





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、107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 Oracle E-Business Suite バージョン 6、リリース 1、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。

IBM 発行のマニュアルに関する情報のページ

<http://www.ibm.com/jp/manuals/>

こちらから、日本語版および英語版のオンライン・ライブラリーをご利用いただけます。また、マニュアルに関するご意見やご感想を、上記ページよりお送りください。今後の参考にさせていただきます。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： WebSphere® Adapters
Version 6 Release 1
WebSphere Adapter for Oracle E-Business Suite User Guide
Version 6 Release 1

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

目次

第 1 章 WebSphere Adapter for Oracle

E-Business Suite の概要 1

このリリースの新機能	1
ハードウェアおよびソフトウェア要件	1
Adapter for Oracle E-Business Suite の技術的な概説	2
アダプターのアーキテクチャー	2
Outbound 処理のサンプル・アプリケーション	5
Inbound 処理のサンプル・アプリケーション	8
標準の準拠	9
アクセシビリティ	9
インターネット・プロトコル・バージョン 6 (IPv6)	10

第 2 章 アダプター実装の計画 11

始める前に	11
デプロイメント・オプション	11
クラスター環境での WebSphere Adapters	14

第 3 章 サンプル 17

データベースおよびアプリケーションのユーザー・アカウント要件	17
サンプル・ファイルへのアクセス	18
サンプル 1: Outbound データの Oracle データベースへの送信	18
Outbound 処理の準備	19
プロジェクトの作成	20
Outbound 処理のモジュールの構成	20
テスト用のモジュールのデプロイ	33
Create 操作のテスト	34
Retrieve 操作のテスト	41
サンプル・コンテンツのクリア	42
サンプル 2: データベースからの Inbound データの受信	43
Inbound 処理の準備	43
Oracle E-Business Suite 内部でのセットアップ	45
プロジェクトの作成	48
Inbound 処理のモジュールの構成	49

Inbound コンポーネントの作成	59
テスト用のモジュールのデプロイ	63
アセンブル済みアダプター・アプリケーションのテスト	63
サンプル・コンテンツのクリア	66
サンプル 3: Outbound 処理を行うための Oracle E-Business Suite API の呼び出し	66
プロジェクトの作成	67
API 呼び出しの Outbound 処理を行うためのモジュールの構成	67
テスト用のモジュールのデプロイ	74
アセンブル済みアダプター・アプリケーションのテスト	75
サンプル・コンテンツのクリア	78
サンプル 4: Business Event System およびワークフローを使用したビジネス・オブジェクトのインポート	78
Outbound 処理の準備	79
Oracle E-Business Suite 内部でのセットアップ	80
プロジェクトの作成	84
Outbound 処理のモジュールの構成	84
テスト用のモジュールのデプロイ	94
Create 操作のテスト	95
サンプル・コンテンツのクリア	101
サンプル・アプリケーションのトラブルシューティング情報	101
サンプル・アダプター成果物の表示	102

第 4 章 参照情報 105

関連情報	105
----------------	-----

特記事項 107

プログラミング・インターフェース情報	109
商標	109

索引 111

第 1 章 WebSphere Adapter for Oracle E-Business Suite の概要

WebSphere Adapter for Oracle E-Business Suite は、既存の WebSphere コンポーネントを使用して、Oracle E-Business Suite と WebSphere Business Integration システム間の双方向通信を実現するソリューションです。このソリューションでは、WebSphere Adapter for JDBC を使用して Oracle E-Business Suite のデータベース・コンポーネントと対話し、外部ソースとの間で交換されるデータを処理します。

WebSphere Adapter for Oracle E-Business Suite ソリューションは、従来のリソース・アダプターとは異なります。このソリューションでは、Adapter for JDBC を使用して Oracle E-Business Suite データベースと通信し、さらにサンプル・アプリケーションを使用することにより、この両者間の対話をセットアップする方法を示します。操作手順説明では、Inbound イベント処理と Outbound 要求処理の両方について、Oracle E-Business Suite と WebSphere Process Server または WebSphere Enterprise Service Bus との間の情報の流れを有効にする方法を説明します。

このソリューションを構成するサンプル・コンテンツおよび成果物は、Oracle E-Business Suite との統合を有効にする過程を迅速化する上で役立ちます。一部のサンプル・アプリケーションには、Oracle Customer および Oracle API を備えた Receivables モジュールを使用して、データベース統合のガイドラインを示す機能があります。サンプルで習得した内容は、選択したその他の Oracle E-Business Suite モジュール (General Ledger や Payables など) に適用できます。

このリリースの新機能

WebSphere Adapter for Oracle E-Business Suite バージョン 6.1.0 は、アダプターに対して拡張機能を提供します。

以下の新機能または拡張機能を実現しています。

- Oracle E-Business Suite バージョン 12 をサポートします。
- サンプル・アプリケーションは、Adapter for JDBC バージョン 6.1.0 に対する更新内容に基づき、必要に応じて変更されています。この変更には、ビジネス・オブジェクト間に親子関係を作成する外部サービス・ウィザードの機能が含まれます。この処理は、以前は WebSphere Integration Developer のビジネス・オブジェクト・エディターを使用して実行していました。

この情報についての更新は、WebSphere Adapters 製品のサポート Web サイトで参照できます。更新情報や追加情報を確認するには、<http://www.ibm.com/software/integration/wbiadapters/support/>を参照してください。

ハードウェアおよびソフトウェア要件

WebSphere Adapters のハードウェアおよびソフトウェア要件は、以下のロケーションにある IBM® Web サイトに記載されています。

WebSphere Adapters のハードウェアおよびソフトウェア要件: <http://www.ibm.com/support/docview.wss?uid=swg27006249>

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に必要な場合がある追加情報が記載されています。

- WebSphere Business Integration Adapters と WebSphere Adapters の互換性一覧表には、使用するアダプターに必要なソフトウェアのサポートされるバージョンが記載されています。この資料を参照するには、WebSphere Adapters のサポート・ページにアクセスし、**アップグレードの計画 (Planning upgrades)**: <http://www.ibm.com/software/integration/wbiadapters/support/> の下にある互換性一覧表のリンクをクリックします。
- WebSphere Adapters のテクニカル・ノートには、製品資料に記載されていない次善策および追加情報が収録されています。アダプターのテクニカル・ノートを参照するには、Web ページ <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、「**Product category**」 リストからアダプターの名前を選択し、検索アイコンをクリックします。

Adapter for Oracle E-Business Suite の技術的な概説

WebSphere Adapter for Oracle E-Business Suite を使用すると、Outbound 要求と Inbound イベント処理の両方について、WebSphere システムと Oracle E-Business Suite 間の接続が使用可能になります。この製品では、データを処理するために、WebSphere Adapter for JDBC と Oracle E-Business Suite の複数のデータベース機構を組み合わせて使用します。アダプターは 4 つのサンプル・アプリケーションで構成されており、そのうち 3 つは Outbound 要求の処理用であり、残りの 1 つは Inbound イベント用です。

アダプターのアーキテクチャー

Oracle E-Business Suite のイベント処理機構は、Oracle データベースからの Inbound イベントを処理するために、WebSphere Adapter for JDBC と組み合わせて使用します。Adapter for JDBC は、Outbound 要求を処理し、Oracle データベース内のデータの作成、更新、または検索を行います。

概説

Oracle E-Business Suite は、Oracle データベースを操作する目的で設計されています。Oracle E-Business Suite には、独自のイベント処理機構である Business Event System (BES) があります。イベントは、例えばデータベース内でのカスタマー・レコードの作成や更新である場合がありますが、これは、ビジネス・インテグレーション・システムにインターネットやイントラネットのアプリケーションまたはプログラムが存在することに起因します。Business Event System を使用すると、Oracle データベース内でイベントを生成し、そのイベントをコンシュームして、事前設定の構成に基づいてアクションを実行するサブスクリプションを使用できます。イベントを生成すると、Oracle イベント・マネージャーに対するイベントを確認できるため、そのイベントに関連したサブスクリプションを開始できます。サブ

スクリプションが決まると、実行するアクションの内容も決まります。アクションには、イベントからの情報をテーブルまたは Oracle AQ キューに取り込むことや、何らかのカスタム SQL コードまたは Java™ コードを実行することなどがあります。

イベントが生成されると、Business Event System は、そのイベント・タイプを使用するサブスクリプションを検索し、それらのサブスクリプションに定義されているアクションを処理します。Oracle ユーザーは、実行できるアクションを柔軟に定義できます。例えば、Oracle データベース内にある Customer ビジネス・オブジェクトに対する更新内容に基づいて起動するようにイベントを設計することができます。変更されたオブジェクトのキー値をイベント・テーブルに取り込んで、別の Service Component Architecture (SCA) コンポーネントがコンシュームするアクションにすることもできます。

WebSphere Adapter for JDBC は、データベース成果物とのインターフェースを処理する目的で使用します。これを使用すると、SQL テーブルを直接呼び出して、データベースのストアード・プロシージャおよび API と対話できます。このソリューションでは、Oracle E-Business Suite に固有の組み込みイベント処理サポート機能を使用することにより、Adapter for JDBC がコンシュームできるイベントを処理します。このソリューションでは、Adapter for JDBC を使用して、アプリケーションまたはコンポーネントから Oracle データベースへ情報を取り込むことや、ストアード・プロシージャを呼び出してデータを処理し、そのデータを Oracle システムに送ることもできます。

Inbound アーキテクチャー

Inbound イベント処理のサンプル・アプリケーションであるサンプル 2 では、Adapter for JDBC を使用して、Oracle アプリケーションのデータベースに作成したイベント・テーブルのポーリングを実行します。アダプターは、オブジェクト・キーやオブジェクト・タイプなど、イベント・テーブルからのイベント情報を使用して、Oracle 基本テーブルからビジネス・オブジェクト情報を取り出します。Adapter for JDBC のアーキテクチャーについては、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」を参照してください。

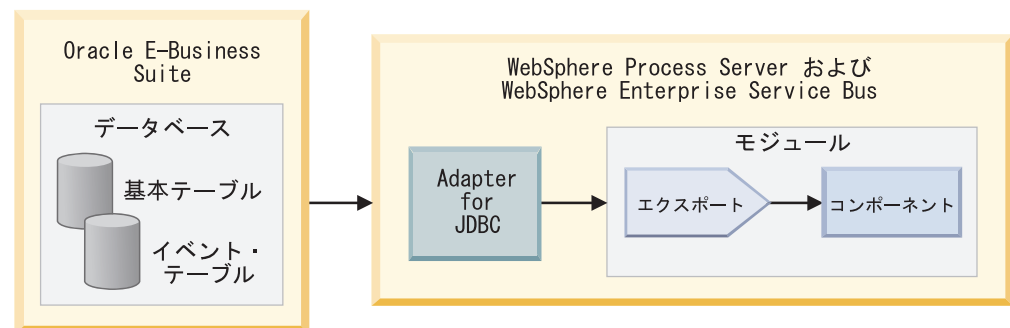


図 1. Inbound イベント処理

Oracle の内部 Business Event System を使用すると、Oracle プロセスは、ビジネス・オブジェクトがいつ変更されたかを認識して、その情報を JDBC イベント・ストアに取り込むことができます。Business Event System の特定の使用方法について詳しくは、サンプル・アプリケーションを参照してください。Business Event

System について詳しくは、Oracle Applications の資料、特に「*Workflow Administrator's Guide*」を参照してください。

Outbound アーキテクチャー

Outbound 要求の処理の場合は、サンプル 1 の場合と同様に、Adapter for JDBC を使用して、オブジェクトの情報を Oracle アプリケーションのインターフェース・テーブルに取り込みます。インターフェース・テーブルは、Outbound 情報を処理して Oracle 基本テーブルに格納できるようにするための標準のテーブルです。Oracle では、この処理をサポートすることにより、基本テーブルを直接更新することによって生じる、望ましくない内容変更を防止します。

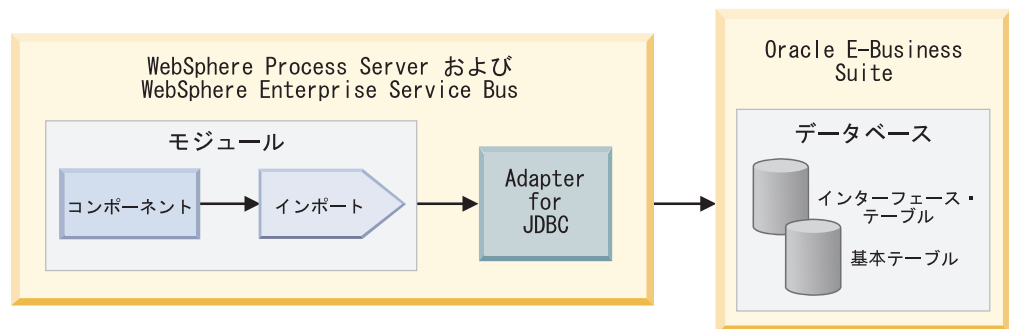


図2. Outbound 要求の処理

Adapter for JDBC は、インターフェース・テーブルにデータを取り込みます。その後は、Oracle アプリケーションに組み込まれている標準の関数呼び出しを起動するストアード・プロシージャーを呼び出す場合に使用されます。Oracle データベースは、インターフェース・テーブル内にあるデータを処理し、そのデータを基本テーブルに取り込みます。

Oracle アプリケーションからデータを取り出すには、Adapter for JDBC を使用する Retrieve 操作を基本テーブルに対して実行します。これは、テーブルのデータが変更されることがないためです。

さらに、サンプル 3 の場合と同様に、Adapter for JDBC を使用すると、Oracle の組み込みデータベース・アプリケーション・プログラミング・インターフェース (API) を直接呼び出すことができます。ただし、Adapter for JDBC から API を直接呼び出す場合は、JDBC ドライバーにより、パラメーターは単純なデータ型 (レコード以外のデータ型) に制限されます。大多数の Oracle API ではレコード・パラメーターを使用するため、これらの API はストアード・プロシージャーの内側にラップし、後で Adapter for JDBC によって呼び出すことができます。

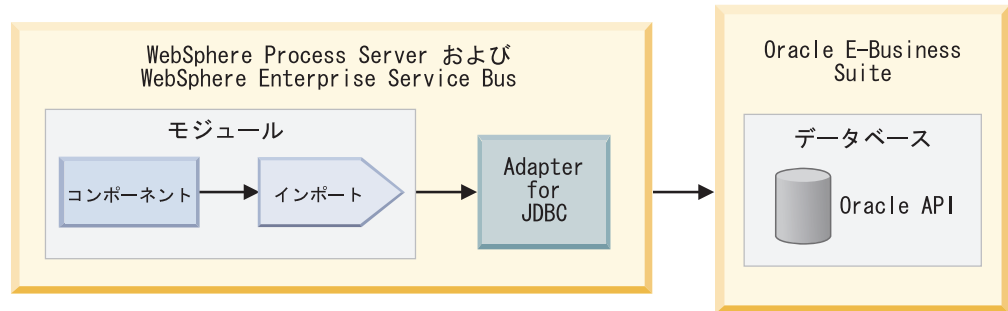


図3. API 呼び出しを使った Outbound 処理

サンプル 4 では、Business Event System およびワークフローを使用してビジネス・オブジェクトをインポートします。Adapter for JDBC は、Outbound 要求を Oracle E-Business Suite に送信して、Oracle のカスタム・イベント・テーブルにカスタマー・データを作成します。Oracle Business Event System は、イベントからワークフローを起動します。ワークフローにより、Oracle E-Business Suite にカスタマー・レコードが作成されます。処理中のエラーもワークフローによって処理されます。

Outbound 処理のサンプル・アプリケーション

このソリューションには、Outbound 処理のサンプル・アプリケーションがいくつか用意されています。サンプル 1 では Oracle データベースのインターフェース・テーブルを使用しますが、これは Oracle E-Business Suite の標準 Outbound シナリオになっています。サンプル 3 では API 呼び出しを使用し、サンプル 4 ではワークフローを使用します。

インターフェース・テーブルを使用した Outbound 処理

最初の Outbound 要求処理サンプルであるサンプル 1 は、以下の 2 つの部分から構成されます。

- Create 操作の使用によるビジネス・オブジェクトの作成
- Retrieve 操作の使用によるオブジェクトの検索

Oracle データベースを使用すると、アプリケーションの基本テーブルからデータを検索できます。Retrieve 操作に使用されるビジネス・オブジェクトは、データの基本テーブルの表現を反映しています。Oracle データベースでは、基本テーブルのデータを手動で変更することはできません。このため、このサンプルでは、Oracle データベースの内容を変更するビジネス・オブジェクトを Adapter for JDBC が送信すると、インターフェース・テーブルが使用されます。これが、標準的な Outbound 要求処理のシナリオです。ビジネス・オブジェクト・データを使用してインターフェース・テーブルを更新すると、アダプターにより、ストアード・プロシージャを介して関数が呼び出され、この関数によってインターフェースから基本テーブルへデータが移動します。

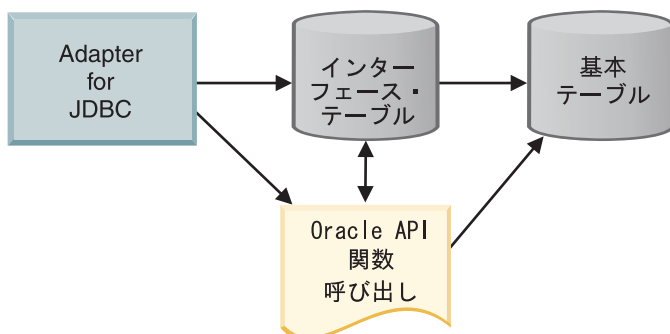


図4. Create のシナリオ

Adapter for JDBC は、インターフェース・テーブルにビジネス・オブジェクト・データを取り込む目的で使用します。Oracle 基本テーブルに対して更新を直接行うことはできないため、Update 操作は使用できません。Create 操作は、新規データと変更済みデータの両方を Oracle インターフェース・テーブルを介して処理する場合に使用します。ビジネス・オブジェクト・データを Oracle 基本テーブルに移動するには、内部の標準 Oracle 関数を呼び出します。この関数は、ストアード・プロシージャを介して呼び出され、ビジネス・オブジェクト上にある AfterCreateSP というアプリケーション固有の情報を使用して、データを基本テーブルに移動します。

Create 操作および Retrieve 操作は、Oracle データベース内で異なるテーブルを表現するため、使用する一連のビジネス・オブジェクトが異なります。このサンプル・アプリケーションには、関数呼び出しによってデータを基本テーブルに移動するためのラッパー・ストアード・プロシージャと、さらに、このストアード・プロシージャで使用するためのサンプル・コンテンツが組み込まれています。

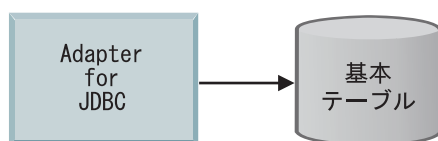


図5. Retrieve のシナリオ

API 呼び出しを使った Outbound 処理

サンプル 3 では、Adapter for JDBC が Oracle E-Business Suite API を直接呼び出すことができる仕組みを示しています。Adapter for JDBC が API を直接呼び出す場合は、JDBC ドライバーにより、パラメーターが単純なデータ型 (レコード以外のデータ型) に制限されます。大半の Oracle API には、レコード・パラメーターが組み込まれています。レコード・タイプのパラメーターを使用する API 呼び出しを Adapter for JDBC を介して呼び出すことができるのは、その API 呼び出しがラップされている場合のみであるため、ラッパー・ストアード・プロシージャの呼び出しでは、レコード・タイプのパラメーターを使用しません。このサンプルの API 呼び出しでは、単純なデータ型のパラメーターを使用します。

このサンプルでは、Adapter for JDBC を使用して Oracle API を直接呼び出し、この Oracle API により、並行プログラムの実行可能ファイルが Oracle データベース

内に作成されます。このサンプルのコンテンツには、API 呼び出しのサンプルが組み込まれています。

Business Event System およびワークフローを使用した Outbound 処理

サンプル 4 は、Business Event System でのイベントに基づいてデータベースにカスタマー・レコードを作成するためのワークフローを使用します。Adapter for JDBC は、Create 操作を使用してカスタマー・データを Oracle カスタム・イベント・テーブルに挿入します。アダプターは、ストアード・プロシージャを呼び出して Business Event System にカスタム・イベントを作成します。Business Event System には、このイベントに対する組み込みのサブスクリプションがあります。Business Event System は、Oracle E-Business Suite にカスタマー・レコードを作成するワークフローを作成します。

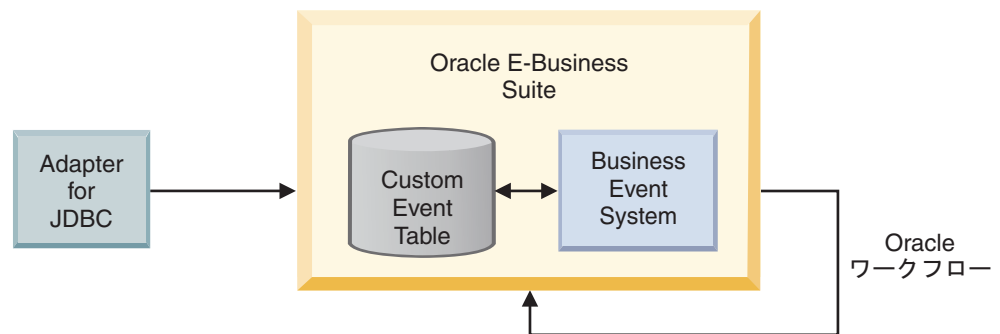


図 6. ワークフローを使用した Outbound 処理

関連タスク

18 ページの『サンプル 1: Outbound データの Oracle データベースへの送信』
WebSphere Adapter for Oracle E-Business Suite を使用すると、データを Oracle アプリケーションに作成して、Oracle アプリケーションからデータを取り出すことができます。このシナリオでは、WebSphere Adapter for JDBC を統合の方法として使用し、Oracle E-Business Suite の Financials データベースにカスタマー・レコードを作成して、そのカスタマー情報を Financials データベースから取り出します。

66 ページの『サンプル 3: Outbound 処理を行うための Oracle E-Business Suite API の呼び出し』

アダプターは、単純なデータ・パラメーターを使用する任意の Oracle E-Business Suite API を直接呼び出すことができます。Outbound 処理を行うために、並行プログラム実行可能ファイルを作成する API を呼び出すには、このサンプルの手順に従います。このシナリオを実行することにより、アダプターと Oracle E-Business Suite データベース間の単純なデータ API 呼び出しを作成する方法が分かります。

78 ページの『サンプル 4: Business Event System およびワークフローを使用したビジネス・オブジェクトのインポート』

Adapter for JDBC は、Outbound 要求を Oracle E-Business Suite に送信して、Oracle のカスタム・イベント・テーブルにカスタマー・データを作成します。そのイベントにより、Oracle Business Event System は、Oracle E-Business Suite

にカスタマー・レコードを作成するワークフローを起動します。このワークフローは、この処理中に発生するエラーの処理も行います。

Inbound 処理のサンプル・アプリケーション

サンプル 2 では、Oracle Business Event System を使用して、Oracle E-Business Suite からイベント・テーブルへ Inbound カスタマー・データを送信します。このイベント・テーブルでは、Adapter for JDBC によってデータを読み取ることができます。Adapter for JDBC はイベントをイベント・テーブルから取り出し、イベントで提供される情報に基づいて Oracle データベースからビジネス・オブジェクトを取得します。次に、Adapter for JDBC は、Oracle データベースから取得したビジネス・オブジェクトを処理します。このサンプルに組み込まれているコンテンツについては、このセクションで説明します。

このサンプルでは、Oracle E-Business Suite からの Inbound カスタマー・データの処理について示します。Oracle 組み込みの Business Event System により、イベントが生成され、必要な値 (新規および更新されたカスタマー・データ) が Adapter for JDBC イベント・テーブルに取り込まれます。カスタムのイベント・テーブルが Oracle 内部に作成され、Business Event System により使用されます。

カスタムの並行プログラムにより、Customer テーブル (Oracle 基本テーブル) に対する更新の有無が検索され、新規または変更されたカスタマー・レコードが検出されると、イベントが生成されます。並行プログラムの構成方法には、設定した間隔で実行する方法、手動で呼び出す方法、またはストアード・プロシージャを介して呼び出す方法があります。このサンプルの場合、並行プログラムは設定した間隔で実行するよう構成されています。サンプルのステップが実行される時間を節約するには、並行プログラムを手動で呼び出します。データがイベント・テーブルに到達すると、Adapter for JDBC は、object_key、object_name、および object_function の有無についてイベント・テーブルをポーリングします。その後、Oracle データベース・アプリケーションからビジネス・オブジェクトを処理します。

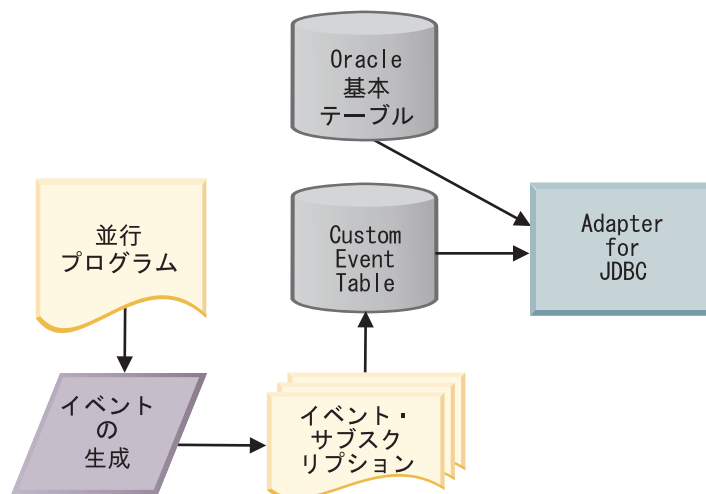


図 7. Inbound イベント処理

Business Event System の場合、イベントは Customer テーブルの更新情報に対して定義されます。変更内容を保存すると、イベントが生成され、そのイベント・タイ

プを調べるためにサブスクリプションが Oracle Event Manager によって検索されます。サンプルは、カスタマー・データの変更を認識するカスタム・サブスクリプションを使用します。イベントはこのサブスクリプションに渡され、ここではサブスクリプションに定義されているアクションに基づいてデータが処理されます。イベント情報は、Adapter for JDBC のイベント・テーブルに取り込まれます。アダプターは、そのテーブルを照会してイベント情報を使用し、ビジネス・オブジェクトの基本テーブルから Retrieve 操作を実行します。

このサンプルのコンテンツには、アクションおよびイベント・テーブルをセットアップするのに必要なストアド・プロシージャが組み込まれています。

注: Customer テーブルで、トリガーを使用してイベント・テーブルでのデータの取り込みを処理することは可能ですが、この方法は Oracle からは推奨されていません。

関連タスク

43 ページの『サンプル 2: データベースからの Inbound データの受信』

このサンプルでは、Oracle E-Business Suite 内部で Business Event System を使用して、イベント・テーブルにカスタマー・イベントを取り込みます。さらに、WebSphere Adapter for JDBC を使用してイベントのポーリングを実行し、カスタマー・データを取り出します。このシナリオを実行すると、Oracle E-Business Suite と WebSphere Process Server または WebSphere Enterprise Service Bus との間の情報の流れを Inbound イベントに対して確立する方法が分かります。

標準の準拠

この製品は、アクセシビリティ標準やインターネット・プロトコル標準といった、いくつかの行政標準および業界標準に準拠しています。

アクセシビリティ

IBM は、年齢や能力を問わず、すべての人が便利に使用できる製品の提供に努めています。WebSphere Adapters は、完全にアクセス可能で、米国リハビリテーション法第 508 条に準拠しています。アクセシビリティ機能を使用すると、運動障害または視覚障害など身体に障害を持つユーザーがソフトウェア・プロダクトを快適に操作できるようになります。これらの機能は、WebSphere Adapters のインストール機能や管理機能に組み込まれています。

管理

ランタイム管理コンソールは、エンタープライズ・アプリケーションのデプロイメントおよび管理用の基本インターフェースです。このコンソールは、標準の Web ブラウザー内に表示されます。Microsoft® Internet Explorer や Netscape Browser などのアクセス可能な Web ブラウザーを使用すると、次のことが可能になります。

- スクリーン・リーダー・ソフトウェアとデジタル・スピーチ・シンセサイザーを使用して、画面上に表示されている内容を聞く
- IBM ViaVoice® などの音声認識ソフトウェアを使用したデータの入力とユーザー・インターフェースへのナビゲート
- マウスの代わりにキーボードを使用して機能进行操作する

プロダクト機能は、提供されるグラフィカル・インターフェースではなく、標準テキスト・エディターおよびスクリプト・インターフェースまたはコマンド行インターフェースにより、構成および使用できます。

場合によっては、特定の製品機能についての文書に、その機能のアクセシビリティについての追加情報が記載されています。

外部サービス・ウィザード

外部サービス・ウィザードは、モジュールを作成するのに使用する主コンポーネントです。WebSphere Integration Developer を通して使用可能な Eclipse プラグインとして実装されるこのウィザードは、完全にアクセス可能です。

キーボード・ナビゲーション

この製品では、標準の Microsoft Windows® ナビゲーション・キーを使用します。

IBM とアクセシビリティ

IBM のアクセシビリティに対する取り組みについては、*IBM Accessibility Center* の Web サイト (<http://www.ibm.com/able/>) を参照してください。

インターネット・プロトコル・バージョン 6 (IPv6)

WebSphere Process Server および WebSphere Enterprise Service Bus は、Internet Protocol Version 6 (IPv6) の互換性について、WebSphere Application Server に依存しています。

IBM WebSphere Application Server バージョン 6.1.0 以上は、デュアル・スタックのインターネット・プロトコル・バージョン 6.0 (IPv6) をサポートしています。

この互換性の WebSphere Application Server での詳細については、<http://www.ibm.com/software/webservers/appserv/was/library/>で、IPv6 サポートを参照してください。

IPv6 について詳しくは、<http://www.ipv6.org> を参照してください。

第 2 章 アダプター実装の計画

WebSphere Adapter for Oracle E-Business Suite を使用する前に、作業者に必要な経験と、アダプターが稼働するサーバー環境について理解しておきます。アダプターをサーバー環境にデプロイする上での考慮事項を理解すると共に、クラスター・サーバー環境の使用によって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方法を検討します。

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、Oracle E-Business Suite 環境、WebSphere Adapter for JDBC、および WebSphere Integration Developer と WebSphere Process Server または WebSphere Enterprise Service Bus の機能と処理能力について徹底的に理解しておくことが必要です。

WebSphere Adapter for Oracle E-Business Suite を使用するには、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。
- 作業を行う Oracle E-Business Suite 環境。
- 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。
- 統合ソリューションへの使用を計画しているサーバーの機能および要件。ホスト・サーバーの構成と管理の方法、および管理コンソールを使用したプロパティ定義の設定と変更の方法、接続ファクトリーの構成方法、イベントの管理方法を理解しておく必要があります。
- WebSphere Integration Developer によって提供されるツールおよび機能。これらのツールを使用したモジュールの作成方法、コンポーネントの接続およびテスト方法、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。
- WebSphere Adapter for JDBC によって提供される機能。Oracle データベースとの通信に使用します。Adapter for JDBC に精通している必要があります。例えば、資料を読んで単純なデータベース・テーブルで使用してみてください。

デプロイメント・オプション

デプロイされている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、RAR ファイルを単体でデプロイするかを選択できます。

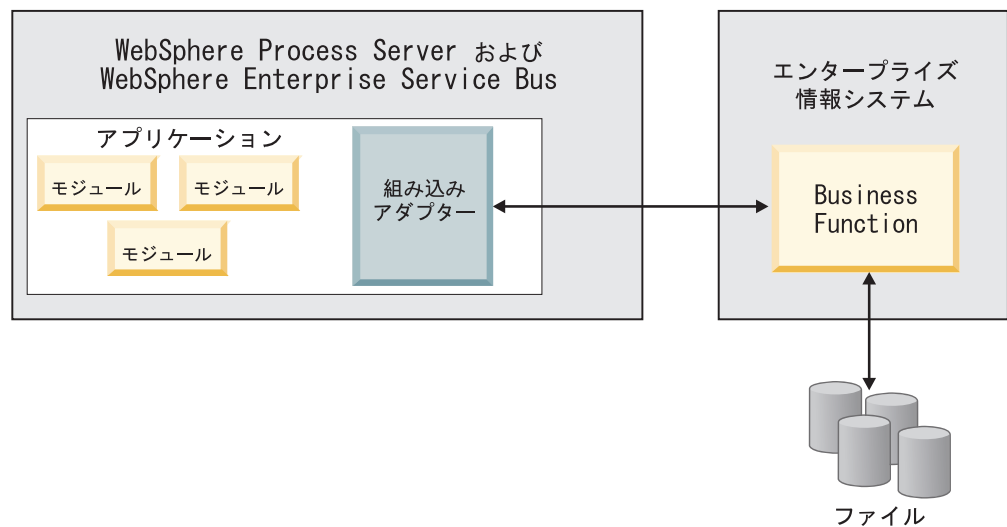
デプロイメント・オプションについて以下に説明します。

- 「**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、モジュールをすべてのアプリケーション・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

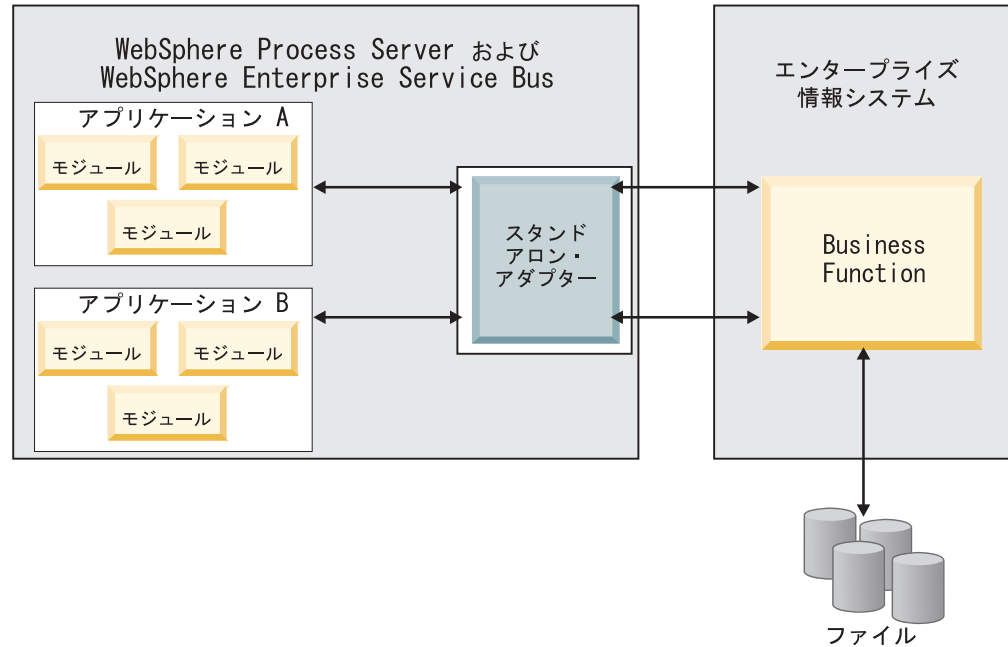
することで、それらのモジュールを不安定にするリスクを生じることなく、1つのモジュール内でアダプターをアップグレードできます。

- 「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PLOY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PLOYされた後、サーバー・インスタンス内のすべてのデPLOY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する方法、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。

- 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。

クラスター環境での 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、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、高可用性とスケー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 JCA (Java EE Connector Architecture) 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様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、アダプター・インスタンスは、Inbound 処理および Outbound 処理の両方を行えます。

Inbound 処理の高可用性

Inbound 処理は、Oracle データベース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 Oracle E-Business Suite 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 (Java EE Connector Architecture) コンテナーにより、enableHASupport リソース・アダプター・プロパティーが検査されます。enableHASupport プロパティー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 プロパティー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 Oracle E-Business Suite 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。

第 3 章 サンプル

WebSphere Adapter for Oracle E-Business Suite を構成するサンプル・コンテンツおよび成果物を使用すると、Oracle E-Business Suite データベースと WebSphere Business Integration システム間で Outbound および Inbound の統合を構造化するための方法を示すことができます。

データベースおよびアプリケーションのユーザー・アカウント要件

サンプルを使用するには、サンプル・コンテンツを実行するのに必要な成果物に対する権限があるデータベース・アカウントを使用する必要があります。また、Web アプリケーションのワークフロー管理者、システム管理者、および売掛金管理者の任務を遂行できる Oracle E-Business Suite アカウントを使用する必要があります。

これらのサンプルの目的を考慮すると、すべてのスクリプトを実行するユーザー・アカウントは、Oracle データベースの APPS ユーザーであるとみなされます。このユーザーには、以下の権限があります。

- アプリケーション (APPS) スキーマのコンテンツを変更する権限、およびコンテンツを作成する権限
- テーブルにデータを追加したり、テーブルからデータを削除したりする権限
- APPS スキーマで必須の実行可能コードを実行する権限

Oracle データベース管理者に問い合わせて、サンプル・コンテンツを実行するために使用するアカウントを確認してください。別のユーザー・アカウントを選択する場合は、データベース管理者と協力して、サンプル・コンテンツを実行するために必要なすべてのデータベース成果物に対する権限がそのユーザーにあることを確認してください。

Oracle E-Business Suite では、以下の任務に対する権限を持つアカウントが必要です。

- Web アプリケーションのワークフロー管理者
- システム管理者
- 売掛金管理者

注: 使用する Oracle E-Business Suite アカウントに、これらの任務に対するアクセス権限がない場合は、アクセス権限が追加されます。使用するアカウントに割り当てられている任務を変更するには、システム管理者の任務権限を持つアカウントにログオンし、「セキュリティ (Security)」->「ユーザー (User)」->「定義 (Define)」メニュー・オプションを選択します。ユーザーの任務変更についての具体的な情報については、Oracle Applications の資料を参照してください。

サンプル・ファイルへのアクセス

WebSphere Adapter for Oracle E-Business Suite をインストールすると、サンプル・ディレクトリーが作成されます。サンプル・ディレクトリーには、サンプル・アプリケーションを実行するためのコンテンツが含まれています。

このタスクの手順

1. WebSphere Adapter for Oracle E-Business Suite のインストール・ディレクトリーの下で Sample フォルダーを見つけます。
2. oracleEBSSamples.zip ファイルを unzip します。 サンプル・ファイルの説明を以下の表に示します。

表 1. サンプル・ファイル

ファイル名	説明
Ibm_websphere_event_table_create.sql	イベント・テーブル作成スクリプト
Ibm_websphere_events_s.sql	イベント ID シーケンス作成スクリプト
Ibm_customer_event_pkg.pls	顧客オブジェクト作成サンプルで使用されるプロシージャのパッケージ作成スクリプト
Ibm_customer_event_key_s.sql	イベント・キー・シーケンス作成スクリプト
Ibm_submit_request.sql	データをベース・テーブルに移動するために、インターフェース・テーブルに対してイベントが入力された後で呼び出される特殊なプロシージャを挿入するための SQL スクリプト
Raise_inbound_event.sql	顧客 Inbound イベントを発生させる SQL スクリプト
Ibm_create_synonyms.sql	サンプルのビジネス・オブジェクトの作成に使用される同義語名を作成するための SQL スクリプト
IMPCUST.wft	サンプル 4 で使用されるワークフロー・ファイル

追加サンプル: WebSphere® Integration Developer のオンライン・サンプル・ギャラリーでは、WebSphere Adapter を使用するとき役に立つサンプルやチュートリアルを提供しています。オンラインのサンプルおよびチュートリアル・ギャラリーには、WebSphere Integration Developer の始動時に開くウェルカム・ページからアクセスできます。WebSphere Adapter for Oracle E-Business Suite 用のサンプルを表示させるには、「取得」をクリックします。次に、表示されたディレクトリーをナビゲートして必要な選択を行います。

サンプル 1: Outbound データの Oracle データベースへの送信

WebSphere Adapter for Oracle E-Business Suite を使用すると、データを Oracle アプリケーションに作成して、Oracle アプリケーションからデータを取り出すことができます。このシナリオでは、WebSphere Adapter for JDBC を統合の方法として使用し、Oracle E-Business Suite の Financials データベースにカスタマー・レコードを作成して、そのカスタマー情報を Financials データベースから取り出します。

このタスクを実行する理由および時期

このシナリオを実行すると、Oracle E-Business Suite と WebSphere Process Server または WebSphere Enterprise Service Bus との間の情報の流れを Outbound 要求に対して確立する方法を示すことができます。その場合、使用する Oracle E-Business Suite モジュールは何でもかまいません。例えば、Receivables ではなく、Payables や General Ledger を使用できます。

関連概念

5 ページの『Outbound 処理のサンプル・アプリケーション』

このソリューションには、Outbound 処理のサンプル・アプリケーションがいくつか用意されています。サンプル 1 では Oracle データベースのインターフェース・テーブルを使用しますが、これは Oracle E-Business Suite の標準 Outbound シナリオになっています。サンプル 3 では API 呼び出しを使用し、サンプル 4 ではワークフローを使用します。

Outbound 処理の準備

Outbound 処理の準備のために SQL スクリプト・ファイルを実行する必要があります。後で Oracle データベースからビジネス・オブジェクトを生成する際の時間を節約するために、同義語を作成する必要があります。また、データベース表を使用してオブジェクトを処理するには、事前に Oracle E-Business Suite に成果物を挿入しておく必要があります。

このタスクの手順

1. 同義語の作成

SQL*Plus ツール (または SQL ステートメントを処理するための類似したプログラム) を開きます。データベース管理者 (DBA) 権限が付与されているデータベース・アカウントを使用して SQL*Plus にログオンします。

SQL スクリプト `ibm_create_synonyms.sql` を実行します。

例: `SQL> @C:\samplecontent\ibm_create_synonyms.sql;`

データベースのサイズ、および Oracle データベースに含まれているデータベース成果物の数によっては、外部サービス・ウィザードを実行してオブジェクトを生成するときに、大規模なリストを検索する処理時間が長くなる場合があります。同義語を作成すると、検索を高速に実行できるようになります。さらに同義語は、Oracle データベース・ドライバーに関連した問題を解決するのに役立ちます。この問題については、IBM テクニカル・ノート 1218775 を参照してください。WebSphere Adapters Technotes へのリンクについては、『参照』セクションの『関連情報』を参照してください。

2. Oracle E-Business Suite への成果物の挿入

ビジネス・オブジェクトを処理する前に、SQL スクリプト `ibm_submit_request.sql` を実行します。

これにより、ストアド・プロシージャ `IBM_WEBSPHERE_CUSTOMER_IMP` が Oracle APPS スキーマに置かれます。このプロシージャは、顧客データを

Oracle インターフェース・テーブルからベース・テーブルに移動するために使用されます。インターフェース・テーブルは、Create ビジネス・オブジェクトを構成するときに使用します。

プロジェクトの作成

Oracle E-Business Suite アプリケーションと通信するためのモジュールを作成およびデプロイする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。このウィザードは Adapter for JDBC プロジェクトを作成します。これは、アダプター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

Adapter for JDBC の資料にアクセスできることを確認します。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=//com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

このタスクの手順

1. 「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『デプロイメントのためのモジュールの構成』の情報を参照します。
2. 「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『プロジェクトの作成』に記載されている作業を実行します。このセクションに説明されている手順に従って、ご使用のワークスペースのプロジェクトを WebSphere Integration Developer にセットアップします。

Outbound 処理のモジュールの構成

データを処理するために必要なビジネス・オブジェクトを作成して、Oracle E-Business Suite に置く必要があります。このためには、WebSphere Adapter for JDBC の WebSphere Integration Developer で外部サービス・ウィザードを使用します。最初に、アダプターが特定のデータベースへの通信チャンネルをセットアップできるようにプロパティの値を構成して、ウィザードを初期設定します。次に、ビジネス・オブジェクトをディスカバリーするための照会を実行します。最後に、選択したビジネス・オブジェクトの操作などのプロパティを指定する必要があります。

外部サービス・ウィザードの接続プロパティの設定

ご使用のデータベース・インスタンス用に接続プロパティの値を設定します。これらのプロパティを設定すると、Adapter for JDBC の外部サービス・ウィザードで Oracle データベースに接続して、ディスカバリーを実行したり、サービス記述を作成したりすることができます。

始める前に

このタスクは、『プロジェクトの作成』のタスクで既に開始している外部サービス・ウィザード内で実行します。

このタスクの手順

1. 「必要なファイルおよびライブラリー」ウィンドウで、JDBC ドライバーの JAR ファイルに対応する上部ペインの横にある「追加」をクリックします。外部 JDBC ドライバーの場所を参照し、ドライバーの JAR ファイルを選択して「開く」をクリックします。

ウィザードでプロジェクトに追加する必要がある JDBC ドライバーは、ご使用のデータベースに付属している外部 JAR ファイルであり、データベース管理者またはデータベースの Web サイトから入手できます。

2. 「次へ」をクリックします。

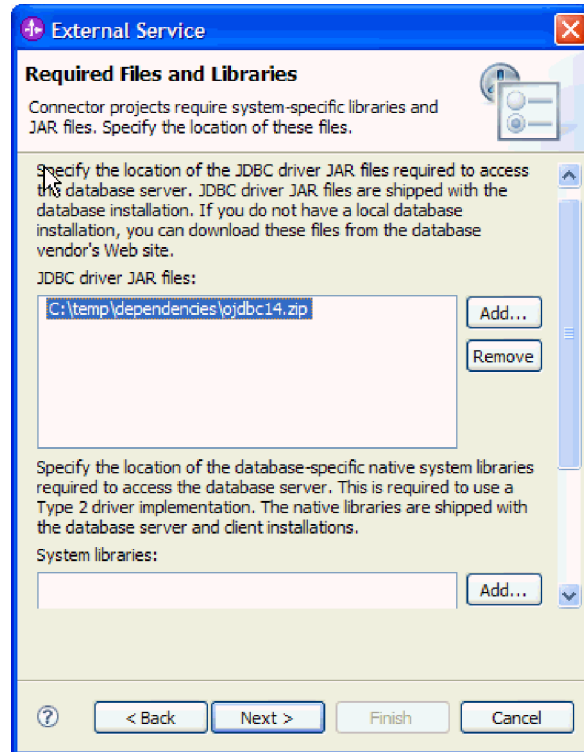


図 8. プロジェクトへの JDBC ドライバー JAR ファイルの追加

3. 「処理指示」ウィンドウで「Outbound」をクリックし、「次へ」をクリックします。
4. 「ディスカバリー構成」ウィンドウの左側のペインで、データベース・ベンダー、ドライバー、およびバージョンを選択します。
5. 以下の表に示すようにデータベースの情報を入力します。

以下の表に、サンプルで必要なプロパティーとその説明を示します。これらのプロパティーについて詳しくは、「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『参照』セクションの『外部サービス・ウィザードの接続プロパティー』を参照してください。この資料を参照するには、

http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

表 2. 外部サービス・ウィザードの接続プロパティ

プロパティ	説明
データベース	データベース名。Oracle データベースの場合、これはシステム ID (SID) です。データベースは、ホスト名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ホスト名	データベース・サーバーのホスト名または IP アドレス。ホスト名は、データベース名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ポート番号	データベース名およびホスト名とともに、データベースに接続するためのデータベース URL を生成するポート番号。
JDBC ドライバーのクラス名	データベースへの接続に使用される JDBC ドライバーのクラス名。
ユーザー名	使用しているデータベース・アカウント。ディスカバリー処理中に使用されます。ウィザードの後のほうでは、実行時として別の名前およびパスワードを指定できます。
パスワード	使用するアカウントのパスワード。

「次へ」をクリックします。

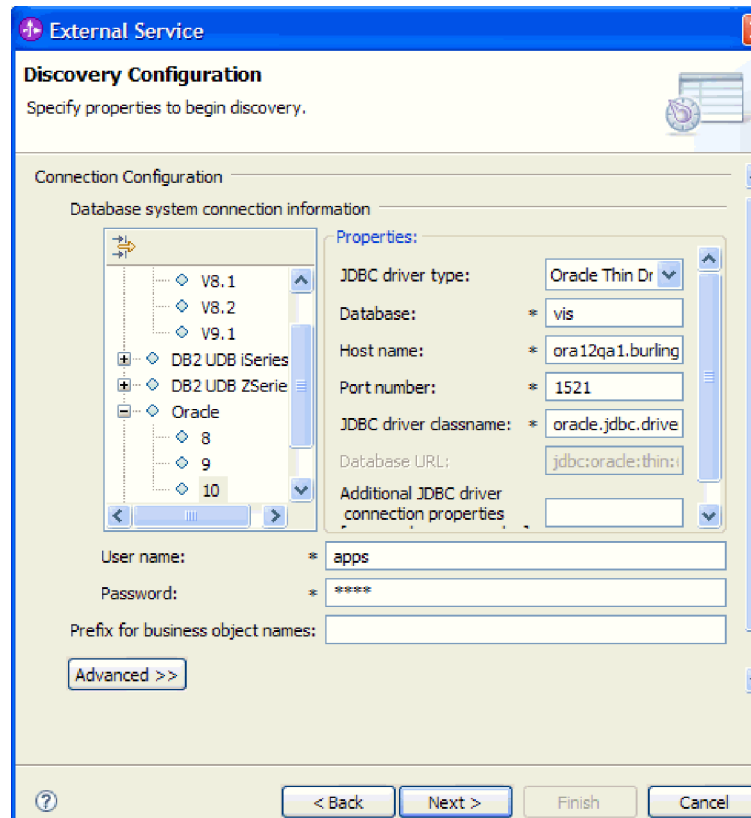


図9. 接続プロパティ値の追加

結果

ウィザードが、接続プロパティを使用してデータベースに接続します。

Create 操作でのビジネス・オブジェクトの選択およびリンク

このサンプルでは、1 次ビジネス・オブジェクトと子ビジネス・オブジェクトの組を使用して、カスタマー・データを Oracle E-Business Suite インターフェース・テーブルに取り込みます。Create 操作は、新規のカスタマー・データと、既存のカスタマー・データへの更新情報を書き込む場合に使用します。Create 操作に対応する 5 つのビジネス・オブジェクトを構成する必要があります。

データベース・テーブルでのビジネス・オブジェクト

Customer Profile オブジェクトは 1 次オブジェクトであり、残りのオブジェクトは子ビジネス・オブジェクトとして構造化されています。ビジネス・オブジェクトをリンクするには、親オブジェクトと子オブジェクトとの間にアプリケーション固有情報の外部キー関係を設定します。ビジネス・オブジェクトの性質や、アプリケーション固有情報の値の設定について詳しくは、「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『ビジネス・オブジェクト』セクションを参照してください。

次の表に、テーブル名、推奨の同義語、および各オブジェクトに収録されている情報の要旨を含む売掛金 (AR) スキーマのデータベース・テーブルを示します。

注: テーブル・データの詳細については、Oracle Applications の資料、特にカスタマー・インターフェースについて説明している Receivables のユーザー・ガイドのセクションを参照してください。

表 3. AR スキーマのデータベース・テーブル

テーブル名	同義語	説明
AR.RA_CUSTOMER_PROFILES_INT_ALL	ArIbm_C_Cust_Profile	Customer Profile インターフェース・テーブル。ここでは、カスタマーについての概略の情報が収録されています。
AR.RA_CUSTOMERS_INTERFACE_ALL	ArIbm_C_Cust_Interface	カスタマーおよびカスタマーの住所の 1 次テーブル
AR.RA_CONTACT_PHONES_INT_ALL	ArIbm_C_Contacts	連絡先および電話の情報
AR.RA_CUST_PAY_METHOD_INT_ALL	ArIbm_C_Pay_Methods	カスタマーに関連付けられている決済方法
AR.RA_CUSTOMER_BANKS_INT_ALL	ArIbm_C_Cust_Banks	カスタマーに関連付けられている銀行

次は、AR スキーマに対して照会を実行して、ビジネス・オブジェクトを選択します。

ビジネス・オブジェクトおよびサービスの選択

接続プロパティを設定した後は、照会を実行してデータベース・オブジェクトを検索します。ディスカバーしたオブジェクトのメタデータ・ツリーを参照して Oracle データベース内でのオブジェクトの構造を把握し、サービス記述に組み込むオブ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

このサンプルの場合は、売掛金 (AR) スキーマについて照会を実行し、Oracle E-Business Suite インターフェース・テーブルのデータを表す同義語を選択します。

このタスクの手順

1. フィルター・プロパティを指定します。
 - a. 「オブジェクトのディスカバーおよび選択」ウィンドウで、「照会の編集」をクリックします。
 - b. 「照会プロパティ」ウィンドウの「Schema 名パターン」フィールドに AR と入力して、Accounts Receivable (AR) スキーマを表示します。
 - c. 「ビジネス・オブジェクトの追加時に追加構成設定のプロンプトを出す」チェック・ボックスを選択して、「OK」をクリックします。

これで、インポート対象のディスカバー済みオブジェクトのいずれかを選択するたびに、そのオブジェクトのアプリケーション固有情報を入力するよう要求されるようになりました。

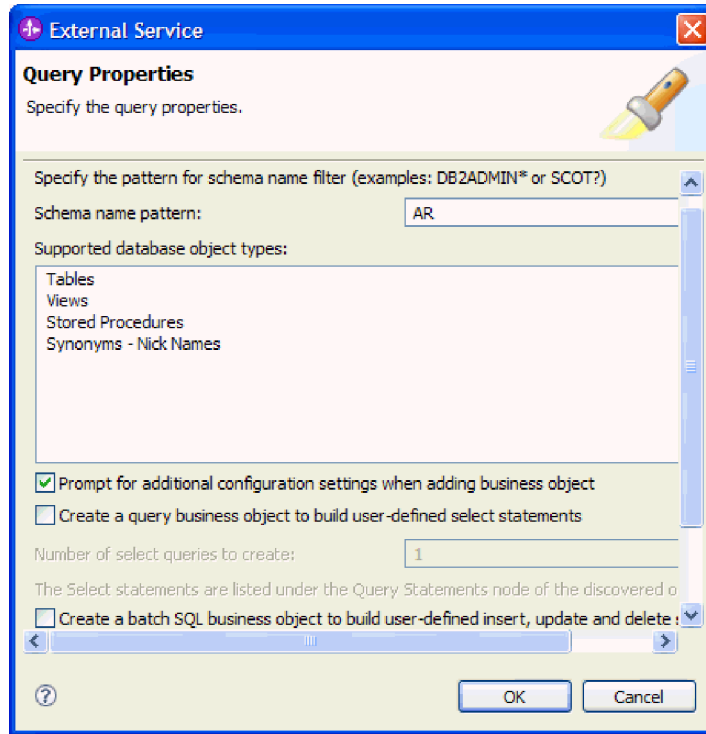


図 10. 照会プロパティの指定

2. メタデータ照会を実行します。

- a. 照会によってディスカバーされたオブジェクトを表示します。

「照会の実行」をクリックします。AR スキーマおよびスキーマのデータ・エレメントが表示されます。

- b. インポートするオブジェクトを選択します。

AR スキーマを展開します。「同義語 - ニックネーム (Synonyms - Nicknames)」を展開します。

同義語 **IBM_C_CUST_PROFILE** を強調表示し、> (「追加」アイコン) をクリックして、このオブジェクトを選択し、インポートの対象にします。

- c. ビジネス・オブジェクト・アプリケーション固有情報を追加します。

「**IBM_C_CUST_PROFILE** の構成プロパティ」ウィンドウで、「追加」をクリックして、**IBM_C_CUST_PROFILE** に関連するテーブルの 1 次キーを選択します。

REQUEST_ID を選択して、「OK」をクリックします。

「追加」をクリックし、この同義語に関連付けるストアード・プロシージャを選択して構成します。

「追加」ウィンドウで、**AfterCreateSP** を選択し、「OK」をクリックします。

「IBM_C_CUST_PROFILE の構成プロパティ」ウィンドウの **AfterCreateSP** 領域で、以下の値を設定します。

- 「スキーマ名パターン」フィールドに APPS と入力します。
- 「スキーマ名」リストから APPS を選択します。

注: APPS スキーマを選択してから「ストアード・プロシージャ名パターン」フィールドがアクティブになるまでに長時間の遅延が発生する場合があります。

- 「ストアード・プロシージャ名パターン」フィールドに IBM_WEBSPPHERE_CUSTOMER_IMP と入力します。
- 「ストアード・プロシージャ名」リストから IBM_WEBSPPHERE_CUSTOMER_IMP を選択します。

「OK」をクリックします。

The screenshot shows a dialog box titled "Configuration Properties for 'IBM_C_CUST_PROFILE'". It has a subtitle "Specify configuration properties for the selected object." and a help icon. The dialog is divided into several sections:

- Select primary key for table IBM_C_CUST_PROFILE**: A list box containing "REQUEST_ID" with "Add..." and "Remove" buttons.
- AfterCreateSP**: A list box containing "AfterCreateSP" with "Add..." and "Remove" buttons.
- AfterCreateSP**: A section for configuring the stored procedure:
 - Schema name pattern: Text box with "APPS".
 - Schema name: Dropdown menu with "APPS" selected.
 - Stored procedure name pattern: Text box with "IBM_WEBSPPHERE_".
 - Stored procedure name: Dropdown menu with "IBM_WEBSPPHEI" selected.

At the bottom, there are "OK" and "Cancel" buttons, and a help icon.

図 11. ビジネス・オブジェクト・アプリケーション固有情報を追加します。

しばらくすると、選択したオブジェクトが「選択済みオブジェクト」ペインに表示されます。

このサンプルでは、Oracle の組み込みインターフェース・テーブルおよび基本テーブルを使用してデータを処理します。Create 操作が必要なのは、新規のカスタマー・データと既存のカスタマー・データへの更新情報の両方を書き込む場合に限られます。オブジェクトの Create 操作を呼び出す一環として、処理の対象となるオブジェクトに AfterCreateASI という値を使用して、

ストアード・プロシージャ IBM_WEBSPPHERE_CUSTOMER_IMP が実行されます。このストアード・プロシージャを使用すると、インターフェース・テーブルから基本テーブルヘデータが移動します。

- d. インポートする別のオブジェクトを選択して、子オブジェクトを親オブジェクトにリンクします。

「オブジェクトのディスカバリーおよび選択」ウィンドウで、同義語 **IBM_C_CUST_INTERFACE** を強調表示して「>」（「追加」アイコン）をクリックし、このオブジェクトをインポート対象として選択します。

「IBM_C_CUST_INTERFACE の構成プロパティ」ウィンドウで、「追加」をクリックして 1 次キーを選択します。

CUSTOMER_KEY を選択して、「OK」をクリックします。

IBM_C_CUST_PROFILE を選択して、IBM_C_CUST_INTERFACE に関連するテーブルの親を設定します。

「子の列ごとに親テーブル列を選択して外部キー関係を作成する」で、**request_id** を **REQUEST_ID** 外部キー属性の外部キー属性値として選択することにより、子の属性を親の属性にリンクします。

「親オブジェクトが子オブジェクトを所有する (カスケード削除)」にチェック・マークを付け、「OK」をクリックします。

- e. 残りの同義語を選択します。

次の表の残りの同義語ごとに手順 2d を繰り返します。各同義語に 1 次キーが指定されます。

表 4. インポートの対象を選択し、親オブジェクトへリンクするための残りの同義語

同義語	1 次キー
AR.IBM_C_CONTACTS	CONTACT_KEY
AR.IBM_C_CUST_BANKS	BANK_ACCOUNT_NUM
AR.IBM_C_PAY_METHODS	ORG_ID

完了したら、「次へ」をクリックします。

オブジェクト・レベル、操作、および属性のアプリケーション固有の情報について詳しくは、「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『参照』セクションの『ビジネス・オブジェクト情報』を参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=/com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

Retrieve 操作でのビジネス・オブジェクトの選択およびリンク

このサンプルでは、2 番目のオブジェクト・セットを使用して、カスタマー・データを Oracle E-Business Suite 基本テーブルから直接取り出します。Retrieve 操作では基本テーブル内の値は変更されないため、これらのテーブルからはデータを直接

取り出すことができます。インターフェース・テーブルを使用する必要はありません。 Retrieve 操作に対応するビジネス・オブジェクトを構成する必要があります。

売掛金 (AR) スキーマの基本テーブル

次の表に、テーブル名、推奨の同義語、および説明を含む売掛金 (AR) スキーマの基本テーブルを示します。次のセクションでは、Retrieve 操作のために構成するビジネス・オブジェクトの関係を表す図を示します。

表 5. AR スキーマの基本テーブル

テーブル名	同義語	説明
AR.HZ_CUST_ACCOUNTS	ArIbm_R_Cust_Accounts	カスタマー番号、タイプ、およびその他のカスタマー情報。
AR.HZ_CUSTOMER_PROFILES	ArIbm_R_Cust_Profiles	カスタマーの概略プロフィール情報。
AR.HZ_PARTIES	ArIbm_R_Parties	カスタマー名。会社ではなく個人の場合は、個人の詳細情報も付記。
AR.HZ_PARTY_SITES	ArIbm_R_Party_Sites	カスタマーの住所 ID と住所のロケーション ID との間のリンク。
AR.HZ_CUST_ACCT_SITES_ALL	ArIbm_R_Acct_Sites	住所 ID。
AR.HZ_CUST_SITE_USES_ALL	ArIbm_R_Site_Uses	カスタマーのビジネス上の住所の目的 (BILL_TO、SHIP_TO など)。
AR.HZ_LOCATIONS	ArIbm_R_Locations	住所の詳細。
AR.HZ_CONTACT_POINTS	ArIbm_R_Contacts_Sites	連絡先および電話の詳細情報。 サイトとカスタマー (PARTY_SITES および PARTIES) の両方が連絡先の詳細について同じテーブルを使用するため、このテーブルはリストに 2 回登録されます。
AR.HZ_CONTACT_POINTS	ArIbm_R_Contacts_Customer	連絡先および電話の詳細情報。 サイトとカスタマー (PARTY_SITES および PARTIES) の両方が連絡先の詳細について同じテーブルを使用するため、このテーブルはリストに 2 回登録されます。

ビジネス・オブジェクトの関係

Retrieve 操作を正常に実行するため、カスタマー・オブジェクトとその子オブジェクトを接続するリンクは、Oracle データベースでのリンクと一致する必要があります。次の図では、Retrieve 操作のために構成する必要があるオブジェクトの関係を示します。

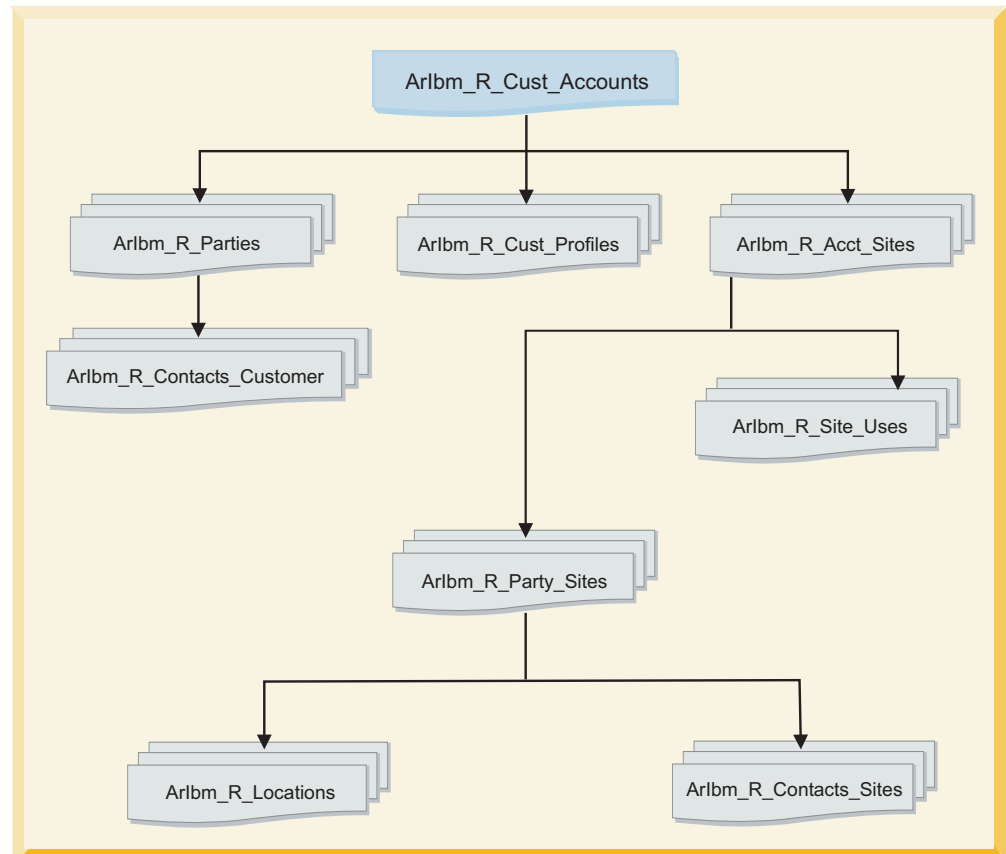


図 12. Retrieve 操作のためのオブジェクトの関係

オブジェクトをリンクするには、子オブジェクトを作成し、親オブジェクトと子オブジェクトとの間にアプリケーション固有情報の外部キー関係を設定します。ビジネス・オブジェクトの性質について詳しくは、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『技術的な概説』の『ビジネス・オブジェクト』セクションを参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=/com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

Retrieve のためのビジネス・オブジェクトの選択およびリンク

売掛金 (AR) スキーマについて照会を実行したら、Oracle E-Business Suite 基本テーブルのデータを表す同義語を選択します。親と子の階層におけるオブジェクトの構成も行います。

始める前に

接続プロパティの設定、フィルター・プロパティの指定、およびディスカバリー済みオブジェクトを表示するための照会の実行を既に行っていることが必要です。

このタスクの手順

1. インポートするオブジェクトを選択します。

「オブジェクトのディスカバリーおよび選択」ウィンドウで、**AR** スキーマおよび **Synonyms - Nicknames** を展開したら、同義語である **IBM_R_CUST_ACCOUNTS** を強調表示して > (「追加」アイコン) をクリックし、インポートするオブジェクトを選択します。

2. ビジネス・オブジェクト・アプリケーション固有情報を追加します。

「IBM_R_CUST_ACCOUNTS の構成パラメーター (Configuration Parameters for IBM_R_CUST_ACCOUNTS)」ウィンドウで、「追加」をクリックして、**IBM_R_CUST_ACCOUNTS** に関連するテーブルの 1 次キーを選択します。

CUST_ACCOUNT_ID を選択して、「OK」をクリックします。

3. インポートする別のオブジェクトを選択して、子オブジェクトを親オブジェクトにリンクします。

- a. 同義語 **IBM_R_CUST_PROFILES** を強調表示して、> (「追加」アイコン) をクリックします。

- b. 「IBM_R_CUST_PROFILES の構成パラメーター (Configuration Parameters for IBM_R_CUST_PROFILES)」ウィンドウで、「追加」をクリックして、**IBM_R_CUST_PROFILES** に関連するテーブルの 1 次キーを選択します。

APPLICATION_ID を選択して、「OK」をクリックします。

- c. **IBM_R_CUST_ACCOUNTS** を選択して、**IBM_R_CUST_PROFILES** に関連するテーブルの親を設定します。

- d. 「子の列ごとに親テーブル列を選択して外部キー関係を作成する」の下で、外部キー属性およびその属性の外部キー値を選択して、子を親の属性にリンクします。

外部キー属性 **CUST_ACCOUNT_ID** の場合は、外部キー属性値 **CUST_ACCOUNT_ID** を選択します。

「親オブジェクトが子オブジェクトを所有する (カスケード削除)」にチェック・マークを付け、「OK」をクリックします。

4. 残りの同義語を選択します。

残りの同義語については、次の表に示す順に、同義語ごとに手順 3 を繰り返します。同義語ごとに表に示されている 1 次キー、親、外部キー属性、および外部キー属性値を使用します。

完了したら、「次へ」をクリックします。

表 6. 外部キー値および子オブジェクトを必要とするオブジェクト

同義語	1 次キー	親	外部キー属性	外部キー属性値
ArIbm_R_Parties	party_id	Aribm_R_Cust_Accounts	party_id	party_id

表 6. 外部キー値および子オブジェクトを必要とするオブジェクト (続き)

同義語	1 次キー	親	外部キー属性	外部キー属性値
ArIbm_R_Acct_Sites	party_site_id および cust_acct_ site_id	Aribm_R_Cust_ Accounts	cust_account_id	cust_account_id
ArIbm_R_Party_Sites	location_id および party_site_id	Aribm_R_Acct_Sites	party_site_id	party_site_id
ArIbm_R_Site_Uses	application_id	Aribm_R_Acct_Sites	cust_acct_site_id	cust_acct_site_id
ArIbm_R_Locations	address_key	Aribm_R_Party_Sites	location_id	location_id
ArIbm_R_Contacts_Sites	application_id	Aribm_R_Party_Sites	owner_table_id	party_site_id
ArIbm_R_Contacts_ Customer	application_id	Aribm_R_Parties	owner_table_id	party_id

選択済みオブジェクトの構成

データベース・オブジェクトを選択してリンクした後、すべてのビジネス・オブジェクトに適用する操作などのプロパティを指定します。

このタスクの手順

1. 操作を選択します。

外部サービス・ウィザードの「複合プロパティの構成」ウィンドウでは、Outbound サービス・タイプとしてアダプターがサポートしている操作が「操作」ペインに表示されます。次の操作を強調表示させて「除去」をクリックすることにより、除去します。

- **Update**
- **Delete**
- **RetrieveAll**
- **ApplyChanges**

以下の操作が残ります。

- **Create**
 - **Retrieve**
2. 以下のフィールドのデフォルト値は、そのままにします。
 - **RetrieveAll** 操作の最大レコード数
 - **ビジネス・オブジェクト Namespace**
 - **フォルダー**
 - **ビジネス・オブジェクトごとにビジネス・グラフを生成**
 3. 「次へ」をクリックします。

次のタスク

次に、実行時に使用するデプロイメント情報、およびサービスをモジュールとして保存するための情報を指定します。

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

アダプターが使用するプロパティを構成して、Outbound 処理を行うための特定のデータベースへの通信チャンネルをセットアップするには、外部サービス・ウィザードを使用します。また、すべての成果物およびプロパティ値を保管できるビジネス・インテグレーション・モジュールを作成する必要があります。

このタスクの手順

1. 構成プロパティ値を設定する

- a. 「サービスの生成とデプロイメントの構成」ウィンドウの「**コネクター・プロジェクトのデプロイ**」フィールドで、オプション「**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**」が選択されていることを確認します。
- b. 「**拡張**」をクリックし、「**接続プロパティ**」、「**データベース・システム接続情報**」の下にあるすべてのプロパティを確認します。

Outbound 処理の場合は、管理接続ファクトリー・プロパティおよびリソース・アダプター・プロパティ用のプロパティ・フィールドが表示されます。

- c. 次の表に示すように必須接続プロパティの値を設定します。

フィールドは、ウィザードの開始時に指定した接続情報で初期化されます。「データベース・ベンダー」および「ユーザー名」の既存の値は、必要に応じて編集できます。パスワードを入力する必要があります。

表7. データベース・システム接続情報

プロパティ	値
ユーザー名	使用するデータベースのアカウント
パスワード	使用するアカウントのパスワード
データベース・ベンダー	Oracle

- d. 「**次へ**」をクリックします。

これらのプロパティについては、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『**参照**』セクションを参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

2. モジュールの作成

- a. 「サービス・ロケーション・プロパティ」ウィンドウの「**モジュール**」フィールドの隣にある「**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)**」が選択されていることを確認し、「**次へ**」をクリックします。
- c. 「モジュール」ウィンドウで `Oracle_Outbound` と入力し、「**終了**」をクリックします。
- d. 「サービス・ロケーション・プロパティ」ウィンドウで、「**終了**」をクリックします。

結果

このモジュールには、生成されたすべての成果物が保持されます。

モジュールの保存

Outbound 処理を行うため、作成した成果物およびプロパティ値が収容されているビジネス・インテグレーション・モジュールを保存します。

このタスクの手順

1. 「ビジネス・インテグレーション (Business Integration)」ビューで、**Oracle_Outbound** モジュールをクリックします。
2. 「ファイル (File)」>「保管」を選択します。

注: 「保管」のオプションが非アクティブ (グレー表示) になっている場合、そのモジュールは既に保存されています。

テスト用のモジュールのデプロイ

アダプター・アプリケーションをテストするには、Outbound 処理用のモジュールを WebSphere Integration Developer 統合テスト・クライアントにインストールする必要があります。

始める前に

外部サービス・ウィザードを実行して、Oracle データベースのインポート・ファイルを収録したモジュールを生成します。

このタスクの手順

1. WebSphere Integration Developer で「サーバー (Servers)」タブをクリックして、「サーバー (Servers)」ビューを開きます。

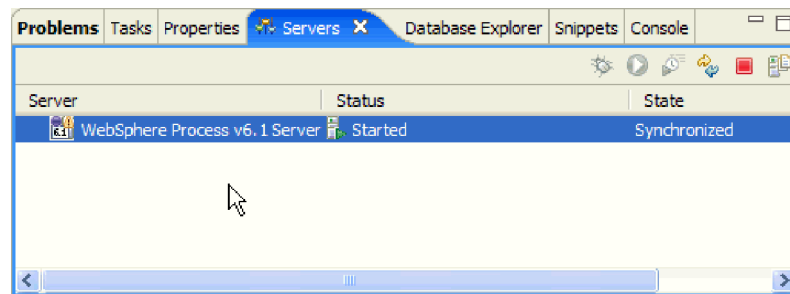


図 13. 「サーバー (Servers)」ビュー

2. サーバーが「サーバー」ビューに表示されない場合は、「サーバー」ビューを右クリックして「新規」→「サーバー」を選択します。ご使用の WebSphere Process Server または WebSphere Enterprise Service Bus を選択します。

「次へ」をクリックし、「終了」をクリックします。

3. WebSphere Process Server または WebSphere Enterprise Service Bus を実行していない場合は、その名前を右クリックし、「開始 (Start)」を選択します。「状況」項目が「始動済み」に変化します。

4. WebSphere Process Server または WebSphere Enterprise Service Bus の名前を右クリックし、「プロジェクトの追加および除去」を選択します。
5. 左側の「使用可能プロジェクト」リストから **Oracle_OutboundApp** モジュールを選択し、「追加」をクリックします。「終了」をクリックします。

次のタスク

次に、統合テスト・クライアント内でアSEMBルされたアプリケーションをテストできます。

Create 操作のテスト

WebSphere Process Server または WebSphere Enterprise Service Bus への Outbound 処理アプリケーションのデプロイ後に、WebSphere Integration Developer 統合テスト・クライアントを使用して、アSEMBルされたアダプター・アプリケーションをテストします。最初に Create 操作をテストし、次に Retrieve 操作をテストします。

このタスクを実行する理由および時期

Create 操作をテストするには、顧客項目を作成して 2 つのアドレスと 1 つの電話番号を指定します。

このタスクの手順

1. テストするモジュールの選択

WebSphere Integration Developer で、「ビジネス・インテグレーション」パースペクティブの「ビジネス・インテグレーション」ビューに切り替えます。

Oracle_Outbound モジュールを右クリックして、「テスト」>「モジュールのテスト (Test Module)」を選択します。

2. 「**verb**」はデフォルト値のままにしておきます。

値は、下記の『属性値の設定』の図に示す値と異なる場合があります。

3. テスト値の設定

Oracle_Outbound_Test ウィンドウの矢印を使用して、以下の表に示されているテスト値を設定します。

表 8. Create 操作用のテスト値

項目	テスト値
構成	デフォルト・モジュール・テスト
モジュール	Oracle_Outbound
コンポーネント	JDBCOutboundInterface
インターフェース	JDBCOutboundInterface
操作	createArIbm_C_Cust_ProfileBG

注: 「コンポーネント」 および「インターフェース」のテスト値には番号が付いていることがあります (JDBCOutboundInterface1 など)。WebSphere Integration Developer は、それぞれの名前がワークスペース内で固有になるように番号を付加します。

4. 入力ビジネス・オブジェクトの属性値の設定

顧客プロフィール・オブジェクト (最上位オブジェクト ArIbm_C_Cust_Profile) の属性値を設定します。この例では、2 つのアドレスと 1 つの連絡先を使用しているため、Addresses タイプの 2 つの子オブジェクト (Aribm_C_Cust_Interfaceobj) と、Contacts タイプの 1 つの子オブジェクト (Aribm_C_Contactsobj) の値を設定する必要があります。以下の表では、各オブジェクトの値と制約事項 (値が固有でなければならないかどうかなど) を示しています。

a. 固有の ID 値を作成して記録します。

Oracle では、Outbound 処理用の ID 値が必要です。これらの値により、Oracle E-Business Suite に挿入されるレコードが一意的に識別されます。これらの値は、対応する挿入済みデータを正常に取得するために必要です。ID 値は複数のオブジェクトで使用されるため、さまざまなオブジェクトで繰り返し入力するには、これらの値を書き留めておく必要があります。以下の 5 つの値のそれぞれについて、固有のストリング値を作成してください。

- CustomerReference#
- AddressReference#1
- AddressReference#2
- TelephoneReference#
- CustomerName

注: Oracle インターフェース・テーブル内のデータの必須の値および他の指定可能な値について詳しくは、Oracle アプリケーションの資料 (特に「Oracle Receivables ユーザーズ・ガイド」) を参照してください。

b. 「イベント」ウィンドウの「初期要求パラメーター」の下で、値を設定する必要がある各属性の名前の向かい側にある「値」列をクリックして、以下の表の値を入力します。

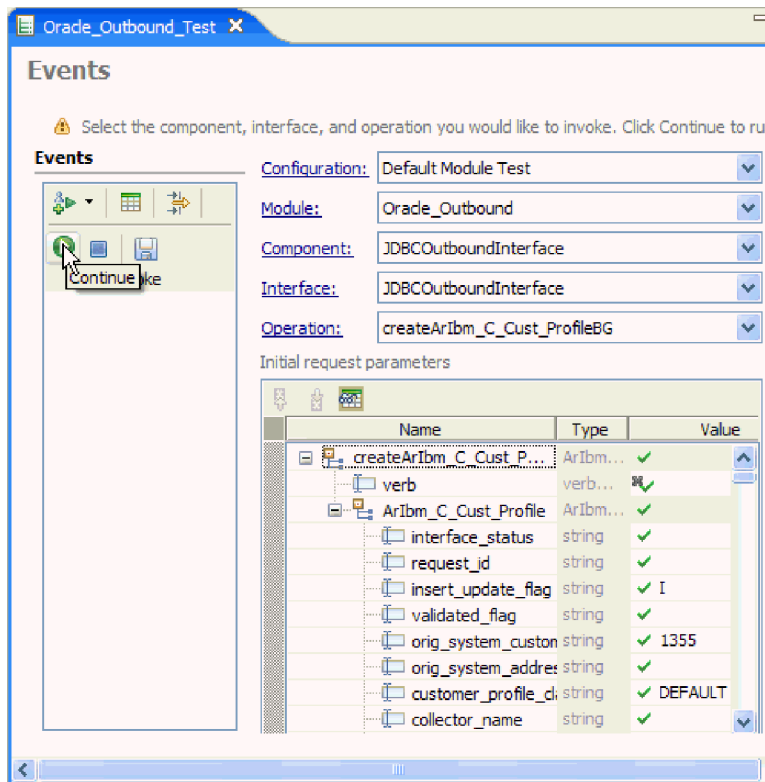


図 14. 属性値の設定

重要:

- 属性の値を <null> に設定するには、その属性の値フィールドを右クリックします。「値の設定 (Set to)」を選択し、スクロールダウンして「<null>」を選択します。
- 子オブジェクトを追加するには、該当するタイプの属性を右クリックして「エレメントの追加」を選択します。属性ごとに追加するエレメント数を入力します。このサンプルでは、属性 ArIbm_C_Cust_Interfaceobj にエレメントを追加するときには、このタイプの子オブジェクトが 2 つ存在するため 2 を入力します (2 つのアドレス)。属性 ArIBM_C_Contactsobj にエレメントを追加するときには、このタイプの子オブジェクトが 1 つ存在するため 1 を入力します (1 つの連絡先)。

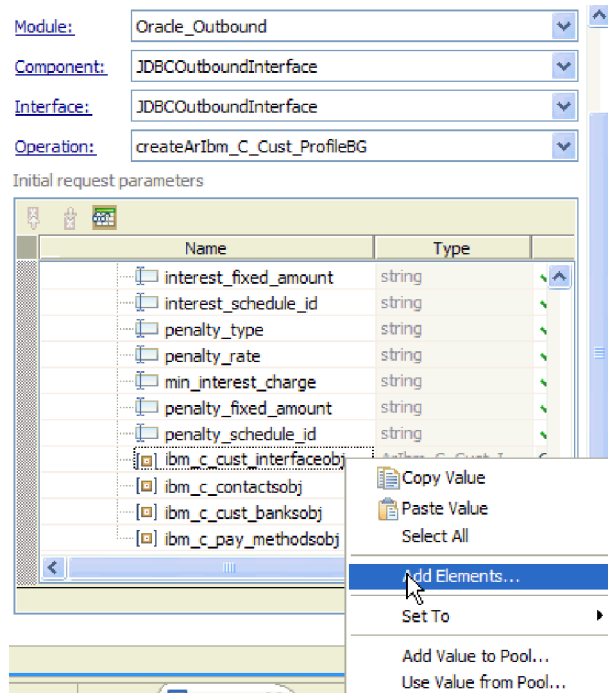


図 15. 「エレメントの追加」を使用した子オブジェクトの追加

表 9. 最上位オブジェクトの値

属性	値	注
Insert_update_flag	I	
Orig_system_customer_ref	CustomerReference #	記録した値を使用してください。値は、固有でなければなりません。
Customer_profile_class_name	DEFAULT	
Credit_hold	N	
Last_updated_by	-1	
Last_update_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Creation_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Created_by	-1	
Org_id	204	

以下の表に、Address 属性の最初のエレメントの値をリストします。

表 10. Address 属性の最初のエレメントの値

属性	値	注
Orig_system_customer_ref	CustomerReference#	記録した値を使用してください。値は、固有でなければなりません。
Site_use_code	BILL_TO	

表 10. Address 属性の最初のエレメントの値 (続き)

属性	値	注
Orig_system_address_ref	<i>AddressReference#1</i>	記録した値を使用してください。値は、固有でなければなりません。
Insert_update_flag	I	
Customer_name	<i>CustomerName</i>	記録した値を使用してください。値は、固有でなければなりません。
Customer_number	<NULL>	
Customer_status	A	
Primary_site_use_flag	Y	
Location	<NULL>	
Address1	Test Address 1	独自のアドレス情報を使用することもできます。
Address2	<NULL>	
Address3	<NULL>	
Address4	<NULL>	
City	San Mateo	
State	CA	
Province	<NULL>	
County	San Mateo	
Postal_code	94010	
country	US	
Cust_category_code	CUSTOMER	
Last_updated_by	-1	
Last_update_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Created_by	-1	
Creation_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Org_id	204	
Customer_name_phonetic	<i>CustomerName</i>	記録した値を使用してください。

以下の表に、Address 属性の 2 番目のエレメントの値をリストします。

表 11. Address 属性の 2 番目のエレメントの値

属性	値	注
Orig_system_customer_ref	<i>CustomerReference#</i>	記録した値を使用してください。値は、固有でなければなりません。
Site_use_code	SHIP_TO	

表 11. Address 属性の 2 番目のエレメントの値 (続き)

属性	値	注
Orig_system_address_ref	AddressReference#2	記録した値を使用してください。値は、固有でなければなりません。
Insert_update_flag	I	
Customer_name	CustomerName	記録した値を使用してください。値は、固有でなければなりません。
Customer_number	<NULL>	
Customer_status	A	
Primary_site_use_flag	Y	
Location	<NULL>	
Address1	Test Address 2	独自のアドレス情報を使用することもできます。
Address2	<NULL>	
Address3	<NULL>	
Address4	<NULL>	
City	San Mateo	
State	CA	
Province	<NULL>	
County	San Mateo	
Postal_code	94010	
country	US	
Cust_category_code	CUSTOMER	
Last_updated_by	-1	
Last_update_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Created_by	-1	
Creation_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Org_id	204	
Customer_name_phonetic	CustomerName	記録した値を使用してください。

以下の表に、Contacts 属性のエレメントの値をリストします。

表 12. Contacts 属性のエレメントの値

属性	値	注
Orig_system_contact_ref	<NULL>	
Orig_system_telephone_ref	TelephoneReference#	記録した値を使用してください。値は、固有でなければなりません。

表 12. *Contacts* 属性のエレメントの値 (続き)

属性	値	注
Orig_system_customer_ref	<i>CustomerReference#</i>	記録した値を使用してください。値は、固有でなければなりません。
Orig_system_address_ref	<i>AddressReference#1</i>	記録した値を使用してください。値は、固有でなければなりません。
Insert_update_flag	I	
Contact_first_name	<NULL>	
Contact_last_name	<NULL>	
Contact_title	<NULL>	
Contact_job_title	<NULL>	
Telephone	5551212	独自の電話番号を使用することもできます。
Telephone_extension	1234	ない場合は <null> を使用してください。
Telephone_type	FAX	
Telephone_area_code	650	
Last_update_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Last_updated_by	-1	
Creation_date	2007-06-25 12:00:00	今日の日時。時刻は正確でなくても構いません。
Created_by	-1	
Email_address	<NULL>	
Org_id	204	

- 「イベント (Events)」の下の左側のペインで、「**継続 (Continue)**」をクリックします。「デプロイメント・ロケーション (Deployment Location)」ウィンドウで、アプリケーションのデプロイ先のサーバーの名前を選択して「**終了**」をクリックします。「統合テスト・クライアントの開始 (Starting the Integration Test Client)」ウィンドウが一時的に表示されます。
- 顧客データが正しくインポートされていることを確認します。

以下の 2 つの方法 (a または b) のうち、いずれかの方法で顧客データを検証します。

- Oracle E-Business Suite にログオンして、売掛金管理者の任務に切り替えます。「顧客 (Customers)」>「標準 (Standard)」メニューにナビゲートして、「顧客の検索 (Find Customers)」ウィンドウを開きます。インポートされた顧客を照会します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

Oracle の顧客画面に表示された顧客番号を書き留めてください。

SQL*Plus ツール、または SQL コマンドを処理するための類似したプログラムを開始します。

```
次の照会を実行します。SQL> select cust_account_id from  
ar.hz_cust_accounts where account_number = customer number;
```

ここで、*customer number* は Oracle の顧客画面で書き留めた番号です。

この照会の **cust_account_id** 値を書き留めます。この番号は、Retrieve 操作をテストするときに使用します。

- b. オプションで、データベース・エディターを開いて、このサンプルで使用する Oracle データベースに接続できます。

creation_date 属性を参照して、挿入した顧客の行を探します。該当する行では、この属性が今日 (トップレベル・オブジェクトの場合は挿入日) になっています。

cust_account_id の値を書き留めます。この番号は、Retrieve 操作をテストするときに必要です。

次のタスク

次に Retrieve 操作をテストします。

関連資料

101 ページの『サンプル・アプリケーションのトラブルシューティング情報』これらのサンプルについてタスクの実行中に問題が発生した場合、問題は Adapter for JDBC の実行または Oracle データベースとの対話に関連する可能性があります。2 つの既知の問題について説明します。

Retrieve 操作のテスト

Create 操作のテストが完了し、インポートされた顧客が Oracle E-Business Suite 基本テーブルに存在することを確認したら、Retrieve 操作をテストします。

始める前に

Create 操作をテストしたときに顧客データが適切にインポートされたことを確認した後、実行した SQL 照会で得られた *customer_account_id* 値が必要です。

このタスクを実行する理由および時期

WebSphere Integration Developer 統合テストクライアントを使用して、アセンブルされたアダプター・アプリケーションをテストします。

このタスクの手順

1. テストするモジュールの選択

WebSphere Integration Developer の「ビジネス・インテグレーション (Business Integration)」パースペクティブで、**Oracle_Outbound** モジュールを右クリックして、「テスト (Test)」>「モジュールのテスト (Test Module)」を選択します。

2. 「**verb**」はデフォルト値のままにしておきます。

3. テスト値の設定

Oracle_Outbound_Test ウィンドウの矢印を使用して、以下の表に示されているテスト値を設定します。

表 13. Retrieve 操作用のテスト値

項目	テスト値
モジュール	Oracle_Outbound
コンポーネント	JDBCOutboundInterface
インターフェース	JDBCOutboundInterface
操作	retrieveArIbm_R_Cust_AccountsBG

注: 「コンポーネント」 および 「インターフェース」 のテスト値には番号が付いていることがあります (JDBCOutboundInterface1 など)。WebSphere Integration Developer は、それぞれの名前がワークスペース内で固有になるように番号を付加します。

4. 入力ビジネス・オブジェクトの属性値の設定

- 「イベント」ウィンドウの「初期要求パラメーター」で、「名前」列の **cust_account_id** の向かい側にある「値」をクリックします。『Create 操作のテスト』で実行した SQL 照会で得られた *cust_account_id* 値に一致する値を設定します。
- 残りの値のデフォルト値は変更しないでください。
- 「イベント (Events)」の下の左側のペインで、「**継続 (Continue)**」をクリックします。「デプロイメント・ロケーション (Deployment Location)」ウィンドウで、オブジェクトのデプロイ先のサーバーの名前を選択して「終了」をクリックします。

オブジェクトの処理中に「統合テスト・クライアントの開始 (Starting the Integration Test Client)」ウィンドウが表示されます。

- 返されたオブジェクトが、Oracle インターフェース・テーブルにインポートされた顧客情報と一致することを確認します。

サンプル・コンテンツのクリア

Create 操作および Retrieve 操作をテストした後は、サンプル・コンテンツをクリアしてデータを元の状態に戻します。このためには、Oracle E-Business Suite でカスタマーを非活動状態にしてから SQL コマンドを実行して、ストアード・プロシージャを削除します。

このタスクの手順

- Oracle E-Business Suite Client を使用して、売掛金管理者の任務にログインします。「カスタマー (Customers)」> 「標準 (Standard)」メニューに移動して、カスタマーを探します。カスタマーの「状況 (Status)」の値を「活動停止 (Inactive)」に変更します。その後、そのカスタマーを保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

2. SQL*Plus ツール、または SQL コマンドを処理するための同様なプログラムを使用してデータベースにログインし、コマンドを実行して、データベースから成果物を消去します。このタスクを実行する際に支援が必要な場合、データベース管理者に相談してください。

サンプル 2: データベースからの Inbound データの受信

このサンプルでは、Oracle E-Business Suite 内部で Business Event System を使用して、イベント・テーブルにカスタマー・イベントを取り込みます。さらに、WebSphere Adapter for JDBC を使用してイベントのポーリングを実行し、カスタマー・データを取り出します。このシナリオを実行すると、Oracle E-Business Suite と WebSphere Process Server または WebSphere Enterprise Service Bus との間の情報の流れを Inbound イベントに対して確立する方法がわかります。

関連概念

8 ページの『Inbound 処理のサンプル・アプリケーション』

サンプル 2 では、Oracle Business Event System を使用して、Oracle E-Business Suite からイベント・テーブルへ Inbound カスタマー・データを送信します。このイベント・テーブルでは、Adapter for JDBC によってデータを読み取ることができます。Adapter for JDBC はイベントをイベント・テーブルから取り出し、イベントで提供される情報に基づいて Oracle データベースからビジネス・オブジェクトを取得します。次に、Adapter for JDBC は、Oracle データベースから取得したビジネス・オブジェクトを処理します。このサンプルに組み込まれているコンテンツについては、このセクションで説明します。

Inbound 処理の準備

Inbound 処理に対応する環境を準備するために必要なセットアップ手順を実行します。ある手順では、ストアド・プロシージャを Oracle E-Business Suite に挿入し、別の手順では、外部サービス・ウィザードで実行したデータベース検索の効率を向上させます。オプションの手順を実行すると、Oracle データベースおよびランタイム環境で、活動を停止しているカスタマーについての並列データを保持できます。

delete 操作の変更

このタスクはオプションです。このサンプル・アプリケーションでは、WebSphere Adapter for JDBC は、Oracle から送信されたイベント・データによって非アクティブであると識別される顧客オブジェクトのランタイム環境に、Delete 操作を受け渡すよう設定されています。アダプターが非アクティブな顧客の Delete 操作を受け渡さないようにする場合は、SQL スクリプトを変更して、操作を Delete から Update に変更します。

このタスクを実行する理由および時期

このタスクは、Oracle データベース内で非アクティブになっている顧客レコードに対して Adapter for JDBC が実行する操作を変更する場合にのみ必要です。

Oracle E-Business Suite を使用すると、顧客を非アクティブにしたり、非アクティブな顧客のデータを変更することができますが、ベース・テーブルから顧客データを削除することはできません。Inbound 処理に使用されるストアド・プロシージャ

ーが、Oracle から顧客が非アクティブになったことを示すイベント・データを受信すると、顧客オブジェクトに対する操作が delete に設定され、Adapter for JDBC が使用するイベント・テーブルにその情報が送られます。アダプターはその情報を使用してオブジェクトを処理し、WebSphere Process Server または WebSphere Enterprise Service Bus に受け渡します。ただし、顧客オブジェクトを削除せずにランタイム環境に残しておく場合は、サンプル SQL パッケージを編集して、アダプターがイベント・テーブルから読み取る操作を変更します。

このタスクの手順

1. サンプル・ファイルにアクセスして、ibm_customer_event_pkg.pls ファイルをテキスト・エディターで開きます。
2. 以下のステートメントをスクリプトから除去します。

```
IF v_status <> 'A' THEN
  v_object_function := 'Delete';
END IF;
```
3. スクリプトを保存します。
4. SQL* Plus ツールまたは SQL ステートメントを処理するための類似したツールを使用して、このスクリプトを実行します。

Oracle E-Business Suite への成果物の挿入

4 つの SQL スクリプトを実行し、成果物を Oracle データベース に挿入して、サンプルを実行できるようにします。これらの成果物には、テーブル、ストアド・プロシージャ、イベント・パッケージ、トリガー、およびシーケンスがあります。

このタスクを実行する理由および時期

Inbound 処理では、Oracle E-Business Suite の Business Event System を使用して、顧客テーブルに対する変更を識別したり、変更された顧客の鍵情報を Adapter for JDBC イベント・テーブルに転送したりします。このタスクのスクリプトは、このサンプルを実行するために必要な成果物を Oracle データベースに取り込むために実行する必要があります。

このタスクの手順

サンプル・ディレクトリーから、以下の 4 つの SQL スクリプトをリストされている順序で実行します。

- Ibm_websphere_event_table_create.sql
- Ibm_websphere_events_s.sql
- Ibm_customer_event_key_s.sql
- Ibm_customer_event_pkg.pls

例: SQL>@C:¥ibm¥ibm_websphere_event_table_create.sql;

結果

これらのスクリプトは、ストアド・プロシージャおよびその他の成果物を Oracle Accounts Receivable (AR) スキーマに保管します。

同義語の作成

同義語を作成して、外部サービス・ウィザードで実行されるデータベース検索の効率を高めめます。同義語は、SQL スクリプトを実行して作成します。

このタスクを実行する理由および時期

データベースのサイズ、および Oracle データベースに含まれているデータベース成果物の数によっては、外部サービス・ウィザードを実行してオブジェクトを生成するときに、大規模なリストを検索する処理時間が長くなる場合があります。同義語を作成すると、検索を高速に実行できるようになります。さらに同義語は、Oracle データベース・ドライバーに関連した問題を解決するのに役立ちます。この問題については、IBM テクニカル・ノート 1218775 を参照してください。

このタスクの手順

1. SQL*Plus ツール (または SQL ステートメントを処理するための類似したプログラム) を開きます。
2. スクリプト `ibm_create_synonyms.sql` を実行します。

例えば、次のように入力します。SQL>

```
@C:%samplecontent%ibm_create_synonyms.sql;
```

Oracle E-Business Suite 内部でのセットアップ

このサンプル・アプリケーションでは、定義するイベント、サブスクリプション、およびアクションに基づいてデータを処理する機構である Oracle E-Business Suite Business Event System を使用します。Business Event System を使用するには、その前に Oracle 内部でいくつかのセットアップ作業を完了しておく必要があります。作業の内容は、カスタム・イベントの作成、そのイベントに対するサブスクリプションの作成、およびそのイベントの並行プログラムの作成です。

カスタム・イベントの作成

Oracle E-Business Suite の組み込み Business Event System (BES) には、イベント、サブスクリプション、およびデータを処理するための並行プログラムが必要です。カスタム・イベントを作成することにより、Business Event System メカニズムの 1 つの要件を満たすことができます。

このタスクを実行する理由および時期

これは、Business Event System が、データの変更を検出して Adapter for JDBC イベント・テーブルに受け渡せるようにするために、Oracle E-Business Suite 内で最初に実行するセットアップ・ステップです。

このタスクの手順

1. ユーザー SYSADMIN として、Oracle E-Business Suite アプリケーション・インスタンスにアクセスします。
2. Web アプリケーション・ワークフロー管理者の職責を使用します。
3. 「ビジネス・イベント (Business Events)」メニュー・オプションを選択して、新規ビジネス・イベントを作成できるページにナビゲートします。

- 新規ビジネス・イベントを作成して、以下のすべての情報を入力または選択します。

表 14. ビジネス・イベントの値

フィールド	値
名前	ibm.apps.ar.customer.create
表示名 (Display name)	IBM Create Customer Event
状況 (Status)	有効 (Enabled)
オーナー名 (Owner Name)	JTF
オーナー・タグ (Owner Tag)	JTF

- イベント定義を保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

カスタム・イベントのサブスクリプションの作成

カスタム・イベントの作成後に、そのイベントで使用する Business Event System 用のサブスクリプションを作成する必要があります。

このタスクの手順

- 「サブスクリプション (Subscriptions)」メニューにアクセスします。
- 新規サブスクリプションを作成できるページにナビゲートします。
- 以下の表の情報を入力して、サブスクリプションを作成します。

表 15. サブスクリプションの値

フィールド	値
システム (System)	Oracle インスタンスの名前
ソース・タイプ (Source Type)	ローカル (Local)
イベント・フィルター (Event Filter)	ibm.apps.ar.customer.create
フェーズ (Phase)	10
状況 (Status)	有効 (Enabled)
ルール・データ (Rule Data)	メッセージ (Message)
アクション・グループ (Action Type)	カスタム (Custom)
エラー時 (On Error)	停止してロールバック (Stop and Rollback)

- サブスクリプションの追加情報を入力できる次のセクションにナビゲートして、以下の表に示されている情報を入力します。

表 16. サブスクリプションの追加の値

フィールド	値
PL/SQL ルール関数 (PL/SQL Rule Function)	ibm_websphere_pkg.rule_function
プライオリティー	Normal
オーナー名 (Owner Name)	JTF
オーナー・タグ (Owner Tag)	JTF

- サブスクリプションを保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

並行プログラムの作成

Oracle E-Business Suite Business Event System では、定義するイベント、サブスクリプション、およびアクションに基づいてデータを処理するために、並行プログラムが必要です。カスタム・イベントおよびサブスクリプションを作成した後に、並行プログラムを作成する必要があります。このためには、プログラムの実行可能ファイルを作成し、このファイルにプロパティを定義して、このファイルへのアクセス権を付与します。

並行プログラム実行可能ファイルの作成:

並行プログラムは、Business Event System のアクションを定義するために必要です。並行プログラムを作成するために最初に実行する必要があるのは、プログラム実行可能ファイルを作成する作業です。次に、そのプロパティを定義して、アクセス権を付与します。

始める前に

必ずシステム管理者の職責を使用してください。

このタスクの手順

1. 「並行プログラム (Concurrent Program)」 > 「実行可能ファイル (Executable)」メニュー・オプションを選択します。
2. 以下の情報をフォームに入力するか、選択します。

表 17. 並行プログラム実行可能ファイルの値

フィールド	値
実行可能ファイル (Executable)	IBMCUSTOMEREVENT
ショート・ネーム	IBMCUSTOMEREVENT
アプリケーション	Receivables
実行メソッド (Execution Method)	PL/SQL ストアド・プロシージャ (PL/SQL Stored Procedure)
実行ファイル名 (Execution File Name)	IBM_WEBSPPHERE_PKG_RAISE_ CUSTOMER_EVENT

3. プログラム実行可能ファイル定義を保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

並行プログラムのプロパティの定義:

並行プログラムのプログラム実行可能ファイルの作成後に、並行プログラムのプロパティを定義する必要があります。これは、Business Event System で並行プログラムを使用できるようにするために実行する必要がある 3 つのタスクの 2 番目です。

このタスクの手順

1. 「並行プログラム (Concurrent Program)」 > 「定義 (Define)」メニュー・オプションを選択します。
2. 以下の情報を入力してください。

表 18. 並行プログラム定義の値

フィールド	値
プログラム (Program)	IBM Raise Customer Event
ショート・ネーム	IBMCUSTOMEREVENT
アプリケーション (Applications)	Receivables
説明	IBM Raise Customer Event
実行可能ファイル (Executable)	IBMCUSTOMEREVENT

3. 並行プログラムのプロパティ定義を保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

並行プログラムの使用可能化:

並行プログラムのプログラム実行可能ファイルを作成して、並行プログラムのプロパティを定義したら、実行可能ファイルにアクセス権を付与して、Business Event System のアクションを指示できるようにする必要があります。これは、Business Event System システムで並行プログラムを使用できるようにするために実行する必要があります最後の手順です。

このタスクの手順

1. 「セキュリティ (Security)」 > 「職責 (Responsibility)」 > 「要求 (Request)」メニュー・オプションを選択します。
2. フォームを「照会の入力 (Enter Query)」モードに変更します。
3. グループ・フィールドに「Receivables All」を指定します。
4. 照会を実行します。
5. 「要求 (詳細) (Requests (details))」セクションにナビゲートし、「新規作成 (New)」ボタンをクリックして新規要求を詳細へ追加します。
6. 名前フィールドに「IBM Raise Customer Event」を指定します。
7. 変更を保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

結果

これで、顧客イベントを処理できるよう Business Event System がセットアップされました。

プロジェクトの作成

Oracle E-Business Suite アプリケーションと通信するためのモジュールを作成およびデプロイする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。このウィザードは Adapter for JDBC プロジェクトを作成します。これは、アダプター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

Adapter for JDBC の資料にアクセスできることを確認します。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=//com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

このタスクの手順

1. 「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『デプロイメントのためのモジュールの構成』の情報を参照します。
2. 「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『プロジェクトの作成』に記載されている作業を実行します。このセクションに説明されている手順に従って、ご使用のワークスペースのプロジェクトを WebSphere Integration Developer にセットアップします。

Inbound 処理のモジュールの構成

Oracle E-Business Suite から WebSphere Process Server または WebSphere Enterprise Service Bus に送信されたデータを処理するには、ビジネス・オブジェクトを作成する必要があります。このためには、WebSphere Adapter for JDBC の外部サービス・ウィザードを使用します。最初に、アダプターが特定のデータベースへの通信チャンネルをセットアップできるようにプロパティの値を構成して、ウィザードを初期設定します。次に、ビジネス・オブジェクトをディスカバーするための照会を実行します。最後に、エクスポート・ファイルの接続プロパティの値を指定します。

外部サービス・ウィザードの接続プロパティの設定

ご使用のデータベース・インスタンス用に接続プロパティの値を設定します。これらのプロパティを設定すると、Adapter for JDBC の外部サービス・ウィザードで Oracle データベースに接続して、ディスカバリーを実行したり、サービス記述を作成したりすることができます。

始める前に

このタスクは、『プロジェクトの作成』のタスクで既に開始している外部サービス・ウィザード内で実行します。

このタスクの手順

1. 「必要なファイルおよびライブラリー」ウィンドウで、JDBC ドライバーの JAR ファイルに対応する上部ペインの横にある「追加」をクリックします。外部 JDBC ドライバーの場所を参照し、ドライバーの JAR ファイルを選択して「開く」をクリックします。

ウィザードでプロジェクトに追加する必要がある JDBC ドライバーは、ご使用のデータベースに付属している外部 JAR ファイルであり、データベース管理者またはデータベースの Web サイトから入手できます。

2. 「次へ」をクリックします。

3. 「処理指示」ウィンドウで「**Inbound**」をクリックし、「**次へ**」をクリックします。
4. 「ディスカバリー構成」ウィンドウの左側のペインで、データベース・ベンダー、ドライバー、およびバージョンを選択します。
5. 以下の表に示すようにデータベースの情報を入力します。

以下の表に、サンプルで必要なプロパティとその説明を示します。これらのプロパティの詳細については、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『参照』セクションの『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。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

表 19. 外部サービス・ウィザードの接続プロパティ

プロパティ	説明
データベース	データベース名。Oracle データベースの場合、これはシステム ID (SID) です。データベースは、ホスト名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ホスト名	データベース・サーバーのホスト名または IP アドレス。ホスト名は、データベース名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ポート番号	データベース名およびホスト名とともに、データベースに接続するためのデータベース URL を生成するポート番号。
JDBC ドライバーのクラス名	データベースへの接続に使用される JDBC ドライバーのクラス名
ユーザー名	使用しているデータベース・アカウント。ディスカバリー処理中に使用されます。ウィザードの後の方では、実行時として別の名前およびパスワードを指定できます。
パスワード	使用しているアカウントのパスワード。

「次へ」をクリックします。

結果

ウィザードが、接続プロパティを使用してデータベースに接続します。

ビジネス・オブジェクトの選択およびリンク

この Inbound 処理のサンプルでは、サンプル 1 で Retrieve 操作を実行するために使用したものと同一オブジェクトを選択して同一オブジェクト階層を作成します。ビジネス・オブジェクトとサービスを選択したら、オブジェクトをインポートし、親オブジェクトと子オブジェクトとの間にアプリケーション固有情報 (外部キー関係など) を設定します。

ビジネス・オブジェクトの操作

このサンプルでは、アダプターが Oracle E-Business Suite で変更されるカスタマー・データの Create、Update、および Delete 操作をサポートしています。並行プログラムを実行すると、その最終実行日と、Customer テーブルにある作成日および変更日とが比較されます。次の表では、各条件に割り当てられる操作の内容を示します。

表 20. 操作を割り当てる方法

操作	条件
カスタマー項目に Create 操作が割り当てられる	カスタマー情報の作成日が並行プログラムの最終実行日より後の場合
カスタマー項目に Update 操作が割り当てられる	作成日が並行プログラムの最終実行日より前の場合
カスタマー項目に Delete 操作が割り当てられる	カスタマーの「状況 (Status)」が「活動中 (Active)」(状況の列に「A」がある) ではない場合

売掛金 (AR) スキーマの基本テーブル

次の表に、テーブル名、推奨の同義語、および説明を含む売掛金 (AR) スキーマの基本テーブルを示します。次のセクションでは、Inbound 処理のために構成するビジネス・オブジェクトの関係を表す図を示します。

表 21. AR スキーマの基本テーブル

テーブル名	同義語	説明
AR.HZ_CUST_ACCOUNTS	ArIbm_R_Cust_Accounts	カスタマー番号、タイプ、およびその他のカスタマー情報。
AR.HZ_CUSTOMER_PROFILES	ArIbm_R_Cust_Profiles	カスタマーの概略プロフィール情報。
AR.HZ_PARTIES	ArIbm_R_Parties	カスタマー名。会社ではなく個人の場合は、個人の詳細情報も付記。
AR.HZ_PARTY_SITES	ArIbm_R_Party_Sites	カスタマーの住所 ID と住所のロケーション ID との間のリンク。
AR.HZ_CUST_ACCT_SITES_ALL	ArIbm_R_Acct_Sites	住所 ID。
AR.HZ_CUST_SITE_USES_ALL	ArIbm_R_Site_Uses	カスタマーのビジネス上の住所の目的 (BILL_TO、SHIP_TO など)。
AR.HZ_LOCATIONS	ArIbm_R_Locations	住所の詳細。

表 21. AR スキーマの基本テーブル (続き)

テーブル名	同義語	説明
AR.HZ_CONTACT_POINTS	ArIbm_R_Contacts_Sites	連絡先および電話の詳細情報。 サイトとカスタマー (PARTY_SITES および PARTIES) の両方が連絡先の詳細について同じテーブルを使用するため、このテーブルはリストに 2 回登録されます。
AR.HZ_CONTACT_POINTS	ArIbm_R_Contacts_Customer	連絡先および電話の詳細情報。 サイトとカスタマー (PARTY_SITES および PARTIES) の両方が連絡先の詳細について同じテーブルを使用するため、このテーブルはリストに 2 回登録されます。

ビジネス・オブジェクトの関係

Inbound 処理を正常に実行するため、カスタマー・オブジェクトとその子オブジェクトを接続するリンクは、Oracle データベースでのリンクと一致する必要があります。次の図では、Inbound 処理のために構成する必要があるオブジェクトの関係を示します。

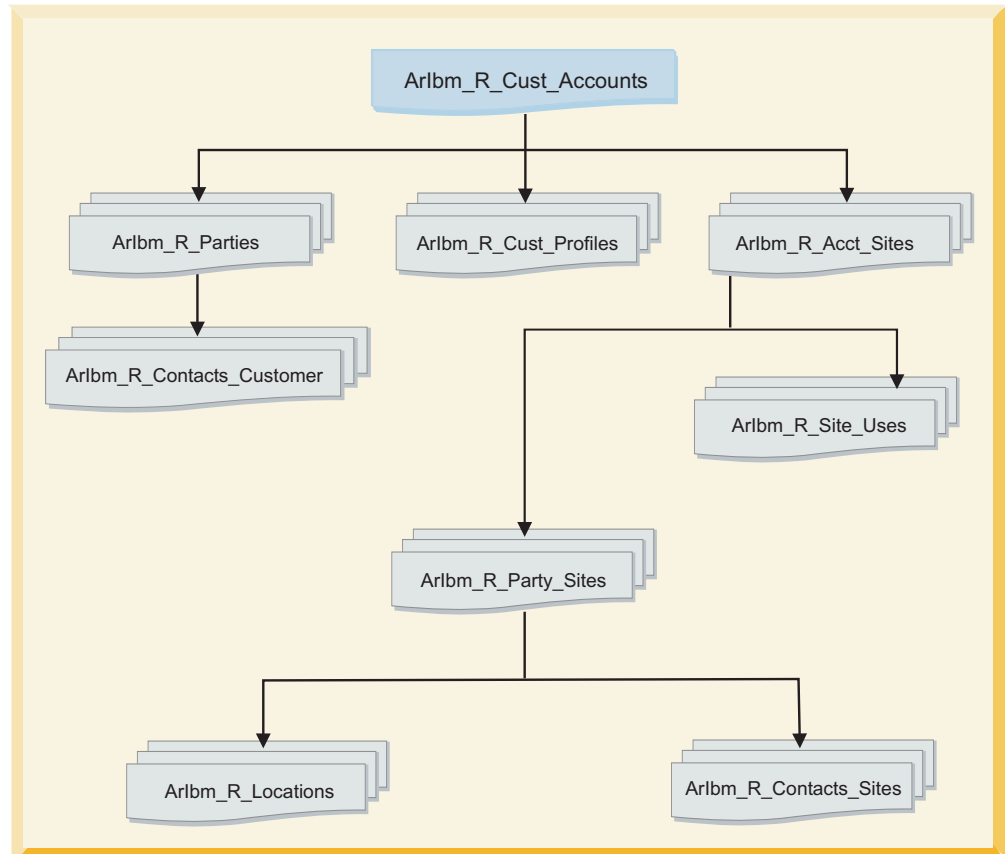


図 16. Inbound 処理のためのオブジェクトの関係

オブジェクトをリンクするには、子オブジェクトをインポートし、親オブジェクトと子オブジェクトとの間に外部キー関係を設定します。ビジネス・オブジェクトの性質について詳しくは、「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『技術的な概説』の『ビジネス・オブジェクト』セクションを参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=//com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

ビジネス・オブジェクトおよびサービスの選択

接続プロパティの構成後に、データベース・オブジェクトに対する照会を実行します。メタデータ・ツリー構造を参照すると、Oracle データベース内のオブジェクトの構造を理解できます。親オブジェクトと子オブジェクトをリンクします。

このタスクを実行する理由および時期

このサンプルでは、Accounts Receivable (AR) スキーマで照会を実行し、Oracle E-Business Suite 基本テーブル内のデータを表す同義語を選択します。

このタスクの手順

1. フィルター・プロパティの指定
 - a. 「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、「照会の編集」をクリックします。

- b. 「照会プロパティ」ウィンドウの「**Schema 名パターン**」フィールドに AR と入力して、Accounts Receivable (AR) スキーマを表示します。
- c. 「**ビジネス・オブジェクトの追加時に追加構成設定のプロンプトを出す**」チェック・ボックスを選択して、「**OK**」をクリックします。

これで、インポート対象のディスカバー済みオブジェクトのいずれかを選択するたびに、そのオブジェクトのアプリケーション固有情報を入力するよう要求されるようになりました。

2. メタデータ照会の実行

- a. 照会によってディスカバーされたオブジェクトを表示します。

「**照会の実行**」をクリックします。AR スキーマおよびスキーマのデータ・エレメントが表示されます。

- b. インポートするオブジェクトの選択

AR スキーマを展開します。「**同義語 - ニックネーム (Synonyms - Nicknames)**」を展開します。

同義語 **IBM_R_CUST_ACCOUNTS** を強調表示して「>」（「追加」アイコン）をクリックし、このオブジェクトをインポート対象として選択します。

- c. ビジネス・オブジェクトのアプリケーション固有情報の追加

「**IBM_R_CUST_ACCOUNTS** の構成プロパティ」ウィンドウで、「**追加**」をクリックして、**IBM_R_CUST_ACCOUNTS** に関連するテーブルの 1 次キーを選択します。「**CUST_ACCOUNT_ID**」を選択して、「**OK**」をクリックします。

- d. インポート用の別のオブジェクトの選択および子オブジェクトの親へのリンク

「オブジェクトのディスカバーおよび選択」ウィンドウで、同義語 **IBM_R_CUST_PROFILES** を強調表示して「>」（「追加」アイコン）をクリックし、このオブジェクトをインポート対象として選択します。

「**IBM_R_CUST_PROFILES** の構成プロパティ」ウィンドウの「**追加**」をクリックして、1 次キーを選択します。**APPLICATION_ID** を選択して、「**OK**」をクリックします。

「**IBM_R_CUST_ACCOUNTS**」を選択して、**IBM_R_CUST_PROFILES** に関連付けられたテーブルの親を設定します。

「子の列ごとに親テーブル列を選択して外部キー関係を作成する」の下で、外部キー属性およびその属性の外部キー値を選択して、子を親の属性にリンクします。

外部キー属性 **CUST_ACCOUNT_ID** の場合は、外部キー属性値 **CUST_ACCOUNT_ID** を選択します。

「**親オブジェクトが子オブジェクトを所有する (カスケード削除)**」にチェック・マークを付け、「**OK**」をクリックします。

Configuration Properties for 'IBM_R_CUST_PROFILES'

Specify configuration properties for the selected object.

Select primary key for table IBM_R_CUST_PROFILES

Select one or more columns:*

APPLICATION_ID

Add...
Remove

Choose parent table from the list for the selected child

Choose parent table : IBM_R_CUST_ACCOUNTS (AR)

Single cardinality

Build a foreign key relationship by selecting a parent table column for each child column

CUST_ACCOUNT_PROFILE_ID: NONE

LAST_UPDATED_BY: NONE

LAST_UPDATE_DATE: NONE

LAST_UPDATE_LOGIN: NONE

CREATED_BY: NONE

CREATION_DATE: NONE

CUST_ACCOUNT_ID: CUST_ACCOUNT_ID

STATUS: NONE

図 17. 構成プロパティの指定

e. 残りの同義語の選択

以下の表に示されている同義語の順序で、残りの同義語のそれぞれについてステップ 2d を繰り返します。同義語ごとに表に示されている 1 次キー、親、外部キー属性、および外部キー属性値を使用します。

表 22. 外部キー値および子オブジェクトを必要とするオブジェクト

同義語	1 次キー	親	外部キー属性	外部キー属性値
ArIbm_R_Parties	party_id	Aribm_R_Cust_Accounts	party_id	party_id
ArIbm_R_Acct_Sites	party_site_id および cust_acct_site_id	Aribm_R_Cust_Accounts	cust_account_id	cust_account_id
ArIbm_R_Party_Sites	location_id および party_site_id	Aribm_R_Acct_Sites	party_site_id	party_site_id
ArIbm_R_Site_Uses	application_id	Aribm_R_Acct_Sites	cust_acct_site_id	cust_acct_site_id
ArIbm_R_Locations	address_key	Aribm_R_Party_Sites	location_id	location_id
ArIbm_R_Contacts_Sites	application_id	Aribm_R_Party_Sites	owner_table_id	party_site_id
ArIbm_R_Contacts_Customer	application_id	Aribm_R_Parties	owner_table_id	party_id

完了したら、「次へ」をクリックします。

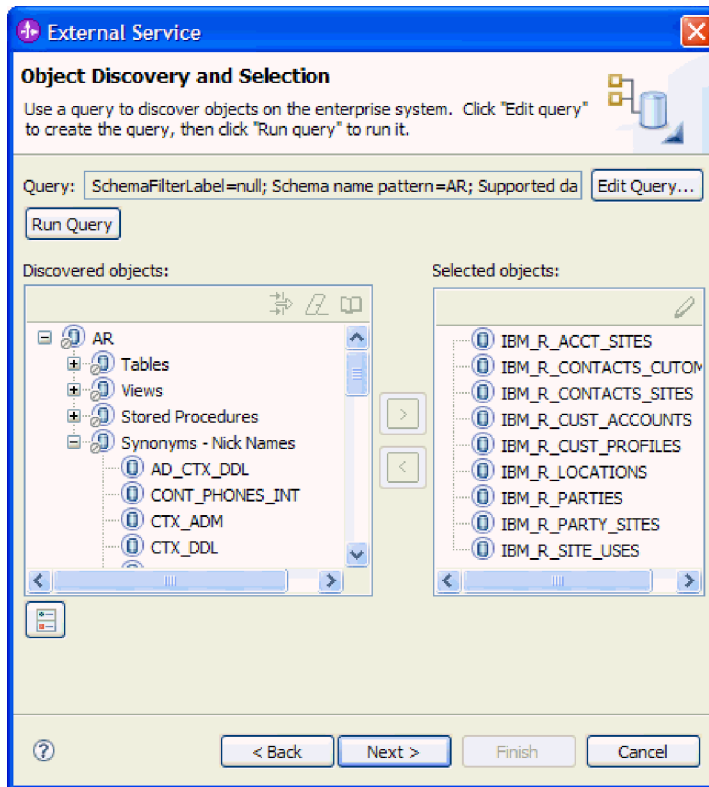


図 18. 残りの同義語の選択

選択済みオブジェクトの構成

データベース・オブジェクトを選択してリンクした後、選択したオブジェクトに適用する操作などのプロパティを指定します。

このタスクの手順

1. 操作を選択します。

外部サービス・ウィザードの「複合プロパティの構成」ウィンドウでは、Inbound サービス・タイプとしてアダプターがサポートしている操作が「操作」ペインに表示されます。操作のリストは変更しないでください。

以下の操作が表示されます。

- **Create**
 - **Update**
 - **Delete**
2. 以下のフィールドのデフォルト値は、そのままにします。
 - ビジネス・オブジェクト **Namespace**
 - フォルダー
 - ビジネス・オブジェクトごとにビジネス・グラフを生成

3. 「次へ」をクリックします。

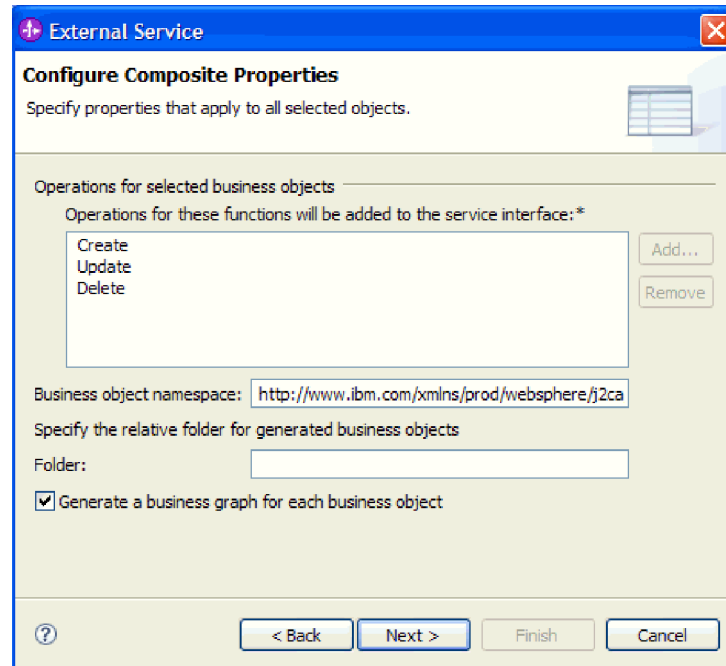


図 19. 選択済みオブジェクトの構成

次のタスク

次に、実行時に使用するデプロイメント情報、およびサービスをモジュールとして保存するための情報を指定します。

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

アダプターが Inbound 処理用の特定のデータベースへの通信チャンネルをセットアップするために使用する、アクティベーション・スペックおよびリソース・アダプター・プロパティを構成します。次に、すべての成果物およびプロパティ値を入れるビジネス・インテグレーション・モジュールを作成します。

このタスクを実行する理由および時期

外部サービス・ウィザードを使用してプロパティを構成します。

このタスクの手順

1. 構成プロパティ値の設定
 - a. 「サービスの生成とデプロイメントの構成」ウィンドウの「コネクター・プロジェクトのデプロイ」フィールドで、オプション「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」が選択されていることを確認します。
 - b. 「拡張」をクリックし、「接続プロパティ」、「データベース・システム接続情報」の下にあるすべてのプロパティを確認します。

Inbound 処理の場合は、アクティベーション・スペック・プロパティーおよびリソース・アダプター・プロパティー用のプロパティー・フィールドが表示されます。

- c. 次の表に示すように必須接続プロパティーの値を設定します。

フィールドは、ウィザードの開始時に指定した接続情報で初期化されます。「データベース・ベンダー」および「ユーザー名」の既存の値は、必要に応じて編集できます。パスワードを入力する必要があります。

表 23. データベース・システム接続情報

プロパティー	値
ユーザー名	使用しているデータベース・アカウント
パスワード	使用しているアカウントのパスワード
データベース・ベンダー	Oracle

- d. 「次へ」をクリックします。

これらのプロパティーについては、「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『参照』セクションを参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

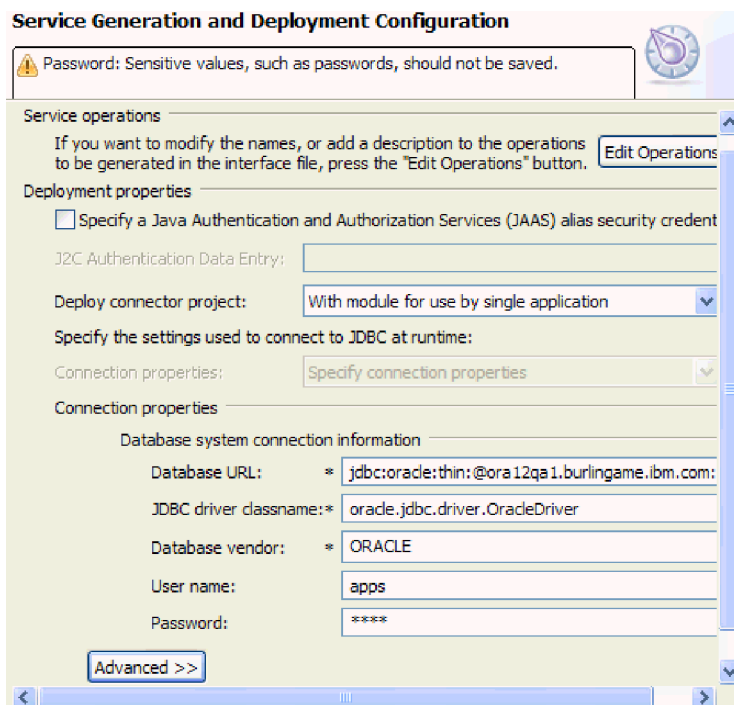


図 20. デプロイメント・プロパティーの設定

2. モジュールの作成

- a. 「サービス・ロケーション・プロパティー」ウィンドウの「モジュール」フィールドの隣にある「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)**」が選択されていることを確認し、「次へ」をクリックします。
- c. 「モジュール」ウィンドウで「Oracle_Inbound」を入力し、「終了」をクリックします。
- d. 「サービス・ロケーション・プロパティ」ウィンドウで、「終了」をクリックします。

結果

生成されるすべての成果物が、このモジュール内で保持されます。

モジュールの保存

作成した成果物およびプロパティ値を入れる Inbound 処理用のビジネス・インテグレーション・モジュールを保存する必要があります。

このタスクの手順

1. 「ビジネス・インテグレーション (Business Integration)」ビューで、「**Oracle_Inbound**」モジュールをクリックします。
2. 「ファイル」 → 「保管」を選択します。

注: 「保管」のオプションが非アクティブ (グレー表示) になっている場合、そのモジュールは既に保存されています。

Inbound コンポーネントの作成

このサンプルはスタンドアロン・アプリケーションであるため、実動アプリケーションのように処理中のエクスポート・ファイルには接続されていません。このため、プロジェクトをデプロイするには、その前に、Inbound イベントを検出し、データを取り出して、取り出したデータを管理コンソールで表示するコンポーネントを作成する必要があります。この管理コンソールでは、データを見直すことができます。

コンポーネントの作成

WebSphere Integration Developer内のアセンブリ・エディターを使用して、新規コンポーネントを作成します。これを実行することで、Inbound サンプル・アプリケーションが、イベントを認識してデータを検索および処理できるエクスポートを持つようにする必要があります。サンプル・アプリケーションがテスト環境にデプロイされるため、この手順が必要になります。

このタスクの手順

1. アセンブリ・ダイアグラムを開く

WebSphere Integration Developer 内の「ビジネス・インテグレーション (Business Integration)」ビューから、「**Oracle_Inbound**」モジュールを展開します。「**アセンブリ・ダイアグラム (Assembly Diagram)**」をダブルクリックします。

これで JDBCInboundInterface コンポーネントのアセンブリ・ダイアグラムが開きます。

2. アセンブリ・ダイアグラムの左側で、「コンポーネント」を展開し、「型なしコンポーネント」をクリックします。

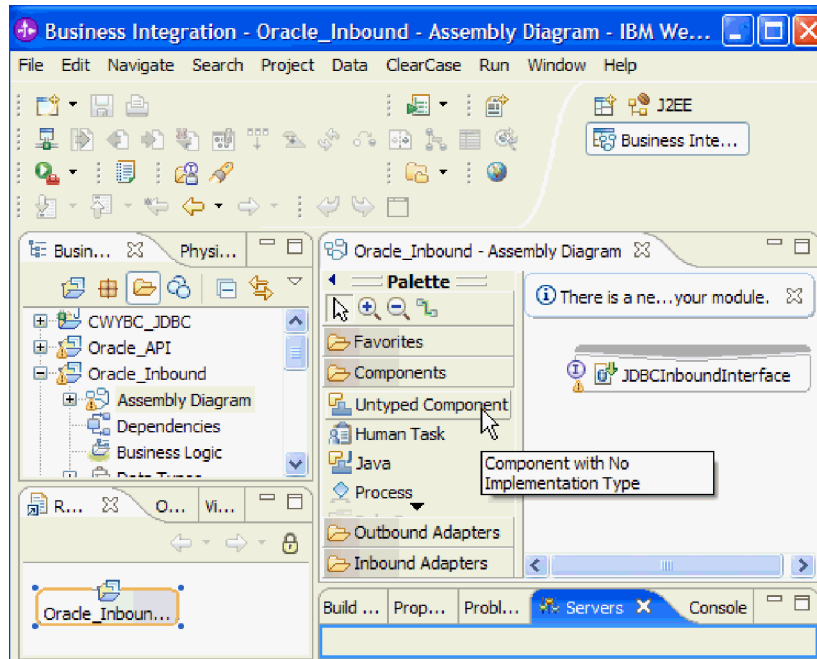


図 21. コンポーネントの作成

3. 新規の型なしコンポーネントを作成するには、アセンブリ・ダイアグラムの背景をクリックします。

新規コンポーネントが「**Component1**」としてアセンブリ・ダイアグラム内に表示されます。

4. 黄色の線が表示されるまで、マウス・ポインターを「**JDBCInboundInterface**」コンポーネントの右側に移動します。線をクリックして、「**Component1**」の左側までドラッグします。

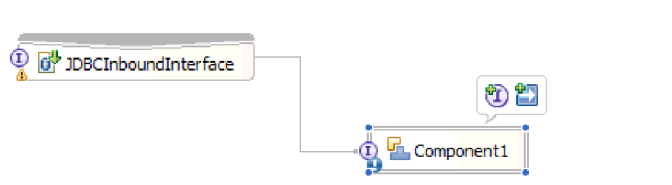


図 22. 線の追加

5. 「線の追加 (Add Wire)」ダイアログ・ボックスで「OK」をクリックします。

次のタスク

次に、新規 Inbound コンポーネントの実装を定義します。

実装の定義

Adapter for JDBC のイベント・ストアに送られたオブジェクトを取り出して管理コンソールにプリントするためのコンポーネントを有効にするには、新規 Inbound コンポーネントの動作を定義するコードを追加します。

このタスクの手順

1. JDBCInboundInterface の「アセンブリー・ダイアグラム (Assembly Diagram)」で「Component1」を右クリックします。
2. 「実装の生成 (Generate Implementation)」 → 「Java」をクリックします。
3. 「実装の生成 (Generate Implementation)」ウィンドウで、「デフォルト・パッケージ (default package)」が選択されていることを確認して「OK」をクリックします。
4. Component1Impl.java ウィンドウが表示され、ここでカスタム・コードを挿入できます。

上部で、import ステートメントの残りの部分に、以下を追加します。

```
import com.ibm.j2ca.base.AdapterBOUtil;
```

実装のデフォルトのセクションを、イベント・ストアに送られたオブジェクトを取り出して管理コンソールにプリントするようにコンポーネントに指示するコードで置き換えます。このユーザーズ・ガイドの次のセクション『Inbound コンポーネントを実装するための Java コード』に記載されているコードを使用してください。

5. カスタム・コードの追加が完了したら、「ファイル」 → 「保管」を選択して、コンポーネント・コードを保存します。Component1Impl.java ウィンドウを閉じます。
6. 「ファイル」 → 「保管」を選択してアセンブリー・ダイアグラムを保管します。

Inbound コンポーネントを実装するための Java コード

カスタムの Java コードは、Inbound コンポーネント内のデフォルトのコードを置き換えます。管理コンソールへの出力を可能にする 3 つのコード・ブロックが提供されています。各コードは、作成済みオブジェクト、更新済みオブジェクト、削除済みオブジェクトに対するものです。

Inbound コンポーネントの適切な空メソッドに以下のコード・ブロックを挿入します。例えば、最初のメソッド・ブロックの場合は以下の行を探します。

```
public void createArIbm_R_Cust_AccountsBG(  
    DataObject createArIbmRCustAccountsBGInput{}
```

次に、空のメソッドを最初のコード・ブロックで置き換えます。

3 つのコード・ブロックをすべて挿入したら、このユーザー・ガイドの前のセクションに戻って、コンポーネント・コードおよびアセンブリー・ダイアグラムを保存します。

作成済みオブジェクトを出力するためのコード

```
public void createArIbm_R_Cust_AccountsBG(
    DataObject createArIbmRCustAccountsBGInput) {
    System.out.println("End point for createArIbm_R_Cust_AccountsBG ");
    int i = createArIbmRCustAccountsBGInput.getType().getProperties().size()-1;
    DataObject dataObj = createArIbmRCustAccountsBGInput.getDataObject(i);
    System.out.println("トレースされるデータ・オブジェクト: " + dataObj.getType().getName());
    try
    {
        String xmlString = AdapterBOUtil.serializeDataObject(dataObj);
        System.out.println(xmlString);
    }
    catch(Exception e)
    {
        System.out.println("データ・オブジェクトのトレース時エラー");
    }
}
```

更新済みオブジェクトを出力するためのコード

```
public void updateArIbm_R_Cust_AccountsBG(
    DataObject updateArIbmRCustAccountsBGInput) {
    System.out.println("End point for updateArIbm_R_Cust_AccountsBG ");
    int i = updateArIbmRCustAccountsBGInput.getType().getProperties().size()-1;
    DataObject dataObj = updateArIbmRCustAccountsBGInput.getDataObject(i);
    System.out.println("トレースされるデータ・オブジェクト: " + dataObj.getType().getName());
    try
    {
        String xmlString = AdapterBOUtil.serializeDataObject(dataObj);
        System.out.println(xmlString);
    }
    catch(Exception e)
    {
        System.out.println("データ・オブジェクトのトレース時エラー");
    }
}
```

削除済みオブジェクトを出力するためのコード

```
public void deleteArIbm_R_Cust_AccountsBG(
    DataObject deleteArIbmRCustAccountsBGInput) {
    System.out.println("End point for deleteArIbm_R_Cust_AccountsBG ");
    int i = deleteArIbmRCustAccountsBGInput.getType().getProperties().size()-1;
    DataObject dataObj = deleteArIbmRCustAccountsBGInput.getDataObject(i);
    System.out.println("トレースされるデータ・オブジェクト: " + dataObj.getType().getName());
    try
    {
        String xmlString = AdapterBOUtil.serializeDataObject(dataObj);
        System.out.println(xmlString);
    }
    catch(Exception e)
    {
        System.out.println("データ・オブジェクトのトレース時エラー");
    }
}
```

重要:

コンポーネント・コードおよびアセンブリー・ダイアグラムを保存する手順については、ユーザー・ガイドの前のセクション『実装の定義』に戻ってください。

テスト用のモジュールのデプロイ

アダプター・アプリケーションをテストするには、Inbound 処理用のモジュールを WebSphere Integration Developer 統合テスト・クライアントにインストールします。

始める前に

外部サービス・ウィザードを実行して、Oracle データベースのエクスポート・ファイルを収録したモジュールを生成します。

このタスクの手順

1. WebSphere Integration Developer で「**サーバー (Servers)**」タブをクリックして、「サーバー (Servers)」ビューを開きます。
2. サーバーが「サーバー」ビューに表示されない場合は、「サーバー」ビューを右クリックして「**新規**」 → 「**サーバー**」を選択します。ご使用の WebSphere Process Server または WebSphere Enterprise Service Bus を選択します。

「**次へ**」をクリックし、「**終了**」をクリックします。
3. WebSphere Process Server または WebSphere Enterprise Service Bus が実行されていない場合は、その名前を右クリックして「**開始 (Start)**」を選択します。「**状況**」が「**始動済み**」に変化します。
4. WebSphere Process Server または WebSphere Enterprise Service Bus の名前を右クリックして、「**プロジェクトの追加および除去**」を選択します。
5. 左側の「**使用可能プロジェクト**」リストから、「**Oracle_InboundApp**」モジュールを選択して、「**追加**」をクリックします。

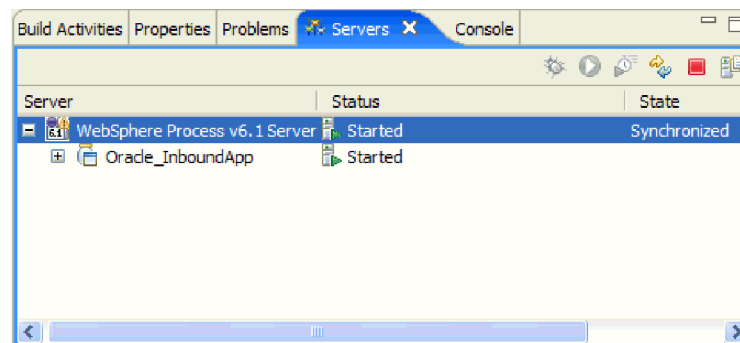


図 23. 「サーバー (Servers)」ビュー

6. 「**終了**」をクリックします。

次のタスク

次に、統合テスト・クライアント内でアセンブルされたアプリケーションをテストできます。

アセンブル済みアダプター・アプリケーションのテスト

生成済みのサービスをテストして、このサービスが予想どおりの結果を示すことを確認します。このサンプルをテストするには、3 つの作業を行います。Oracle Receivables では、カスタマー・レコードを作成します。次に、並行プログラムを

実行して、Create 操作のカスタマー・オブジェクト・イベントを生成します。最後に、WebSphere Integration Developer でテスト・クライアントを使用して、イベントが処理されていることを確認します。

イベントの取得を目的としたモジュール用のテスト・クライアントの開始

WebSphere Integration Developer で、テスト・クライアントを開始し、モジュールのテスト処理を開始して、Inbound イベントが処理および受信されていることを確認します。

このタスクの手順

1. テストするモジュールの選択

WebSphere Integration Developer で、「ビジネス・インテグレーション」パースペクティブの「ビジネス・インテグレーション」ビューに切り替えます。

2. **Oracle_Inbound** モジュールを右クリックして、「テスト」 > 「接続」を選択します。

「Oracle_Inbound_Test」ウィンドウが表示されます。

次のタスク

次に、アSEMBルされたアプリケーションをテストするためのカスタマー・レコードをセットアップします。

アSEMBルされたアダプター・アプリケーションのテストを目的とした顧客の作成

Oracle Receivables 内で顧客レコードをセットアップして、Inbound 処理シナリオをテストします。

始める前に

顧客レコードを作成する前に、テスト・クライアントを開始します。

このタスクの手順

1. Oracle アプリケーション・インスタンスに Oracle E-Business Suite ユーザーとしてログインします。
2. 「Receivables 管理者 (Receivables Manager)」職責を選択します。

この職責が職責リストから選択可能になっていない場合は、ユーザー定義セキュリティで追加する必要があります。

3. 「顧客 (Customers)」->「標準 (Standard)」メニュー・オプションを選択します。
4. 「顧客 (Customer)」ウィンドウで、「作成」ボタンをクリックして新規顧客を作成します。
5. 「組織の作成 (Create Organization)」ウィンドウで、新規顧客レコードの作成に必要なすべての情報を入力するか選択します。
 - 組織名 (顧客名) を入力します
 - 「国 (Country)」の値が定義されていない場合は、選択します

- 「住所 (Address)」、「市区町村 (City)」、および「国 (County)」を入力します
 - 「都道府県 (State)」の値を選択します
 - 「郵便番号 (Postal Code)」を入力します (背景が黄色のフィールドは必須フィールドです)
 - 「目的 (Purpose)」選択項目で「請求先 (Bill TO)」を設定します
6. 「適用」および「保管」をクリックし、顧客の情報を保存します。

注:

Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

次のタスク

次に、顧客イベントを処理するための並行プログラムを実行します。

サンプルのテストを目的とした並行プログラムの実行

Oracle 内で並行プログラムを実行して顧客イベントを処理し、期待どおりに機能していることを確認します。

始める前に

このタスクを実行する前に、Oracle Receivables 内に顧客レコードを作成しておく必要があります。

このタスクの手順

1. 「ホーム」リンクを使用して「売掛金管理者 (Receivables Manager)」職責に戻ります。
2. 「制御: 要求 (Control: Requests)」メニュー・オプション -> 「実行」をクリックします。
3. 「新規要求の実行依頼 (Submit a New Request)」ダイアログ・ボックスが表示されたら、「単一要求 (Single Request)」のボタンが選択されていることを確認して「OK」をクリックします。
4. 「要求の実行依頼 (Submit Request)」ウィンドウで、「名前」フィールド・リストから「IBM Raise Customer Event」を選択して要求を実行依頼します。
5. 別の要求を実行依頼するためのオプションが表示されたら「いいえ」をクリックします。
6. メインメニューから「表示 (View)」 -> 「要求」を選択します。
7. 「要求の検索 (Find Requests)」ウィンドウで、「検索 (Find)」をクリックします。
8. 並行要求が正常状態で完了したら、プロセス・ログを表示できる場所にナビゲートします。

ログ画面に、作成した顧客のイベントの詳細 (顧客 ID やイベント・キーを含む) が表示されます。例えば、次のように表示されます。

カスタマー ID 1234 のイベントを生成中 (Raising event for customer id: 1234)
 イベント・キー: 100000 (Event Key: 100000)

並行要求が正常状況で完了しなかった場合は、ログ・ファイルを参照して、発生したエラーを調べます。例えば、並行プログラムの作成時にスペルを間違えた可能性があります。エラーを解決して並行プログラムを再実行します。エラーについて詳しくは、Oracle Applications の資料を参照してください。

9. テスト・ウィンドウの WebSphere Integration Developer コンソール・ビューにイベントが表示されます。返されたオブジェクトが、顧客レコードを作成したときに入力した顧客情報と一致することを確認します。

サンプル・コンテンツのクリア

アプリケーションのテストが完了したら、サンプル・コンテンツをクリアして、データを元の状態に戻します。Oracle E-Business Suite で顧客を非活動化して、ストアド・プロシージャを削除する SQL コマンドを実行する必要があります。

このタスクの手順

1. 顧客を非活動化します。

Oracle E-Business Suite Client を使用して、売掛金管理者の任務にログインします。「顧客 (Customers)」>「標準 (Standard)」メニューに移動して顧客を検索します。「顧客 (Customer)」ウィンドウの「アクティブ (Active)」ボックスのチェック・マークを外して顧客を保存します。

Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

2. SQL*Plus ツール、または SQL コマンドを処理するための同様なプログラムを使用してデータベースにログインし、コマンドを実行して、データベースから成果物を消去します。このタスクを実行する際に支援が必要な場合、データベース管理者に相談してください。

以下の成果物を除去してください。

- IBM_CUSTOMER_EVENT_KEY_S
- IBM_WEBSPHHERE_PKG
- IBM_WEBSPHHERE_EVENTS

テーブルを除去するには、Drop コマンドを実行します。

サンプル 3: Outbound 処理を行うための Oracle E-Business Suite API の呼び出し

アダプターは、単純なデータ・パラメーターを使用する任意の Oracle E-Business Suite API を直接呼び出すことができます。Outbound 処理を行うために、並行プログラム実行可能ファイルを作成する API を呼び出すには、このサンプルの手順に従います。このシナリオを実行することにより、アダプターと Oracle E-Business Suite データベース間の単純なデータ API 呼び出しを作成する方法が分かります。

関連概念

5 ページの『Outbound 処理のサンプル・アプリケーション』

このソリューションには、Outbound 処理のサンプル・アプリケーションがいくつか用意されています。サンプル 1 では Oracle データベースのインターフェー

ス・テーブルを使用しますが、これは Oracle E-Business Suite の標準 Outbound シナリオになっています。サンプル 3 では API 呼び出しを使用し、サンプル 4 ではワークフローを使用します。

プロジェクトの作成

Oracle E-Business Suite アプリケーションと通信するためのモジュールを作成およびデプロイする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。このウィザードは Adapter for JDBC プロジェクトを作成します。これは、アダプター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

Adapter for JDBC の資料にアクセスできることを確認します。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=//com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

このタスクの手順

1. 「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『デプロイメントのためのモジュールの構成』の情報を参照します。
2. 「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『プロジェクトの作成』に記載されている作業を実行します。このセクションに説明されている手順に従って、ご使用のワークスペースのプロジェクトを WebSphere Integration Developer にセットアップします。

API 呼び出しの Outbound 処理を行うためのモジュールの構成

モジュールを構成するには、ストアード・プロシージャ API 呼び出しを表すビジネス・オブジェクトを作成します。この作業を完了するために必要な手順を実行するには、WebSphere Integration Developer for WebSphere Adapter for JDBC の外部サービス・ウィザードを使用します。

このタスクを実行する理由および時期

最初に、アダプターが特定のデータベースへの通信チャンネルをセットアップできるようにプロパティの値を構成します。次に、ビジネス・オブジェクトをディスカバリーするための照会を実行します。最後に、選択したビジネス・オブジェクトに適用する操作などのプロパティを指定する必要があります。

外部サービス・ウィザードの接続プロパティの設定

ご使用のデータベース・インスタンス用に接続プロパティの値を設定します。これらのプロパティを設定すると、外部サービス・ウィザードで Oracle データベースに接続して、ディスカバリーを実行したり、サービス記述を作成したりすることができます。

始める前に

このタスクは、『プロジェクトの作成』のタスクで既に開始している外部サービス・ウィザードで実行します。

このタスクの手順

1. 「必要なファイルおよびライブラリー」ウィンドウで、JDBC ドライバーの JAR ファイルに対応する上部ペインの横にある「追加」をクリックします。外部 JDBC ドライバーの場所を参照し、ドライバーの JAR ファイルを選択して「開く」をクリックします。

ウィザードでプロジェクトに追加する必要がある JDBC ドライバーは、ご使用のデータベースに付属しているか、データベース管理者またはデータベースの Web サイトから入手できる外部 JAR ファイルです。

2. 「次へ」をクリックします。
3. 「処理指示」ウィンドウで「Outbound」をクリックし、「次へ」をクリックします。
4. 「ディスカバリー構成 (Discover Configuration)」ウィンドウの左側のペインで、データベース・ベンダー、ドライバー、およびバージョンを選択します。
5. 以下の表に示すようにデータベースの情報を入力します。

以下の表に、サンプルで必要なプロパティーとその説明を示します。これらのプロパティーの詳細については、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『参照』セクションの『外部サービス・ウィザードの接続プロパティー』を参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

表 24. 外部サービス・ウィザードの接続プロパティー

プロパティー	説明
データベース	データベース名。Oracle データベースの場合、これはシステム ID (SID) です。データベースは、ホスト名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ホスト名	データベース・サーバーのホスト名または IP アドレス。ホスト名は、データベース名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ポート番号	データベース名およびホスト名とともに、データベースに接続するためのデータベース URL を生成するポート番号。
JDBC ドライバーのクラス名	データベースへの接続に使用される JDBC ドライバーのクラス名
ユーザー名	使用しているデータベース・アカウント。ディスカバリー処理中に使用されます。ウィザードの後のほうでは、実行時として別の名前およびパスワードを指定できます。
パスワード	使用しているアカウントのパスワード。

「次へ」をクリックします。

結果

ウィザードが、接続プロパティーを使用してデータベースに接続します。

ビジネス・オブジェクトおよびサービスの選択

接続プロパティを設定した後は、照会を実行してデータベース・オブジェクトを検索します。ディスカバーされたオブジェクトのメタデータ・ツリーを参照して Oracle データベース 内のオブジェクトの構造を理解したり、サービス記述に必要なオブジェクトを選択したり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

このサンプルでは、アプリケーション (APPS) スキーマに対して照会を実行し、ビジネス・オブジェクトが表すストアード・プロシージャを選択します。

このタスクの手順

1. フィルター・プロパティの指定

「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、「照会の編集」をクリックします。

「照会プロパティ」ウィンドウの「**Schema 名パターン**」フィールドに APPS と入力して、アプリケーション (APPS) スキーマを表示します。

「サポートされるデータベース・オブジェクト・タイプ」フィールドに、テーブル、ビュー、ストアード・プロシージャ、同義語/ニックネームなどの項目がリストされます。「テーブル」、「ビュー (Views)」、および「同義語 - ニックネーム (Synonyms - Nicknames)」のノードを除去して、「OK」をクリックします。

2. メタデータ照会の実行

a. 照会の対象となるオブジェクトの表示

「照会の実行」をクリックします。

APPS スキーマおよびスキーマのデータ・エレメントが表示されます。

b. インポートするオブジェクトの選択

APPS スキーマを展開します。「ストアード・プロシージャ (Stored Procedures)」をクリックします。ただし、展開しないでください。フィルターを使用すると、ストアード・プロシージャを見つけて選択するために役立ちます。

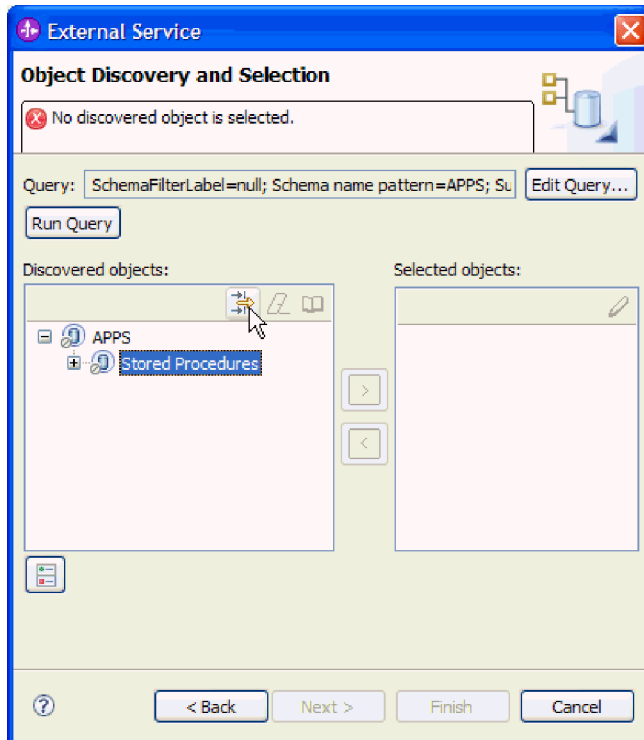


図 24. フィルターを使用したストアード・プロシージャの検索

「検出済みオブジェクト」ペインの「フィルター」アイコンをクリックします。「ストアード・プロシージャのフィルター・プロパティ (Filter Properties for Stored Procedures)」ウィンドウの「オブジェクト名フィルター」フィールドで、EXECUTABLE と入力して「OK」をクリックします。

「ストアード・プロシージャ (Stored Procedures)」ノードを展開します。アダプターが、ストアード・プロシージャのリストを返します。

注: スキーマに多数のストアード・プロシージャが含まれているため、リストにデータが即時に取り込まれない場合があります。

ストアード・プロシージャのリストから **EXECUTABLE (FND_PROGRAM)** を選択します。「>」(追加アイコン) をクリックします。

「EXECUTABLE(FND_PROGRAM) の構成プロパティ」ウィンドウで「OK」をクリックします。ストアード・プロシージャが「選択済みオブジェクト」ペインに追加されます。

「次へ」をクリックします。

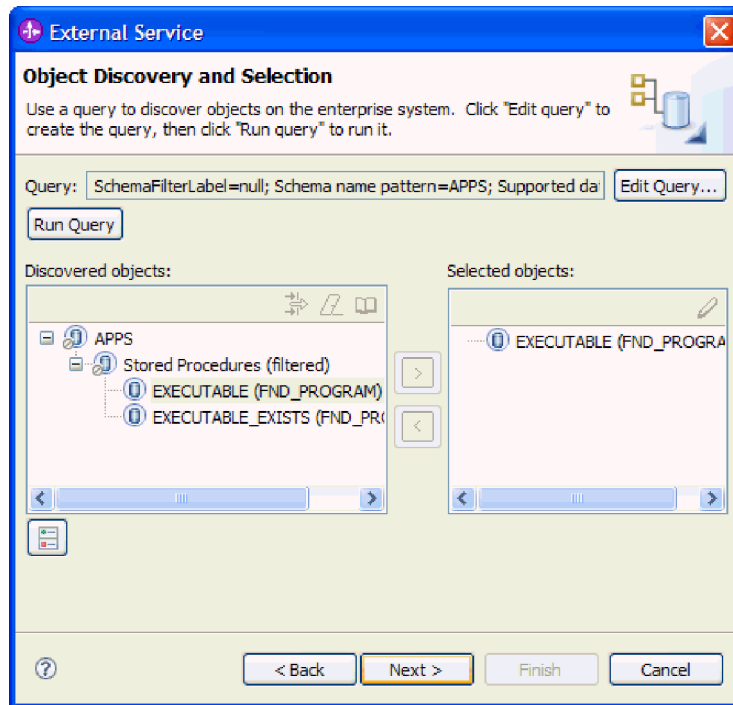


図 25. ストアド・プロシージャの選択

選択済みオブジェクトの構成

データベース・オブジェクトを選択した後、選択したすべてのオブジェクトに適用するプロパティの値を指定します。

このタスクの手順

1. 外部サービス・ウィザードの「複合プロパティの構成」ウィンドウにある以下のフィールドは、デフォルト値のままにしておきます。
 - RetrieveAll 操作の最大レコード数
 - ビジネス・オブジェクト Namespace
 - フォルダー
 - ビジネス・オブジェクトごとにビジネス・グラフを生成
2. 「次へ」をクリックします。

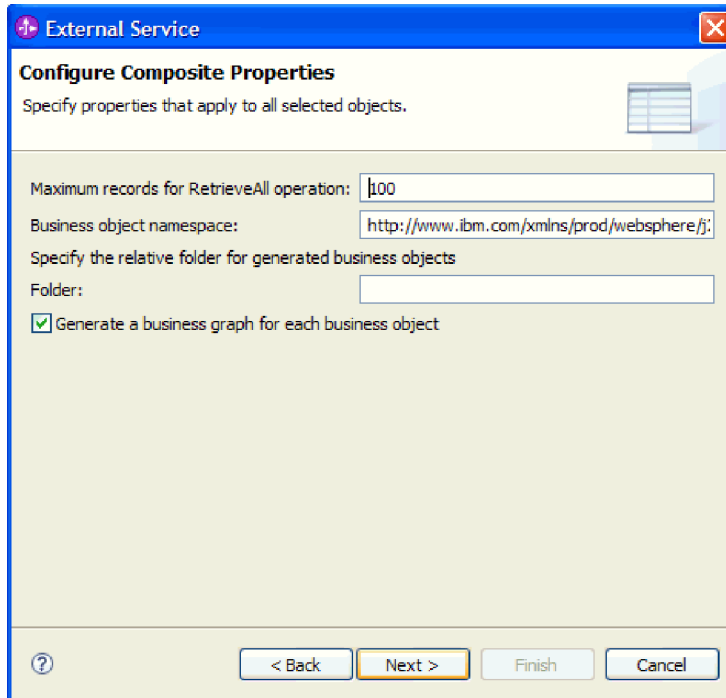


図 26. 複合プロパティの指定

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

外部サービス・ウィザードを使用して、アダプターが Outbound 処理用の特定のデータベースへの通信チャンネルをセットアップするために使用するプロパティを構成します。また、すべての成果物およびプロパティ値を保管できるビジネス・インテグレーション・モジュールを作成する必要があります。

このタスクの手順

1. 構成プロパティ値の設定

- a. 「サービスの生成とデプロイメントの構成」ウィンドウの「コネクター・プロジェクトのデプロイ」フィールドで、オプション「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」が選択されていることを確認します。
- b. 「拡張」をクリックし、「接続プロパティ」、「データベース・システム接続情報」の下にあるすべてのプロパティを確認します。

Outbound 処理の場合は、管理接続ファクトリー・プロパティおよびリソース・アダプター・プロパティ用のプロパティ・フィールドが表示されません。

- c. 次の表に示すように必須接続プロパティの値を設定します。

フィールドは、ウィザードの開始時に指定した接続情報で初期化されます。「データベース・ベンダー」および「ユーザー名」の既存の値は、必要に応じて編集できます。パスワードを入力する必要があります。

表 25. データベース・システム接続情報

プロパティ	値
ユーザー名	使用しているデータベース・アカウント
パスワード	使用しているアカウントのパスワード
データベース・ベンダー	Oracle

d. 「次へ」をクリックします。

これらのプロパティについては、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『参照』セクションを参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=//com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

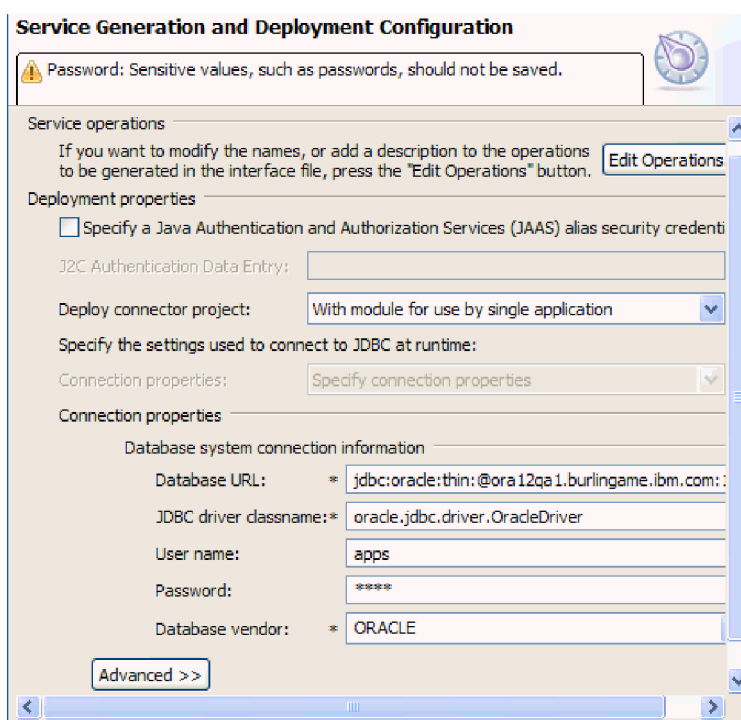


図 27. デプロイメント・プロパティの設定

2. モジュールの作成

- a. 「サービス・ロケーション・プロパティ」ウィンドウの「モジュール」フィールドの隣にある「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)」が選択されていることを確認し、「次へ」をクリックします。
- c. 「モジュール」ウィンドウで「Oracle_API」を入力し、「終了」をクリックします。

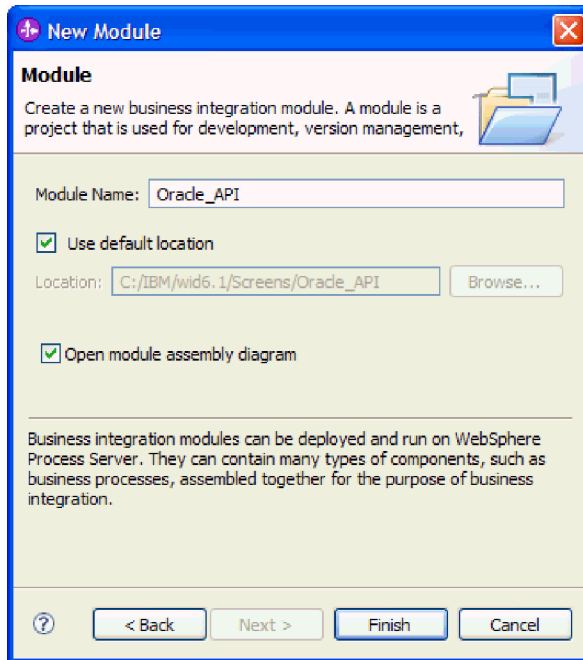


図 28. モジュールの作成

- d. 「サービス・ロケーション・プロパティ」ウィンドウで、「終了」をクリックします。

結果

生成されるすべての成果物が、このモジュール内で保持されます。

モジュールの保存

作成した成果物およびプロパティ値を入れる Outbound 処理用のビジネス・インテグレーション・モジュールを保存する必要があります。

このタスクの手順

1. 「ビジネス・インテグレーション (Business Integration)」ビューで、「Oracle_API」モジュールをクリックします。
2. 「ファイル」 → 「保管」を選択します。

注: 「保管」オプションが非アクティブ (グレー表示) になっている場合、そのモジュールは既に保存されています。

テスト用のモジュールのデプロイ

アダプター・アプリケーションをテストする前に、WebSphere Integration Developer 統合テスト・クライアントでの Outbound 処理用のモジュールをインストールする必要があります。

始める前に

外部サービス・ウィザードを実行して、Oracle データベースのインポート・ファイルを収録したモジュールを生成します。

このタスクの手順

1. WebSphere Integration Developer で「**サーバー (Servers)**」タブをクリックして、「**サーバー (Servers)**」ビューを開きます。
2. サーバーが「**サーバー**」ビューに表示されない場合は、「**サーバー**」ビューを右クリックして「**新規**」 → 「**サーバー**」を選択します。ご使用の WebSphere Process Server または WebSphere Enterprise Service Bus を選択します。

「**次へ**」をクリックし、「**終了**」をクリックします。
3. WebSphere Process Server または WebSphere Enterprise Service Bus が実行されていない場合は、その名前を右クリックして「**開始 (Start)**」を選択します。「**状況**」が「**始動済み**」に変化します。
4. WebSphere Process Server または WebSphere Enterprise Service Bus の名前を右クリックし、「**プロジェクトの追加および除去**」を選択します。
5. 左側の「**使用可能プロジェクト**」リストから、「**Oracle_APIApp**」モジュールを選択して、「**追加**」をクリックします。
6. 「**終了**」をクリックします。

アセンブル済みアダプター・アプリケーションのテスト

生成済みのサービスをテストして、このサービスが予想どおりの結果を示すことを確認します。 WebSphere Integration Developer のテスト・クライアントを使用してテスト値を設定し、サービスを実行します。次に、Oracle E-Business Suite で、サービスが正常に実行されたことを確認します。

WebSphere Integration Developer テスト・クライアントでのテスト

Outbound 処理モジュールのランタイム環境へのデプロイが完了したため、WebSphere Integration Developer 統合テスト・クライアントを使用して、アセンブルされたアプリケーションをテストできます。テスト・クライアントでのテストの完了後に、Oracle E-Business Suite で結果を確認します。

このタスクの手順

1. テストするモジュールの選択

WebSphere Integration Developer で、「**ビジネス・インテグレーション**」パースパクティブの「**ビジネス・インテグレーション**」ビューに切り替えます。

Oracle_API モジュールを右クリックして、「**テスト**」 → 「**モジュールのテスト (Test Module)**」を選択します。

2. 「**verb**」はデフォルト値のままにしておきます。

値は、下記の『**属性値の設定**』の図に示す値と異なる場合があります。

3. テスト値の設定

Oracle_API_Test ウィンドウの矢印を使用して、以下の表に示されているとおりにテスト値を設定します。

表 26. テスト値

項目	テスト値
構成	デフォルト・モジュール・テスト
モジュール	Oracle_API
コンポーネント	JDBCOutboundInterface
インターフェース	JDBCOutboundInterface
操作	executeAppsFnd_ProgramU46ExecutableBG

注: 「コンポーネント」 および「インターフェース」のテスト値には番号が付いていることがあります (JDBCOutboundInterface1 など)。WebSphere Integration Developer は、それぞれの名前がワークスペース内で固有になるように番号を付加します。

4. 入力ビジネス・オブジェクトに、以下の属性値を設定します。

重要: 属性の値を <null> に設定するには、その属性の値フィールドを右クリックします。「値の設定 (Set to)」を選択し、スクロールダウンして「<null>」を選択します。

表 27. 入力ビジネス・オブジェクトの属性値

属性名	値
executable	IBMSAMPLENAME この値は、サンプルを実行するたびに固有でなければなりません。
application	AR
Short_name	IBMSAMPLENAME この値は、サンプルを実行するたびに固有でなければなりません。
Description	並行プログラム実行可能ファイル定義 (Concurrent Program Executable Definition)
Execution_method	PL/SQL ストアード・プロシージャ (PL/SQL Stored Procedure)
Execution_file_name	TEST_PACKAGE.TEST_PROCEDURE
Subroutine_name	<NULL>
Icon_name	<NULL>
Language_code	US
Execution_file_path	<NULL>

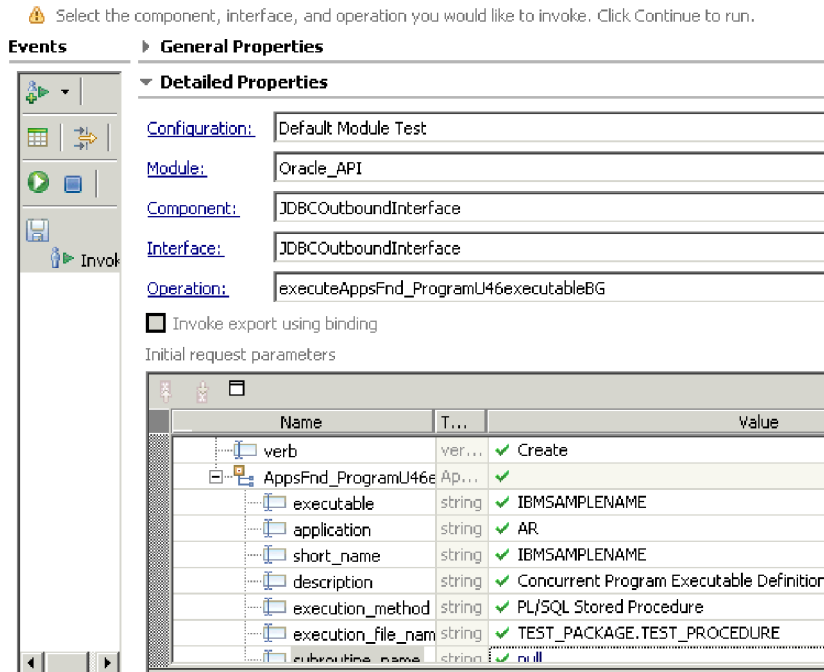


図 29. 属性値の設定

5. 「イベント (Events)」 の下の左側のペインで、「継続 (Continue)」 をクリックします。
6. 「デプロイメント・ロケーション (Deployment Location)」 ウィンドウで、WebSphere Process Server または WebSphere Enterprise Service Bus の名前を選択して、「終了」 をクリックします。「統合クライアントの開始 (Starting the Integration Client)」 ウィンドウが一時的に表示されます。

次のタスク

次に、Oracle E-Business Suite で結果を検証します。

Oracle E-Business Suite でのテスト結果の検証

WebSphere Integration Developer 統合テスト・クライアントでのテストの実行後に、並行プログラムが Oracle E-Business Suite 内に作成されていることを確認します。

このタスクの手順

1. Oracle E-Business Suite にログインして、システム管理の職責を選択します。
2. 「並行プログラム (Concurrent Program)」 メニュー・オプションを選択します。
3. 「並行プログラム (Concurrent Program)」 ウィンドウの「ショート・ネーム (Short Name)」 フィールドに IBMSAMPLENAME と入力します。
4. 照会を実行します。

実行可能プログラムの詳細が表示されます。これらの詳細は、並行プログラムの実行可能ファイルを作成するために指定した入力に一致していなければなりません。

サンプル・コンテンツのクリア

テスト結果のテストおよび検証が完了したら、サンプル・コンテンツをクリアして、データを Oracle E-Business Suite の元の状態に戻します。

このタスクを実行する理由および時期

実行可能プログラムを除去する必要があります。このファイルは、テスト結果を検証した前のセクションと同じ方法で見つけることができます。

このタスクの手順

1. Oracle E-Business Suite にログインして、システム管理の職責を選択します。
2. 「並行プログラム (Concurrent Program)」メニュー・オプションを選択します。
3. 「並行プログラム (Concurrent Program)」ウィンドウの「ショート・ネーム (Short Name)」フィールドに IBMSAMPLENAME と入力します。
4. 照会を実行します。
5. 照会の実行後に、「編集」をクリックしてから「削除」をクリックします。
6. 変更を保存します。

サンプル 4: Business Event System およびワークフローを使用したビジネス・オブジェクトのインポート

Adapter for JDBC は、Outbound 要求を Oracle E-Business Suite に送信して、Oracle のカスタム・イベント・テーブルにカスタマー・データを作成します。そのイベントにより、Oracle Business Event System は、Oracle E-Business Suite にカスタマー・レコードを作成するワークフローを起動します。このワークフローは、この処理中に発生するエラーの処理も行います。

このタスクを実行する理由および時期

Adapter for JDBC は、Outbound 処理時に、Create 操作を使用してカスタマー・データをカスタム・イベント・テーブルに挿入します。アダプターは、Business Event System にカスタム・イベントを作成するストアード・プロシージャ (AfterCreateSP) も呼び出します。Business Event System には、このイベントに対する組み込みのサブスクリプションがあります。Business Event System は、Oracle E-Business Suite にカスタマー・レコードを作成するワークフローを作成します。

関連概念

5 ページの『Outbound 処理のサンプル・アプリケーション』

このソリューションには、Outbound 処理のサンプル・アプリケーションがいくつか用意されています。サンプル 1 では Oracle データベースのインターフェース・テーブルを使用しますが、これは Oracle E-Business Suite の標準 Outbound シナリオになっています。サンプル 3 では API 呼び出しを使用し、サンプル 4 ではワークフローを使用します。

Outbound 処理の準備

Outbound 処理に対応する環境を準備するために必要なセットアップ手順を実行します。ある手順ではストアード・プロシージャを Oracle E-Business Suite に挿入し、別の手順ではカスタマー・イベントを受信するためのカスタム・ワークフローを作成します。

Oracle E-Business Suite への成果物の挿入

4 つの SQL スクリプトを実行し、成果物を Oracle データベース に挿入して、サンプルを実行できるようにします。これらの成果物には、テーブル、ストアード・プロシージャ、イベント・パッケージ、トリガー、およびシーケンスがあります。

このタスクを実行する理由および時期

この Outbound 処理のサンプルでは、Oracle E-Business Suite の Business Event System を使用して、ワークフローを起動するビジネス・イベントを作成します。カスタム・イベント・テーブルが Oracle E-Business Suite に作成された後、Adapter for JDBC を使用してこのカスタム・イベント・テーブルにデータを取り込みます。Adapter for JDBC は、AfterCreateSP ストアード・プロシージャを使用してデータベースにカスタム・イベントを作成し、そのイベントからワークフローを起動します。

このタスクのスクリプトは、このサンプルを実行するために必要な成果物を Oracle データベースに取り込むために実行する必要があります。

このタスクの手順

1. SQL*Plus ツール (または SQL ステートメントを処理するための類似したプログラム) を開きます。 データベース管理者 (DBA) 権限が付与されているデータベース・アカウントを使用して SQL*Plus にログオンします。このサンプルでは、Oracle Applications データベースの APPS スキーマを使用します。
2. サンプル・ディレクトリーから、以下の 4 つの SQL スクリプトをリストされている順序で実行します。

- Ibm_websphere_event_table_create.sql
- Ibm_websphere_events_s.sql
- Ibm_customer_event_key_s.sql
- Ibm_customer_event_pkg.pls

例: SQL>@C:\ibm\ibm_websphere_event_table_create.sql;

結果

これらのスクリプトは、ストアード・プロシージャおよびその他の成果物を Oracle アプリケーション (APPS) スキーマに保管します。

ワークフローの作成

Outbound Create 操作からカスタマー・データを受け取るためのワークフローを Oracle E-Business Suite に作成する必要があります。ワークフローは、データベースにカスタマー・レコードを作成し、エラー処理を行います。

このタスクの手順

1. Oracle Workflow Builder を開き、「ファイル」 > 「開く」メニュー・オプションをクリックします。
2. ワークフロー・スクリプトのファイル名を指定します。samples ディレクトリーを参照して IMPCUST.wft ファイルを探します。
3. 「OK」をクリックし、Workflow Builder にワークフロー・スクリプトをロードします。
4. メニュー・オプション「ファイル」 > 「別名保存」をクリックし、ワークフローをデータベースに保存するための情報を指定します。
 - a. 「データベース (Database)」をクリックし、以下の情報を指定します。これらの情報はデータベース管理者から入手できます。

表 28.

フィールド	値
ユーザー	APPS
パスワード	APPS データベースのユーザーのパスワード
接続 (Connect)	データベース接続ストリング

- b. 「OK」をクリックし、APPS スキーマにワークフローを保存します。

Oracle E-Business Suite 内部でのセットアップ

このサンプル・アプリケーションでは、定義するイベント、サブスクリプション、およびアクションに基づいてデータを処理する機構である Oracle E-Business Suite Business Event System を使用します。Business Event System を使用するには、その前に Oracle 内部でいくつかのセットアップ作業を完了しておく必要があります。カスタム・イベントの作成や、それらのイベントに対するサブスクリプションの作成を含みます。

カスタム・イベントの作成

Oracle E-Business Suite の組み込み Business Event System (BES) には、イベントおよびサブスクリプションが必要です。カスタム・イベントを作成することにより、Business Event System メカニズムの 1 つの要件を満たすことができます。

このタスクを実行する理由および時期

このセットアップ手順を Oracle E-Business Suite 内で実行すると、Business Event System がイベントを検出できるようになります。

このタスクの手順

1. ユーザー SYSADMIN として、Oracle E-Business Suite アプリケーション・インスタンスにアクセスします。
2. Web アプリケーション・ワークフロー管理者の職責を使用します。
3. 「ビジネス・イベント (Business Events)」メニュー・オプションを選択して、新規ビジネス・イベントを作成できるページにナビゲートします。
4. 新規ビジネス・イベントを作成して、以下のすべての情報を入力または選択します。

表 29. ビジネス・イベントの値

フィールド	値
名前	ibm.apps.ar.customer.inbound
表示名 (Display name)	IBM Customer Inbound Event
説明	IBM Customer Inbound Event
状況 (Status)	有効 (Enabled)
オーナー名 (Owner Name)	Oracle E-Business Suite によって認識されているオーナー名
オーナー・タグ (Owner Tag)	Oracle E-Business Suite によって認識されているオーナー・タグ

5. イベント定義を保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

カスタム・イベントのサブスクリプションの作成

カスタム・イベントの作成後に、そのイベントで使用する Business Event System 用のサブスクリプションを作成する必要があります。Outbound 処理中に、Business Event System はイベントを生成し、Oracle イベント・マネージャーに通知します。Oracle イベント・マネージャーは、イベントに関連したサブスクリプションを開始します。サブスクリプションは、実行するアクション (ワークフローを起動してイベントからの情報をテーブルに取り込むなど) を識別します。

このタスクの手順

1. 「サブスクリプション (Subscriptions)」メニューにアクセスします。
2. 新規サブスクリプションを作成できるページにナビゲートします。
3. 以下の表の情報を入力して、サブスクリプションを作成します。

表 30. サブスクリプションの値

フィールド	値
システム (System)	Oracle インスタンスの名前
ソース・タイプ (Source Type)	ローカル (Local)
イベント・フィルター (Event Filter)	ibm.apps.ar.customer.inbound
フェーズ (Phase)	10
状況 (Status)	有効 (Enabled)
ルール・データ (Rule Data)	メッセージ (Message)
アクション・グループ (Action Type)	ワークフローの起動 (Launch Workflow)
エラー時 (On Error)	停止してロールバック (Stop and Rollback)

4. 次のセクションにナビゲートします。このセクションでは、以下の表に示す情報を入力することにより、サブスクリプションの情報を追加できます。

表 31. サブスクリプションの追加の値

フィールド	値
ワークフロー・タイプ (Workflow Type)	IMPCUST

表 31. サブスクリプションの追加の値 (続き)

フィールド	値
ワークフロー・プロセス (Workflow Process)	IMPORT_CUSTOMER
プライオリティー	高
オーナー名 (Owner Name)	Oracle E-Business Suite によって認識されているオーナー名 (カスタム・イベントに使用したものと同名前)
オーナー・タグ (Owner Tag)	Oracle E-Business Suite によって認識されているオーナー・タグ (カスタム・イベントに使用したものと同名タグ)

5. サブスクリプションを保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

エラー・イベントの作成

Oracle E-Business Suite の組み込み Business Event System (BES) には、データを処理するためのイベントおよびサブスクリプションが必要です。このタスクでは、データベースにカスタマー・レコードを作成している間に発生したエラーをワークフローで処理できるようにするためのエラー・イベントを作成します。

このタスクを実行する理由および時期

このセットアップ手順は Oracle E-Business Suite で実行します。実行すると、Business Event System がエラー・イベントを検出できるようになります。

このタスクの手順

1. ユーザー SYSADMIN として、Oracle E-Business Suite アプリケーション・インスタンスにアクセスします。
2. Web アプリケーション・ワークフロー管理者の職責を使用します。
3. 「ビジネス・イベント (Business Events)」メニュー・オプションを選択して、新規ビジネス・イベントを作成できるページにナビゲートします。
4. 新規ビジネス・イベントを作成して、以下のすべての情報を入力または選択します。

表 32. ビジネス・イベントの値

フィールド	値
名前	ibm.apps.ar.customer.error
表示名 (Display name)	IBM Error Event For Customer Import
説明	IBM Error Event For Customer Import
状況 (Status)	有効 (Enabled)
オーナー名 (Owner Name)	Oracle E-Business Suite によって認識されているオーナー名
オーナー・タグ (Owner Tag)	Oracle E-Business Suite によって認識されているオーナー・タグ

5. イベント定義を保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

エラー・イベントのサブスクリプションの作成

エラー・イベントの作成後に、そのイベントで使用する Business Event System 用のサブスクリプションを作成する必要があります。サブスクリプションにより、エラー・イベントの場合に取るアクションが指示されます。

このタスクの手順

1. 「サブスクリプション (Subscriptions)」メニューにアクセスします。
2. 新規サブスクリプションを作成できるページにナビゲートします。
3. 以下の表の情報を入力して、サブスクリプションを作成します。

表 33. サブスクリプションの値

フィールド	値
システム (System)	Oracle インスタンスの名前
ソース・タイプ (Source Type)	ローカル (Local)
イベント・フィルター (Event Filter)	ibm.apps.ar.customer.error
フェーズ (Phase)	10
状況 (Status)	有効 (Enabled)
ルール・データ (Rule Data)	メッセージ (Message)
アクション・グループ (Action Type)	カスタム (Custom)
エラー時 (On Error)	停止してロールバック (Stop and Rollback)

4. サブスクリプションの情報を追加できる次のセクションにナビゲートして、以下の表に示す情報を入力します。

表 34. サブスクリプションの追加の値

フィールド	値
PL/SQL ルール関数 (PL/SQL Rule Function)	IBM_WEBSPPHERE_PKG.RULE_FUNCTION
出力エージェント (Out Agent)	WF_JMS_OUT 注: これはオプションです。イベント情報を JMS キューに取り込む必要がある場合にのみ値を入力します。
プライオリティー	Normal
オーナー名 (Owner Name)	Oracle E-Business Suite によって認識されているオーナー名 (エラー・イベントに使用したものと同名前)
オーナー・タグ (Owner Tag)	Oracle E-Business Suite によって認識されているオーナー・タグ (エラー・イベントに使用したものと同名タグ)

5. サブスクリプションを保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

プロジェクトの作成

Oracle E-Business Suite アプリケーションと通信するためのモジュールを作成およびデプロイする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。このウィザードは Adapter for JDBC プロジェクトを作成します。これは、アダプター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

Adapter for JDBC の資料にアクセスできることを確認します。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=//com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

このタスクの手順

1. 「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『デプロイメントのためのモジュールの構成』の情報を参照します。
2. 「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『プロジェクトの作成』に記載されている作業を実行します。このセクションに説明されている手順に従って、ご使用のワークスペースのプロジェクトを WebSphere Integration Developer にセットアップします。

Outbound 処理のモジュールの構成

データを処理するために必要なビジネス・オブジェクトを作成して、Oracle E-Business Suite に置く必要があります。このためには、WebSphere Adapter for JDBC の WebSphere Integration Developer で外部サービス・ウィザードを使用します。最初に、アダプターが特定のデータベースへの通信チャンネルをセットアップできるようにプロパティの値を構成して、ウィザードを初期設定します。次に、ビジネス・オブジェクトをディスカバリーするための照会を実行します。最後に、選択したビジネス・オブジェクトの操作などのプロパティを指定する必要があります。

外部サービス・ウィザードの接続プロパティの設定

ご使用のデータベース・インスタンス用に接続プロパティの値を設定します。これらのプロパティを設定すると、Adapter for JDBC の外部サービス・ウィザードで Oracle データベースに接続して、ディスカバリーを実行したり、サービス記述を作成したりすることができます。

始める前に

このタスクは、『プロジェクトの作成』のタスクで既に開始している外部サービス・ウィザード内で実行します。

このタスクを実行する理由および時期

まず、アダプターがデータベースと通信できるようにするには、データベースの特定ファイルのコピーが必要です。これらの JAR ファイルには、必要な JDBC ドライバーおよびすべてのネイティブ・システム・ライブラリー・ファイルが含まれて

います。データベース・ソフトウェアおよびオペレーティング・システムに必要なファイルは、データベース管理者またはデータベース・ソフトウェアの Web サイトから入手できます。

このタスクの手順

1. 外部ソフトウェア依存関係を追加します。「必要なファイルおよびライブラリ」ウィンドウで、JDBC ドライバーの JAR ファイルに対応する上部ペインの横にある「追加」をクリックします。外部 JDBC ドライバーの場所を参照し、ドライバーの JAR ファイルを選択して「開く」をクリックします。
2. 「次へ」をクリックします。

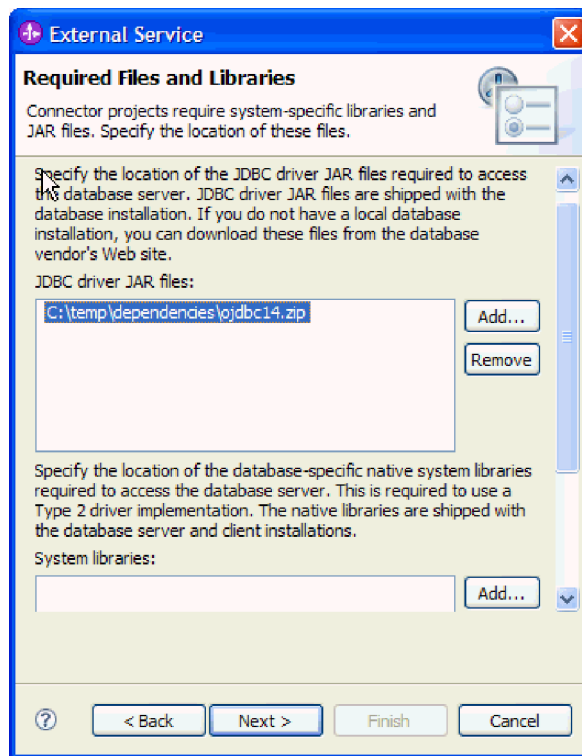


図 30. プロジェクトへの JDBC ドライバー JAR ファイルの追加

3. 「処理指示」ウィンドウで「Outbound」をクリックし、「次へ」をクリックします。
4. 「ディスカバリー構成」ウィンドウの左側のペインで、データベース・ベンダー、ドライバー、およびバージョンを選択します。
5. 以下の表に示すようにデータベースの情報を入力します。

以下の表に、サンプルに必要なプロパティーとその説明を示します。これらのプロパティーについて詳しくは、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『参照』セクションの『外部サービス・ウィザードの接続プロパティー』を参照してください。この資料を参照するには、

http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=//com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

表 35. 外部サービス・ウィザードの接続プロパティ

プロパティ	説明
データベース	データベース名。Oracle データベースの場合、これはシステム ID (SID) です。データベースは、ホスト名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ホスト名	データベース・サーバーのホスト名または IP アドレス。ホスト名は、データベース名およびポート番号とともに、データベースに接続するためのデータベース URL を生成するために使用されます。
ポート番号	データベース名およびホスト名とともに、データベースに接続するためのデータベース URL を生成するポート番号。
JDBC ドライバーのクラス名	データベースへの接続に使用される JDBC ドライバーのクラス名。
ユーザー名	使用しているデータベース・アカウント。ディスカバリー処理中に使用されます。ウィザードの後のほうでは、実行時用として別の名前およびパスワードを指定できます。
パスワード	使用しているアカウントのパスワード。

「次へ」をクリックします。

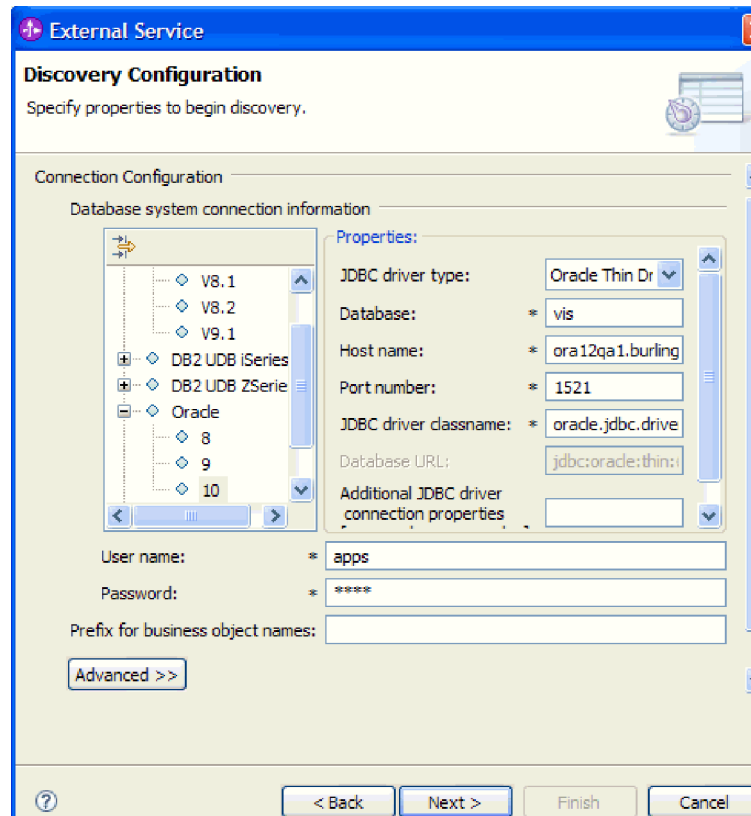


図 31. 接続プロパティ値の追加

結果

ウィザードが、接続プロパティを使用してデータベースに接続します。

ビジネス・オブジェクトおよびサービスの選択

接続プロパティを設定した後は、照会を実行してデータベース・オブジェクトを検索します。ディスカバーしたオブジェクトのメタデータ・ツリーを参照して Oracle データベース内でのオブジェクトの構造を把握し、サービス記述に組み込むオブ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

このサンプルの場合は、売掛金 (AR) スキーマについて照会を実行し、Oracle E-Business Suite インターフェース・テーブルのデータを表すテーブルを選択します。

このタスクの手順

1. フィルター・プロパティを指定します。
 - a. 「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、「照会の編集」をクリックします。
 - b. 「照会プロパティ」ウィンドウの「Schema 名パターン」フィールドに AR と入力して、Accounts Receivable (AR) スキーマを表示します。
 - c. 「ビジネス・オブジェクトの追加時に追加構成設定のプロンプトを出す」チェック・ボックスを選択して、「OK」をクリックします。

これで、インポート対象のディスカバー済みオブジェクトのいずれかを選択するたびに、そのオブジェクトのアプリケーション固有情報を入力するよう要求されるようになりました。

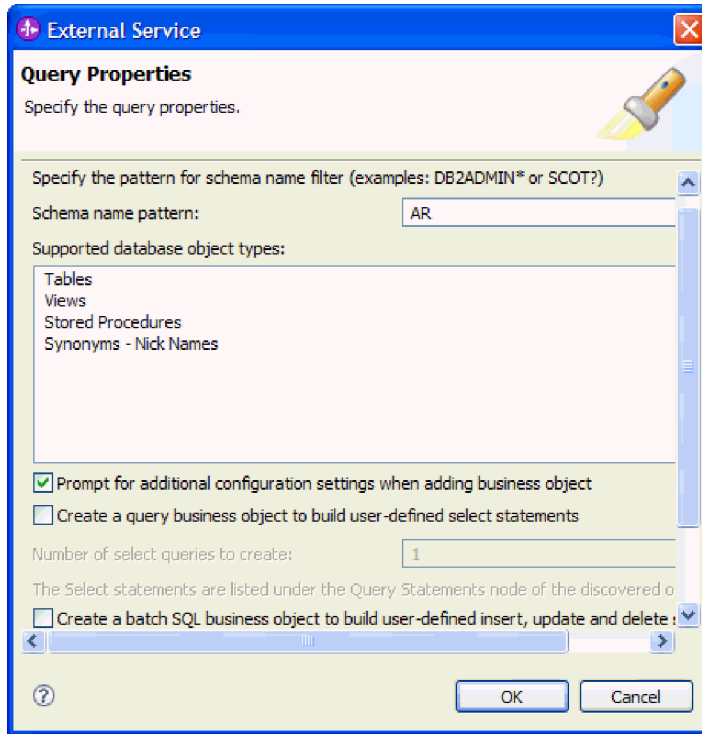


図 32. 照会プロパティの指定

2. メタデータ照会を実行します。

- a. 照会によってディスカバーされたオブジェクトを表示します。

「照会の実行」をクリックします。AR スキーマおよびスキーマのデータ・エレメントが表示されます。

- b. インポートするオブジェクトを選択します。

AR スキーマを展開します。「テーブル」を展開します。

テーブル **RA_CUSTOMERS_INTERFACE_ALL** を強調表示して「>」（「追加」アイコン）をクリックし、このオブジェクトをインポート対象として選択します。

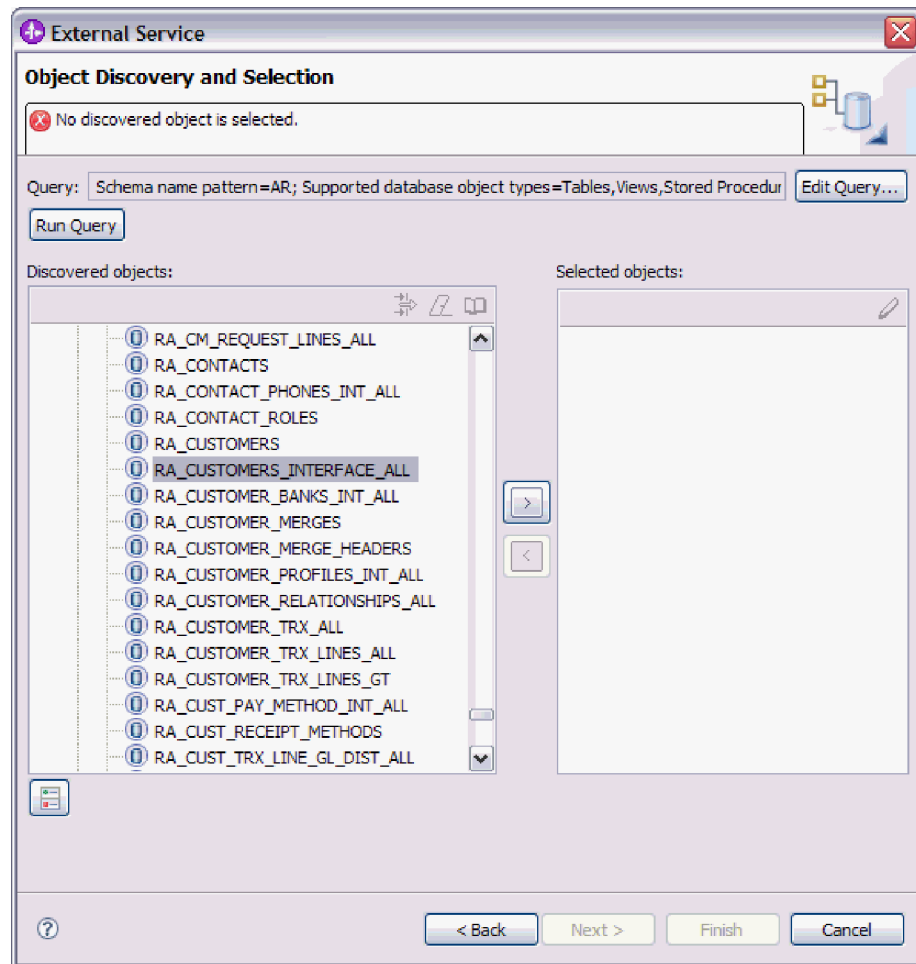


図 33. インポートするオブジェクトの選択

- c. ビジネス・オブジェクト・アプリケーション固有情報を追加します。

「**RA_CUSTOMERS_INTERFACE_ALL** の構成プロパティ」ウィンドウで、「追加」をクリックして、テーブルの 1 次キーを選択します。

REQUEST_ID を選択して、「OK」をクリックします。

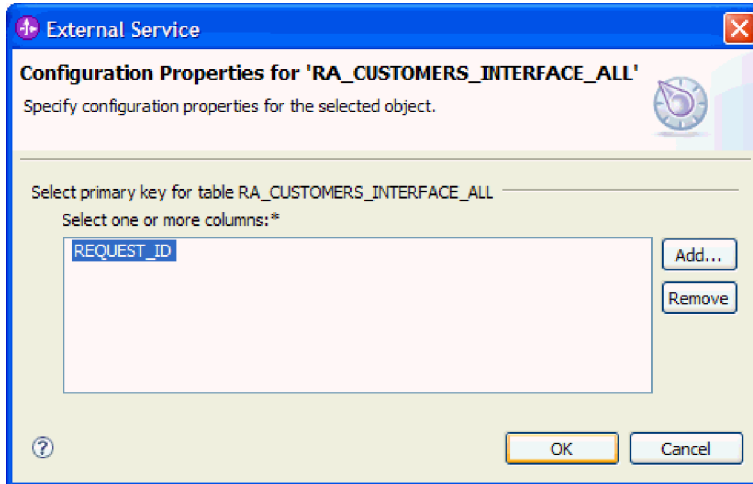


図 34. ビジネス・オブジェクトのアプリケーション固有情報の追加

しばらくすると、選択したオブジェクトが「選択済みオブジェクト」ペインに表示されます。

d. 残りのテーブルの選択

次の表の残りのデータベース・テーブルごとに手順 2b および 2c を繰り返します。各テーブルに 1 次キーが指定されます。

表 36. インポート対象として選択する必要がある残りのテーブルとその 1 次キー

テーブル	1 次キー
AR.RA_CUSTOMER_PROFILES_INT_ALL	REQUEST_ID
AR.RA_CONTACT_PHONES_INT_ALL	REQUEST_ID

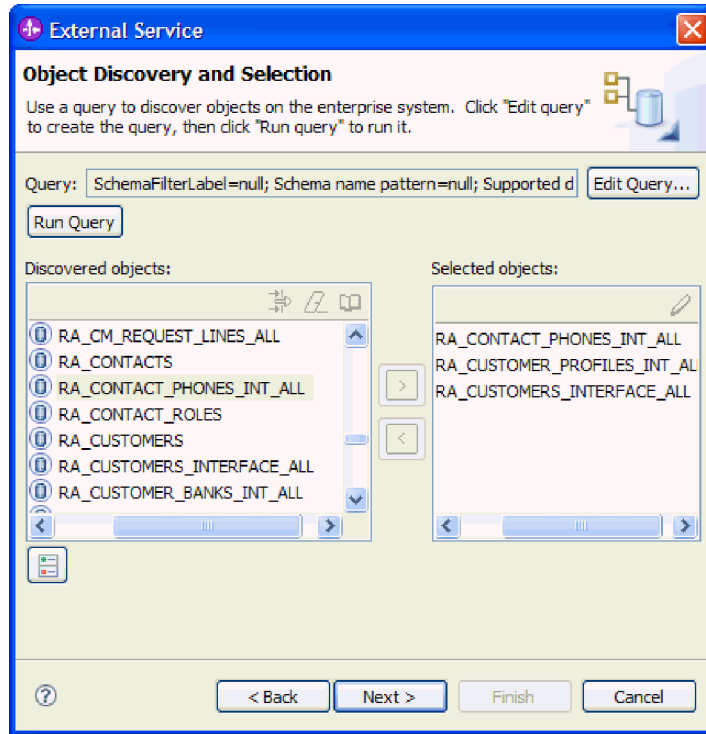


図 35. インポートする残りのオブジェクトの選択

完了したら、「次へ」をクリックします。

オブジェクト・レベル、操作、および属性のアプリケーション固有の情報について詳しくは、「IBM WebSphere Adapter for JDBC ユーザーズ・ガイド」の『参照』セクションの『ビジネス・オブジェクト情報』を参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=/com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

選択済みオブジェクトの構成

データベース・オブジェクトを選択して 1 次キーを追加した後、選択したビジネス・オブジェクトに適用する操作などのプロパティを指定できます。

このタスクの手順

1. 操作を選択します。

外部サービス・ウィザードの「複合プロパティの構成」ウィンドウでは、Outbound サービス・タイプとしてアダプターがサポートしている操作が「操作」ペインに表示されます。次の操作を強調表示させて「除去」をクリックすることにより、除去します。

- Update
- Delete
- RetrieveAll
- ApplyChanges

以下の操作が残ります。

- **Create**
 - **Retrieve**
2. 以下のフィールドのデフォルト値は、そのままにします。
 - **RetrieveAll** 操作の最大レコード数
 - ビジネス・オブジェクト **Namespace**
 - フォルダー
 - ビジネス・オブジェクトごとにビジネス・グラフを生成
 3. 「次へ」をクリックします。

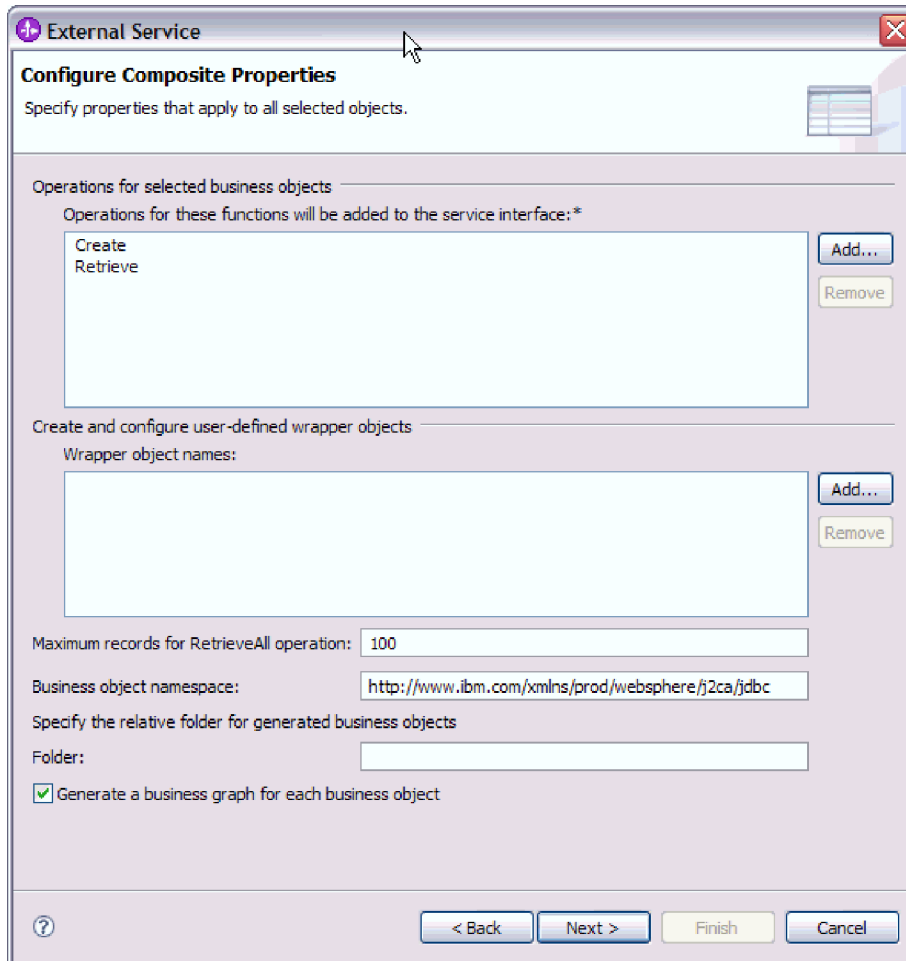


図 36. 操作を選択します。

次のタスク

次に、実行時に使用するデプロイメント情報、およびサービスをモジュールとして保存するための情報を指定します。

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

アダプターが使用するプロパティを構成して、Outbound 処理を行うための特定のデータベースへの通信チャンネルをセットアップするには、外部サービス・ウィザード

ドを使用します。また、すべての成果物およびプロパティ値を保管できるビジネス・インテグレーション・モジュールを作成します。

このタスクの手順

1. 構成プロパティ値を設定する

- a. 「サービスの生成とデプロイメントの構成」ウィンドウの「コネクター・プロジェクトのデプロイ」フィールドで、オプション「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」が選択されていることを確認します。
- b. 「拡張」をクリックし、「接続プロパティ」、「データベース・システム接続情報」の下にあるすべてのプロパティを確認します。

Outbound 処理の場合は、管理接続ファクトリー・プロパティおよびリソース・アダプター・プロパティ用のプロパティ・フィールドが表示されません。

- c. 次の表に示すように必須接続プロパティの値を設定します。

フィールドは、ウィザードの開始時に指定した接続情報で初期化されます。「データベース・ベンダー」および「ユーザー名」の既存の値は、必要に応じて編集できます。パスワードを入力する必要があります。

表 37. データベース・システム接続情報

プロパティ	値
ユーザー名	使用しているデータベースのアカウント
パスワード	使用しているアカウントのパスワード
データベース・ベンダー	Oracle

- d. 「次へ」をクリックします。

これらのプロパティについては、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『参照』セクションを参照してください。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

2. モジュールの作成

- a. 「サービス・ロケーション・プロパティ」ウィンドウの「モジュール」フィールドの隣にある「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)」が選択されていることを確認し、「次へ」をクリックします。
- c. 「モジュール」ウィンドウで TestBESWorkFlow と入力し、「終了」をクリックします。

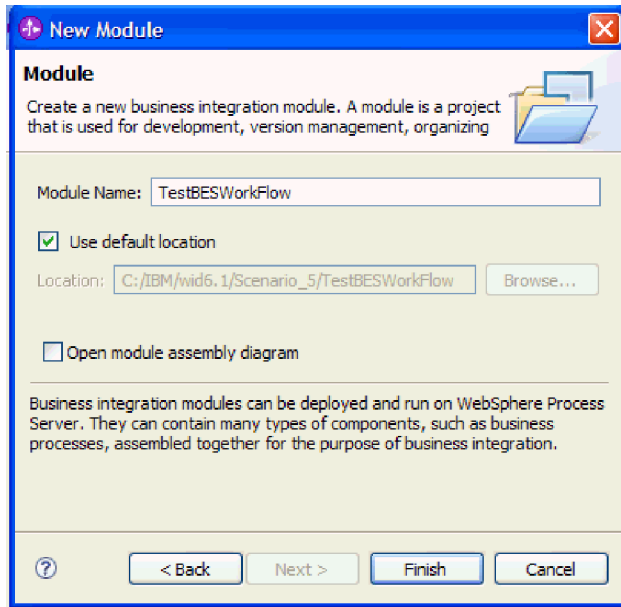


図 37. モジュールの作成

- d. 「サービス・ロケーション・プロパティ」ウィンドウで、「終了」をクリックします。

結果

このモジュールには、生成されたすべての成果物が保持されます。

モジュールの保存

Outbound 処理を行うため、作成した成果物およびプロパティ値が収容されているビジネス・インテグレーション・モジュールを保存します。

このタスクの手順

1. 「ビジネス・インテグレーション」ビューで、**TestBESWorkFlow** モジュールをクリックします。
2. 「ファイル (File)」 > 「保管」を選択します。

注: 「保管」のオプションが非アクティブ (グレー表示) になっている場合、そのモジュールは既に保存されています。

テスト用のモジュールのデプロイ

アダプター・アプリケーションをテストするには、Outbound 処理用のモジュールを WebSphere Integration Developer 統合テスト・クライアントにインストールする必要があります。

始める前に

外部サービス・ウィザードを実行して、Oracle データベースのインポート・ファイルを収録したモジュールを生成します。

このタスクの手順

1. WebSphere Integration Developer で「**サーバー (Servers)**」タブをクリックして、「**サーバー (Servers)**」ビューを開きます。

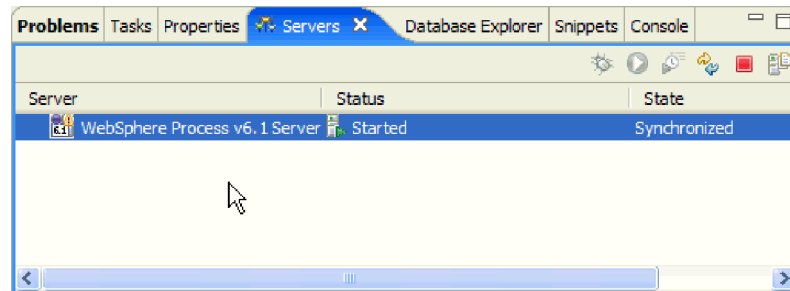


図 38. 「サーバー (Servers)」ビュー

2. サーバーが「サーバー」ビューに表示されない場合は、「サーバー」ビューを右クリックして「**新規**」 → 「**サーバー**」を選択します。ご使用の WebSphere Process Server または WebSphere Enterprise Service Bus を選択します。

「**次へ**」をクリックし、「**終了**」をクリックします。

3. WebSphere Process Server または WebSphere Enterprise Service Bus を実行していない場合は、その名前を右クリックし、「**開始 (Start)**」を選択します。「**状況**」項目が「**始動済み**」に変化します。
4. WebSphere Process Server または WebSphere Enterprise Service Bus の名前を右クリックし、「**プロジェクトの追加および除去**」を選択します。
5. 左側の「**使用可能プロジェクト**」リストから **TestBESWorkFlow** モジュールを選択し、「**追加**」をクリックします。「**終了**」をクリックします。

次のタスク

次に、統合テスト・クライアント内でアセンブルされたアプリケーションをテストできます。

Create 操作のテスト

WebSphere Process Server または WebSphere Enterprise Service Bus への Outbound 処理アプリケーションのデプロイ後に、WebSphere Integration Developer 統合テスト・クライアントを使用して、アセンブルされたアプリケーションをテストします。

このタスクを実行する理由および時期

Create 操作をテストするには、3 つのデータベース・テーブルに顧客データを入力します。

このタスクの手順

1. テストするモジュールの選択

WebSphere Integration Developer で、「**ビジネス・インテグレーション**」パースパクティブの「**ビジネス・インテグレーション**」ビューに切り替えます。

TestBESWorkFlow モジュールを右クリックして、「テスト」 > 「モジュールのテスト (Test Module)」を選択します。

2. 「verb」はデフォルト値のままにしておきます。

値は、下記の『属性値の設定』の図に示す値と異なる場合があります。

3. テスト値の設定

TestBESWorkFlow_Test ウィンドウの矢印を使用して、以下の表に示されているテスト値を設定します。テスト値は、リストした Create 操作ごとに 1 回、合計 3 回設定します。

表 38. Create 操作用のテスト値

項目	テスト値
構成	デフォルト・モジュール・テスト
モジュール	TestBESWorkFlow
コンポーネント	JDBCOutboundInterface
インターフェース	JDBCOutboundInterface
操作 (一度に 1 つの操作を選択します)	createArRa_Customers_Interface_AllBG createArRa_Customer_Profiles_Int_AllBG createArRa_Contact_Phones_Int_AllBG

注: 「コンポーネント」 および「インターフェース」のテスト値には番号が付いていることがあります (JDBCOutboundInterface1 など)。WebSphere Integration Developer は、それぞれの名前がワークスペース内で固有になるように番号を付加します。

4. 入力ビジネス・オブジェクトの属性値の設定

外部サービス・ウィザードでのオブジェクト・ディスカバリー中に選択した 3 つのテーブルの属性値を設定します。テーブルは以下のとおりです。

- ArRa_Customers_Interface_AllBG
- ArRa_Customer_Profiles_Int_AllBG
- ArRa_Contact_Phones_Int_AllBG

以下の表では、各オブジェクトの値と制約事項 (値が固有でなければならないかどうかなど) を示しています。

- a. 固有の ID 値を作成して記録します。

Oracle では、Outbound 処理用の ID 値が必要です。これらの値により、Oracle E-Business Suite に挿入されるレコードが一意的に識別されます。これらの値は、対応する挿入済みデータを正常に取得するために必要です。ID 値は複数のオブジェクトで使用されるため、さまざまなオブジェクトで繰り返し入力するには、これらの値を書き留めておく必要があります。以下の値のそれぞれについて、固有のストリング値を作成してください。

- CustomerReference#
- AddressReference#1
- TelephoneReference#
- CustomerName

注: Oracle インターフェース・テーブル内のデータの必須の値および他の指定可能な値について詳しくは、Oracle アプリケーションの資料 (特に「Oracle Receivables ユーザーズ・ガイド」) を参照してください。

- b. 「イベント」ウィンドウの「初期要求パラメーター」の下で、値を設定する必要がある各属性の名前の向かい側にある「値」列をクリックして、以下の表の値を入力します。

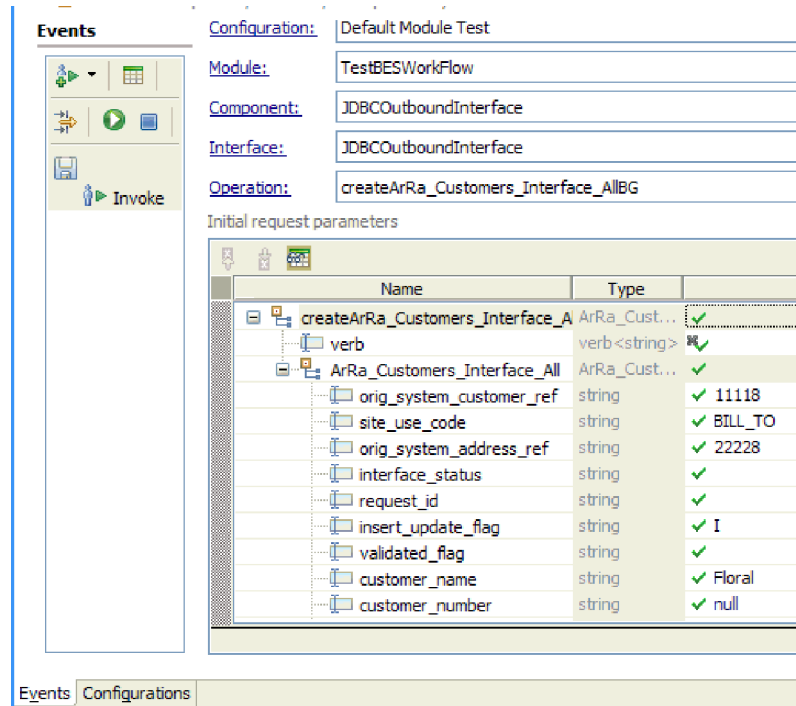


図 39. 属性値の設定

重要:

属性の値を <null> に設定するには、その属性の値フィールドを右クリックします。「値の設定 (Set to)」を選択し、スクロールダウンして「<null>」を選択します。

ArRa_Customers_Interface_All テーブルの属性の値を以下の表に示します。

表 39. ArRa_Customers_Interface_All テーブルの属性の値

属性	値	注
Orig_system_customer_ref	CustomerReference#	記録した値を使用してください。値は、固有でなければなりません。
Site_use_code	BILL_TO	
Orig_system_address_ref	AddressReference#1	記録した値を使用してください。値は、固有でなければなりません。
Insert_update_flag	I	

表 39. ArRa_Customers_Interface_All テーブルの属性の値 (続き)

属性	値	注
Customer_name	CustomerName	記録した値を使用してください。値は、固有でなければなりません。
Customer_number	<NULL>	
Customer_status	A	
Primary_site_use_flag	Y	
Location	<NULL>	
Address1	Test Address	独自のアドレス情報を使用することもできます。
Address2	<NULL>	
Address3	<NULL>	
Address4	<NULL>	
City	San Mateo	
State	CA	
Province	<NULL>	
County	San Mateo	
Postal_code	94401	
country	US	
Cust_category_code	CUSTOMER	
Last_updated_by	-1	
Last_update_date	2007-01-08 12:00:00	今日の日時。時刻は正確でなくても構いません。日付形式には yyyy-mm-dd hh:mm:ss を使用します
Created_by	-1	
Creation_date	2007-01-08 12:00:00	今日の日時。時刻は正確でなくても構いません。日付形式には yyyy-mm-dd hh:mm:ss を使用します
Org_id	204	
Customer_name_phonetic	CustomerName	記録した値を使用してください。

以下の表に、ArRa_Customer_Profiles_Int_All テーブル属性の値をリストします。

表 40. ArRa_Customer_Profiles_Int_All テーブル属性の値

属性	値	注
Orig_system_customer_ref	CustomerReference#	記録した値を使用してください。値は、固有でなければなりません。
Insert_update_flag	I	

表 40. ArRa_Customer_Profiles_Int_All テーブル属性の値 (続き)

属性	値	注
Customer_profile_class_name	DEFAULT	
Credit_hold	N	
Last_updated_by	-1	
Last_update_date	2007-01-08 12:00:00	今日の日時。時刻は正確でなくても構いません。日付形式には yyyy-mm-dd hh:mm:ss を使用します
Created_by	-1	
Creation_date	2007-01-08 12:00:00	今日の日時。時刻は正確でなくても構いません。日付形式には yyyy-mm-dd hh:mm:ss を使用します
Org_id	204	

以下の表に、ArRa_Contact_Phones_Int_All テーブル属性の値をリストします。

表 41. ArRa_Contact_Phones_Int_All テーブル属性の値

属性	値	注
Orig_system_contact_ref	<NULL>	
Orig_system_telephone_ref	<i>TelephoneReference#</i>	記録した値を使用してください。値は、固有でなければなりません。
Orig_system_customer_ref	<i>CustomerReference#</i>	記録した値を使用してください。値は、固有でなければなりません。
Orig_system_address_ref	<i>AddressReference#1</i>	記録した値を使用してください。値は、固有でなければなりません。
Insert_update_flag	I	
Contact_first_name	<NULL>	
Contact_last_name	<NULL>	
Contact_title	<NULL>	Contact_title は ar_lookups に存在する必要があります。ここで、lookup_type=Contact_title です
Contact_job_title	<NULL>	
Telephone	4555555	独自の電話番号を使用することもできます。
Telephone_extension	89555	ない場合は <null> を使用してください。

表 41. ArRa_Contact_Phones_Int_All テーブル属性の値 (続き)

属性	値	注
Telephone_type	FAX	Telephone_type は ar_lookups に存在する必要があります。ここで、lookup_type=Phone_line_type です
Telephone_area_code	407	
Last_update_date	2007-01-08 12:00:00	システムの日時を使用します。日付形式には yyyy-mm-dd hh:mm:ss を使用します
Last_updated_by	-1	
Creation_date	2007-01-08 12:00:00	システムの日時を使用します。日付形式には yyyy-mm-dd hh:mm:ss を使用します
Created_by	-1	
Email_address	<NULL>	
Org_id	204	

5. 「イベント (Events)」の下の左側のペインで、「**継続 (Continue)**」をクリックします。「デプロイメント・ロケーション (Deployment Location)」ウィンドウで、アプリケーションのデプロイ先のサーバーの名前を選択して「**終了**」をクリックします。「統合テスト・クライアントの開始 (Starting the Integration Test Client)」ウィンドウが一時的に表示されます。
6. SQL*Plus ツール、または SQL コマンドを処理するための類似したプログラムを開始します。raise_inbound_event.sql スクリプトを実行します。

例: SQL>@c:\ibm\raise_inbound_event.sql;

スクリプトは値「Customer Reference」を受け入れます。

7. 顧客データをインポートするワークフローの結果を確認します。
 - a. システム管理者ユーザーとして Oracle E-Business Suite にログオンし、Web アプリケーションのワークフロー管理者の職責を選択します。
 - b. 「管理者ワークフロー (Administrator Workflow)」 > 「状況モニター (Status Monitor)」メニュー・オプションを選択します。
 - c. 「内部名の入力 (Type Internal Name)」フィールドに IMPCUST と入力して「実行 (Go)」をクリックします。
 - d. ワークフローの結果を表示する画面で、先ほど実行したワークフロー・プロセスを選択し、「状況ダイアグラム (Status Diagram)」ビューを選択します。
 - e. Java アプレット・ウィンドウにワークフロー・ダイアグラムが表示され、完了パスが示されます。顧客オブジェクトが正常にインポートされた場合は、売掛金管理者の職責に切り替えて「顧客 (Customer)」 > 「標準 (Standard)」メニュー・オプションを選択することで、Oracle E-Business Suite の顧客データを確認します。

- f. インターフェース・プログラムでエラーが発生して終了した場合、またはデータ・エラーのために顧客オブジェクトをインポートできない場合には、ワークフロー・プロセスがエラー・イベントを生成し、インポート・エラー・イベントのダイアグラムを表示します。インターフェースで顧客オブジェクトが失敗した場合は、以下の SQL ステートメントを実行して、イベント・テーブルのエラー・イベント・レコードを確認します。

```
select * from ibm_websphere_events where object_key = <customer  
reference> and object_function = 'Error';
```

次のタスク

次に、サンプルの内容をクリアしてデータを元の状態に戻します。

サンプル・コンテンツのクリア

カスタマー・オブジェクトをインポートするためのワークフローのテストが完了したら、サンプル・コンテンツをクリアして、データを元の状態に戻します。このためには、Oracle E-Business Suite でカスタマーを非活動状態にしてから SQL コマンドを実行して、ストアード・プロシージャを削除します。

このタスクの手順

1. Oracle E-Business Suite Client を使用して、売掛金管理者の任務にログインします。「カスタマー (Customers)」>「標準 (Standard)」メニューに移動して、カスタマーを探します。カスタマーの「状況」の値を「活動停止 (Inactive)」に変更します。その後、そのカスタマーを保存します。

注: Oracle インターフェースの具体的な使用方法については、ご使用の Oracle アプリケーションの資料を参照してください。

2. SQL*Plus ツール、または SQL コマンドを処理するための同様なプログラムを使用してデータベースにログインし、コマンドを実行して、データベースから成果物を消去します。このタスクを実行する際に支援が必要な場合、データベース管理者に相談してください。

サンプル・アプリケーションのトラブルシューティング情報

これらのサンプルについてタスクの実行中に問題が発生した場合、問題は Adapter for JDBC の実行または Oracle データベースとの対話に関連する可能性があります。2 つの既知の問題について説明します。

オブジェクトの生成に時間がかかる

オブジェクトを生成するための所要時間は、Oracle E-Business Suite を実行しているハードウェア・システムのコンピューティング処理能力に応じて変動します。

Create 操作中にオブジェクトが基本テーブルに移動しない

Create 操作中は、アダプターがインターフェース・テーブルに値を設定した後に、ストアード・プロシージャによって Oracle 関数呼び出しが行われます。この関数呼び出しにより、Oracle インターフェース・テーブルから基本テーブルヘデータが

移動します。この関数呼び出しが行われたときに問題が発生すると、Oracle では、INTERFACE_STATUS と呼ばれるインターフェース・テーブル内の列にエラー・コードが書き込まれます。

Create 操作の実行後、カスタマー・オブジェクトがアプリケーション内に表示されないにもかかわらず、Adapter for JDBC からはエラーが通知されない場合は、各インターフェース・テーブルの INTERFACE_STATUS 列を照会して、エラーが通知されていないか確認してください。これらのエラーの原因は、必須の値が設定されていないためか、Oracle アプリケーションの基本テーブルに既に存在する参照番号またはカスタマー名を使用したためと考えられます。

注: この列のエラー・コードについては、Oracle Applications の資料を参照してください。

実稼働環境では、これらのエラーはいくつかの方法で検出できます。別の Adapter for JDBC インスタンスを実行すると、これらの列からエラーを検出して通知できます。あるいは、Oracle ワークフローを作成してエラーの有無を確認し、Oracle E-Business Suite のレポート・システムを使用してエラーを通知することもできます。

Adapter for JDBC の実行中に発生する問題については、「*IBM WebSphere Adapter for JDBC ユーザーズ・ガイド*」の『トラブルシューティングとサポート』セクションを参照してください。ロギングおよびトレースの構成方法、エラーおよび障害を検出および解決する方法、IBM ソフトウェア・サポート・リソースの使用方法が記載されています。この資料を参照するには、http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/index.jsp?topic=/com.ibm.wsadapters.610.doc/doc/stbp_jdb_welcome.html にアクセスしてください。

サンプル・アダプター成果物の表示

各サンプルのサンプル成果物を表示するには、アダプターに同梱される参照ファイルを IBM WebSphere Integration Developer にインポートします。これらの成果物は、参照専用であることに注意してください。多くの場合、ご使用の Oracle 環境では動作しません。サンプルを一通り完了していない場合でも、自分で成果物を作成する前に、参照ファイルを使用して、正しく生成された成果物の例を表示できます。

始める前に

参照ファイルを samples ディレクトリーの referencefiles サブディレクトリーに配置します。各サンプルには、プロジェクト交換 zip ファイルがあります。例えば、サンプル 1 には、Tutorial1.zip があります。

重要: 参照ファイルで提供される成果物は、変更または使用しないでください。これらは、表示専用で提供されています。

参照ファイルには、サード・パーティーのライブラリーは含まれていません。参照ファイルが IBM WebSphere Integration Developer にインポートされると、従属するライブラリーがないという理由でコンパイル・エラー・メッセージが生成されることがあります。

参照ファイル内の成果物は、バージョンや構成によっては、使用している Oracle E-Business Suite とは互換性がない場合があります。これらの成果物は、Oracle E-Business Suite のバージョン 12 を使用して生成されました。

このタスクを実行する理由および時期

参照ファイルを WebSphere Integration Developer にインポートして、各サンプルに関連付けられているサンプル成果物を表示します。

このタスクの手順

1. WebSphere Integration Developer の「ビジネス・インテグレーション (Business Integration)」パースペクティブで、「ファイル (File)」→「インポート」をクリックします。
2. 「インポート」ウィンドウで「プロジェクト交換 (Project Interchange)」を選択し、「次へ」をクリックします。
3. 表示したいチュートリアル成果物を含むプロジェクト交換ファイルを選択します。
4. 「すべて選択」をクリックして、プロジェクト交換ファイルのすべてのプロジェクトをインポートします。
5. 「終了」をクリックします。

結果

ビジネス・インテグレーション・モジュールが、以下の成果物とともに作成されます。

- サービスのインポートおよびエクスポート定義
- ビジネス・オブジェクト (サービス・データ・オブジェクト)
- インターフェース。

第 4 章 参照情報

関連製品の情報を調べるには、WebSphere Adapter for Oracle E-Business Suite の参照情報を利用してください。

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 Oracle E-Business Suite 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

WebSphere Integration Developer のオンライン・サンプル/チュートリアル・ギャラリーには、WebSphere Adapters を使用するのに役立つサンプルおよびチュートリアルが置かれています。オンライン・サンプル/チュートリアル・ギャラリーへのアクセス先のページは、以下のとおりです。

- WebSphere Integration Developer を始動すると表示される「ようこそ」ページ。
WebSphere Adapter for Oracle E-Business Suite 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 6.1.0 インフォメーション・センター (<http://publib.boulder.ibm.com/infocenter/dmndhelp/v6r1mx/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 6.0.2 インフォメーション・センター：
http://publib.boulder.ibm.com/infocenter/dmndhelp/v6rxmx/topic/com.ibm.wsadapters602.doc/welcome_top_wsa602.html
- WebSphere Adapters バージョン 6.0 インフォメーション・センター：
http://publib.boulder.ibm.com/infocenter/wbihelp/v6rxmx/topic/com.ibm.wsadapters.doc/welcome_wsa.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 ゾーン (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、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを

経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生した創作物にも、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、developerWorks、Redbooks、ViaVoice、および WebSphere は、International Business Machines 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。

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。

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]

アクション 3
アクセシビリティ
 外部サービス・ウィザード 10
 管理コンソール 9
 キーボード 10
 ショートカット・キー 10
 IBM アクセシビリティ・センター 10
アダプターの使用に関する前提条件 11
アダプターのテクニカル・ノート 106
一覧表、互換性 2
イベント 2
イベントの生成 2
インターネット・プロトコル・バージョン 6.0 (IPv6) 10
インターフェース・テーブル 4, 5
エラー、検出 102
オブジェクトの生成に時間がかかる 101

[カ行]

外部サービス・ウィザード
 アクセシビリティ 10
関連情報 105
関連製品、情報 105
キーボード 10
機能、新しい 1
基本テーブル 4, 5
クイック・スタート参照ファイル 102
組み込みアダプター
 使用する場合の考慮事項 13
 説明 11
クラスター環境
 説明 14
 デプロイ 14
 Inbound 処理 15
 Outbound 処理 15
研修、WebSphere Adapters 105
高可用性環境
 説明 14
 デプロイ 14
 Inbound 処理 15
 Outbound 処理 15
互換性一覧表 2

[サ行]

サブスクリプション 2
サポート
 テクニカル 106
参照ファイル 102
サンプル 1 4, 5, 19
 サンプル 2 3, 8
 サンプル 3 4, 6
 サンプル 4 5, 7, 78
 サンプル成果物 102
 サンプルのファイル 18
 サンプル・ファイル 18
 ショートカット・キー 10
資料、Adapter for JDBC 20, 84
新機能 1
スタンドアロン・アダプター
 使用する場合の考慮事項 14
 説明 11
成果物、サンプル 102
ソフトウェア要件 2

[タ行]

追加サンプル 18
データベース・ユーザー・アカウント 17
テクニカル・サポート 106
テクニカル・ノート 2, 106
テクニカル・ノート、WebSphere Adapters 105
デプロイメント
 オプション 11
同義語 45
 検索を高速化するための 19
 作成 19
トラブルシューティング情報 102
トリガー 9

[ハ行]

ハードウェアおよびソフトウェア要件 2
ハードウェア要件 2
始める前に 11
パラメーター
 レコード 4
 レコード以外 4
ビジネス・オブジェクト情報 23
標準の準拠 9
ファイル
 クイック・スタート参照 102
 プロジェクト交換 102
プロジェクト、作成 20, 49, 67, 84

プロジェクト交換ファイル 102
プロジェクトの作成 20, 49, 67, 84
並行プログラム 8

[マ行]

問題、既知の 101
問題、Create 操作中の 101

[ヤ行]

ユーザズ・ガイド、Adapter for JDBC 20, 84
ユーザー・アカウント
 権限 17
 任務の変更 17
要件、ハードウェアおよびソフトウェア 2

[ラ行]

レコード以外のパラメーター 4
レコード・パラメーター 4

[ワ行]

ワークフロー 7

A

Adapter for Oracle E-Business Suite
 アクセシビリティ 9
 標準の準拠 9

B

Business Event System (BES) 2

C

Create 操作 6

D

developerWorks 106
developerWorks リソース、WebSphere Adapters 105

E

enableHASupport プロパティ 15

I

IBM WebSphere Adapter Toolkit 106
Inbound 処理のサンプル 3

112 WebSphere Adapters: WebSphere Adapter for Oracle E-Business Suite

IPv6 10

O

Outbound 処理サンプル 4

R

Redbooks、WebSphere Adapters 105

W

WebSphere Adapters バージョン 6.0 情報 105
WebSphere Adapters バージョン 6.0.2 情報 105
WebSphere Application Server 情報 105
WebSphere Business Integration Adapters の情報 105
WebSphere Business Process Management バージョン 6.1.0 情報
 105
WebSphere Enterprise Service Bus
 情報 105
WebSphere Extended Deployment 15
WebSphere Integration Developer
 情報 105
WebSphere Process Server
 情報 105



Printed in Japan