダプターとしてインストールする必要があります。複数のモジュールが同じバージョンのアダプターを使用可能で、アダプターを中央の場所で管理する場合は、スタンドアロン・アダプターを使用します。スタンドアロン・アダプターの場合も、複数のモジュールに対して単一のアダプター・インスタンスを実行することにより、必要なリソースが軽減されます。
4. 前のステップで「**複数アプリケーションが使用するサーバー上(On server for use by multiple applications)**」を選択した場合は、「**接続プロパティー**」リストがアクティブになります。以下のうちから 1 つを選択します。
- 構成情報をここで指定する場合は、「**接続プロパティーの指定 (Specify connection properties)**」を選択します。ステップ 5 に進みます。
 - 既存の接続ファクトリー構成を使用する場合は、「**定義済みの接続プロパティーの使用 (Use predefined connection properties)**」を選択します。

定義済みの接続プロパティーを使用する場合は、ご使用のリソース・アダプター名とインストールされているアダプター名を一致させる必要があります。これが、アダプターのインスタンスをこれらのプロパティーに関連付ける方法であるためです。インポートまたはエクスポートのリソース・アダプターの名前を変更したい場合は、WebSphere Integration Developer のアセンブリー・エディターを使用して、インポートまたはエクスポートの値を変更します。

- 「**定義済みの接続プロパティーの使用 (Use predefined connection properties)**」を選択すると、プロパティーの代わりに「**JNDI ルックアップ名 (JNDI Lookup Name)**」フィールドが表示されます。
- a. 「**JNDI ルックアップ名 (JNDI Lookup Name)**」に値を入力します。
 - b. 「**次へ**」をクリックします。
 - c. ステップ 9 に進みます。
5. 「**接続プロパティー**」セクションで、ご使用の構成に適用される接続プロパティーがあれば、それを設定または変更します。

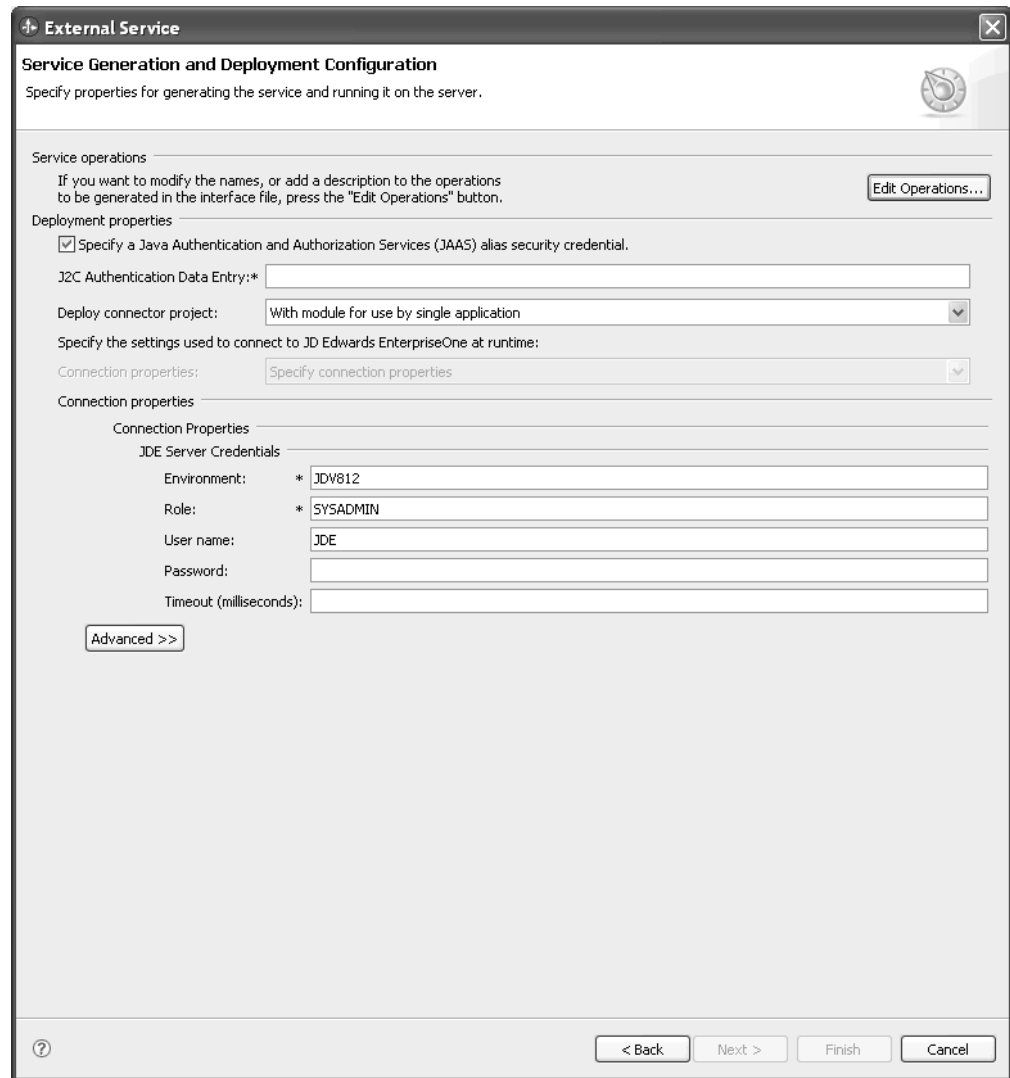


図 24. 「接続プロパティ」を表示している「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウ

これらのプロパティの詳細については、136 ページの『Managed Connection Factory プロパティ』を参照してください。

アスタリスク (*) が付いているプロパティは必須です。

6. その他のプロパティを設定するには、「拡張」をクリックし、次のセクションを展開します。

- **ロギングおよびトレース**

- 複数のアダプター・インスタンスがある場合は、アダプター ID をこのインスタンスに固有の値に設定します。

7. 「次へ」をクリックします。

8. 新規のモジュールを作成します。

- a. 「サービス・ロケーション・プロパティ (Service Location Properties)」ウィンドウの「モジュール (Module)」フィールドで「新規作成」をクリックします。

- b. 「統合プロジェクト (Integration Project)」ウィンドウで、「**モジュール・プロジェクトの作成 (Create a module project)**」または「**メディエーション・モジュール・プロジェクトの作成 (Create a mediation module project)**」を選択して、「次へ」をクリックします。
- c. 「新規モジュール (New Module)」ウィンドウで、モジュールの名前を入力します。

注: 名前を入力すると、「**ロケーション (Location)**」フィールドで指定されているワークスペース・ディレクトリーにその名前が追加されます。

これはデフォルトのロケーションです。別のロケーションを指定する場合は、「**デフォルト・ロケーションを使用する (Use default location)**」からチェック・マークを外し、新しいロケーションを入力するか、または、「**参照**」をクリックしてロケーションを選択します。

- d. アセンブリー・ダイアグラムでモジュールを開くか (モジュール・プロジェクトの場合)、またはメディエーション・フロー・コンポーネントを作成するか (メディエーション・モジュール・プロジェクトの場合) を指定します。デフォルトでは、これらの選択項目は選択されています。
 - e. 「**終了**」をクリックします。新規のモジュールが作成されます。作成プロセスが終了したら、「新規モジュール (New Module)」ウィンドウが閉じ、新規のモジュールが「サービス・ロケーション・プロパティー (Service Location Properties)」ウィンドウの「**モジュール (Module)**」リストに表示されます。
9. 「サービス・ロケーション・プロパティー (Service Location Properties)」ウィンドウで、以下の手順を実行すれば、新規モジュールのデプロイメント・プロパティーを指定できます。
 - a. デフォルトの名前空間を変更する場合は、「**デフォルト名前空間を使用する (Use Default Namespace)**」チェック・ボックスをクリアして、「**名前空間**」フィールドに新しいパスを入力します。
 - b. サービス記述を保存するモジュール内のフォルダーを指定します。そのためには、「**フォルダー (Folder)**」フィールドに名前を入力するか、フォルダーを見つけて選択します。
 - c. インターフェースの名前を変更します。

デフォルト名は「JDEOutboundInterface」です。希望する場合、これをより説明的なタイトルに変更することができます。

- d. ビジネス・オブジェクトを別のアプリケーションが使用できるように保存する場合は、「**ビジネス・オブジェクトをライブラリーに保存 (Save business objects to a library)**」をクリックしてからリストからライブラリーを選択するか、または「**新規作成**」をクリックして、新規ライブラリーを作成します。
 - e. モジュールの説明を入力します。
10. 「**終了**」をクリックします。

結果

これで、WebSphere Integration Developer の Business Integration パースペクティブに、新しいモジュールが作成され、構成、保存されました。

次のタスク

配置のための EAR ファイルとしてモジュールをエクスポートします。

外部サービス・ディスカバリーを使用した XML List の生成

XML List を使用して WebSphere Adapter for JD Edwards EnterpriseOne を構成するには、WebSphere Integration Developer で外部サービス・ウィザードを使用して、JD Edwards EnterpriseOne サーバー 内にある XML List を選択し、Outbound 処理のビジネス・オブジェクト定義および関連成果物を生成します。

ビジネス・オブジェクトおよびサービスの選択

XML List について処理するデータを指定するため、外部サービス・ウィザードに情報を入力します。

始める前に

外部サービス・ウィザードの接続プロパティを設定したことを確認してください。また、作成したい XML List のテーブル名、テーブル・タイプを含むテーブル情報を収集してあることを確認します。

ヒント: 作成したい XML List のテーブル情報を収集するには、JD Edwards EnterpriseOne Universal Table Browser (UTB) を使用します。JD Edwards EnterpriseOne UTB の使用方法については、「*JD Edwards EnterpriseOne Tools 8.96 System Administration Guide*」を参照してください。

このタスクを実行する理由および時期

外部サービス・ウィザードが JD Edwards EnterpriseOne サーバー でデータベース・テーブル情報を検出するために使用する検索基準を指定します。外部サービス・ウィザードは、その検索基準に合致したオブジェクトのリストを戻します。

XML List を作成するための検索基準を指定するには、以下の手順を使用します。

このタスクの手順

1. 「オブジェクトのディスカバリーおよび選択」ウィンドウで、照会を編集し、XML List の作成に必要なテーブル情報を検索するように準備します。
 - a. 「照会の編集」をクリックします。
 - b. 「照会プロパティ (Query Properties)」ウィンドウで、「XML List」を選択します。

注: オプションで、「Business Function」のチェック・ボックスをクリアし、照会で XML List のみを検索するようにすることができます。

「Business Function」を選択したままにすると、照会では、XML List とビジネス関数の両方を返します。

- c. テーブル名を追加するには、「追加」をクリックします。

- d. 「追加」ウィンドウで、照会に追加したいデータベース・テーブルの名前を入力します。例えば、データベース・テーブルとして F0116 と入力します。必要なテーブル名が分からない場合は、JD Edwards EnterpriseOne Universal Table Browser (UTB) を使用すれば、JD Edwards EnterpriseOne サーバーでテーブル名を検索できます。
- e. 「追加」ウィンドウで、「OK」をクリックします。入力したテーブル名が、「テーブル」リストに表示されます。
- f. 「照会プロパティ (Query Properties)」ウィンドウで、「OK」をクリックします。

これで、XML List を作成するために指定した JD Edwards EnterpriseOne データベース・テーブルを、照会で検出する準備ができました。

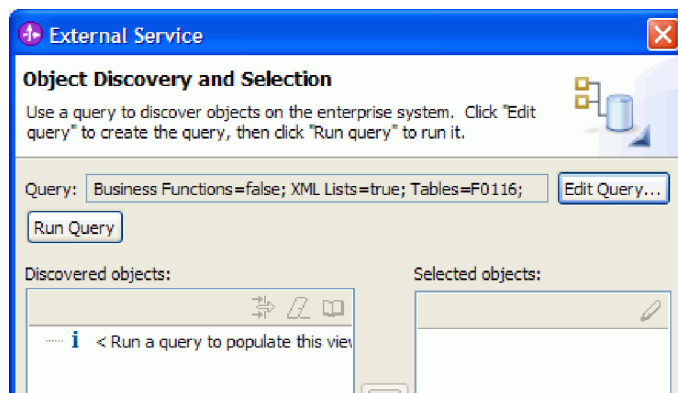


図 25. XML List 照会の実行準備が整った「照会」フィールドを示す「オブジェクトのディスカバリーおよび選択」ウィンドウ

2. XML List 照会を実行し、指定したテーブルに一致する JD Edwards EnterpriseOne サーバー内のテーブルを検索し、検出します。
 - a. 「照会の実行」をクリックします。外部サービス・ウィザードが、JD Edwards EnterpriseOne サーバーを照会します。照会の結果は、「ディスカバーされたオブジェクト」リストに表示されます。
 - b. 「ディスカバーされたオブジェクト」リストで、「XML Lists」を展開します。
 - c. 作成したテーブルに一致するテーブルにナビゲートします。
 - d. 矢印ボタンをクリックして、テーブルを「選択済みオブジェクト」リストに追加します。

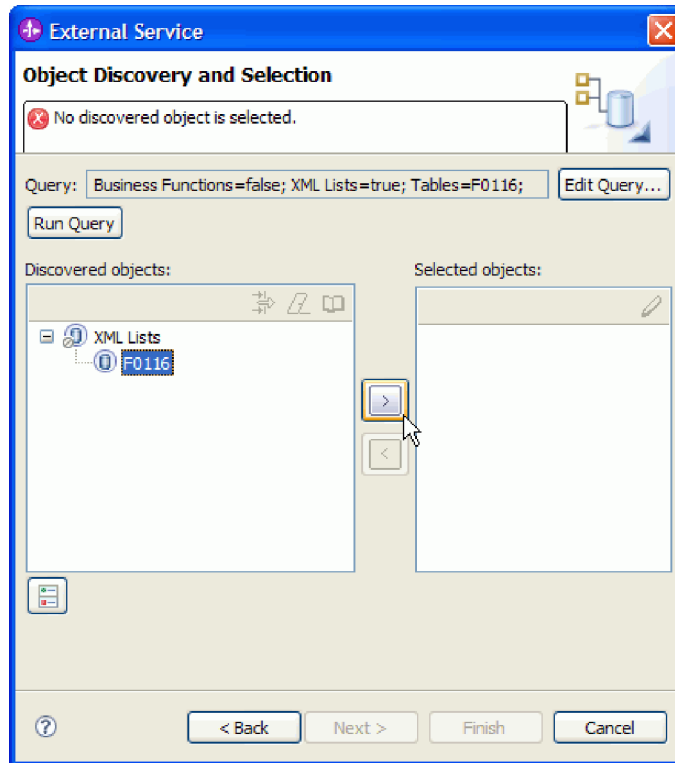


図 26. 矢印をクリックして検出されたテーブルを「選択済みオブジェクト」リストに追加する

3. 「構成パラメーター (Configuration Parameters)」ウィンドウで、データを JD Edwards EnterpriseOne サーバー からインポートする前に、検索基準を XML List 照会に追加します。これによって、テーブル・タイプやソート条件などの照会パラメーターを指定することができます。

- a. 「ビジネス・オブジェクト名」フィールドで、デフォルトのビジネス・オブジェクト名をそのまま使用するか、ニーズに合うように名前を変更します。デフォルト名は、テーブルの名前になります。

注: このビジネス・オブジェクト名は、意味のある値になっていませんので、わかりやすい名前を設定できます。設定した名前は、ビジネス・オブジェクトの機能には影響しません。

- b. 「テーブル・タイプ」フィールドで、リストからテーブルのタイプを選択します。例えば、OWTABLE テーブル・タイプを F0116 テーブルに選択します。

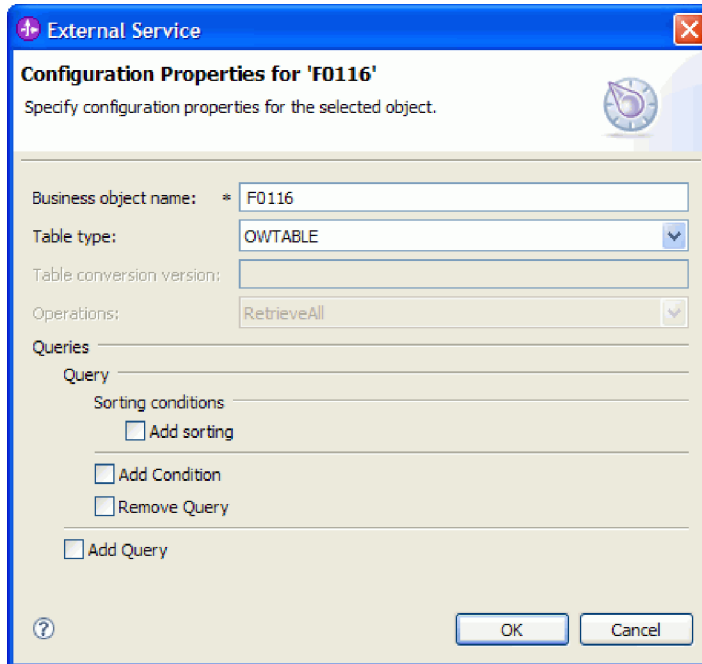


図 27. 「テーブル・タイプ」フィールドに *OWTABLE* を選択した状態を示す「構成プロパティ (Configuration Properties)」ウィンドウ

次のテーブル・タイプを選択できます。

表 3. *XML List* のテーブル・タイプ

テーブル・タイプ	説明	用途
OWTABLE	JD Edwards EnterpriseOne データベース・テーブル	このテーブル・タイプは、必要なテーブルが JD Edwards EnterpriseOne データベースにある場合に使用します。
OWVIEW	2 つ以上のテーブル間の関係を定義するビジネス・ビューで、データを単一のビューに結合します。	このビジネス・ビューは、TABLE_CONVERSION テーブル・タイプの入力として使用します。
FOREIGN_TABLE	Oracle、Access、iSeries®、SQL Server など、JD Edwards EnterpriseOne がサポートするデータベースに常駐する、JD Edwards EnterpriseOne 以外のデータベース・テーブル	このテーブル・タイプは、必要なテーブルが JD Edwards EnterpriseOne 以外のデータベースにある場合に使用します。

表 3. XML List のテーブル・タイプ (続き)

テーブル・タイプ	説明	用途
TABLE_CONVERSION	テーブル内のデータを素早く操作できるバッチ処理を使用するテーブル・タイプ。テーブル変換をテンプレートとしてセットアップし、それらを複数回実行し、環境のニーズに合うように異なるバージョンを使用して改訂することができます。	<p>このテーブル・タイプは、データのバッチを次のいずれかで操作する予定がある場合に使用します。</p> <ul style="list-style-type: none"> データ変換: 入力テーブルまたはビジネス・ビューから出力テーブルにデータを転送することができます。また、テーブルまたはビジネス・ビューのレコードを更新できます。 データ・コピー: テーブルが同一の場合、あるデータ・ソースまたは環境から別のデータ・ソースまたは環境にテーブルをコピーできます。 テーブル入力によるデータ・コピー: 入力テーブルからの情報に基づいてテーブルをコピーすることができます。 バッチ削除: テーブルまたはビジネス・ビューからレコードを削除できます。

- c. 照会にソート条件を追加するには、「ソート条件の追加」をクリックし、「選択」をクリックして、ソートする属性を選択してから、ソート・リストから、「昇順」または「降順」を選択します。

注: ソート条件を削除する場合は、「ソート条件の削除」をクリックします。

- d. 他の条件を照会に追加する場合、「条件の追加」をクリックし、以下のいずれかの条件を選択します。
- 属性: 条件を追加する属性を選択します。
 - 文節: 照会条件の文節を選択します。デフォルトは、**Where** です。
 - 演算子: 属性と列値とを比較する場合の演算子を選択します。
 - 属性値の使用: 比較する属性を選択します。
 - デフォルト: 照会条件のデフォルト値を指定します。

注: 条件を指定しないと、すべてのレコードが検索されます。複数の条件を照会に追加する場合、「条件の追加」を再度クリックします。不要な条件を削除する場合は、「条件の削除」をクリックします。

- e. 「OK」をクリックします。テーブル名が「選択済みオブジェクト」リストに表示されます。

4. 「次へ」をクリックします。

結果

以上で、作業を行うテーブルを選択し、作成する XML List 用にそのテーブルを構成しました。

「複合プロパティの構成 (Configure Composite Properties)」ウィンドウからオプションで、生成済み XML List を格納する名前空間およびディレクトリーを指定します。

選択されたオブジェクトの構成

XML List を構成するため、取り出すレコードの最大数など、そのオブジェクトに関する情報を指定します。

始める前に

XML List を選択してインポートしたことを確認してください。

このタスクを実行する理由および時期

XML List を構成するには、以下の手順を使用します。

このタスクの手順

1. オプション: 「複合プロパティの構成 (Configure Composite Properties)」ウィンドウで、以下の情報を入力します。

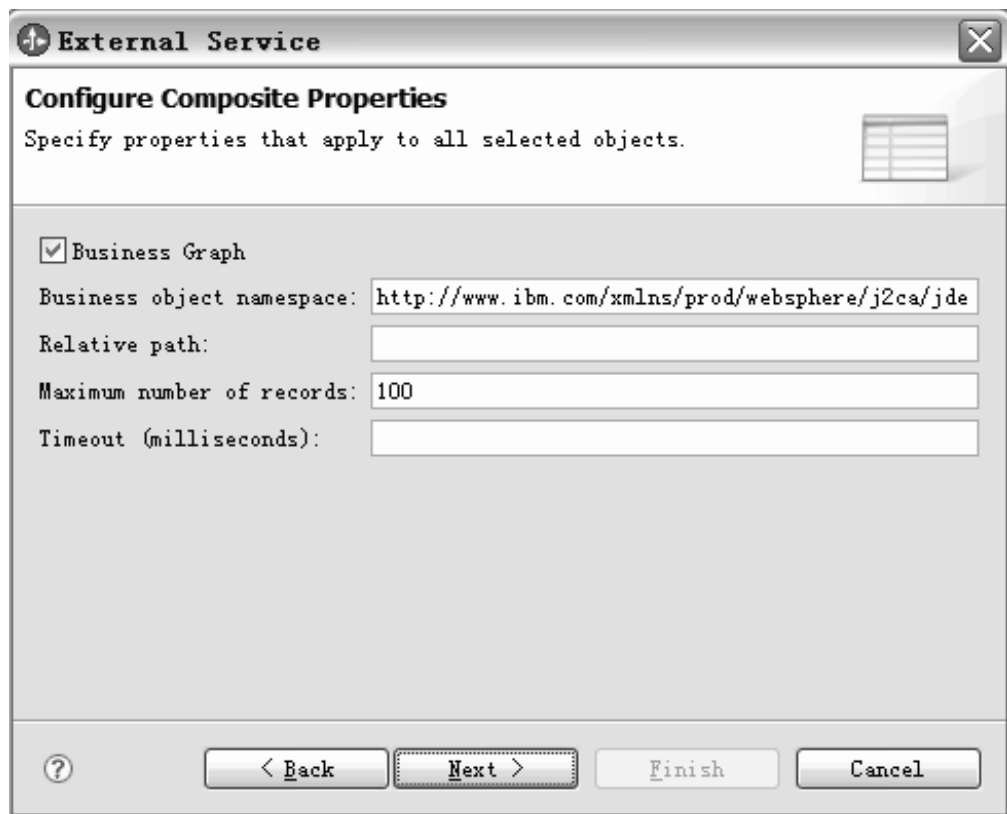


図 28. XML List のオプション構成設定を表示する「複合プロパティの構成 (Configure Composite Properties)」ウィンドウ

- a. 「ビジネス・グラフ」は、選択したままにします。

注: この選択は XML List の場合必須です。これによって、複数のレコードを取り出すことができます。

- b. 「ビジネス・オブジェクト Namespace」フィールドでは、次の場合を除いてデフォルトの名前空間 (<http://www.ibm.com/xmlns/prod/websphere/j2ca/jde>) を使用します。すなわち、既存のモジュールに XML List を追加するときに、そのモジュールに対して外部サービス・ウィザードを実行したことがあり、

XML List が既に取り込まれているという場合に限っては、名前空間の値を変更します。例えば、名前空間を `http://www.ibm.com/xmlns/prod/websphere/j2ca/jde1` に変更します。

- c. 「**相対パス**」フィールドで、XML List を保管するディレクトリーを指定します。
- d. 「**レコードの最大数**」フィールドに、RetrieveAll 操作の処理時に取り出すレコードの最大数を指定します。デフォルト値は 100 です。
- e. 「**タイムアウト (ミリ秒)**」フィールドで、ミリ秒単位のタイムアウト値を指定します。値を設定しないと、デフォルト値は 30,000 ミリ秒 (30 秒) になります。

重要: XML List に対するビジネス・オブジェクト・コンテナ名および関連操作は、デフォルトで次のように設定されます。

- ビジネス・オブジェクト・コンテナ名: `<XML_List_object>Container`
- 操作: RetrieveAll

2. 「次へ」をクリックします。

結果

XML List オブジェクトに対してオプションの構成パラメーターを設定しました。外部サービス・ウィザードにより、XML List に対するビジネス・オブジェクト・コンテナ名と関連操作が自動的に割り当てられました。「成果物の生成」ウィンドウが開きます。

次のタスク

アダプターとビジネス・オブジェクトを組み込む配置可能なモジュールを生成します。

デプロイメント・プロパティの設定およびサービスの生成

モジュール (WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイされる成果物) を生成するため、モジュールを作成し、モジュールにアダプターを関連付けます。

始める前に

XML List を構成したことを確認してください。「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウが表示されるはずですが。

このタスクを実行する理由および時期

アダプターと構成済みのビジネス・オブジェクトを組み込むモジュールを生成します。モジュールとは、サーバーに配置する成果物です。

モジュールを生成するには、以下の手順を使用します。

このタスクの手順

1. オプション: デフォルトの操作名を変更するには、以下の手順を使用します。

- a. 「**操作の編集**」をクリックします。
 - b. 「**操作名の編集**」ウィンドウで、編集する操作を選択し、「**編集**」をクリックします。
 - c. 「**プロパティの追加/編集 (Add/Edit properties)**」ウィンドウで、新規の操作名とオプションの説明を入力し、「**終了**」をクリックします。
2. JD Edwards EnterpriseOne サーバー への接続を確立するために、認証別名を (ユーザー ID とパスワードを入力する代わりに) 使用するかどうかを示します。
- 認証別名を指定する場合は、「**Java Authentication and Authorization Service (JAAS) の別名セキュリティ・クリデンシャルの指定 (Specify a Java Authentication and Authorization Services (JAAS) alias security credential)**」を選択したままにします。次に「**J2C 認証データ項目**」フィールドに、管理コンソールの「**セキュリティ**」セクションで指定されている名前を入力します。
 - 認証別名を使用しない場合は、「**Java Authentication and Authorization Service (JAAS) の別名セキュリティ・クリデンシャルの指定 (Specify a Java Authentication and Authorization Services (JAAS) alias security credential)**」をクリアします。
3. 「**単一アプリケーションが使用するモジュールで (With module for use by single application)**」を選択して、アプリケーション・サーバーにデプロイされるモジュールにアダプター・ファイルを組み込むか、「**複数アプリケーションが使用するサーバー上(On server for use by multiple applications)**」を選択して、スタンドアロン・アダプターとしてアダプター・ファイルをインストールします。
- 「**単一アプリケーションが使用するモジュールとともにデプロイする**」。アダプター・ファイルをモジュール内に組み込むと、モジュールをあらゆるアプリケーション・サーバーにデプロイすることができます。組み込みアダプターを使用するのは、組み込みアダプターを使用するモジュールが 1 つある場合か、複数のモジュールでバージョンの異なるアダプターを実行する必要がある場合です。組み込みアダプターを使用すると、他のモジュールのアダプター・バージョンを変更することで、それらのモジュールを不安定にするリスクを生じることなく、1 つのモジュール内でアダプターをアップグレードできます。
 - 「**複数アプリケーションが使用するサーバー上**」。モジュール内にアダプター・ファイルを組み込まない場合は、このモジュールを実行するアプリケーション・サーバーごとにモジュールをスタンドアロン・アダプターとしてインストールする必要があります。複数のモジュールが同じバージョンのアダプターを使用可能で、アダプターを中央の場所で管理する場合は、スタンドアロン・アダプターを使用します。スタンドアロン・アダプターの場合も、複数のモジュールに対して単一のアダプター・インスタンスを実行することにより、必要なリソースが軽減されます。
4. 前のステップで「**複数アプリケーションが使用するサーバー上(On server for use by multiple applications)**」を選択した場合は、「**接続プロパティ**」リストがアクティブになります。以下のうちから 1 つを選択します。
- 構成情報をここで指定する場合は、「**接続プロパティの指定 (Specify connection properties)**」を選択します。ステップ 5 に進みます。

- 既存の接続ファクトリー構成を使用する場合は、「**定義済みの接続プロパティの使用 (Use predefined connection properties)**」を選択します。

定義済みの接続プロパティを使用する場合は、ご使用のリソース・アダプター名とインストールされているアダプター名を一致させる必要があります。これが、アダプターのインスタンスをこれらのプロパティに関連付ける方法であるためです。インポートまたはエクスポートのリソース・アダプターの名前を変更したい場合は、WebSphere Integration Developer のアセンブリー・エディターを使用して、インポートまたはエクスポートの値を変更します。

「**定義済みの接続プロパティの使用 (Use predefined connection properties)**」を選択すると、プロパティの代わりに「**JNDI ルックアップ名 (JNDI Lookup Name)**」フィールドが表示されます。

- a. 「**JNDI ルックアップ名 (JNDI Lookup Name)**」に値を入力します。
 - b. 「次へ」をクリックします。
 - c. ステップ 9 に進みます。
5. 「接続プロパティ」セクションで、ご使用の構成に適用される接続プロパティがあれば、それを設定または変更します。

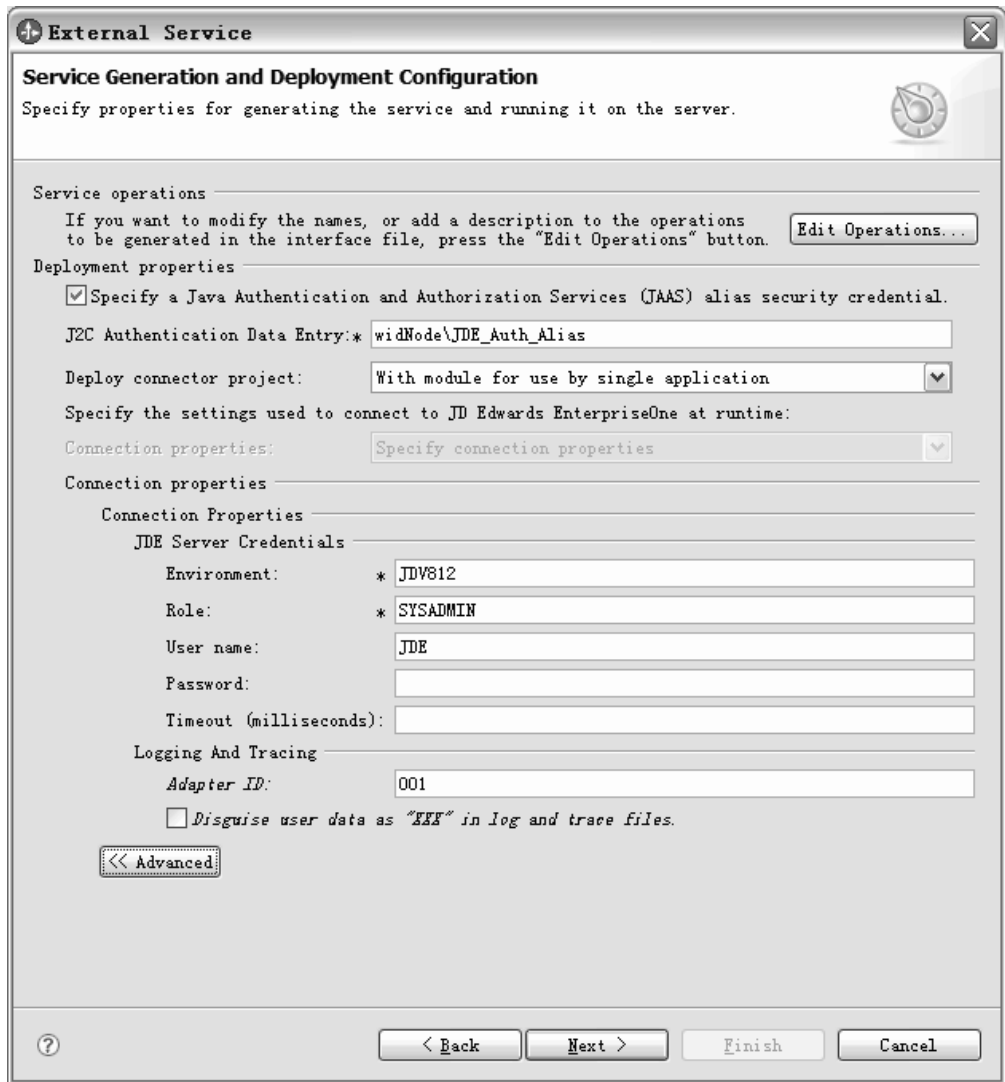


図 29. 「接続プロパティ」セクションを表示している「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウ

これらのプロパティの詳細については、136 ページの『Managed Connection Factory プロパティ』を参照してください。

アスタリスク (*) が付いているプロパティは必須です。

6. その他のプロパティを設定するには、「**拡張**」をクリックし、次のセクションを展開します。
 - **ロギングおよびトレース**
 - 複数のアダプター・インスタンスがある場合は、アダプター ID をこのインスタンスに固有の値に設定します。
7. 「**次へ**」をクリックします。
8. 新規のモジュールを作成します。
 - a. 「**サービス・ロケーション・プロパティ (Service Location Properties)**」ウィンドウの「**モジュール (Module)**」フィールドで「**新規作成**」をクリックします。

- b. 「統合プロジェクト (Integration Project)」ウィンドウで、「**モジュール・プロジェクトの作成 (Create a module project)**」または「**メディエーション・モジュール・プロジェクトの作成 (Create a mediation module project)**」を選択して、「次へ」をクリックします。
- c. 「新規モジュール (New Module)」ウィンドウで、モジュールの名前を入力します。

注: 名前を入力すると、「**ロケーション (Location)**」フィールドで指定されているワークスペース・ディレクトリーにその名前が追加されます。

これはデフォルトのロケーションです。別のロケーションを指定する場合は、「**デフォルト・ロケーションを使用する (Use default location)**」からチェック・マークを外し、新しいロケーションを入力するか、または、「**参照**」をクリックしてロケーションを選択します。

- d. アセンブリー・ダイアグラムでモジュールを開くか (モジュール・プロジェクトの場合)、またはメディエーション・フロー・コンポーネントを作成するか (メディエーション・モジュール・プロジェクトの場合) を指定します。デフォルトでは、これらの選択項目は選択されています。
 - e. 「終了」をクリックします。新規のモジュールが作成されます。作成プロセスが終了したら、「新規モジュール (New Module)」ウィンドウが閉じ、新規のモジュールが「サービス・ロケーション・プロパティ (Service Location Properties)」ウィンドウの「**モジュール (Module)**」リストに表示されます。
9. 「サービス・ロケーション・プロパティ (Service Location Properties)」ウィンドウで、以下の手順を実行し、モジュールの生成プロセスを完了します。
 - a. デフォルトの名前空間を変更する場合は、「**デフォルト名前空間を使用する (Use Default Namespace)**」チェック・ボックスをクリアして、「名前空間」フィールドに新しいパスを入力します。
 - b. サービス記述を保存するモジュール内のフォルダーを指定します。そのためには、「**フォルダー (Folder)**」フィールドに名前を入力するか、フォルダーを見つけて選択します。
 - c. インターフェースの名前を変更します。

デフォルト名は「JDEOutboundInterface」です。希望する場合、これをより説明的なタイトルに変更することができます。
 - d. ビジネス・オブジェクトを別のアプリケーションが使用できるように保存する場合は、「**ビジネス・オブジェクトをライブラリーに保存 (Save business objects to a library)**」をクリックしてからリストからライブラリーを選択するか、または「**新規作成**」をクリックして、新規ライブラリーを作成します。
 - e. モジュールの説明を入力します。
 10. 「終了」をクリックします。

結果

これで、WebSphere Integration Developer の Business Integration パースペクティブに、新しいモジュールが作成され、構成、保存されました。

次のタスク

配置のための EAR ファイルとしてモジュールをエクスポートします。

Inbound 処理のモジュールの構成

アダプターを Inbound 処理に使用するようにモジュールを構成するには、WebSphere Integration Developer 内で 外部サービス・ウィザードを使用して、JD Edwards EnterpriseOne サーバーからビジネス・オブジェクトおよびサービスを検出して選択し、ビジネス・オブジェクト定義および関連する成果物を生成します。

ビジネス・オブジェクトおよびサービスの選択

処理対象のリアルタイム・イベントを指定するため、外部サービス・ウィザードに情報を入力します。

始める前に

外部サービス・ウィザードの接続プロパティを適切に設定したことを確認してください。

このタスクを実行する理由および時期

外部サービス・ウィザードが JD Edwards EnterpriseOne サーバー でリアルタイム・イベントを検出するために使用する検索基準を指定します。外部サービス・ウィザードは、その検索基準に合致したビジネス関数のリストを戻します。

検索基準を指定してリアルタイム・イベントを選択するには、以下の手順を使用します。

このタスクの手順

1. 「オブジェクトのディスカバリーおよび選択」ウィンドウで、JD Edwards EnterpriseOne サーバー でリアルタイム・イベントを検索する照会が設定されていることを確認します。「照会」フィールドで、照会基準: **Real-Time Events=true** が設定されていることを確認します。「**Real-Time=false**」となっている場合は、以下の手順に従ってこれを「true」に設定します。
 - a. 「照会の編集」をクリックします。
 - b. 「照会プロパティ (Query Properties)」ウィンドウで、「**Real-Time Events**」を選択します。
 - c. 「OK」をクリックします。
2. 「照会の実行」をクリックします。
3. 「ディスカバリーされたオブジェクト」リストで、ルート・ノード: **Real-Time Events** を展開し、処理したいリアルタイム・イベントを 1 つ以上選択します。
4. 矢印ボタンをクリックして、リアルタイム・イベントを「**選択済みオブジェクト**」リストに追加します。

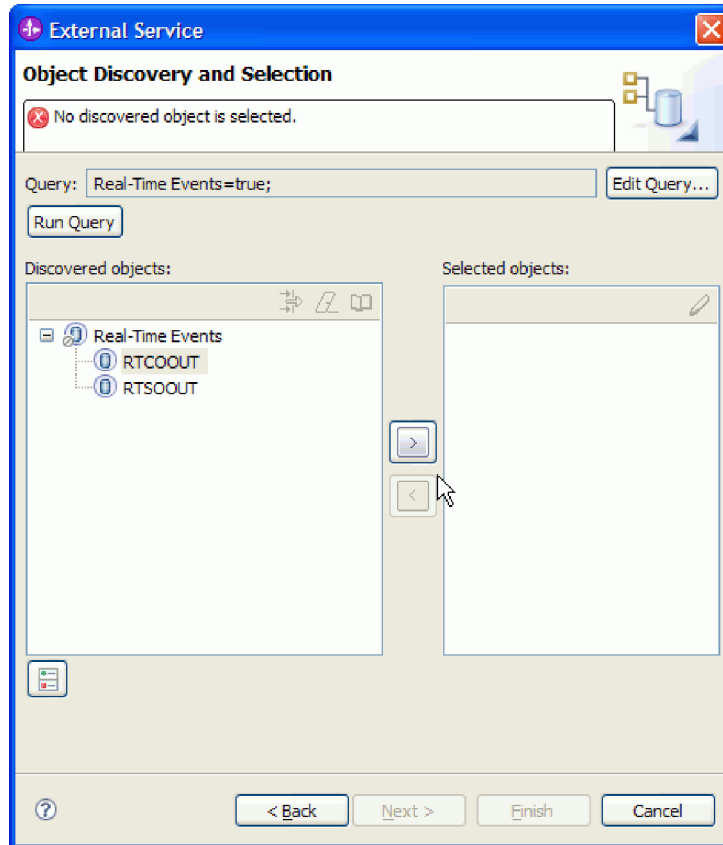


図 30. 矢印をクリックして *RTCOOUT* リアルタイム・イベントを「選択済みオブジェクト」リストに追加する

5. 「構成プロパティ (Configuration Properties)」ウィンドウでは、選択したリアルタイム・イベントに必要なアクションと操作を関連付けることができます。これは、まだリアルタイム・イベントのアクションを保管する操作フィールドがないリアルタイム・イベントに必要な作業です。リアルタイム・イベントに既に操作フィールドがある場合は、必要なのは操作名の指定だけで、関連付けられたアクションを含むフィールドは必要ありません。
 - 操作フィールドを選択したリアルタイム・イベントに関連付けるには、以下の手順を使用します。
 - a. 「選択」フィールドで、「操作フィールドのある構成」を選択します。以下の図に示されているように、「構成プロパティ (Configuration Properties)」ウィンドウが展開されます。



図 31. 「構成プロパティ (Configuration Properties)」 ウィンドウ

- b. 展開されたウィンドウで、以下のフィールドに情報を入力します。
 - 子のリスト: コンテナ・リアルタイム・イベントまたは単一のリアルタイム・イベント自体に組み込まれているすべての単一リアルタイム・イベント
 - 操作フィールド: 操作フィールドには、その操作に関連付けられたアクションの情報が保管されます。
 - 1) 「選択」をクリックし、操作フィールドの値を選択します。
 - 2) 「選択」ウィンドウで、操作のアクションを選択し、「OK」をクリックします。

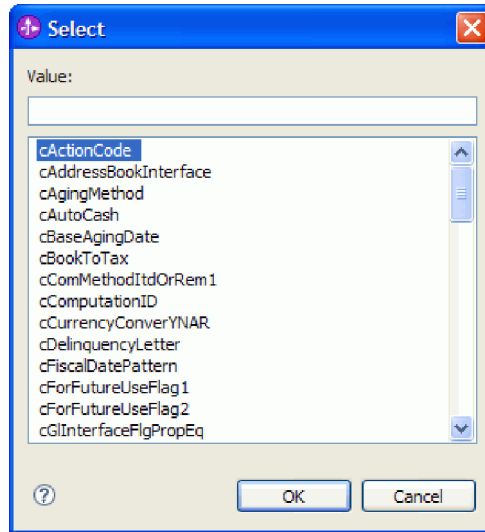


図 32. 選択可能なアクションを示す「選択」ウィンドウ

- User-defined: Create: Create 操作を選択したアクションにマッピングして使用する場合は、デフォルト番号 ("1") のままにします。そうでない場合は、新規の番号を指定します。
 - User-defined: Update: Update 操作を選択したアクションにマッピングして使用する場合は、デフォルト番号 ("2") のままにします。そうでない場合は、新規の番号を指定します。
 - User-defined: Delete: Delete 操作を選択したアクションにマッピングして使用する場合は、デフォルト番号 ("3") のままにします。そうでない場合は、新規の番号を指定します。
- c. 「OK」をクリックします。
- 操作フィールドを関連付けずにリアルタイム・イベントを選択するには、以下の手順を使用します。
 - a. 「選択」フィールドで、「操作フィールドなしの構成」を選択します。
 - b. 「構成プロパティ (Configuration Properties)」ウィンドウで、選択したリアルタイム・イベントで使用する操作を指定します。

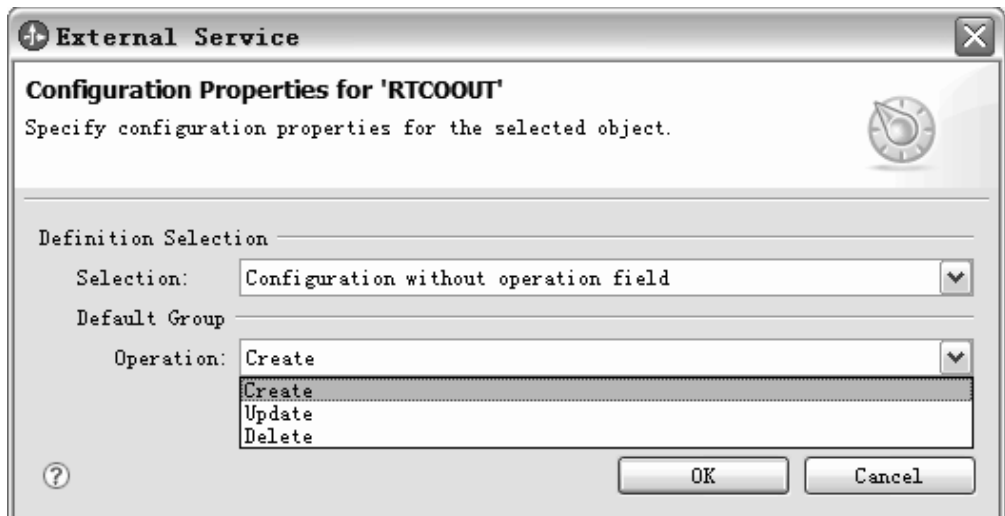


図 33. 「構成プロパティ (Configuration Properties)」 ウィンドウ

c. 「OK」 をクリックします。

選択したリアルタイム・イベントが、「選択済みオブジェクト」リストに追加されます。

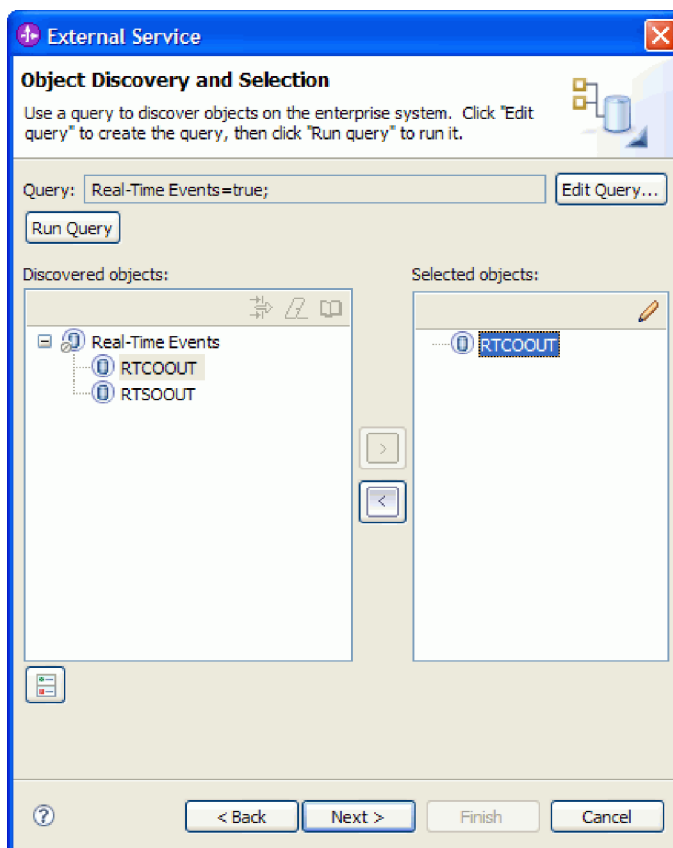


図 34. 選択したリアルタイム・イベントが表示された「選択済みオブジェクト」リスト

6. 「次へ」 をクリックします。

結果

以上で、作業を行うリアルタイム・イベントを選択し、適切な操作マッピング関係を構成しました。

次のタスク

「複合プロパティの構成 (Configure Composite Properties)」ウィンドウで、リアルタイム・イベントを保管するディレクトリーを指定します。

選択されたオブジェクトの構成

リアルタイム・イベントを構成するため、オブジェクトに関する情報を指定します。

始める前に

リアルタイム・イベントを選択し、インポートしたことを確認します。

このタスクを実行する理由および時期

リアルタイム・イベントを構成するには、以下の手順を使用します。

このタスクの手順

1. 「複合プロパティの構成 (Configure Composite Properties)」ウィンドウの「**相対パス**」フィールドで、ビジネス・オブジェクトを保管するディレクトリーを指定します。
2. 「**次へ**」をクリックします。

結果

リアルタイム・イベントを保管する場所を指定しました。「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウが表示されます。

次のタスク

アダプターとリアルタイム・イベントを組み込むデプロイ可能なモジュールを生成します。

デプロイメント・プロパティの設定およびサービスの生成

モジュール (WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイされる成果物) を生成するため、モジュールを作成し、モジュールにアダプターを関連付けます。

始める前に

リアルタイム・イベントを構成したことを確認してください。「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウが表示されるはずですが。

このタスクを実行する理由および時期

アダプターと構成済みのビジネス・オブジェクトを組み込むモジュールを生成します。モジュールとは、サーバーに配置する成果物です。

モジュールを生成するには、以下の手順を使用します。

このタスクの手順

1. オプション: デフォルトの操作名を変更するには、以下の手順を使用します。
 - a. 「**操作の編集**」をクリックします。
 - b. 「**操作名の編集**」ウィンドウで、編集する操作を選択し、「**編集**」をクリックします。
 - c. 「**プロパティの追加/編集 (Add/Edit properties)**」ウィンドウで、新規の操作名とオプションの説明を入力し、「**終了**」をクリックします。
2. JD Edwards EnterpriseOne サーバー への接続を確立するために、認証別名を (ユーザー ID とパスワードを入力する代わりに) 使用するかどうかを示します。
 - 認証別名を指定する場合は、「**Java Authentication and Authorization Service (JAAS) の別名セキュリティ・クリデンシャルの指定 (Specify a Java Authentication and Authorization Services (JAAS) alias security credential)**」を選択したままにします。次に「**J2C 認証データ項目**」フィールドに、管理コンソールの「**セキュリティ**」セクションで指定されている名前を入力します。
 - 認証別名を使用しない場合は、「**Java Authentication and Authorization Service (JAAS) の別名セキュリティ・クリデンシャルの指定 (Specify a Java Authentication and Authorization Services (JAAS) alias security credential)**」をクリアします。
3. 「**単一アプリケーションが使用するモジュールで (With module for use by single application)**」を選択して、アプリケーション・サーバーにデプロイされるモジュールにアダプター・ファイルを組み込むか、「**複数アプリケーションが使用するサーバー上(On server for use by multiple applications)**」を選択して、スタンドアロン・アダプターとしてアダプター・ファイルをインストールします。
 - 「**単一アプリケーションが使用するモジュールとともにデプロイする**」。アダプター・ファイルをモジュール内に組み込むと、モジュールをあらゆるアプリケーション・サーバーにデプロイすることができます。組み込みアダプターを使用するのは、組み込みアダプターを使用するモジュールが 1 つある場合か、複数のモジュールでバージョンの異なるアダプターを実行する必要がある場合です。組み込みアダプターを使用すると、他のモジュールのアダプター・バージョンを変更することで、それらのモジュールを不安定にするリスクを生じることなく、1 つのモジュール内でアダプターをアップグレードできます。
 - 「**複数アプリケーションが使用するサーバー上**」。モジュール内にアダプター・ファイルを組み込まない場合は、このモジュールを実行するアプリケーション・サーバーごとにモジュールをスタンドアロン・アダプターとしてインストールする必要があります。複数のモジュールが同じバージョンのアダプターを使用可能で、アダプターを中央の場所で管理する場合は、スタンドアロン・アダプターを使用します。スタンドアロン・アダプターの場合も、

複数のモジュールに対して単一のアダプター・インスタンスを実行することにより、必要なリソースが軽減されます。

4. 前のステップで「**複数アプリケーションが使用するサーバー上(On server for use by multiple applications)**」を選択した場合は、「**接続プロパティ**」リストがアクティブになります。以下のうちから 1 つを選択します。
 - 構成情報をここで指定する場合は、「**接続プロパティの指定 (Specify connection properties)**」を選択します。ステップ 5 に進みます。
 - 既存の接続ファクトリー構成を使用する場合は、「**定義済みの接続プロパティの使用 (Use predefined connection properties)**」を選択します。

定義済みの接続プロパティを使用する場合は、ご使用のリソース・アダプター名とインストールされているアダプター名を一致させる必要があります。これが、アダプターのインスタンスをこれらのプロパティに関連付ける方法であるためです。インポートまたはエクスポートのリソース・アダプターの名前を変更したい場合は、WebSphere Integration Developer のアセンブリー・エディターを使用して、インポートまたはエクスポートの値を変更します。

「**定義済みの接続プロパティの使用 (Use predefined connection properties)**」を選択すると、プロパティの代わりに「**JNDI ルックアップ名 (JNDI Lookup Name)**」フィールドが表示されます。

- a. 「**JNDI ルックアップ名 (JNDI Lookup Name)**」に値を入力します。
 - b. 「**次へ**」をクリックします。
 - c. ステップ 9 に進みます。
5. 「**接続プロパティ**」セクションで、ご使用の構成に適用される接続プロパティがあれば、それを設定または変更します。

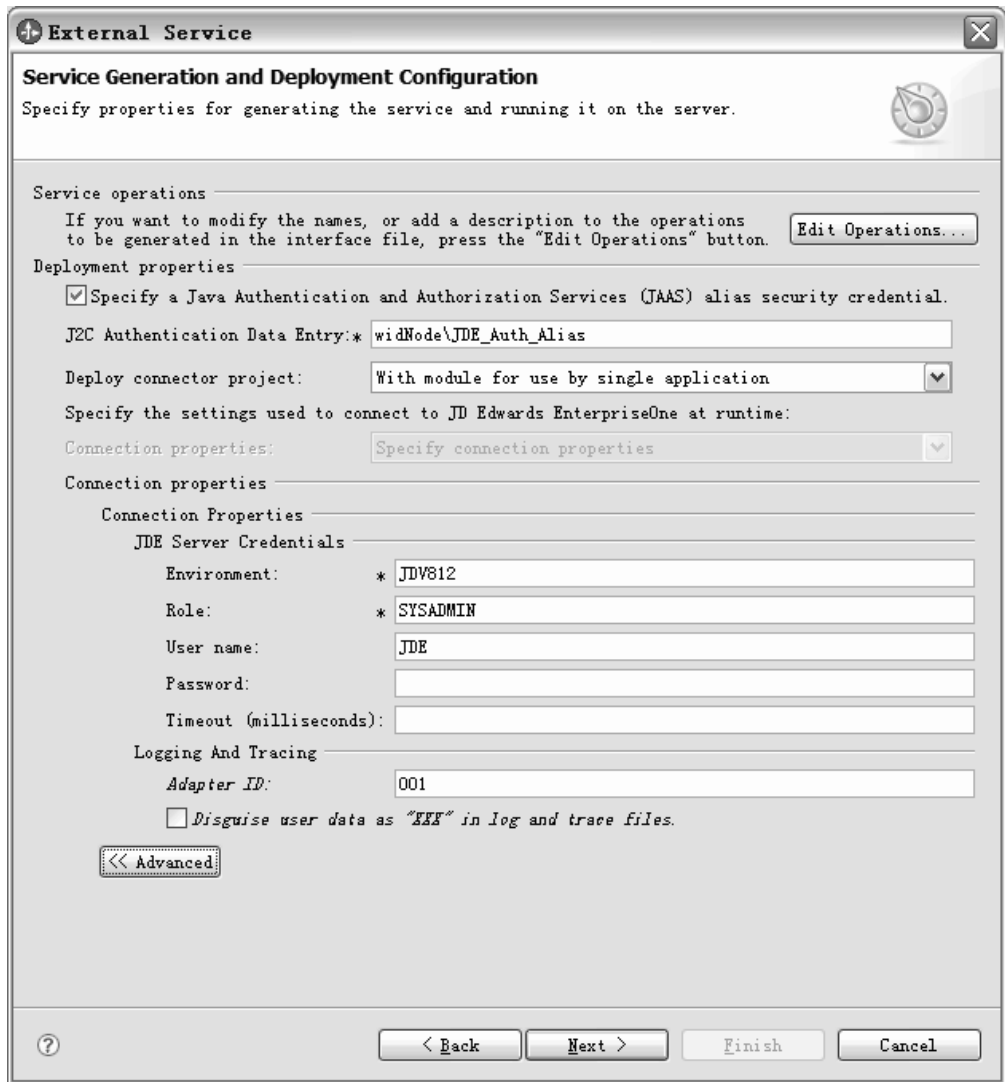


図 35. 「接続プロパティ」セクションを表示している「サービスの生成およびデプロイメントの構成 (Service Generation and Deployment Configuration)」ウィンドウ

これらのプロパティの詳細については、136 ページの『Managed Connection Factory プロパティ』を参照してください。

アスタリスク (*) が付いているプロパティは必須です。

6. その他のプロパティを設定するには、「拡張」をクリックし、次のセクションを展開します。

- イベント・ポーリング構成

オプションで、開始時に失敗した接続をアダプターで再試行するようにする場合は、「開始時に EIS 接続を再試行する」を選択することができます。詳しくは、157 ページの『開始時に EIS 接続を再試行する (RetryConnectionOnStartup)』を参照してください。

- ロギングおよびトレース

- 複数のアダプター・インスタンスがある場合は、アダプター ID をこのインスタンスに固有の値に設定します。

7. 「次へ」をクリックします。
8. 新規のモジュールを作成します。
 - a. 「サービス・ロケーション・プロパティ (Service Location Properties)」ウィンドウの「モジュール (Module)」フィールドで「新規作成」をクリックします。
 - b. 「統合プロジェクト (Integration Project)」ウィンドウで、「モジュール・プロジェクトの作成 (Create a module project)」または「メディエーション・モジュール・プロジェクトの作成 (Create a mediation module project)」を選択して、「次へ」をクリックします。
 - c. 「新規モジュール (New Module)」ウィンドウで、モジュールの名前を入力します。

注: 名前を入力すると、「ロケーション (Location)」フィールドで指定されているワークスペース・ディレクトリーにその名前が追加されます。

これはデフォルトのロケーションです。別のロケーションを指定する場合は、「デフォルト・ロケーションを使用する (Use default location)」からチェック・マークを外し、新しいロケーションを入力するか、または、「参照」をクリックしてロケーションを選択します。

- d. アセンブリー・ダイアグラムでモジュールを開くか (モジュール・プロジェクトの場合)、またはメディエーション・フロー・コンポーネントを作成するか (メディエーション・モジュール・プロジェクトの場合) を指定します。デフォルトでは、これらの選択項目は選択されています。
 - e. 「終了」をクリックします。新規のモジュールが作成されます。作成プロセスが終了したら、「新規モジュール (New Module)」ウィンドウが閉じ、新規のモジュールが「サービス・ロケーション・プロパティ (Service Location Properties)」ウィンドウの「モジュール (Module)」リストに表示されます。
9. 「サービス・ロケーション・プロパティ (Service Location Properties)」ウィンドウで、以下の手順を実行すれば、新規モジュールのデプロイメント・プロパティを指定できます。
 - a. デフォルトの名前空間を変更する場合は、「デフォルト名前空間を使用する (Use Default Namespace)」チェック・ボックスをクリアして、「名前空間」フィールドに新しいパスを入力します。
 - b. サービス記述を保存するモジュール内のフォルダーを指定します。そのためには、「フォルダー (Folder)」フィールドに名前を入力するか、フォルダーを見つけて選択します。
 - c. インターフェースの名前を変更します。

デフォルト名は「JDEOutboundInterface」です。希望する場合、これをより説明的なタイトルに変更することができます。

- d. ビジネス・オブジェクトを別のアプリケーションが使用できるように保存する場合は、「ビジネス・オブジェクトをライブラリーに保存 (Save business objects to a library)」をクリックしてからリストからライブラリーを選択するか、または「新規作成」をクリックして、新規ライブラリーを作成します。

- e. モジュールの説明を入力します。
10. 「終了」をクリックします。

結果

これで、WebSphere Integration Developer の Business Integration パースペクティブに、新しいモジュールが作成され、構成、保存されました。

次のタスク

配置のための EAR ファイルとしてモジュールをエクスポートします。

第 5 章 アセンブリー・エディターによる対話仕様プロパティの変更

サービスの生成後にアダプター・モジュールの対話仕様プロパティを変更するには、WebSphere Integration Developer のアセンブリー・エディターを使用します。

始める前に

アダプターに対してサービスを生成するには、あらかじめ外部サービス・ウィザードを使用しておく必要があります。

このタスクを実行する理由および時期

アダプターのサービスを生成後に、対話仕様プロパティの変更が必要になる場合があります。対話仕様プロパティはオプションですが、特定のビジネス・オブジェクトの特定の操作に対して、メソッド・レベルで設定されます。指定した値は、外部サービス・ウィザードによって生成されるすべての親ビジネス・オブジェクトのデフォルトとして表示されます。これらのプロパティは、EAR ファイルをエクスポートする前に変更できます。アプリケーションをデプロイした後にこれらのプロパティを変更することはできません。

対話仕様プロパティを変更するには、以下の手順を実行します。

このタスクの手順

1. WebSphere Integration Developer の Business Integration パースペクティブで、モジュール名を展開します。
2. 「アセンブリー・ダイアグラム」を展開して、インターフェースをダブルクリックします。
3. アセンブリー・エディターでインターフェースをクリックします。(追加のクリックをしない限り、モジュールのプロパティが表示されています。)
4. 「プロパティ」タブをクリックします。(ダイアグラム内でインターフェースを右クリックし、「プロパティを表示」をクリックすることもできます。)
5. 「バインディング」で、「メソッド・バインディング」をクリックします。インターフェースのメソッドが、ビジネス・オブジェクトと操作の組み合わせごとに 1 つずつ表示されます。
6. 変更する対話仕様プロパティを持つメソッドを選択します。
7. 「汎用」タブでプロパティを変更します。変更する対話仕様プロパティを持つメソッドごとにこの手順を繰り返します。

結果

アダプター・モジュールに関連付けられている対話仕様プロパティが変更されました。

次のタスク

モジュールをデプロイします。

第 6 章 モジュールのデプロイ

モジュールをデプロイし、モジュールおよびアダプターを構成するファイルを、実稼働またはテストのための動作環境に配置します。WebSphere Integration Developer では、統合テスト環境は、インストール時に選択したテスト環境プロファイルに応じて、WebSphere Process Server または WebSphere Enterprise Service Bus、あるいはその両方に対する実行時サポート機能を備えています。

デプロイメント環境

モジュールおよびアダプターのデプロイ先には、テスト環境と実稼働環境があります。

WebSphere Integration Developerでは、モジュールをテスト環境内の 1 つ以上のサーバーに追加できます。通常は、これがビジネス・インテグレーション・モジュールの実行およびテストを行うための最も一般的な手法です。ただし、WebSphere Process Server または WebSphere Enterprise Service Bus 上で管理コンソールまたはコマンド行ツールを使用して、サーバーへのデプロイメント用のモジュールを EAR ファイルとしてエクスポートすることもできます。

テスト用のモジュールのデプロイ

WebSphere Integration Developer では、組み込みアダプターを内蔵するモジュールをテスト環境にデプロイし、サーバー構成の編集、サーバーの始動および停止、モジュール・コードのテストによるエラー有無の確認などの作業を実行できるサーバー・ツールと連携できます。テストは通常、コンポーネントのインターフェース操作について実行されますが、このテストを実行すると、コンポーネントが正しく実装されているかどうかを判断できます。

外部依存関係の追加

依存関係のある JAR は、ライブラリー・ディレクトリーに追加するか、または EAR にパッケージ化する必要があります。

このタスクを実行する理由および時期

JAR は、クラス・パスに設定され、これら依存関係のあるライブラリーは、モジュールのデプロイ時に、ランタイムに使用できるようにする必要があります。依存関係のあるライブラリーを使用可能にする方法には、スタンドアロン・デプロイメントまたは組み込みデプロイメントのいずれの場合にも対応する方法と、組み込みデプロイメントのみに対応する方法の 2 つの方法があります。

サーバー・ランタイム環境への外部ソフトウェア依存関係の追加

アダプター・アプリケーションを実行するには、まず、必要な外部ソフトウェア依存関係ファイルをご使用のランタイム環境にコピーする必要があります。

このタスクを実行する理由および時期

必要なファイルを WebSphere Process Server または WebSphere Enterprise Service Bus にコピーするには、以下の手順に従います。

このタスクの手順

1. 外部ソフトウェア依存関係ファイルを JD Edwards EnterpriseOne 管理者から入手します。ファイルは、表 4 にリストされています。

表 4. JD Edwards EnterpriseOne Tools で必要とされる外部ソフトウェア依存関係ファイル

JD Edwards EnterpriseOne Tools、バージョン 8.9 (SP1、SP2)、 8.93	JD Edwards EnterpriseOne Tools、バージョン 8.94	JD Edwards EnterpriseOne Tools、バージョン 8.95、 8.96
connector.jar	Common_Jar.jar	ApplicationAPIs_JAR.jar
database.jar	Connector.jar	ApplicationLogic_JAR.jar
jdeinterop.ini	database.jar	Base_JAR.jar
jdeLog.properties	EventProcessor_EJB.jar	BizLogicContainer_JAR.jar
kernel.jar	jdeutil.jar	BizLogicContainerClient_JAR.jar
log4j.jar	jdbj.ini	bootstrap.jar
owra.jar	jdeinterop.ini	castor.jar
xalan.jar	jdelog.properties	Connector.jar
xerces.jar	kernel.jar	ecutils.jar
JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar 	log4j.jar	EventProcessor_JAR.jar
	xalan.jar	EventProcessor_EJB.jar
	xerces.jar	jdbj.ini
	JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar 	JdbjBase_JAR.jar
		JdbjInterfaces_JAR.jar
		jdeinterop.ini
		jdelog.properties
		JdeNet_JAR.jar
		Improxy.jar
		log4j.jar
		messagingClient.jar
		naming.jar

表 4. JD Edwards EnterpriseOne Tools で必要とされる外部ソフトウェア依存関係ファイル (続き)

JD Edwards EnterpriseOne Tools、バージョン 8.9 (SP1、SP2)、 8.93	JD Edwards EnterpriseOne Tools、バージョン 8.94	JD Edwards EnterpriseOne Tools、バージョン 8.95、8.96
		PMApi_JAR.jar
		Spec_JAR.jar
		System_JAR.jar
		urlprotocols.jar
		xalan.jar
		xerces.jar
		JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar

2. ファイルをサーバーにコピーします。

- WebSphere Integration Developer のテスト環境の場合は、ファイルを `${WAS_INSTALL_ROOT}/runtimes/bi_v62/lib/ext` ディレクトリーにコピーします。
- 実稼働環境の場合は、ファイルを WebSphere Process Server または WebSphere Enterprise Service Bus の `${WAS_INSTALL_ROOT}/lib/ext` ディレクトリーにコピーします。

結果

これで、JD Edwards EnterpriseOne の外部ソフトウェア依存関係ファイルが、ご使用のランタイム環境に組み込まれました。

アダプターがバンドルされている場合の外部ソフトウェア依存関係の追加

アダプター・アプリケーションを実行するには、まず、依存関係のある JAR ファイルを EAR アプリケーションにコピーする必要があります。この方法を使用するのは、組み込みデプロイメントの場合に限られます。

このタスクを実行する理由および時期

必要なファイルを手入手して、それらを EAR アプリケーションにコピーするには、以下の手順に従います。

このタスクの手順

1. 該当するモジュールから、ワークスペースに進み、JAR ファイルをディレクトリーにコピーします。例えば、モジュール名が「ModuleName」の場合、ワークスペースに進んで、JAR ファイルを `ModuleNameApp/EarContent` ディレクトリーにコピーします。

2. アダプター RAR のマニフェスト・ファイル、manifest.mf を、アダプターが必要とする JAR ファイルのリストで変更します。以下の書式で JAR ファイルを追加します: Class-Path: dependantjar1.jar, dependantjar2.jar
3. ネイティブ・ライブラリーをランタイム bin ディレクトリーにコピーし、アプリケーションをデプロイします。

結果

これでサード・パーティー・ライブラリーが、ご使用のランタイム環境に含まれるようになりました。

Inbound 処理をテストするためのターゲット・コンポーネントの生成および接続

Inbound 処理用のアダプターが組み込まれているモジュールをテスト環境にデプロイする前に、まずターゲット・コンポーネントを生成して接続する必要があります。このターゲット・コンポーネントは、アダプターがイベントを送信する宛先として機能します。

始める前に

外部サービス・ウィザードを使用してエクスポート・モジュールを生成してあるはずです。

このタスクを実行する理由および時期

Inbound 処理のためにターゲット・コンポーネントを生成して接続する必要があるのは、テスト環境のみです。実稼働環境でアダプターを配置する際には必要ありません。

ターゲット・コンポーネントは、イベントを受信します。WebSphere Integration Developer のアセンブリー・エディターを使用して、エクスポート・コンポーネントを (2 つのコンポーネントを接続している) ターゲット・コンポーネントに接続します。アダプターはこのワイヤーを使用して、(エクスポート・コンポーネントからターゲット・コンポーネントへ) イベント・データを受け渡します。

このタスクの手順

1. ターゲット・コンポーネントを作成します。
 - a. WebSphere Integration Developer の Business Integration パースペクティブで、「アセンブリー・ダイアグラム」を展開して、エクスポート・コンポーネントをダブルクリックします。デフォルト値を変更しなかった場合、エクスポート・コンポーネントの名前は、ご使用のアダプター + **InboundInterface** になります。

インターフェースにより、呼び出すことができる操作と渡されるデータ (入力引数、戻り値、例外など) が指定されます。 **InboundInterface** コンポーネントには、Inbound 処理をサポートするためにアダプターが必要とする操作が格納されています。また、このコンポーネントは 外部サービス・ウィザード を実行すると作成されます。

- b. 「コンポーネント」を展開して「型なしコンポーネント」を選択し、そのコンポーネントをアセンブリ・ダイアグラムまでドラッグして、新規コンポーネントを作成します。

カーソルが配置アイコンに変わります。
 - c. コンポーネントをクリックして、そのコンポーネントをアセンブリ・ダイアグラムに表示します。
2. コンポーネントを接続します。
 - a. エクスポート・コンポーネントをクリックして、新規コンポーネントにドラッグします。 これにより、次の図に示すように、エクスポート・コンポーネントから新規コンポーネントへ線を引くことができます。
 - b. アセンブリ・ダイアグラムを保存します。「ファイル」→「保管」をクリックします。
 3. 新規コンポーネントの実装を生成します。
 - a. 新規コンポーネントを右クリックして、「実装の生成」を選択します。

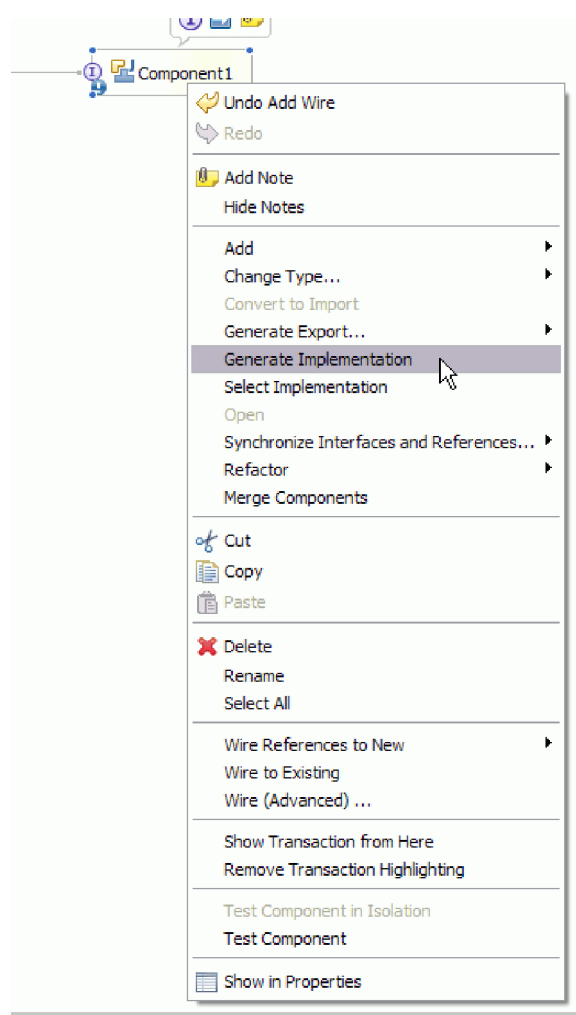


図 36. Java 実装環境の生成

- b. 「(デフォルト・パッケージ)」を選択して、「OK」をクリックします。これにより、Inbound モジュールのエンドポイントが作成されます。

別のタブに Java 実装環境が表示されます。

- c. **オプション:** print ステートメントを追加して、エンドポイント・メソッドのそれぞれのエンドポイントで受信したデータ・オブジェクトを出力します。
- d. 「ファイル」 → 「保管」をクリックして、変更内容を保存します。

次のタスク

テストを行うモジュールのデプロイを続行します。

サーバーへのモジュールの追加

WebSphere Integration Developer では、モジュールをテスト環境内の 1 つ以上のサーバーに追加できます。

このタスクを実行する理由および時期

ご使用のモジュールと、モジュールによるアダプターの使用をテストするため、そのモジュールをサーバーに追加する必要があります。

このタスクの手順

1. サーバーにモジュールを追加します。
 - a. 「サーバー・ビュー」に切り替えます。 WebSphere Integration Developer で、「ウィンドウ」 → 「ビューの表示」 → 「サーバー」を選択します。
 - a. サーバーを始動します。 WebSphere Integration Developer 画面の右下のペインにある「サーバー」タブで、サーバーを右クリックし、「開始」を選択します。
2. サーバーの状況が「開始済み」である場合は、サーバーを右クリックし、「プロジェクトの追加および除去」を選択します。
3. 「プロジェクトの追加および除去」画面で、対象のプロジェクトを選択して「追加」をクリックします。 プロジェクトは、「使用可能プロジェクト」のリストから「構成プロジェクト」のリストに移動します。
4. 「終了」をクリックします。 これにより、モジュールがサーバーにデプロイされます。

モジュールがサーバーに追加されている間に、右下のペインの「コンソール」タブに、ログが表示されます。

次のタスク

モジュールおよびアダプターの機能をテストします。

テスト・クライアントを使用した Outbound 処理用モジュールのテスト

Outbound 処理用のアセンブル済みモジュールおよびアダプターを、WebSphere Integration Developer の統合テスト・クライアントを使用してテストします。

始める前に

最初に、モジュールをサーバーに追加する必要があります。

このタスクを実行する理由および時期

モジュールのテストは、通常、コンポーネントのインターフェース操作について実行されますが、このテストを実行すると、コンポーネントが正しく実装され、参照先が正しく接続されているかどうかを判断できます。

このタスクの手順

1. テストするモジュールを選択し、右クリックして、「テスト」 → 「テスト・モジュール」を選択します。
2. テスト・クライアントを使用したモジュールのテストについて詳しくは、WebSphere Integration Developer インフォメーション・センターの『モジュールおよびコンポーネントのテスト』のトピックを参照してください。

次のタスク

ご使用のモジュールおよびアダプターのテスト結果に納得したら、モジュールおよびアダプターを実稼働環境にデプロイできます。

実稼働用のモジュールのデプロイ

外部サービス・ウィザードを使用して作成したモジュールを、実稼働環境で WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイする処理は、2 段階構成になっています。最初に、WebSphere Integration Developer 内にモジュールをエンタープライズ・アーカイブ (EAR) ファイルの形でエクスポートします。次に、WebSphere Process Server または WebSphere Enterprise Service Bus 管理コンソール を使用して、EAR ファイルをデプロイします。

このタスクを実行する理由および時期

サーバー・ランタイム環境への外部ソフトウェア依存関係の追加

アダプター・アプリケーションを実行するには、まず、必要な外部ソフトウェア依存関係ファイルをご使用のランタイム環境にコピーする必要があります。

このタスクを実行する理由および時期

必要なファイルを WebSphere Process Server または WebSphere Enterprise Service Bus にコピーするには、以下の手順に従います。

このタスクの手順

1. 外部ソフトウェア依存関係ファイルを JD Edwards EnterpriseOne 管理者から入手します。ファイルは、88 ページの表 5 にリストされています。

表 5. JD Edwards EnterpriseOne Tools で必要とされる外部ソフトウェア依存関係ファイル

JD Edwards EnterpriseOne Tools、バージョン 8.9 (SP1、SP2)、 8.93	JD Edwards EnterpriseOne Tools、バージョン 8.94	JD Edwards EnterpriseOne Tools、バージョン 8.95、 8.96
connector.jar	Common_Jar.jar	ApplicationAPIs_JAR.jar
database.jar	Connector.jar	ApplicationLogic_JAR.jar
jdeinterop.ini	database.jar	Base_JAR.jar
jdeLog.properties	EventProcessor_EJB.jar	BizLogicContainer_JAR.jar
kernel.jar	jdeutil.jar	BizLogicContainerClient_JAR.jar
log4j.jar	jdbj.ini	bootstrap.jar
owra.jar	jdeinterop.ini	castor.jar
xalan.jar	jdelog.properties	Connector.jar
xerces.jar	kernel.jar	ecutils.jar
JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar 	log4j.jar	EventProcessor_JAR.jar
	xalan.jar	EventProcessor_EJB.jar
	xerces.jar	jdbj.ini
JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar 	JDBC ドライバー・ファイル	JdbjBase_JAR.jar
		JdbjInterfaces_JAR.jar
		jdeinterop.ini
		jdelog.properties
		JdeNet_JAR.jar
		Improxy.jar
		log4j.jar
		messagingClient.jar
		naming.jar
		PMApi_JAR.jar
		Spec_JAR.jar
		System_JAR.jar
		urlprotocols.jar
		xalan.jar

表 5. JD Edwards EnterpriseOne Tools で必要とされる外部ソフトウェア依存関係ファイル (続き)

JD Edwards EnterpriseOne Tools、バージョン 8.9 (SP1、SP2)、 8.93	JD Edwards EnterpriseOne Tools、バージョン 8.94	JD Edwards EnterpriseOne Tools、バージョン 8.95、8.96
		xerces.jar
		JDBC ドライバー・ファイル 例えば、Oracle データベース・サーバーを使用している場合、以下の JDBC ドライバー・ファイルを使用します。 <ul style="list-style-type: none"> • tnsnames.ora • ojdbc14.jar

2. ファイルをサーバーにコピーします。

- WebSphere Integration Developer のテスト環境の場合は、ファイルを `${WAS_INSTALL_ROOT}/runtimes/bi_v62/lib/ext` ディレクトリーにコピーします。
- 実稼働環境の場合は、ファイルを WebSphere Process Server または WebSphere Enterprise Service Bus の `${WAS_INSTALL_ROOT}/lib/ext` ディレクトリーにコピーします。

結果

これで、JD Edwards EnterpriseOne の外部ソフトウェア依存関係ファイルが、ご使用のランタイム環境に組み込まれました。

RAR ファイルのインストール (スタンドアロン・アダプターを使用するモジュールの場合のみ)

アダプターをモジュールに組み込まないが、サーバー・インスタンス内にデプロイされているすべてのアプリケーションに対してアダプターを使用可能にする場合は、アダプターを RAR ファイルの形式でアプリケーション・サーバーにインストールすることが必要になります。RAR ファイルとは、Java 2 Connector (J2C) アーキテクチャーに合わせてリソース・アダプターを圧縮するとき使用する Java アーカイブ (JAR) ファイルのことです。

始める前に

外部サービス・ウィザードの「サービス生成およびデプロイメント構成 (Service Generation and Deployment Configuration)」ウィンドウで、「コネクタ・プロジェクトのデプロイ」を「複数アダプターが使用するサーバー上 (On server for use by multiple adapters)」に設定してある必要があります。

このタスクを実行する理由および時期

アダプターを RAR ファイルの形式でインストールすると、そのアダプターは、サーバー・ランタイムで実行されているすべての J2EE アプリケーション・コンポーネントで使用可能になります。

このタスクの手順

1. 管理コンソール を開始します。
2. 「リソース」 → 「リソース・アダプター」 → 「リソース・アダプター」 をクリックします。
3. 「リソース・アダプター」 ページで、「RAR のインストール」 をクリックします。

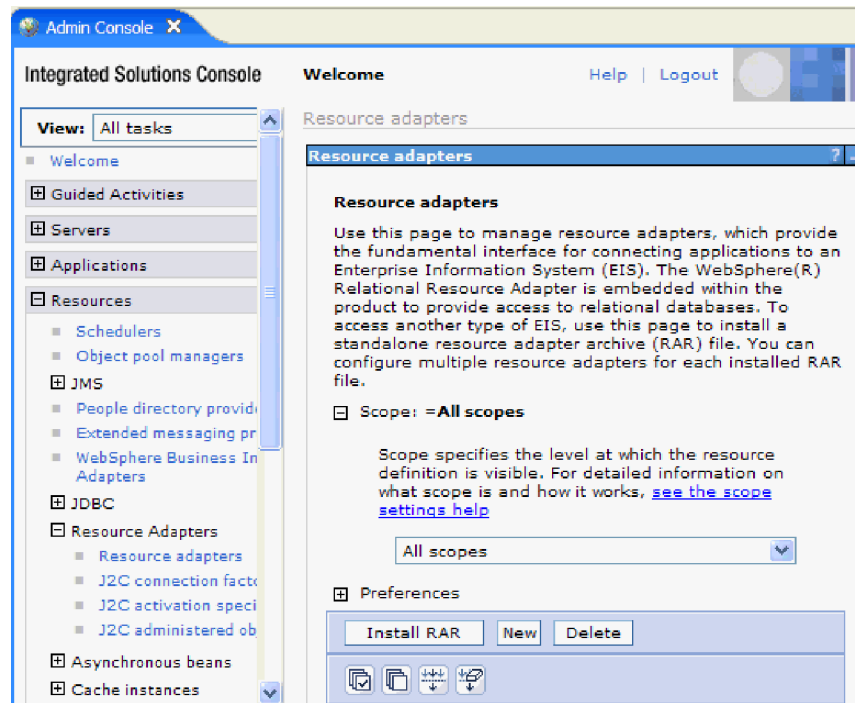


図 37. 「リソース・アダプター」 ページの「RAR のインストール」 ボタン

4. 「RAR ファイルのインストール」 ページで、「参照」 をクリックし、ご使用のアダプターの RAR ファイルへ移動します。

RAR ファイルは、通常、`WID_installation_directory/ResourceAdapters/adapter_name/deploy/adapter.rar` のパスにインストールされます。

5. 「次へ」 をクリックします。
6. 「リソース・アダプター」 ページで、必要に応じてアダプターの名前を変更し、説明を追加します。
7. 「OK」 をクリックします。
8. ページの上部にある「メッセージ」 ボックスで「保管」 をクリックします。

次のタスク

次の手順は、サーバーにデプロイできる EAR ファイルとしてモジュールをエクスポートすることです。

EAR ファイルとしてのモジュールのエクスポート

WebSphere Integration Developer を使用して、モジュールを EAR ファイルとしてエクスポートします。EAR ファイルを作成することによって、モジュールのすべての内容を WebSphere Process Server または WebSphere Enterprise Service Bus に容易にデプロイできる形式で取り込みます。

始める前に

モジュールを EAR ファイルとしてエクスポートするには、事前にサービスと通信するためのモジュールを作成しておく必要があります。このモジュールを、WebSphere Integration Developer ビジネス・インテグレーション・パースペクティブ内に表示する必要があります。

このタスクを実行する理由および時期

モジュールを EAR ファイルとしてエクスポートするには、以下の手順を実行します。

このタスクの手順

1. モジュールを右クリックして、「**エクスポート**」を選択します。
2. 「選択」ウィンドウで、「**Java EE**」を展開します。
3. 「**EAR ファイル**」を選択して、「**次へ**」をクリックします。
4. オプション: 正しい EAR アプリケーションを選択します。EAR アプリケーションにはモジュールと同じ名前が付けられますが、名前の末尾に「App」が追加されます。
5. EAR ファイルを格納するローカル・ファイル・システム上で、「**参照**」を選択してフォルダーを参照します。
6. 必要に応じて、ソース・ファイルをエクスポートする場合は、「**ソース・ファイルのエクスポート**」を選択します。このオプションは、EAR ファイルのほかにソース・ファイルをエクスポートする場合に表示されます。ソース・ファイルには、Java コンポーネント、データ・マップなどに関連付けられているファイルがあります。
7. 既存のファイルを上書きする場合は、「**既存のファイルを上書き (Overwrite an existing file)**」をクリックします。
8. 「**終了**」をクリックします。

結果

モジュールの内容が EAR ファイルとしてエクスポートされます。

このモジュールを 管理コンソール にインストールします。これにより、モジュールが WebSphere Process Server または WebSphere Enterprise Service Bus にデプロイされます。

EAR ファイルのインストール

EAR ファイルのインストールは、デプロイメント・プロセスの最終手順です。EAR ファイルをサーバーにインストールして実行すると、EAR ファイルの一部として組み込まれているアダプターが、インストール済みアプリケーションの一部として稼働します。

始める前に

モジュールを WebSphere Process Server または WebSphere Enterprise Service Bus にインストールするには、その前にモジュールを EAR ファイルとしてエクスポートしておく必要があります。

このタスクを実行する理由および時期

EAR ファイルをインストールするには、次の手順を実行します。アダプター・モジュール・アプリケーションのクラスター化については、<http://www.ibm.com/software/webservers/appserv/was/library/> を参照してください。

このタスクの手順

1. サーバー・インスタンスを右クリックし、「管理コンソール の実行」を選択して、WebSphere Process Server または WebSphere Enterprise Service Bus 管理コンソール を開きます。
2. 「管理コンソール」ウィンドウで、「アプリケーション」 → 「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。



図 38. 「アプリケーション・インストールの準備」ウィンドウ

3. 「参照」をクリックして、EAR ファイルを位置指定し、「次へ」をクリックします。EAR ファイル名は、モジュール名の後に「App」が付いたものです。
4. オプション: クラスター化された環境にデプロイする場合は、以下の手順を実行します。

- a. 「**ステップ 2: サーバーにモジュールをマップ**」ウィンドウで、モジュールを選択します。
 - b. サーバー・クラスターの名前を選択します。
 - c. 「**適用**」をクリックします。
5. 「**次へ**」をクリックして、「**要約**」を開きます。すべての設定が正しいことを確認して、「**終了**」をクリックします。
6. オプション: 認証別名を使用している場合は、以下の手順を実行します。
- a. 「**セキュリティ**」を展開して、「**ビジネス・インテグレーションの認証別名 (Business Integration Authentication Aliases)**」を選択します。
 - b. 構成する認証別名を選択します。認証別名の構成を変更するための管理者権限またはオペレーター権限を持っている必要があります。
 - c. オプション: 「**ユーザー名**」を入力します (まだ入力されていない場合)。
 - d. 「**パスワード**」を入力します (まだ入力されていない場合)。
 - e. 「**確認パスワード (Confirm Password)**」フィールドに再度パスワードを入力します (まだ入力されていない場合)。
 - f. 「**OK**」をクリックします。

結果

この時点で、プロジェクトがデプロイメントされ、「**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。

次のタスク

いずれかのプロパティを設定または再設定する場合、あるいは、アダプター・プロジェクトのアプリケーションをクラスター化したい場合は、トラブルシューティング・ツールを構成する前に、**管理コンソール** を使用して対応する変更を行ってください。

第 7 章 アダプター・モジュールの管理

アダプターをスタンドアロンのデプロイメントで稼働している場合は、アダプター・モジュールの開始、停止、モニター、およびトラブルシューティングには、サーバーの管理コンソールを使用します。組み込みアダプターを使用しているアプリケーションでは、アプリケーションの開始時または停止時にアダプター・モジュールが開始または停止します。

組み込みアダプターの構成プロパティーの変更

アダプターをモジュールの一部としてデプロイした後に構成プロパティーを変更するには、実行時環境の管理コンソールを使用します。リソース・アダプター・プロパティー (一般的なアダプター操作に使用)、Managed Connection Factory プロパティー (Outbound 処理に使用) を更新できます。

組み込みアダプターのリソース・アダプター・プロパティーの設定

アダプターをモジュールの一部としてデプロイした後に、このアダプターのリソース・アダプター・プロパティーを設定するには、管理コンソールを使用します。構成するプロパティーの名前を選択してから、その値を変更または設定します。

始める前に

アダプター・モジュールを WebSphere Process Server または WebSphere Enterprise Service Bus 上にデプロイする必要があります。

このタスクを実行する理由および時期

カスタム・プロパティーとは、すべての WebSphere アダプターが共用するデフォルト構成プロパティーです。

管理コンソールを使用してプロパティーを構成するには、以下の手順を実行します。

このタスクの手順

1. 管理コンソールを開始します。
2. 「アプリケーション」の下で、「エンタープライズ・アプリケーション」を選択します。
3. 「エンタープライズ・アプリケーション」リストから、プロパティーを変更するアダプター・モジュールの名前をクリックします。「構成」ページが表示されます。

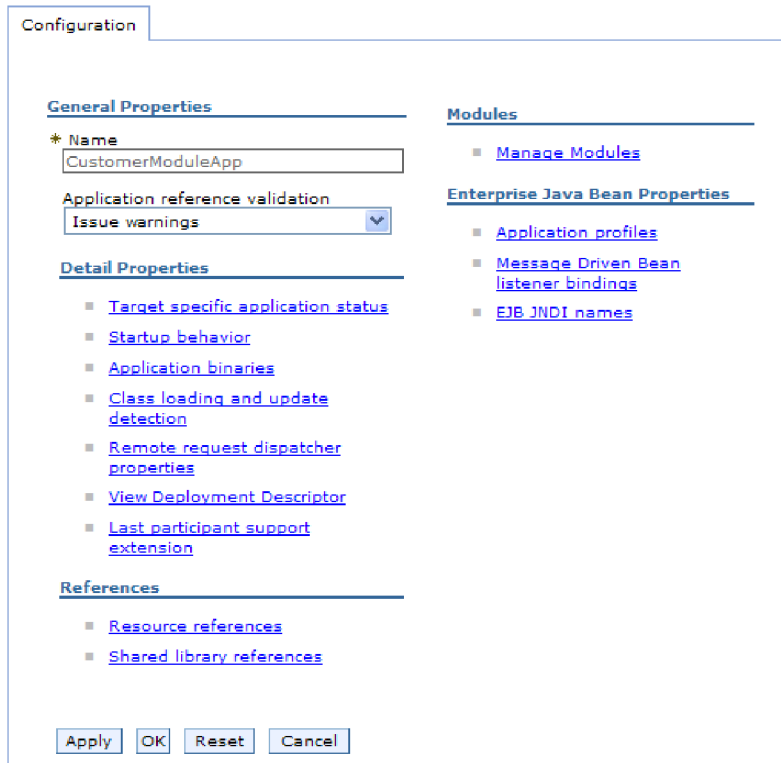


図 39. 「構成」タブでの「モジュールの管理」の選択

4. 「モジュール」の下で、「モジュールの管理」をクリックします。
5. 「**IBM WebSphere Adapter for JD Edwards EnterpriseOne**」をクリックします。
6. 「追加プロパティ」リストから、「リソース・アダプター」をクリックします。
7. 次のページで、「追加プロパティ」リストから、「カスタム・プロパティ」をクリックします。
8. 変更するプロパティごとに、以下の手順を実行します。

注: これらのプロパティの詳細については、132 ページの『リソース・アダプター・プロパティ』を参照してください。

- a. プロパティの名前をクリックします。選択したプロパティの「構成」ページが表示されます。例えば、「logNumberOfFiles」プロパティをクリックすると、下図に示されたページが表示されます。

Configuration

General Properties

* Scope
widNode

Required

Name
logNumberOfFiles

Value
1

Description

Type
java.lang.String

Apply OK Reset Cancel

図 40. logNumberOfFiles プロパティの「構成」タブ

- b. 「値」フィールドの値の内容を変更するか、フィールドが空の場合は値を入力します。
 - 「値」フィールドの数値を変更して、プロパティの説明を追加できます。
 - c. 「OK」をクリックします。
9. ウィンドウの上部にある「メッセージ」ボックス内の「保管」リンクをクリックします。

結果

アダプター・モジュールに関連付けられているリソース・アダプター・プロパティが変更されました。

組み込みアダプターの Managed (J2C) Connection Factory プロパティの設定

アダプターをモジュールの一部としてデプロイした後に、このアダプターの Managed Connection Factory プロパティを設定するには、管理コンソールを使用します。構成するプロパティの名前を選択してから、その値を変更または設定します。

始める前に

アダプター・モジュールを WebSphere Process Server または WebSphere Enterprise Service Bus 上にデプロイする必要があります。

このタスクを実行する理由および時期

Managed Connection Factory プロパティは、ターゲット・JD Edwards EnterpriseOne サーバーのインスタンスを構成する場合に使用します。

注: 管理コンソール内では、このプロパティを「J2C 接続ファクトリー・プロパティ」と呼びます。

管理コンソールを使用してプロパティを構成するには、以下の手順を実行します。

このタスクの手順

1. 管理コンソールを開始します。
2. 「アプリケーション」の下で、「エンタープライズ・アプリケーション」を選択します。
3. 「エンタープライズ・アプリケーション」リストで、プロパティを変更するアダプター・モジュールの名前をクリックします。
4. 「モジュール」の下で、「モジュールの管理」をクリックします。

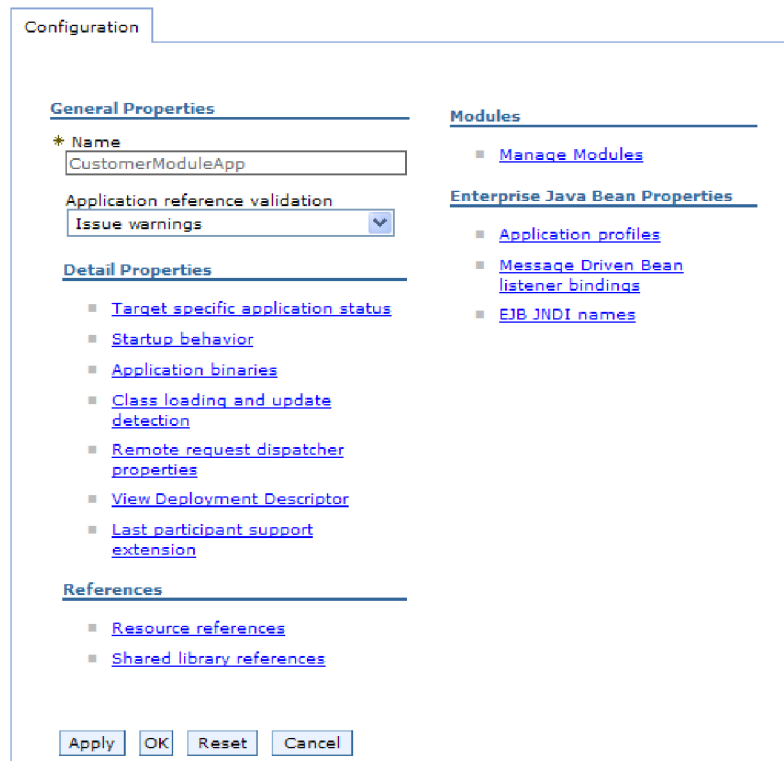


図 41. 「構成」タブでの「モジュールの管理」の選択

5. 「IBM WebSphere Adapter for JD Edwards EnterpriseOne」をクリックします。
6. 「追加プロパティ」リストで、「リソース・アダプター」をクリックします。
7. 次のページで、「追加プロパティ」リストから「J2C 接続ファクトリー」をクリックします。

8. アダプター・モジュールに関連付けられた接続ファクトリーの名前をクリックします。
9. 「追加プロパティ」リストで、「カスタム・プロパティ」をクリックします。

カスタム・プロパティは、Adapter for JD Edwards EnterpriseOne に特有の J2C 接続ファクトリー・プロパティです。接続プールおよび拡張接続ファクトリー・プロパティは、ユーザーが独自にアダプターを作成する場合に構成するプロパティです。

10. 変更するプロパティごとに、以下の手順を実行します。

注: これらのプロパティの詳細については、136 ページの『Managed Connection Factory プロパティ』を参照してください。

- a. プロパティの名前をクリックします。
 - b. 「値」フィールドの値の内容を変更するか、フィールドが空の場合は値を入力します。
 - c. 「OK」をクリックします。
11. ウィンドウの上部にある「メッセージ」ボックス内の「保管」リンクをクリックします。

結果

アダプター・モジュールに関連付けられた Managed Connection Factory プロパティが変更されます。

組み込みアダプターの活動化仕様プロパティの設定

アダプターをモジュールの一部としてデプロイした後に、そのアダプターの活動化仕様プロパティを設定するには、管理コンソールを使用します。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てから、その値を変更または設定します。

始める前に

アダプター・モジュールを WebSphere Process Server または WebSphere Enterprise Service Bus 上にデプロイする必要があります。

このタスクを実行する理由および時期

活動化仕様プロパティは、エンドポイントを Inbound 処理用に構成する場合に使用します。

管理コンソールを使用してプロパティを構成するには、以下の手順を実行します。

このタスクの手順

1. 管理コンソールを開始します。
2. 「アプリケーション」の下で、「エンタープライズ・アプリケーション」を選択します。

3. 「エンタープライズ・アプリケーション」リストから、プロパティーを変更するアダプター・モジュールの名前をクリックします。
4. 「モジュール」の下で、「モジュールの管理」をクリックします。

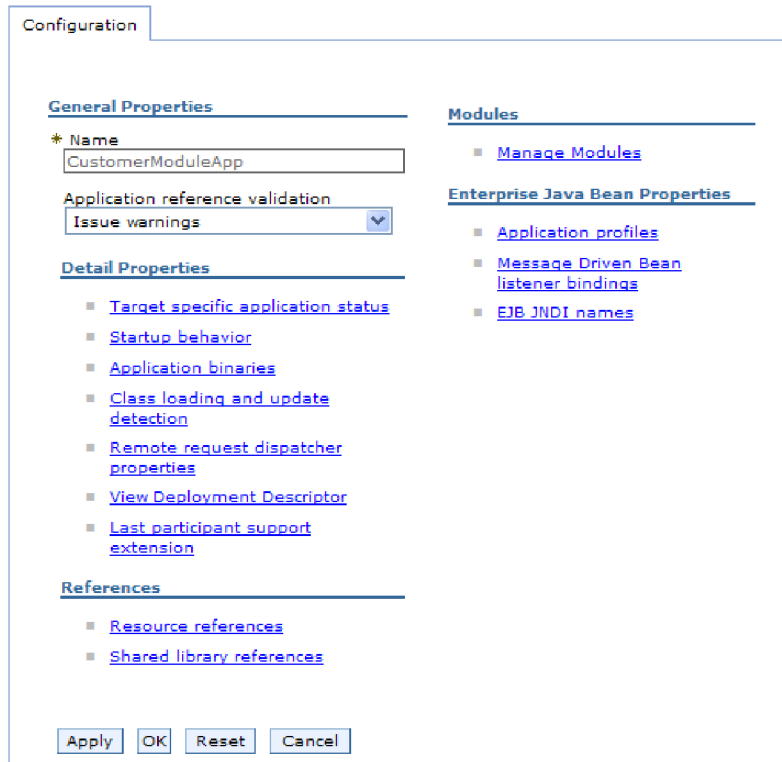


図 42. 「構成」タブでの「モジュールの管理」の選択

5. 「**IBM WebSphere Adapter for JD Edwards EnterpriseOne**」をクリックします。
6. 「追加プロパティー」リストから、「リソース・アダプター」をクリックします。
7. 次のページで、「追加プロパティー」リストから、「**J2C 活動化仕様**」をクリックします。
8. アダプター・モジュールに関連付けられている活動化仕様の名前をクリックします。
9. 「追加プロパティー」リストから、「**J2C 活動化仕様のカスタム・プロパティー**」をクリックします。
10. 変更するプロパティーごとに、以下の手順を実行します。

注: これらのプロパティーの詳細については、151 ページの『活動化仕様プロパティー』を参照してください。

- a. プロパティーの名前をクリックします。
- b. 「値」フィールドの値の内容を変更するか、フィールドが空の場合は値を入力します。
- c. 「**OK**」をクリックします。

11. ウィンドウの上部にある「メッセージ」ボックス内の「保管」リンクをクリックします。

結果

アダプター・モジュールに関連付けられている活動化仕様プロパティーが変更されました。

スタンドアロン・アダプターの構成プロパティーの変更

スタンドアロン・アダプターのインストール後に構成プロパティーを設定するには、実行時環境の管理コンソールを使用します。アダプターに関する一般的な情報を入力して、(汎用のアダプター操作に使用される) リソース・アダプター・プロパティーを設定します。アダプターを **Outbound** 操作に使用する場合は、接続ファクトリーを作成して、それに対してプロパティーを設定します。

スタンドアロン・アダプターのリソース・アダプター・プロパティーの設定

スタンドアロン・アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus にインストールした後に、そのアダプターのリソース・アダプター・プロパティーを設定するには、管理コンソールを使用します。構成するプロパティーの名前を選択してから、その値を変更または設定します。

始める前に

アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus にインストールしておく必要があります。

このタスクを実行する理由および時期

カスタム・プロパティーとは、すべての WebSphere アダプターが共用するデフォルト構成プロパティーです。

管理コンソールを使用してプロパティーを構成するには、以下の手順を実行します。

このタスクの手順

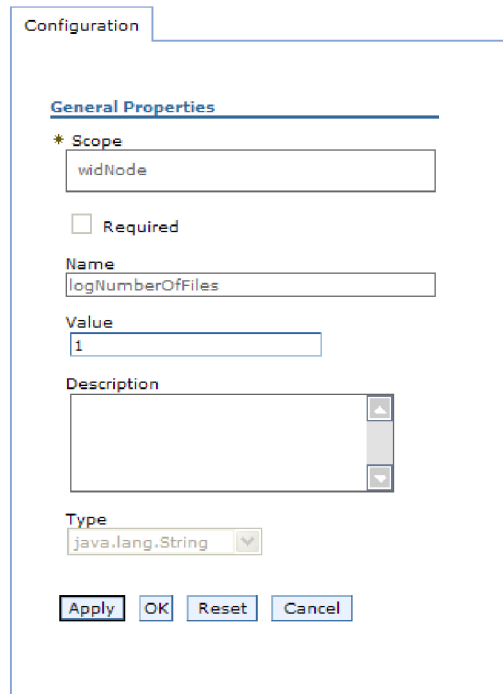
1. 管理コンソールを開始します。
2. 「リソース」 → 「リソース・アダプター」 → 「リソース・アダプター」をクリックします。
3. 「リソース・アダプター」ページで、「**IBM WebSphere Adapter for JD Edwards EnterpriseOne**」をクリックします。
4. 「追加プロパティー」リストで、「**カスタム・プロパティー**」をクリックします。
5. 変更するプロパティーごとに、以下の手順を実行します。

注: これらのプロパティーの詳細については、132 ページの『リソース・アダプター・プロパティー』を参照してください。

- a. プロパティーの名前をクリックします。

- b. 「値」フィールドの値の内容を変更するか、フィールドが空の場合は値を入力します。

例えば、「logNumberOfFiles」をクリックすると、次のページが表示されます。



The screenshot shows a configuration window titled "Configuration" with a "General Properties" section. The "Scope" field is set to "widNode" and has a "Required" checkbox that is unchecked. The "Name" field contains "logNumberOfFiles". The "Value" field contains "1". The "Description" field is empty. The "Type" dropdown menu is set to "java.lang.String". At the bottom, there are four buttons: "Apply", "OK", "Reset", and "Cancel".

図 43. logNumberOfFiles プロパティの「構成」タブ

「値」フィールドの数値を変更して、プロパティの説明を追加できます。

- c. 「OK」をクリックします。
6. ページの上部にある「メッセージ」ボックスで「保管」をクリックします。

結果

アダプターに関連付けられているリソース・アダプター・プロパティが変更されました。

スタンドアロン・アダプターの Managed (J2C) Connection Factory プロパティの設定

スタンドアロン・アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus にインストールした後に、そのアダプターの管理接続ファクトリー・プロパティを設定するには、管理コンソールを使用します。構成するプロパティの名前を選択してから、その値を変更または設定します。

始める前に

アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus にインストールしておく必要があります。

このタスクを実行する理由および時期

Managed Connection Factory プロパティは、ターゲット・JD Edwards EnterpriseOne サーバーのインスタンスを構成する場合に使用します。

注: 管理コンソール内では、このプロパティを「J2C 接続ファクトリー・プロパティ」と呼びます。

管理コンソールを使用してプロパティを構成するには、以下の手順を実行します。

このタスクの手順

1. 管理コンソールを開始します。
2. 「リソース」 → 「リソース・アダプター」 → 「リソース・アダプター」をクリックします。
3. 「リソース・アダプター」ページで、「**IBM WebSphere Adapter for JD Edwards EnterpriseOne**」をクリックします。
4. 「追加プロパティ」リストで、「**J2C 接続ファクトリー**」をクリックします。
5. 既存の接続ファクトリーを使用する場合は、ここをスキップして、既存の接続ファクトリー・リストから選択します。

注: 外部サービス・ウィザードを使用してアダプター・モジュールを構成したときに「事前定義された接続プロパティを使用する」を選択していた場合は、接続ファクトリーを作成する必要はありません。

接続ファクトリーを作成する場合は、以下の手順を実行します。

- a. 「新規作成」をクリックします。
- b. 「構成」タブの「一般プロパティ」セクションで、接続ファクトリーの名前を入力します。例えば、AdapterCF と入力できます。
- c. 「JNDI 名」に値を入力します。例えば、com/eis/AdapterCF と入力できます。
- d. 「コンポーネント管理認証別名」リストから認証別名を選択します。
- e. 「OK」をクリックします。
- f. ページの上部にある「メッセージ」ボックスで「保管」をクリックします。

新規に作成された接続ファクトリーが表示されます。

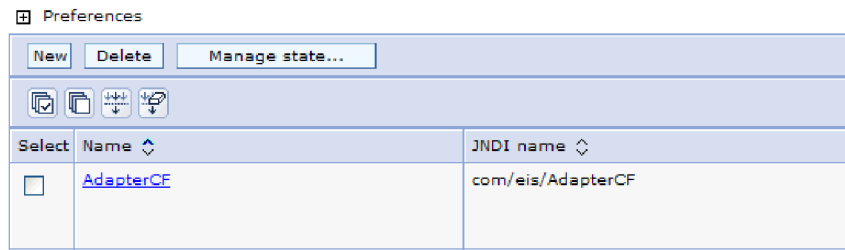


図 44. リソース・アダプターと併用するためのユーザー定義接続ファクトリー

6. 接続ファクトリーのリストで、使用するものをクリックします。
7. 「追加プロパティ」リストで、「カスタム・プロパティ」をクリックします。

カスタム・プロパティは、Adapter for JD Edwards EnterpriseOne に特有の J2C 接続ファクトリー・プロパティです。接続プールおよび拡張接続ファクトリー・プロパティは、ユーザーが独自にアダプターを作成する場合に構成するプロパティです。

8. 変更するプロパティごとに、以下の手順を実行します。

注: これらのプロパティの詳細については、136 ページの『Managed Connection Factory プロパティ』を参照してください。

- a. プロパティの名前をクリックします。
 - b. 「値」フィールドの値の内容を変更するか、フィールドが空の場合は値を入力します。
 - c. 「OK」をクリックします。
9. プロパティの設定が終了したら、「適用」をクリックします。
 10. ウィンドウの上部にある「メッセージ」ボックスで「保管」をクリックします。

結果

アダプターに関連付けられている Managed Connection Factory プロパティが設定されます。

スタンドアロン・アダプターの活動化仕様プロパティの設定

スタンドアロン・アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus にインストールした後に、そのアダプターのアクティベーション・スペック・プロパティを設定するには、管理コンソールを使用します。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てから、その値を変更または設定します。

始める前に

アダプターを WebSphere Process Server または WebSphere Enterprise Service Bus にインストールしておく必要があります。

このタスクを実行する理由および時期

活動化仕様プロパティは、エンドポイントを Inbound 処理用に構成する場合に使用します。

管理コンソールを使用してプロパティを構成するには、以下の手順を実行します。

このタスクの手順

1. 管理コンソールを開始します。
2. 「リソース」 → 「リソース・アダプター」 → 「リソース・アダプター」をクリックします。

3. 「リソース・アダプター」ページで、「**IBM WebSphere Adapter for JD Edwards EnterpriseOne**」をクリックします。
4. 「追加プロパティ」リストで、「**J2C 活動化仕様**」をクリックします。
5. 既存の活動化仕様を使用する場合は、ここをスキップして、既存の活動化仕様リストから選択します。

注: 外部サービス・ウィザードを使用してアダプター・モジュールを構成したときに「事前定義された接続プロパティを使用する」を選択していた場合は、活動化仕様を作成する必要はありません。

活動化仕様を作成する場合は、以下の手順を実行します。

- a. 「**新規作成**」をクリックします。
- b. 「**構成**」タブの「**一般プロパティ**」セクションで、活動化仕様の名前を入力します。例えば、AdapterAS と入力できます。
- c. 「**JNDI 名**」に値を入力します。例えば、com/eis/AdapterAS と入力できません。
- d. 「**認証別名**」リストから認証別名を選択します。
- e. メッセージ・リスナー・タイプを選択します。
- f. 「**OK**」をクリックします。
- g. ページの上部にある「**メッセージ**」ボックスで「**保管**」をクリックします。

新規に作成された活動化仕様が表示されます。

6. 活動化仕様のリストで、使用するものをクリックします。
7. 「追加プロパティ」リストで、「**J2C 活動化仕様のカスタム・プロパティ**」をクリックします。
8. 設定するプロパティごとに、次の手順を実行します。

注: これらのプロパティの詳細については、151 ページの『活動化仕様プロパティ』を参照してください。

- a. プロパティの名前をクリックします。
 - b. 「**値**」フィールドの値の内容を変更するか、フィールドが空の場合は値を入力します。
 - c. 「**OK**」をクリックします。
9. プロパティの設定が終了したら、「**適用**」をクリックします。
 10. ページの上部にある「**メッセージ**」ボックスで「**保管**」をクリックします。

結果

アダプターに関連付けられた活動化仕様プロパティが設定されます。

アダプターを使用するアプリケーションの開始

アダプターを使用するアプリケーションを開始するには、サーバーの管理コンソールを使用します。デフォルトでは、サーバーが始動すると、アプリケーションは自動的に開始します。

このタスクを実行する理由および時期

アプリケーションを開始するには、アプリケーションが組み込みアダプター、スタンドアロン・アダプターのいずれを使用しているかにかかわらず、以下の手順に従います。組み込みアダプターを使用するアプリケーションの場合、アダプターはアプリケーションの開始時に開始されます。スタンドアロン・アダプターを使用するアプリケーションの場合、アダプターはアプリケーション・サーバーの始動時に開始されます。

このタスクの手順

1. 管理コンソールで、「アプリケーション」 → 「エンタープライズ・アプリケーション」をクリックします。

注: 管理コンソールには、「Integrated Solutions Console」というラベルが付いています。

2. 開始するアプリケーションのチェック・ボックスを選択します。アプリケーション名は、インストールした EAR ファイルの名前からファイル拡張子 .EAR を除いたものです。
3. 「開始 (Start)」をクリックします。

結果

アプリケーションの状況が「開始済み」に変化し、アプリケーションが開始されたことを示すメッセージが管理コンソールの上部に表示されます。

アダプターを使用するアプリケーションの停止

アダプターを使用するアプリケーションを停止するには、サーバーの管理コンソールを使用します。デフォルトでは、サーバーが停止すると、アプリケーションは自動的に停止します。

このタスクを実行する理由および時期

アプリケーションを停止するには、アプリケーションが組み込みアダプター、スタンドアロン・アダプターのいずれを使用しているかにかかわらず、以下の手順に従います。アプリケーションと組み込みアダプターの組み合わせの場合、アダプターはアプリケーションの停止時に停止します。スタンドアロン・アダプターを使用するアプリケーションの場合、アダプターはアプリケーション・サーバーの停止時に停止します。

このタスクの手順

1. 管理コンソールで、「アプリケーション」 → 「エンタープライズ・アプリケーション」をクリックします。

注: 管理コンソールには、「Integrated Solutions Console」というラベルが付いています。

2. 停止するアプリケーションのチェック・ボックスを選択します。アプリケーション名は、インストールした EAR ファイルの名前からファイル拡張子 .EAR を除いたものです。
3. 「停止 (Stop)」をクリックします。

結果

アプリケーションの状況が「停止」に変化し、アプリケーションが停止したことを示すメッセージが管理コンソールの上部に表示されます。

Performance Monitoring Infrastructure を使用したパフォーマンスのモニター

Performance Monitoring Infrastructure (PMI) は、管理コンソールの機能の 1 つで、これを使用すると、実稼働環境内で Adapter for JD Edwards EnterpriseOne を含む、コンポーネントのパフォーマンスを動的にモニターすることができます。PMI は、サーバー内のさまざまなコンポーネントから、平均応答時間や要求の総数などのアダプターのパフォーマンス・データを収集して、そのデータをツリー構造に編成します。このデータは、Tivoli® Performance Viewer (WebSphere Process Server の管理コンソールに統合されているグラフィカル・モニター・ツール) を通して表示することができます。

このタスクを実行する理由および時期

使用するアダプター用に PMI を使用可能に設定し、構成するためには、まず、トレースの詳細レベルを設定し、パフォーマンス・データの収集元となるいくつかのイベントを実行する必要があります。

ご使用のアダプター環境の全体的なパフォーマンスをモニターし、それを向上させるために PMI を役立てる方法について詳しくは、WebSphere Application Server の Web サイト (<http://www.ibm.com/software/webservers/appserv/was/library/>) で PMI を検索してください。

Performance Monitoring Infrastructure の構成

Performance Monitoring Infrastructure (PMI) を、アダプターのパフォーマンス・データ (平均応答時間や要求の総数など) を収集するように構成することができます。使用するアダプター用に PMI を構成した後、Tivoli Performance Viewer を使用してアダプターのパフォーマンスをモニターすることができます。

始める前に

使用するアダプター用に PMI を構成するためには、まず、トレースの詳細レベルを設定し、パフォーマンス・データの収集元となるいくつかのイベントを実行する必要があります。

1. トレース機能を使用可能にしてイベント・データを受け取るためには、トレース・レベルを `fine`、`finer`、`finest`、または `all` のいずれかに設定する必要があります。`*=info` の後に、コロンとストリングを追加します。例えば、次のように入力します。

```
*=info: WBILocationMonitor.CEI.ResourceAdapter.  
*=finest: WBILocationMonitor.LOG.ResourceAdapter.*=finest:
```

トレース・レベルの設定方法については、111 ページの『Common Event Infrastructure (CEI) を使用したトレースの使用可能化』を参照してください。

2. 1 つ以上の Outbound 要求を生成して、構成可能なパフォーマンス・データを生成します。

このタスクの手順

1. アダプターに対して PMI を使用可能にします。
 - a. 管理コンソールで、「モニターおよびチューニング」を展開してから、「Performance Monitoring Infrastructure (PMI)」を選択します。
 - b. サーバーのリストから、ご使用のサーバーの名前をクリックします。
 - c. 「構成」タブを選択してから、「Performance Monitoring (PMI) を使用可能にする (Enable Performance Monitoring (PMI))」チェック・ボックスを選択します。
 - d. 「カスタム」を選択して、選択的に統計を使用可能または使用不可に設定します。

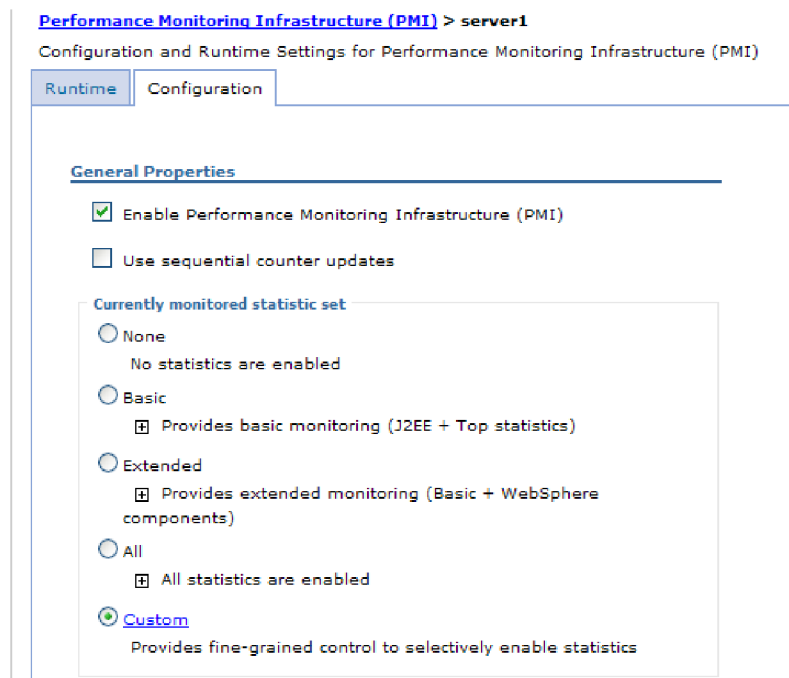


図 45. Performance Monitoring Infrastructure の使用可能化

- e. 「適用」または「OK」をクリックします。
 - f. 「保管」をクリックします。これで、PMI が使用可能になりました。
2. アダプター用に PMI を構成します。

- a. 管理コンソールで、「モニターおよびチューニング」を展開してから、「Performance Monitoring Infrastructure (PMI)」を選択します。
- b. サーバーのリストから、ご使用のサーバーの名前をクリックします。
- c. 「カスタム」を選択します。
- d. 「ランタイム」タブを選択します。以下の図は、「ランタイム」タブを示しています。

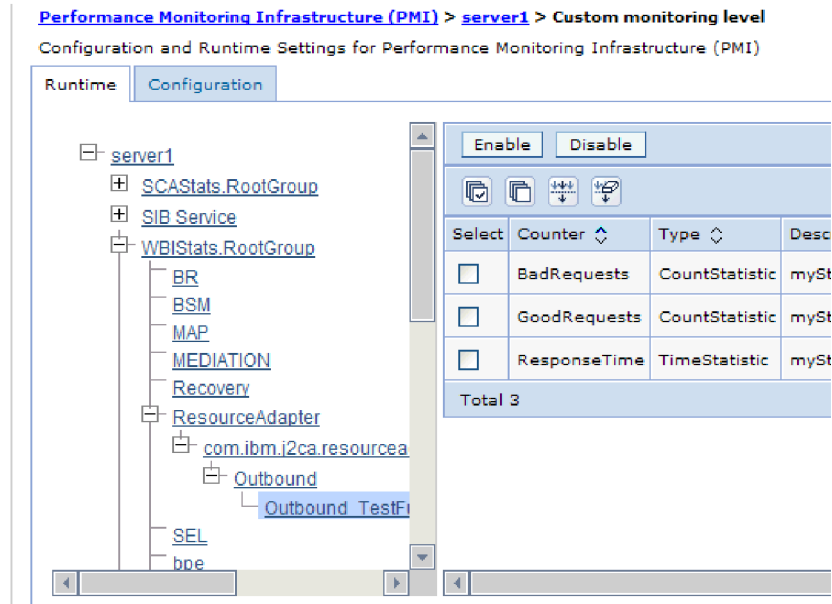


図 46. PMI の構成に使用される「ランタイム」タブ

- e. 「WBISStats.RootGroup」をクリックします。これは、ルート・グループで収集されるデータ用の PMI サブモジュールです。この例では、ルート・グループに WBISStats という名前を使用しています。
- f. 「ResourceAdapter」をクリックします。これは、JCA アダプターについて収集されるデータ用のサブモジュールです。
- g. アダプターの名前をクリックして、モニターするプロセスを選択します。
- h. 右側のペインで、収集する統計のチェック・ボックスを選択してから、「使用可能」をクリックします。

結果

PMI がアダプター用に構成されます。

次のタスク

これで、アダプターのパフォーマンス統計を表示できるようになりました。

パフォーマンスに関する統計の表示

アダプターのパフォーマンス・データは、グラフィカル・モニター・ツール Tivoli Performance Viewer を使用して表示することができます。Tivoli Performance Viewer は、WebSphere Process Server の管理コンソールに組み込まれています。

始める前に

アダプター用の Performance Monitoring Infrastructure の構成。

このタスクの手順

1. 管理コンソールで、「**モニターおよびチューニング**」を展開し、「**Performance Viewer**」を展開した後、「**現行アクティビティ**」を選択します。
2. サーバーのリストにて、ご使用のサーバーの名前をクリックします。
3. サーバー名の下で、「**パフォーマンス・モジュール**」を展開します。
4. 「**WBISStatsRootGroup**」をクリックします。
5. 「**ResourceAdapter**」およびアダプター・モジュールの名前をクリックします。
6. 複数のプロセスがある場合は、統計を表示させるプロセスのチェック・ボックスを選択します。

結果

右側のパネルに統計が表示されます。「**グラフの表示**」をクリックして、データのグラフを表示するか、または「**表の表示**」をクリックして、統計を表形式で表示することができます。以下の図では、アダプターのパフォーマンス統計をグラフの形で表示しています。

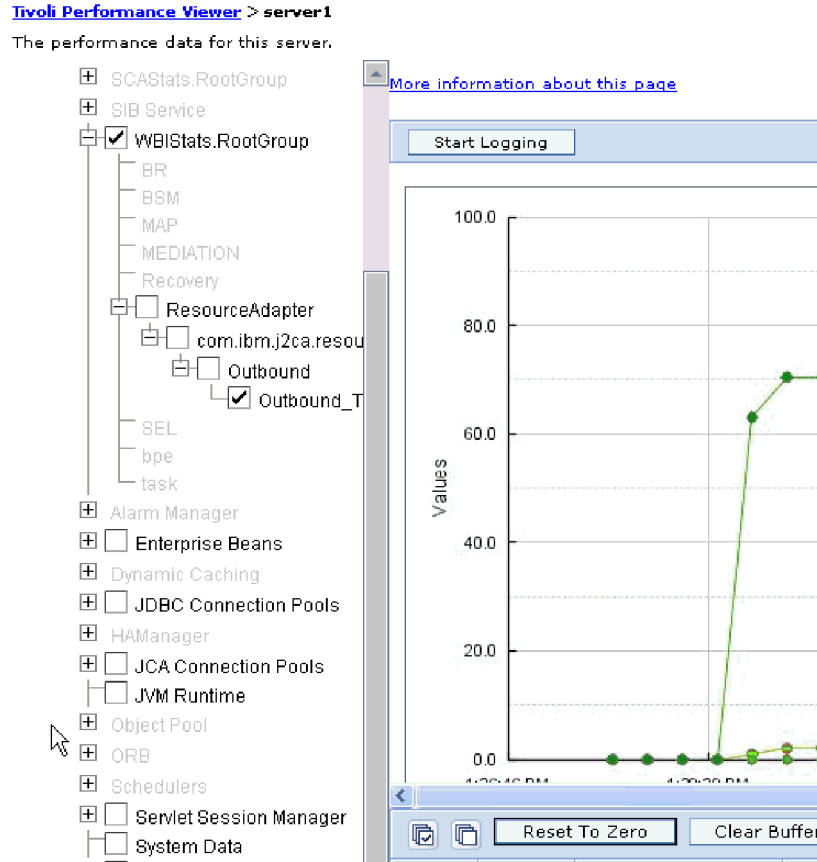


図 47. グラフ表示によるアダプターのパフォーマンス統計

Common Event Infrastructure (CEI) を使用したトレースの使用可能化

アダプターは、サーバー内に組み込まれたコンポーネントである Common Event Infrastructure を使用して、ポーリング周期の開始または停止などの重要なビジネス・イベントに関するデータを通知できます。 イベント・データの書き込み先は、構成設定に応じてデータベースまたはトレース・ログ・ファイルになります。

このタスクを実行する理由および時期

このタスクの手順

1. 管理コンソールにて、「トラブルシューティング」をクリックします。
2. 「ログおよびトレース」を選択します。
3. サーバーのリストにて、ご使用のサーバーの名前をクリックします。
4. 「ログ詳細レベルの変更」ボックスで、アダプターによるイベント・データの書き込み先にする CEI データベースの名前 (例えば、WBIEventMonitor.CEI.ResourceAdapter.*) またはトレース・ログ・ファイルの名前 (例えば、WBIEventMonitor.LOG.ResourceAdapter.*) をクリックします。
5. アダプターを使用してデータベースまたはトレース・ログ・ファイルに書き込むビジネス・イベントの詳細レベルを選択し、(必要に応じて) メッセージおよびトレースに関連付けられている詳細レベルの細分度を調整します。
 - **ロギングなし。** イベント・ロギングをオフにします。

- **メッセージのみ。** アダプターはイベントを通知します。
- **すべてのメッセージおよびトレース。** アダプターは、イベントの詳細を通知します。
- **メッセージとトレースのレベル。** イベントに関連付けられているビジネス・オブジェクト・ペイロードについてアダプターが通知する詳細度を制御するための設定です。詳細度を調整する場合は、以下のいずれかを選択してください。

詳細 - 中。アダプターはイベントを通知しますが、ビジネス・オブジェクト・ペイロードについては通知しません。

詳細 - 高。アダプターは、イベントおよびビジネス・オブジェクト・ペイロードの説明を通知します。

詳細 - 最高。アダプターは、イベントおよび全ビジネス・オブジェクト・ペイロードを通知します。

6. 「**OK**」をクリックします。

結果

イベント・ロギングが使用可能になります。CEI 項目は、トレース・ログ・ファイル内で参照できます。または、管理コンソール内で **Common Base Event Browser** を使用して表示することもできます。

第 8 章 トラブルシューティングおよびサポート

一般的なトラブルシューティング技術やセルフ・ヘルプ情報は、問題の確認および早期解決に役立ちます。

Log and Trace Analyzer のサポート

アダプターは、Log and Trace Analyzer で表示できるログ・ファイルとトレース・ファイルを作成します。

Log and Trace Analyzer は、ログ・ファイルとトレース・ファイルをフィルタリングして、アダプターのメッセージとトレース情報を分離することができます。また、ログ・ビューアーの中で、アダプターのメッセージとトレース情報を強調表示することもできます。

フィルタリングと強調表示の対象とするアダプターのコンポーネント ID は、文字 JDERA とアダプター ID プロパティの値から構成されるストリングです。例えば、アダプター ID プロパティが、001 に設定されている場合、コンポーネント ID は、JDERA001 となります。

同じアダプターを複数インスタンス実行する場合、アダプター ID プロパティの最初の 8 文字は、必ずインスタンスごとに固有にし、ログおよびトレース情報を特定のアダプター・インスタンスに相互に関連付けられるようにしてください。アダプター ID プロパティの最初の 7 文字を固有にすることにより、そのアダプターの複数インスタンスのコンポーネント ID も固有になり、アダプターの特定インスタンスにログおよびトレース情報を相互に関連付けることができます。アダプター ID プロパティの長さが、ログおよびトレース・ファイルのフィルター処理にどのように関係するかを示すため、2 つのインスタンスの WebSphere Adapter for JD Edwards EnterpriseOne のアダプター ID プロパティを 001 と 002 に設定するとします。これらのインスタンスのコンポーネント ID、JDERA001 および JDERA002 は、短いので固有性を保つことができ、別のアダプター・インスタンスとして区別することができます。しかし、もっと長いアダプター ID プロパティのインスタンスの場合、互いを区別できなくなります。例えば、2 つのインスタンスのアダプター ID プロパティを、Instance01 および Instance02 に設定するとします。この場合、各アダプター・インスタンスのログおよびトレース情報を個別に調べることはできなくなります。これは、両方のインスタンスのコンポーネント ID が JDERAInstance に切り詰められるためです。

Outbound 処理については、アダプター ID プロパティは、リソース・アダプターおよび Managed Connection Factory プロパティ・グループの両方にあります。外部サービス・ウィザードを使用して、Outbound 処理用にアダプターを構成後、アダプター ID プロパティを更新する場合、リソース・アダプター・プロパティと Managed Connection Factory プロパティを矛盾がないように必ず設定し、ログおよびトレース・エントリーのマーキングが不整合にならないようにしてください。Inbound 処理については、アダプター ID プロパティは、リソース・アダプター・プロパティのみに設定されますので、このような配慮は不要です。

アダプター ID プロパティの詳細については、133 ページの『アダプター ID (AdapterID)』 . Log and Trace Analyzer について詳しくは、<http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/topic/org.eclipse.hyades.log.ui.doc.user/concepts/cltviews.htm>を参照してください。

ロギングおよびトレースの構成

要件に合うようロギングおよびトレースを構成します。アダプターのロギングを使用可能にし、イベント処理の状況を制御します。アダプターのログ・ファイル名およびトレース・ファイル名を変更して、ほかのログ・ファイルおよびトレース・ファイルと区別します。

このタスクを実行する理由および時期

ロギング・プロパティの構成

管理コンソールを使用して、ロギングを使用可能にして、ログの出力プロパティ（ログの場所、詳細レベル、および出力フォーマットなど）を設定します。

このタスクを実行する理由および時期

アダプターでモニター対象イベントをログに記録できるようにするには、モニターしようとするサービス・コンポーネントのイベント・ポイント、イベントごとに必要となる詳細レベル、およびイベントをログにパブリッシュするのに使用する出力のフォーマットを、あらかじめ指定する必要があります。管理コンソールを使用して、次のタスクを実行します。

- 特定のイベント・ログを使用可能または使用不可に設定する
- ログの詳細レベルを指定する
- ログ・ファイルの保管場所および保有数を指定する
- ログ出力のフォーマットを指定する

ログ・アナライザー出力のフォーマットを設定する場合、ログ・アナライザー・ツールを使用してトレース出力を開くことができます。ログ・アナライザー・ツールは、プロセス・サーバーに組み込まれたアプリケーションです。このツールは、ログ・アナライザーのマージ機能を使用できるため、異なる 2 つのサーバー・プロセスのトレースを相関させる場合に便利です。

サービス・コンポーネントやイベント・ポイントなどのプロセス・サーバー 上でのモニターについて詳しくは、プロセス・サーバー の資料を参照してください。

ログ構成は、静的または動的に変更できます。アプリケーション・サーバーを開始または再始動すると、静的構成が有効になります。動的構成変更（またはランタイム構成変更）は、即時適用されます。

ログが作成されると、そのログの詳細レベルが構成データから設定されます。特定のログ名に対して、構成データが使用可能でない場合、そのログのレベルは、ログの親から取得されます。親ログに構成データが存在しない場合、そのログの親が確認される、という具合に、ヌル以外のレベル値があるログが見つかるまでツリーを上昇します。ログのレベルを変更すると、その変更はログの子に伝搬されます。また、必要に応じて、ログの子からその子へと変更が再帰的に伝搬されます。

ロギングを使用可能にし、ログの出力プロパティを設定するには、次の手順を実行します。

このタスクの手順

1. 管理コンソールのナビゲーション・ペインで、「サーバー」 → 「アプリケーション・サーバー」をクリックします。
2. 作業したいサーバーの名前をクリックします。
3. 「トラブルシューティング」で「ログおよびトレース」をクリックします。
4. 「ログ詳細レベルの変更」をクリックします。
5. 変更を有効にするには、以下を行います。
 - 構成を静的に変更する場合は、「構成」タブをクリックします。
 - 構成を動的に変更する場合は、「ランタイム」タブをクリックします。
6. 変更したいロギング・レベルのパッケージの名前をクリックします。
WebSphere Adapters 用のパッケージ名は、**com.ibm.j2ca.*** で始まります。
 - アダプターの基本コンポーネントの場合は、**com.ibm.j2ca.base.*** を選択します。
 - アダプターの基本コンポーネントとすべてのデプロイ済みアダプターの場合は、**com.ibm.j2ca.*** を選択します。
 - Adapter for JD Edwards EnterpriseOne の場合のみ、**com.ibm.j2ca.jde.*** パッケージを選択します。
7. ロギング・レベルを選択します。

ロギング・レベル	説明
致命的	タスクを続行できない。または、コンポーネントが機能しない。
重大	タスクを続行できないが、コンポーネントは機能する。このロギング・レベルには、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唆する) 状況も含まれる。
警告	潜在的なエラーが発生したか、重大エラーが差し迫っている。このロギング・レベルには、例えばリソース・リークの可能性など、進行性のフォールトを示す状況も含まれる。
監査	サーバーの状態やリソースに影響を与える重大なイベントが発生した。
情報	タスクが稼働中である。このロギング・レベルには、タスクの全体的な進行を概説する一般情報が含まれる。
構成	構成の状況が報告されるか、構成変更が発生した。
詳細	サブタスクが稼働中である。このロギング・レベルには、サブタスクの進行を詳細に説明した一般情報が含まれる。

8. 「適用」をクリックします。
9. 「OK」をクリックします。
10. 静的な構成変更を有効にするには、プロセス・サーバー を停止し、再始動します。

結果

これ以降、ログ項目には、選択したアダプター・コンポーネントについての指定したレベルの情報が格納されます。

ログ・ファイル名およびトレース・ファイル名の変更

アダプター・ログおよびトレース情報を他のプロセスとは分離して保持するには、管理コンソールを使用してファイル名を変更します。デフォルトでは、プロセス・サーバー 上にあるすべてのプロセスおよびアプリケーションのログ情報およびトレース情報は、それぞれ `SystemOut.log` ファイルおよび `trace.log` ファイルに書き込まれます。

始める前に

アダプター・モジュールをアプリケーション・サーバーにデプロイした後は、ログ・ファイル名およびトレース・ファイル名はいつでも変更できます。

このタスクを実行する理由および時期

ログ・ファイルおよびトレース・ファイルは、静的または動的に変更できます。アプリケーション・サーバーを開始または再始動すると、静的変更が有効になります。動的変更またはランタイム構成変更は、即座に適用されます。

ログ・ファイルおよびトレース・ファイルは、`install_root/profiles/profile_name/logs/server_name` フォルダーにあります。

ログ・ファイル名およびトレース・ファイル名を設定または変更するには、以下の手順を使用します。

このタスクの手順

1. 管理コンソールのナビゲーション・ペインで、「アプリケーション」>「エンタープライズ・アプリケーション」を選択します。
2. 「エンタープライズ・アプリケーション」リストから、アダプター・アプリケーションの名前をクリックします。これは、アダプターの EAR ファイルの名前から `.ear` ファイル拡張子を除いたものです。例えば、EAR ファイルの名前が `Accounting_OutboundApp.ear` である場合は、**Accounting_OutboundApp** をクリックします。
3. 「構成」タブの「モジュール」リストから、「**モジュールの管理**」をクリックします。
4. モジュールのリストで、**IBM WebSphere Adapter for JD Edwards EnterpriseOne** をクリックします。
5. 「構成」タブの「追加プロパティ」の下で、「**リソース・アダプター**」をクリックします。
6. 「構成」タブの「追加プロパティ」の下で、「**カスタム・プロパティ**」をクリックします。
7. 「カスタム・プロパティ」テーブル内で、ファイル名を変更します。
 - a. 「**logFilename**」をクリックして、ログ・ファイルの名前を変更します。あるいは、「**traceFilename**」をクリックして、トレース・ファイルの名前を変更します。

- b. 「構成」タブで、「値」フィールドに新しい名前を入力します。デフォルトでは、ログ・ファイルの名前は `SystemOut.log`、トレース・ファイルの名前は `trace.log` になります。
- c. 「適用」または「OK」をクリックします。変更内容がローカル・マシン上に保存されます。
- d. 変更内容をサーバー上のマスター構成に保存するには、次のいずれかの手順を実行します。
 - **静的変更:** サーバーを停止してから再始動します。この方法では、変更を行うことは可能ですが、サーバーを停止してから始動するまで、行った変更は有効になりません。
 - **動的変更:** 「カスタム・プロパティ」テーブルの上にあるメッセージ・ボックス内にある「保管」リンクをクリックします。プロンプトが出されたら、再度「保管」をクリックします。この方法では、行った変更をすぐに有効にすることができます。

First Failure Data Capture (FFDC) サポート

アダプターは、WebSphere Process Server または WebSphere Enterprise Service Bus の実行時に発生する障害や重大なソフトウェアの問題の永続的な記録を提供する First Failure Data Capture (FFDC) をサポートしています。

FFDC 機能はバックグラウンドで実行され、実行時に発生するイベントやエラーを収集します。この機能はさまざまな障害を相互に関連付ける手段を提供するため、この機能を利用すると、ソフトウェアは、ある 1 つの障害の影響をその原因に結びつけ、その結果、障害の根本原因を素早く突き止めることが容易になります。取り込まれたデータは、アダプターの実行時に発生した例外処理を識別するときに使用できます。

問題が発生すると、例外メッセージおよびコンテキスト・データがアダプターによってログ・ファイルに書き込まれます。このログ・ファイルは `install_root/profiles/profile/logs/ffdc` ディレクトリーに置かれます。

First Failure Data Capture (FFDC) について詳しくは、WebSphere Process Server または WebSphere Enterprise Service Bus の資料を参照してください。

ビジネス・フォールト

アダプターは、予想される例外で `Outbound` サービス記述で宣言されている例外であるビジネス・フォールトか、インポートをサポートします。ビジネス・フォールトは、ビジネス・ルールの違反または制約違反が原因で、ビジネス・プロセスの予測可能なポイントに発生します。

WebSphere Process Server と WebSphere Enterprise Service Bus は、他のタイプのフォールトをサポートしますが、アダプターが生成するのはビジネス・フォールトのみです。本書では、これらを単にフォールトと呼びます。すべての例外がフォールトになるわけではありません。フォールトは、アクション可能なエラー、つまり、アプリケーションの終了を必要としないリカバリー・アクションが可能なエラーに対して生成されます。例えば、アダプターで必要なデータが含まれていない

Outbound 処理のビジネス・オブジェクトを受け取るか、あるいはアダプターで Outbound 処理中にエラーが発生した場合に、アダプターによってフォールトが生成されます。

フォールト・ビジネス・オブジェクト

外部サービス・ウィザードで、アダプターが生成できるフォールトごとのビジネス・オブジェクトを作成します。さらに、このウィザードは `WBIFault` スーパーセット・ビジネス・オブジェクトを作成します。このオブジェクトには、図 48 に示されているとおり、すべてのフォールトに共通の `message`、`errorCode`、および `primarySetKey` 属性など情報が含まれています。

WBIFault	
message	string
errorCode	string
primaryKeySet	PrimaryKeyPairType []

図 48. `WBIFault` ビジネス・オブジェクトの構造

フォールトによっては、`matchCount` 属性が含まれるものもありますが、この属性は、エラーに関する追加情報を提供します。その他のフォールトの場合、`WBIFault` に、フォールトの処理に必要なすべての情報が含まれています。

ウィザードは、以下のフォールト・ビジネス・オブジェクトを作成します。

- `InvalidRequestFault`

アダプターでは、JD Edwards EnterpriseOne サーバー にビジネス・オブジェクトを送る前に、そのビジネス・オブジェクトのプロパティ値にエラーを検出した場合に、このフォールトをスローします。例えば、日付値ストリングが、サポートされる日付形式 (JDEDate プロパティ・タイプの場合 `yyyy-MM-dd`、JDEUTime プロパティ・タイプの場合 `yyyy-MM-dd'T'hh:mm`) になっていない場合、このフォールトがスローされます。

- `MatchesExceededLimitFault`

`RetrieveAll` 操作を処理する場合に、データベース照会から返されたレコード数が、対話仕様内のレコードの `MaxRecords` プロパティの値を超えると、アダプターはこのフォールトをスローします。

返されるレコード数を増やすには、`RetrieveAll` 操作の対話仕様プロパティ内の `MaxRecords` プロパティの値を増やします。

このフォールトのビジネス・オブジェクトには 1 つのプロパティ `matchCount` があり、このプロパティは一致した数が含まれるストリングです。

- `MissingDataFault`

Outbound 操作に渡されたビジネス・オブジェクトに必要なすべての属性がない場合、アダプターはこのフォールトをスローします。このフォールトは、Create、Delete、Update、Retrieve、ApplyChanges および Exists の各操作に対して発生する可能性があります。

例えば、以下の状態の場合に、アダプターによりこの例外がスローされます。

- requiredType のアプリケーション固有情報が「はい」で、ioType が「デフォルト」ではなく、またプロパティーが設定されていない場合
- reference のアプリケーション固有情報が設定され、requiredType のアプリケーション固有情報が「はい」で、また参照されるプロパティーが設定されていない場合

- RecordNotFoundFault

データ検索時に、指定されたキーのレコードがデータベース内に見つからなかった場合に、アダプターはこのフォールトをスローします。このフォールトは、Delete、Update、Retrieve および RetrieveAll の各操作に対して発生する可能性があります。

XAResourceNotAvailableException

プロセス・サーバーのログに

com.ibm.ws.Transaction.XAResourceNotAvailableException 例外のレポートが繰り返し出力されている場合、トランザクション・ログを除去して問題を訂正してください。

症状:

アダプターを始動すると、プロセス・サーバーのログ・ファイルに以下の例外が繰り返し記録されます。

```
com.ibm.ws.Transaction.XAResourceNotAvailableException
```

問題:

プロセス・サーバーが、リソースのトランザクションをコミットまたはロールバックしている間に、そのリソースが除去されました。アダプターを始動すると、アダプターはトランザクションをリカバリーしようとしませんが、リソースが除去されているためできません。

解決策:

この問題を解決するには、以下の手順を使用します。

1. プロセス・サーバー を停止します。
2. トランザクションを含むトランザクション・ログ・ファイルを除去します。例外トレース内の情報を使用して、トランザクションを識別します。これにより、サーバーはこれらのトランザクションのリカバリーを試行しなくなります。

注: 通常、テスト環境または開発環境では、すべてのトランザクション・ログを削除できます。WebSphere Integration Developer で、トランザクション・ログ・ディレクトリー `server_install_directory\profiles\profile_name\tranlog` のファイルおよびサブディレクトリーを削除します。

実稼働環境では、処理が不要なイベントを表すトランザクションのみを削除します。この処理を行う方法の 1 つとしては、アダプターを再インストールし、使用されている元のイベント・データベースにアダプターを位置指定して、不要なトランザクションのみを削除する方法があります。別の方法として、以下のディレクトリーにある `log1` または `log2` ファイルのどちらかから、トランザクションを削除します。

```
server_install_directory\profiles\profile_name\tranlog\node_name\wps\
server_name\transaction\tranlog
```

3. プロセス・サーバー を開始します。

セルフ・ヘルプ・リソース

IBM ソフトウェア・サポートのリソースは、最新のサポート情報やテクニカル文書を手入したり、サポート・ツールやフィックスをダウンロードしたり、WebSphere Adapters の問題を回避したりするために使用することができます。また、セルフ・ヘルプ・リソースは、アダプターに関連する問題を診断するのに役立ち、IBM ソフトウェア・サポートへの連絡方法についての情報を提供します。

サポート Web サイト

WebSphere Adapters ソフトウェアのサポート Web サイト (<http://www.ibm.com/software/integration/wbiadapters/support/>) では、WebSphere Adapters の学習、使用、およびトラブルシューティングに役立つ多数のリソースへのリンクを提供しています。以下の種類のリソースがあります。

- フラッシュ (製品に関する警告)
- 製品のインフォメーション・センター、マニュアル、IBM Redbooks®、およびホワイト・ペーパーなどの技術情報
- 研修関連
- テクニカル・ノート

推奨フィックス

適用することが望ましい推奨フィックスのリストは、<http://www.ibm.com/support/docview.wss?fdoc=aimadp&rs=695&uid=swg27010397> にあります。

テクニカル・ノート

テクニカル・ノートは、Adapter for JD Edwards EnterpriseOne に関する最新の資料を提供します。以下のトピックがあります。

- 問題とそれに対する現在使用可能な解決策
- よくある質問に対する答え
- アダプターのインストール、構成、使用法、トラブルシューティングに関する手引きとなる情報
- IBM ソフトウェア・サポート・ハンドブック

WebSphere Adapters のテクニカル・ノートのリストについては、以下のアドレスにアクセスしてください。

[http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8
&dc=DB520+D800+D900+DA900+DA800+DB560&dtm](http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm)

IBM Support Assistant のプラグイン

Adapter for JD Edwards EnterpriseOne では、IBM Support Assistant のプラグインを提供します。これは、無料の保守容易性ローカル・ソフトウェア・ワークベンチです。このプラグインは、動的トレース・フィーチャーをサポートしています。IBM Support Assistant のインストールおよび使用については、以下のアドレスにアクセスしてください。

<http://www.ibm.com/software/support/isa/>

第 9 章 参照情報

ユーザーの作業をサポートするための参照情報として、外部サービス・ウィザードによって生成されるビジネス・オブジェクトに関する詳細情報や、アダプター・プロパティに関する情報 (双方向変換をサポートするアダプター・プロパティなど) を提供しています。また、アダプターのメッセージや関連製品情報についても示しています。

ビジネス・オブジェクト情報

ビジネス・オブジェクト定義ファイル内のアプリケーション固有情報、およびビジネス・オブジェクト名を調べれば、ビジネス・オブジェクトの目的を判断することができます。アプリケーション固有情報は、JD Edwards EnterpriseOne サーバー で実行可能な操作を指定します。ビジネス・オブジェクト名は、一般に実行される操作、およびビジネス・オブジェクトの構造を表します。

アプリケーション固有情報

アプリケーション固有の情報 (ASI) とは、Adapter for JD Edwards EnterpriseOne 用のビジネス・オブジェクトの処理方法に関するアダプター依存情報を指定するメタデータです。

外部サービス・ウィザードがビジネス・オブジェクトを生成すると、ビジネス・オブジェクト定義が自動的に生成され、XSD (XML スキーマ定義) ファイルとして保存されます。ビジネス・オブジェクト定義には、そのビジネス・オブジェクトに関するアプリケーション固有の情報が含まれています。生成済みの ASI を変更する場合、メタデータ値を WebSphere Integration Developer の「Business Integration」パースペクティブの「プロパティ」タブから変更するか、またはビジネス・オブジェクト・エディターを使用して変更できます。

Adapter for JD Edwards EnterpriseOne は、アプリケーション固有情報 (ASI) を使用して、Create、Retrieve、Update、Delete の各操作のための照会を作成します。ASI は、ビジネス・オブジェクト・レベル、プロパティ・レベル、および操作レベルの 3 つのレベルで、外部サービス・ウィザードによって生成されます。

ビジネス・オブジェクト・レベルのアプリケーション固有情報

ビジネス・オブジェクト・レベルのアプリケーション固有情報 (ASI) は、通常、対応するデータベース表の名前を指定する場合、および物理的または論理的な削除操作の実行に必要な情報を指定する場合に使用されます。以下の表は、ビジネス・オブジェクト・レベルの ASI を示しています。

表 6. ビジネス・オブジェクト・レベルのアプリケーション固有情報

アプリケーション固有情報	説明
Name	操作の名前
BSEFN	操作に関連付けられたビジネス関数のリスト

プロパティ・レベルのアプリケーション固有情報

プロパティ・レベルのアプリケーション固有情報 (ASI) は、通常、プロパティのメタデータを指定する場合に使用されます。プロパティ・レベルの ASI は、子オブジェクトまたは子オブジェクトの配列を表します。次の表に、複合プロパティ (子) または構造/テーブル・プロパティ (子オブジェクトの配列) の ASI を示します。

表7. プロパティ・レベルのアプリケーション固有情報

アプリケーション固有情報	説明	使用可能な値
Name	JD Edwards EnterpriseOne で表記されるビジネス関数のパラメーター名	BSFNName
Type	JD Edwards EnterpriseOne に存在する場合のビジネス関数のパラメーター・タイプ	BSFN
IOType	JD Edwards EnterpriseOne に存在する場合のビジネス関数のパラメーター・タイプ	<ul style="list-style-type: none"> IN: パラメーターは、ビジネス・オブジェクトからビジネス関数にマップされます。 OUT: パラメーターは、ビジネス関数からビジネス・オブジェクトにマップされます。 INOUT: パラメーターは両方向にマップされます。 DEFAULT: パラメーターは、デフォルトの JD Edwards EnterpriseOne 値を使用してマップされます。アダプター目的の場合、INOUT として処理されます。
RequiredType	パラメーターが必須かどうかを指定します。	<ul style="list-style-type: none"> YES: このパラメーターは必須です。 NO: このパラメーターは必須ではありません。 DEFAULT: このパラメーターは、JD Edwards EnterpriseOne 値を使用します。アダプター目的の場合、NO として処理されます。
Length	パラメーター値の最大長	なし
Reference	この属性の値を取得するために使用されるビジネス・オブジェクト・プロパティの XPath。XPath 式の先頭は、ビジネス関数レベルになります。	BusinessFunctionContainer BusinessFunction1 Prop1 BusinessFunction2 Prop2 BusinessFunction2/Prop2 プロパティを BusinessFunction1/Prop1 の値で設定する必要がある場合、BusinessFunction2/Prop2 の Reference の値は、BusinessFunction1/Prop1 に設定する必要があります。

操作レベルのアプリケーション固有情報

操作レベルのアプリケーション固有情報 (ASI) は、JD Edwards EnterpriseOne サーバー内の情報を取得または更新するなどの操作を実行する場合に、アダプターによって使用されます。以下の表は、操作レベルの ASI を示しています。

表 8. 操作レベルのアプリケーション固有情報

アプリケーション固有情報	説明	値
Name	ビジネス・オブジェクト操作の名前	<ul style="list-style-type: none"> • Create • Retrieve • Update • Delete • RetrieveAll
BSFN.Name	処理を行うビジネス関数の名前	<ul style="list-style-type: none"> • Name • RollbackOnWarnings
BSFN.RollbackOnWarnings	ビジネス関数が警告を返した場合に、アダプターが現行トランザクションをロールバックする必要があるかどうかを指定します。	False (デフォルト設定)

サポートされる操作

操作とは、アダプターが、Outbound 処理中に JD Edwards EnterpriseOne サーバー上で実行できるアクションです。操作の名前は、*create*、*update* など、通常、アダプターが行うアクションのタイプを示します。

以下の表は、Adapter for JD Edwards EnterpriseOne が、Outbound 処理時にビジネス関数および XML List に対してサポートする操作を定義しています。

表 9. サポートされるビジネス関数の操作

操作	定義
Create	トップレベルのビジネス・オブジェクトおよびそれに含まれているすべての子が作成されます。
Update	トップレベルのビジネス・オブジェクトが変更されます。この操作には追加中および削除中の子オブジェクトを含めることができます。
Delete	トップレベルのビジネス・オブジェクトおよびそれに含まれているすべての子が削除されます。
Retrieve	トップレベルのビジネス・オブジェクトおよびそれに含まれているすべての子が取得されます。

表 10. サポートされる XML List の操作

操作	定義
RetrieveAll	XML List に指定された照会値に該当するすべてのレコードを JD Edwards EnterpriseOne サーバーから取り出します。JD Edwards EnterpriseOne 照会ビジネス・オブジェクト (テーブルから取り出した各行のデータを表す) のコンテナの形式で結果セットを戻します。

命名規則

外部サービス・ウィザードがビジネス・オブジェクトを生成する際、そのビジネス・オブジェクトの名前は、そのビジネス・オブジェクトの構築に使用する JD Edwards EnterpriseOne サーバー 内のオブジェクトの名前に基づいて指定します。

外部サービス・ウィザードがビジネス・オブジェクト名を生成するときは、オブジェクト名を大/小文字混合に変換します。スペースまたは下線などのあらゆる分離文字が除去され、各単語の先頭文字が大文字にされます。例えば、外部サービス・ウィザードが CUSTOMER_ADDRESS という名前の JD Edwards EnterpriseOne サーバー・オブジェクトを使用してビジネス・オブジェクトを生成する場合、CustomerAddress という名前のビジネス・オブジェクトを生成します。

生成したビジネス・オブジェクト名で、ビジネス・オブジェクトの構造を示すことができます。ただし、ビジネス・オブジェクト名には、アダプターを意味する値は含まれません。これは、ビジネス・オブジェクト名を変更しても、ビジネス・オブジェクトの動作は変わらないことを意味します。

重要: ビジネス・オブジェクトを名前変更する場合、WebSphere Integration Developer のリファクタリング機能を使用して、すべてのビジネス・オブジェクト依存関係を確実に更新してください。リファクタリングを使用したビジネス・オブジェクトの名前変更についての説明は、以下のリンクを参照してください：
<http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/topic/com.ibm.wbit.help.refactor.doc/topics/trenameboatt.html>。

表 11. 命名規則

エレメント	命名規則	例
ビジネス・グラフの名前	親ビジネス・オブジェクトを含むビジネス・グラフは、含まれるビジネス・オブジェクトに基づいて命名され、後ろに「BG」というストリングが付されます。	GetEffectiveAddressContainerBG
ビジネス・オブジェクト・コンテナの名前	<name_of_business_object>Container	GetEffectiveAddressContainer
ビジネス関数の名前	外部サービス・ウィザードによってディスカバーされたビジネス関数の名前	GetEffectiveAddress
XML List の名前	外部サービス・ウィザードによってディスカバーされた XML List テーブルの名前	F0116

注: ビジネス・グラフの生成はオプションで、WebSphere Process Server または WebSphere Enterprise Service Bus でのみサポートされています。

Outbound 構成プロパティ

WebSphere Adapter for JD Edwards EnterpriseOne には、オブジェクトやサービスを生成したり作成したりするときに、外部サービス・ウィザードを使用して設定する、いくつかの種類 Outbound 接続構成プロパティがあります。リソース・アダプターおよび管理接続ファクトリーのプロパティは、WebSphere Process Server

または WebSphere Enterprise Service Bus にモジュールをデプロイした後に、WebSphere Integration Developer または 管理コンソール を使用して変更できますが、外部サービス・ウィザードの接続プロパティは、デプロイメント後に変更することはできません。

プロパティの詳細についてのガイド

WebSphere Adapter for JD Edwards EnterpriseOne を構成するときに使用されるプロパティは、リソース・アダプター・プロパティや Managed Connection Factory プロパティなど、それぞれの構成プロパティのトピックに記載されている表で詳細に説明されています。これらの表を使用しやすくするため、参照する各行の情報を以下に説明します。

次の表では、構成プロパティの表に表示される場合がある各行の意味を説明します。

行	説明
必須	<p>アダプターが動作するためには、必須フィールド (プロパティ) に値が必要です。必須プロパティに対しては、外部サービス・ウィザードがデフォルト値を提供する場合があります。</p> <p>外部サービス・ウィザードの必須フィールドからデフォルト値を除去しても、デフォルト値は変更されません。必須フィールドに値がまったく入っていないと、外部サービス・ウィザードは、その割り当て済みのデフォルト値を使用してフィールドを処理し、そのデフォルト値は管理コンソールにも表示されます。</p> <p>可能な値は「はい」および「いいえ」です。</p> <p>プロパティは、他のプロパティが特定の値の場合のみ必須となることがあります。その場合は、表にこの依存関係が記載されます。以下に例を示します。</p> <ul style="list-style-type: none"> • EventQueryType プロパティが Dynamic に設定された場合は「はい」 • Oracle データベースの場合は「はい」
使用可能な値	プロパティで選択可能な値をリストして説明します。
デフォルト	<p>外部サービス・ウィザードによって設定される事前定義値。プロパティが必須の場合は、デフォルト値を受け入れるか、ユーザーが値を指定する必要があります。プロパティにデフォルト値がない場合、表には「デフォルト値なし」と記載されます。</p> <p>None という語は、受け入れ可能なデフォルト値です。デフォルト値がないという意味ではありません。</p>
計測単位	プロパティの計測単位を指定します (例: キロバイト、秒)。
プロパティ・タイプ	<p>プロパティ・タイプを示します。有効なプロパティ・タイプとしては、以下のものがあります。</p> <ul style="list-style-type: none"> • Boolean • String • Integer

行	説明
使用法	<p>プロパティに適用される場合がある使用の条件または制限について記述します。制限の記載例を以下に示します。</p> <p>Rational® Application Developer for WebSphere Software バージョン 6.40 またはそれ以前では、パスワードに以下の制限があります。</p> <ul style="list-style-type: none"> • 大文字である必要があります • 長さが 8 文字である必要があります <p>Rational Application Developer for WebSphere Software バージョン 6.40 よりも後のバージョンでは、パスワードの制限が以下のように変更されました。</p> <ul style="list-style-type: none"> • 大文字小文字を区別しません • 長さが 40 文字まで可能です <p>このセクションでは、このプロパティに影響を及ぼす他のプロパティ、またはこのプロパティによって影響を受ける他のプロパティをリストし、その条件付き関係の内容を説明します。</p>
例	<p>プロパティ値のサンプルを示します。例:</p> <p>「言語が JA (日本語) に設定された場合、コード・ページ番号は 8000 に設定されます。」</p>
グローバル化	<p>グローバル化されたプロパティには各国語サポートが備わっているので、値を各国語で設定できます。</p> <p>有効な値は「はい」および「いいえ」です。</p>
BIDI 対応	<p>プロパティが双方向 (bidi) 処理でサポートされるかどうかを示します。双方向処理とは、1 つのファイルに左から右 (ヘブライ語やアラビア語など) と右から左 (URL やファイル・パスなど) の両方の意味内容を含むデータを処理するタスクを指します。</p> <p>有効な値は「はい」および「いいえ」です。</p>

外部サービス・ウィザードの接続プロパティ

外部サービス・ディスカバリー接続プロパティでは、ビジネス・オブジェクトを作成するためのツールである WebSphere Integration Developer の外部サービス・ウィザードと JD Edwards EnterpriseOne サーバー の間の接続を確立します。外部サービス・ウィザードで構成するプロパティでは、接続構成、BiDi プロパティ、トレース/ロギング・オプションなどを指定します。

外部サービス・ウィザードと JD Edwards EnterpriseOne サーバーの間の接続を確立すると、外部サービス・ウィザードは、ビジネス・オブジェクトを作成するために JD Edwards EnterpriseOne サーバーから取得しなければならないメタデータにアクセスできるようになります。

注: 外部サービス・ウィザードで設定するプロパティのいくつかは、後からそのウィザードで指定できるリソース・アダプター・プロパティおよび Managed Connection Factory プロパティの初期値として使用されます。

外部サービス接続プロパティと各プロパティの目的を以下の表にまとめます。各プロパティの完全な説明は、表に続くセクションに記載しています。表の後の

セクションにあるプロパティ詳細表の読み方については、127 ページの『プロパティの詳細についてのガイド』を参照してください。

重要: 双方向スクリプトを使用する接続プロパティを設定する場合は、そのプロパティのために入力する双方向スクリプトの形式を示す値を設定する必要があります。

表 12. Adapter for JD Edwards EnterpriseOne の外部サービス接続プロパティ

プロパティ名	説明
『環境』	JD Edwards EnterpriseOne 環境名を指定します。
『ログ・ファイル出力場所』	外部サービスのログ・ファイルの場所を指定します。
130 ページの『ロギング・レベル』	外部サービスの実行中にロギングを行うタイプのエラーを指定します。
130 ページの『パスワード』	JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントのパスワード。
131 ページの『ロール』	JD Edwards EnterpriseOne 環境へのアクセスに使用されるユーザー名に関連付けられたロール名。
131 ページの『ユーザー名』	JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントの名前。

環境

このプロパティは、JD Edwards EnterpriseOne 環境名を指定します。

表 13. 「環境」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティ・タイプ	String
使用法	JD Edwards EnterpriseOne 環境は、JD Edwards EnterpriseOne サーバー上のデータおよびオブジェクトの場所を示すユーザー定義のポインターです。ユーザーは、1 つの JD Edwards EnterpriseOne サーバーで複数の JD Edwards EnterpriseOne 環境を使用する権限を得ることができます。
グローバル化	はい
BIDI 対応	はい

ログ・ファイル出力場所

このプロパティでは、外部サービスのログ・ファイルの場所を指定します。

表 14. 「ログ・ファイル出力場所」の詳細

必須	はい
デフォルト	ワークスペースの .metadata ディレクトリー
プロパティ・タイプ	String
使用法	ディスカバリー・プロセスで発生したエラーを記述したログ・ファイルをこのディレクトリーに格納します。ロギングを行うディスカバリー・エラーのタイプは、ロギング・レベル・プロパティで制御します。

表 14. 「ログ・ファイル出力場所」の詳細 (続き)

例	C:\IBM\wid6.1.0\workspace\metadata\JDEMetadataDiscovery.log
グローバル化	はい
BIDI 対応	いいえ

ロギング・レベル

このプロパティーでは、外部サービスの実行時にロギングを行うタイプのエラーを指定します。

表 15. 「ロギング・レベル」の詳細

必須	いいえ
使用可能な値	ALL OFF FINE FINER FINEST CONFIG INFO SEVERE WARNING
デフォルト	SEVERE
プロパティー・タイプ	String
使用法	このプロパティーを使用して、トレース機能を調整します。エラー・タイプを指定すると、そのタイプのエラーについてのみトレース操作が実行されるようになります。
例	<p>デフォルト値の SEVERE を受け入れると、SEVERE カテゴリに属するエラーのトレース情報が記述されます。SEVERE のエラー (重大エラー) が発生すると、アダプターが機能していても、操作を続行できなくなります。重大エラーには、致命的エラーが差し迫っていることを示す (すなわちリソースが枯渇する寸前であることを強く示唆する状況を報告する) エラー状態も含まれます。</p> <p>その他のエラーの説明を以下にまとめます。</p> <ul style="list-style-type: none"> 致命的 - アダプターを続行できません。アダプターは機能しません。 警告 - 潜在的なエラーまたはエラーが差し迫っていることを示します。これには、リソース・リークの可能性など、障害が進行していることを示す状態も含まれます。 監査 - アダプターの状態やリソースに影響を及ぼす重要なイベント。 情報 - 操作全体の進行状況の概略を示す一般情報。 構成 - 構成変更または状況。 詳細 - 操作の進行状況の詳細を示す一般情報。
グローバル化	はい
BIDI 対応	いいえ

パスワード

このプロパティーは、JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントのパスワードを指定します。

表 16. 「パスワード」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティタイプ	String
使用法	パスワードは、JD Edwards EnterpriseOne 管理者によって作成され、指定されます。パスワードに使用される文字のタイプ、文字数、または大文字/小文字の指定に制限はありません。
グローバル化	いいえ
BIDI 対応	はい

ロール

このプロパティでは、JD Edwards EnterpriseOne 環境へのアクセスに使用されるユーザー名に関連付けられたロール名を指定します。

表 17. 「ロール」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティタイプ	String
使用法	ロールは、ユーザーに設定されている権限を定義します。ユーザーには、複数のロールを設定できます。特定のユーザーが、アプリケーション、フォーム、テーブル列、データ・ソースなどにアクセスする権限は、そのユーザーが割り当てられている 1 つ以上のロールによって決まります。ロールは、JD Edwards EnterpriseOne 管理者によって作成され、名前が指定されます。
例	<ul style="list-style-type: none"> • システム管理者 • 人事 • 会計
グローバル化	いいえ
BIDI 対応	はい

ユーザー名

このプロパティは、JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントの名前を指定します。

表 18. 「ユーザー名」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティタイプ	String
使用法	ユーザー名は、JD Edwards EnterpriseOne 管理者によって作成されます。ユーザー名に使用される文字のタイプ、文字数、または大文字/小文字の指定に制限はありません。
グローバル化	はい
BIDI 対応	はい

リソース・アダプター・プロパティ

リソース・アダプター・プロパティでは、アダプターの一般的な操作を制御します。アダプターの構成時にリソース・アダプター・プロパティを設定する場合は、外部サービス・ウィザードを使用します。アダプターのデプロイ後にこれらのプロパティを変更する場合は、管理コンソールを使用します。

ロギングおよびトレースに関する次のプロパティは、非推奨になっています。

- ログ・ファイル最大サイズ
- ログ・ファイル名
- ログ・ファイル数
- トレース・ファイル最大サイズ
- トレース・ファイル名
- トレース・ファイル数

バージョン 6.0.2.x でリソース・アダプター・プロパティとして指定されていた以下のプロパティは、バージョン 6.1.x では Managed Connection Factory プロパティのグループに該当します。

- タイムアウト

以下の表は、リソース・アダプター・プロパティをリストおよび説明しています。各プロパティの詳細については、表の後のセクションで説明します。表の後のセクションにあるプロパティ詳細表の読み方については、127 ページの『プロパティの詳細についてのガイド』を参照してください。

表 19. Adapter for JD Edwards EnterpriseOne のリソース・アダプター・プロパティ

名前		説明
ウィザード内	管理コンソール内	
アダプター ID	AdapterID	PMI イベントのアダプター・インスタンス、ロギングおよびトレースのアダプター・インスタンスを識別する場合に使用します。
ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する	HideConfidentialTrace	ログおよびトレース・ファイルへの書き込み時に、ユーザー・データを「X」のストリングで記述し、潜在的な機密データを隠すようにするかどうかを指定します。
(なし)	enableHASupport	このプロパティは変更しないでください。
(なし)	ログ・ファイル最大サイズ	非推奨
(なし)	LogFilename	非推奨
(なし)	ログ・ファイル数	非推奨
(なし)	トレース・ファイル最大サイズ	非推奨
(なし)	トレース・ファイル名	非推奨
(なし)	トレース・ファイル数	非推奨

アダプター ID (AdapterID)

このプロパティは、アダプターの特定のデプロイメント (インスタンス) を識別します。

表 20. 「アダプター ID」の詳細

必須	はい
デフォルト	001
プロパティ・タイプ	String
使用法	<p>このプロパティは、ログおよびトレース・ファイル内のアダプター・インスタンスを識別し、また、アダプターのモニター時にアダプター・インスタンスを識別する場合に役立ちます。アダプター ID は、アダプター固有の ID、JDERA と共に使用され、Log and Trace Analyzer ツールによって使用されるコンポーネント名を構成します。例えば、アダプター ID プロパティが、001 に設定されている場合、コンポーネント ID は、JDERA001 となります。</p> <p>同じアダプターを複数インスタンス実行する場合、アダプター ID プロパティの最初の 8 文字は、必ずインスタンスごとに固有にし、ログおよびトレース情報を特定のアダプター・インスタンスに相互に関連付けられるようにしてください。アダプター ID プロパティの最初の 7 文字を固有にすることにより、そのアダプターの複数インスタンスのコンポーネント ID も固有になり、アダプターの特定インスタンスにログおよびトレース情報を相互に関連付けることができます。</p> <p>アダプター ID プロパティの長さが、ログおよびトレース・ファイルのフィルター処理にどのように関係するかを示すため、2 つのインスタンスの WebSphere Adapter for JD Edwards EnterpriseOne のアダプター ID プロパティを 001 と 002 に設定するとします。これらのインスタンスのコンポーネント ID、JDERA001 および JDERA002 は、短いので固有性を保つことができ、別のアダプター・インスタンスとして区別することができます。しかし、もっと長いアダプター ID プロパティのインスタンスの場合、互いを区別できなくなります。例えば、2 つのインスタンスのアダプター ID プロパティを、Instance01 および Instance02 に設定するとします。この場合、各アダプター・インスタンスのログおよびトレース情報を個別に調べることはできなくなります。これは、両方のインスタンスのコンポーネント ID が JDERAInstance に切り詰められるためです。</p> <p>Inbound 処理の場合、このプロパティの値は、リソース・アダプター・レベルで設定されます。Outbound 処理の場合は、リソース・アダプター・レベルと Managed Connection Factory レベルの両方で設定できます。外部サービス・ウィザードを使用して、アダプターを Outbound 処理用に構成した後、リソース・アダプター・プロパティおよび Managed Connection Factory プロパティを個別に設定できます。WebSphere Integration Developer アセンブリー・エディターまたは管理コンソールを使用して、これらのプロパティを再設定する場合、ログおよびトレース・エントリーのマーキングが不整合にならないように、矛盾がない設定になっていることを確認してください。</p>
グローバル化	はい
BIDI 対応	いいえ

「ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する」 (HideConfidentialTrace) プロパティ

このプロパティは、ログおよびトレース・ファイル中のユーザー・データを「X」のストリングに置換し、潜在的な機密データが許可なく外部に漏れないようにします。

表 21. 「ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する」の詳細

必須	いいえ
使用可能な値	True False
デフォルト	False
プロパティ・タイプ	Boolean
使用法	このプロパティを True に設定すると、アダプターでは、ログおよびトレース・ファイルに書き込む時に、ユーザー・データを「X」のストリングに置換します。 Inbound 処理の場合、このプロパティの値は、リソース・アダプター・レベルで設定されます。Outbound 処理の場合は、リソース・アダプター・レベルと Managed Connection Factory レベルの両方で設定できます。外部サービス・ウィザード を使用して、アダプターを Outbound 処理用に構成した後、リソース・アダプター・プロパティおよび Managed Connection Factory プロパティを個別に設定できます。WebSphere Integration Developer アセンブリー・エディターまたは管理コンソールを使用して、これらのプロパティを再設定する場合、ログおよびトレース・エントリーのマーキングが不整合にならないように、矛盾がない設定になっていることを確認してください。
グローバル化	いいえ
BIDI 対応	いいえ

高可用性サポートを使用可能にする (Enable high availability support) (enableHASupport)

このプロパティは変更しないでください。true に設定してください。

ログ・ファイルの最大サイズ (Log file maximum size) (LogFileMaxSize)

このプロパティでは、ログ・ファイルのサイズをキロバイト単位で指定します。

表 22. 「ログ・ファイルの最大サイズ (Log file maximum size)」の詳細

必須	いいえ
デフォルト	0
プロパティ・タイプ	Integer
使用法	ログ・ファイルがその最大サイズに到達すると、アダプターは新規のログ・ファイルの使用を開始します。ファイル・サイズを 0 と指定するか、最大サイズを指定しなかった場合、そのファイルに最大サイズはありません。
グローバル化	はい
BIDI 対応	いいえ

ログ・ファイル名 (LogFilename)

このプロパティでは、ログ・ファイルの絶対パス名を指定します。

表 23. 「ログ・ファイル名」の詳細

必須	いいえ
----	-----

表 23. 「ログ・ファイル名」の詳細 (続き)

デフォルト	デフォルト値なし
プロパティ・タイプ	String
使用法	このプロパティは非推奨です。
グローバル化	はい
BIDI 対応	はい

ログ・ファイルの数 (Log number of files) (LogNumberOfFiles)

このプロパティでは、ログ・ファイルの数を指定します。

表 24. 「ログ・ファイルの数 (Log number of files)」の詳細

必須	いいえ
デフォルト	1
プロパティ・タイプ	Integer
使用法	ログ・ファイルがその最大サイズに到達すると、アダプターは別のログ・ファイルの使用を開始します。値を指定しなかった場合、アダプターはログ・ファイルを 1 つ作成します。
グローバル化	はい
BIDI 対応	いいえ

トレース・ファイルの最大サイズ (Trace file maximum size) (TraceFileMaxSize)

このプロパティでは、トレース・ファイルのサイズをキロバイト単位で指定します。

表 25. 「トレース・ファイルの最大サイズ (Trace file maximum size)」の詳細

必須	いいえ
デフォルト	0
プロパティ・タイプ	Integer
使用法	値を指定しなかった場合は、トレース・ファイルに最大サイズが設定されません。
グローバル化	はい
BIDI 対応	いいえ

トレース・ファイル名 (TraceFilename)

このプロパティでは、トレース・ファイルの絶対パスを指定します。

表 26. 「トレース・ファイル名」の詳細

必須	いいえ
デフォルト	デフォルト値なし
計測単位	キロバイト

表 26. 「トレース・ファイル名」の詳細 (続き)

プロパティタイプ	String
使用法	このプロパティは非推奨です。
グローバル化	はい
BIDI 対応	はい

トレース・ファイルの数 (Trace number of files) (TraceNumberOfFiles)

このプロパティでは、使用するトレース・ファイルの数を指定します。トレース・ファイルがその最大サイズに到達すると、アダプターは別のトレース・ファイルの使用を開始します。

表 27. 「トレース・ファイルの数 (Trace number of files)」の詳細

必須	いいえ
デフォルト	1
プロパティタイプ	Integer
使用法	値を指定しなかった場合、アダプターはトレース・ファイルを 1 つ使用します。
グローバル化	はい
BIDI 対応	いいえ

Managed Connection Factory プロパティ

Managed Connection Factory プロパティは、JD Edwards EnterpriseOne サーバーとの Outbound 接続インスタンスを作成するために、アダプターがランタイムに使用します。

Managed Connection Factory プロパティは、外部サービス・ウィザードまたは管理コンソール (デプロイメント後) を使用して設定します。

Managed Connection Factory プロパティとその説明を、以下の表に示します。各プロパティの詳細については、表の後のセクションで説明します。表の後のセクションにあるプロパティ詳細表の読み方については、127 ページの『プロパティの詳細についてのガイド』を参照してください。

注: 外部サービス・ウィザードは、これらのプロパティを Managed Connection Factory プロパティとして参照し、WebSphere Process Server または WebSphere Enterprise Service Bus 管理コンソールは、(J2C) 接続ファクトリー・プロパティとして参照します。

表 28. Adapter for JD Edwards EnterpriseOne の Managed Connection Factory プロパティ

プロパティ名		説明
ウィザード内	管理コンソール内	
アダプター ID	AdapterID	PMI イベントのアダプター・インスタンス、ロギングおよびトレースのアダプター・インスタンスを識別する場合に使用します。

表 28. Adapter for JD Edwards EnterpriseOne の Managed Connection Factory プロパティ (続き)

プロパティ名		説明
ウィザード内	管理コンソール内	
ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する	HideConfidentialTrace	ログおよびトレース・ファイルへの書き込み時に、ユーザー・データを「X」のストリングで記述し、潜在的な機密データを隠すようにするかどうかを指定します。
139 ページの『環境』	environment	JD Edwards EnterpriseOne 環境名を指定します。
139 ページの『パスワード』	password	JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントのパスワード。
140 ページの『ロール』	role	JD Edwards EnterpriseOne 環境へのアクセスに使用されるユーザー名に関連付けられたロール名。
140 ページの『タイムアウト』	timeout	このプロパティは、ミリ秒単位のグローバル・タイムアウト値で、XML List 要求実行呼び出しで設定されます。
140 ページの『ユーザー名』	userName	JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントの名前。

アダプター ID (AdapterID)

このプロパティは、アダプターの特定のデプロイメント (インスタンス) を識別します。

表 29. 「アダプター ID」の詳細

必須	はい
デフォルト	001
プロパティ・タイプ	String

表 29. 「アダプター ID」の詳細 (続き)

<p>使用法</p>	<p>このプロパティは、ログおよびトレース・ファイル内のアダプター・インスタンスを識別し、また、アダプターのモニター時にアダプター・インスタンスを識別する場合に役立ちます。アダプター ID は、アダプター固有の ID、JDERA と共に使用され、Log and Trace Analyzer ツールによって使用されるコンポーネント名を構成します。例えば、アダプター ID プロパティが、001 に設定されている場合、コンポーネント ID は、JDERA001 となります。</p> <p>同じアダプターを複数インスタンス実行する場合、アダプター ID プロパティの最初の 8 文字は、必ずインスタンスごとに固有にし、ログおよびトレース情報を特定のアダプター・インスタンスに相互に関連付けられるようにしてください。アダプター ID プロパティの最初の 7 文字を固有にすることにより、そのアダプターの複数インスタンスのコンポーネント ID も固有になり、アダプターの特定インスタンスにログおよびトレース情報を相互に関連付けることができます。</p> <p>アダプター ID プロパティの長さが、ログおよびトレース・ファイルのフィルター処理にどのように関係するかを示すため、2 つのインスタンスの WebSphere Adapter for JD Edwards EnterpriseOne のアダプター ID プロパティを 001 と 002 に設定するとします。これらのインスタンスのコンポーネント ID、JDERA001 および JDERA002 は、短いので固有性を保つことができ、別のアダプター・インスタンスとして区別することができます。しかし、もっと長いアダプター ID プロパティのインスタンスの場合、互いを区別できなくなります。例えば、2 つのインスタンスのアダプター ID プロパティを、Instance01 および Instance02 に設定するとします。この場合、各アダプター・インスタンスのログおよびトレース情報を個別に調べることはできなくなります。これは、両方のインスタンスのコンポーネント ID が JDERAInstance に切り詰められるためです。</p> <p>Inbound 処理の場合、このプロパティの値は、リソース・アダプター・レベルで設定されます。Outbound 処理の場合、リソース・アダプター・レベルと Managed Connection Factory レベルの両方で設定できます。外部サービス・ウィザードを使用して、アダプターを Outbound 処理用に構成した後、リソース・アダプター・プロパティおよび Managed Connection Factory プロパティを個別に設定できます。WebSphere Integration Developer アセンブリー・エディターまたは管理コンソールを使用して、これらのプロパティを再設定する場合、ログおよびトレース・エントリーのマーキングが不整合にならないように、矛盾がない設定になっていることを確認してください。</p>
<p>グローバル化</p>	<p>はい</p>
<p>BIDI 対応</p>	<p>いいえ</p>

「ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する」 (HideConfidentialTrace) プロパティ

このプロパティは、ログおよびトレース・ファイル中のユーザー・データを「X」のストリングに置換し、潜在的な機密データが許可なく外部に漏れないようにします。

表 30. 「ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する」の詳細

<p>必須</p>	<p>いいえ</p>
<p>使用可能な値</p>	<p>True False</p>
<p>デフォルト</p>	<p>False</p>
<p>プロパティ・タイプ</p>	<p>Boolean</p>

表 30. 「ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する」の詳細 (続き)

使用法	このプロパティを True に設定すると、アダプターでは、ログおよびトレース・ファイルに書き込む時に、ユーザー・データを「X」のストリングに置換します。 Inbound 処理の場合、このプロパティの値は、リソース・アダプター・レベルで設定されます。Outbound 処理の場合は、リソース・アダプター・レベルと Managed Connection Factory レベルの両方で設定できます。外部サービス・ウィザード を使用して、アダプターを Outbound 処理用に構成した後、リソース・アダプター・プロパティおよび Managed Connection Factory プロパティを個別に設定できます。WebSphere Integration Developer アセンブリー・エディターまたは管理コンソールを使用して、これらのプロパティを再設定する場合、ログおよびトレース・エントリーのマーキングが不整合にならないように、矛盾がない設定になっていることを確認してください。
グローバル化	いいえ
BIDI 対応	いいえ

環境

このプロパティは、JD Edwards EnterpriseOne 環境名を指定します。

表 31. 「環境」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティ・タイプ	String
使用法	JD Edwards EnterpriseOne 環境は、JD Edwards EnterpriseOne サーバー上のデータおよびオブジェクトの場所を示すユーザー定義のポインターです。ユーザーは、1 つの JD Edwards EnterpriseOne サーバーで複数の JD Edwards EnterpriseOne 環境を使用する権限を得ることができます。
例	
グローバル化	はい
BIDI 対応	はい

パスワード

このプロパティは、JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントのパスワードを指定します。

表 32. 「パスワード」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティ・タイプ	String
使用法	パスワードは、JD Edwards EnterpriseOne 管理者によって作成され、指定されます。パスワードに使用される文字のタイプ、文字数、または大文字/小文字の指定に制限はありません。
例	
グローバル化	いいえ
BIDI 対応	はい

ロール

このプロパティーでは、JD Edwards EnterpriseOne 環境へのアクセスに使用されるユーザー名に関連付けられたロール名を指定します。

表 33. 「ロール」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティー・タイプ	String
使用法	ロールは、ユーザーに設定されている権限を定義します。ユーザーには、複数のロールを設定できます。特定のユーザーが、アプリケーション、フォーム、テーブル列、データ・ソースなどにアクセスする権限は、そのユーザーが割り当てられている 1 つ以上のロールによって決まります。ロールは、JD Edwards EnterpriseOne 管理者によって作成され、名前が指定されます。
例	<ul style="list-style-type: none">システム管理者人事会計
グローバル化	いいえ
BIDI 対応	はい

タイムアウト

このプロパティーは、ミリ秒単位のタイムアウト値で、XML List 要求呼び出しで設定されます。

表 34. 「タイムアウト」の詳細

必須	はい
デフォルト	30,000
計測単位	ミリ秒
プロパティー・タイプ	Integer
使用法	「タイムアウト」プロパティーは、アダプターが XML List を使用した RetrieveAll 操作の実行にかかる時間を指定する場合に使用します。値を指定しない場合、アダプターでは、30 秒 (30,000 ミリ秒) 後にタイムアウトになります。
グローバル化	はい
BIDI 対応	いいえ

ユーザー名

このプロパティーは、JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントの名前を指定します。

表 35. 「ユーザー名」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティー・タイプ	String

表 35. 「ユーザー名」の詳細 (続き)

使用法	ユーザー名は、JD Edwards EnterpriseOne 管理者によって作成されます。ユーザー名に使用される文字のタイプ、文字数、または大文字/小文字の指定に制限はありません。
例	
グローバル化	はい
BIDI 対応	はい

対話仕様プロパティ

対話仕様プロパティは、操作の対話処理を制御します。アダプターの構成時には、外部サービス・ウィザードによって対話仕様プロパティを設定します。基本的に、それらのプロパティを変更する必要はありません。ただし、一部の Outbound 操作のプロパティは、ユーザーによる変更が可能です。

対話仕様プロパティを変更する理由としては、例えば、RetrieveAll 操作から完全な情報が返されない場合に、RetrieveAll 操作で返す最大レコード数を指定する対話仕様プロパティの値を大きくする、といったことが考えられます。アプリケーションのデプロイ後にこれらのプロパティを変更する場合、WebSphere Integration Developer のアセンブリー・エディターを使用します。これらのプロパティは、インポートのメソッド・バインディングの中にあります。

以下の表に、ユーザーが設定できる対話仕様プロパティを示します。プロパティ詳細の表の見方については、127 ページの『プロパティの詳細についてのガイド』を参照してください。

表 36. Adapter for JD Edwards EnterpriseOne の対話仕様プロパティ

プロパティ名		説明
ウィザード内	アセンブリー・エディター内	
RetrieveAll 操作の最大レコード数	最大レコード数	RetrieveAll 操作時に返すレコードの最大数。

RetrieveAll 操作のレコードの最大数

このプロパティは、RetrieveAll 操作が返すレコードの最大数を指定します。

表 37. 「RetrieveAll 操作の最大レコード数」の詳細

必須	はい
デフォルト	100
使用法	データベース内のヒット数がレコードの最大数プロパティの値を超えると、アダプターはエラー MatchesExceededLimitException および MatchesExceededLimitFault を返します。このプロパティは、メモリー不足の問題を回避するために使用します。
プロパティ・タイプ	Integer
グローバル化	いいえ
BIDI 対応	いいえ

Inbound 構成プロパティ

WebSphere Adapter for JD Edwards EnterpriseOne には、オブジェクトやサービスを生成したり作成したりするときに、外部サービス・ウィザードを使用して設定する、いくつかの種類の Inbound 接続構成プロパティがあります。リソース・アダプターおよびアクティベーション・スペックのプロパティは、モジュールをデプロイした後に WebSphere Integration Developer または 管理コンソール を使用して変更できますが、外部サービス・ウィザードの接続プロパティは、デプロイメント後に変更することはできません。

プロパティの詳細についてのガイド

WebSphere Adapter for JD Edwards EnterpriseOne を構成するときに使用されるプロパティは、リソース・アダプター・プロパティや Managed Connection Factory プロパティなど、それぞれの構成プロパティのトピックに記載されている表で詳細に説明されています。これらの表を使用しやすくするため、参照する各行の情報を以下に説明します。

次の表では、構成プロパティの表に表示される場合がある各行の意味を説明します。

行	説明
必須	<p>アダプターが動作するためには、必須フィールド (プロパティ) に値が必要です。必須プロパティに対しては、外部サービス・ウィザードがデフォルト値を提供する場合があります。</p> <p>外部サービス・ウィザードの必須フィールドからデフォルト値を除去しても、デフォルト値は変更されません。必須フィールドに値がまったく入っていないと、外部サービス・ウィザードは、その割り当て済みのデフォルト値を使用してフィールドを処理し、そのデフォルト値は管理コンソールにも表示されます。</p> <p>可能な値は「はい」および「いいえ」です。</p> <p>プロパティは、他のプロパティが特定の値の場合のみ必須となることがあります。その場合は、表にこの依存関係が記載されます。以下に例を示します。</p> <ul style="list-style-type: none">• EventQueryType プロパティが Dynamic に設定された場合は「はい」• Oracle データベースの場合は「はい」
使用可能な値	プロパティで選択可能な値をリストして説明します。
デフォルト	<p>外部サービス・ウィザードによって設定される事前定義値。プロパティが必須の場合は、デフォルト値を受け入れるか、ユーザーが値を指定する必要があります。プロパティにデフォルト値がない場合、表には「デフォルト値なし」と記載されます。</p> <p>None という語は、受け入れ可能なデフォルト値です。デフォルト値がないという意味ではありません。</p>
計測単位	プロパティの計測単位を指定します (例: キロバイト、秒)。
プロパティ・タイプ	<p>プロパティ・タイプを示します。有効なプロパティ・タイプとしては、以下のものがあります。</p> <ul style="list-style-type: none">• Boolean• String• Integer

行	説明
使用法	<p>プロパティに適用される場合がある使用の条件または制限について記述します。制限の記載例を以下に示します。</p> <p>Rational Application Developer for WebSphere Software バージョン 6.40 またはそれ以前では、パスワードに以下の制限があります。</p> <ul style="list-style-type: none"> • 大文字である必要があります • 長さが 8 文字である必要があります <p>Rational Application Developer for WebSphere Software バージョン 6.40 よりも後のバージョンでは、パスワードの制限が以下のように変更されました。</p> <ul style="list-style-type: none"> • 大文字小文字を区別しません • 長さが 40 文字まで可能です <p>このセクションでは、このプロパティに影響を及ぼす他のプロパティ、またはこのプロパティによって影響を受ける他のプロパティをリストし、その条件付き関係の内容を説明します。</p>
例	<p>プロパティ値のサンプルを示します。例:</p> <p>「言語が JA (日本語) に設定された場合、コード・ページ番号は 8000 に設定されます。」</p>
グローバル化	<p>グローバル化されたプロパティには各国語サポートが備わっているので、値を各国語で設定できます。</p> <p>有効な値は「はい」および「いいえ」です。</p>
BIDI 対応	<p>プロパティが双方向 (bidi) 処理でサポートされるかどうかを示します。双方向処理とは、1 つのファイルに左から右 (ヘブライ語やアラビア語など) と右から左 (URL やファイル・パスなど) の両方の意味内容を含むデータを処理するタスクを指します。</p> <p>有効な値は「はい」および「いいえ」です。</p>

外部サービス・ウィザードの接続プロパティ

外部サービス・ディスカバリー接続プロパティでは、ビジネス・オブジェクトを作成するためのツールである WebSphere Integration Developer の外部サービス・ウィザードと JD Edwards EnterpriseOne サーバー の間の接続を確立します。外部サービス・ウィザードで構成するプロパティでは、接続構成、BiDi プロパティ、トレース/ロギング・オプションなどを指定します。

外部サービス・ウィザードと JD Edwards EnterpriseOne サーバーの間の接続を確立すると、外部サービス・ウィザードは、ビジネス・オブジェクトを作成するために JD Edwards EnterpriseOne サーバーから取得しなければならないメタデータにアクセスできるようになります。

注: 外部サービス・ウィザードで設定するプロパティのいくつかは、後からそのウィザードで指定できるリソース・アダプター・プロパティおよび Managed Connection Factory プロパティの初期値として使用されます。

外部サービス接続プロパティと各プロパティの目的を以下の表にまとめます。各プロパティの完全な説明は、表に続くセクションに記載しています。表の後の

セクションにあるプロパティ詳細表の読み方については、127 ページの『プロパティの詳細についてのガイド』を参照してください。

重要: 双方向スクリプトを使用する接続プロパティを設定する場合は、そのプロパティのために入力する双方向スクリプトの形式を示す値を設定する必要があります。

表 38. Adapter for JD Edwards EnterpriseOne の外部サービス接続プロパティ

プロパティ名	説明
『環境』	JD Edwards EnterpriseOne 環境名を指定します。
『ログ・ファイル出力場所』	外部サービスのログ・ファイルの場所を指定します。
145 ページの『ロギング・レベル』	外部サービスの実行中にロギングを行うタイプのエラーを指定します。
145 ページの『パスワード』	JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントのパスワード。
146 ページの『ロール』	JD Edwards EnterpriseOne 環境へのアクセスに使用されるユーザー名に関連付けられたロール名。
146 ページの『ユーザー名』	JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントの名前。

環境

このプロパティは、JD Edwards EnterpriseOne 環境名を指定します。

表 39. 「環境」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティ・タイプ	String
使用法	JD Edwards EnterpriseOne 環境は、JD Edwards EnterpriseOne サーバー上のデータおよびオブジェクトの場所を示すユーザー定義のポインターです。ユーザーは、1 つの JD Edwards EnterpriseOne サーバーで複数の JD Edwards EnterpriseOne 環境を使用する権限を得ることができます。
グローバル化	はい
BIDI 対応	はい

ログ・ファイル出力場所

このプロパティでは、外部サービスのログ・ファイルの場所を指定します。

表 40. 「ログ・ファイル出力場所」の詳細

必須	はい
デフォルト	ワークスペースの .metadata ディレクトリー
プロパティ・タイプ	String
使用法	ディスカバリー・プロセスで発生したエラーを記述したログ・ファイルをこのディレクトリーに格納します。ロギングを行うディスカバリー・エラーのタイプは、ロギング・レベル・プロパティで制御します。

表 40. 「ログ・ファイル出力場所」の詳細 (続き)

例	C:\IBM\wid6.1.0\workspace\metadata\JDEMetadataDiscovery.log
グローバル化	はい
BIDI 対応	いいえ

ロギング・レベル

このプロパティーでは、外部サービスの実行時にロギングを行うタイプのエラーを指定します。

表 41. 「ロギング・レベル」の詳細

必須	いいえ
使用可能な値	ALL OFF FINE FINER FINEST CONFIG INFO SEVERE WARNING
デフォルト	SEVERE
プロパティー・タイプ	String
使用法	このプロパティーを使用して、トレース機能を調整します。エラー・タイプを指定すると、そのタイプのエラーについてのみトレース操作が実行されるようになります。
例	<p>デフォルト値の SEVERE を受け入れると、SEVERE カテゴリに属するエラーのトレース情報が記述されます。SEVERE のエラー (重大エラー) が発生すると、アダプターが機能していても、操作を続行できなくなります。重大エラーには、致命的エラーが差し迫っていることを示す (すなわちリソースが枯渇する寸前であることを強く示唆する状況を報告する) エラー状態も含まれます。</p> <p>その他のエラーの説明を以下にまとめます。</p> <ul style="list-style-type: none"> • 致命的 - アダプターを続行できません。アダプターは機能しません。 • 警告 - 潜在的なエラーまたはエラーが差し迫っていることを示します。これには、リソース・リークの可能性など、障害が進行していることを示す状態も含まれます。 • 監査 - アダプターの状態やリソースに影響を及ぼす重要なイベント。 • 情報 - 操作全体の進行状況の概略を示す一般情報。 • 構成 - 構成変更または状況。 • 詳細 - 操作の進行状況の詳細を示す一般情報。
グローバル化	はい
BIDI 対応	いいえ

パスワード

このプロパティーは、JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントのパスワードを指定します。

表 42. 「パスワード」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティタイプ	String
使用法	パスワードは、JD Edwards EnterpriseOne 管理者によって作成され、指定されます。パスワードに使用される文字のタイプ、文字数、または大文字/小文字の指定に制限はありません。
グローバル化	いいえ
BIDI 対応	はい

ロール

このプロパティでは、JD Edwards EnterpriseOne 環境へのアクセスに使用されるユーザー名に関連付けられたロール名を指定します。

表 43. 「ロール」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティタイプ	String
使用法	ロールは、ユーザーに設定されている権限を定義します。ユーザーには、複数のロールを設定できます。特定のユーザーが、アプリケーション、フォーム、テーブル列、データ・ソースなどにアクセスする権限は、そのユーザーが割り当てられている 1 つ以上のロールによって決まります。ロールは、JD Edwards EnterpriseOne 管理者によって作成され、名前が指定されます。
例	<ul style="list-style-type: none"> システム管理者 人事 会計
グローバル化	いいえ
BIDI 対応	はい

ユーザー名

このプロパティは、JD Edwards EnterpriseOne 環境のアダプター・ユーザー・アカウントの名前を指定します。

表 44. 「ユーザー名」の詳細

必須	はい
デフォルト	デフォルト値なし
プロパティタイプ	String
使用法	ユーザー名は、JD Edwards EnterpriseOne 管理者によって作成されます。ユーザー名に使用される文字のタイプ、文字数、または大文字/小文字の指定に制限はありません。
グローバル化	はい
BIDI 対応	はい

リソース・アダプター・プロパティ

リソース・アダプター・プロパティでは、アダプターの一般的な操作を制御します。アダプターの構成時にリソース・アダプター・プロパティを設定する場合は、外部サービス・ウィザードを使用します。アダプターのデプロイ後にこれらのプロパティを変更する場合は、管理コンソールを使用します。

ロギングおよびトレースに関する次のプロパティは、非推奨になっています。

- ログ・ファイル最大サイズ
- ログ・ファイル名
- ログ・ファイル数
- トレース・ファイル最大サイズ
- トレース・ファイル名
- トレース・ファイル数

バージョン 6.0.2.x でリソース・アダプター・プロパティとして指定されていた以下のプロパティは、バージョン 6.1.x では Managed Connection Factory プロパティのグループに該当します。

- タイムアウト

以下の表は、リソース・アダプター・プロパティをリストおよび説明しています。各プロパティの詳細については、表の後のセクションで説明します。表の後のセクションにあるプロパティ詳細表の読み方については、127 ページの『プロパティの詳細についてのガイド』を参照してください。

表 45. Adapter for JD Edwards EnterpriseOne のリソース・アダプター・プロパティ

名前		説明
ウィザード内	管理コンソール内	
アダプター ID	AdapterID	PMI イベントのアダプター・インスタンス、ロギングおよびトレースのアダプター・インスタンスを識別する場合に使用します。
ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する	HideConfidentialTrace	ログおよびトレース・ファイルへの書き込み時に、ユーザー・データを「X」のストリングで記述し、潜在的な機密データを隠すようにするかどうかを指定します。
(なし)	enableHASupport	このプロパティは変更しないでください。
(なし)	ログ・ファイル最大サイズ	非推奨
(なし)	LogFilename	非推奨
(なし)	ログ・ファイル数	非推奨
(なし)	トレース・ファイル最大サイズ	非推奨
(なし)	トレース・ファイル名	非推奨
(なし)	トレース・ファイル数	非推奨

アダプター ID (AdapterID)

このプロパティは、アダプターの特定のデプロイメント (インスタンス) を識別します。

表 46. 「アダプター ID」の詳細

必須	はい
デフォルト	001
プロパティ・タイプ	String
使用法	<p>このプロパティは、ログおよびトレース・ファイル内のアダプター・インスタンスを識別し、また、アダプターのモニター時にアダプター・インスタンスを識別する場合に役立ちます。アダプター ID は、アダプター固有の ID、JDERA と共に使用され、Log and Trace Analyzer ツールによって使用されるコンポーネント名を構成します。例えば、アダプター ID プロパティが、001 に設定されている場合、コンポーネント ID は、JDERA001 となります。</p> <p>同じアダプターを複数インスタンス実行する場合、アダプター ID プロパティの最初の 8 文字は、必ずインスタンスごとに固有にし、ログおよびトレース情報を特定のアダプター・インスタンスに相互に関連付けられるようにしてください。アダプター ID プロパティの最初の 7 文字を固有にすることにより、そのアダプターの複数インスタンスのコンポーネント ID も固有になり、アダプターの特定インスタンスにログおよびトレース情報を相互に関連付けることができます。</p> <p>アダプター ID プロパティの長さが、ログおよびトレース・ファイルのフィルター処理にどのように関係するかを示すため、2 つのインスタンスの WebSphere Adapter for JD Edwards EnterpriseOne のアダプター ID プロパティを 001 と 002 に設定するとします。これらのインスタンスのコンポーネント ID、JDERA001 および JDERA002 は、短いので固有性を保つことができ、別のアダプター・インスタンスとして区別することができます。しかし、もっと長いアダプター ID プロパティのインスタンスの場合、互いを区別できなくなります。例えば、2 つのインスタンスのアダプター ID プロパティを、Instance01 および Instance02 に設定するとします。この場合、各アダプター・インスタンスのログおよびトレース情報を個別に調べることはできなくなります。これは、両方のインスタンスのコンポーネント ID が JDERAInstance に切り詰められるためです。</p> <p>Inbound 処理の場合、このプロパティの値は、リソース・アダプター・レベルで設定されます。Outbound 処理の場合は、リソース・アダプター・レベルと Managed Connection Factory レベルの両方で設定できます。外部サービス・ウィザードを使用して、アダプターを Outbound 処理用に構成した後、リソース・アダプター・プロパティおよび Managed Connection Factory プロパティを個別に設定できます。WebSphere Integration Developer アセンブリー・エディターまたは管理コンソールを使用して、これらのプロパティを再設定する場合、ログおよびトレース・エントリーのマーキングが不整合にならないように、矛盾がない設定になっていることを確認してください。</p>
グローバル化	はい
BIDI 対応	いいえ

「ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する」 (HideConfidentialTrace) プロパティ

このプロパティは、ログおよびトレース・ファイル中のユーザー・データを「X」のストリングに置換し、潜在的な機密データが許可なく外部に漏れないようにします。

表 47. 「ログ・ファイルおよびトレース・ファイルで、ユーザー・データを「XXX」と記述する」の詳細

必須	いいえ
使用可能な値	True False
デフォルト	False
プロパティ・タイプ	Boolean
使用法	このプロパティを True に設定すると、アダプターでは、ログおよびトレース・ファイルに書き込む時に、ユーザー・データを「X」のストリングに置換します。 Inbound 処理の場合、このプロパティの値は、リソース・アダプター・レベルで設定されます。Outbound 処理の場合は、リソース・アダプター・レベルと Managed Connection Factory レベルの両方で設定できます。外部サービス・ウィザード を使用して、アダプターを Outbound 処理用に構成した後、リソース・アダプター・プロパティおよび Managed Connection Factory プロパティを個別に設定できます。WebSphere Integration Developer アセンブリー・エディターまたは管理コンソールを使用して、これらのプロパティを再設定する場合、ログおよびトレース・エントリーのマーキングが不整合にならないように、矛盾がない設定になっていることを確認してください。
グローバル化	いいえ
BIDI 対応	いいえ

高可用性サポートを使用可能にする (Enable high availability support) (enableHASupport)

このプロパティは変更しないでください。true に設定してください。

ログ・ファイルの最大サイズ (Log file maximum size) (LogFileMaxSize)

このプロパティでは、ログ・ファイルのサイズをキロバイト単位で指定します。

表 48. 「ログ・ファイルの最大サイズ (Log file maximum size)」の詳細

必須	いいえ
デフォルト	0
プロパティ・タイプ	Integer
使用法	ログ・ファイルがその最大サイズに到達すると、アダプターは新規のログ・ファイルの使用を開始します。ファイル・サイズを 0 と指定するか、最大サイズを指定しなかった場合、そのファイルに最大サイズはありません。
グローバル化	はい
BIDI 対応	いいえ

ログ・ファイル名 (LogFilename)

このプロパティでは、ログ・ファイルの絶対パス名を指定します。

表 49. 「ログ・ファイル名」の詳細

必須	いいえ
----	-----

表 49. 「ログ・ファイル名」の詳細 (続き)

デフォルト	デフォルト値なし
プロパティ・タイプ	String
使用法	このプロパティは非推奨です。
グローバル化	はい
BIDI 対応	はい

ログ・ファイルの数 (Log number of files) (LogNumberOfFiles)

このプロパティでは、ログ・ファイルの数を指定します。

表 50. 「ログ・ファイルの数 (Log number of files)」の詳細

必須	いいえ
デフォルト	1
プロパティ・タイプ	Integer
使用法	ログ・ファイルがその最大サイズに到達すると、アダプターは別のログ・ファイルの使用を開始します。値を指定しなかった場合、アダプターはログ・ファイルを 1 つ作成します。
グローバル化	はい
BIDI 対応	いいえ

トレース・ファイルの最大サイズ (Trace file maximum size) (TraceFileMaxSize)

このプロパティでは、トレース・ファイルのサイズをキロバイト単位で指定します。

表 51. 「トレース・ファイルの最大サイズ (Trace file maximum size)」の詳細

必須	いいえ
デフォルト	0
プロパティ・タイプ	Integer
使用法	値を指定しなかった場合は、トレース・ファイルに最大サイズが設定されません。
グローバル化	はい
BIDI 対応	いいえ

トレース・ファイル名 (TraceFilename)

このプロパティでは、トレース・ファイルの絶対パスを指定します。

表 52. 「トレース・ファイル名」の詳細

必須	いいえ
デフォルト	デフォルト値なし
計測単位	キロバイト

表 52. 「トレース・ファイル名」の詳細 (続き)

プロパティ・タイプ	String
使用法	このプロパティは非推奨です。
グローバル化	はい
BIDI 対応	はい

トレース・ファイルの数 (Trace number of files) (TraceNumberOfFiles)

このプロパティでは、使用するトレース・ファイルの数を指定します。トレース・ファイルがその最大サイズに到達すると、アダプターは別のトレース・ファイルの使用を開始します。

表 53. 「トレース・ファイルの数 (Trace number of files)」の詳細

必須	いいえ
デフォルト	1
プロパティ・タイプ	Integer
使用法	値を指定しなかった場合、アダプターはトレース・ファイルを 1 つ使用します。
グローバル化	はい
BIDI 対応	いいえ

活動化仕様プロパティ

活動化仕様プロパティには、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。活動化仕様プロパティは、外部サービス・ウィザードまたは管理コンソールを使用して設定できます。

活動化仕様プロパティは、エンドポイントのアクティベーション時に、適格なイベント・リスナーについてアダプターに通知するために使用されます。Inbound 処理中に、アダプターはこれらのイベント・リスナーを使用してイベントを受信してから、エンドポイントにイベントを転送します。

以下の表は、Inbound 通信の活動化仕様プロパティを示します。活動化仕様プロパティは、外部サービス・ウィザードを使用して設定します。これらのプロパティは、WebSphere Integration Developer アセンブリー・エディターを使用して変更することができます。また、デプロイメント後には、管理コンソールを使用して変更することができます。各プロパティの詳細については、表の後のセクションで説明します。表の後のセクションにあるプロパティ詳細表の読み方については、127 ページの『プロパティの詳細についてのガイド』を参照してください。

表 54. 活動化仕様プロパティ

プロパティ名		説明
ウィザード内	管理コンソール内	
自動確認	AutoAcknowledge	使用されるイベント確認モードを指定します。

表 54. 活動化仕様プロパティ (続き)

プロパティ名		説明
ウィザード内	管理コンソール内	
送達タイプ	DeliveryType	イベントがアダプターによってエクスポートに配信される順序を指定します。 .
イベントを一度のみ送達する	AssuredOnceDelivery	アダプターにより、1 回のイベント送達を確保する機能が提供されるかどうかを指定します。 .
154 ページの『失敗イベント・フォルダー』	FailedEventsFolder	処理に失敗したイベントがファイル形式で保存されるローカル・システム上のファイル・フォルダーへの絶対パス。
失敗したイベントの再試行制限 (FailedEventRetryLimit)	FailedEventRetryLimit	アダプターの再送信の試行回数で、この回数に達すると失敗とマークされます。
ポーリング期間の間隔	ポーリング間隔	ポーリング期間中にアダプターが待機する時間の長さ。
最大接続数	MaximumConnections	アダプターが Inbound イベント送達に使用できる接続の最大数。
ポーリング期間内の最大イベント数	ポーリング数量	各ポーリング期間中にアダプターがエクスポートに配信するイベント数。
最小接続数	MinimumConnections	アダプターが Inbound イベント送達に使用できる接続の最小数。
待機なし	NoWait	アダプターが一定の時間間隔の間待機してから、Dynamic Java Connector API を起動し、JD Edwards EnterpriseOne トランザクション・サーバーからイベントを取得するかどうかを指定します。
システム接続を再試行する回数	RetryLimit	エラーが発生したあと、アダプターが Inbound 接続の再確立を試行する回数。 .
開始時に EIS 接続を再試行する	RetryConnectionOnStartup	アダプターが始動時に JD Edwards EnterpriseOne サーバー に接続できない場合に、接続を再試行するかどうかを指定します。
接続が失敗した場合の再試行間隔	RetryInterval	Inbound 操作時のエラー後、新規接続を確立しようとする試行間にアダプターが待機する時間の長さ。
ポーリング時にエラーが検出された場合はアダプターを停止する	StopPollingOnError	ポーリング時にアダプターがエラーを検出した場合、アダプターがイベントのポーリングを停止するかどうかを指定します。 .
待機時間	WaitTime	「待機なし」プロパティが False の場合の待機時間を指定します。

自動確認 (AutoAcknowledge)

このプロパティは、使用されるイベント確認モードを指定します。自動確認モードまたはクライアント確認モードのいずれかを指定できます。

表 55. 「自動確認」の詳細

必須	いいえ
使用可能な値	True False
デフォルト	False

表 55. 「自動確認」の詳細 (続き)

プロパティ・タイプ	Boolean
使用法	イベント確認モードを指定します。モードは、自動確認モードまたはクライアント確認モードのいずれかです。
例	False
グローバル化	いいえ
BIDI 対応	はい

送達のタイプ (DeliveryType)

このプロパティでは、イベントがアダプターによってエクスポートに配信される順序を指定します。

表 56. 送達のタイプの詳細

必須	いいえ
使用可能な値	ORDERED UNORDERED
デフォルト	ORDERED
プロパティ・タイプ	String
使用法	以下の値がサポートされています。 <ul style="list-style-type: none"> • ORDERED: アダプターは、一度に 1 つのイベントをエクスポートに配信します。 • UNORDERED: アダプターは、一度にすべてのイベントをエクスポートに配信します。
グローバル化	いいえ
BIDI 対応	いいえ

イベント送達は 1 回のみ (AssuredOnceDelivery)

このプロパティでは、Inbound イベントに対して、「イベントを一度のみ送達する」の機能を提供するかどうかを指定します。

表 57. 「イベント送達は 1 回のみ」の詳細

必須	はい
使用可能な値	True False
デフォルト	True
プロパティ・タイプ	Boolean

表 57. 「イベント送達は 1 回のみ」の詳細 (続き)

使用法	<p>このプロパティを True に設定すると、アダプターにより、1 回のイベント送達を確保する機能が提供されます。つまり、各イベントは 1 回のみ配信されます。値を False にすると、1 回のイベント送達を確保する機能は提供されませんが、パフォーマンスは向上します。</p> <p>このプロパティを True に設定すると、アダプターにより、トランザクション (XID) 情報のイベント・ストアへの保管が試行されます。このプロパティを False に設定した場合は、アダプターではこの情報の保管は行われません。</p> <p>このプロパティは、エクスポート・コンポーネントがトランザクションの対象である場合にのみ使用されます。そうでない場合は、このプロパティの値に関係なく、トランザクションを使用することはできません。</p>
グローバル化	いいえ
BIDI 対応	いいえ

失敗イベント・フォルダー

このプロパティは、処理に失敗したイベントがファイル形式で保存されるローカル・システム上のファイル・フォルダーを指定します。

表 58. 「失敗イベント・フォルダー」の詳細

必須	いいえ
使用可能な値	デフォルト値なし
デフォルト	NULL
プロパティ・タイプ	String
使用法	アダプターが実行される前に、このフォルダーを同じシステム上に手動で作成します。失敗イベント・フォルダーが指定されていない場合、アダプターは処理に失敗した イベントを保存しません。
グローバル化	はい
BIDI 対応	いいえ

失敗したイベントの再試行制限 (FailedEventRetryLimit)

このプロパティは、アダプターがイベントの再送信を試みる回数を指定します。この回数に達すると失敗とマークされます。

表 59. 「失敗したイベントの再試行制限」の詳細

必須	いいえ
使用可能な値	整数
デフォルト	5
プロパティ・タイプ	Integer

表 59. 「失敗したイベントの再試行制限」の詳細 (続き)

使用法	このプロパティは、この回数以上は失敗とマークする場合の、アダプターがイベントを送信する回数を指定する時に使用します。以下のいずれかの値を取ります。 デフォルト このプロパティが設定されない場合、アダプターは、イベント送信失敗の後、さらに 5 回イベントの送信を試み、それでも送達できない場合にイベントを失敗とマーク付けします。 0 アダプターは、回数無制限でイベントの送信を試行します。このプロパティが 0 に設定されると、イベントはイベント・ストアに残されたままになり、イベントが失敗とマークされることはなくなります。 > 0 正の整数の場合、アダプターは、指定した回数再試行を行った後、イベントを失敗とマークします。 <0 負の整数の場合、アダプターは失敗したイベントの送信を再試行しません。
グローバル化	いいえ
BIDI 対応	いいえ

ポーリング期間の間隔 (ポーリング間隔)

このプロパティでは、ポーリング期間中にアダプターが待機する時間の長さを指定します。

表 60. 「ポーリング期間の間隔」の詳細

必須	はい
使用可能な値	0 以上の整数
デフォルト	2000
計測単位	ミリ秒
プロパティ・タイプ	Integer
使用法	ポーリング期間は一定の割合で確立されます。つまり、ポーリング周期の実行が何らかの理由で遅延すると (例えば、前のポーリング周期が完了するまでに予想より時間がかかった場合)、遅延によって失った時間を取り戻すために次のポーリング周期がすぐに開始されます。
グローバル化	いいえ
BIDI 対応	いいえ

最大接続数 (Maximum connections) (MaximumConnections)

このプロパティでは、アダプターが Inbound イベント送達に使用できる接続の最大数を指定します。

表 61. 「最大接続数 (Maximum connections)」の詳細

必須	いいえ
デフォルト	1
プロパティ・タイプ	Integer

表 61. 「最大接続数 (Maximum connections)」の詳細 (続き)

使用法	正の値のみが有効です。アダプターは、1 より小さい正の入力値を 1 であるとみなします。このプロパティに対して負の値または 1 を入力すると、実行時エラーが発生することがあります。
グローバル化	いいえ
BIDI 対応	いいえ

ポーリング期間内の最大イベント数 (ポーリング数量)

このプロパティでは、各ポーリング期間中にアダプターがエクスポートに配信するイベント数を指定します。

表 62. 「ポーリング期間内の最大イベント数」の詳細

必須	はい
デフォルト	10
プロパティ・タイプ	Integer
使用法	値は 0 より大きくする必要があります。この値を大きくすると、ポーリング期間ごとに処理されるイベントの数が増加し、アダプターのパフォーマンス効率が低下する場合があります。この値を小さくすると、ポーリング期間ごとに処理されるイベントの数が減少し、アダプターのパフォーマンスが若干向上することがあります。
グローバル化	いいえ
BIDI 対応	いいえ

最小接続数 (Minimum connections) (MinimumConnections)

このプロパティでは、アダプターが Inbound イベント送達に使用できる接続の最小数を指定します。

表 63. 「最小接続数 (Minimum connections)」の詳細

必須	いいえ
デフォルト	1
プロパティ・タイプ	Integer
使用法	正の値のみが有効です。1 より小さい値は、アダプターによって 1 として処理されます。このプロパティに対して負の値または 1 を入力すると、実行時エラーが発生することがあります。
グローバル化	いいえ
BIDI 対応	いいえ

待機なし (NoWait)

このプロパティは、アダプターが一定の時間間隔の間待機してから、Dynamic Java Connector API を起動し、JD Edwards EnterpriseOne トランザクション・サーバーからイベントを取得するかどうかを指定します。

表 64. 「待機なし」の詳細

必須	いいえ
使用可能な値	True False
デフォルト	True
プロパティ・タイプ	Boolean
使用法	アダプターが一定の時間間隔の間待機してから、Dynamic Java Connector API を起動し、JD Edwards EnterpriseOne トランザクション・サーバーからイベントを取得するかどうかを指定します。
例	True
グローバル化	いいえ
BIDI 対応	はい

システム接続を再試行する回数 (RetryLimit)

このプロパティでは、アダプターが Inbound 接続の再確立を試行する回数を指定します。

表 65. 「システム接続を再試行する回数」の詳細

必須	いいえ
使用可能な値	0 および正の整数
デフォルト	0
プロパティ・タイプ	Integer
使用法	このプロパティは、アダプターが JD Edwards EnterpriseOne サーバー に接続して Inbound 処理を実行できない場合に、接続を再試行する回数を指定します。値が 0 の場合は、再試行回数が無制限になることを指定します。 アダプターの当初始動時に、JD Edwards EnterpriseOne サーバー に接続できない場合、アダプターが再試行するかどうかを制御するには、RetryConnectionOnStartup プロパティを使用します。
グローバル化	はい
BIDI 対応	いいえ

開始時に EIS 接続を再試行する (RetryConnectionOnStartup)

このプロパティは、アダプターが始動時に JD Edwards EnterpriseOne サーバー に接続できない場合に、再度接続を試みるかどうかを指定します。

表 66. 「開始時に EIS 接続を再試行する」の詳細

必須	いいえ
使用可能な値	True False
デフォルト	False
プロパティ・タイプ	Boolean

表 66. 「開始時に EIS 接続を再試行する」の詳細 (続き)

使用法	<p>このプロパティは、アダプターの始動時に、JD Edwards EnterpriseOne サーバー に接続できない場合に、接続を再試行するかどうかを指定します。</p> <ul style="list-style-type: none"> アダプターが、JD Edwards EnterpriseOne サーバー に対する接続を確立できたかどうかに関するフィードバックを直ちに必要とする場合、例えば、アダプターからのイベントを受信するアプリケーションを作成し、テストしている場合は、このプロパティを <code>False</code> に設定します。アダプターが接続できない場合、アダプターは、ログおよびトレース情報を書き込んで、停止します。管理コンソールは、アプリケーション状況を <code>Stopped</code> と表示します。この場合、接続の問題を解決後、手動でアダプターを始動してください。 接続に関するフィードバックをすぐに必要としない場合は、このプロパティは <code>True</code> に設定します。アダプターが始動時に接続できない場合、アダプターはログおよびトレース情報を書き込んでから、<code>RetryInterval</code> プロパティで再試行の頻度を判別して再接続を試み、<code>RetryLimit</code> プロパティの値で指定された値に達するまで、再試行を複数回行います。管理コンソールは、アプリケーション状況を <code>Started</code> と表示します。
グローバル化	いいえ
BIDI 対応	いいえ

接続が失敗した場合の再試行間隔 (RetryInterval)

このプロパティでは、アダプターが Inbound 接続に関連したエラーを検出した場合に、アダプターが新規接続を確立しようとするまで待機する時間の長さを指定します。

表 67. 再試行間隔の詳細

必須	はい
デフォルト	2000
計測単位	ミリ秒
プロパティ・タイプ	Integer
使用法	<p>正の値のみが有効です。このプロパティでは、アダプターが Inbound 接続に関連したエラーを検出した場合に、アダプターが新規接続を確立しようとするまで待機する時間の長さを指定します。</p>
グローバル化	はい
BIDI 対応	いいえ

ポーリング時にエラーが検出された場合はアダプターを停止する (StopPollingOnError)

このプロパティでは、ポーリング時にアダプターがエラーを検出した場合、アダプターがイベントのポーリングを停止するかどうかを指定します。

表 68. 「ポーリング時にエラーが検出された場合はアダプターを停止する」の詳細

必須	いいえ
使用可能な値	<p>True False</p>
デフォルト	False

表 68. 「ポーリング時にエラーが検出された場合はアダプターを停止する」の詳細 (続き)

プロパティタイプ	Boolean
使用法	このプロパティを True に設定した場合、アダプターはエラーを検出するとポーリングを停止します。 このプロパティを False に設定した場合、アダプターはポーリング時にエラーを検出すると例外をログに記録し、ポーリングを続行します。
グローバル化	いいえ
BIDI 対応	いいえ

待機時間 (WaitTime)

このプロパティは、「待機なし」プロパティが False の場合の待機時間を指定します。

表 69. 「待機時間」の詳細

必須	いいえ
使用可能な値	任意の正整数。負の整数は、デフォルト値 (3000 ミリ秒) とみなされます。
デフォルト	3000
計測単位	ミリ秒
プロパティタイプ	Integer
使用法	このプロパティは、「待機なし」プロパティが False の場合の待機時間を指定します。
例	5000
グローバル化	いいえ
BIDI 対応	はい

グローバル化

WebSphere Adapter for JD Edwards EnterpriseOne は、複数の言語および国/地域別環境で使用することができる、グローバル化されたアプリケーションです。アダプターは、文字セット・サポートおよびホスト・サーバーのロケールに基づいて、メッセージ・テキストを適切な言語で送信します。アダプターは、統合コンポーネント間の双方向スクリプト・データの変換をサポートします。

グローバル化および双方向データ変換

このアダプターは、1 バイト文字セットとマルチバイト文字セットをサポートし、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グローバル化されています。アダプターは双方向のスクリプト・データ変換も実行します。双方向変換とは、1 つのファイルに右から左 (ヘブライ語やアラビア語など) と左から右 (URL やファイル・パスなど) の両方の意味内容を含むデータを処理するタスクのことを指します。

グローバル化

グローバル化されたソフトウェア・アプリケーションは、単一の環境ではなく複数の言語環境や国/地域別環境で使用することを目的として設計され、開発されています。WebSphere Adapters、WebSphere Integration Developer、WebSphere Process Server、および WebSphere Enterprise Service Bus は、Java で作成されています。Java 仮想マシン (JVM) 内の Java ランタイム環境は、Unicode 文字コード・セットでデータを表現します。Unicode には、ほとんどの既知の文字コード・セット (1 バイト系とマルチバイト系を含む) の文字に対応できるエンコード方式が組み込まれています。そのため、これらの統合システム・コンポーネント間でデータを転送するときに文字を変換する必要はありません。

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。

双方向スクリプト・データ変換

アラビア語やヘブライ語などの言語は右から左に書きますが、テキストには左から右に書かれる部分も埋め込まれるため、双方向スクリプトになります。ソフトウェア・アプリケーションで双方向スクリプト・データを扱う場合は、その表示と処理のためにさまざまな規格を使用します。双方向スクリプト・データ変換の適用対象は、ストリング・タイプのデータのみです。WebSphere Process Server または WebSphere Enterprise Service Bus は Windows® 標準形式を使用しますが、サーバーとの間でデータをやり取りするアプリケーションやファイル・システムは別の形式を使用する可能性があります。アダプターでは、トランザクションが行われる 2 つのシステムでデータが正確に処理および表示されるように、これらのシステム間でやり取りされる双方向スクリプト・データが変換されます。スクリプト・データの変換は、スクリプト・データの形式を定義する 1 組のプロパティーと、変換の適用先となるコンテンツまたはメタデータを指定するプロパティーを使用することによって行われます。

双方向スクリプト・データ形式

WebSphere Process Server または WebSphere Enterprise Service Bus は、ILYNN (暗黙、左から右、オン、オフ、公称) の双方向形式を使用します。これらの 5 属性は、Windows で使用される形式で構成されます。サーバーとの間でデータの送信または受信を行うアプリケーションまたはファイル・システムが別の形式を使用した場合、アダプターは形式を変換してからデータをサーバーに投入します。変換を実行するには、外部サービス・ウィザードを使用して、送信側のアプリケーションまたはファイル・システムが使用する双方向形式を表す属性値を設定します。アダプターを初めてデプロイするときに、これを実行します。

双方向形式は、5 つの属性で構成されます。双方向プロパティーを設定する場合、これらの各属性に値を割り当てます。属性と設定値を次の表に示します。

表 70. 双方向形式の属性

文字の位置	目的	値	説明	デフォルト設定
1	スキーマの配列	I	暗黙 (論理的)	I
		V	表示	

表 70. 双方向形式の属性 (続き)

文字の位置	目的	値	説明	デフォルト設定
2	方向	L	左から右	L
		R	右から左	
		C	コンテキスト上の左から右	
		D	コンテキスト上の右から左	
3	対称スワッピング	Y	対称スワッピングのオン	Y
		N	対称スワッピングのオフ	
4	テキストの形状指定	S	テキストの形状を指定する	N
		N	テキストの形状を指定しない (名目)	
		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	数字の形状指定	H	各国 (ヒンディ語)	N
		C	コンテキストによる形状指定	
		N	数字の形状を指定しない (名目)	

変換するデータを指定する双方向プロパティ

変換の対象となるビジネス・データを指定するには、BiDiContextEIS プロパティを設定します。この操作を行うには、このプロパティの 5 つの双方向形式属性 (前掲の表に記載) のそれぞれに対して値を指定します。BiDiContextEIS プロパティは、Managed Connection Factory と活動化仕様に対して設定できます。

変換の対象となるアプリケーション固有のデータを指定するには、ビジネス・オブジェクト内部の BiDiContextEIS プロパティおよび BiDiMetadata プロパティに注釈を付けます。この操作を行うには、WebSphere Integration Developer 内部のビジネス・オブジェクト・エディターを使用して、ビジネス・オブジェクトのアプリケーション固有の要素としてプロパティを追加します。

双方向データ変換で使用可能なプロパティ

双方向データ変換プロパティでは、アプリケーションと、統合ツールおよびランタイム環境の間でやり取りされる双方向スクリプト・データに対して正しい形式を強制的に設定します。これらのプロパティが設定されると、双方向スクリプト・データは、WebSphere Integration Developer および WebSphere Process Server または WebSphere Enterprise Service Bus で正しく処理および表示されます。

管理接続プロパティ

以下の管理接続プロパティは、双方向スクリプト・データ変換を制御します。

- ユーザー名
- パスワード

- 環境
- ロール

活動化仕様プロパティ

以下の活動化仕様プロパティは、双方向スクリプト・データ変換のために有効化されます。

- 自動確認
- イベント送達の保証
- 待機なし
- 待機時間

関連情報

以下の、インフォメーション・センター、IBM Redbooks および Web ページには、WebSphere Adapter for JD Edwards EnterpriseOne の関連情報が含まれています。

サンプルおよびチュートリアル

ユーザーが、WebSphere Adapters を円滑に使用できるようにするため、「Business Process Management Samples and Tutorials」Web サイトからサンプルおよびチュートリアルが利用できます。サンプルおよびチュートリアルには、以下のいずれかの方法でアクセスできます。

- WebSphere Integration Developer を始動すると表示される「ようこそ」ページ。WebSphere Adapter for JD Edwards EnterpriseOne のサンプルおよびチュートリアルを表示するには、「取得」をクリックします。表示されたカテゴリーをブラウズして、選択を行います。
- Web 上の <http://publib.boulder.ibm.com/bpcsamp/index.html> のページ。

情報リソース

- WebSphere Business Process Management の情報リソース Web ページ (<http://www14.software.ibm.com/webapp/wsbroker/redirect?version=pix&product=wps-dist&topic=bpmroadmaps>) には、記事、Redbooks、資料、および研修用資料へのリンクが組み込まれており、WebSphere Adapters を習得するのに役立ちます。
- WebSphere Adapters ライブラリーのページ (<http://www.ibm.com/software/integration/wbiadapters/library/infocenter/>) には、資料の全バージョンへのリンクが組み込まれています。

関連製品の情報

- WebSphere Business Process Management バージョン 6.2 インフォメーション・センター (<http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r2mx/index.jsp>)。ここでは、WebSphere Process Server WebSphere Enterprise Service Bus および WebSphere Integration Developer の情報が記載されています。
- WebSphere Adapters バージョン 6.1.x インフォメーション・センター: <http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/topic/com.ibm.wbit.612.help.adapter.emd.ui.doc/topics/tcreatecmps.html>

- WebSphere Business Integration Adapters インフォメーション・センター:
http://publib.boulder.ibm.com/infocenter/wbihelp/v6rxmx/index.jsp?topic=/com.ibm.wbi_adapters.doc/welcome_adapters.htm

developerWorks® リソース

- WebSphere Adapter Toolkit
- WebSphere business integration zone

サポートおよび支援

- WebSphere Adapters テクニカル・サポート: <http://www.ibm.com/software/integration/wbiadapters/support/>
- WebSphere Adapters テクニカル・ノート: <http://www.ibm.com/support/search.wss?tc=SSMKUK&rs=695&rank=8&dc=DB520+D800+D900+DA900+DA800+DB560&dtm>。「**Product category**」リストで、アダプターの名前を選択して、「**Go**」をクリックします。

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。

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
Department 2Z4A/SOM1
294 Route 100
Somers, NY 10589-0100
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、さまざま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した創作物にも、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 LOGO、および ibm.com は、International Business Machines Corporation の米国およびその他の国における商標です。これらおよび他の IBM 商標に、この情報の最初に現れる個所で商標表示 (® または ™) が付されている場合、これらの表示は、この情報が公開された時点で、米国において、IBM が所有する登録商標またはコモン・ロー上の商標であることを示しています。このような商標は、その他の国においても登録商標またはコモン・ロー上の商標である可能性があります。現時点での IBM の詳しい商標リストについては、<http://www.ibm.com/legal/copytrade.shtml> をご覧ください。

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

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。

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。

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

他の会社名、製品名およびサービス名等はそれぞれ各社の商標です。

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。

索引

日本語、数字、英字、特殊文字の順に配列されています。なお、濁音と半濁音は清音と同等に扱われています。

[ア行]

アダプターのパフォーマンス 107
アダプター用パッケージ・ファイル 115
アダプター・アプリケーション
開始 106
停止 106
アダプター・アプリケーションの開始 106
アダプター・アプリケーションの停止 106
アダプター・テクニカル・ノート 163
アプリケーション固有情報
操作レベル 125
ビジネス・オブジェクト・レベル 123
プロパティ・レベル 124
イベント・ストア
概要 5

[カ行]

外部依存関係、追加 37, 81, 83, 87
外部依存関係、編集 41
外部サービス
プロパティ、接続 128, 143
外部サービス・ウィザード
開始 34
概要 11
接続プロパティの設定 42
認証 16
カスタム・プロパティ
活動化仕様 99, 104
リソース・アダプター 95, 101
Managed Connection Factory 97, 102
活動化仕様プロパティ
外部サービス・ウィザードによる設定 151
管理コンソールでの設定 99, 104
リスト 151
関連情報 162
関連製品、情報 162
技術情報 1
機密データ、隠蔽 15
機密データ、秘匿 15
機密トレース 15
組み込みアダプター
活動化仕様プロパティ、設定 99
使用する場合の考慮事項 19
説明 18

組み込みアダプター (続き)
リソース・アダプター・プロパティ、設定 95
Managed Connection Factory プロパティ、設定 97
組み込みデプロイメント 83
クラスター化された環境
説明 20
デプロイ 20
Inbound 処理 21
Outbound 処理 21
研修、WebSphere Adapters 162
高可用性環境
説明 20
デプロイ 20
Inbound 処理 21
Outbound 処理 21
構成
トレース 114
ロギング 114
Performance Monitoring Infrastructure (PMI) 107
構成の概要 32
後方互換性
プロジェクト 26
プロジェクト交換ファイル 26
互換性マトリックス 1
コンポーネントの接続 84

[サ行]

再試行制限プロパティ 157
サポート
概要 113
セルフ・ヘルプ・リソース 120
テクニカル 163
サポート、Log and Trace Analyzer 113
サポートされる操作 125
サンプル 29
実行時環境
認証 17
EAR ファイルのデプロイ先 87
実装環境、Java 85
スタンドアロン・アダプター
活動化仕様プロパティ、設定 104
使用する場合の考慮事項 20
説明 18
リソース・アダプター・プロパティ、設定 101
Managed Connection Factory プロパティ、設定 102
セキュリティ
機密データの秘匿 15
セキュリティ、Java 2 17
セキュリティ機能、アダプター 15
接続プロパティ、外部サービス・ウィザード 42

接続プロパティの設定 42
セルフ・ヘルプ・リソース 120
ソフトウェア依存関係、外部の追加 37, 81, 83, 87
ソフトウェア依存関係、外部の編集 41
ソフトウェア要件 1

[タ行]

ターゲット・コンポーネント 84
対話仕様プロパティ 141
 変更 79
チュートリアル 29
テクニカル・サポート 163
テクニカル・ノート 120, 163
テクニカル・ノート、WebSphere Adapters 162
テスト環境
 デプロイ先 81, 86
 モジュールの追加先 86
 モジュールのテスト 87
デバッグ
 セルフ・ヘルプ・リソース 120
 XAResourceNotAvailableException 例外 119
デプロイメント
 オプション 18
 環境 81
 実稼働環境への 87
 テスト環境への 81
トラブルシューティング
 概要 113
 セルフ・ヘルプ・リソース 120
 XAResourceNotAvailableException 例外 119
トレース
 管理コンソールを使用したプロパティの構成 114
トレース・ファイル
 使用可能化 114
 詳細レベル 114
 使用不可化 114
 場所 116
 ファイル名の変更 116

[ナ行]

認証
 外部サービス・ウィザード 16
 実行時 17
 説明 16
認証別名 17, 33

[ハ行]

ハードウェア要件 1
ハードウェア要件とソフトウェア要件 1
パフォーマンスに関する統計 110
パフォーマンスのモニター 107
ビジネス・オブジェクト 6, 123

ビジネス・オブジェクト (続き)
 アプリケーション固有情報 123
 外部サービス・ウィザードを使用した XML List の生成 57
 外部サービス・ウィザードを使用したビジネス関数の生成 44
 サポートされる操作 125
 命名規則 126
ビジネス・オブジェクトの命名規則 126
ビジネス・フォールト 117
ファイル
 SystemOut.log ログ・ファイル 116
 trace.log トレース・ファイル 116
フォールト
 説明 117
プロジェクト、作成 34
プロジェクト交換 (PI) ファイル
 マイグレーションなしでの更新 26
プロパティ
 外部サービス接続 128, 143
 活動化仕様 99, 104
 構成プロパティ
 Inbound 142
 Outbound 127
 リソース・アダプター 95, 101
 リスト 132, 147
 Inbound 構成 142
 Managed (J2C) Connection Factory 97, 102
 外部サービス時の設定 53, 63, 73
 リスト 136
 Outbound 構成 127
別名、認証 33

[マ行]

マイグレーションに関する考慮事項 22
マトリックス、互換性 1
メタデータ 123
 操作レベル 125
 ビジネス・オブジェクト・レベル 123
 プロパティ・レベル 124
モジュールの構成のためのロードマップ 31
問題判別
 セルフ・ヘルプ・リソース 120
 XAResourceNotAvailableException 例外 119

[ヤ行]

要件、ハードウェアおよびソフトウェア 1

[ラ行]

リソース・アダプター・アーカイブ (RAR) ファイル
 サーバーへのインストール 89
 説明 89

リソース・アダプター・プロパティ
管理コンソールでの設定 95, 101
リスト 132, 147
例外
XAResourceNotAvailableException 119
ロギング
管理コンソールを使用したプロパティの構成 114
ログ・アナライザー 114
ログ・ファイル
使用可能化 114
詳細レベル 114
使用不可化 114
場所 116
ファイル名の変更 116
ログ・ファイルとトレース・ファイル 113

A

Adapter for JD Edwards EnterpriseOne
管理 95
セキュリティ 15
Adapter for JD Edwards EnterpriseOne モジュール
開始 106
停止 106
EAR ファイルとしてのエクスポート 91
EAR ファイルのサーバーへのインストール 92
AssuredOnceDelivery プロパティ 151
AutoAcknowledge プロパティ 151

C

CEI (Common Event Infrastructure) 111
Common Event Infrastructure (CEI) 111

D

DeliveryType プロパティ 151
developerWorks 163
developerWorks リソース、WebSphere Adapters 162

E

EAR ファイル
エクスポート 91
サーバーへのインストール 92
EAR ファイルとしてのモジュールのエクスポート 91
EAR ファイルのインストール 92
enableHASupport プロパティ 21
EventTypeFilter プロパティ 151

F

FFDC (First Failure Data Capture) 117
First Failure Data Capture (FFDC) 117

I

IBM WebSphere Adapter Toolkit 163
Inbound 構成プロパティ 142
Inbound 処理
概要 3
ビジネス・グラフ 3

J

JAR ファイル、外部の追加 37, 81, 83
Java 2 セキュリティ 17
Java 実装環境 85

M

Managed (J2C) Connection Factory プロパティ
外部サービス時の設定 53, 63, 73
管理コンソールでの設定 97, 102
リスト 136
MaximumConnections プロパティ 151
MinimumConnections プロパティ 151

N

NoWait プロパティ 151

O

Outbound 構成プロパティ 127
Outbound 処理 3
概要 2

P

Performance Monitoring Infrastructure (PMI)
構成 107
説明 107
パフォーマンスに関する統計の表示 110
PMI (Performance Monitoring Infrastructure)
構成 107
説明 107
パフォーマンスに関する統計の表示 110
PollPeriod プロパティ 151
PollQuantity プロパティ 151

R

RAR (リソース・アダプター・アーカイブ) ファイル
サーバーへのインストール 89
説明 89
Redbooks、WebSphere Adapters 162
RetryInterval プロパティ 151
RetryLimit プロパティ 151

S

StopPollingOnError プロパティ 151
SystemOut.log ファイル 116

T

trace.log ファイル 116

W

WaitTime プロパティ 151
WebSphere Adapters バージョン 6.0 情報 162
WebSphere Adapters バージョン 6.0.2.x 情報 162
WebSphere Application Server 情報 162
WebSphere Business Integration Adapters 情報 162
WebSphere Business Process Management バージョン 6.1.x 情報
162
WebSphere Enterprise Service Bus
情報 162
WebSphere Extended Deployment 21
WebSphere Integration Developer
開始 34
情報 162
テスト環境 81
WebSphere Process Server
情報 162
WebSphere Process Server または WebSphere Enterprise Service
Bus
デプロイ先 87

X

XAResourceNotAvailableException 119
XML List
外部サービス・ウィザードを使用した生成 57



Printed in Japan