

**Oracle® Fusion Middleware Application Adapters  
Oracle WebLogic Server  
Application Adapter for SAP R/3 (SAP JCo 3.0)**

ユーザーズ・ガイド

12c リリース 1 (12.1.3.0.0)

部品番号 : E61977-01

2014 年 9 月

SAP R/3 システムと統合する方法およびアプリケーションを開発する方法に関する情報を提供します。

Oracle Fusion Middleware Application Adapters Oracle WebLogic Server Application Adapter for SAP R/3 (SAP JCo 3.0) ユーザーズ・ガイド, 12c リリース 1 (12.1.3.0.0)

部品番号 : E61977-01

原本名 : Oracle Fusion Middleware Application Adapters Application Adapter for SAP R/3 (SAP JCo 3.0) User's Guide for Oracle WebLogic Server 12c Release 1 (12.1.3.0.0)

原本部品番号 : E58251-01

原著者 : Stefan Kostial

原協力者 : Vikas Anand, Marian Jones, Sunil Gopal, Bo Stern

Copyright © 2001, 2014 Oracle Corporation. All rights reserved.

#### 制限付権利の説明

このプログラム（ソフトウェアおよびドキュメントを含む）には、オラクル社およびその関連会社に所有権のある情報が含まれています。このプログラムの使用または開示は、オラクル社およびその関連会社との契約に記された制約条件に従うものとします。著作権、特許権およびその他の知的財産権と工業所有権に関する法律により保護されています。

独立して作成された他のソフトウェアとの互換性を得るために必要な場合、もしくは法律によって規定される場合を除き、このプログラムのリバース・エンジニアリング、逆アセンブル、逆コンパイル等は禁止されています。

このドキュメントの情報は、予告なしに変更される場合があります。オラクル社およびその関連会社は、このドキュメントに誤りが無いことの保証は致し兼ねます。これらのプログラムのライセンス契約で許諾されている場合を除き、プログラムを形式、手段（電子的または機械的）、目的に関係なく、複製または転用することはできません。

このプログラムが米国政府機関、もしくは米国政府機関に代わってこのプログラムをライセンスまたは使用する者に提供される場合は、次の注意が適用されます。

#### U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation, and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

このプログラムは、核、航空産業、大量輸送、医療あるいはその他の危険が伴うアプリケーションへの用途を目的としておりません。このプログラムをかかるとして使用する際、上述のアプリケーションを安全に使用するために、適切な安全装置、バックアップ、冗長性（redundancy）、その他の対策を講じることは使用者の責任となります。万一かかるプログラムの使用に起因して損害が発生いたしましても、オラクル社およびその関連会社は一切責任を負いかねます。

Oracle は Oracle Corporation およびその関連会社の登録商標です。その他の名称は、Oracle Corporation または各社が所有する商標または登録商標です。

---

---

# 目次

<b>はじめに</b> .....	<b>xi</b>
対象読者 .....	xi
ドキュメントのアクセシビリティについて .....	xi
関連ドキュメント .....	xi
表記規則 .....	xii
<b>1 概要</b>	
<b>アダプタの機能</b> .....	1-1
サポート対象バージョンおよびプラットフォーム .....	1-3
<b>ABAP 用の従来の SAP テクノロジ</b> .....	1-3
<b>SAP R/3 との統合</b> .....	1-4
<b>アダプタのアーキテクチャ</b> .....	1-5
Oracle Adapter Business Services Engine (BSE) のアーキテクチャ .....	1-5
Oracle Adapter J2CA アーキテクチャ .....	1-6
<b>BSE と Oracle Adapter J2CA デプロイメントの比較</b> .....	1-7
<b>サンプル・プロジェクト</b> .....	1-8
<b>クイック・スタート・ガイド</b> .....	1-9
インストール .....	1-9
サード・パーティ・ライブラリ・ファイルのコピー .....	1-10
構成 .....	1-10
WebLogic Server のデプロイメントおよび統合 .....	1-11
アプリケーション・エクスペローラでの構成、ターゲットおよびチャネルの作成 .....	1-11
SOA Suite でのサービス・コンポーネントの使用 .....	1-12
Oracle Service Bus の使用 .....	1-12
同期 SAP R/3 イベントの使用 .....	1-12
追加の機能 .....	1-12
その他の機能 .....	1-13
<b>2 スタート・ガイド</b>	
<b>SAP Java Connector (SAP JCo) の検証</b> .....	2-1
Windows プラットフォームでの SAP JCo の検証 .....	2-1
UNIX プラットフォームでの SAP JCo の検証 .....	2-2
<b>SAP R/3 ログオン・パラメータの識別</b> .....	2-3
ユーザー・パラメータ .....	2-3
システム設定 (アプリケーション・サーバー) パラメータ .....	2-4
システム設定 (メッセージ・サーバー) パラメータ .....	2-4

接続プール・パラメータ .....	2-5
SAP ゲートウェイ・パラメータ .....	2-5
ALE パラメータ .....	2-6
グローバル処理パラメータ .....	2-6
SNC パラメータ .....	2-7

### 3 SAP Java Connector 3.0xx の考慮事項

サポートされているプラットフォーム .....	3-1
接続管理 .....	3-2
接続プーリング .....	3-2
単一サーバーのサポート .....	3-4
サーバー・スレッド .....	3-4
一意性 .....	3-4
ロード・バランシングのためのアダプタ構成 .....	3-4
ロード・バランシングのための SAP 構成 .....	3-5
パラメータおよび表パラメータの変更 .....	3-5
トレース・レベル・パラメータ .....	3-6

### 4 構成 Oracle Application Adapter for SAP R/3

アプリケーション・エクスプローラの起動 .....	4-1
リポジトリ設定の構成 .....	4-2
リポジトリ構成の作成 .....	4-2
BSE の構成の作成 .....	4-2
J2CA の構成の作成 .....	4-3
BSE または J2CA 構成への接続 .....	4-4
SAP R/3 の接続 (ターゲット) の確立 .....	4-5
SAP R/3 へのターゲットの定義 .....	4-5
定義済 SAP R/3 ターゲットへの接続 .....	4-12
SAP R/3 への接続の管理 .....	4-12
アプリケーション・システム・オブジェクトの表示 .....	4-14
XML スキーマの作成 .....	4-14
WSDL の生成 (J2CA 構成のみ) .....	4-15
Web サービスの作成およびテスト (BSE 構成のみ) .....	4-16
Web サービスの作成 .....	4-17
Web サービスのテスト .....	4-18
アイデンティティ伝播 .....	4-18
イベント・アダプタの構成 .....	4-19
チャンネルの作成および編集 .....	4-19
スキーマの検証 .....	4-25
SAProuter サービスの構成 .....	4-27
SAProuter の構成 .....	4-27
Oracle コンポーネントを使用した SAProuter のテスト .....	4-28
SAProuter 構成の検証 .....	4-29

### 5 Oracle WebLogic Server のデプロイメントおよび統合

アダプタと Oracle WebLogic Server との統合 .....	5-1
アダプタのデプロイメント .....	5-1



アダプタ構成の更新 .....	5-2
管理対象コネクタ・ファクトリ・オブジェクトの作成 .....	5-3
複数の管理対象コネクタ・ファクトリ・オブジェクトの作成 .....	5-5
追加の接続ファクトリ値のための WSDL ファイルの変更 .....	5-6

## 6 Oracle SOA Suite での BPEL サービス・コンポーネントとの統合

概要 .....	6-1
アダプタのデプロイメント .....	6-2
新規アプリケーション・サーバー接続の構成 .....	6-2
サービス統合のためのアウトバウンド BPEL プロセスの設計 (J2CA 構成) .....	6-8
リクエスト/レスポンス・サービスの WSDL の生成 .....	6-9
SOA 用の空のコンポジットの作成 .....	6-11
BPEL アウトバウンド・プロセスの定義 .....	6-13
BPEL アウトバウンド・プロセスのデプロイ .....	6-34
Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し .....	6-39
アウトバウンド BPEL およびメディエータ・プロセスのテスト .....	6-42
イベント統合のためのインバウンド BPEL プロセスの設計 (J2CA 構成) .....	6-42
イベント統合のための WSDL の生成 .....	6-42
SOA 用の空のコンポジットの作成 .....	6-49
BPEL インバウンド・プロセスの定義 .....	6-49
BPEL インバウンド・プロセスのデプロイ .....	6-59
SAP R/3 でのイベントのトリガー .....	6-60
BPEL を使用した同期 SAP R/3 イベントの処理 (J2CA 構成) .....	6-62
概要 .....	6-62
アダプタ・ターゲットの構成 .....	6-63
チャネルの構成 .....	6-63
メッセージの WSDL の生成 .....	6-64
JDeveloper での BPEL プロセスの作成 .....	6-64
サービス統合のためのアウトバウンド BPEL プロセスの設計 (BSE 構成) .....	6-70
Web サービスを使用したリクエストおよびレスポンス・サービスの WSDL ファイルの生成 .....	6-71
SOA 用の空のコンポジットの作成 .....	6-72
BPEL アウトバウンド・プロセスの定義 .....	6-72

## 7 Oracle SOA Suite でのメディエータ・サービス・コンポーネントとの統合

新規アプリケーション・サーバー接続の構成 .....	7-2
メディエータ・アウトバウンド・プロセスの構成 (J2CA 構成) .....	7-2
SOA 用の空のコンポジットの作成 .....	7-2
メディエータ・アウトバウンド・プロセスの定義 .....	7-3
メディエータ・アウトバウンド・プロセスのデプロイ .....	7-12
Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し .....	7-12
メディエータ・インバウンド・プロセスの構成 (J2CA 構成) .....	7-13
SOA 用の空のコンポジットの作成 .....	7-13
メディエータ・インバウンド・プロセスの定義 .....	7-13
メディエータ・アウトバウンド・プロセスの構成 (BSE 構成) .....	7-20
SOA 用の空のコンポジットの作成 .....	7-20
メディエータ・アウトバウンド・プロセスの定義 .....	7-21

## 8 Oracle SOA Suite での BPM サービス・コンポーネントとの統合

概要 .....	8-1
アダプタのデプロイメント .....	8-2
新規アプリケーション・サーバー接続の構成 .....	8-2
トランスフォーメーションを使用したサービス統合のためのアウトバウンド BPM プロセスの設計 (J2CA 構成) .....	8-2
BPM 用の空のコンポジットの作成 .....	8-2
BPM アウトバウンド・プロセスの定義 .....	8-3
12c でのデプロイメントに関する既知の問題の調整 .....	8-26
BPM アウトバウンド・プロセスのデプロイ .....	8-26
Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し .....	8-26
トランスフォーメーションを使用したイベント統合のためのインバウンド BPM プロセスの設計 (J2CA 構成) .....	8-29
BPM 用の空のコンポジットの作成 .....	8-29
BPM インバウンド・プロセスの定義 .....	8-29
<b>BPM を使用した同期 SAP R/3 イベントの処理 (J2CA 構成) .....</b>	<b>8-47</b>
アダプタ・ターゲットおよびチャネルの構成 .....	8-48
メッセージの WSDL ファイルの生成 .....	8-48
JDeveloper での BPM プロセスの作成 .....	8-48
トランスフォーメーションを使用したサービス統合のためのアウトバウンド BPM プロセスの設計 (BSE 構成) .....	8-54
BPM 用の空のコンポジットの作成 .....	8-55
BPM アウトバウンド・プロセスの定義 .....	8-55

## 9 sbconsole を使用した Oracle Service Bus のアウトバウンドおよびインバウンド・プロセスの構成

アプリケーション・アダプタと Oracle Service Bus との統合の概要 .....	9-1
<b>sbconsole を使用したアウトバウンド・プロセスの構成 (J2CA 構成) .....</b>	<b>9-1</b>
Oracle Service Bus の起動およびプロジェクト・フォルダの作成 .....	9-2
アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定 .....	9-6
アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ .....	9-6
WSDL ベースのビジネス・サービスの構成 .....	9-8
ファイル・タイプ・ビジネス・サービスの構成 .....	9-11
プロキシ・サービスでのパイプラインの構成 .....	9-16
<b>sbconsole を使用したインバウンド・プロセスの構成 (J2CA 構成) .....</b>	<b>9-25</b>
Oracle Service Bus の起動およびプロジェクト・フォルダの作成 .....	9-26
アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定 .....	9-26
アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ .....	9-26
WSDL ベースのプロキシ・サービスの構成 .....	9-28
ファイル・タイプ・ビジネス・サービスの構成 .....	9-30
パイプラインの構成 .....	9-33
<b>sbconsole を使用したアウトバウンド・プロセスの構成 (BSE 構成) .....</b>	<b>9-40</b>
Oracle Service Bus の起動およびプロジェクト・フォルダの作成 .....	9-41
アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定 .....	9-41
アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ .....	9-41

ファイル・タイプ・ビジネス・サービスの構成 .....	9-43
WSDL ベースのビジネス・サービスの構成 .....	9-43
プロキシ・サービスでのパイプラインの構成 .....	9-46
<b>Oracle Service Bus を使用した同期 SAP R/3 イベントの使用 (J2CA 構成)</b> .....	9-55
Oracle Service Bus の起動およびプロジェクト・フォルダの作成 .....	9-56
アダプタ・ターゲットおよびチャネルの構成 .....	9-56
メッセージの WSDL ファイルの生成 .....	9-56
Oracle Service Bus コンソールでのビジネス・サービスの作成 .....	9-57
Oracle Service Bus コンソールでのプロキシ・サービスの作成 .....	9-58
プロキシ・サービスでのパイプラインの構成 .....	9-62
<b>Oracle Service Bus を使用した JMS プロキシ・サービスの構成 (J2CA 構成)</b> .....	9-64
<b>Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成)</b> .....	9-82

## 10 JDeveloper を使用した Oracle Service Bus のアウトバウンドおよびインバウンド・プロセスの構成

<b>JDeveloper を使用した OSB アウトバウンド・プロセスの構成 (J2CA 構成)</b> .....	10-1
OSB の Service Bus アプリケーションの作成 .....	10-2
OSB アウトバウンド・プロセスの定義 .....	10-3
OSB アウトバウンド・プロセスのデプロイ .....	10-17
<b>JDeveloper を使用した OSB インバウンド・プロセスの構成 (J2CA 構成)</b> .....	10-20
OSB の Service Bus アプリケーションの作成 .....	10-20
OSB インバウンド・プロセスの定義 .....	10-20
OSB インバウンド・プロセスのデプロイ .....	10-32
<b>JDeveloper を使用した OSB アウトバウンド・プロセスの構成 (BSE 構成)</b> .....	10-32
OSB の Service Bus アプリケーションの作成 .....	10-32
OSB アウトバウンド・プロセスの定義 .....	10-32
OSB アウトバウンド・プロセスのデプロイ .....	10-47
<b>JDeveloper を使用した Oracle Service Bus の同期 SAP R/3 イベントの処理 (J2CA 構成)</b> .....	10-47
OSB の Service Bus アプリケーションの作成 .....	10-47
OSB プロセスの定義 .....	10-47
OSB プロセスのデプロイ .....	10-52
<b>JDeveloper を使用した JMS インバウンド・プロセスの構成 (J2CA 構成)</b> .....	10-52
<b>JDeveloper を使用した JMS アウトバウンド・プロセスの構成 (J2CA 構成)</b> .....	10-59
<b>JDeveloper を使用した HTTP アウトバウンド・プロセスの構成 (J2CA 構成)</b> .....	10-68

## 11 主要な機能

<b>例外フィルタ</b> .....	11-1
例外フィルタの構成 .....	11-2
<b>Oracle SOA Suite (BPEL、メディエータまたは BPM) の資格証明マッピング</b> .....	11-15
資格証明マッピングの構成 .....	11-15
<b>JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング</b> .....	11-19
資格証明マッピングの構成 .....	11-20
<b>テキストからの Unicode 文字のフィルタリング</b> .....	11-28
概要 .....	11-28
フィルタの有効化 .....	11-28
置換文字の指定 .....	11-29

設定ファイルの保存およびサーバーの再起動 .....	11-29
フィルタの検証 .....	11-29
クイック・スタート・ガイド .....	11-29
既知の制限 .....	11-30

## 12 トラブルシューティングおよびエラー・メッセージ

トラブルシューティング .....	12-1
アプリケーション・エクスプローラ .....	12-2
SAP R/3 .....	12-4
Oracle Adapter J2CA .....	12-4
BSE エラー・メッセージ .....	12-5
BSE での一般的なエラー処理 .....	12-5
アダプタ固有のエラー処理 .....	12-5

## A リモート処理での SAP のシステム・ロールについて

アダプタの機能 .....	A-1
ロール .....	A-2
クライアント .....	A-2
サーバー .....	A-2
SAP インバウンド・メッセージ処理 .....	A-3
SAP R/3 インバウンド処理の構成 .....	A-4
論理システムの構成 .....	A-6
論理システム用の配信モデルの構成 .....	A-7
パートナ・プロファイルの定義 .....	A-8
SAP アウトバウンド・メッセージ処理 .....	A-9
SAP ゲートウェイ .....	A-11
プログラム ID とロード・バランシング .....	A-11
SAP GUI での RFC 宛先とプログラム ID の登録 .....	A-13
イベント・アダプタ用の Application Link Embedding 構成 .....	A-15
ポートの定義 .....	A-15
論理システムの構成 .....	A-15
論理システム用の配信モデルの構成 .....	A-17
パートナ・プロファイルの定義 .....	A-18
SAP R/3 ALE 構成のテスト .....	A-19

## 用語集

## 索引

---

---

# はじめに

『*Oracle Fusion Middleware Application Adapters Oracle WebLogic Server Application Adapter for SAP R/3 (SAP JCo 3.0) ユーザーズ・ガイド*』によるこそ。このマニュアルは、SAP R/3 システムと統合する方法およびアプリケーションを開発する方法に関する情報を提供します。

## 対象読者

このマニュアルは、SAP R/3 システムと統合し、アプリケーションを開発するシステム管理者および開発者を対象としています。

## ドキュメントのアクセシビリティについて

オラクルのアクセシビリティについての詳細情報は、Oracle Accessibility Program の Web サイト (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>) を参照してください。

### Oracle Support へのアクセス

お客様には、My Oracle Support を通して電子支援サービスを提供しています。詳細情報は <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> か、聴覚に障害のあるお客様は <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> を参照してください。

## 関連ドキュメント

詳細は、Oracle Enterprise Repository 12c リリース 1 (12.1.3.0.0) ドキュメント・セット内の次のドキュメントを参照してください。

- *Oracle Fusion Middleware Oracle WebLogic Server Application Adapters* インストール・ガイド
- *Oracle Fusion Middleware Application Adapters Oracle WebLogic Server Application Adapter* アップグレード・ガイド
- *Oracle Fusion Middleware Application Adapters Oracle WebLogic Server Application Adapter* ベスト・プラクティス・ガイド
- Oracle Unified Method (OUM)

Oracle Unified Method (OUM) では、その他の管理情報を豊富に提供しています。OUM は、オラクル社の従業員、パートナー・ネットワーク認定パートナーや認定アドバンテージ・パートナーとともに、OUM 顧客プログラム、またはオラクル社によるコンサルティング・サービス提供プロジェクトにご参加いただいているお客様にもご利用いただけます。OUM はソフトウェア開発および実装プロジェクトの計画、実施、管理用の Web デプロイ型ツールキットです。

OUM の詳細は、次の URL の OUM FAQ を参照してください。

[http://my.oracle.com/portal/page/myo/ROOTCORNER/KNOWLEDGEAREAS1/BUSINESS\\_PRACTICE/Methods/Learn\\_about\\_OUM.html](http://my.oracle.com/portal/page/myo/ROOTCORNER/KNOWLEDGEAREAS1/BUSINESS_PRACTICE/Methods/Learn_about_OUM.html)

## 表記規則

このドキュメントでは次のテキスト表記規則を使用します。

表記規則	意味
太字	太字タイプは、操作に関連するグラフィカル・ユーザー・インタフェース要素、またはテキストや用語集で定義されている用語を示します。
斜体	斜体タイプは、マニュアル・タイトル、強調またはユーザーが特定の値を指定するプレースホルダ変数を示します。
等幅	等幅タイプは、段落内のコマンド、URL、サンプル内のコード、画面に表示されるテキストまたは入力するテキストを示します。

Oracle WebLogic Server は、Oracle Application Adapter for SAP R/3 を介して SAP R/3 システムに接続します。Oracle Application Adapter for SAP R/3 は接続性を提供し、SAP R/3 システム上で相互作用を実行します。

---

---

**注意：** このマニュアル全体で、<ORACLE\_HOME> は、12c がインストールされたホームの場所を指しています。

<ADAPTER\_HOME> は次を指しています。

- SOA の場合：

<ORACLE\_HOME>\soa\soa\thirdparty\ApplicationAdapters

- OSB の場合：

<ORACLE\_HOME>\osb\3rdparty\ApplicationAdapters

---

---

この章は次の項で構成されています。

- [1.1 項「アダプタの機能」](#)
- [1.2 項「ABAP 用の従来の SAP テクノロジー」](#)
- [1.3 項「SAP R/3 との統合」](#)
- [1.4 項「アダプタのアーキテクチャ」](#)
- [1.5 項「BSE と Oracle Adapter J2CA デプロイメントの比較」](#)
- [1.6 項「サンプル・プロジェクト」](#)
- [1.7 項「クイック・スタート・ガイド」](#)

## 1.1 アダプタの機能

SAP Java Connector (SAP JCo) 3.0xx を使用する Oracle Application Adapter for SAP R/3 は、SAP Enterprise Central Component (ECC) 5.0/6.0 システムと他のアプリケーション、データベースまたは外部のビジネス・パートナー・システムとの間でリアルタイムのビジネス・データを交換する手段を提供する、リモート関数呼出しアダプタです。

---

---

**注意:** SAP JCo 3.0xx の xx は、SAP Java Connector の特定のバージョンを示します。

SAP ポリシーでは、現在リリースされているバージョンの SAP JCo のみがサポートされているバージョンとして規定されています。

サポートについて問題のあるユーザーは、より新しいバージョンの SAP JCo (存在する場合) を使用してテストし、問題がすでに解決されていることを確認することが必要な場合があります。

また、Oracle JRockit JVM は SAP JCo ではサポートされていません。

---

---

**アダプタ**により、外部アプリケーションで SAP R/3 とのインバウンドおよびアウトバウンド処理が可能となります。Oracle Application Adapter for SAP R/3 は、J2EE Connector Architecture (J2CA) バージョン 1.0 リソース・アダプタとしてデプロイできます。このデプロイメントは Oracle Adapter J2CA と呼ばれます。また、Web サービス・サーブレットとしてもデプロイでき、この場合には、Oracle Adapter Business Services Engine (BSE) と呼ばれます。

この項では、次の項目について説明します。

#### ■ 1.1.1 項「サポート対象バージョンおよびプラットフォーム」

Oracle Application Adapter for SAP R/3 は、XML メッセージを使用して、SAP R/3 以外のアプリケーションがサービスおよびイベントを介して SAP R/3 と通信し、トランザクションを交換できるようにします。サービスおよびイベントの役割の概要を示します。サービスおよびイベントの説明は次のとおりです。

- サービス (アウトバウンド処理とも呼ばれます): アプリケーションが SAP R/3 ビジネス・オブジェクトまたはビジネス操作を呼び出すことができます。
- イベント (インバウンド処理とも呼ばれます): アプリケーションが、イベント発生時のみ SAP R/3 データにアクセスできるようにします。

イベント機能に対応するために、チャンネルがサポートされています。**チャンネル**は、バックエンドまたは他のタイプのシステムの特定のインスタンスへの構成済の接続を表します。

チャンネルは、EIS アプリケーションからリアルタイムでイベントを受信するアダプタ・コンポーネントです。ファイル・リーダー、HTTP リスナー、TCP/IP リスナーまたは FTP リスナーをチャンネル・コンポーネントにすることができます。チャンネルは常に EIS に固有です。アダプタでは特定の EIS に対して複数のチャンネルがサポートされるため、ユーザーは、デプロイメント要件に基づいて最適なチャンネル・コンポーネントを選択できます。このアダプタの場合、チャンネルは RFC サーバーです。

Oracle Application Adapter for SAP R/3 は次のものを備えています。

- 双方向メッセージ相互作用のサポート。
- Oracle Adapter Application Explorer (アプリケーション・エクスプローラ)。SAP R/3 オブジェクト・リポジトリ・メタデータを使用して、アダプタ・リクエストやイベント・データを処理するための XML スキーマと Web サービスを構築する GUI ツールです。
- SAP R/3 へのリモート関数呼出し (RFC)、Business Application Programming Interfaces (BAPI) および Intermediate Documents (IDoc) インタフェースのサポート。
- J2CA 1.0 と J2CA 1.5 リソース・アダプタ用の XML スキーマおよび WSDL ファイル。
- BSE 用の Web サービス。

**データ型の制限:** データ型 h および g はサポートされていません。データ型 h はディープ構造を表します。データ型 g は可変長の文字列を表します。SAPRFC.H で定義された RFCTYPE\_XSTRING および RFCTYPE\_XMLDATA は、RFC プロトコルの制限のためにサポートされません。

**関連項目:** Oracle Application Server Adapter コンセプト・ガイド



## 1.1.1 サポート対象バージョンおよびプラットフォーム

Oracle Application Adapter for SAP R/3 では、次の SAP R/3 プラットフォームがサポートされています。

- SAP R/3 Enterprise 47x100
- SAP R/3 Enterprise 47x200
- mySAP ERP Central Component (ECC) 5.0 (SAP NetWeaver 2004 上にデプロイ)
- mySAP ERP Central Component (ECC) 6.0 (SAP NetWeaver 2004s 上にデプロイ)
- SAP Java Connector (SAP JCo) バージョン 3.0xx

SAP Java Connector の現在のリリース・ステータスについては、SAP Service Marketplace で SAP ノート #549268 を参照してください。

---

**注意：** リリース・バージョンは、製品コンポーネントごとに異なる場合があります。さらに、SAP 関数も SAP 製品バージョンおよびサポート・パッケージごとに異なる場合があります。

---

## 1.2 ABAP 用の従来の SAP テクノロジ

Oracle Application Adapter for SAP R/3 は、既存のビジネス・プロセスをサポートするために使用されるリモート関数呼出し (RFC) モジュール、BAPI (Business Application Programming Interfaces)、IDoc (Intermediate Documents) などの SAP R/3 インタフェースに対する標準アクセスを提供するように設計されています。

アダプタでは、従来の SAP テクノロジによってアクセスされる Enterprise Central Components (ECC) のみがサポートされています。その他の SAP 機能およびコンポーネントに対するサポートが必要な場合は、Oracle ソフトウェアの営業担当員に問い合せてください。

これらのビジネス・コンポーネントおよびメソッドは、アダプタでは SAP R/3 のリクエストとして使用でき、イベント・アダプタでは SAP がリモート・リクエストを呼び出したときに使用可能となり、次のように動作します。

- Business Application Programming Interfaces (BAPI) は、SAP コンポーネントを相互にリンクしたり、SAP コンポーネントとサード・パーティ・コンポーネントをリンクするために使用される、ビジネス・フレームワーク内のインタフェースです。BAPI は同期的に呼び出され、情報を返します。
- リモート関数呼出し (RFC) モジュールは、クライアントが SAP テクノロジを起動してレスポンスを受信できるようにする、SAP アプリケーション・インタフェースです。

---

**注意：** インストールされているリリースまたはサービス・パックによっては、特定の SAP R/3 システムに RFC\_CUSTOMER\_GET などの特定の RFC が存在しないことがあります。したがって、このマニュアルで説明する例が、使用しているシステムに該当しない場合があります。このような場合、これらの例をアダプタ機能の一般的な参考情報として使用したうえで、SAP R/3 アプリケーション環境内に存在する RFC を選択する必要があります。

SAP リリース・ノート 109533 で説明されているように、SAP 関数モジュール (RFC) は様々なリリース・ステータスで配布されている可能性があります。SAP では、「Released for Customer」ステータスの RFC のみをサポートしています。インタフェースのリリースからの独立性とモジュールの存在および機能の継続性に関する報告はありません。特定の関数モジュールのステータスの詳細は、SAP Service Marketplace を参照してください。

---

- **Intermediate Documents (IDoc)** は、様々なビジネス・プロセスに対応する論理メッセージです。これにより、様々なアプリケーション・システムがメッセージベースのインタフェースによってリンクできます。IDoc タイプは、ビジネス・トランザクションのデータを転送するために使用する SAP フォーマットを示します。IDoc は、IDoc タイプの形式で表された実際のビジネス・プロセスで、複数のメッセージ・タイプを転送できます。IDoc タイプは、次のコンポーネントにより記述されます。
  - 制御レコード。制御レコードには、送信者、受信者および IDoc 構造を示すデータが含まれています。IDoc には、1 つの制御レコードが含まれます。
  - データ・レコード。データ・レコードは、固定の管理部分とデータ部分 (セグメント) で構成されています。セグメントの数とフォーマットは、IDoc タイプごとに異なる可能性があります。
  - ステータス・レコード。ステータス・レコードは、IDoc が通過する処理ステージを表しています。次のシナリオは、IDoc 機能とそのコンポーネントの例です。  
注文書番号 4711 が IDoc 番号 0815 としてベンダーに送信されました。IDoc 番号 0815 は、IDoc タイプ ORDERS01 で書式設定され、そのステータス・レコードは「created」および「sent」です。注文書は論理メッセージ ORDERS に対応しています。

## 1.3 SAP R/3 との統合

Oracle Application Adapter for SAP R/3 は、アカウントの追加や更新などの SAP R/3 ビジネス・プロセスを開始したり、SAP R/3 システムと SAP R/3 以外のシステムとを接続するための統合の一部として使用できます。すべての関数は同期的に処理されますが、ALE IDoc のすべてのコンテンツは非同期です。

Oracle Application Adapter for SAP R/3 では、サービス・モードで BAPI、RFC または ALE インタフェースを使用して SAP にリクエストを送信できます。

アダプタにより、SAP R/3 の IDoc、RFC および BAPI が、基幹となる SAP R/3 システム・アプリケーションおよびその他のエンタープライズ・アプリケーションと迅速かつ簡単に統合されます。アダプタの利点は次のとおりです。

- カスタム・コーディングが不要。
- 一貫性のあるデータ表記。  
イベント・データの標準 XML 表現および SAP R/3 のリクエスト/レスポンス・ドキュメントを提供します。  
開発者は、SAP R/3 インタフェース (BAPI、RFC、IDoc) に関する特定の詳細やターゲット SAP R/3 システムに関する特定の構成詳細を考慮する必要がなくなります。
- SAP AG によって公開されている、SAP R/3 ABAP シリアライズ・ルールおよび SAP R/3 インタフェース・リポジトリ標準の順守。

イベント処理時に、アダプタは SAP R/3 から RFC と IDoc を直接受信します。特定のイベントが発生したときに IDoc または RFC を論理システム (この場合はアダプタ) に送信するように SAP R/3 システムを構成できます。SAP R/3 によって送信される出力に使用可能な形式は、次のいずれかです。

- RFC リクエスト (RFC\_SYSTEM\_INFO など)
- BAPI リクエスト (BAPI\_COMPANYCODE\_GETLIST など)
- IDoc

## 1.4 アダプタのアーキテクチャ

この項では、Oracle Application Adapter for SAP R/3 のアーキテクチャについて説明します。ここでは、次の項目について説明します。

- 1.4.1 項「Oracle Adapter Business Services Engine (BSE) のアーキテクチャ」
- 1.4.2 項「Oracle Adapter J2CA アーキテクチャ」

Oracle Application Adapter for SAP R/3 は、アプリケーション・エクスプローラを次のコンポーネントとともに使用します。

- Oracle Adapter Business Services Engine (BSE)
- J2EE Connector Architecture (J2CA) 用エンタープライズ・コネクタ

アプリケーション・エクスプローラ (SAP 接続の構成および Web サービスとイベントの作成に使用) は、Web サービス環境で BSE と連携するように構成できます。J2CA 環境で動作している場合、コネクタは Common Client Interface (CCI) を使用して Web サービスのかわりにアダプタを使用した統合サービスを提供します。

### 1.4.1 Oracle Adapter Business Services Engine (BSE) のアーキテクチャ

図 1-1 に、パッケージ化されたアプリケーション用の BSE の汎用アーキテクチャを示します。アダプタは、J2EE アプリケーション・サーバー内の Web コンテナにデプロイされていると、BSE と連携します。

アプリケーション・エクスプローラは、BSE とともにデプロイされるデザインタイム・ツールであり、アダプタ接続の構成、EIS オブジェクトの参照、サービスの構成および EIS イベントをリスニングするためのリスナーの構成に使用されます。これらの操作の実行中に作成されたメタデータは、BSE によってリポジトリに格納されます。

BSE はプロトコルとして SOAP を使用し、クライアントからのリクエストを受信し、EIS と相互作用して EIS からのレスポンスをクライアントに送信します。

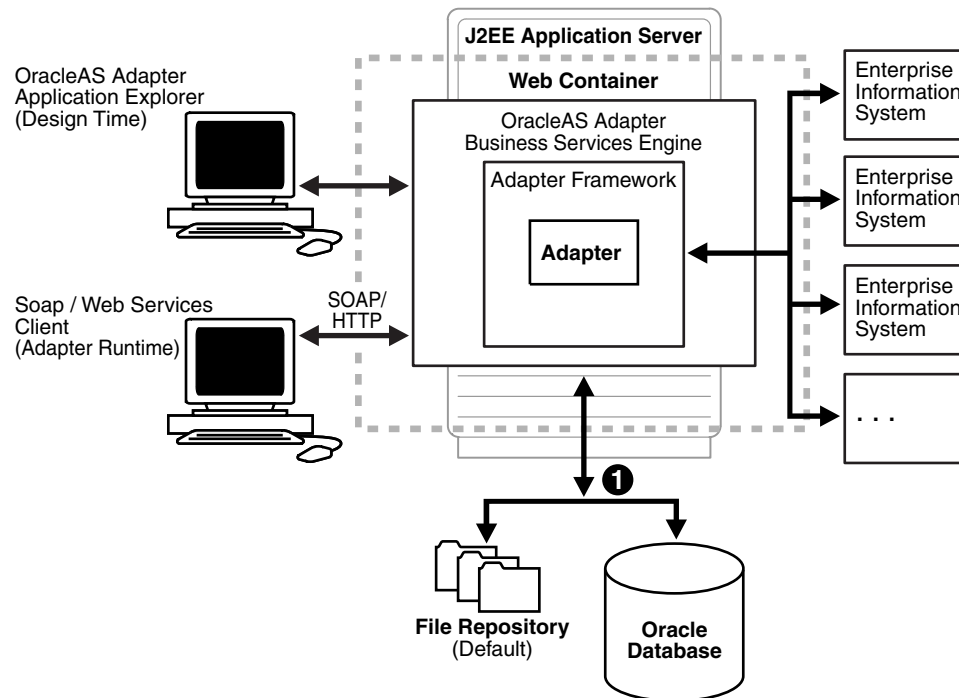
BSE はファイルベースのリポジトリと Oracle Database リポジトリの両方をサポートします。BSE リポジトリには、EIS 接続情報およびアダプタ・サービス用の Web Service Definition Language (WSDL) が格納されます。単一の BSE インスタンスで複数の EIS アプリケーションに接続できます。

---

**注意：** BSE 用のファイル・リポジトリを本番環境で使用しないでください。

---

図 1-1 Oracle Adapter Business Services (BSE) のアーキテクチャ



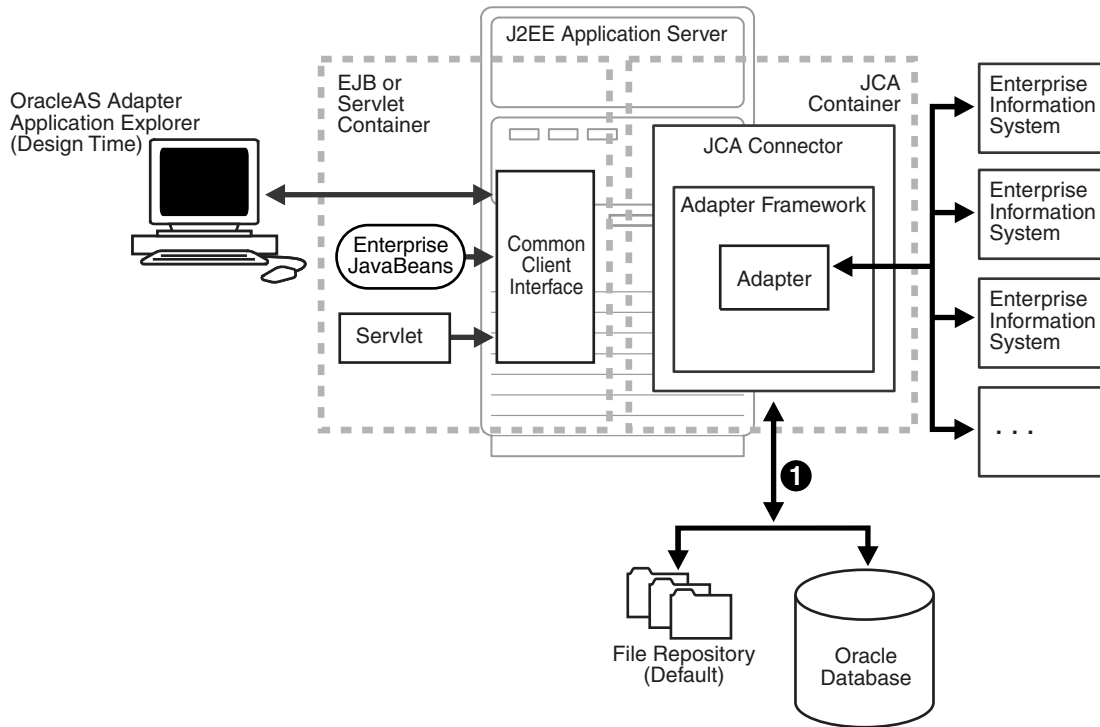
① Use either the default file repository or an Oracle database as your repository.

## 1.4.2 Oracle Adapter J2CA アーキテクチャ

図 1-2 に、パッケージ化されたアプリケーション用の Oracle Adapter J2CA の汎用アーキテクチャを示します。これは、管理モードで Oracle WebLogic Server にデプロイされた純粋な J2CA 1.0 リソース・アダプタです。これはユニバーサル・アダプタです。1つのアダプタで複数の EIS アプリケーションに接続できます。

Oracle Adapter J2CA のリポジトリには、EIS 接続名のリストおよび関連する接続パラメータのリストが含まれています。リポジトリには、ファイル・システムまたは Oracle Database を使用できます。これは RAR ファイルとしてデプロイされ、ra.xml と呼ばれるデプロイメント・ディスクリプタが関連付けられます。Oracle WebLogic Server デプロイメント・ディスクリプタ ra.xml を編集して、複数のコネクタ・ファクトリを作成できます。詳細は、第 5 章「Oracle WebLogic Server のデプロイメントおよび統合」を参照してください。

図 1-2 Oracle Adapter J2CA アーキテクチャ



① Use either the default file repository or an Oracle database as your repository.

#### 関連項目：

- *Oracle Application Server Adapter* コンセプト・ガイド
- *Oracle Fusion Middleware Oracle WebLogic Server Application Adapters* インストール・ガイド

## 1.5 BSE と Oracle Adapter J2CA デプロイメントの比較

Oracle Application Adapter for SAP R/3 を Oracle SOA Suite コンポーネント (BPEL、メディアエータ、BPM、OSB など) とともに使用する場合には、次の点に注意してください。

- Oracle SOA Suite コンポーネントとのインバウンド統合 (イベント通知) は、Oracle Adapter J2CA デプロイメントでのみサポートされます。
- Oracle SOA Suite コンポーネントとのアウトバウンド統合 (リクエスト/レスポンス・サービス) は、Oracle Adapter J2CA および BSE のデプロイメントでサポートされます。

BSE と Oracle Adapter J2CA のデプロイの相違には、次の 2 つの要因があります。これらの要因を理解すると、デプロイメント・オプションの選択に役立ちます。

1. BSE には次の利点があります。
  - Oracle WebLogic Server の個別のインスタンスでデプロイできます。
  - 優れた負荷分散を提供します。
  - アプリケーションの構築について、サービス指向アーキテクチャ (SOA) モデルに、より厳密に準拠します。
2. Oracle Adapter J2CA では、BSE よりもわずかにパフォーマンスが向上します。

## 1.6 サンプル・プロジェクト

アプリケーション・アダプタ・インストールには、Oracle BPEL、メディエータ、BPM および OSB ツールを使用したアウトバウンドとインバウンドの統合シナリオのデモンストレーションを示す、Oracle Application Adapter for SAP R/3 用のサンプル・プロジェクトがパッケージ化されています。次の表に、サンプル・プロジェクトの場所を示します。

サンプル・プロジェクト	場所
Outbound BPEL Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\J2CA\Outbound_Project
Inbound BPEL Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\J2CA\Inbound_Project
Synchronous BPEL Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\J2CA\Synchronous_Events
Outbound BPEL Process (BSE)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\BSE\Outbound_Project
Outbound Mediator Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\Mediator\J2CA\Outbound_Project
Inbound Mediator Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\Mediator\J2CA\Inbound_Project
Outbound Mediator Process (BSE)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\Mediator\BSE\Outbound_Project
Outbound BPM Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPM\J2CA\Outbound_Project
Inbound BPM Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPM\J2CA\Inbound_Project
Synchronous BPM Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPM\J2CA\Synchronous_Events
Outbound BPM Process (BSE)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPM\BSE\Outbound_Project
Outbound OSB sbconsole Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\J2CA\SAP_Sample_J2CA_OSB_Outbound_Project
Inbound OSB sbconsole Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\J2CA\SAP_Sample_J2CA_OSB_Inbound_Project
Outbound OSB sbconsole Process (BSE)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\BSE\SAP_Sample_BSE_OSB_Outbound_Project
Synchronous OSB sbconsole Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\J2CA\SAP_Sample_J2CA_OSB_Synchronous_Events

サンプル・プロジェクト	場所
Outbound OSB Jdeveloper Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\J2CA\SAP_Sample_J2CA_OSB_Outbound_Project
Inbound OSB Jdeveloper Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\J2CA\SAP_Sample_J2CA_OSB_Inbound_Project
Outbound OSB Jdeveloper Process (BSE)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\BSE\SAP_Sample_BSE_OSB_Outbound_Project
Synchronous OSB Jdeveloper Process (J2CA)	<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\J2CA\SAP_Sample_J2CA_OSB_Synchronous_Events

## 1.7 クイック・スタート・ガイド

この項では、Oracle Application Adapter for SAP R/3 をインストールおよび構成するための基本的な手順を簡単に学習し、即時に使用できるようにします。ここでは、次の項目について説明します。

- 1.7.1 項「インストール」
- 1.7.2 項「サード・パーティ・ライブラリ・ファイルのコピー」
- 1.7.3 項「構成」
- 1.7.4 項「WebLogic Server のデプロイメントおよび統合」
- 1.7.5 項「アプリケーション・エクスプローラでの構成、ターゲットおよびチャネルの作成」
- 1.7.6 項「SOA Suite でのサービス・コンポーネントの使用」
- 1.7.7 項「Oracle Service Bus の使用」
- 1.7.8 項「同期 SAP R/3 イベントの使用」
- 1.7.9 項「追加の機能」
- 1.7.10 項「その他の機能」

### 1.7.1 インストール

1. 使用中の対応するプラットフォーム用の Oracle Fusion Middleware Application Adapter インストール・ファイルをダウンロードして、ファイルを実行します。
  - **Windows:** `iwora12c_application-adapters_win.exe`
  - **Linux:** `iwora12c_application-adapters_linux.bin`
  - **Solaris:** `iwora12c_application-adapters_solaris.bin`
  - **HPUX:** `iwora12c_application-adapters_hpux.bin`
  - **AIX:** `iwora12c_application-adapters_aix.bin`

- Oracle Fusion Middleware Application Adapter は、次のいずれかのディレクトリにインストールする必要があります。

- SOA の場合:

```
<ORACLE_HOME>\soa\soa\thirdparty\ApplicationAdapters
```

- OSB の場合:

```
<ORACLE_HOME>\osb\3rdparty\ApplicationAdapters
```

Oracle Fusion Middleware Application Adapter のインストールの詳細は、『*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストレーション・ガイド*』を参照してください。

## 1.7.2 サード・パーティ・ライブラリ・ファイルのコピー

アダプタ・インストールが完了したら、SAP Java Connector (SAP JCo) に必要なサード・パーティ・ライブラリ・ファイル (*sapjco3.jar* および *sapjco3.dll*) を次のディレクトリにコピーします。

```
<ADAPTER_HOME>\lib
```

```
<ORACLE_HOME>\user_projects\domains\base_domain\lib
```

*sapjco3.dll* ファイルは、次のディレクトリにコピーする必要があります。

```
C:\Windows\system32
```

その他のオペレーティング・システムでの SAP JCo のインストールの詳細は、『*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストレーション・ガイド*』を参照してください。

## 1.7.3 構成

<ADAPTER\_HOME> にナビゲートして、次の変更を行います。

- iwafjca.rar\META-INF\ra.xml* を開き、表 1-1 に示すように、指定された `config-property-name` パラメータに次の値を追加します。

表 1-1

config-property-name	config-property-value
<i>IWayHome</i>	<ADAPTER_HOME> 次に例を示します。 <ul style="list-style-type: none"> <li>■ SOA の場合: C:\12c_soa\soa\soa\thirdparty\ApplicationAdapters</li> <li>■ OSB の場合: C:\12c_osb\osb\3rdparty\ApplicationAdapters</li> </ul>
<i>IWayConfig</i>	構成の名前。次に例を示します。 jca_sample

- ibse.war\WEB-INF\web.xml* を開き、表 1-2 に示すように、指定された `param-name` パラメータに次の値を追加します。



表 1-2

param-name	param-value
<code>ibseroot</code>	<p>&lt;ADAPTER_HOME&gt;\ibse.war</p> <p>次に例を示します。</p> <ul style="list-style-type: none"> <li>■ <b>SOA の場合:</b> <p>C:\12C_ soa\soa\soa\thirdparty\ApplicationAdapters\ibse.war</p> </li> <li>■ <b>OSB の場合:</b> <p>C:\12c_OSB\osb\3rdparty\ApplicationAdapters\ibse.war</p> </li> </ul>
<code>IWay.home</code>	<p>&lt;ADAPTER_HOME&gt;</p> <p>次に例を示します。</p> <ul style="list-style-type: none"> <li>■ <b>SOA の場合:</b> <p>C:\12c_soa\soa\soa\thirdparty\ApplicationAdapters</p> </li> <li>■ <b>OSB の場合:</b> <p>C:\12c_OSB\osb\3rdparty\ApplicationAdapters</p> </li> </ul>
<code>IWay.config</code>	<p>構成の名前。次に例を示します。</p> <p>IBSE</p>

---

**注意:** これらの手順は、ファイル・リポジトリを構成する場合にのみ実行します。データベース・リポジトリの構成の詳細および一般的な構成情報は、『*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストレーション・ガイド*』の Oracle Database リポジトリの構成に関する項を参照してください。

---

## 1.7.4 WebLogic Server のデプロイメントおよび統合

1. WebLogic Server を起動して WebLogic コンソールを開きます。
2. アダプタ・コンポーネント (ibse.war、iwafjca.war および iwafjca.rar ファイル) をデプロイして、デプロイされたアダプタ・コンポーネントを起動します。

デプロイメント、統合およびターゲット作成の詳細は、『*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストレーション・ガイド*』の J2CA の構成とデプロイに関する項および Business Services Engine の構成とデプロイに関する項を参照してください。

## 1.7.5 アプリケーション・エクスプローラでの構成、ターゲットおよびチャネルの作成

アプリケーション・エクスプローラでの構成、ターゲットおよびチャネルの作成の詳細は、このユーザー・ガイドの次の項を参照してください。

- アプリケーション・エクスプローラの起動: [4.1 項「アプリケーション・エクスプローラの起動」](#)
- BSE 構成の作成: [4.3.1 項「BSE の構成の作成」](#)
- J2CA 構成の作成: [4.3.2 項「J2CA の構成の作成」](#)
- 作成された構成の接続: [4.3.3 項「BSE または J2CA 構成への接続」](#)

- ターゲットの作成およびターゲットへの接続：4.4 項「SAP R/3 の接続 (ターゲット) の確立」
- Web サービスの作成およびテスト：4.8 項「Web サービスの作成およびテスト (BSE 構成のみ)」
- WSDL ファイルの生成：4.7 項「WSDL の生成 (J2CA 構成のみ)」
- チャンネルの作成および使用：4.9 項「イベント・アダプタの構成」

## 1.7.6 SOA Suite でのサービス・コンポーネントの使用

Oracle Application Adapter for SAP R/3 は、BPEL、メディエータ、BPM などの、SOA Suite 内のサービス・コンポーネントと統合されています。必要なプロセスは、JDeveloper で作成されてから SOA サーバーにデプロイされます。

BPEL、メディエータおよび BPM サービス・コンポーネントの使用の詳細は、次を参照してください。

- 第 6 章「Oracle SOA Suite での BPEL サービス・コンポーネントとの統合」
- 第 7 章「Oracle SOA Suite でのメディエータ・サービス・コンポーネントとの統合」
- 第 8 章「Oracle SOA Suite での BPM サービス・コンポーネントとの統合」

## 1.7.7 Oracle Service Bus の使用

Oracle Application Adapter for SAP R/3 は Oracle Service Bus (OSB) と統合され、これにより Web サービス統合が簡略化されます。必要なプロセスは、Oracle Service Bus コンソールで作成されます。プロセスを JDeveloper で作成してから SOA サーバーにデプロイすることもできます。

OSB の使用の詳細は、第 9 章「sbconsole を使用した Oracle Service Bus のアウトバウンドおよびインバウンド・プロセスの構成」を参照してください。

OSB JDeveloper の使用の詳細は、第 10 章「JDeveloper を使用した Oracle Service Bus のアウトバウンドおよびインバウンド・プロセスの構成」を参照してください。

## 1.7.8 同期 SAP R/3 イベントの使用

同期 SAP R/3 イベントは、BPEL、BPM および OSB でサポートされています。同期イベントの使用の詳細は、このユーザー・ガイドの次の項を参照してください。

- BPEL: 6.6 項「BPEL を使用した同期 SAP R/3 イベントの処理 (J2CA 構成)」
- BPM: 8.6 項「BPM を使用した同期 SAP R/3 イベントの処理 (J2CA 構成)」
- OSB: 9.5 項「Oracle Service Bus を使用した同期 SAP R/3 イベントの使用 (J2CA 構成)」

## 1.7.9 追加の機能

次に、追加の機能とそれに関連するこのユーザー・ガイド内の項を示します。

- 接続プーリングの接続：3.3 項「接続プーリング」
- インバウンド・スキーマ検証：6.5.1.2 項「イベント通知のための WSDL の生成」の手順 7C を参照してください。

## 1.7.10 その他の機能

次に、その他の機能とそれに関連するこのユーザー・ガイド内の項を示します。

- JMS プロキシ・サービス：9.6 項「Oracle Service Bus を使用した JMS プロキシ・サービスの構成 (J2CA 構成)」
- HTTP プロキシ・サービス：9.7 項「Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成)」
- 例外フィルタの使用：11.1 項「例外フィルタ」
- 資格証明マッピングの構成：
  - 11.2 項「Oracle SOA Suite (BPEL、メディエータまたは BPM) の資格証明マッピング」
  - 11.3 項「JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング」



---

---

## スタート・ガイド

ここでは、Oracle Application Adapter for SAP R/3 を使用するためのクイック・スタート・ガイドを示します。この章は次の項で構成されています。

- [2.1 項「SAP Java Connector \(SAP JCo\) の検証」](#)
- [2.2 項「SAP R/3 ログオン・パラメータの識別」](#)

### 2.1 SAP Java Connector (SAP JCo) の検証

SAP Java Connector (SAP JCo) をインストールしたら、ベスト・プラクティスとして、コネクタが正しくインストールされ、必要なすべての SAP JCo ライブラリ・ファイルが使用可能であることを確認するために、コネクタを検証できます。ここでは、次の項目について説明します。

- [2.1.1 項「Windows プラットフォームでの SAP JCo の検証」](#)
- [2.1.2 項「UNIX プラットフォームでの SAP JCo の検証」](#)

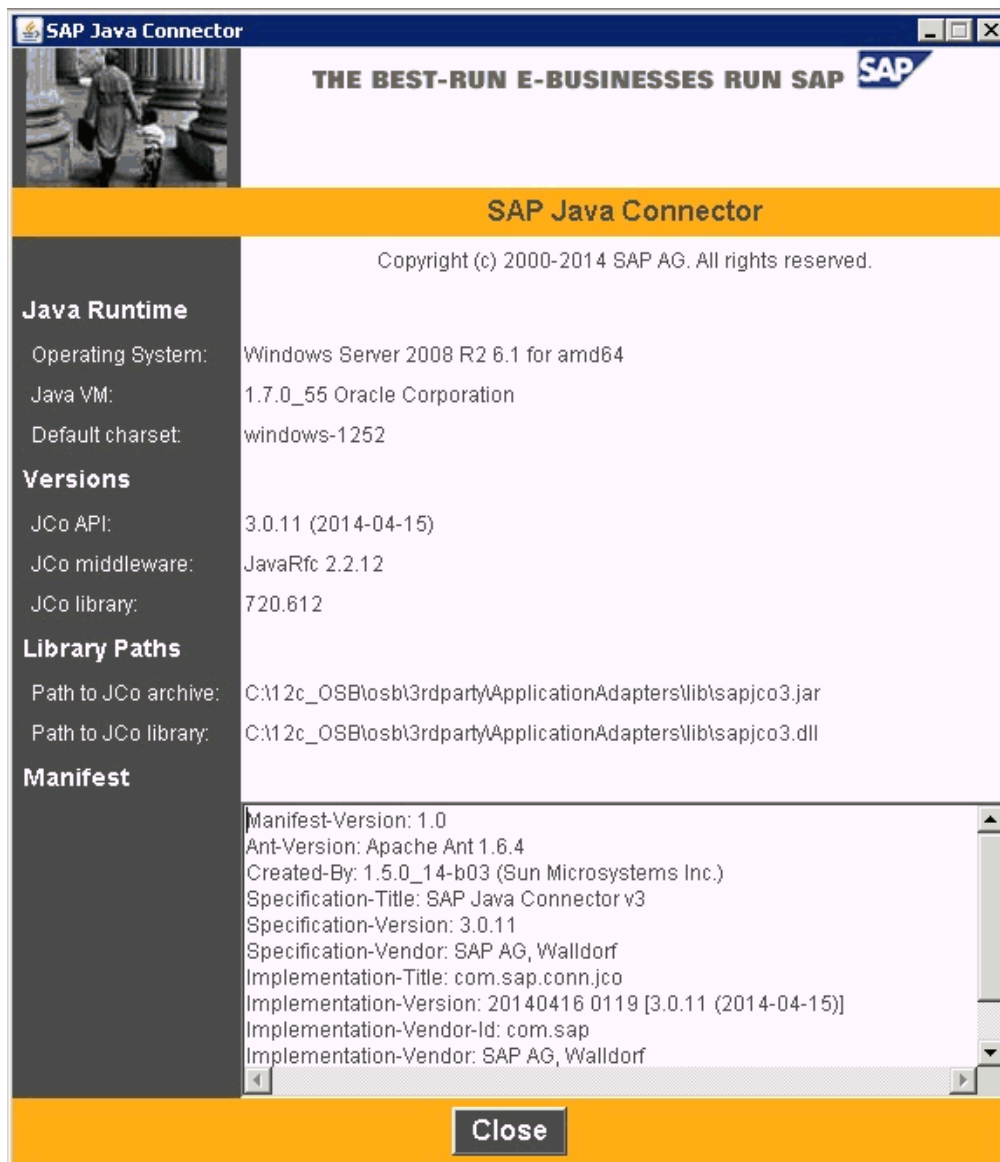
#### 2.1.1 Windows プラットフォームでの SAP JCo の検証

Windows で SAP JCo を検証するには、次のようにします。

1. sapjco3.jar ファイルが配置されているディレクトリにナビゲートします。
2. sapjco3.jar ファイルを右クリックして、コンテキスト・メニューから「プログラムから開く」を選択し、Java 2 Platform Standard Edition バイナリをクリックします。

☒ [2-1](#) に示すように、SAP Java Connector (JCo) のダイアログが表示されます。

図 2-1 SAP Java Connector (JCo) のダイアログ



Windows プラットフォームでの SAP Java Connector に関連した必須情報がすべて提供されます。

3. SAP Java Connector ファイルを確認した後、「Close」をクリックします。

## 2.1.2 UNIX プラットフォームでの SAP JCo の検証

UNIX で SAP JCo を検証するには、次のようにします。

1. UNIX コマンド・プロンプトにナビゲートします。
2. 次のコマンドを実行します。

```
$ java -jar sapjco3.jar -stdout
```

次の例に示すように、ご使用の UNIX プラットフォームでの SAP Java Connector に関連する必須情報がすべて提供されます。

```
-----
| SAP Java Connector (JCo) |
| Copyright (c) 2000-2014 SAP AG. All rights reserved. |
```

```

| Version Information |
-----
Java Runtime:
Operating System: Linux 2.6.32-100.28.5.el6.x86_64 for amd64
Java VM: 1.7.0_55 Oracle Corporation
Default Charset: UTF-8
Versions:
JCo API: 3.0.11 (2014-04-05)
JCo middleware: JavaRfc 2.2.12
JCo library: 720.612
Library Paths:
Path to JCo archive: /rdbms/oracem/Oracle/Middleware/Oracle_
Home/soa/soa/thirdparty/ApplicationAdapters/lib/sapjco3.jar
Path to JCo library: /rdbms/oracem/Oracle/Middleware/Oracle_
Home/soa/soa/thirdparty/ApplicationAdapters/lib/libsapjco3.so
-----
| Manifest |
-----
Manifest-Version: 1.0
Ant-Version: Apache Ant 1.6.4
Created-By: 5.1.028 (SAP AG)
Specification-Title: SAP Java Connector v3
Specification-Version: 3.0.11
Specification-Vendor:SAP AG, Walldorf
Implementation-Title: com.sap.conn.jco
Implementation-Version: 20140415 1946 [3.0.11 (2014-04-15)]
Implementation-Vendor-Id: com.sap
Implementation-Vendor:SAP AG, Walldorf
Main-Class: com.sap.conn.jco.rt.About
-----
$

```

3. UNIX プラットフォームでの SAP Java Connector の情報を確認します。

## 2.2 SAP R/3 ログオン・パラメータの識別

この項では、Oracle Application Adapter for SAP R/3 を使用して SAP R/3 への接続を構成するために使用する、SAP R/3 ログオン・パラメータを識別します。この情報は、参照として使用可能です。ここでは、次の項目について説明します。

- [2.2.1 項「ユーザー・パラメータ」](#)
- [2.2.2 項「システム設定 \(アプリケーション・サーバー\) パラメータ」](#)
- [2.2.3 項「システム設定 \(メッセージ・サーバー\) パラメータ」](#)
- [2.2.4 項「接続プール・パラメータ」](#)
- [2.2.5 項「SAP ゲートウェイ・パラメータ」](#)
- [2.2.6 項「ALE パラメータ」](#)
- [2.2.7 項「グローバル処理パラメータ」](#)
- [2.2.8 項「SNC パラメータ」](#)

### 2.2.1 ユーザー・パラメータ

表 2-1 に、ユーザー・パラメータのリストとその説明を示します。

表 2-1 ユーザー・パラメータ

パラメータ	説明	例	コメント
クライアント	SAP クライアントを識別します。	800	個別のマスター・レコードおよび独自の表のセットを備えた、商用、組織および技術の観点における、SAP システム内の自己完結単位。クライアントとしては、企業グループなどがあります。
ユーザー	SAP ログイン ID。	abc123	ダイアログを使用しないシステム間通信のためのユーザー・タイプ。
パスワード	機密認証情報。	xyz999	SAP システムへのアクセスについてユーザーを識別または認証する、保護された単語または文字列。
言語	言語キー。デフォルトは EN (英語) です。	EN	
コードページ	文字コード・ページ値。	Cp1252	
認証モード	接続の検証方法。	選択 (横の列を参照)。	パスワード - 指定されたフィールドの値を使用します。  ここでのパスワード・パラメータとは、アプリケーション・エクスプローラの「パスワード」パラメータ・フィールドのことです。

## 2.2.2 システム設定 (アプリケーション・サーバー) パラメータ

表 2-2 に、システム設定 (アプリケーション・サーバー) パラメータのリストとその説明を示します。

表 2-2 システム設定 (アプリケーション・サーバー) パラメータ

パラメータ	説明	例	コメント
アプリケーション・サーバー	ABAP アプリケーション・サーバーに接続します。	iwjpsap	R/3 システム内のアプリケーション・プログラムは、アプリケーション・サーバー上で実行されます。メタデータ情報を取得するには、単一のアプリケーション・サーバーへの接続が必要です。
システム番号	アプリケーション・サーバー上の一意のインスタンスを識別します。	00	アプリケーション・サーバーには複数のシステム番号がある場合があります。管理者によって提供されたものを使用します。

## 2.2.3 システム設定 (メッセージ・サーバー) パラメータ

表 2-3 に、システム設定 (メッセージ・サーバー) パラメータのリストとその説明を示します。



表 2-3 システム設定 (メッセージ・サーバー) パラメータ

パラメータ	説明	例	コメント
メッセージ・サーバー	ABAP メッセージ・サーバーに接続します。	iwjpsap	ロード・バランシングの目的で、1つのSAP システムからのアプリケーション・サーバーは、通常、各グループが特定の種類のユーザーに対処するログオン・グループに構成されています。メッセージ・サーバーは、アプリケーション・サーバー間の通信を処理します。これにより、システム内のアプリケーション・サーバー間でリクエストが渡されます。これには、アプリケーション・サーバー・グループおよびグループ内の現在のロード・バランシングに関する情報も含まれます。ユーザーがシステムにログオンすると、この情報を使用して適切なサーバーが選択されます。
R/3 名	アプリケーション・サーバー上の一意のインスタンスを識別します。	P47	システムを識別するために使用される SAP システムのシンボリック名。
サーバー・グループ	ログオン・グループを識別します。		ユーザー ID が属するログオン・グループ。

## 2.2.4 接続プール・パラメータ

表 2-4 に、接続プール・パラメータのリストとその説明を示します。

表 2-4 接続プール・パラメータ

パラメータ	説明	例	コメント
接続プール・サイズ	プールの最大接続数。	2	プールから割当て可能な最大接続数を設定します。
接続タイムアウト	空き接続をオープンしたまま保持する最大時間 (分単位)。	10	接続タイムアウト間隔を超えて使用されなかった接続は、クローズされます。
接続待機時間	空き接続の最大待機時間。	30 (秒)	接続リクエストで空き接続を待機する最大時間を設定します。プールがすべて使用済になり、指定された時間が経過しても使用可能になる接続が依然としてない場合は、キー JCO_ERROR_RESOURCE を含む JCO 例外が生成されます。デフォルト値は 30 秒です。

## 2.2.5 SAP ゲートウェイ・パラメータ

表 2-5 に、SAP ゲートウェイ・パラメータのリストとその説明を示します。

表 2-5 SAP ゲートウェイ・パラメータ

パラメータ	説明	例	コメント
SAP ゲートウェイ・ホスト	SAP ゲートウェイ・サーバーの名前を入力します。	isdsrv2	SAP ゲートウェイは、TCP/IP に基づく CPI-C サービスを SAP 内で実行します。これらのサービスにより、SAP システムと外部プログラムが相互に通信できるようになります。

表 2-5 SAP ゲートウェイ・パラメータ (続き)

パラメータ	説明	例	コメント
SAP ゲートウェイ・サービス	サービス名 (通常、サービス名とシステム番号の組合せ) を入力します。	sapgw00	ゲートウェイ・ホスト上のサービス名。
プログラム ID	SAP ゲートウェイ・サーバーで指定されたプログラム識別子 (大文字と小文字が区別されます)。	S1PROG	システム管理者によって指定された、通信セッション用の一意の識別子。このフィールドに入力する値は、ゲートウェイ上で公開されている値と一致する必要があります。

## 2.2.6 ALE パラメータ

表 2-6 に、ALE パラメータのリストとその説明を示します。

表 2-6 ALE パラメータ

パラメータ	説明	例	コメント
Edi バージョン	ターゲット・システムの ALE バージョンを指定します。	3	リリース 4.0 以上を使用するすべての R/3 パートナ・システムのポート説明には、バージョン 3 (リリース 4.0 以上) を選択する必要があります。  リリース 4.0 未満を使用するすべての R/3 パートナ・システムのポート説明には、バージョン 2 (リリース 3.0/3.1) を選択する必要があります。
IDOC リリース	IDOC 定義がリリースされたバージョンを指定します。	空白または特定の SAP リリース・バージョン (46C など)。	現在のリリース内の IDoc タイプに、以前のリリースからのセグメント定義を割り当てることができます。たとえば、現在の IDoc タイプをサポートしているが現在のセグメント定義はサポートしていない古いリリースをパートナが使用している場合、これが必要になる可能性があります。
IDOC リリース・プロバイダ	アダプタがリリース情報を受信する場所を指定します。	選択 (横の列を参照)。	IDOCDOREL の場合、IDOC ヘッダーの情報を 사용합니다。  SAP リリースの場合、ユーザー・アカウント・ログオンから情報を取得します。  USERINPUT の場合、前述の IDOC リリース・フィールドを使用して情報を取得します。

## 2.2.7 グローバル処理パラメータ

表 2-7 に、グローバル処理パラメータのリストとその説明を示します。

表 2-7 グローバル処理パラメータ

パラメータ	説明	例	コメント
エラー処理	アダプタのエラー処理方法を指定します。	選択 (横の列を参照)。	「エラー・ドキュメントが作成されます」では、例外ドキュメントがエラーの全文とともに出力先に書き込まれます。 「例外がスローされます」では、Java 例外が作成されますが、エラーの全文が表示されるかどうかは基礎となるコンポーネント・エラーによって異なります。
待機中コミット	コミット動作を指定します。	選択 (横の列を参照)。	オフ・デフォルト。 ドキュメント内のアプリケーション・サーバーにコミット・リクエストを送信します。コミット・エラーがある場合は反映されません (最適なパフォーマンス)。 オン (チェック) - ドキュメントで全データベース・サーバーのコミットを待機してから復帰します。コミット・エラーはアダプタ・レベルに反映されます (最遅パフォーマンス)。 推奨設定については、SAP 管理者に問い合せてください。
SAP トレース	SAP Java Connector のトレース動作を有効にします。	選択 (横の列を参照)。	オフ (デフォルト) - 追加モードでは、ハード・エラーのみがトレース・ファイル (dev rfc.trc) に記録されます。 オン・リクエストごとに、個々の rfc*.trc および JCO*.trc が書き込まれます。エラーの検出に役立ちますが、本番システムでは推奨されません。
トレース・レベル	SAP トレースの詳細レベルを指定します。	5	リストで 0 から 10 の範囲の値を選択します。

## 2.2.8 SNC パラメータ

表 2-8 に、SNC パラメータのリストとその説明を示します。

**注意:** このマニュアルで SAP SNC パラメータについて説明する理由は、この機能が SAP Java Connector (SAP JCo) から使用可能であるためです。SNC をアダプタとともに使用する場合、アダプタから直接 SNC を使用するのではなく、SNC が有効になっている状態で SAP ルーターを構成することをお勧めします。これは、最も保護された非常に強い接続方法です。SNC はローカル SAP ホスト・サーバーによって発行される証明書の有効性に依存するため、ローカル SAP サポート・スタッフには、SNC 証明書が有効であること、および SAP ルーターと SAP SNC の構成が正しいことを確認する責任があります。

表 2-8 SNC パラメータ

パラメータ	説明	例	コメント
SNC モード	SNC をアクティブ化するためのフラグ。	1 (オン)	必須。

表 2-8 SNC パラメータ ( 続き )

パラメータ	説明	例	コメント
SNC パートナ	アプリケーション・サーバーの SNC 名を指定します。	p:CN=ABC, O=MyCompany C=US	アプリケーション・サーバーの SNC 名は、プロファイル・パラメータ <code>snc/identity/as</code> にあります。
SNC レベル	接続に使用する保護のレベルを指定します。	選択 ( 横の列を参照 )。	1: 認証のみ 2: 整合性保護 3: プライバシ保護 ( デフォルト ) 8: アプリケーション・サーバー上の <code>snc/data protection/use</code> の値を使用 9: 値を使用 <code>snc/data_protection/max</code> アプリケーション・サーバー デフォルト値 = 3
SNC 名	SNC 名を指定します。	p:CN=SAPJ2EE O=MyCompany, C=US	このパラメータはオプションですが、接続に確実に正しい SNC 名が使用されるようにするには、このパラメータを設定することをお勧めします。
SNC ライブラリ・パス	外部ライブラリのパスおよびファイル名を指定します。	C:\SAP J2EE_ Engine\SAPCryp tolib\sap crypto.dll	デフォルトは、環境変数 <code>SNC LIB</code> で定義されたシステム定義ライブラリです。

---

---

## SAP Java Connector 3.0xx の考慮事項

この章では、SAP Java Connector (SAP JCo) 3.0xx に関連した、Oracle Application Adapter for SAP R/3 の使用方法に関する新しい考慮事項について説明します。次の項で構成されています。

- 3.1 項「サポートされているプラットフォーム」
- 3.2 項「接続管理」
- 3.3 項「接続プーリング」
- 3.4 項「単一サーバーのサポート」
- 3.5 項「サーバー・スレッド」
- 3.6 項「パラメータおよび表パラメータの変更」
- 3.7 項「トレース・レベル・パラメータ」

---

---

**注意：** SAP JCo 3.0xx の xx は、SAP Java Connector の特定のバージョンを示します。

SAP ポリシーでは、現在リリースされているバージョンの SAP JCo のみがサポートされているバージョンとして規定されています。

サポートについて問題のあるユーザーは、より新しいバージョンの SAP JCo (存在する場合) を使用してテストし、問題がすでに解決されていることを確認することが必要な場合があります。

また、Oracle JRockit JVM は SAP JCo ではサポートされていません。

---

---

### 3.1 サポートされているプラットフォーム

SAP JCo 3.0xx では、UNIX プラットフォームでの 32 ビットのサポートが行われなくなりました。現在、32 ビットの SAP JCo は、Linux (Intel プロセッサ搭載) および Windows (Intel プロセッサ搭載) でのみサポートされています。その他すべてのプラットフォームでは、完全な 64 ビット処理が必要となります。一部の JVM 設定では、JVM が SAP JCo と互換性のない混合モードとして解釈されます。一部のプラットフォームでは、JVM で完全な 64 ビット処理を可能にするために、*d64* という特定のフラグが必要になります。

SAP JCo 3.0xx でサポートされているプラットフォームの詳細は、『*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストレーション・ガイド*』を参照してください。

各オペレーティング・システムに対応した、サポートされている JVM の情報については、SAP Service Marketplace で SAP ノート #1077727 を参照してください。サポートされている JVM のリストに含まれていない場合、その JVM は SAP R/3 でサポートされていません。

---

---

**注意:** Oracle JRockit JVM は SAP JCo 3.0.11 ではサポートされていません。

---

---

## 3.2 接続管理

SAP Java Connector (SAP JCo) 3.0xx では、アダプタは、特定の JCo インスタンスが、そのインスタンスから SAP システムへのすべての接続を管理することを支援します。各チャンネルは SAP 固有のパラメータで構成され、これらは SAP システム接続の SAP JCo 宛先となります。

## 3.3 接続プーリング

この機能により、mySAP アダプタの共有接続プーリングが実装されます。ターゲットという用語は、mySAP30 アダプタにおいて、特定の SAP インスタンス (アプリケーション・サーバーまたはメッセージ・サーバー)、特定のログオンおよびパスワード資格証明の特定に使用される、SAP 接続パラメータの保存されたセットを示すために使用されます。(アプリケーション・サーバーのターゲットには、アプリケーション・サーバー ID およびシステム番号が含まれ、メッセージ・サーバーのターゲットには、メッセージ・サーバー・パラメータ、ログオン・グループ・パラメータおよびその他のパラメータが含まれます。ターゲット・パラメータの完全なリストは、mySAP 30 ユーザー・ガイドを参照してください)

ユーザーが SAP ユーザー / サーバー・パラメータを入力および保存すると、アプリケーション・エクスプローラでターゲットが成されます。このターゲットは、ランタイムに、関連する SAP システムを識別し、接続プールを介して接続するために使用されます。

接続プールでは、接続には同じユーザー ID とパスワードが使用されますが、トランザクションごとに SAP ログオンとログオフのオーバーヘッドが発生しないように、接続ハンドルのプールがアクティブな状態で維持されます。接続ハンドルは何度も再利用され、そのサイズはシステムの負荷に応じて動的に変動します。SAP システムとの接続を確立するタスクは、クライアントとサーバーの両方にとってリソースを消費するうえに時間がかかりますが、既存の接続を再利用すると時間が大幅に短縮され、クライアントとサーバーにかかるリソースの負担が軽減されます。

共有 JCo 機能を使用すると、同じ Java 仮想マシン内の複数のアダプタ・インスタンス間でターゲットを共有できます。アプリケーション・サーバーでのリソース消費を軽減し、アプリケーションの応答時間を高速化するために、この機能を有効にすることをお勧めします。

### 説明

SAP 接続プールは、指定されたパラメータに従って開かれる接続を作成するよう求める、SAP Java Connector へのリクエストとして発行されます。アダプタが接続を割り当てたり予約するのではなく、すべての接続処理は SAP Java Connector によって行われます。

単一の SAP システムについて、複数のターゲットまたは単一のターゲットを構成できます。複数の SAP システムがインストールされており、システムごとに単一または複数のターゲットを作成する場合、複数の SAP システムに同時に接続でき、各 SAP システム・ターゲットには、そのターゲット・システムを対象とした基本接続プールが含まれます。

### 共有 JCo プール機能の制限

共有 JCo 機能はアダプタ全体の機能です。この機能を有効にすると、同じターゲット名を使用するすべての接続で、同じ接続プールが共有されます。

アダプタは、管理対象接続インタフェース内のアプリケーション・サーバーによって管理され、SAP JCo 接続プールを介して SAP と通信しますが、ここでは、SAP JCo プールについてのみ説明しています。

接続プールでは同じユーザーとパスワードが使用されるため、適切な SAP アプリケーション認可を取得するには、機能領域およびユーザー ID 別に接続プールを分割することが望ましい場合があります。

たとえば、会計および財務で同じ SAP システムが共有される場合でも、ユーザー ID とパスワードは各ターゲットで異なり、各ターゲットは SAP 機能領域に従って割り当てられます。

拡張アダプタ・ターゲット・オプション (IDoc バージョン、IDOC リリース・プロバイダなど) を使用し、アプリケーション・タイプごとに異なる設定が必要な場合、これらのパラメータが動的ではないため、2つのターゲットを作成する必要があります。

### 共有 JCo 機能なしでの接続の使用

共有 JCo を使用しない場合、アプリケーション・サーバーによって割り当てられた各アダプタ・インスタンス (スレッドまたはプロセスのいずれか) は、接続プールおよびリポジトリ接続を開こうとします。接続をリクエストするアプリケーション・サーバー・リソースが削除されたり終了した場合、接続プールとそれに伴うすべてのリソースは、タイムアウト・チェックによる削除対象としてマークされます。複数のアプリケーション・サーバー・スレッドを含む共有 JCo を使用しない場合、多数のアダプタ・インスタンスを開いてアプリケーション・サーバー・リソースを消費できます。

共有 SAP JCo を使用しておらず、接続プールの最大接続数が 60 にマークされており、10 個のアプリケーション・サーバー・スレッドがアダプタ呼出しに対してインスタンス化されている場合、アダプタ・インスタンスごとに 60 個の接続の取得を試みることができます。

### 共有 JCo を有効にする方法 (推奨オプション)

プロセスで作成または使用するターゲット (複数可) を識別します。プールを消費するすべてのリソースに十分な接続ハンドルが使用可能となるよう、ターゲットを作成または変更します。

#### アプリケーション・サーバーの手順

1. アプリケーション・サーバーが実行中の場合には停止します。
2. `ra.xml` ファイル (<ADAPTER\_HOME>\iwafjca.rar\META-INF) で、`ShareJCO` オプションを `true` に設定します。
3. 必要なロギングを有効にして、サーバーを再起動します。
4. WebLogic コンソールから、`iwafjca.rar` ファイルおよび `iwafjca.war` ファイルを更新します。
5. 更新オプションが適切に動作しない場合は、`iwafjca.rar` ファイルおよび `iwafjca.war` ファイルをアンデプロイして再デプロイします。これらのファイルをデプロイする方法の詳細は、『*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストール・ガイド*』を参照してください。

同じ SAP システムおよびユーザー・ログオン資格証明について異なるターゲットが存在する場合、わかりやすくするため、およびパフォーマンスを最大にするために、ターゲット名を統合することを検討します。

### 検証 (オプション)

1つのアダプタ・ターゲットを共有する2つのアウトバウンド・プロセスを構成し、このターゲットで「SAP トレース」パラメータを有効にすることにより、クイック検証プロセスを実行できます。トレースには、同じ SAP 宛先へのすべてのリクエストを処理する、まったく同じ接続プールが表示されます。検証は開発システムまたはテスト・システムで実行し、本番システムで SAP トレースを実行しないことをお勧めします。

### 接続パラメータのチューニング (オプション)

接続プールは、アダプタ・ターゲットでのパラメータによって決定されます。接続プールは動的エンティティであり、プールは、リクエストが追加されると指定された最大サイズになるまで増加し、新しいリクエストが受信されない場合には縮小されます。

接続プール・サイズ: プールに設定可能な最大サイズ (接続は、1 から開始してこの制限まで増加します。接続が必要なくなった場合、プールのサイズは初期サイズまで縮小されます)。

接続タイムアウト (分): 未使用の空き接続を削除するまでプールに保持できる最大時間。

接続待機時間 (秒): 使用可能な接続が存在しない場合に、リクエストが接続を待機する最大時間。この待機時間の期限が切れるまでに接続ハンドルが戻されない場合、リソース例外がトリガーされ、リクエストに対する処理は停止します。

システム・トポロジによっては、SAP に接続し、必要なメタデータを取得してリクエストを実行するために、プールから接続すれば 10 ミリ秒以下で接続できるのに対し、新しい接続を使用すると最大で 300 ミリ秒以上かかる場合があります。これにより、アプリケーション全体のパフォーマンスに多大な影響を及ぼす可能性があります。

接続をチューニングするには、1 つずつパラメータを変更し、アプリケーション・パフォーマンスに及ぼす影響を確認します。不適切な接続チューニングの結果には、次のようなものがあります。

- 十分な接続が割り当てられていないと、アプリケーション・サーバー上でリクエストが待機中になる可能性があります。
- 接続タイムアウトが短すぎると、処理時間が長期化します。これは、新しい接続を開く際、タイムアウト時間が長すぎると、未使用接続がプール内に保持されるためです。
- 待機時間が短すぎると、リソース例外が過剰に発生する可能性があります。

より細かい粒度でチューニングするには、SAP JCo トレースを有効にし、接続リクエスト結果を確認することを検討します。

## 3.4 単一サーバーのサポート

SAP Java Connector (SAP JCo) 3.0xx を使用すると、SAP JCo のインストールごとに、SAP JCo サーバーの単一インスタンスが許可されます。Oracle Application Adapter for SAP R/3 では、ポート、チャンネルおよびサーバー・スレッド間でマッピングを行います。変更はエンドユーザーにとって透過的ですが、ロード・バランスされたマルチスレッド・インストールにおいては、アダプタのサーバー・スレッド件数と SAP ゲートウェイのアクティブなサーバー・スレッド件数が一致するよう、特に注意が必要です。詳細は、ロード・バランシングおよびサーバー・スレッドに関する包括的な情報が公開されている、SAP R/3 ヘルプ・ファイル「Configuring the SAP Gateway」で参照できます。

## 3.5 サーバー・スレッド

ここでは、ロード・バランシングのコンテキスト内でのサーバー・スレッドの動作について説明します。次の項で構成されています。

- [3.5.1 項「一意性」](#)
- [3.5.2 項「ロード・バランシングのためのアダプタ構成」](#)
- [3.5.3 項「ロード・バランシングのための SAP 構成」](#)

### 3.5.1 一意性

SAP ゲートウェイ・サーバー、SAP ゲートウェイ・サービスおよびプログラム ID を組み合わせて一度にアクティブにできるのは、1 つのサーバーのみです。このチャンネルを介して送信されたすべてのメッセージは、メッセージ・スキーマによってルーティングされないかぎり、同じアダプタ宛先で受信されます。

### 3.5.2 ロード・バランシングのためのアダプタ構成

デフォルトでは、SAP R/3 アウトバウンド・ゲートウェイにより、SAP R/3 以外のシステムの RFC 宛先ごとにアクティブな接続が 1 つ作成されます。スループットを最大にするには、スレッド件数パラメータを 3 に設定します (RFC 宛先ごとに 3 つのサーバー・スレッド)。このパラメータは、チャンネル構成時のアプリケーション・エクスプローラの「詳細」タブにあります。通常、これにより、リクエストのリスニング、現在のリクエストの処理および処理からのクリーンアップ・タスクに、それぞれ 1 つのスレッドを使用できます。



### 3.5.3 ロード・バランシングのための SAP 構成

特定の RFC 宛先で大量のトラフィックが必要な場合は、SAP でのスレッド出力サイズを増加し、スレッド件数を増加して、ゲートウェイ・スレッド対サーバー・スレッドの比率が常に 1:3 の比率となるよう維持します。たとえば、ゲートウェイのアウトバウンド・サーバー・チャンネル側に宛先 X に対するアクティブ・スレッドが 3 つ存在する場合、この時点でのサーバー・スレッド件数は 9 となります。

ゲートウェイのロード・バランシングに加えて、SAP ゲートウェイ・スレッドの数も増加できます。詳細は、SAP R/3 管理者に問い合わせてください。

1 つの SAP ゲートウェイ (個別の JVM を使用) に、同じ名前の複数のプログラム ID が登録されている場合、SAP R/3 システムでのゲートウェイ・プロファイル内のパラメータによっては、このゲートウェイをロード・バランシング・シナリオに使用可能です。詳細は、SAP R/3 システム管理者に問い合わせてください。ただし、プロファイルは 0 (ロード・バランシングなし)、1 (最低カウンタ) または 2 (最低ロード) のいずれかに設定されています。

SAP R/3 システム管理者は、各状況に応じた宛先の構成方法について必要な情報を提供できます。ロード・バランシングが有効になっておらず、同じプログラム ID を含む複数のサーバー・チャンネルを Oracle スキーマ検証を実行せずに登録する場合、チャンネルまたは宛先に対するルーティングが行われなため、不適切なチャンネルで IDoc を受信する可能性があります。

## 3.6 パラメータおよび表パラメータの変更

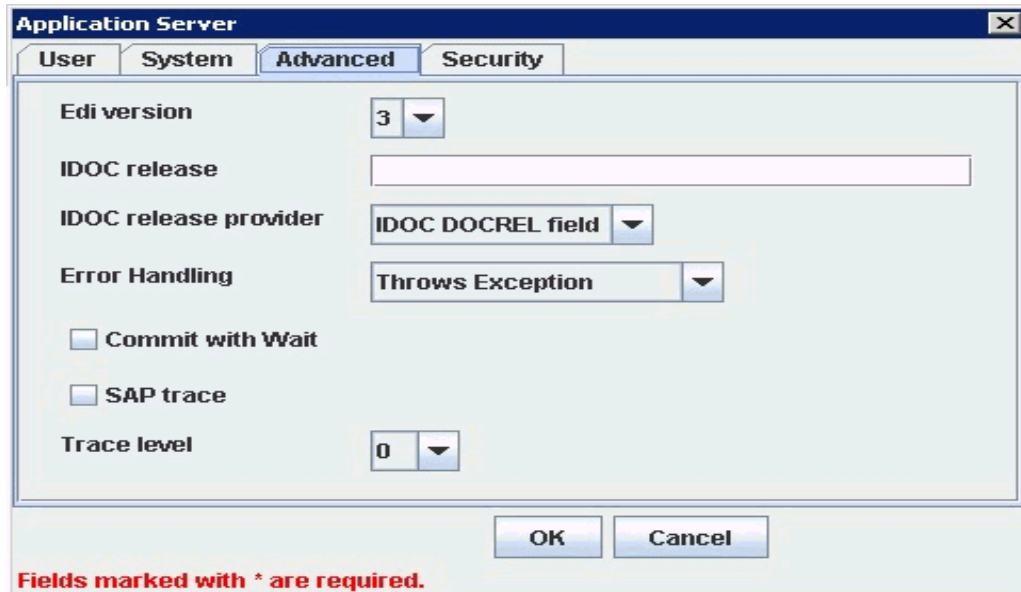
SAP 変更パラメータでは、ネスト構造 (列自体が構造である表) や固定長のないデータ (文字列データ型など) を含む、ほとんどすべての SAP R/3 タイプを使用できます。変更パラメータは、関数モジュールへの入力や関数モジュールからエクスポートのどちらにも使用できます。これは、特定の SAP 関数の設計者の要件によって異なります。表パラメータは、定義された長さの固定列と必須ヘッダー行を含むフラット・データ構造 (リレーショナル) に制限されています。変更パラメータでは、ネスト構造 (列自体が構造である表) や固定長のないデータ (文字列データ型など) を含む、ほとんどすべての SAP R/3 タイプを使用できます。変更パラメータは、関数モジュールへの入力や関数モジュールからエクスポートのどちらにも使用できます。これは、特定の SAP 関数の設計者の要件によって異なります。

変更パラメータを使用する関数の XML スキーマによって、パラメータが個別のカテゴリとしてリストされることはありません。ただし、表パラメータと同様、リクエストとレスポンスの XML スキーマに表示されます。パラメータがオプションとしてマークされている場合、関数の設計に応じて、値を指定する必要はありません。オプションのインジケータが、アプリケーション・エクスプローラと XML スキーマに表示されます。変更パラメータの使用の詳細は、SAP R/3 のドキュメントを参照してください。

## 3.7 トレース・レベル・パラメータ

Oracle Application Adapter for SAP R/3 (SAP JCo 3.0xx を使用) にはトレース・レベルと呼ばれるパラメータが含まれており、このパラメータは、[図 3-1](#) に示すように、アプリケーション・エクスプローラの「詳細」タブにあります。

図 3-1 「詳細」タブ



このパラメータは、SAP トレースの詳細レベルを示します。リストで 0 から 10 の範囲の値を選択します。使用される共通の値は、次のとおりです。

- 0 - なし
- 1 - エラーと警告
- 2 - 実行パス、エラーおよび警告
- 3 - フル実行パス、エラーおよび警告
- 4 - 実行パス、情報メッセージ、エラーおよび警告
- 6 - フル実行パス、情報メッセージ、エラーおよび警告
- 7 - デバッグ・メッセージ、フル実行パス、情報メッセージ、エラーおよび警告
- 8 - 詳細デバッグ・メッセージ、フル実行パス、情報メッセージ、エラーおよび警告

null、stdout および stderr のパス値が許可されます。既存のパスも許可されます。パスが null、stdout または stderr と等しい場合、SAP JCo により、標準出力 (エラー) ストリームへのトレースが生成されます。少なくとも 1 つの JCoTraceListener が登録されている場合、トレース出力はリスナーにのみ送信されます。SAP JCo トレースをファイルにリダイレクトするには、パス値を既存のディレクトリに設定する必要があります。

次の環境変数を ORACLE\_JAVA\_INVOKE\_HOME で設定する必要があります。

```
Java Djco.trace_path=drive:/directory
```

次に、Solaris プラットフォームでのパスの例を示します。

```
/path1/dir1/
```

次の変数は、オペレーティング・システム・レベルで設定できます。

```
RFC_TRACE=1  
CPIC_TRACE=(1,2,3)  
RFC_TRACE_DIR=  
CPIC_TRACE_DIR=
```

ここで、**1**はエラー・トレース、**2**は完全プロセス・トレース、**3**は完全プロセス・トレースの短いデータ・トレースです。

ホスト環境では特に、環境設定を確認することをお勧めします。



---

---

# 構成 Oracle Application Adapter for SAP R/3

この章では、Oracle Adapter Application Explorer (アプリケーション・エクスプローラ) を使用して、SAP R/3 システムに接続するターゲットの定義、システム・オブジェクトの表示および XML スキーマと Web サービスの作成を行う方法について説明します。また、イベント・アダプタの構成方法についても説明します。

この章は次の項で構成されています。

- 4.1 項「アプリケーション・エクスプローラの起動」
- 4.2 項「リポジトリ設定の構成」
- 4.3 項「リポジトリ構成の作成」
- 4.4 項「SAP R/3 の接続 (ターゲット) の確立」
- 4.5 項「アプリケーション・システム・オブジェクトの表示」
- 4.6 項「XML スキーマの作成」
- 4.7 項「WSDL の生成 (J2CA 構成のみ)」
- 4.8 項「Web サービスの作成およびテスト (BSE 構成のみ)」
- 4.9 項「イベント・アダプタの構成」
- 4.10 項「SAProuter サービスの構成」

## 4.1 アプリケーション・エクスプローラの起動

アプリケーション・エクスプローラを起動するには、次のようにします。

1. Oracle WebLogic Server (アプリケーション・エクスプローラがデプロイされている) が起動されていることを確認します。
2. コマンド・プロンプトを開きます。
3. 次のディレクトリにナビゲートします。

```
<ORACLE_HOME>\user_projects\domains\base_domain\bin
```

4. `setDomainEnv.cmd` (Windows) または `../setDomainEnv.sh` (UNIX/Linux) を実行します。

このコマンドにより、Oracle WebLogic Server 環境のアプリケーション・エクスプローラのクラス・パスと他の環境変数が設定されます。また、これにより、アプリケーション・エクスプローラは Oracle WebLogic Server API にアクセスし、WSDL ファイルを Oracle Service Bus (OSB) コンソールにパブリッシュすることが可能になります。

5. コマンド・プロンプトを閉じないでください。

6. 次のディレクトリにナビゲートします。

```
<ADAPTER_HOME>\tools\iwae\bin
```

7. `ae.bat` (Windows) または `iwae.sh` (UNIX/Linux) を実行して、アプリケーション・エクスプローラを起動します。

アプリケーション・エクスプローラが起動します。これで、SAP R/3 システムへのターゲットを定義する準備ができました。

---

---

**注意:** UNIX または Linux プラットフォーム上で `iwae.sh` ファイルを実行する前に、権限を変更する必要があります。次に例を示します。

```
chmod +x iwae.sh
```

---

---

## 4.2 リポジトリ設定の構成

リポジトリには、構成の詳細、アダプタ・ターゲットおよびチャネルに関する情報や、その他の構成情報が保持されます。BSE および J2CA リポジトリ設定の構成方法の詳細は、『*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストレーション・ガイド*』を参照してください。

## 4.3 リポジトリ構成の作成

アプリケーション・エクスプローラを Oracle Application Adapter for SAP R/3 とともに使用するには、まず、リポジトリ構成を作成する必要があります。作成できるリポジトリ構成は Web サービスと J2CA の 2 種類で、アダプタがデプロイされているコンテナによって決まります。

ここでは、次の項目について説明します。

- [4.3.1 項「BSE の構成の作成」](#)
- [4.3.2 項「J2CA の構成の作成」](#)
- [4.3.3 項「BSE または J2CA 構成への接続」](#)

デザインタイムの間、リポジトリは、アプリケーション・エクスプローラを使用してアダプタ接続の構成、EIS オブジェクトの参照、サービスの構成および EIS イベントをリスニングするリスナーの構成を行うときに作成されたメタデータを格納するために使用されます。リポジトリ内の情報はランタイムにも参照されます。

Web サービスおよび BSE は、同じタイプのデプロイメントを参照します。詳細は、1-1 ページの「[アダプタの機能](#)」を参照してください。

### 4.3.1 BSE の構成の作成

アプリケーション・エクスプローラを使用して BSE の構成を作成するには、まず、新規構成を定義する必要があります。

この項では、次の項目について説明します。

- [4.3.1.1 項「BSE の新規構成の定義」](#)

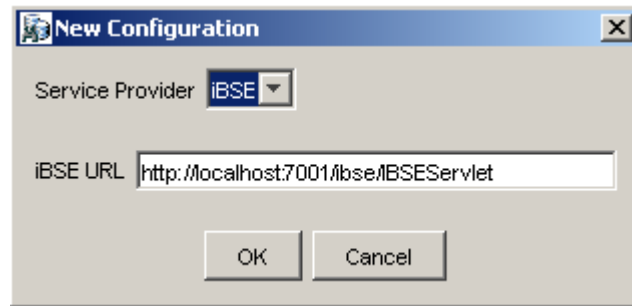
#### 4.3.1.1 BSE の新規構成の定義

BSE の新規構成を定義するには、次のようにします。

1. アプリケーション・エクスプローラを起動します。
2. 「構成」を右クリックして、「新規」を選択します。  
「新規構成」ダイアログが表示されます。

- 新規構成の名前を入力し (SampleConfig など)、「OK」をクリックします。

図 4-1 「新規構成」ダイアログ



- 図 4-1 に示すように、「サービス・プロバイダ」リストで **iBSE** を選択します。
- 「iBSE URL」フィールドで、デフォルトの URL を受け入れるか、次の形式で別の URL を指定します。

`http://host name:port/ibse/IBSEServlet`

ここで、*host name* は Oracle WebLogic Server が常駐するシステムであり、*port* は管理対象 Oracle WebLogic Server (soa\_server1 など) の HTTP ポート番号です。

- 「OK」をクリックします。
- 図 4-2 に示すように、ルートの「構成」ノードの下に新規構成を表すノードが表示されます。

図 4-2 新規構成ノード



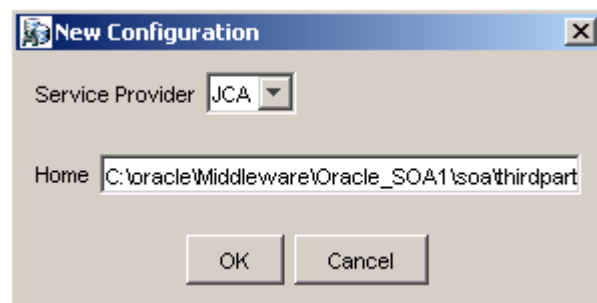
## 4.3.2 J2CA の構成の作成

アプリケーション・エクスプローラを使用して Oracle Adapter J2CA の構成を作成するには、まず、新規構成を定義する必要があります。

J2CA の新規構成を定義するには、次のようにします。

- アプリケーション・エクスプローラを起動します。
- 「構成」を右クリックして、「新規」を選択します。  
「新規構成」ダイアログが表示されます。
- 新規構成の名前を入力し (SampleConfig など)、「OK」をクリックします。

図 4-3 「新規構成」ダイアログ



4. 図 4-3 に示すように、「サービス・プロバイダ」リストで「JCA」を選択します。
5. 「OK」をクリックします。  
 図 4-4 に示すように、ルートの「構成」ノードの下に新規構成を表すノードが表示されます。

図 4-4 「SampleConfig」ノード



Oracle Adapter J2CA 構成フォルダは、アダプタのインストールに基づいた場所に格納されます。

`<ADAPTER_HOME>\config\configuration_name`

`configuration_name` は、作成した構成の名前です (SampleConfig など)。

### 4.3.3 BSE または J2CA 構成への接続

新規構成に接続するには、次のようにします。

1. 接続先の構成を右クリックします (SampleConfig など)。
2. 「接続」を選択します。

「アダプタ」、「イベント」および「ビジネス・サービス」(Web サービスとも呼ばれる)の各ノードが表示されます。「ビジネス・サービス」ノードは、BSE 構成でのみ使用可能です。J2CA 構成に接続されている場合、「ビジネス・サービス」ノードは表示されません。

BSE 構成を使用している場合、イベントは適用されません。J2CA 構成を使用している場合にのみイベントを構成できます。

図 4-5 に、「SampleConfig」という名前の BSE 構成の例を示します。

図 4-5 SampleConfig



- 「アダプタ」ノードは、SAP R/3 とのインバウンド相互作用の作成に使用します。たとえば、SAP R/3 を更新するサービスを構成するには、「アダプタ」ノードの SAP ノードを使用します。
- 「イベント」ノードは、SAP R/3 内のイベントをリスニングするリスナーの構成に使用します。
- 「ビジネス・サービス」ノード (BSE 構成でのみ使用可能) は、Web サービスのテストに使用します。「ビジネス・サービス」ノードのセキュリティ機能を使用することによって Web サービスのセキュリティ設定を制御することも可能です。

これで、SAP R/3 への新規ターゲットを定義できます。



## 4.4 SAP R/3 の接続 ( ターゲット ) の確立

アプリケーションの定義には、Oracle Application Adapter for SAP R/3 のターゲットの追加も含まれます。アプリケーション・エクスプローラでターゲットを設定するには、そのターゲット固有の情報が必要です。

ここでは、次の項目について説明します。

- 4.4.1 項 「SAP R/3 へのターゲットの定義」
- 4.4.2 項 「定義済 SAP R/3 ターゲットへの接続」
- 4.4.3 項 「SAP R/3 への接続の管理」

使用可能なビジネス関数を参照するには、まず、SAP R/3 へのターゲットを定義する必要があります。定義したターゲットは自動的に保存されます。SAP R/3 システムへの接続は、アプリケーション・エクスプローラを起動するたびに、または切断の後に確立する必要があります。

アプリケーション・エクスプローラを起動すると、インストールされているアダプタに基づいて、アプリケーション・エクスプローラでサポートされているアプリケーション・システムが ( ノードとして ) 左ペインに表示されます。

### 4.4.1 SAP R/3 へのターゲットの定義

初めて SAP R/3 に接続するには、新規ターゲットを定義する必要があります。Oracle Application Adapter for SAP R/3 は、SAP R/3 の標準セキュリティと SNC の追加プロトコルをサポートしています。SAP R/3 アプリケーション・サーバーに接続されると、アプリケーション・セキュリティは、ユーザー ID、ロールおよびプロファイルにより管理されます。SAP アプリケーション・セキュリティの詳細は、該当の SAP ドキュメントを参照してください。

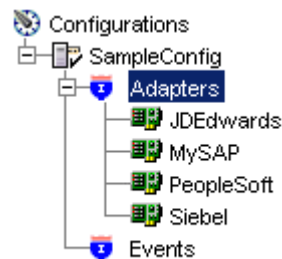
J2CA 構成で作業している場合、ターゲットの作成、更新および削除にはアプリケーション・サーバーの再起動が必要です。また、ターゲットの作成後、ターゲットへの接続後、SAP R/3 ビジネス・オブジェクト用の WSDL の生成後にもアプリケーション・サーバーの再起動が必要です。さらに、アプリケーション・サーバーを再起動する前に、必ずアプリケーション・エクスプローラを終了してください。

ターゲットを定義するには、次のようにします。

1. 左ペインで、[図 4-6](#) に示すように「アダプタ」ノードを展開します。

インストールされているアダプタに基づいて、アプリケーション・エクスプローラでサポートされているアプリケーション・システムがノードとして表示されます。

図 4-6 「アダプタ」ノード



2. [図 4-7](#) に示すように、「MySAP」ノードを右クリックして「ターゲットの追加」を選択します。

図 4-7 「MySAP」ノード



「ターゲットの追加」ダイアログが表示されます。次の情報を指定します。

- a. 「名前」フィールドに、「SAPTarget」などのわかりやすい名前を入力します。
- b. 「説明」フィールドにターゲットの説明を入力します (オプション)。
- c. 「タイプ」リストで、接続先のターゲットのタイプを選択します。サポートされるターゲット・タイプには、「メッセージ・サーバー」や「アプリケーション・サーバー」(デフォルト) などがあります。

---

**注意：** ロード・バランシングの目的で、1つの SAP R/3 システムからのアプリケーション・サーバーは、通常、各グループが特定の種類のユーザーに対処するログオン・グループに構成されています。各グループ内のアプリケーション・サーバーは、最小負荷方針に従ってユーザーに割り当てられます。このロード・バランシングはメッセージ・サーバーによって行われます。各 SAP R/3 システムにはメッセージ・サーバーが1つのみあり、特定のメッセージ・サーバー・ポートで TCP を介してアクセスできます。

---

3. 「OK」をクリックします。

☒ 4-8 に示すように、「アプリケーション・サーバー」ダイアログが表示されます。

図 4-8 「アプリケーション・サーバー」ダイアログ

使用可能なタブは次のとおりです。

- ユーザー (必須)
- システム (必須)
- 詳細
- セキュリティ

4. 「ユーザー」タブ (必須) で、次の表の情報に基づいて、SAP R/3 ターゲットに適切な情報を入力します。

表 4-1 「ユーザー」タブのパラメータ

ターゲット・パラメータ	説明
クライアント	クライアント通信用に SAP R/3 アプリケーションに対して定義されたクライアント番号。
ユーザー	SAP R/3 アプリケーションの有効なユーザー ID。
パスワード	SAP R/3 アプリケーションの有効なパスワード。
言語	言語キー。デフォルトは EN (英語) です。
コードページ	文字コード・ページ値。
認証モード	SAP R/3 システムに接続する際に使用する認証モード。デフォルトでは、リストの「パスワード」が選択されています。

詳細は、SAP R/3 システムのドキュメントを参照してください。

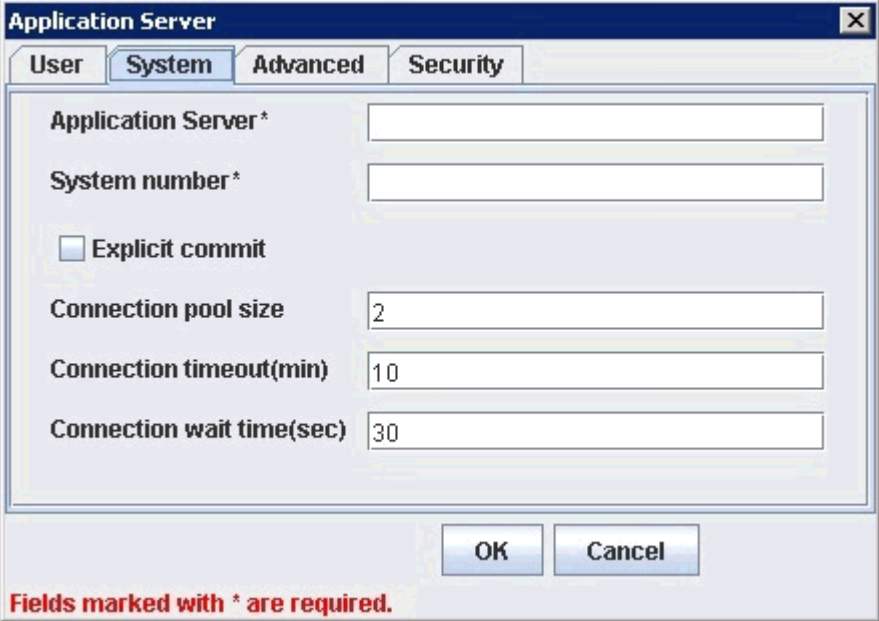

5.  4-9 に示すように、「システム」タブ (必須) で、この項の情報に基づいて、SAP R/3 ターゲットに適切な情報を入力します。

図 4-9 「システム」タブ



Application Server

User System Advanced Security

Application Server\*

System number\*

Explicit commit

Connection pool size

Connection timeout(min)

Connection wait time(sec)

OK Cancel

Fields marked with \* are required.

「システム」タブでは、アプリケーション・サーバー名、システム番号、接続先の SAP R/3 システムに関する接続プーリング情報を指定できます。

表 4-2 「システム」タブのパラメータ

ターゲット・パラメータ	説明
アプリケーション・サーバー	SAP R/3 アプリケーションをホストしているシステムのホスト名または IP アドレス。
システム番号	クライアント通信用に SAP R/3 に対して定義されたシステム番号。
明示的なコミット	アダプタにより実行される関数呼出しの後に常に BAPI_TRANSACTION_COMMIT 関数が呼び出されるようにするには、このオプションを選択します。このオプションにより処理時間が長くなる可能性があるため、明示的なコミット機能が必要な場合にのみこの機能を使用することをお勧めします。このオプションにより非同期コミットが可能になります。BAPI_TRANSACTION_COMMIT 関数は、呼び出されて即座に戻ります。データベース・コミットが失敗する可能性は依然として残りますが、通知は受信されません。このオプションは、COMMIT WORK ABAP 文と同等です。
接続プール・サイズ	<p>SAP R/3 が Web サービス・コールに使用できるようにする、プール内のクライアント接続の数。デフォルトでは、接続プール・サイズとして 2 が使用可能です。</p> <p>重要: デフォルト値が 1 の場合、接続プールは作成されません。かわりに、順次処理による単一の SAP R/3 接続が共有されます。プールされた接続により、パラレル処理による SAP R/3 への複数の接続が呼び出されます。</p> <p>アプリケーション・エクスプローラを使用して Web サービスを作成する場合、ランタイム中に、この接続プール・サイズ値が Web サービスによって使用されます。結果として、目的に対して十分な接続プール・サイズが確保されます。</p>
接続タイムアウト (分)	接続プールのタイムアウト値 (分単位)。デフォルト値は 10 分です。
接続待機時間 (秒)	<p>接続プールの待機時間 (秒)。デフォルト値は 30 秒です。</p> <p>ERP サーバーへの接続は、クライアントとリモート・サーバーの両方で貴重なリソースを消費します。接続のプールを作成し、リソースと時間の制約を最小限に抑えることができます。プールのサイズを見積もる際、消費するサーバー・リソースの量、受信するドキュメントの数とサイズおよび Java 仮想マシンのサイズによって、プール・サイズを計算できます。SAP ドキュメント『Memory Management (BC-CST-MM)』の項で、SAP R/3 システムで必要となるリソースについて詳細に説明されています。</p>

6. 「詳細」タブ (オプション) で、この項の情報に基づいて、SAP R/3 ターゲットに適切な情報を入力します。

図 4-10 に示すように、「詳細」タブでは、EDI および IDoc バージョンを指定して、エラー処理を構成できます。

図 4-10 「詳細」タブ

表 4-3 「詳細」タブのパラメータ

ターゲット・パラメータ	説明
Edi バージョン	アダプタとともに使用する Electronic Data Interchange (EDI) ドキュメントのバージョン。バージョン 3 がデフォルト値です。
IDOC リリース	接続に使用する IDOC バージョニング。
IDOC リリース・プロバイダ	接続のための IDOC リリース・プロバイダ。リストで IDOC DOCREL フィールド (デフォルト)、SAP リリースまたはユーザー入力を選択します。
エラー処理	<p>例外の発生時の処理として、リストで「エラー・ドキュメントが作成されます」または「例外がスローされます」を選択できます。詳細なエラー・メッセージを受け取るには、「エラー・ドキュメントが作成されます」を選択します。</p> <p>次の原則に従います。</p> <ul style="list-style-type: none"> <li>■ アプリケーションが <b>Java 中心</b> の場合は、コード・コンポーネントが例外を捕捉して対応できるように「例外がスローされます」を選択します。</li> <li>■ アプリケーションが <b>ドキュメント・ベース</b> の場合は、ドキュメントの作成を選択し、Java 例外を含む XML ドキュメントを作成します。</li> </ul> <p>XML ドキュメントを読み込み、エラーを取得するかどうかは、アプリケーションによります。</p>

表 4-3 「詳細」 タブのパラメータ (続き)

ターゲット・パラメータ	説明
待機中コミット	<p>高度な正確さが要求されるアプリケーションの場合は、「待機中コミット」オプションを選択します。</p> <p>アダプタはすべてのレコードがデータベースに物理的に書き込まれるまで待機した後、関数呼出しから戻ります。「待機中コミット」はアダプタのパフォーマンスに影響を与えるため、選択する際には慎重に検討してください。BAPI のコミット動作の詳細は、SAP ドキュメントの『BAPI Programming Guide and Reference (CA-BFA)』を参照してください。</p> <p>データを変更するすべての SAP ビジネス・オブジェクトは、作業をデータベースにコミットする必要があります。R/3 システムのバージョン 3.1 で開発された一部の BAPI は内部コミット動作を使用しており、アダプタでそのコミット動作を変更できません。これらは呼び出されるとすぐに、実行した作業をコミットします。</p> <p>リリース 3.1 以降に開発された BAPI では、外部のコミット・メソッドが使用されます。アダプタがコミット・コマンドを発行すると、そのコミットはデータベース・キューに入れられます。コミットの最初の部分にアプリケーション・エラーがある場合、「ポスティングを実行できませんでした」というエラー・メッセージが返され、アダプタはそのトランザクションをロールバックします。データベースに書き込む際にデータベース・エラーが発生すると、SAP のデータベース・レコードにショート・ダンプが発行されますが、障害に関するメッセージはアダプタに返されません。</p> <p>「待機中コミット」オプションは、「システム」タブで有効にする明示的なコミット・オプションとともに使用する必要があります。「待機中コミット」オプションにより同期コミットが可能になります。このコミット機能は、実際のデータベース・コミットが実行されるか、エラーが生成されるまで待機します。このプロセスには時間がかかりますが、最も正確です。エラーが生成された場合、それは例外メッセージで返されます。</p> <p>このオプションは、デフォルトでは無効になっています。</p>
SAP トレース	<p>SAP トレースを有効にするには、このオプションを選択します。</p> <p>SAP トレースは、デザインタイムおよびランタイムの間、次の場所に格納されています。</p> <p>デザインタイム:</p> <pre>&lt;ADAPTER_HOME&gt;\tools\iwae\bin</pre> <p>ランタイム:</p> <pre>&lt;ORACLE_HOME&gt;\user_projects\domains\base_domain</pre> <p>SAP トレースに使用されるファイル名形式は、デザインタイムとランタイムで次のようになります。</p> <ul style="list-style-type: none"> <li>■ rfc02664_04332.trc</li> <li>■ JCO100226_043846994.trc</li> </ul>
トレース・レベル	<p>SAP トレースの詳細レベルを指定します。リストで 0 から 10 の範囲の値を選択します。</p>

7. 図 4-11 に示すように、「セキュリティ」タブ (オプション) で、この項の情報に基づいて、SAP R/3 ターゲットに適切な情報を入力します。

図 4-11 「セキュリティ」タブ

「セキュリティ」タブでは、接続先の SAP R/3 システムのセキュア・ネットワーク通信 (SNC) 情報を指定できます。

**注意：** このマニュアルで SAP SNC パラメータについて説明する理由は、この機能が SAP Java Connector (SAP JCo) から使用可能であるためです。SNC をアダプタとともに使用する場合、アダプタから直接 SNC を使用するのではなく、SNC が有効になっている状態で SAP ルーターを構成することをお勧めします。これは、最も保護された非常に強い接続方法です。SNC はローカル SAP ホスト・サーバーによって発行される証明書の有効性に依存するため、ローカル SAP サポート・スタッフには、SNC 証明書が有効であること、および SAP ルーターと SAP SNC の構成が正しいことを確認する責任があります。

表 4-4 「セキュリティ」タブのパラメータ

ターゲット・パラメータ	説明
SNC モード	デフォルトでは、SNC は無効になっています。SNC を有効にするには、リストで 1 を選択します。
SNC パートナ	SNC サービスを提供する RFC サーバーまたはメッセージ・サーバー (ロード・バランシング) の名前を入力します。
SNC レベル	リストで SNC ライブラリのバージョンを選択します。
SNC 名	使用している SNC ライブラリの名前を入力します。
SNC ライブラリ・パス	SNC ライブラリへのパスを入力します。

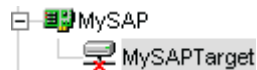
SNC により、SAP R/3 システムの分散コンポーネント間の通信リンクが保護されます。SNC を使用することで、SAP R/3 は GSS-API バージョン 2 標準に準拠している製品をサポートできます。SNC は、アプリケーション・レベル (エンドツーエンド・セキュリティ)、スマートカード認証およびシングル・サインオンをサポートしています。

SAP Enterprise Portal を使用している場合、J2EE エンジンにより自動的に SAP ログオン・チケットが生成されます。考えられる SNC シナリオとしては、SAP Enterprise Portal から Oracle Application Adapter for SAP R/3 への通信があります。

SAP ログオン・チケットを使用して SAP 以外のコンポーネントへのシングル・サインオンを有効にする場合、Pluggable Authentication Services に関する SAP ドキュメントを参照してください。この場合に考えられる SNC シナリオは、SAP Enterprise Portal 以外から Oracle Application Adapter for SAP R/3 への通信です。

- 使用するターゲットについて必要な情報をすべて指定したら、「OK」をクリックします。  
抽出が完了すると、[図 4-12](#) に示すように、「MySAP」アダプタ・ノードの下に新規ターゲット「MySAP Target」が表示されます。

図 4-12 MySAP Target



これで、SAP R/3 ターゲットに接続できるようになりました。

アダプタのスキーマを作成する方法の詳細は、4-14 ページの「XML スキーマの作成」を参照してください。

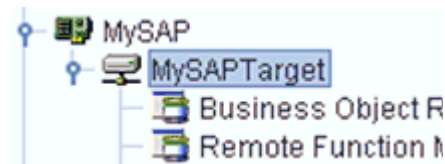
## 4.4.2 定義済 SAP R/3 ターゲットへの接続

既存のターゲットに接続するには、次のようにします。

- 左ペインで「アダプタ」ノードを展開します。
- 「MySAP」ノードを展開します。
- 「MySAP」ノードの下のターゲット名をクリックします (SAPTarget など)。  
「接続」ダイアログに、接続パラメータ用に入力した値が表示されます。
- 接続パラメータを検証します。
- ターゲット名を右クリックし、「接続」を選択します。

[図 4-13](#) に示すように、アイコン上の X が消えてノードが接続されたことが示されます。

図 4-13 接続されている MySAP Target



## 4.4.3 SAP R/3 への接続の管理

ここでは、次の項目について説明します。

- 4.4.3.1 項 「SAP R/3 への接続の切断」
- 4.4.3.2 項 「接続パラメータの変更」
- 4.4.3.3 項 「SAP R/3 への接続の削除」

SAP R/3 接続を管理するために、次の操作を実行できます。

- 現在使用されていない接続を切断します。

様々なトランザクション処理システムへの複数の接続を開いたままにしておくこともできますが、使用していない接続は切断することをお勧めします。



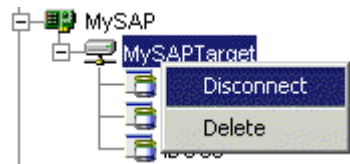
- ターゲットを編集します。  
システム・プロパティを変更する際に接続パラメータを変更できます。切断後に既存のターゲットを変更できます。
- 不要になった接続を削除します。

#### 4.4.3.1 SAP R/3 への接続の切断

ターゲットを切断するには、次のようにします。

1. 「アダプタ」ノードを展開します。
2. 「MySAP」ノードを展開します。
3. 図 4-14 に示すように、接続されているターゲット (SAPTarget など) を右クリックし、「切断」を選択します。

図 4-14 「切断」オプション



SAP R/3 ターゲットから切断すると、SAP R/3 との接続は削除されますが、ノードは残ります。図 4-15 に示すように、X アイコンが表示されてノードが切断されたことが示されます。

図 4-15 切断されたターゲット



#### 4.4.3.2 接続パラメータの変更

アプリケーション・エクスプローラを使用して SAP R/3 のターゲットを作成した後、以前に指定したすべての情報を編集できます。ターゲットを編集した後、Oracle WebLogic Server を再起動して、ランタイムのためにリポジトリを更新する必要があります。

ターゲットを編集するには、次のようにします。

1. 編集するターゲットが切断されていることを確認します。
2. 図 4-16 に示すように、ターゲットを右クリックして、「編集」を選択します。

図 4-16 「編集」オプション



「アプリケーション・サーバー」ダイアログにターゲットの接続情報が表示されます。

3. 必要に応じてダイアログ内のプロパティを変更し、「OK」をクリックします。

#### 4.4.3.3 SAP R/3 への接続の削除

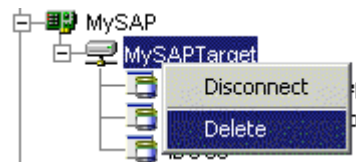
切断は、単に切断したり閉じるだけではなく、削除できます。接続を削除すると、アプリケーション・エクスプローラの左ペインにある SAP R/3 接続のリストにそのノードが表示されなくなります。

接続を削除したら Oracle WebLogic Server を再起動して、ランタイムのためにリポジトリを更新する必要があります。

SAP R/3 への接続を削除するには、次のようにします。

1. 削除するターゲットを特定します。
2.  4-17 に示すように、ターゲット (SAPTarget など) を右クリックし、「削除」を選択します。

図 4-17 「削除」オプション



使用可能な接続のリストにそのノードが表示されなくなります。

## 4.5 アプリケーション・システム・オブジェクトの表示

SAP R/3 に接続すると、既存のビジネス・プロセスのサポートに使用されている SAP R/3 ビジネス・オブジェクトをアプリケーション・エクスプローラで検索および参照できます。

---

**注意：** インストールされているリリースまたはサービス・パックによっては、特定の SAP R/3 システムに RFC\_CUSTOMER\_GET などの特定の RFC が存在しないことがあります。したがって、このマニュアルで説明する例が、使用しているシステムに関連しない場合があります。このような場合、これらの例をアダプタ機能の一般的な参考情報として使用したうえで、SAP R/3 アプリケーション環境内に存在する RFC を選択する必要があります。

SAP リリース・ノート 109533 で説明されているように、SAP 関数モジュール (RFC) は様々なリリース・ステータスで配布されている可能性があります。SAP では、「Released for Customer」ステータスの RFC のみをサポートしています。インタフェースのリリースからの独立性とモジュールの存在および機能の継続性に関する報告はありません。特定の関数モジュールのステータスの詳細は、SAP Service Marketplace を参照してください。

---

## 4.6 XML スキーマの作成

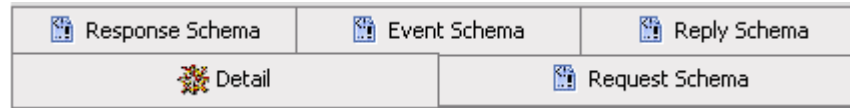
SAP R/3 ビジネス関数ライブラリを検索してオブジェクトを選択した後、アプリケーション・エクスプローラを使用してその関数の XML リクエスト・スキーマと XML レスポンス・スキーマを作成できます。

SAP R/3 ビジネス関数のリクエスト・スキーマおよびレスポンス・スキーマを作成するには、次のようにします。

1. 4-12 ページの「定義済 SAP R/3 ターゲットへの接続」の説明に従って、SAP R/3 ターゲットに接続します。
2. 「ビジネス・オブジェクト・リポジトリ」ノードを展開します。
3. 「Financial Accounting」ノードの左側にあるアイコンをクリックします。

4. 下へスクロールして、「CompanyCode」ビジネス・オブジェクトの左側にあるアイコンをクリックします。
5. 下へスクロールして、「GetDetail」という名前の BAPI を選択します。  
 図 4-18 に示すように、XML スキーマ・タブが右側に表示されます。

図 4-18 XML スキーマ・タブ



6. 各スキーマ・タイプの XML を表示するには、該当するタブをクリックします。

## 4.7 WSDL の生成 (J2CA 構成のみ)

Web Service Definition Language (WSDL) を使用してサービスを記述すると、そのサービスをホスト・サーバー内の他のサービスで使用できるようになります。アプリケーション・エクステンダーを使用して、アダプタのリクエスト/レスポンス (アウトバウンド) とイベント通知 (インバウンド) の両方の J2CA サービスを作成します。

---

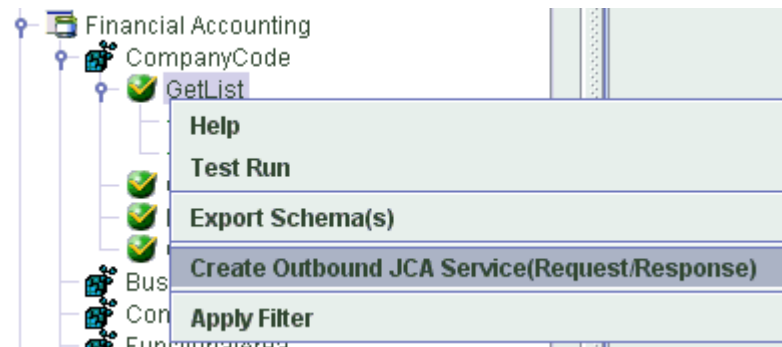
**注意:** インバウンド JCA サービス (イベント) の作成オプションは、選択したノードでイベントがサポートされている場合にのみ使用できます。

---

リクエスト/レスポンス・サービスの WSDL ファイルを生成するには、次のようにします。

1. スキーマを作成した後、各オブジェクトを右クリックします。  
 図 4-19 に示すように、GetList のメニューが表示されます。

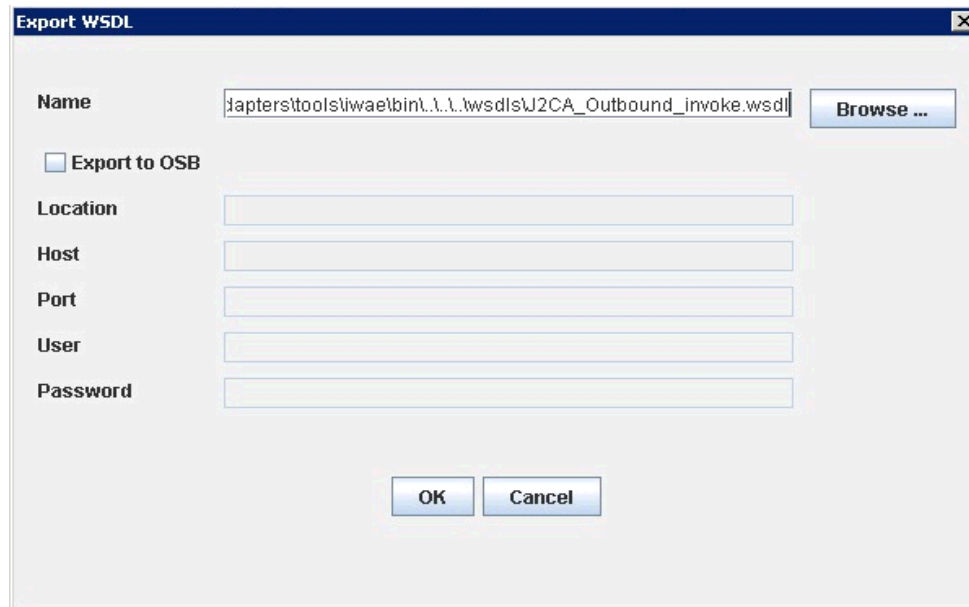
図 4-19 アウトバウンド JCA サービス (リクエスト/レスポンス) の作成



2. アウトバウンド JCA サービス (リクエスト/レスポンス) の作成を選択します。

図 4-20 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

図 4-20 「WSDL のエクスポート」ダイアログ



3. ファイルのデフォルト名を受け入れるか、名前を指定します (J2CA\_Outbound など)。

.wsdl ファイル拡張子が自動的に追加されます。デフォルトでは、リクエスト/レスポンス・サービス用に生成される WSDL ファイルの名前の末尾には `_invoke` が付き、イベント通知用に生成される WSDL ファイル名の末尾には `_receive` が付きます。

4. 「OK」をクリックします。

WSDL ファイルが指定した場所に保存されます。

イベント通知用の WSDL の生成手順は、リクエスト/レスポンスの場合と似ています。イベント通知用の WSDL を生成するには、まず、各イベントのチャンネルを作成する必要があります。詳細は、6-42 ページの「[イベント統合のための WSDL の生成](#)」を参照してください。

## 4.8 Web サービスの作成およびテスト (BSE 構成のみ)

アプリケーション・エクスプローラを使用して、ビジネス関数リポジトリを検索し、アダプタとともに使用する SAP R/3 関数用の Web サービス ([チャンネル](#)とも呼ばれる) を生成できます。次の手順では、`BAPI_MATERIAL_GETLIST` という SAP R/3 BAPI メソッドを例として使用し、SAP R/3 から材料のリストを返します。

---

**注意：** アダプタの J2EE Connector Architecture (J2CA) 実装では、Web サービスは使用できません。アダプタが Oracle Adapter J2CA を使用するようデプロイされている場合、Common Client Interface により、アダプタを使用する統合サービスが提供されます。

---

ここでは、次の項目について説明します。

- [4.8.1 項「Web サービスの作成」](#)
- [4.8.2 項「Web サービスのテスト」](#)
- [4.8.3 項「アイデンティティ伝播」](#)

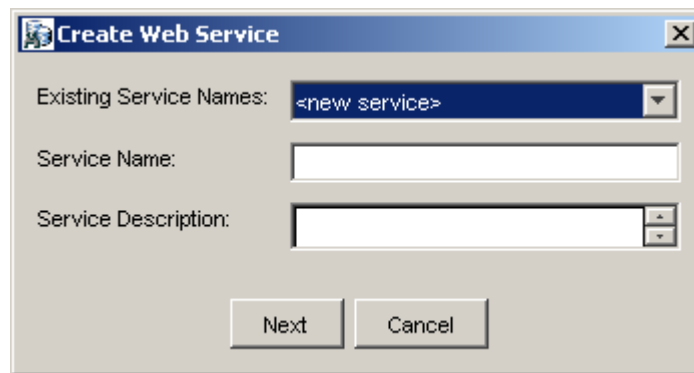
## 4.8.1 Web サービスの作成

SAP R/3 ビジネス関数の Web サービスを作成するには、次のようにします。

1. SAP R/3 ターゲットに接続し、「ビジネス・オブジェクト・リポジトリ」ノードを展開します。
2. 「ビジネス・オブジェクト・リポジトリ」から「BAPI\_MATERIAL\_GETLIST」メソッドを選択します。
3. ビジネス・サービスを作成するノードを右クリックし、「Web サービスの作成」を選択します。

「Web サービスの作成」ダイアログが表示されます。図 4-21 に示すように、新規 Web サービスのメソッドまたは既存の Web サービスのメソッドとしてビジネス関数を追加できます。

図 4-21 「Web サービスの作成」ダイアログ



次のようにします。

- a. 「既存のサービス名」リストで、<新規サービス> または既存のサービスを選択します。
  - b. 新規サービスを作成する場合はサービス名を指定します。この名前により、「ビジネス・サービス」ノードの下のサービス・リストで Web サービスが識別されます。
  - c. サービスに関する簡単な説明を入力します (オプション)。
4. 「次へ」をクリックします。

ライセンスとメソッド・ダイアログが表示されます。

次の情報を指定します。

- a. 「ライセンス名」フィールドで、Web サービスに割り当てるライセンス・コードを 1 つ以上選択します。複数選択するには、**[Ctrl]** キーを押しながらライセンスをクリックします。
  - b. 「メソッド名」フィールドは、デフォルトのメソッド名のままにします。
  - c. 「メソッドの説明」フィールドにメソッドの簡単な説明を入力します (オプション)。
  - d. 「DTD ディレクトリ」フィールドには、使用する DTD の場所を指定します。
5. 「OK」をクリックします。

アプリケーション・エクスプローラの表示が「ビジネス・サービス」ノードに切り替わり、新規 Web サービスが左ペインに表示されます。

6. 新規 Web サービスを右クリックして、メニューの「WSDL の保存」を選択します。「保存」ダイアログが表示されます。

7. WSDL ファイルの名前と、WSDL ファイルを保存するファイル・システム上の場所を指定します。
8. 「保存」をクリックします。

## 4.8.2 Web サービスのテスト

Web サービスを作成した後、テストを実行して正常に機能することを確認できます。Web サービスをテストするために、テスト・ツールが用意されています。

Web サービスをテストするには、次のようにします。

1. 「ビジネス・サービス」ノードをクリックして Web サービスにアクセスします。
2. 「サービス」ノードを展開します。
3. テストするビジネス・サービスの名前を選択します。  
右ペインにビジネス・サービス名がリンクとして表示されます。
4. 右ペインで、指定したビジネス・サービスのリンクをクリックします。  
テスト・オプションが別の BSE サブレット・ページに表示されます。XML 入力が必要な Web サービスをテストする場合は、入力フィールドが表示されます。
5. 適切な入力項目を入力します。
6. 「起動」をクリックします。  
BSE サブレット・ページに結果が表示されます。

## 4.8.3 アイデンティティ伝播

サード・パーティの XML エディタを使用して Web サービスをテストまたは実行する場合、SOAP ヘッダーで指定するユーザー名およびパスワードの値が有効であること、およびこれらが SAP R/3 への接続に使用されることが必要になります。アプリケーション・エクプローラを使用してターゲットの作成時に SAP R/3 用に指定したユーザー名およびパスワードの値は、この Web サービス・リクエストのために上書きされます。次に、Web サービス用の WSDL ファイルに含まれる SOAP ヘッダーの例を示します。

```
<SOAP-ENV:Header>
  <m:ibsinfo xmlns:m="urn:schemas-iwaysoftware-com:iwse">
    <m:service>String</m:service>
    <m:method>String</m:method>
    <m:license>String</m:license>
    <m:disposition>String</m:disposition>
    <m:Username>String</m:Username>
    <m>Password>String</m>Password>
    <m:language>String</m:language>
  </m:ibsinfo>
</SOAP-ENV:Header>
```

<m:disposition> および <m:language> タグは、必須ではないため SOAP ヘッダーから削除できます。

## 4.9 イベント・アダプタの構成

イベントはデータベース内またはアプリケーション・システム内のアクティビティによって生成されます。イベントはアプリケーションでのアクションのトリガーとして使用できます。たとえば、データベースの更新に顧客情報の更新を反映できます。このイベントが発生したときにアプリケーションが処理を実行する必要がある場合、アプリケーションはこのイベントのコンシューマとなります。

---

**注意：** SAP システムでのイベント (アウトバウンド・アクティビティ) を処理するように Oracle Application Adapter for SAP R/3 を構成する前に、SAP サーバーで実行する必要がある重要な構成手順があります。この手順は、アダプタがアプリケーション・エクスプローラを介して SAP システムに接続され、チャンネルの作成が可能になる前に実行する必要があります。詳細は、A-9 ページの A.3 項「SAP アウトバウンド・メッセージ処理」を参照してください。

---

ここでは、次の項目について説明します。

- 4.9.1 項「チャンネルの作成および編集」
- 4.9.2 項「スキーマの検証」

アプリケーション・システムへの接続を作成した後、アプリケーション・エクスプローラを使用してイベントを追加できます。イベントを作成するには、チャンネルを作成する必要があります。

---

**注意：** J2CA 構成を使用している場合、異なるイベント・オブジェクトごとに新規チャンネルを作成し、インバウンド・サービスの作成時にこのチャンネルを選択する必要があります。チャンネルの作成は、J2CA 構成の場合にのみ必要です。さらに、各チャンネルは固有の SAP R/3 プログラム ID と関連付ける必要があります。たとえば、MATMAS と DEBMAS を使用して作業を行う場合、オブジェクトごとに 2 つの個別のチャンネルと、2 つの固有の SAP R/3 プログラム ID が必要です。

---

チャンネルは、バックエンド・システムの特定のインスタンスへの構成済の接続を表します。詳細は、4-19 ページの「チャンネルの作成および編集」を参照してください。

### 4.9.1 チャンネルの作成および編集

次の手順では、イベント用のチャンネルを作成する方法について説明します。すべての定義済イベント・ポートは、チャンネルと関連付けられている必要があります。

ここでは、次の項目について説明します。

- 4.9.1.1 項「チャンネルの作成」
- 4.9.1.2 項「チャンネルの編集」
- 4.9.1.3 項「チャンネルの削除」

チャンネルを作成、変更または削除した場合は、ランタイムのために変更が認識され、リポジットが更新されるように、Oracle WebLogic Server を再起動する必要があります。

---

**注意：** チャンネルの構成と開始は、Oracle Application Adapter for SAP R/3 がインストールされているシステム上でのみ可能です。リモート・ホストのチャンネルの構成と開始はサポートされていません。

---

BSE 構成を使用している場合、イベントは適用されません。J2CA 構成を使用している場合にのみイベントを構成できます。

チャンネルの作成および更新には、アプリケーション・サーバーの再起動が必要です。アプリケーション・サーバーは、チャンネルを作成してインバウンド WSDL を生成した後にも再起動する必要があります。さらに、アプリケーション・サーバーを再起動する前に、必ずアプリケーション・エクスプローラを終了してください。

#### 4.9.1.1 チャンネルの作成

チャンネルを作成するには、次のようにします。

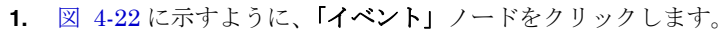
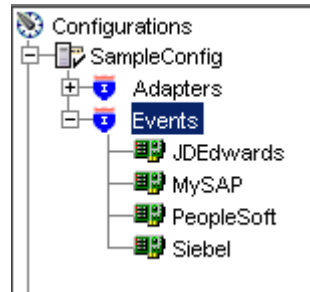
1.  4-22 に示すように、「イベント」ノードをクリックします。

図 4-22 「イベント」ノード



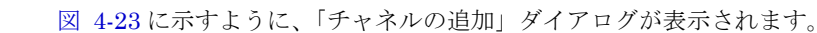
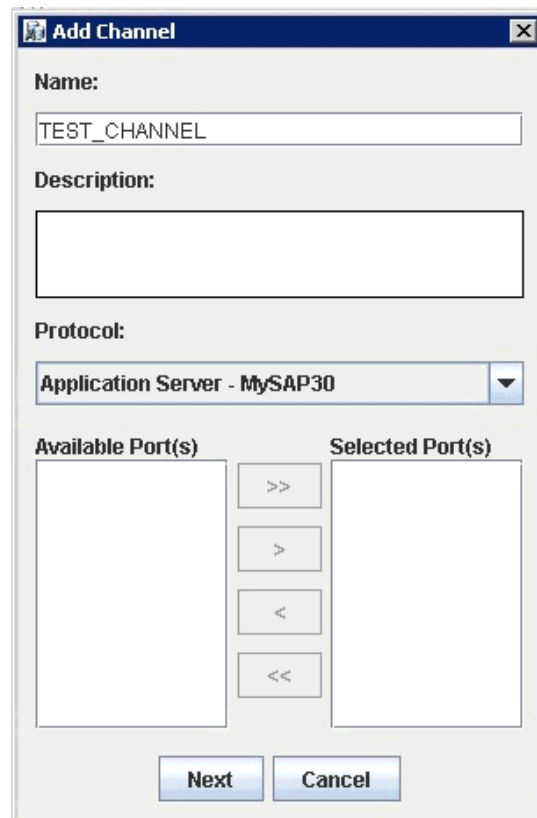
2. 「MySAP」ノードを展開します。  
左ペインにポートおよびチャンネルのノードが表示されます。
3. 「チャンネル」を右クリックして、「チャンネルの追加」を選択します。  
 4-23 に示すように、「チャンネルの追加」ダイアログが表示されます。

図 4-23 「チャンネルの追加」ダイアログ





次のようにします。

- a. チャンネルの名前を入力します (TEST\_CHANNEL など)。
  - b. 簡単な説明を入力します。
  - c. 「プロトコル」リストで、「アプリケーション・サーバー - MySAP30」または「メッセージ・サーバー - MySAP30」を選択します。
4. 「次へ」をクリックします。
- 「メッセージ・サーバー」ダイアログが表示されます。使用可能なタブは次のとおりです。
- ユーザー (必須)
  - システム (必須)
  - セキュリティ
  - 詳細
5. 「ユーザー」タブで、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

**表 4-5 「ユーザー」タブのパラメータ**

ターゲット・パラメータ	説明
クライアント	クライアント通信用に SAP アプリケーションに対して定義されたクライアント番号。
ユーザー	SAP アプリケーションの有効なユーザー ID。
パスワード	SAP アプリケーションの有効なパスワード。
認証モード	SAP R/3 システムに接続する際に使用する認証モード。デフォルトでは、リストの「パスワード」が選択されています。

6. 「システム」タブで、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

**表 4-6 「システム」タブのパラメータ**

ターゲット・パラメータ	説明
ゲートウェイ・ホスト	SAP R/3 ゲートウェイのホスト名。
ゲートウェイ・サービス	SAP R/3 ゲートウェイのサービス。
サーバーのプログラム ID	このチャンネルに使用する SAP R/3 プログラム ID。
アプリケーション・サーバー	アプリケーション・サーバーのホスト名。
システム番号	クライアント通信用に SAP R/3 に対して定義されたシステム番号。
接続プール・サイズ	SAP R/3 が Web サービス・コールに使用できるようにする、プール内のクライアント接続の数。デフォルトでは、接続プール・サイズとして 2 が使用可能です。  <b>重要:</b> デフォルト値が 1 の場合、接続プールは作成されません。かわりに、順次処理による単一の SAP R/3 接続が共有されます。プールされた接続により、パラレル処理による SAP R/3 への複数の接続が呼び出されます。  アプリケーション・エクスペローラを使用して Web サービスを作成する場合、ランタイム中に、この接続プール・サイズ値が Web サービスによって使用されます。結果として、目的に対して十分な接続プール・サイズが確保されます。

表 4-6 (続き) 「システム」タブのパラメータ

ターゲット・パラメータ	説明
接続タイムアウト (分)	接続プールのタイムアウト値 (分単位)。デフォルト値は 10 分です。
接続待機時間 (秒)	使用する接続プールの待機時間 (秒単位)。デフォルト値は 30 秒です。  ERP サーバーへの接続は、クライアントとリモート・サーバーの両方で貴重なリソースを消費します。接続のプールを作成し、リソースと時間の制約を最小限に抑えることができます。プールのサイズを見積もる際、消費するサーバー・リソースの量、受信するドキュメントの数とサイズおよび Java 仮想マシンのサイズによって、プール・サイズを計算できます。SAP ドキュメント『Memory Management (BC-CST-MM)』の項で、SAP R/3 システムで必要となるリソースについて詳細に説明されています。

7. 「セキュリティ」タブ (オプション) で、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

表 4-7 「セキュリティ」タブのパラメータ

ターゲット・パラメータ	説明
SNC モード	デフォルトでは、SNC は無効になっています。SNC を有効にするには、リストで 1 を選択します。
SNC パートナ	SNC サービスを提供する RFC サーバーまたはメッセージ・サーバー (ロード・バランシング) の名前を入力します。
SNC レベル	リストで SNC ライブラリのバージョンを選択します。
SNC 名	使用している SNC ライブラリの名前を入力します。
SNC ライブラリ・パス	SNC ライブラリへのパスを入力します。

8. 「詳細」タブ (オプション) で、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

表 4-8 「詳細」タブのパラメータ

ターゲット・パラメータ	説明
IDOC 確認	このオプションが選択されている場合、SAP に STATUS IDOC が返されて、受信された各 IDOC について受信が確認されます。  標準の IDOC が送信される場合、SAP IDOC モニターを通じて WE02 トランザクション IDOC リストに表示される通常の最新ステータスは、 <code>/Data Sent to Port OK/ - /03/</code> です。WE02 トランザクションを使用して IDOC がターゲット宛先で受信されたことを確認する場合は、このオプションを有効にします。IDOC が受信されると、アダプタは「STATUS IDOC」を SAP に返し、WE02 トランザクション IDOC リスト内の IDOC のステータスが <code>/12/</code> 、 <code>/Dispatch OK/</code> に変わります。  多数の IDOC を一度に送信する場合にこのオプションが有効になっていると、パフォーマンスに影響を及ぼす可能性があることに注意してください。パフォーマンスが問題となる場合は、このオプションを無効にすることをお勧めします。
IDOC フォーマット	リストで IDOC タイプを選択します。 <ul style="list-style-type: none"><li>■ XML (デフォルト)</li><li>■ XML-CDATA-ENVELOPED</li></ul>
IDOC リリース	接続に使用する IDOC バージョニング。

表 4-8 「詳細」 タブのパラメータ ( 続き )

ターゲット・パラメータ	説明
IDOC リリース・プロバイダ	<p>接続のための IDOC リリース・プロバイダ。リストで <b>IDOC DOCREL</b> フィールド ( デフォルト )、<b>SAP</b> リリース、<b>ユーザー入力</b> または <b>自動チェック</b> を選択します。</p> <p>自動チェック・オプションの詳細は、この手順の後に記載されている説明を参照してください。</p>
SAP トレース	<p>SAP トレースを有効にするには、このオプションを選択します。</p> <p>SAP トレースは、デザインタイムおよびランタイムの間、次の場所に格納されています。</p> <p>デザインタイム :</p> <pre>&lt;ADAPTER_HOME&gt;\tools\iwaie\bin</pre> <p>ランタイム :</p> <pre>&lt;ORACLE_HOME&gt;\user_projects\domains\base_domain</pre> <p>SAP トレースに使用されるファイル名形式は、デザインタイムとランタイムで次のようになります。</p> <ul style="list-style-type: none"> <li>■ rfc02664_04332.trc</li> <li>■ JCO100226_043846994.trc</li> </ul>
トレース・レベル	SAP トレースの詳細レベルを指定します。リストで 0 から 10 の範囲の値を選択します。
スレッド件数	オープン・スレッドの数。デフォルトでは、3 スレッドが指定されています。
処理モード	<p>リストで同期処理のタイプを選択します。選択可能な値には、「REQUEST」、「REQUEST_RESPONSE」などがあります。</p> <p>処理モードを「REQUEST_RESPONSE」に設定してチャンネルを作成すると、このチャンネルは、オブジェクトに返信スキーマが含まれる場合にのみ、WSDL 作成プロセス時に表示されます。それ以外の場合にはこのチャンネルは表示されません。</p>

## 9. 「OK」 をクリックします。

図 4-24 に示すように、左ペインの「チャンネル」ノードの下にチャンネルが表示されます。

図 4-24 MySAP\_Channel



アイコンの上の X は、そのチャンネルが現在切断されていることを示します。イベント構成をアクティブ化するには、チャンネルを開始する必要があります。

**注意：** Oracle Application Adapter for SAP R/3 を BPM、BPEL または OSB のインバウンド・プロセス・コンポーネントと統合することを計画している場合、チャンネルを開始しないでください。チャンネルは、BPM、BPEL または OSB のプロセス・コンポーネントがデプロイされた後に、ランタイム・サーバーにより管理されます。テストやデバッグ目的でアプリケーション・エクスペローラからチャンネルを開始した場合は、ランタイム前に停止してください (BPM、BPEL または OSB のプロセス・コンポーネントで作業する場合)。

10. 「チャンネル」ノードを右クリックして、「開始」を選択します。

図 4-25 に示すように、作成したチャンネルがアクティブになります。

図 4-25 アクティブな MySAP\_Channel



アイコン上の X が消えます。

11. チャンネルを停止するには、接続されているチャンネル・ノードを右クリックして「停止」を選択します。

チャンネルは非アクティブになり、アイコン上の X が消えます。

#### IDOC リリース・プロバイダ・パラメータ:自動チェック・オプション

非標準の IDOC リリースの場合、アダプタの以前のバージョンでは、**ユーザー入力**オプションを選択して、手動で IDOC リリースを指定する必要がありました。ただし、このオプションはターゲットによって処理されるすべての IDOC に適用されますが、このことが制限になる可能性があります。**自動チェック**・オプションにより、アダプタは WE02 トランザクションを介して使用できる論理システムから SAP パートナ・プロファイルを読み込むことができます。

API によって、プログラム ID の論理システムとメッセージの受信 IDOC タイプが照合され、これにより、EDI パートナ API を介してパートナ・プロファイルから IDoc のバージョンが取得されます。

**自動チェック**・オプションの場合、すべての IDOC メタデータ構成を IDOC が送信されている SAP システムに保持するため、ユーザーは IDOC タイプが構成される SAP のバージョンを推測する必要はありません。

次のオプションも IDOC リリース・プロバイダ・パラメータに対して設定できます。

- **IDOC DOCREL フィールド** - IDOC の DOCREL フィールドを使用して IDOC のメタデータ・リリースを取得します。
- **SAP リリース** - ログオンの SAP リリースを使用して IDoc のメタデータ・リリースを取得します。
- **ユーザー入力** - 送信されている IDOC の SAP リリースを入力する必要があります (40A、40B、45B、700 など)。

IDOC 用語について次のことに注意してください。

- **メッセージ・タイプ** - ビジネス・メッセージを表します。メッセージ・タイプは具体的な指示ではなく、受注処理オーダーなどのビジネス・プロセスを一般的な用語で表しています。メッセージ・タイプは、通常、リリースやバージョンには依存していません。
- **セグメント** - IDOC データはフィールドのグループまたはセグメントで構成されています。各セグメントはリリースされるか、特定の SAP リリースにバインドされます。特定の SAP リリースに対してアクティブなセグメント・バージョンは一度に 1 つのみの場合もあります。

基本タイプまたは IDOC タイプはセグメントの階層から構成される物理メッセージを表しており、これは特定の順序で構成されています。各セグメントは特定の SAP バージョンに属しています。各 IDOC タイプはセグメントのタイプの順序です。各 IDOC タイプはリリースされるか、SAP の特定のバージョンにバインドされます。SAP の以前のバージョンでパートナにメッセージを送信する必要がある場合、以前の IDOC バージョンと以前の SAP IDOC リリースを使用してメッセージを送信する必要があります。IDOC 内のフィールドの順序は異なります。しかし、各セグメントの終わりにあるフィールドの数は、各セグメント・バージョンの増加に伴って追加され、セグメント数も各 IDOC のバージョンの増加に伴って増加します。

### 4.9.1.2 チャネルの編集

チャネルを編集するには、次のようにします。

1. 左ペインで、編集するチャネルを特定します。
2. チャネルを右クリックし、「**編集**」を選択します。  
「チャネルの編集」ペインが表示されます。
3. チャネル構成に必要な変更を加え、「**終了**」をクリックします。  
チャネルを編集する場合、ランタイムのために変更が認識され、リポジトリが更新されるように、Oracle WebLogic Server を再起動する必要があります。

### 4.9.1.3 チャネルの削除

チャネルを削除するには、次のようにします。

1. 左ペインで、削除するチャネルを特定します。
2. チャネルを右クリックし、「**削除**」を選択します。  
左ペインのリストにチャネルが表示されなくなります。  
チャネルを削除する場合、ランタイムのために変更が認識され、リポジトリが更新されるように、Oracle WebLogic Server を再起動する必要があります。

## 4.9.2 スキーマの検証

インバウンド処理 ( イベント ) のルート検証、ネームスペース検証およびスキーマ検証は、BPEL を使用する Oracle Application Adapter for SAP R/3 でサポートされています。

Oracle Application Adapter for SAP R/3 を使用してインバウンド処理を検証するには、次のようにします。この手順では、MATMAS (Material Master) をインバウンド処理の例として使用します。

1. アプリケーション・エクスプローラを起動します。
2. MySAP ターゲットに接続します。
3. 「IDOCs」ノードを展開します。
4. MySAP アダプタ用にチャネルが作成されていることを確認します。
5. 「MATMAS Material Master」ノードを選択して展開します。
6. 「MATMAS01」を右クリックして、**インバウンド JCA サービス ( イベント ) の作成**を選択します。

図 4-26 に示すように、「WSDL のエクスポート」ダイアログが開き、「ルート」、「ネームスペース」および「スキーマ」の各検証の 3 つのチェック・ボックスが表示されます。

図 4-26 「WSDL のエクスポート」 ダイアログ

- 複数の検証オプションを選択できます。
  - ルート検証は、インバウンド XML ドキュメント内のルート要素の検証に使用します。
  - ネームスペース検証は、インバウンド XML ドキュメント内のネームスペースの検証に使用します。
  - スキーマ検証は、WSDL ドキュメント内のスキーマを使用したインバウンド XML ドキュメントの検証に使用します。
  - ランタイム中、検証は選択された検証オプションに基づいて処理されます。
  - 複数の検証オプションが選択されている場合、ランタイム中に最初の検証オプションが失敗すると、残りの検証オプションは処理されません。
  - ルートとネームスペースの検証は、適度な検証レベルであるとみなされます。スキーマの検証は、より厳密な検証レベルになります。
  - ルート要素とネームスペースが SAP 環境内の IDOC 間で異なっている場合を除いて、ルートとネームスペースの検証オプションを使用することをお勧めします。
7. WSDL ドキュメントを生成して、BPEL プロセスを作成します。
  8. SAP GUI からトランザクション (IDOC) をトリガーします。

検証に失敗したインバウンド・トランザクションは、SAP Transaction Monitor (SM58) に表示されます。

ステータス・テキスト・フィールドには、検証プロセスに失敗したドキュメントの「java.lang.exception」が表示されます。

## 4.10 SAProuter サービスの構成

この項では、SAProuter サービスを構成する方法について説明します。ここでは、次の項目について説明します。

- 4.10.1 項「SAProuter の構成」
- 4.10.2 項「Oracle コンポーネントを使用した SAProuter のテスト」
- 4.10.3 項「SAProuter 構成の検証」

SAProuter は、異なるタイプの SAP 接続 ( 内部接続と外部接続 ) 間のプロキシ・サーバーまたは中間エンドポイントとみなすことができます。ファイアウォールまたはポート・フィルタとしても機能でき、望ましくないトラフィックが内部の SAP ネットワークに侵入しないように保護したり、特定のシステム上の特定のポート間のネットワーク・トラフィックを制限できます。

機能や構成に関連する SAProuter の詳細は、次の Web サイトの SAProuter ドキュメントを参照してください。

<http://help.sap.com>

この Web サイトにアクセスしたら、SAProuter を検索できます。

SAProuter の構成および設定の手順は、プラットフォームおよびインストールされている SAP のバージョンにより異なる場合があります。

### 4.10.1 SAProuter の構成

SAProuter を構成するには、次のようにします。

1. SAProuter のルート表を編集または作成します。

次に、ルート文字列構文の例を示します。

```
/H/ hostname
/S/ service
/W/ password
```

---

**注意：** スラッシュ間の文字では大文字と小文字が区別されます。

---

2. 次の形式を使用して、権限表でルート・エントリを構成します。

```
{P/S/D} <source host> [<dest host>] <dest serv> <password>
```

各項目の説明は次のとおりです。

- コマンドの前に *P* を指定すると、SAProuter により接続が構成 ( 許可 ) されます。P ( 許可 ) のエントリでは、<password> 要素にパスワードを含めることができます。
- コマンドの前に *S* を指定すると、SAP プロトコルを使用した接続のみ許可されます。その他のプロトコル (TCP など) を使用した接続は許可されません。
- コマンドの前に *D* を指定すると、接続の構成を防止 ( 拒否 ) します。
- ソース・ホスト <source host> には、(SAProuter から見て) 接続の発信元のホストを指定します。これは、ホスト名、IP アドレスまたは IP サブネットワークにすることができます。
- 宛先ホスト <dest host> には、(SAProuter から見て) 接続のアクセス先のホストを指定します。これは、ホスト名、IP アドレスまたは IP サブネットワークにすることができます。
- 宛先ポート <dest serv> には、接続のアクセス先である宛先ホストのポート ( サービス ) を指定します。必要に応じて、2 つのポートをピリオド (.) 文字を使用して区切り、ポート範囲を指定することもできます。<dest serv> の値が 3200.3298 の場合、

宛先サーバーへの接続には、3200 から 3298 のすべてのポートが使用されます。次に例を示します。

```
p 192.168.128.166 SAP-Server1 3300
```

### 3. SAProuter を起動します。

これで、Oracle コンポーネントを使用して SAProuter をテストする準備ができました。

## 4.10.2 Oracle コンポーネントを使用した SAProuter のテスト

SAProuter が問題なく起動したら、次の項目を作成できます。

- アプリケーション・サーバーのルート文字列を使用した、アプリケーション・エクスプローラでのターゲット。
- ゲートウェイ・ホストおよびアプリケーション・サーバーのルート文字列を使用したチャンネル。

ゲートウェイ・ホストおよびアプリケーション・サーバーに、問題なく接続を作成できることを確認してください。

ターゲット作成の詳細は、4.4 項「SAP R/3 の接続 (ターゲット) の確立」を参照してください。チャンネル作成の詳細は、4.9.1 項「チャンネルの作成および編集」を参照してください。

次に、ルート文字列構文の例を示します。

```
/H/host/H/host/S/service
```

各項目の説明は次のとおりです。

- *host* は、ホスト名を表します。
- *service* は、サービス (ポート) を指定します。

次に例を示します。

```
/H/192.168.128.166/H/isdsrv2.ibi.com/S/3300
```

図 4-27 に、アプリケーション・エクスプローラでターゲット用に構成されたルート文字列を示します。

図 4-27 構成されたターゲット・ルート文字列

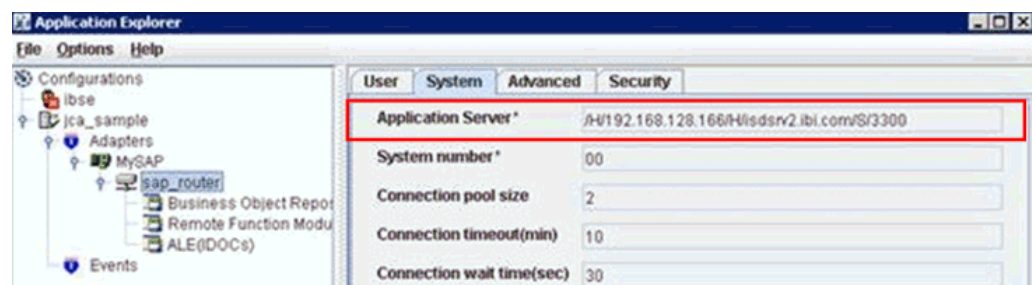


図 4-28 に示すように、メッセージ・サーバーにアクセスするには、サービス名とメッセージ・サーバーのポート (アプリケーション・ホストではありません) を指定して、ルーターに提供する必要があります。



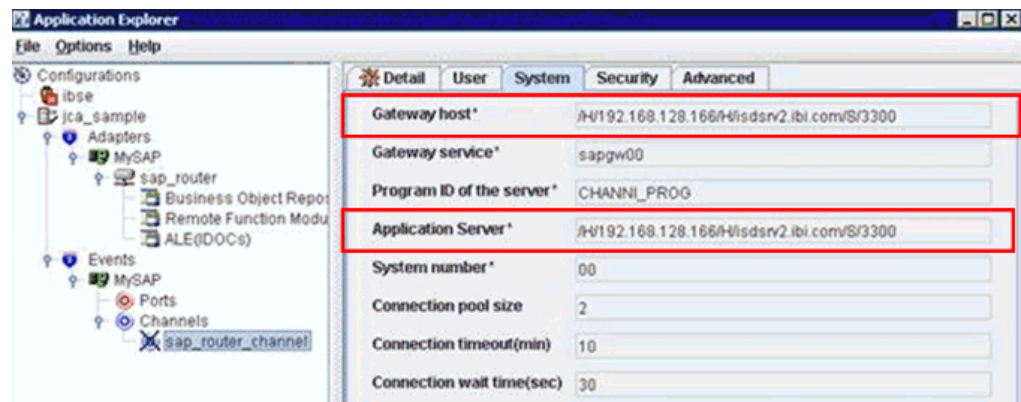
図 4-28 メッセージ・サーバーの接続パラメータ

Name	Value
Message Server	
R/3 name	
Server group	
Gateway host	
Gateway service	

また、接続のバランシングが行われるようにログオン・グループを指定する必要があります。

図 4-29 に、アプリケーション・エクスプローラでチャンネル用に構成されたルート文字列を示します。

図 4-29 構成されたチャンネル・ルート文字列



接続されたら、アウトバウンドおよびインバウンドの WSDL ファイルを作成し、BPEL、メディアータ、BPM および OSB プロセスが正常に機能しているかどうかを確認します。詳細は、このドキュメントの対応する項を参照してください。

### 4.10.3 SAProuter 構成の検証

ベスト・プラクティスとして、エラーを調べるために最初にトレースを有効にするか、コマンド・プロンプト・ウィンドウに出力が表示されるようにして SAProuter を起動し、サンプル・コマンドを送信します。次のコマンドを使用して、必要に応じてロギングおよびロギング・レベルを指定します。

```
saprouter -r -T logfile.name -V3
```

各項目の説明は次のとおりです。

- `-T` は、トレース・オプションをアクティブ化します。
  - `-V(n)` は、ロギング・レベルを指定します。
  - `logfile.name` は、ログ・ファイルの名前 (`logfile.trc` など) です。
1. コマンド・プロンプト・ウィンドウで SAProuter を起動したら、コマンド・プロンプトまたはトレース・ファイル (SAProuter が起動されるフォルダ内に生成されます) 内でエラーがあるかどうか調べます。

図 4-30 に、コマンド・プロンプト・ウィンドウでトレースをアクティブ化するコマンドを指定して起動された SAProuter を示します。

図 4-30 SAProuter コマンド・プロンプト

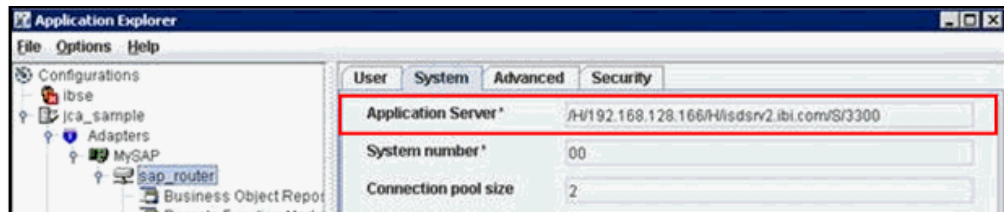
```
C:\sap_router>saprouter -r -T saproutertrace.trc -U3 -G saprouterlog.trc
trcfile saproutertrace.trc
logfile saprouterlog.trc
=
```

- SAProuter が起動したら、アプリケーション・サーバーのルート文字列を使用してアプリケーション・エクスプローラでターゲットを作成します。さらに、ゲートウェイ・ホストおよびアプリケーション・サーバーのルート文字列を使用してチャンネルを作成します。

ゲートウェイ・ホストおよびアプリケーション・サーバーに、問題なく接続を作成できることを確認してください。

図 4-31 に、アプリケーション・エクスプローラでターゲット用に構成されたルート文字列を示します。

図 4-31 構成されたターゲット・ルート文字列



- 接続が適切に確立されたことを検証するには、図 4-32 に示すように、定義済ログ・ファイル (saprouterlog.trc など) を確認します。

図 4-32 SAProuter のログ・ファイル

```
saprouterlog.trc
1 Mon Apr 16 19:59:42 2012 INIT LOGFILE
2 Mon Apr 16 19:59:42 2012 READ ROUTTAB ./saproutab o.k.
3 Mon Apr 16 20:00:49 2012 CONNECT FROM C9/- host 192.168.128.167/2214 (AXTST167.AMTEXPDC)
4 Mon Apr 16 20:00:49 2012 CONNECT TO S9/17 host 172.19.20.59/3300 (isdsv2.ibi.com)
5 Mon Apr 16 20:00:50 2012 ESTABLISHED S9/17
6 Mon Apr 16 20:00:51 2012 CONNECT FROM C25/- host 192.168.128.167/2216 (AXTST167.AMTEXPDC)
7 Mon Apr 16 20:00:51 2012 CONNECT TO S25/33 host 172.19.20.59/3300 (isdsv2.ibi.com)
8 Mon Apr 16 20:00:51 2012 ESTABLISHED S25/33
9 Mon Apr 16 20:12:49 2012 DISCONNECT C9/17 host 192.168.128.167/2214 (AXTST167.AMTEXPDC)
10 Mon Apr 16 20:12:49 2012 DISCONNECT C25/33 host 192.168.128.167/2216 (AXTST167.AMTEXPDC)
11
```

- SAProuter 構成の検証が正常に完了したら、SAProuter を停止し、トレース・オプションを無効にして SAProuter を再起動できます。

---

**注意：** SAProuter の検証が正常に完了した後に、Oracle Application Adapter for SAP R/3 のターゲットが SAProuter と連携できるようになります。アダプタ・ターゲットに定義された SAProuter の構成情報は、アダプタ・リポジトリに格納されます。その結果、Oracle Application Adapter for SAP R/3 は、BPEL、メディエータ、BPM および OSB などの SOA コンポーネントから起動された場合、アダプタ・ターゲットに定義されている SAProuter 設定をランタイム中に利用できます。

---

---

# Oracle WebLogic Server のデプロイメント および統合

この章では、Oracle WebLogic Server (OracleWLS) のデプロイメントおよび Oracle Application Adapter for SAP R/3 との統合について説明します。次の項で構成されています。

- 5.1 項「アダプタと Oracle WebLogic Server との統合」
- 5.2 項「アダプタのデプロイメント」
- 5.3 項「アダプタ構成の更新」

**関連項目：**

- *Oracle WebLogic Server Adapter* コンセプト

## 5.1 アダプタと Oracle WebLogic Server との統合

Oracle Application Adapter for SAP R/3 は、インストール時に OracleWLS コンテナ内にデプロイされます。すべてのクライアント・アプリケーションは、OracleWLS 環境内で実行されます。J2CA デプロイメントでは、Common Client Interface (CCI) により、OracleWLS クライアント・アプリケーションがリソース・アダプタと統合されます。

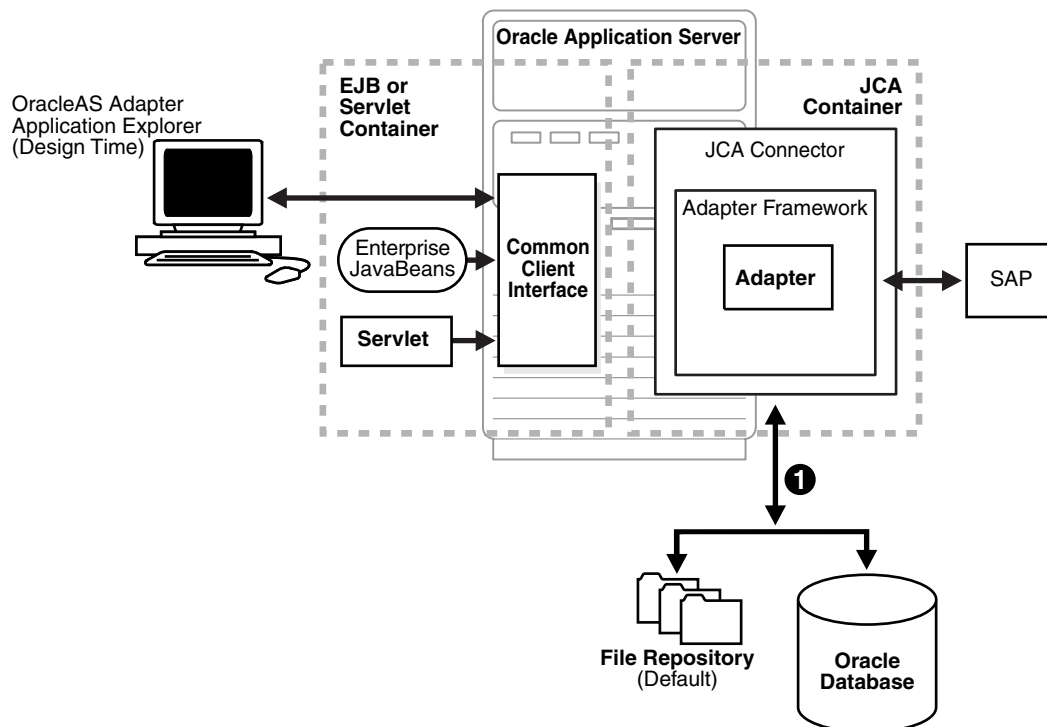
**関連項目：**

- *Oracle WebLogic Server Adapter* コンセプト

## 5.2 アダプタのデプロイメント

図 5-1 に、Oracle Application Server への J2CA コネクタのデプロイメントを示します。ランタイム・サービス・シナリオでは、Enterprise Java Bean、サーブレットまたは Java プログラム・クライアントによって、J2CA リソース・アダプタへの CCI コールが行われます。アダプタは、これらのコールをリクエストとして処理し、EIS に送信します。その後、EIS レスポンスがクライアントに返信されます。

図 5-1 Oracle Application Server J2CA アーキテクチャ



① Use either the default file repository or an Oracle database as your repository.

#### 関連項目：

- [Oracle WebLogic Server Adapter コンセプト](#)

## 5.3 アダプタ構成の更新

ここでは、次の項目について説明します。

- [5.3.1 項「管理対象コネクタ・ファクトリ・オブジェクトの作成」](#)
- [5.3.2 項「複数の管理対象コネクタ・ファクトリ・オブジェクトの作成」](#)
- [5.3.3 項「追加の接続ファクトリ値のための WSDL ファイルの変更」](#)

Oracle Application Adapter for SAP R/3 の J2CA デプロイメント時に、OracleWLS により次の場所に `ra.xml` と呼ばれるデプロイメント・ディスクリプタが生成されます。

```
<ADAPTER_HOME>\iwafjca.rar\META-INF
```

OracleWLS デプロイメント・ディスクリプタは、前述のディレクトリ内にあります。

---

**注意：** 複数の管理対象接続ファクトリは、アウトバウンド処理 (サービス) の場合にのみサポートされます。

---

### 5.3.1 管理対象コネクタ・ファクトリ・オブジェクトの作成

ra.xml ディスクリプタにより、リソース・アダプタに関する OracleWLS 固有のデプロイメント情報が提供されます。たとえば、アプリケーション・エクスプローラ内の jca\_sample 構成は、ra.xml ファイルでは次のように表されます。

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE connector PUBLIC "-//Sun Microsystems, Inc.//DTD Connector 1.0//EN"
'http://java.sun.com/dtd/connector_1_0.dtd'>
<connector>
  <display-name>IWAFJCA10</display-name>
  <vendor-name>IWAY Software</vendor-name>
  <spec-version>1.0</spec-version>
  <eis-type>IWAF</eis-type>
  <version>1.0</version>
  <license>
    <license-required>>false</license-required>
  </license>
  <resourceadapter>

<managedconnectionfactory-class>com.ibi.afjca.spi.IWAFManagedConnectionFactory</managed
connectionfactory-class>

<connectionfactory-interface>javax.resource.cci.ConnectionFactory</connectionfactory-in
terface>

<connectionfactory-impl-class>com.ibi.afjca.cci.IWAFConnectionFactory</connectionfactor
y-impl-class>
  <connection-interface>javax.resource.cci.Connection</connection-interface>
  <connection-impl-class>com.ibi.afjca.cci.IWAFConnection</connection-impl-class>
  <transaction-support>NoTransaction</transaction-support>
  <config-property>
    <config-property-name>AdapterName</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>Config</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayHome</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value>C:\oracle\Middleware\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters</config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayConfig</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value>jca_sample</config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoDriver</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoURL</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoUser</config-property-name>
```

```

    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoPassword</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>LogLevel</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value>DEBUG</config-property-value>
  </config-property>
  <authentication-mechanism>
    <authentication-mechanism-type>BasicPassword</authentication-mechanism-type>

  <credential-interface>javax.resource.spi.security.PasswordCredential</credential-interface>
  </authentication-mechanism>
  <reauthentication-support>true</reauthentication-support>
</resourceadapter>
</connector>

```

次の表に、ra.xml ファイルで定義されているパラメータを示します。

パラメータ名	説明
IWayHome	OracleWLS のパッケージ化されたアプリケーション・アダプタのベース・インストール・ディレクトリ。
IWayConfig	アプリケーション・エクスプローラで定義されているアダプタ構成名。たとえば、Oracle Application Adapter for SAP R/3 には、アプリケーション・エクスプローラ内に事前定義された jca_sample 構成があります。
IWayRepoURL	データベースへの接続を開くときに使用される URL。これは、Oracle Database をリポジトリとして使用する場合にのみ必要となります。
IWayRepoUser	データベースに接続するときに使用されるユーザー名。これは、Oracle Database をリポジトリとして使用する場合にのみ必要となります。
IWayRepoPassword	パスワード。指定した場合、構成が上書きされます。これは、Oracle Database をリポジトリとして使用する場合にのみ必要となります。
loglevel	ManagedConnectorFactory プロパティによって設定されるレベルを上書きします。

## 5.3.2 複数の管理対象コネクタ・ファクトリ・オブジェクトの作成

複数の管理対象コネクタ・ファクトリ・オブジェクトを作成するには、weblogic-ra.xml ファイルを編集して、<connection-instance> ノードをさらに追加する必要があります。このファイルは次の場所にあります。

```
<ADAPTER_HOME>\iwafjca.rar\META-INF
```

たとえば、アプリケーション・エクスプローラ内の最初の jca\_configuration は、weblogic-ra.xml ファイルでは次のように表されます。

```
<?xml version="1.0"?>
<weblogic-connector xmlns="http://www.bea.com/ns/weblogic/90">
  <enable-access-outside-app>true</enable-access-outside-app>
  <enable-global-access-to-classes>true</enable-global-access-to-classes>
  <outbound-resource-adapter>
    <default-connection-properties>
      <pool-params>
        <initial-capacity>0</initial-capacity>
      </pool-params>
      <transaction-support>LocalTransaction</transaction-support>
    </default-connection-properties>
    <connection-definition-group>

<connection-factory-interface>javax.resource.cci.ConnectionFactory</connection-factory-interface>
      <connection-instance>
        <jndi-name>eis/OracleJCAAdapter/DefaultConnection</jndi-name>
      </connection-instance>
    </connection-definition-group>
  </outbound-resource-adapter>
</weblogic-connector>
```

複数の管理対象コネクタ・ファクトリ・オブジェクトを作成するには、新規の<connection-instance> ノードをファイルに追加する必要があります。次に例を示します。

```
<?xml version="1.0"?>
<weblogic-connector xmlns="http://www.bea.com/ns/weblogic/90">

  <enable-access-outside-app>true</enable-access-outside-app>
  <enable-global-access-to-classes>true</enable-global-access-to-classes>

  <outbound-resource-adapter>
    <default-connection-properties>
      <pool-params>
        <initial-capacity>0</initial-capacity>
      </pool-params>
      <transaction-support>LocalTransaction</transaction-support>
    </default-connection-properties>
    <connection-definition-group>

<connection-factory-interface>javax.resource.cci.ConnectionFactory</connection-factory-interface>
      <connection-instance>
        <jndi-name>eis/OracleJCAAdapter/DefaultConnection</jndi-name>
      </connection-instance>
      <connection-instance>
        <jndi-name>eis/OracleJCAAdapter/DefaultConnection1</jndi-name>
        <connection-properties>
          <properties>
            <property>
<name>IWayHome</name>
          <value>C:\oracle\Middleware\Oracle_SOAI\soa\thirdparty\ApplicationAdapters</value>
            </property>
```

```

        <property>
        <name>IWayConfig</name>
        <value>jca_sample2</value>
        </property>
        <property>
<name>IWayRepoURL</name>
<value></value>
        </property>
        <property>
        <name>IWayRepoUser</name>
        <value></value>
        </property>
        <property>
        <name>IWayRepoPassword</name>
        <value></value>
        </property>
        <property>
        <name>LogLevel</name>
        <value>Debug</value>
        </property>
        </properties>
    </connection-properties>
    </connection-instance>
</connection-definition-group>
</outbound-resource-adapter>
</weblogic-connector>

```

<connection-instance> セクションで <property> 要素を指定しない場合、ra.xml ファイルから値が取得されます。ra.xml ファイルでデフォルトのプロパティを指定し、weblogic-ra.xml ファイルで必要に応じてそれらをオーバーライドすることもできます。また、J2CA 構成 (jca\_sample2 など) は、アプリケーション・エクンプローラで作成する必要があります。

---

**注意:** ra.xml および weblogic-ra.xml ファイルを変更する場合、アプリケーション・サーバーを再起動する必要があります。アプリケーション・サーバーが実行中の場合は、アプリケーション・サーバーを停止してから再起動します。

また、これらの変更をアクティブ化するには、iwafjca.rar ファイルを Oracle WebLogic 管理コンソールに再デプロイする必要があります。

---

### 5.3.3 追加の接続ファクトリ値のための WSDL ファイルの変更

アプリケーション・エクンプローラでは、デフォルトの接続ファクトリ名 eis/OracleJCAAdapter/DefaultConnection を使用して J2CA プロパティ・ファイルが生成されます。ユーザーが追加の接続ファクトリを作成した場合、追加の構成および接続ファクトリ用に生成された WSDL は、J2CA プロパティ・ファイル内の jca:address セクションの location フィールドを反映するように変更する必要があります。次の例は、isdsrv2\_conn2 構成が含まれている Oracle Application Adapter for SAP R/3 のデフォルトの J2CA プロパティ・ファイルを示しています。

J2CA プロパティ・ファイルに、デフォルトの接続ファクトリ eis/OracleJCAAdapter/DefaultConnection が含まれていることに注意してください。

```

<jca:address location="eis/OracleJCAAdapter/DefaultConnection"
    ConnectionSpec="com.ibi.afjca.cci.IWAFConnectionSpec"
    cs.AdapterName="MySAP" cs.Config="isdsrv2_conn2"
    UIConnectionName="Connection1"/>

```

この接続ファクトリ値を、eis/OracleJCAAdapter/DefaultConnection1 に変更する必要があります。



次に例を示します。

```
<jca:address location="eis/OracleJCAAdapter/DefaultConnection1"  
             ConnectionSpec="com.ibi.afjca.cci.IWAFConnectionSpec"  
             cs.AdapterName="MySAP" cs.Config="isdsrv2_conn2"  
             UIConnectionName="Connection1"/>
```

`jca:address` セクション内の `location` フィールドの値のみを変更する必要があることに注意してください。その他のフィールドやセクションは変更しないでください。



---

---

## Oracle SOA Suite での BPEL サービス・コンポーネントとの統合

Oracle Application Adapter for SAP R/3 は Business Process Execution Language (BPEL) Process Manager とシームレスに統合され、これにより Web サービス統合が簡略化されます。Oracle BPEL Process Manager は、サービス指向アーキテクチャ (SOA) に基づいています。これは Web Service Definition Language (WSDL) ドキュメントとして公開されているアダプタ・サービスを消費します。

この章は次の項で構成されています。

- 6.1 項「概要」
- 6.2 項「アダプタのデプロイメント」
- 6.3 項「新規アプリケーション・サーバー接続の構成」
- 6.4 項「サービス統合のためのアウトバウンド BPEL プロセスの設計 (J2CA 構成)」
- 6.5 項「イベント統合のためのインバウンド BPEL プロセスの設計 (J2CA 構成)」
- 6.6 項「BPEL を使用した同期 SAP R/3 イベントの処理 (J2CA 構成)」
- 6.7 項「サービス統合のためのアウトバウンド BPEL プロセスの設計 (BSE 構成)」

### 6.1 概要

Oracle BPEL Process Manager を統合するには、Oracle Application Adapter for SAP R/3 が Oracle BPEL Process Manager と同じ WLS コンテナにデプロイされている必要があります。基礎となるアダプタ・サービスは WSDL ファイルとして公開されている必要があります。これらのファイルは、アダプタのリクエスト/レスポンス (アウトバウンド) とイベント通知 (インバウンド) の両方のサービスについて、Oracle Adapter Application Explorer (アプリケーション・エクスプローラ) でデザインタイムに生成されます。詳細は、4-15 ページの「[WSDL の生成 \(J2CA 構成のみ\)](#)」を参照してください。

生成された WSDL ファイルは、インバウンドまたはアウトバウンドのアダプタ・サービスに適した BPEL プロセスを設計するために使用されます。完成した BPEL プロセスは、JDeveloper で正常にコンパイルされ、BPEL サーバーにデプロイされる必要があります。BPEL サーバーへのデプロイメント時に、新しく作成されたプロセスはすべて自動的に Oracle Enterprise Manager コンソールにデプロイされます。このコンソールで BPEL プロセスの実行、監視および管理や、アダプタ・イベントのリスニングを行います。

## 6.2 アダプタのデプロイメント

インストール時に、Oracle Application Adapter for SAP R/3 は、J2CA 1.0 リソース・アダプタとして WLS コンテナ内にデプロイされます。このアダプタは Oracle BPEL Process Manager と同じ WLS コンテナにデプロイされる必要があります。

## 6.3 新規アプリケーション・サーバー接続の構成

Oracle JDeveloper で新規アプリケーション・サーバー接続を構成するには、次のようにします。

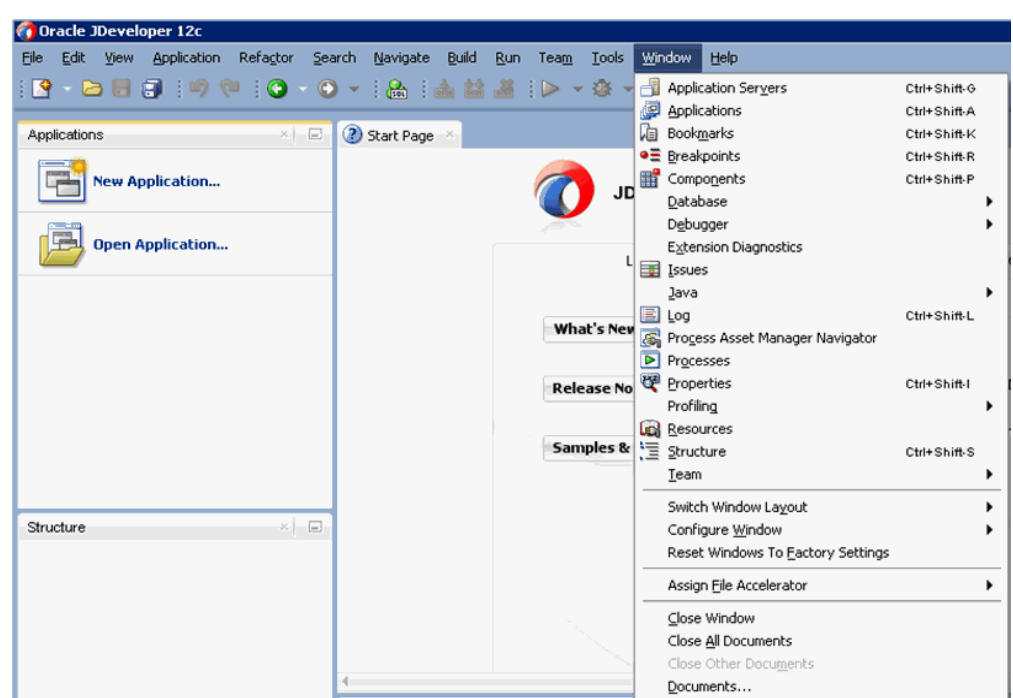
1. システムで **Oracle JDeveloper** を開きます。
2.  6-1 に示すように、メニュー・バーの「ウィンドウ」をクリックし、「アプリケーション・サーバー・ナビゲータ」を選択します。

図 6-1 アプリケーション・サーバー・ナビゲータ

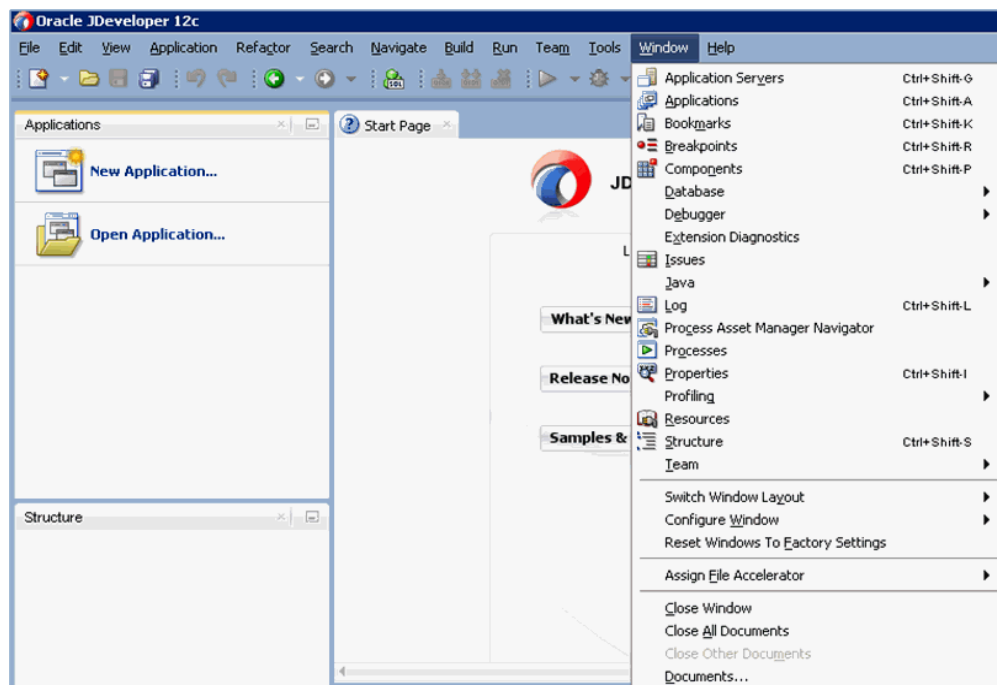
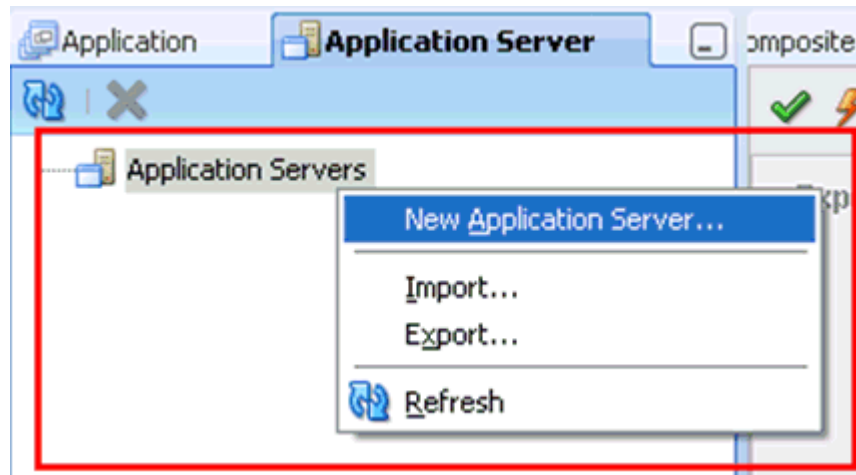


図 6-2 に示すように、「アプリケーション・サーバー」タブが表示されます。

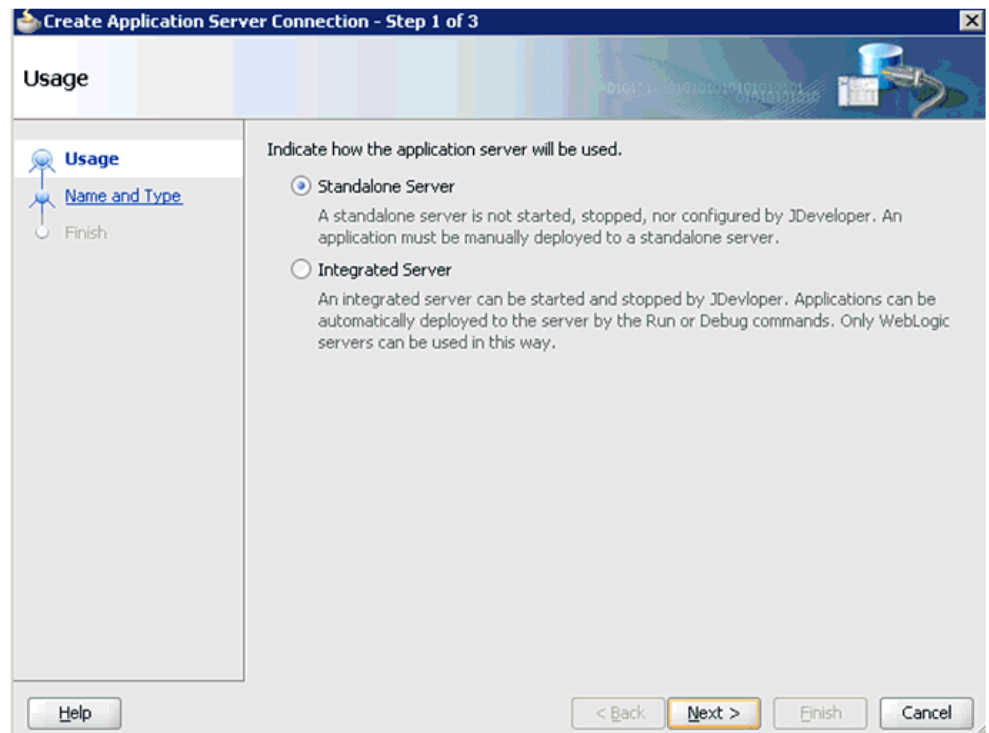
図 6-2 「アプリケーション・サーバー」タブ



3. 「アプリケーション・サーバー」を右クリックし、「新規アプリケーション・サーバー」を選択します。

図 6-3 に示すように、アプリケーション・サーバー接続の作成ウィザードが表示されます。

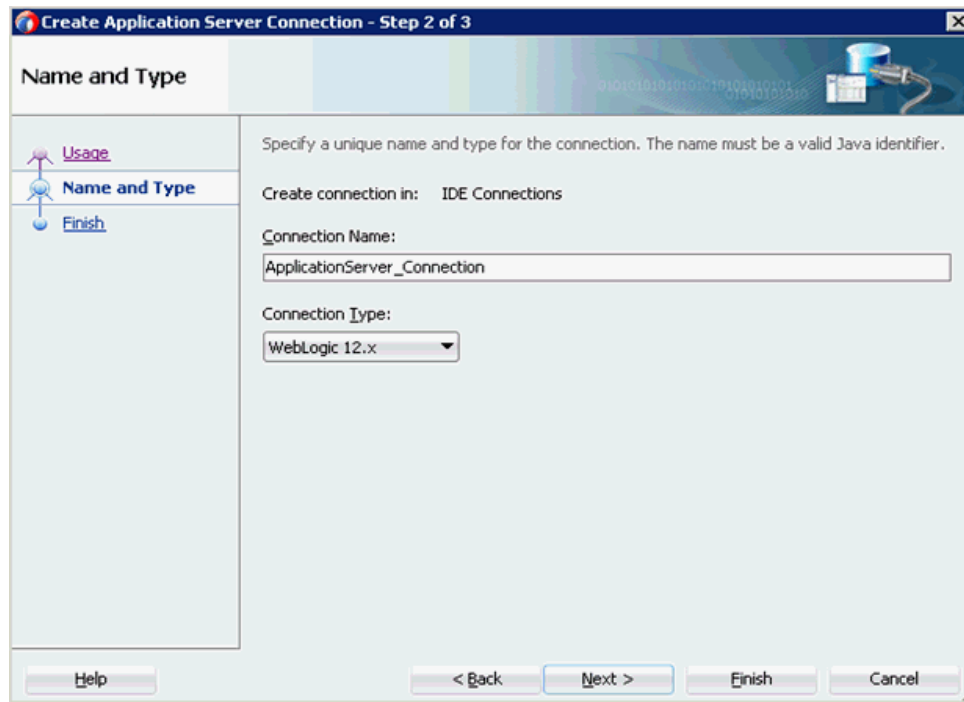
図 6-3 アプリケーション・サーバー接続の作成ウィザード



4. デフォルトの選択である「スタンドアロン・サーバー」のままにして、「次へ」をクリックします。

図 6-4 に示すように、「名前とタイプ」ページが表示されます。

図 6-4 「名前とタイプ」ページ



- アプリケーション・サーバー接続の新しい名前を指定し、「次へ」をクリックします。  
図 6-5 に示すように、「認証」ページが表示されます。

図 6-5 「認証」ページ

Create Application Server Connection - Step 3 of 6

Authentication

Specify a username and password to authenticate the connection.

Username:  
weblogic

Password:  
\*\*\*\*\*

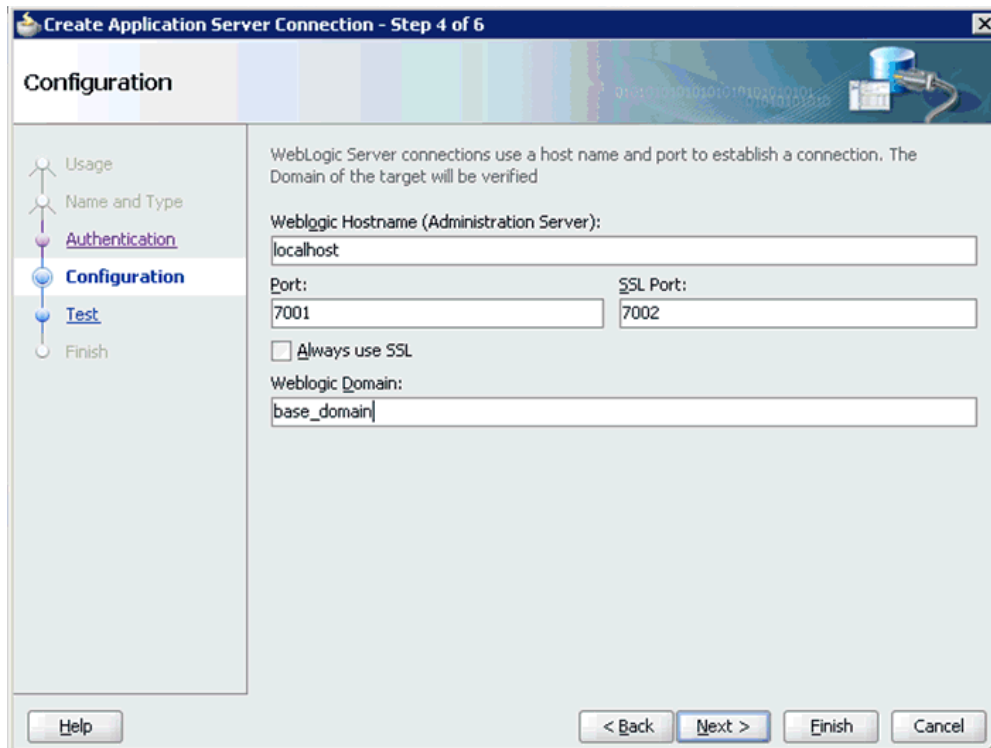
Usage  
Name and Type  
Authentication  
Configuration  
Test  
Finish

Help < Back Next > Finish Cancel

- 新規接続用に有効なユーザー名 (weblogic など) とパスワード (welcome1 など) を指定します。

7. 「次へ」をクリックします。  
☒ 6-6 に示すように、「構成」ページが表示されます。

図 6-6 「構成」ページ

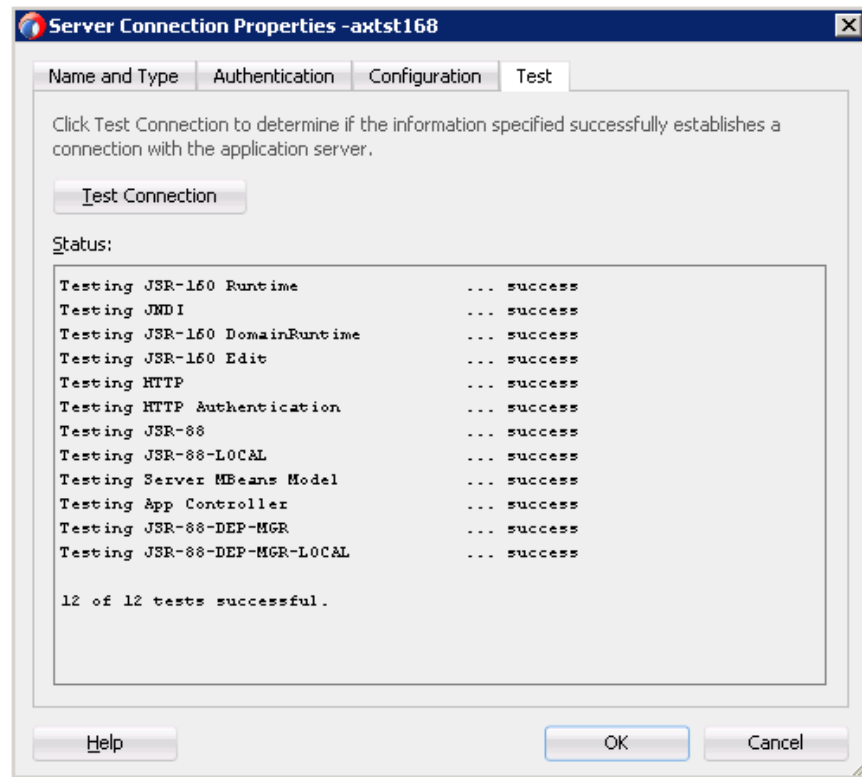


8. Oracle WebLogic ホスト名を指定します (localhost など)。これは、プロセスのデプロイが必要なシステム IP です。また、Oracle WebLogic ドメインも指定します (base\_domain など)。



9. 「次へ」をクリックします。  
図 6-7 に示すように、「テスト」ページが表示されます。

図 6-7 「テスト」ページ

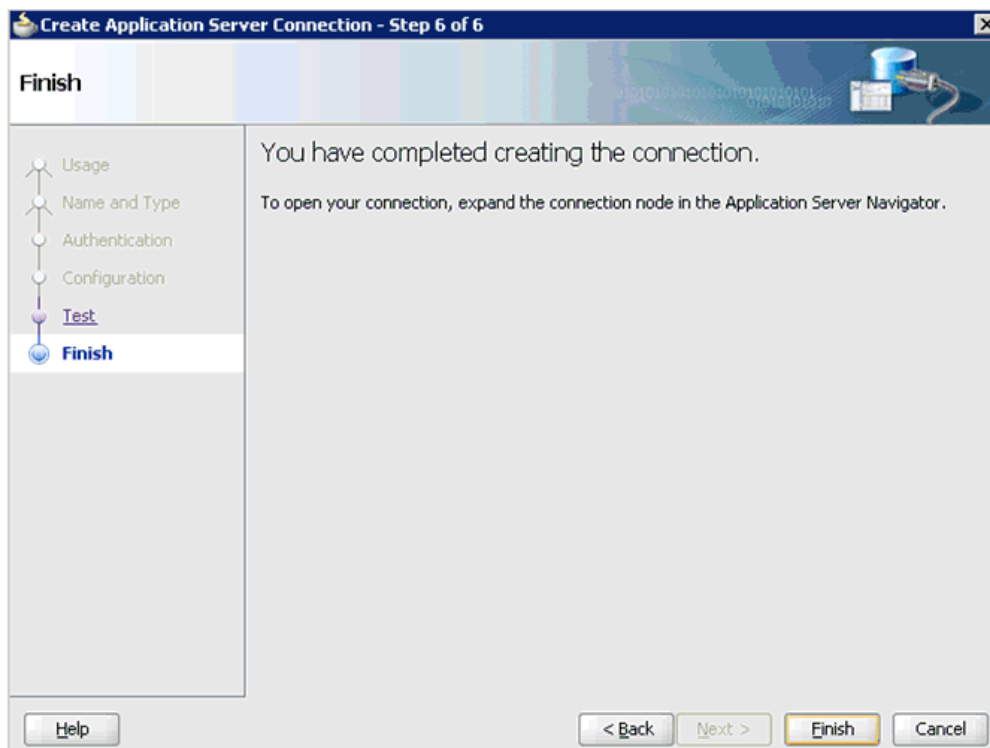


10. 「接続のテスト」をクリックします。
11. テスト・ステータスが成功になっていることを確認します。

12. 「次へ」をクリックします。

図 6-8 に示すように、「終了」ページが表示されます。

図 6-8 「終了」ページ



13. 「終了」をクリックします。

新規アプリケーション・サーバー接続が左ペイン(「アプリケーション・サーバー」タブ)に表示されます。

## 6.4 サービス統合のためのアウトバウンド BPEL プロセスの設計 (J2CA 構成)

この項では、サービス統合のためのアウトバウンド BPEL プロセスを設計する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\J2CA\Outbound_Project
```

アウトバウンド・デザインタイム構成を完了するには、次のツールが必要です。

- Oracle Adapter Application Explorer (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPEL Designer (JDeveloper)

---

**注意：** この章の例では、JDeveloper を使用しています。

---

ここでは、次の項目について説明します。

- [6.4.1 項「リクエスト/レスポンス・サービスの WSDL の生成」](#)

- 6.4.2 項「SOA 用の空のコンポジットの作成」
- 6.4.3 項「BPEL アウトバウンド・プロセスの定義」
- 6.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」
- 6.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」
- 6.4.6 項「アウトバウンド BPEL およびメディアエータ・プロセスのテスト」

## 6.4.1 リクエスト/レスポンス・サービスの WSDL の生成

BPEL プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。

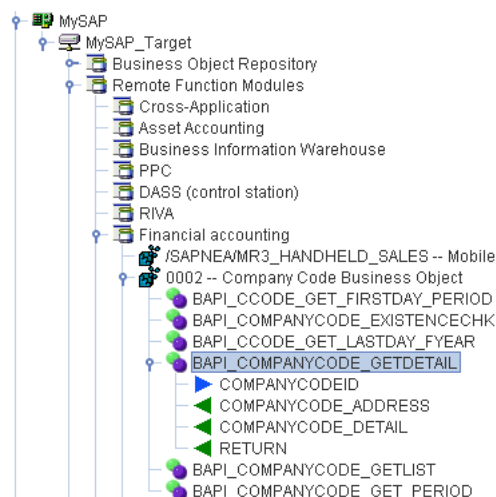
リクエスト/レスポンス・サービスの WSDL ファイルを生成するには、次のようにします。

1. アプリケーション・エクスプローラを起動し、定義済の MySAP ターゲット (J2CA 構成) に接続します。

アプリケーション・エクスプローラの起動、ターゲットの定義および SAP R/3 への接続の詳細は、4-1 ページの 4.1 項「アプリケーション・エクスプローラの起動」および 4-5 ページの 4.4.1 項「SAP R/3 へのターゲットの定義」を参照してください。

2. 接続先の MySAP ターゲットを展開します。
3. 図 6-9 に示すように、「リモート関数モジュール」→「Financial Accounting」→「0002 -- Company Code Business Object」を展開し、「BAPI\_COMPANYCODE\_GETDETAIL」を選択します。

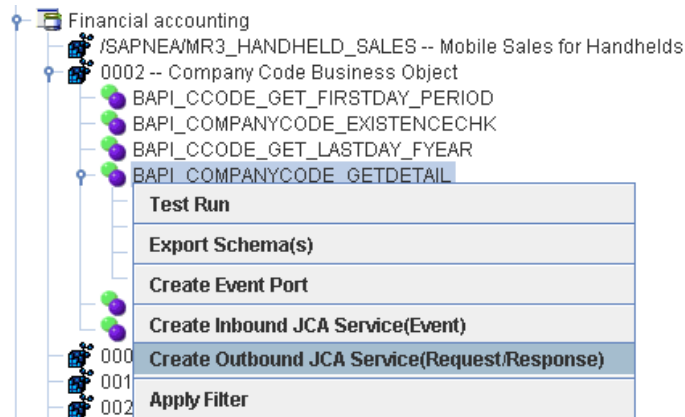
図 6-9 接続された MySAP\_Target



4. 「BAPI\_COMPANYCODE\_GETDETAIL」ノードを右クリックします。

図 6-10 に示すように、メニューが表示されます。

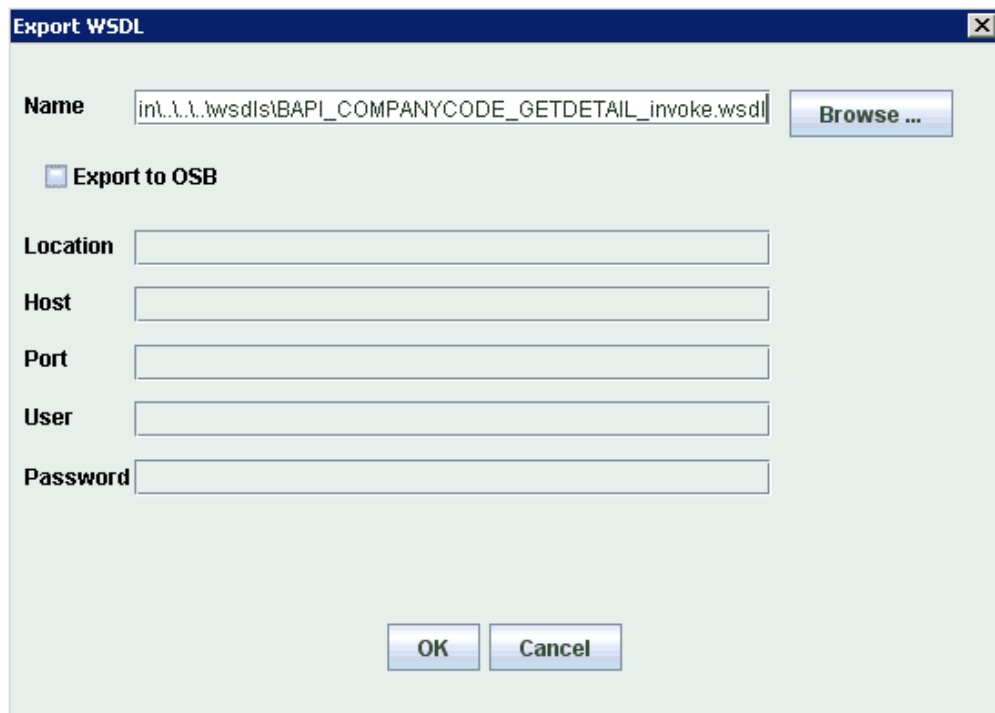
図 6-10 「BAPI\_COMPANYCODE\_GETDETAIL」ノード



5. 「アウトバウンド JCA サービス (リクエスト/レスポンス) の作成」をクリックします。

図 6-11 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

図 6-11 「WSDL のエクスポート」ダイアログ



6. 「OK」をクリックします。

これで、JDeveloper で BPEL アウトバウンド・プロセスを定義するために必要な最初の手順として、SOA 用の空のコンポジットを作成できるようになりました。

## 6.4.2 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

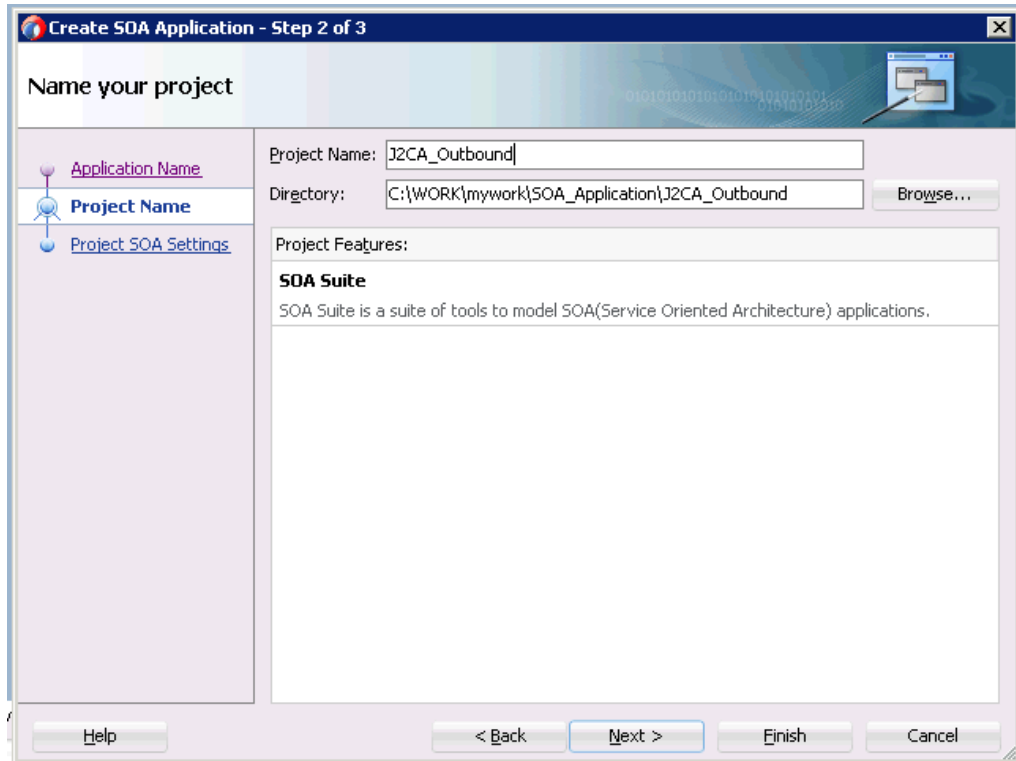
1. 新規 SOA アプリケーションを作成します。
2. 図 6-12 に示すように、新規 SOA アプリケーションの名前を入力し、「次へ」をクリックします。

図 6-12 「アプリケーションの名前付け」ページ

Figure 6-12 shows the 'Name your application' step of the 'Create SOA Application' wizard. The wizard is titled 'Create SOA Application - Step 1 of 3'. The main heading is 'Name your application'. On the left, a tree view shows 'Application Name' selected, with 'Project Name' and 'Project SOA Settings' below it. The main area contains three input fields: 'Application Name' with the text 'SOA\_Application', 'Directory' with the text 'C:\WORK\mywork\SOA\_Application' and a 'Browse...' button, and 'Application Package Prefix' which is empty. At the bottom, there are buttons for 'Help', '< Back', 'Next >', 'Finish', and 'Cancel'.

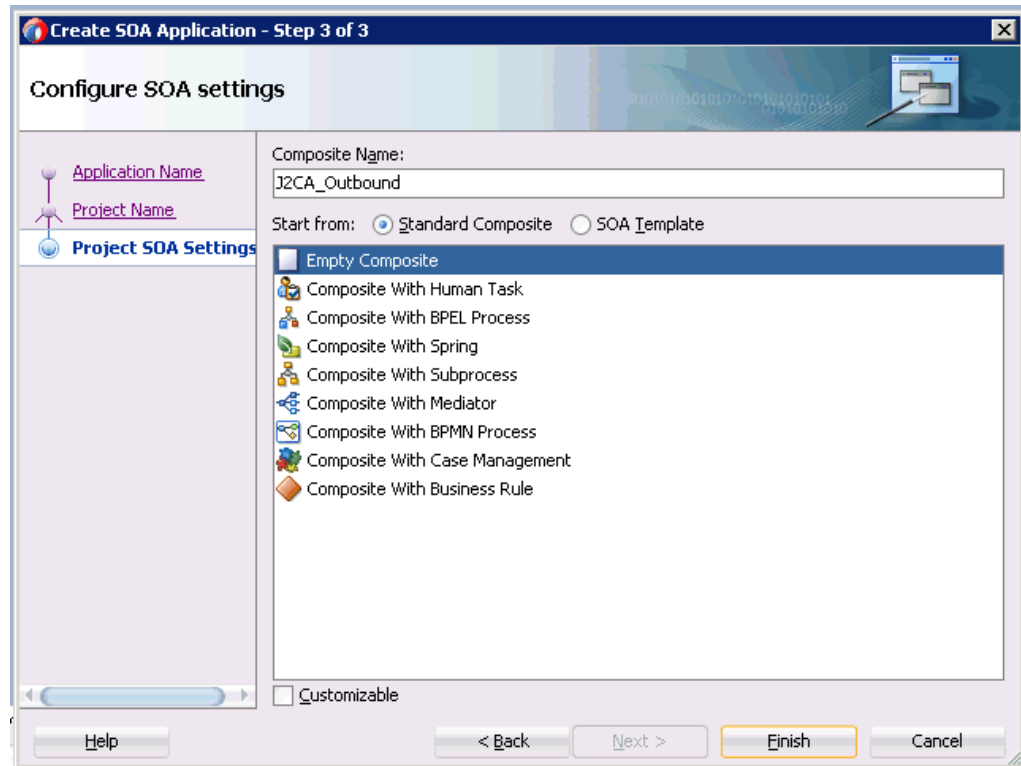
図 6-13 に示すように、「プロジェクトの名前付け」ページが表示されます。

図 6-13 「プロジェクトの名前付け」ページ



- プロジェクト名を入力し、「次へ」をクリックします。  
図 6-14 に示すように、SOA 設定の構成ページが表示されます。

図 6-14 SOA 設定の構成ページ



- 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

### 6.4.3 BPEL アウトバウンド・プロセスの定義

この項では、BPEL アウトバウンド・プロセスを定義する方法について、次の項目で説明します。

- 6.4.3.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」
- 6.4.3.2 項「アウトバウンド BPEL プロセス・コンポーネントの構成」
- 6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」

### 6.4.3.1 サード・パーティ・アダプタ・サービス・コンポーネントの構成

サード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

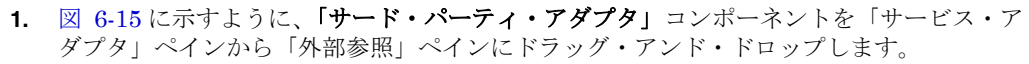
1.  6-15 に示すように、「サード・パーティ・アダプタ」コンポーネントを「サービス・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。

図 6-15 サード・パーティ・アダプタ・コンポーネント

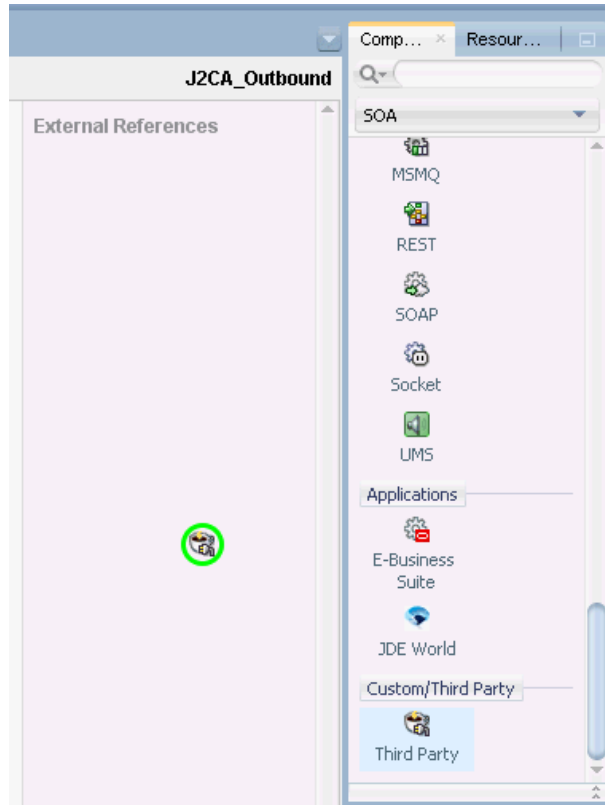
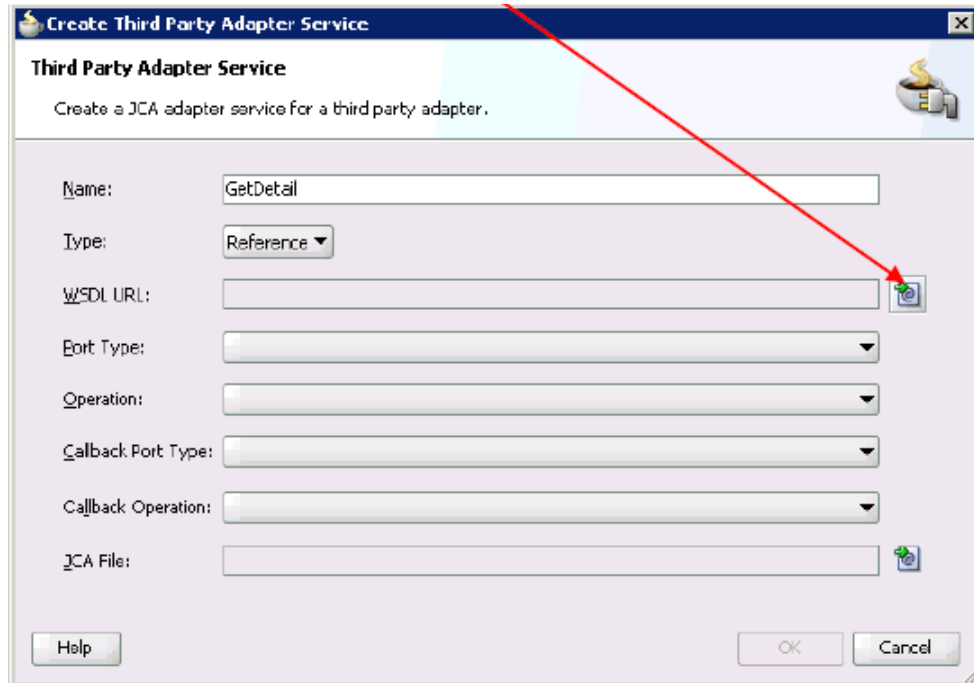




図 6-16 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

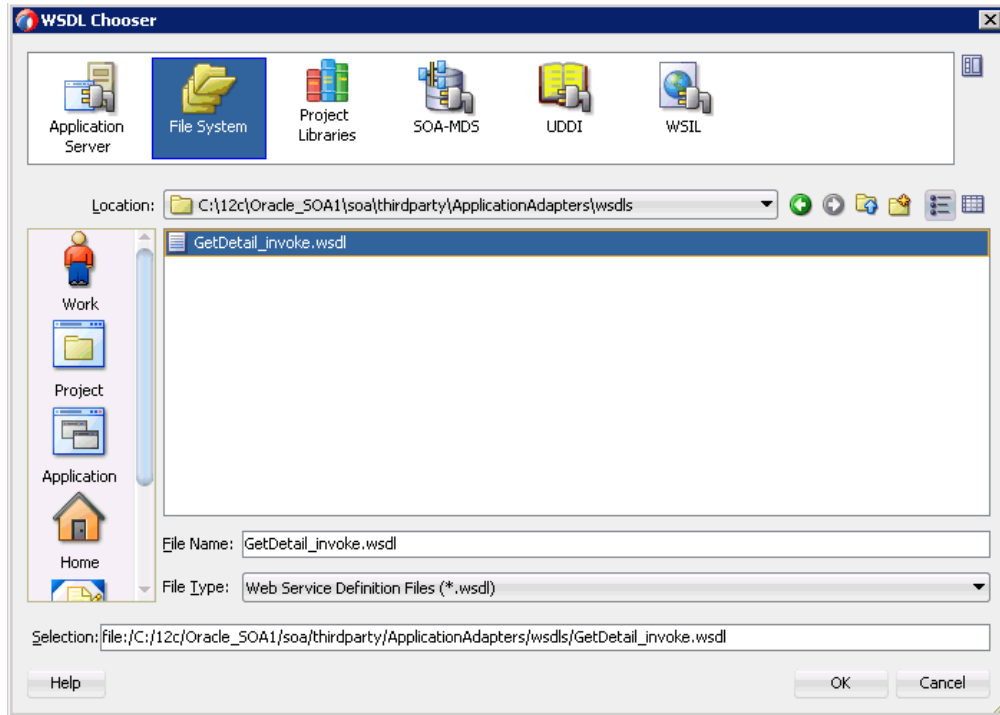
図 6-16 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



2. サード・パーティ・アダプタ・サービスの名前を入力します。
3. 「タイプ」リストで「参照」が選択されていることを確認します (デフォルト)。

4. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。  
図 6-17 に示すように、「WSDL の選択」ダイアログが表示されます。

図 6-17 「WSDL の選択」ダイアログ

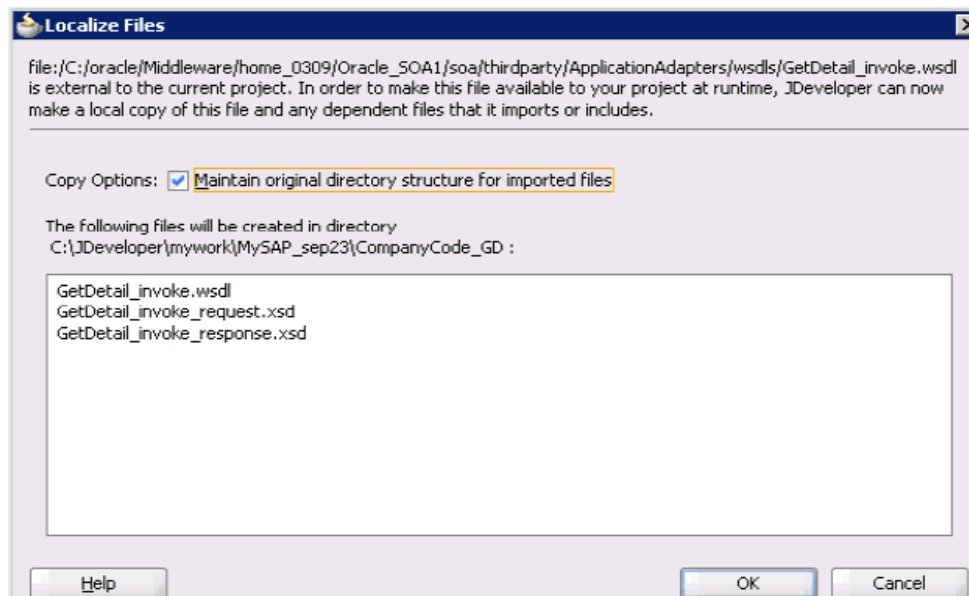


5. 次のディレクトリでアウトバウンド WSDL ファイルを参照して選択します。  
<ADAPTER\_HOME>\wsdls

## 6. 「OK」をクリックします。

図 6-18 に示すように、「ファイルのローカライズ」ダイアログが表示されます。

図 6-18 「ファイルのローカライズ」ダイアログ

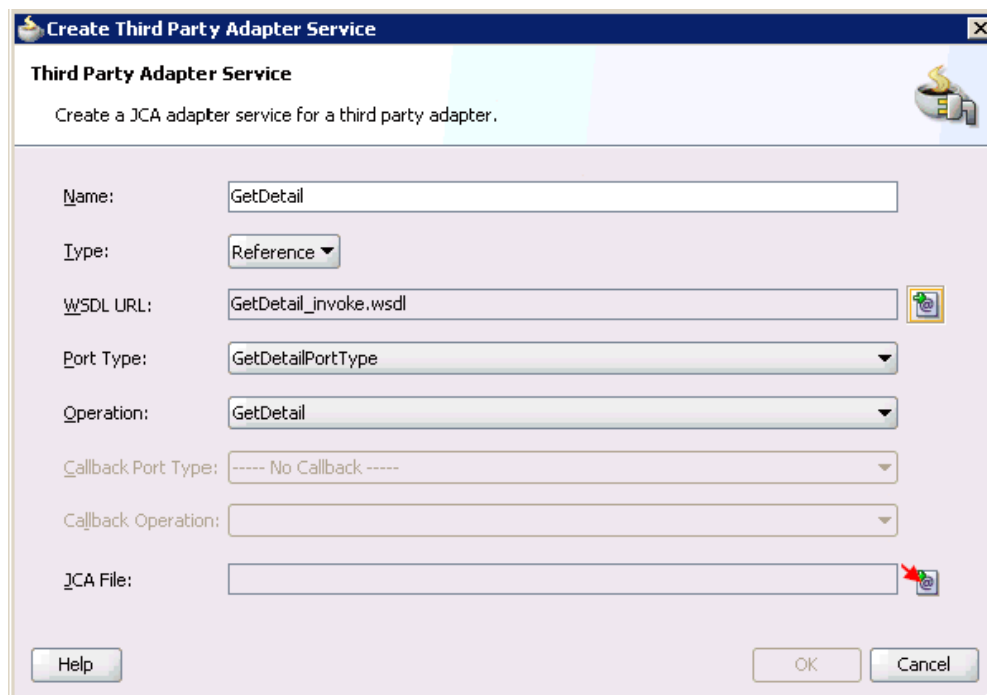


## 7. 「OK」をクリックします。

アウトバウンド WSDL ファイル、および関連するリクエストとレスポンスの XML スキーマ・ファイル (.xsd) が、作成されたプロジェクト・フォルダにインポートされます。

図 6-19 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

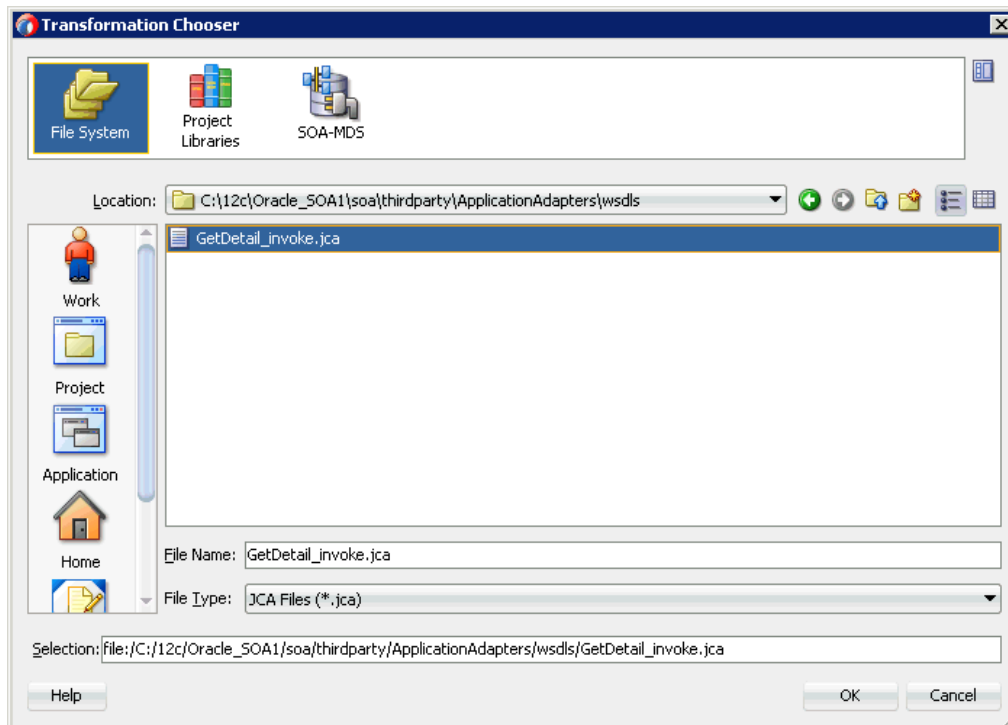
図 6-19 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



8. 「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。

図 6-20 に示すように、トランスフォーメーション・チューザ・ダイアログが表示されま

図 6-20 トランスフォーメーション・チューザ・ダイアログ



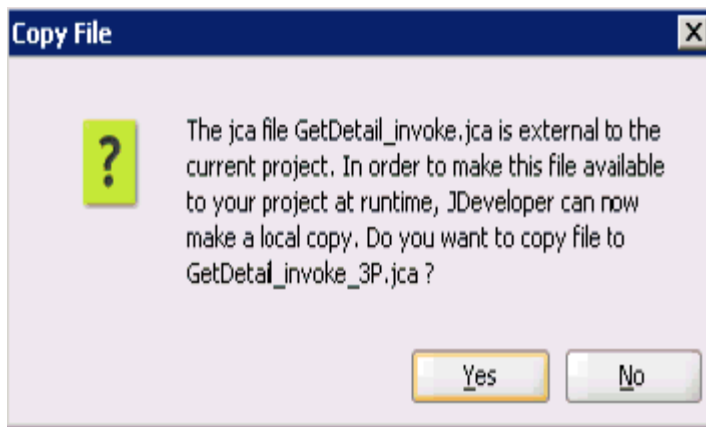
9. 次のディレクトリで JCA プロパティ・ファイルを参照して選択します。

<ADAPTER\_HOME>\wsdls

10. 「OK」をクリックします。

図 6-21 に示すように、「ファイルのコピー」メッセージが表示されます。

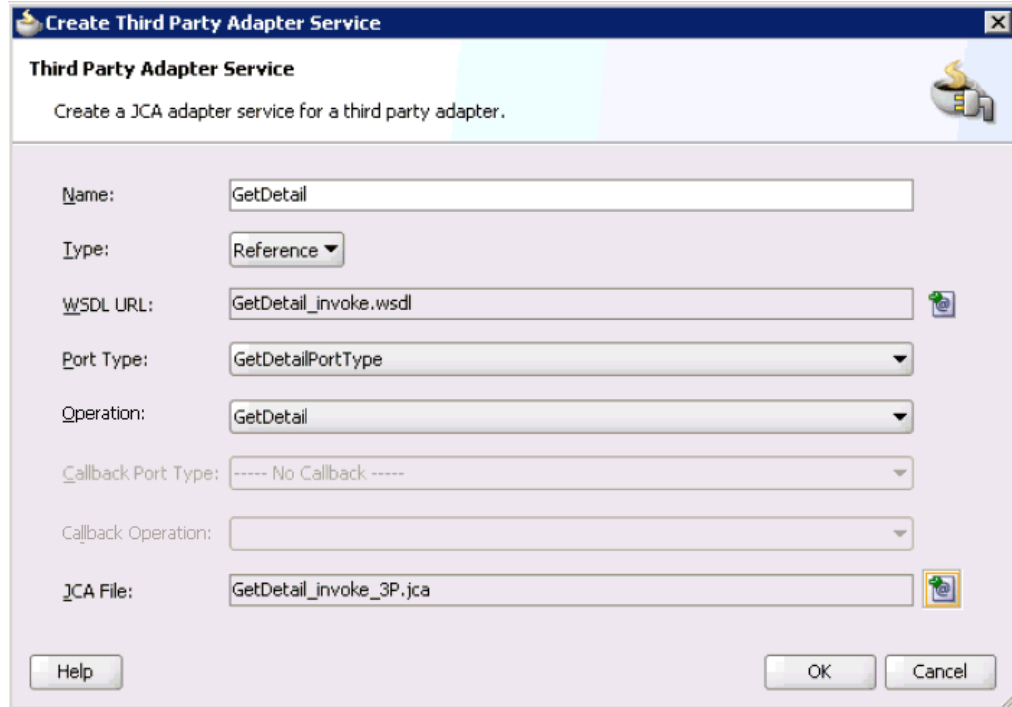
図 6-21 「ファイルのコピー」メッセージ



**11. 「はい」 をクリックします。**

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

図 6-22 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

**図 6-22 「サード・パーティ・アダプタ・サービスの作成」ダイアログ****12. 「OK」 をクリックします。**

サード・パーティ・アダプタ・サービス・コンポーネント (GetDetail) が作成され、「外部参照」ペインに表示されます。

これで、アウトバウンド BPEL プロセス・コンポーネントを構成する準備ができました。

### 6.4.3.2 アウトバウンド BPEL プロセス・コンポーネントの構成

アウトバウンド BPEL プロセス・コンポーネントを構成するには、次のようにします。

1. 「BPEL プロセス」コンポーネントを「コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

☒ 6-23 に示すように、「BPEL プロセスの作成」ダイアログが表示されます。

図 6-23 「BPEL プロセスの作成」ダイアログ

2. 「名前」フィールドに、新規アウトバウンド BPEL プロセス・コンポーネントを識別する名前を入力します (cc\_getdetail など)。

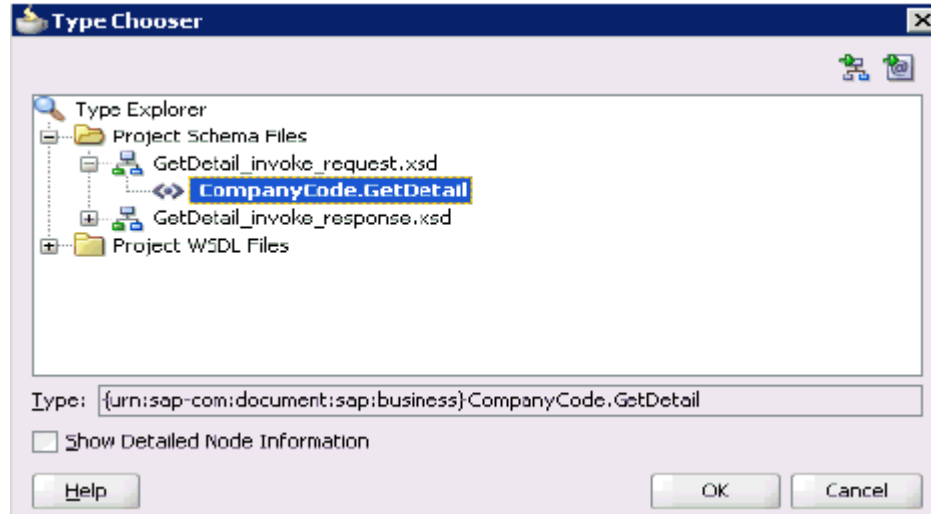
デフォルトでは、「BPEL 2.0 仕様」オプションが選択されています。

3. 「テンプレート」リストで「同期 BPEL プロセス」を選択します。

- 「入力」フィールドの右側にある「参照」アイコンをクリックし、関連する XML リクエスト・スキーマ・ファイルを選択します。

図 6-24 に示すように、「タイプ・チューザ」ダイアログが表示されます。

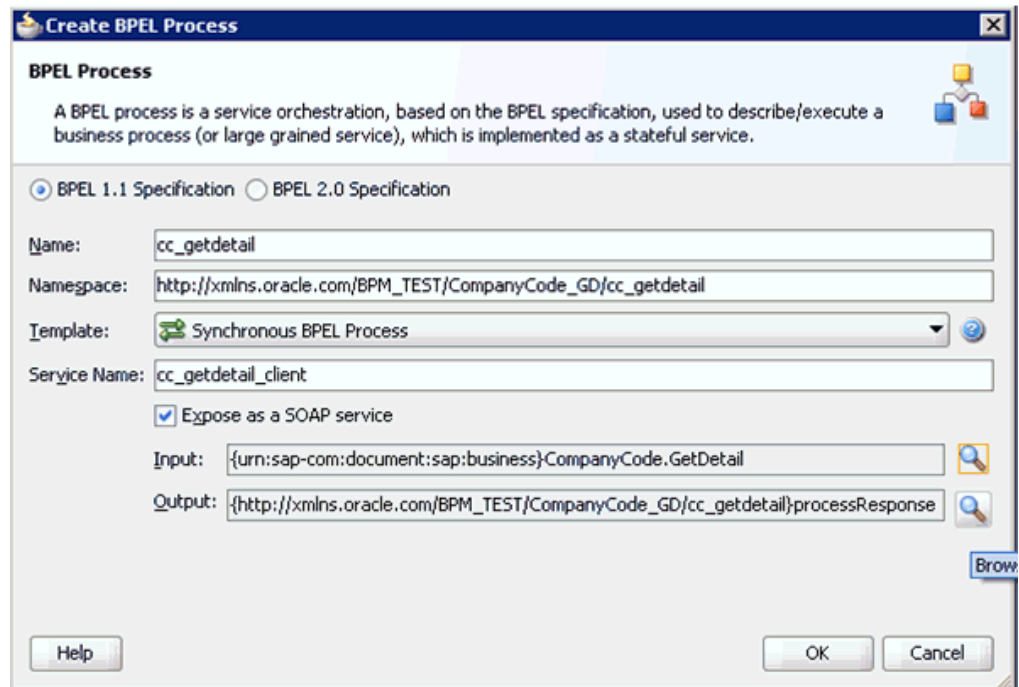
図 6-24 「タイプ・チューザ」ダイアログ



- 「プロジェクトのスキーマ・ファイル」→「GetDetail\_invoke\_request.xsd」を展開し、「CompanyCode.GetDetail」を選択します。
- 「OK」をクリックします。

図 6-25 に示すように、「BPEL プロセスの作成」ダイアログに戻ります。

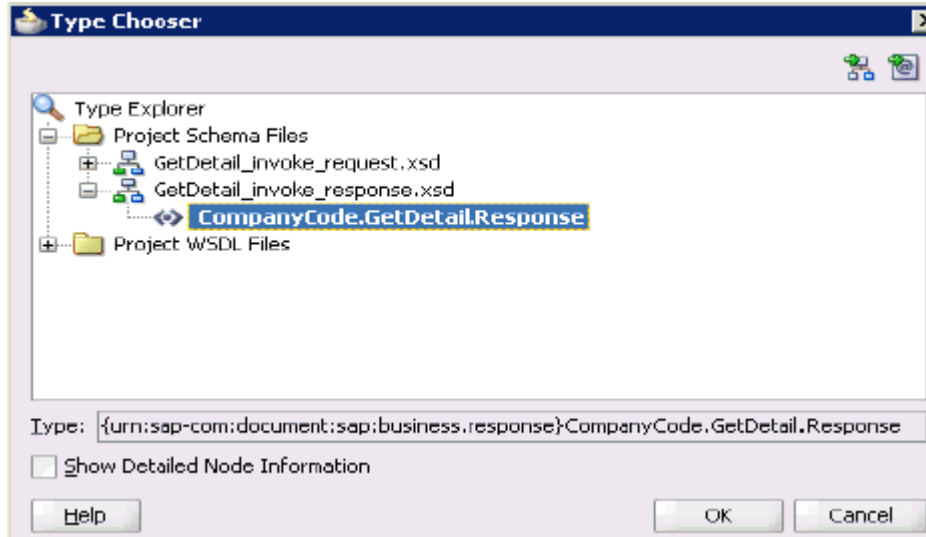
図 6-25 「BPEL プロセスの作成」ダイアログ



7. 「出力」フィールドの右側にある「参照」アイコンをクリックし、関連する XML レスポンス・スキーマ・ファイルを選択します。

図 6-26 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 6-26 「タイプ・チューザ」ダイアログ

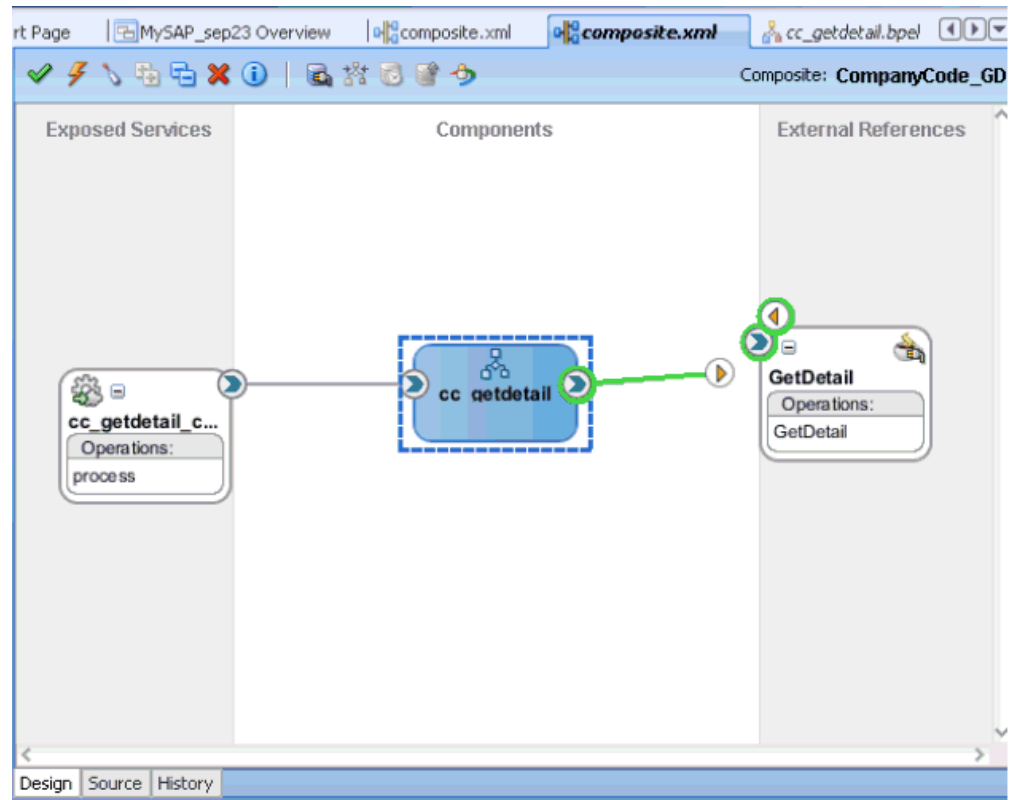


8. 「プロジェクトのスキーマ・ファイル」 → 「GetDetail\_invoke\_response.xsd」を展開し、「CompanyCode.GetDetailResponse」を選択します。
9. 「OK」をクリックします。  
「BPEL プロセスの作成」ダイアログに戻ります。
10. 「OK」をクリックします。



11. 図 6-27 に示すように、アウトバウンド BPEL プロセス・コンポーネント (cc\_getdetail) と サード・パーティ・アダプタ・サービス・コンポーネント (GetDetail) 間の接続を作成します。

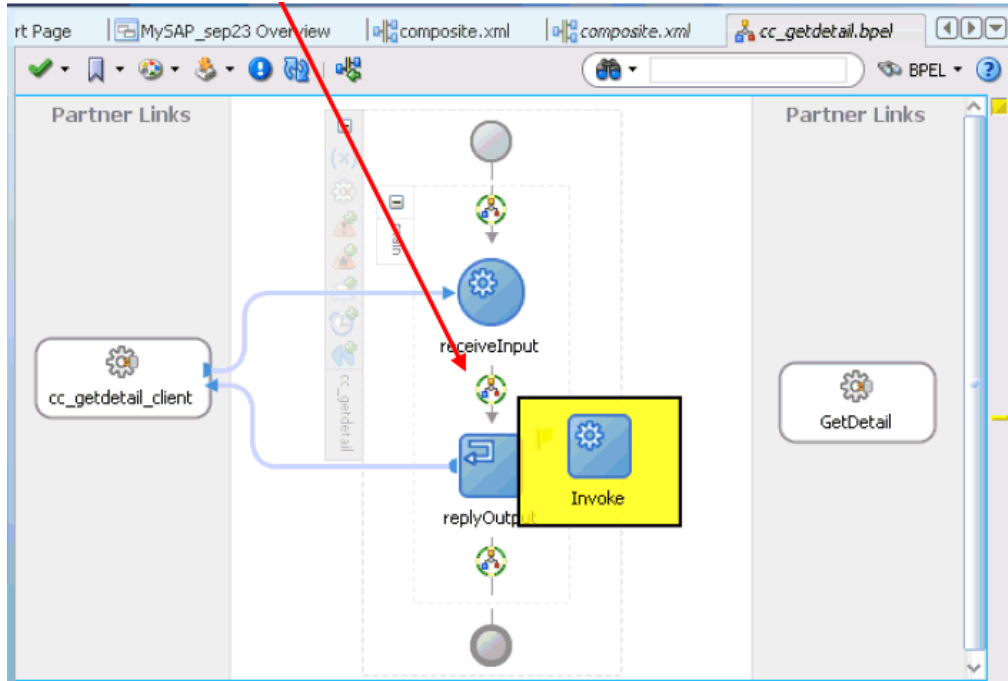
図 6-27 作成された接続



12. 「コンポーネント」ペインにあるアウトバウンド BPEL プロセス・コンポーネント (cc\_getdetail) をダブルクリックします。

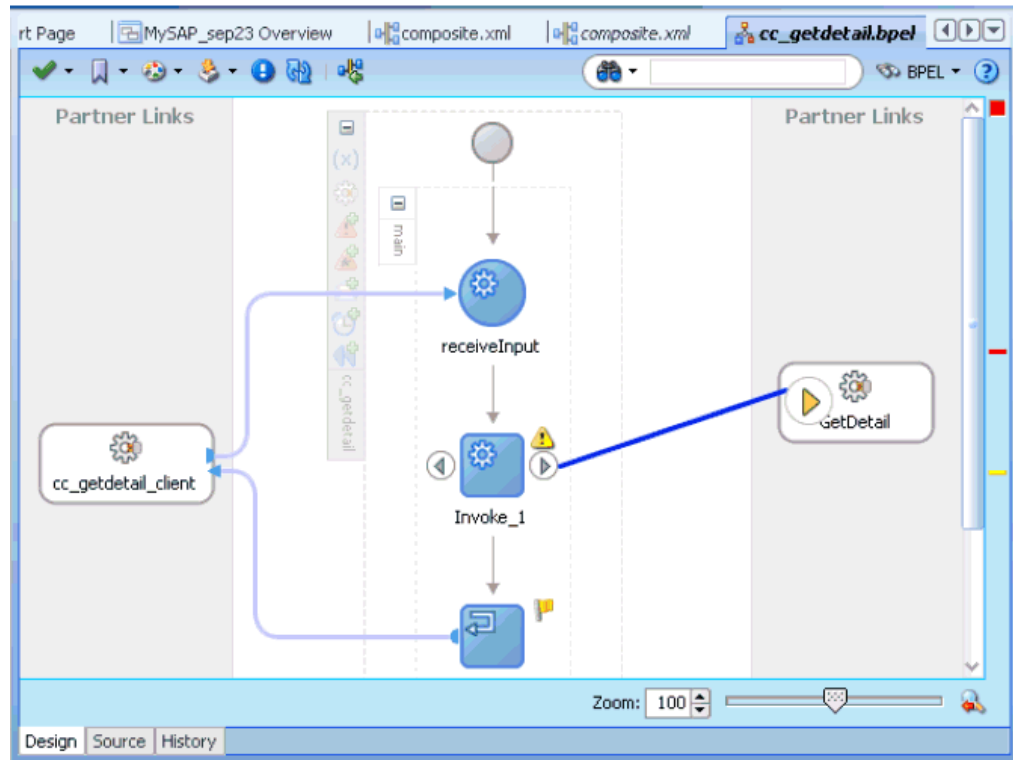
13. 図 6-28 に示すように、「BPEL コンストラクト」 - 「Web サービス」の下の「起動」アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップし、**receiveInput** アクティビティ・コンポーネントと **replyOutput** アクティビティ・コンポーネントの間に配置します。

図 6-28 「起動」アクティビティ・コンポーネント



14. 図 6-29 に示すように、新規「起動」アクティビティ・コンポーネント (Invoke1) とサーブド・パーティ・アダプタ・サービス・コンポーネント (GetDetail) 間の接続を作成します。

図 6-29 作成された接続



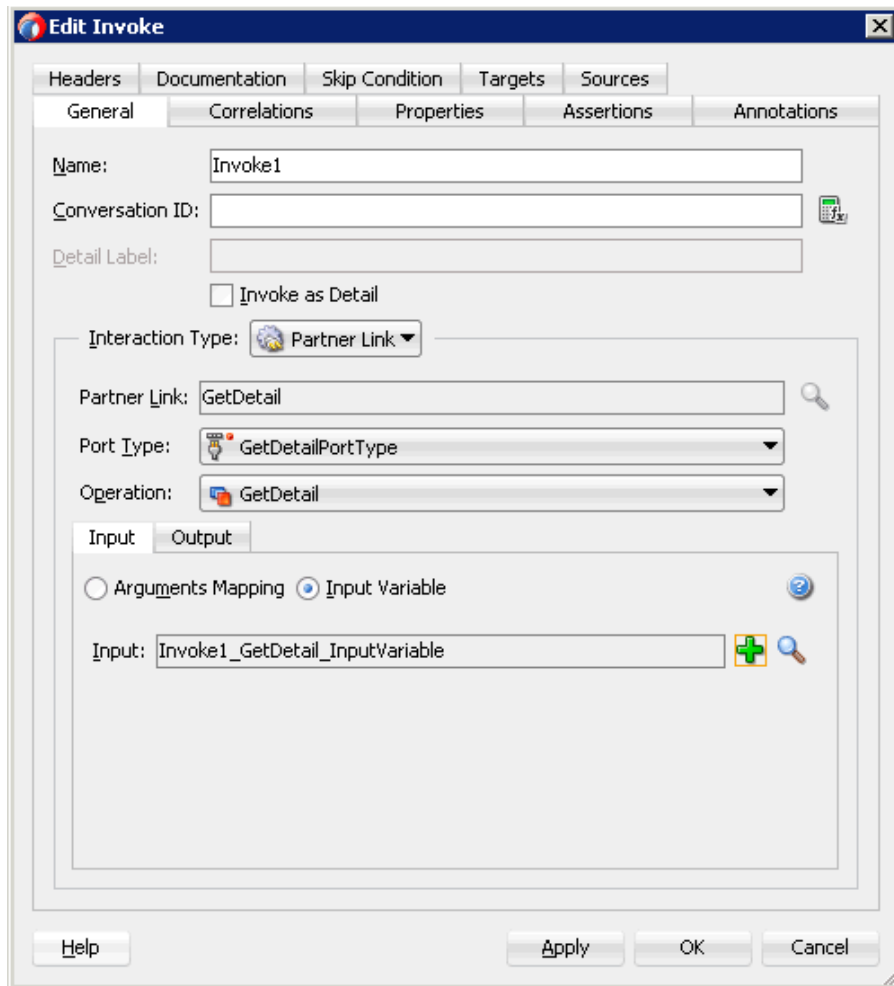
「起動の編集」ダイアログが表示されます。

15. 「入力」フィールドの右側にある**プラス記号**アイコンをクリックし、新規入力変数を構成します。

「変数の作成」ダイアログが表示されます。

16. 新規入力変数に提供されているデフォルト値を受け入れて、「OK」をクリックします。  
 図 6-30 に示すように、「起動の編集」ダイアログに戻ります。

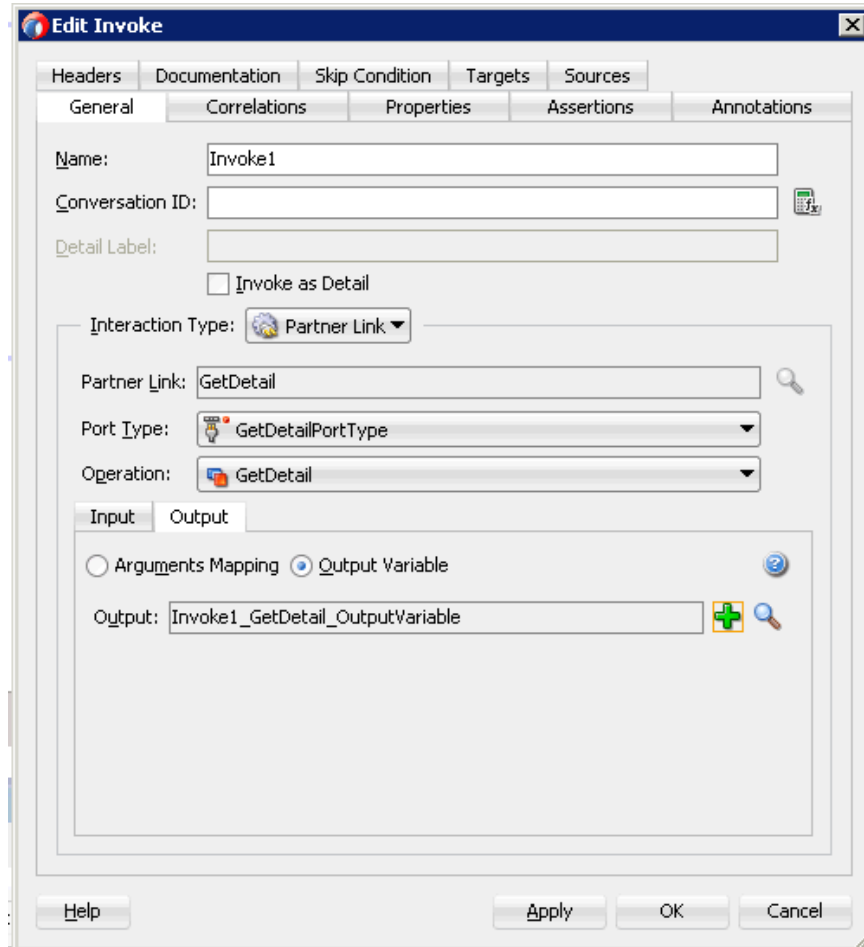
図 6-30 「起動の編集」ダイアログ



17. 「出力」タブを選択し、「出力」フィールドの右側にある**プラス記号**アイコンをクリックし、新規出力変数を構成します。  
 「変数の作成」ダイアログが表示されます。

18. 新規出力変数に提供されているデフォルト値を受け入れて、「OK」をクリックします。  
図 6-31 に示すように、「起動の編集」ダイアログに戻ります。

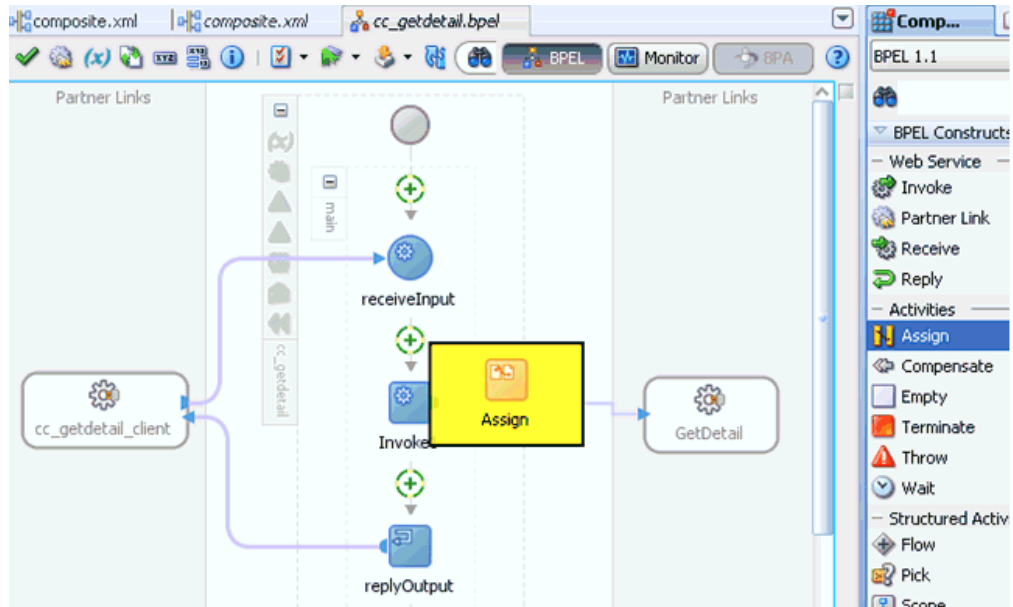
図 6-31 「起動の編集」ダイアログ



19. 「適用」をクリックしてから、「OK」をクリックします。

20. 図 6-32 に示すように、「BPEL コンストラクト」-「基本アクティビティ」コンポーネントの下の「割当て」アクティビティを「コンポーネント」ペインにドラッグ・アンド・ドロップし、「受信」アクティビティ・コンポーネント (receiveInput) と「起動」アクティビティ・コンポーネント (Invoke1) の間に配置します。

図 6-32 「割当て」アクティビティ・コンポーネント

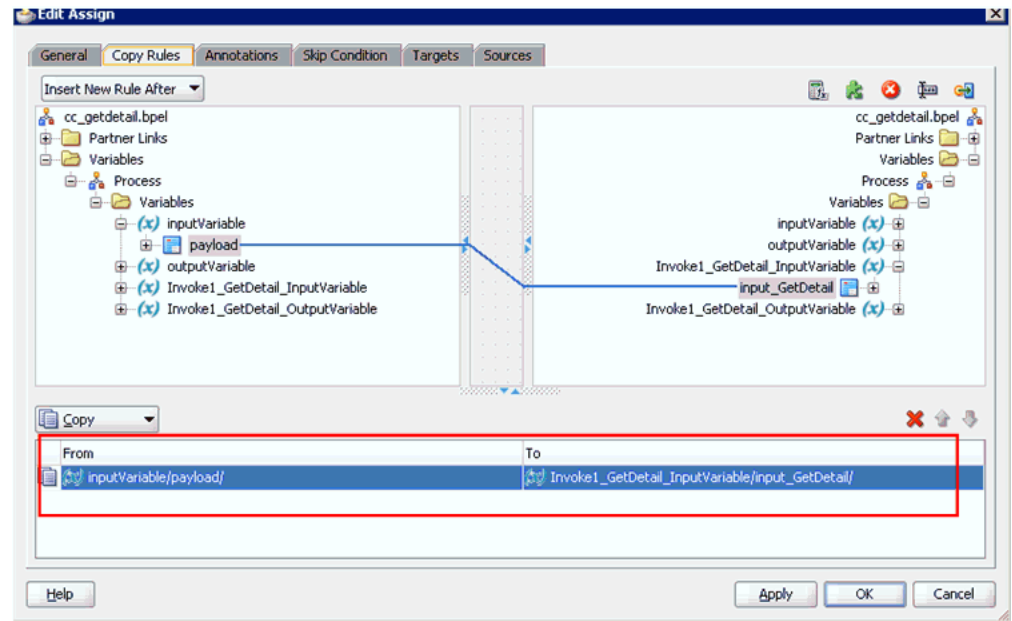


21. 新規「割当て」アクティビティ・コンポーネント (**Assign1**) をダブルクリックします。  
「割当ての編集」ダイアログが表示されます。
22. 左ペインで、「変数」の下の「**InputVariable**」を展開し、「ペイロード」を選択します。
23. 右ペインで、「変数」の下の「**Invoke1\_GetDetail\_InputVariable**」を展開し、「**input\_GetDetail**」を選択します。

24. 「ペイロード」変数を「input\_GetDetail」変数にドラッグしてマップします。

図 6-33 に示すように、マップされた変数が強調表示された領域に表示されます。

図 6-33 「割当ての編集」ダイアログ

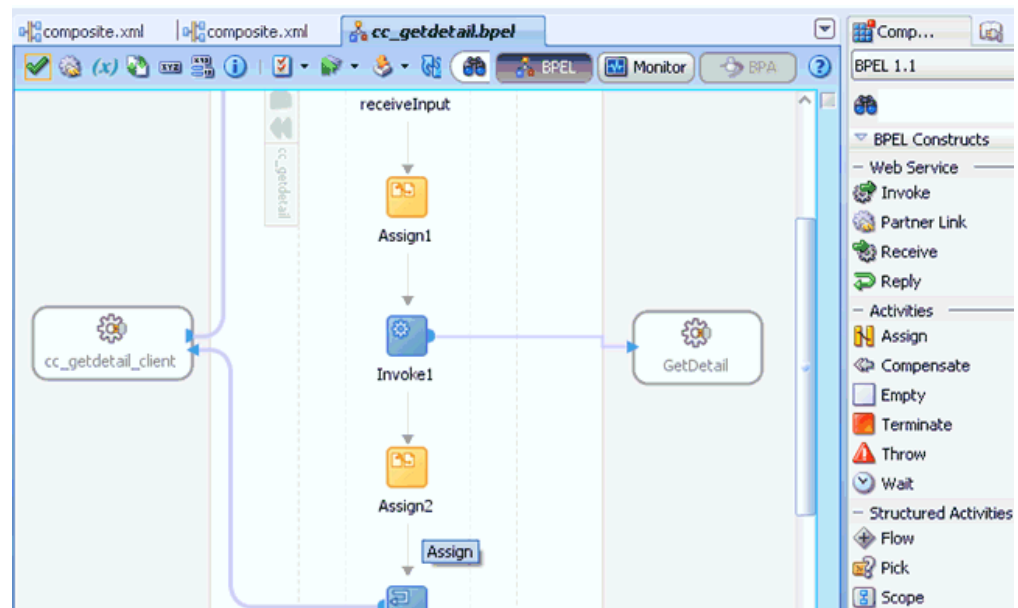


25. 「適用」をクリックしてから、「OK」をクリックします。

26. 「割当て」アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップし、「起動」アクティビティ (Invoke1) と「返信」アクティビティ (replyOutput) の間に配置します。

27. 図 6-34 に示すように、新規「割当て」アクティビティ・コンポーネント (Assign2) をダブルクリックします。

図 6-34 新規「割当て」アクティビティ・コンポーネント

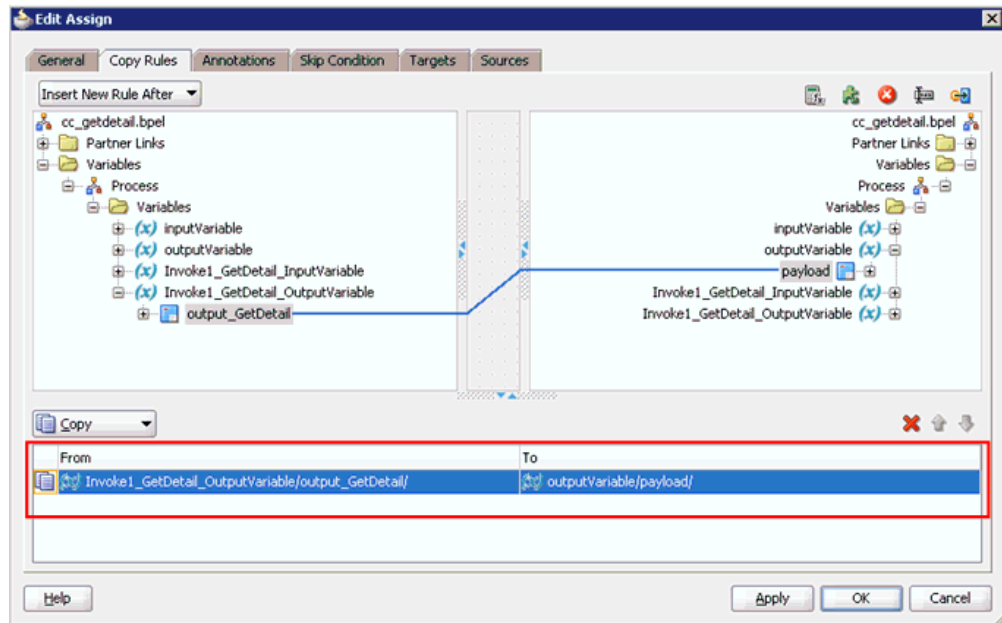


「割当ての編集」ダイアログが表示されます。

28. 左ペインで、「変数」の下の「**Invoke1\_GetDetail\_OutputVariable**」を展開し、「**output\_GetDetail**」を選択します。
29. 右ペインで、「変数」の下の「**outputVariable**」を展開し、「**ペイロード**」を選択します。
30. 「**output\_GetDetail**」変数を「**ペイロード**」変数にドラッグしてマップします。

図 6-35 に示すように、マップされた変数が強調表示された領域に表示されます。

図 6-35 「割当ての編集」ダイアログ

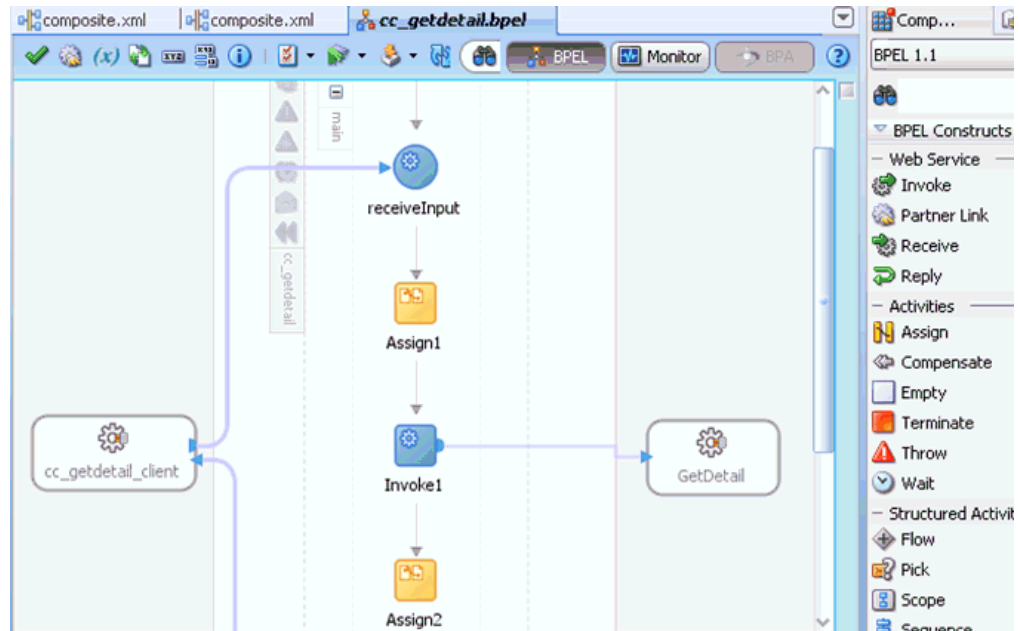




31. 「適用」をクリックしてから、「OK」をクリックします。

図 6-36 に示すように、アクティビティ・コンポーネント・ペインに戻ります。

図 6-36 アクティビティ・コンポーネント・ペイン



32. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド BPEL プロセス・コンポーネントを保存します。

これで、BPEL アウトバウンド・プロセスをデプロイする準備ができました。

### 6.4.3.3 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整するには、次のようにします。

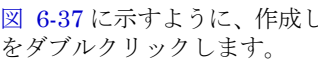
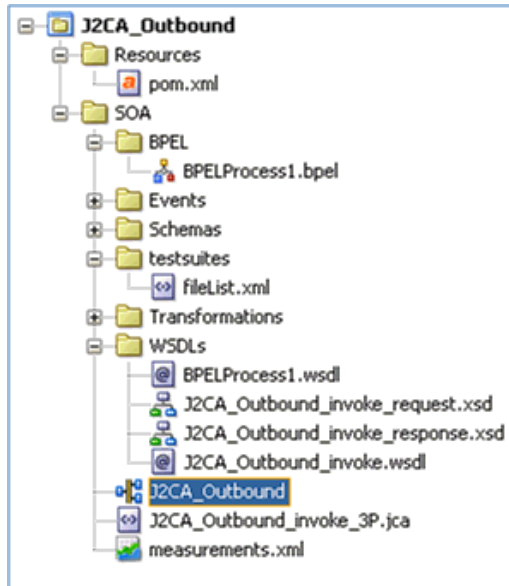
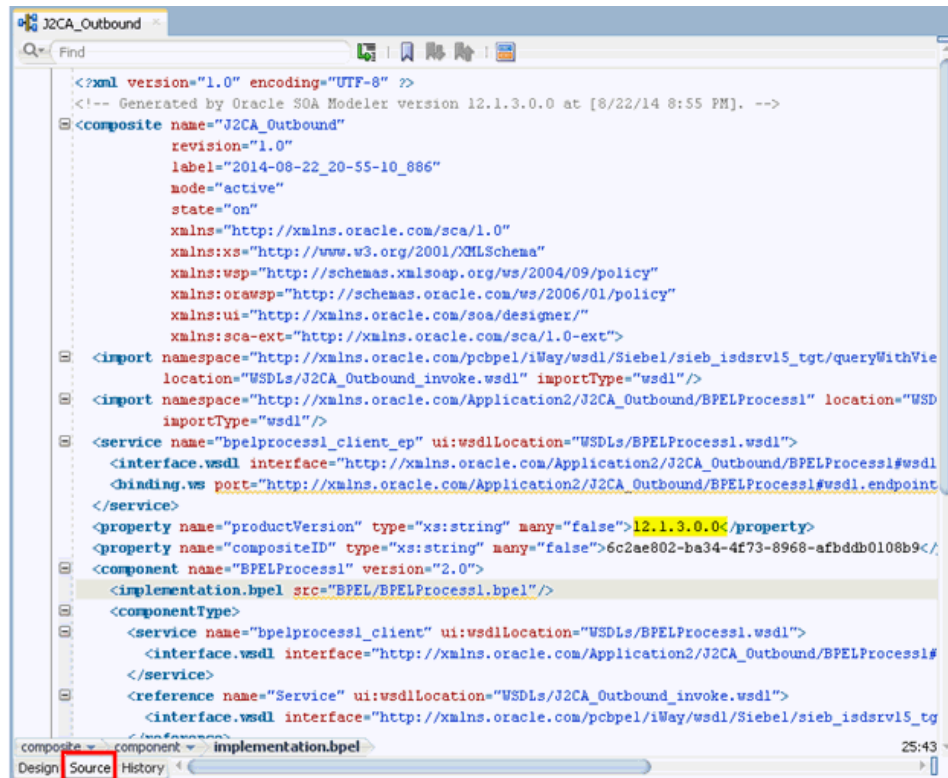
1.  6-37 に示すように、作成したプロセスの **J2CA\_Outbound** (作成した BPEL プロセス) をダブルクリックします。

図 6-37 「J2CA\_Outbound」ノード



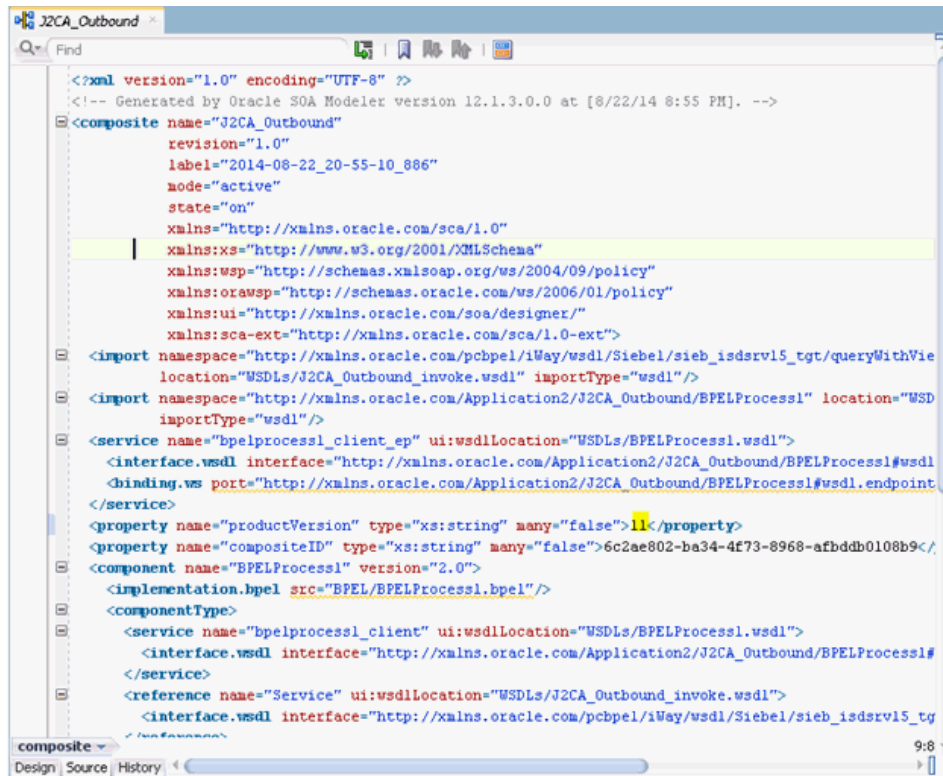
2. 図 6-38 に示すように、開いているプロセスの下部にある「ソース」タブをクリックします。

図 6-38 「ソース」タブ



3. 図 6-39 に示すように、productVersion プロパティ値を「12.1.3.0.0」から「11」に変更します。

図 6-39 プロパティ値



4. 変更を保存してプロジェクトのデプロイに進みます。

## 6.4.4 BPEL アウトバウンド・プロセスのデプロイ

BPEL アウトバウンド・プロセスをデプロイするには、次のようにします。

1. 図 6-40 に示すように、左ペインのプロジェクト名を右クリックし、「デプロイ」を選択してから「J2CA\_Outbound」をクリックします。

図 6-40 J2CA\_Outbound オプション

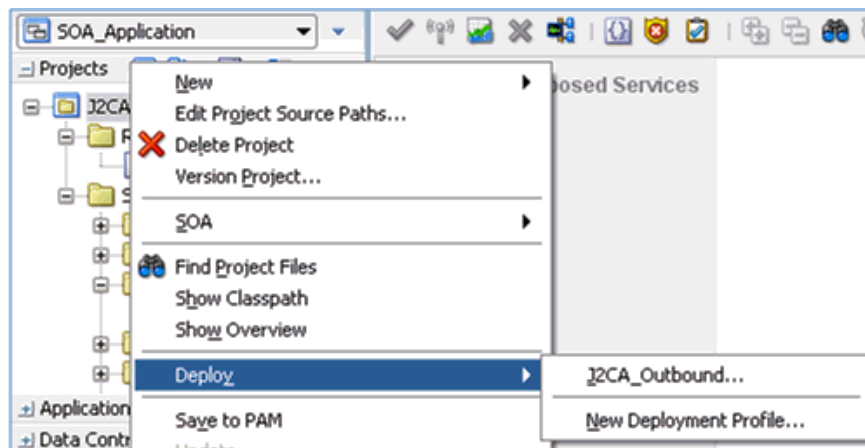
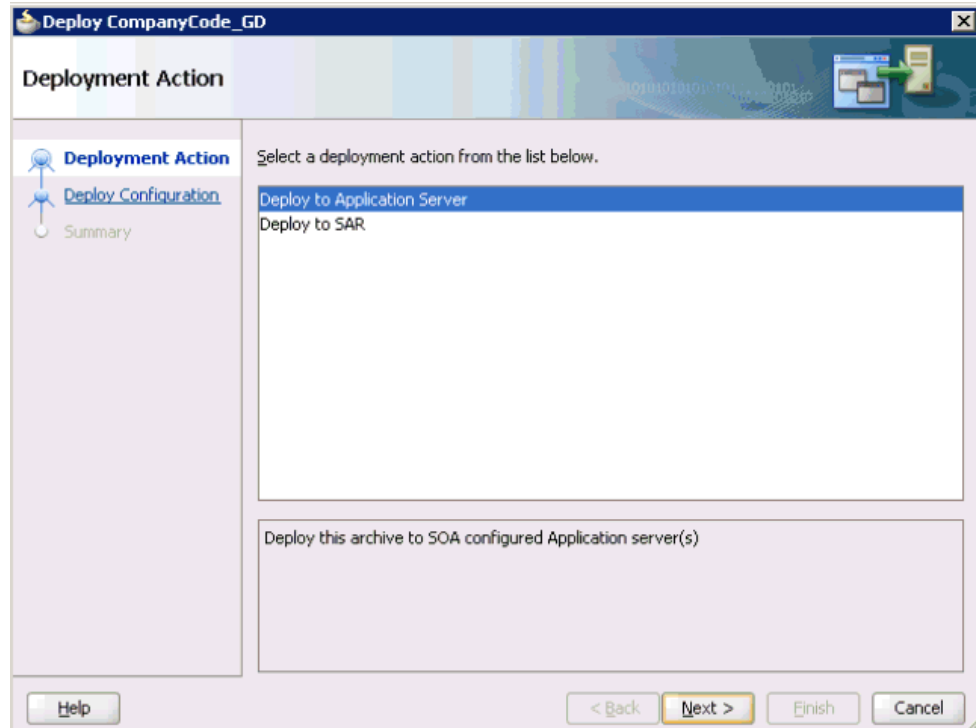


図 6-41 に示すように、「デプロイメント・アクション」ページが表示されます。

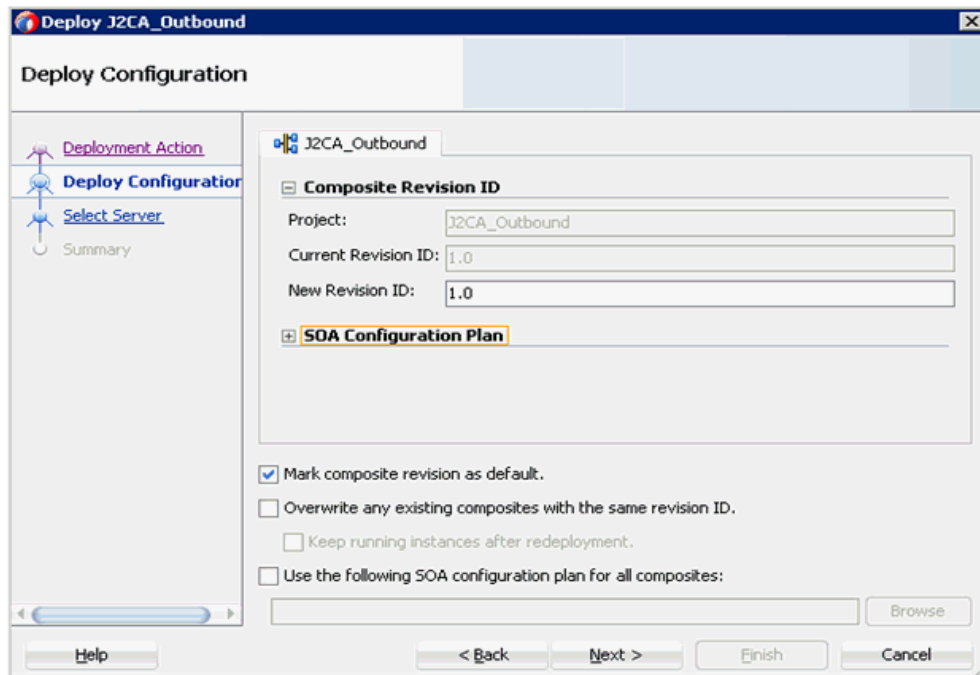
図 6-41 「デプロイメント・アクション」ページ



2. 「アプリケーション・サーバーにデプロイ」が選択されていることを確認します。

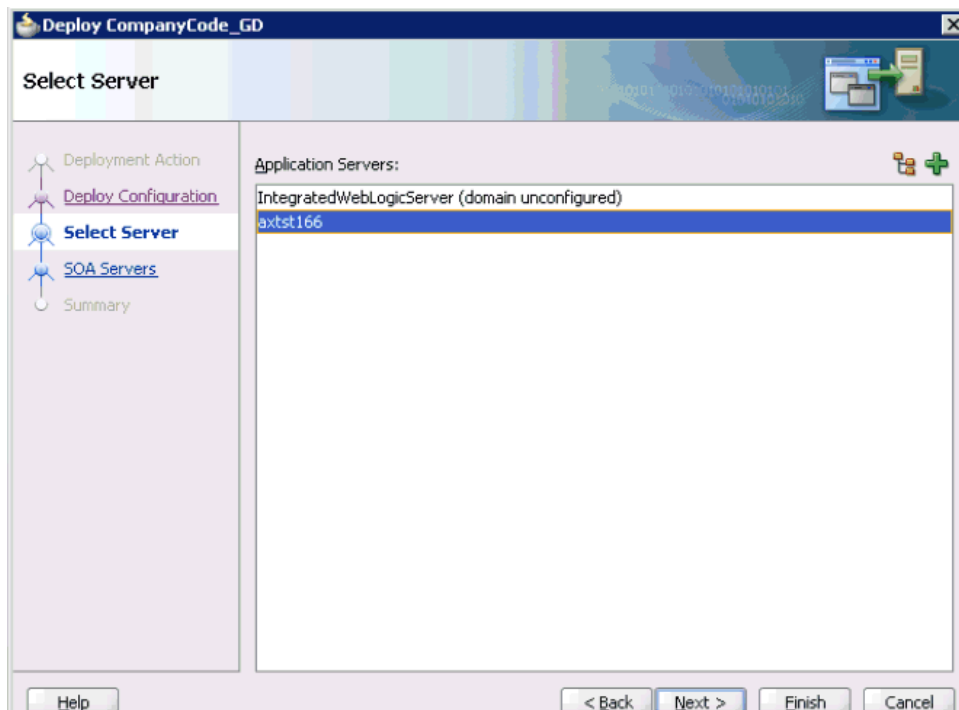
3. 「次へ」をクリックします。  
☒ 6-42 に示すように、「構成のデプロイ」ページが表示されます。

図 6-42 「構成のデプロイ」ページ



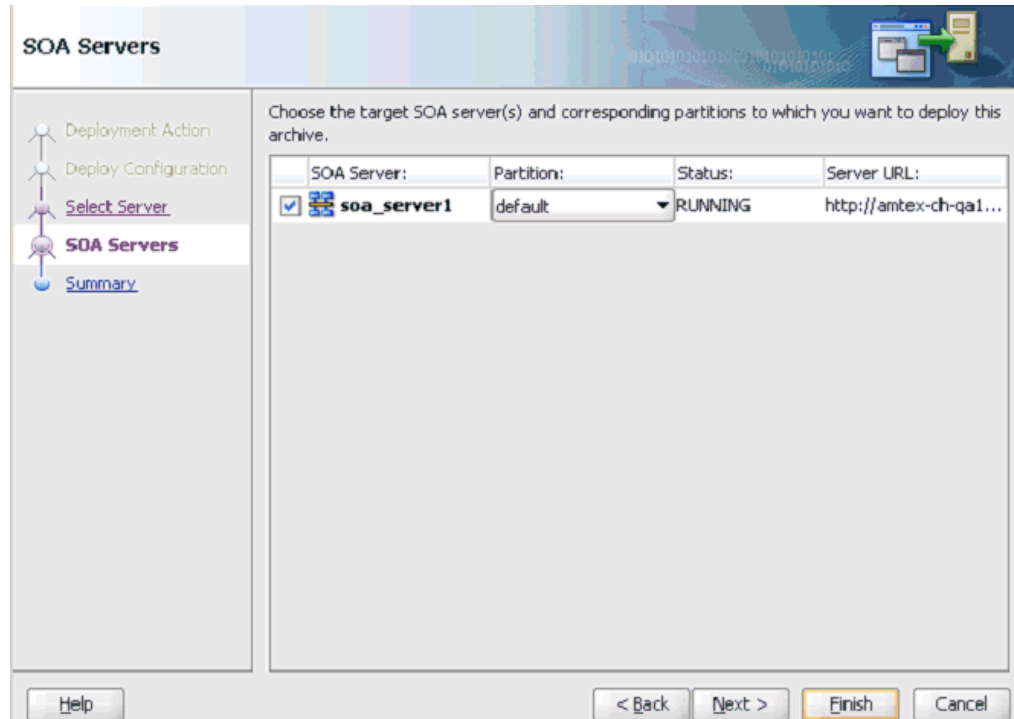
4. デフォルト値を選択済のままにして、「次へ」をクリックします。  
☒ 6-43 に示すように、「サーバーの選択」ページが表示されます。

図 6-43 「サーバーの選択」ページ



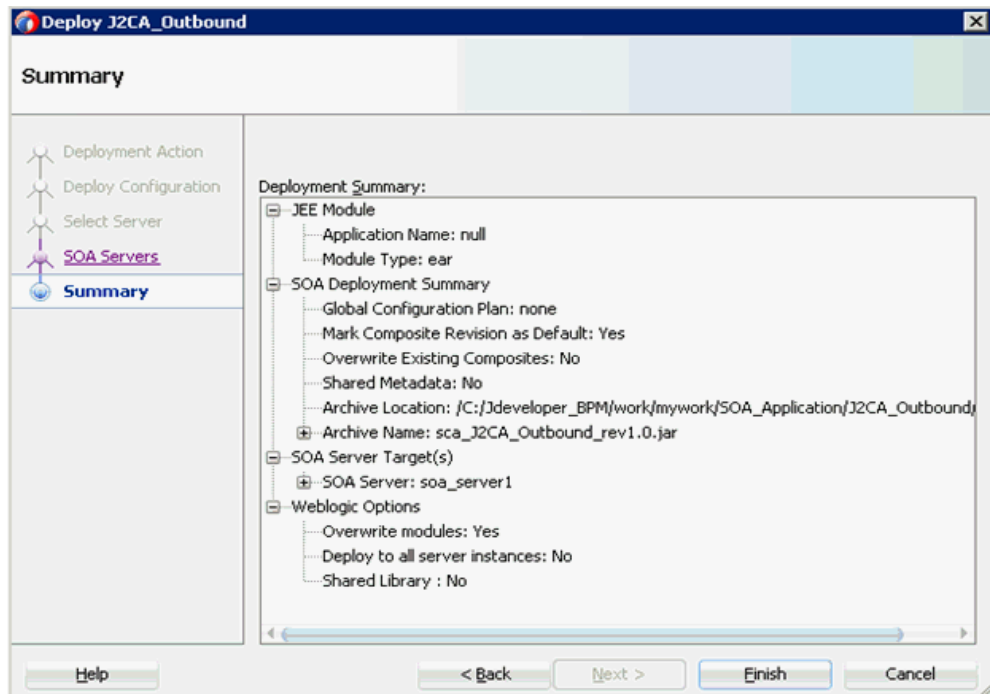
5. 構成済の使用可能なアプリケーション・サーバーを選択し、「次へ」をクリックします。  
図 6-44 に示すように、「SOA サーバー」ページが表示されます。

図 6-44 「SOA サーバー」ページ



6. ターゲット SOA サーバーを選択し、「次へ」をクリックします。  
☒ 6-45 に示すように、「サマリー」ページが表示されます。

図 6-45 「サマリー」ページ



7. プロジェクトに関する使用可能なすべてのデプロイメント情報を確認および検証し、「終了」をクリックします。  
☒ 6-46 に示すように、プロセスが正常にデプロイされます。

図 6-46 成功したデプロイメントのメッセージ



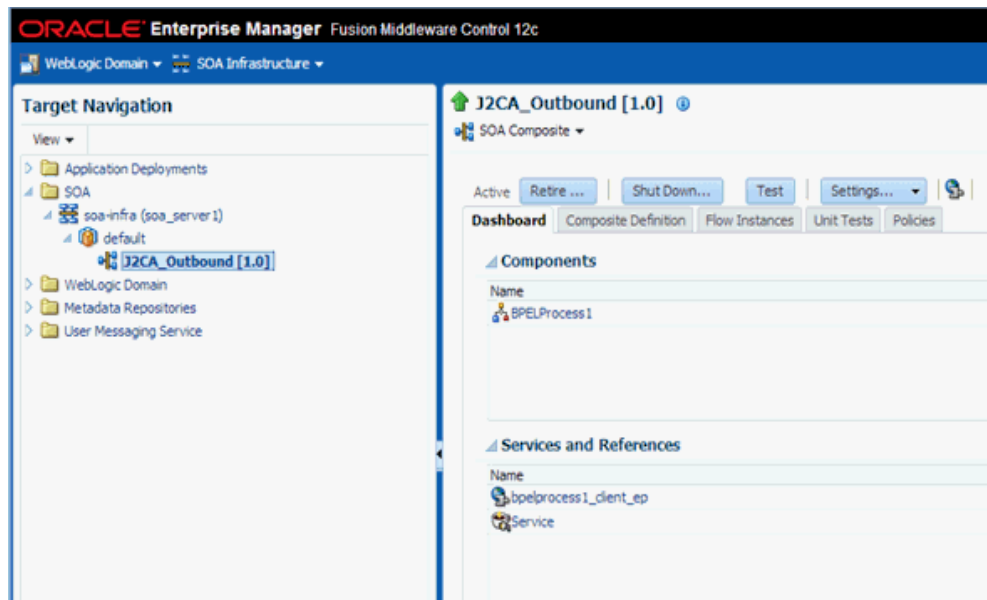


## 6.4.5 Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し

Oracle Enterprise Manager コンソールで入力 XML ドキュメントを呼び出すには、次のようにします。

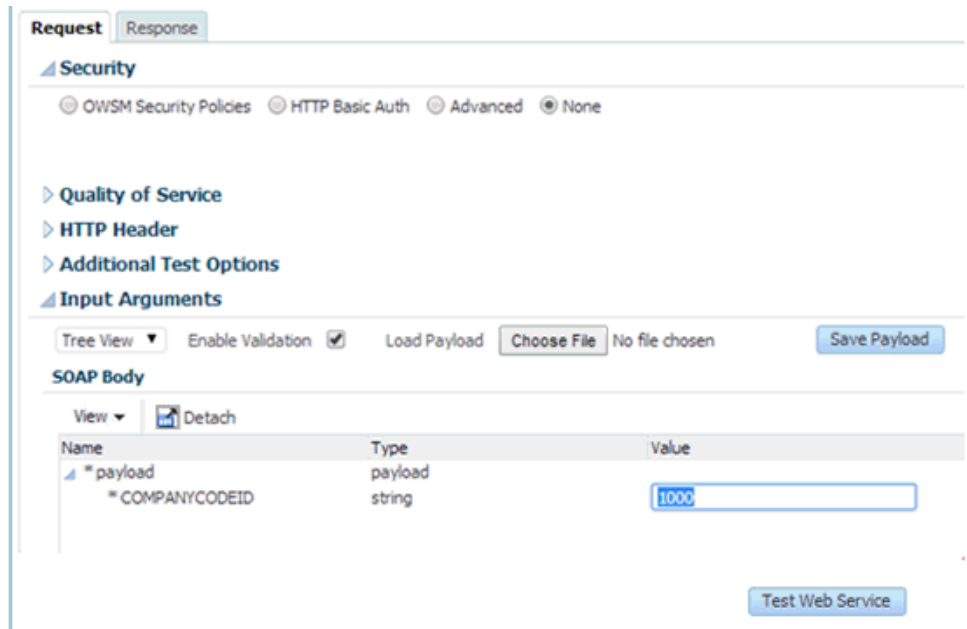
1. Oracle Enterprise Manager コンソールにログオンします。
2. 「SOA」を展開して「soa-infra (soa\_server1)」を選択し、「デフォルト」をクリックします。
3. 使用可能なプロジェクトを選択します (J2CA\_Outbound など)。

図 6-47 「テスト」ボタン



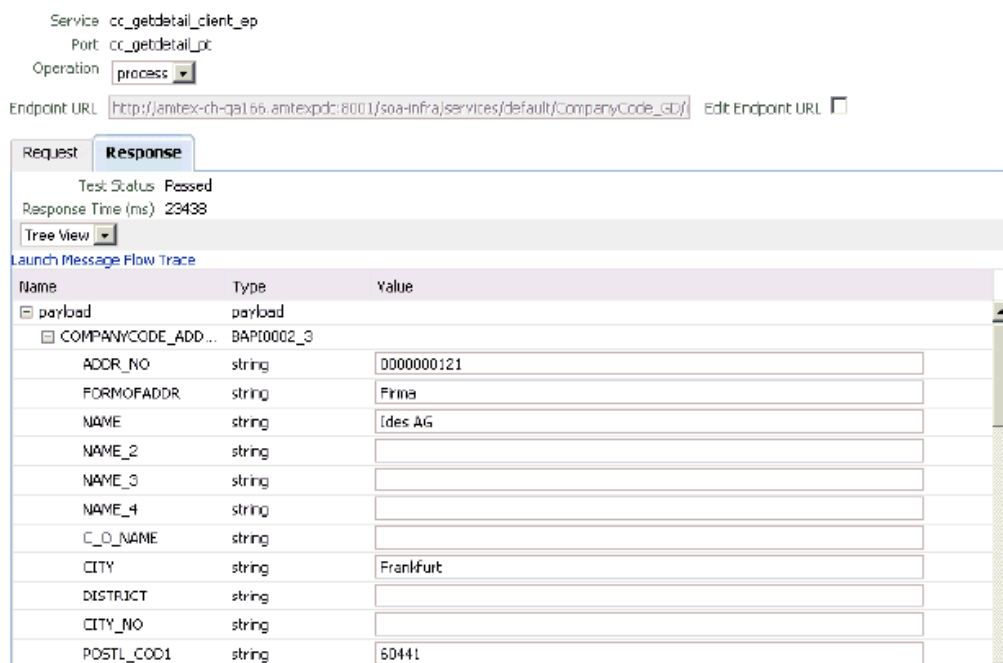
4. 図 6-47 に示すように、「テスト」をクリックします。

図 6-48 「リクエスト」タブ



5. 図 6-48 に示すように、「リクエスト」タブをクリックします。
6. 「値」フィールドに適切な入力値を指定して、「Web サービスのテスト」をクリックします。  
 図 6-49 に示すように、出力レスポンスが Oracle Enterprise Manager コンソールに受信されます。

図 6-49 出力レスポンス



「XML 表示」を使用して入力 XML ドキュメントを呼び出すには、次のようにします。

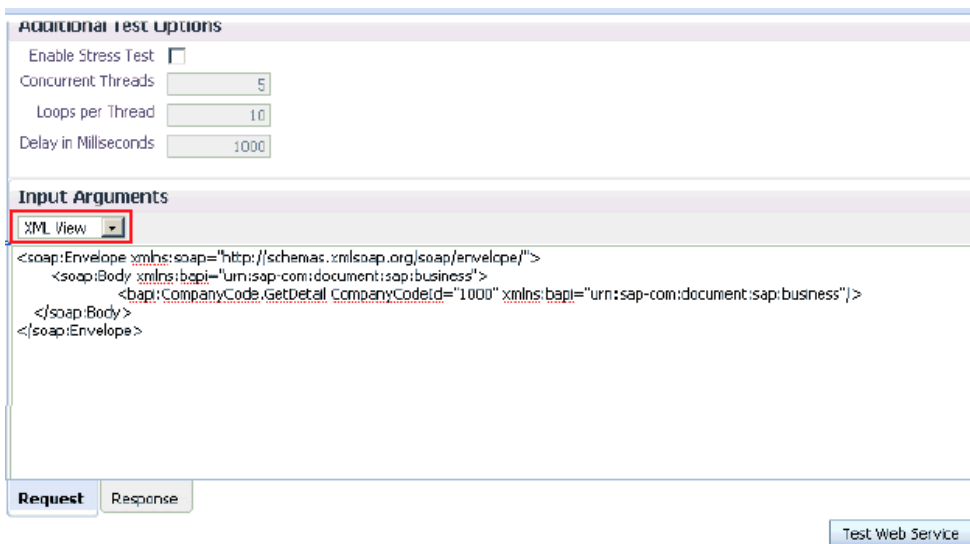
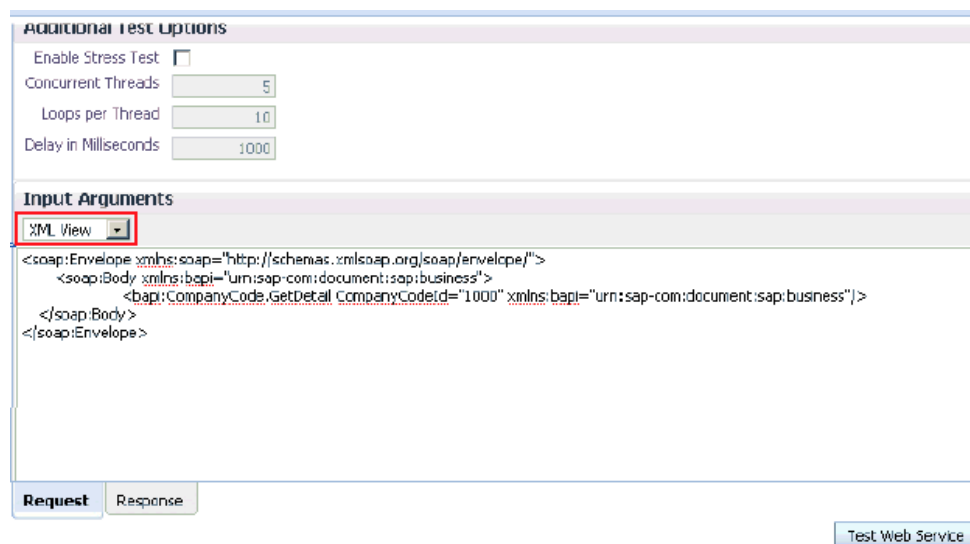
- a.  6-50 に示すように、リストで「XML 表示」を選択します。

図 6-50 「引数の入力」リスト



Additional Test Options

Enable Stress Test

Concurrent Threads

Loops per Thread

Delay in Milliseconds

Input Arguments

XML View

```
<?xml version='1.0' encoding='UTF-8'>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body xmlns:bapi="urn:sap-com:document:sap:business">
    <bapi:CompanyCode.GetDetail CompanyCodeId="1000" xmlns:bapi="urn:sap-com:document:sap:business"/>
  </soap:Body>
</soap:Envelope>
```

Request Response

Test Web Service

- b. 「引数の入力」領域で適切な入力 XML ドキュメントを指定し、「Web サービスのテスト」をクリックします。

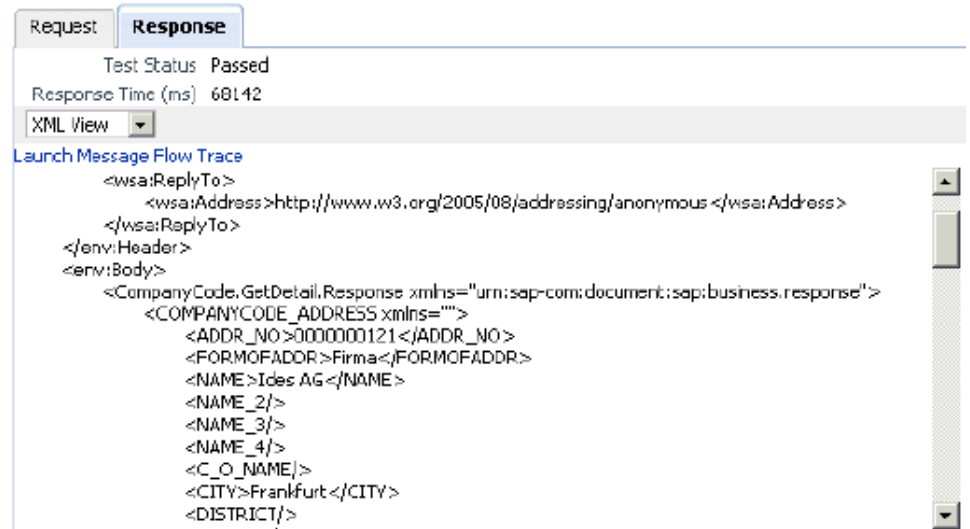
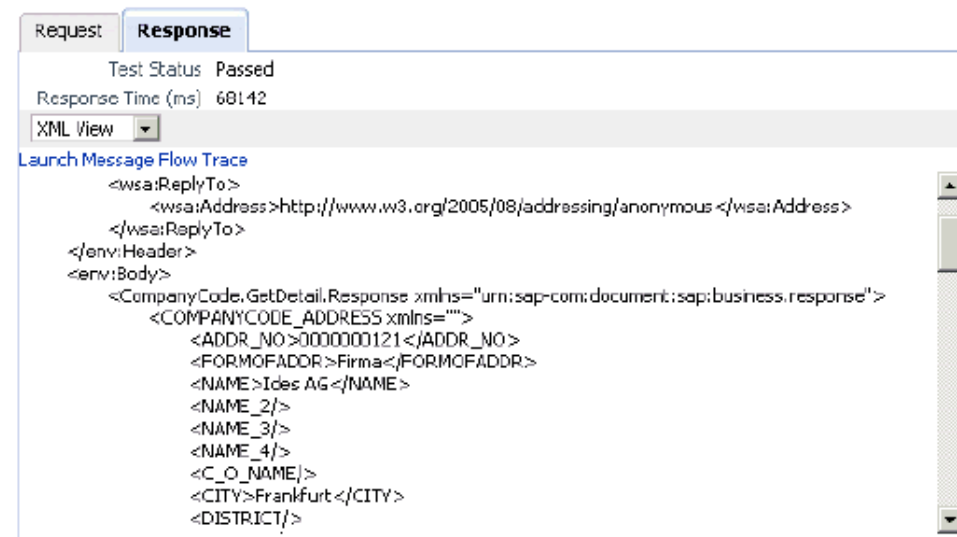
 6-51 に示すように、出力レスポンスが Oracle Enterprise Manager コンソールに受信されます。

図 6-51 受信された出力レスポンス



Request Response

Test Status Passed

Response Time (ms) 60142

XML View

Launch Message Flow Trace

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Header>
    <wsa:ReplyTo>
      <wsa:Address>http://www.w3.org/2005/08/addressing/anonymous</wsa:Address>
    </wsa:ReplyTo>
  </env:Header>
  <env:Body>
    <CompanyCode.GetDetail.Response xmlns="urn:sap-com:document:sap:business:response">
      <COMPANYCODE_ADDRESS xmlns="">
        <ADDR_NO>000000121</ADDR_NO>
        <FORMOFADDR>Firma</FORMOFADDR>
        <NAME>Ides AG</NAME>
        <NAME_2/>
        <NAME_3/>
        <NAME_4/>
        <C_O_NAME/>
        <CITY>Frankfurt</CITY>
        <DISTRICT/>
      </COMPANYCODE_ADDRESS>
    </CompanyCode.GetDetail.Response>
  </env:Body>
</env:Envelope>
```

## 6.4.6 アウトバウンド BPEL およびメディエータ・プロセスのテスト

Oracle Enterprise Manager コンソールからアウトバウンド BPEL プロセスまたはアウトバウンド・メディエータ・プロセスをテストする場合、これらのコンソールで生成される XML エンベロープは使用しないでください。これらを削除して、スキーマから生成される XML ペイロード (ネームスペースの修飾に関する WSDL 仕様に準拠) を使用します。

メディエータ・データ・フローは、Enterprise Manager コンソールを使用してテストできます。メディエータ・データ・フローと相互作用を作成すると、Web サービスが作成され、Oracle Application Server に登録されます。メディエータ・アウトバウンド・プロセスの作成の詳細は、第 7 章「Oracle SOA Suite でのメディエータ・サービス・コンポーネントとの統合」を参照してください。

## 6.5 イベント統合のためのインバウンド BPEL プロセスの設計 (J2CA 構成)

この項では、Oracle Application Adapter for SAP R/3 を SAP R/3 と統合し、イベント・データを受信する方法について説明します。この例では、顧客レコードが SAP R/3 システムに追加されると SAP R/3 イベントが発生します。

アプリケーション・アダプタ・インストールの次のフォルダに、このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\J2CA\Inbound_Project
```

アダプタ・デザインタイム構成を完了するには、次のツールが必要です。

- Oracle Adapter Application Explorer (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPEL Designer (JDeveloper)

---

**注意：** この章の例では、JDeveloper を使用しています。

---

ここでは、次の項目について説明します。

- [6.5.1 項「イベント統合のための WSDL の生成」](#)
- [6.5.2 項「SOA 用の空のコンポジットの作成」](#)
- [6.5.3 項「BPEL インバウンド・プロセスの定義」](#)
- [6.5.4 項「BPEL インバウンド・プロセスのデプロイ」](#)
- [6.5.5 項「SAP R/3 でのイベントのトリガー」](#)

### 6.5.1 イベント統合のための WSDL の生成

BPEL プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。

アプリケーション・エクスプローラを使用してインバウンド相互作用のための WSDL を生成する場合、各インバウンド J2CA サービスに対して個別のチャンネルを作成し、そのチャンネルを選択する必要があります。

---

**注意：** 2 つ以上のイベントで同じチャンネルが共有されている場合、適切な BPEL プロセスにイベント・メッセージが配信されないことがあります。

---

ここでは、次の項目について説明します。

- [6.5.1.1 項「チャンネルの作成」](#)
- [6.5.1.2 項「イベント通知のための WSDL の生成」](#)

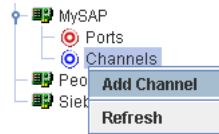
### 6.5.1.1 チャネルの作成

チャネルを作成するには、次のようにします。

1. アプリケーション・エクスプローラを起動し、J2CA 構成に接続します。
2. 左ペインで「イベント」ノードを展開します。
3. 「MySAP」ノードを展開します。

図 6-52 に示すように、左ペインに「ポート」ノードと「チャネル」ノードが表示されます。

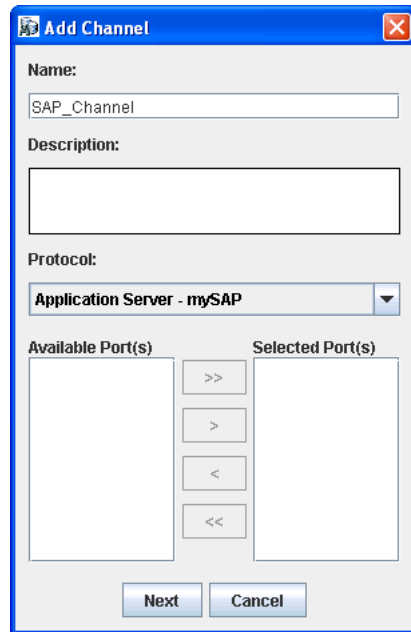
図 6-52 MySAP の「チャネル」ノード



4. 「チャネル」を右クリックして、「チャネルの追加」を選択します。

図 6-53 に示すように、「チャネルの追加」ダイアログが表示されます。

図 6-53 「チャネルの追加」ダイアログ



次のようにします。

- a. チャネルの名前を入力します (SAP\_Channel など)。
  - b. 簡単な説明を入力します (オプション)。
  - c. 「プロトコル」リストで「アプリケーション・サーバー - mySAP」を選択します。
5. 「次へ」をクリックします。

図 6-54 に示すように、「アプリケーション・サーバー」ダイアログが表示されます。使用可能なタブは次のとおりです。

- ユーザー (必須)
- システム (必須)

- セキュリティ
- 詳細

図 6-54 「アプリケーション・サーバー」ダイアログ

The screenshot shows the 'Application Server' dialog box with the 'Security' tab selected. The fields are: Client\* (empty), User\* (empty), Password\* (empty), and Authentication mode (set to Password). There are OK and Cancel buttons at the bottom. A red note at the bottom left states 'Fields marked with \* are required.'

6. 「ユーザー」タブで、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

表 6-1 「ユーザー」タブのパラメータ

ターゲット・パラメータ	説明
クライアント	クライアント通信用に SAP R/3 アプリケーションに対して定義されたクライアント番号。
ユーザー	SAP R/3 アプリケーションの有効なユーザー ID。
パスワード	SAP R/3 アプリケーションの有効なパスワード。
認証モード	SAP R/3 システムに接続する際に使用する認証モード。デフォルトでは、リストの「パスワード」が選択されています。

7. 図 6-55 に示すように、「システム」タブをクリックします。

図 6-55 「システム」タブ

The screenshot shows the 'Application Server' dialog box with the 'System' tab selected. The fields are: Gateway host\* (empty), Gateway service\* (empty), Program ID of the server\* (empty), Application Server\* (empty), System number\* (empty), Connection pool size (2), Connection timeout(min) (10), and Connection wait time(sec) (30).

8. 「システム」タブで、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

表 6-2 「システム」タブのパラメータ

ターゲット・パラメータ	説明
ゲートウェイ・ホスト	SAP R/3 ゲートウェイのホスト名。
ゲートウェイ・サービス	SAP R/3 ゲートウェイのサービス。
サーバーのプログラム ID	このチャンネルに使用する SAP R/3 プログラム ID。
アプリケーション・サーバー	SAP R/3 アプリケーション・サーバーの名前。
システム番号	SAP R/3 システム番号。

9. 「セキュリティ」タブ (オプション) で、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

表 6-3 「セキュリティ」タブのパラメータ

ターゲット・パラメータ	説明
SNC モード	デフォルトでは、SNC は無効になっています。SNC を有効にするには、リストで 1 を選択します。
SNC パートナ	SNC サービスを提供する RFC サーバーまたはメッセージ・サーバー (ロード・バランシング) の名前を入力します。
SNC レベル	リストで SNC ライブラリのバージョンを選択します。
SNC 名	使用している SNC ライブラリの名前を入力します。
SNC ライブラリ・パス	SNC ライブラリへのパスを入力します。

10. 「詳細」タブ (オプション) で、次の表の情報に基づいて、SAP R/3 チャンネルに適切な情報を入力します。

表 6-4 「詳細」タブのパラメータ

ターゲット・パラメータ	説明
IDOC 確認	このオプションが選択されている場合、SAP に STATUS IDOC が返されて、受信された各 IDOC について受信が確認されます。  標準の IDOC が送信される場合、SAP IDOC モニターを通じて WE02 トランザクション IDOC リストに表示される通常の最新ステータスは、「Data Sent to Port OK」 - 「03」です。 WE02 トランザクションを使用して IDOC がターゲット宛先で受信されたことを確認する場合は、このオプションを有効にします。IDOC が受信されると、アダプタは「STATUS IDOC」を SAP に返し、WE02 トランザクション IDOC リスト内の IDOC のステータスが「12」、「Dispatch OK」に変わります。  多数の IDOC を一度に送信する場合にこのオプションが有効になっていると、パフォーマンスに影響を及ぼす可能性があることに注意してください。パフォーマンスが問題となる場合は、このオプションを無効にすることをお勧めします。
IDOC フォーマット	リストで IDOC タイプを選択します。 <ul style="list-style-type: none"><li>■ XML (デフォルト)</li><li>■ XML-CDATA-ENVELOPED</li></ul>
IDOC リリース	接続に使用する IDOC パージョン。

表 6-4 「詳細」 タブのパラメータ (続き)

ターゲット・パラメータ	説明
IDOC リリース・プロバイダ	接続のための IDOC リリース・プロバイダ。リストで <b>IDOC DOCREL</b> フィールド (デフォルト)、 <b>SAP</b> リリースまたはユーザー入力を選択します。
SAP トレース	SAP トレースを有効にするには、このオプションを選択します。 SAP トレースは、デザインタイムおよびランタイムの間、次の場所に格納されています。 デザインタイム: <ADAPTER_HOME>\tools\iwaeb\bin  ランタイム: <ORACLE_HOME>\user_projects\domains\base_domain  SAP トレースに使用されるファイル名形式は、デザインタイムとランタイムで次のようになります。 <ul style="list-style-type: none"> <li>■ rfc02664_04332.trc</li> <li>■ JCO100226_043846994.trc</li> </ul>
処理モード	リストで同期処理のタイプを選択します。選択可能な値には、「REQUEST」、「REQUEST_RESPONSE」などがあります。

11. 「OK」 をクリックします。

チャンネルは、左ペインの「チャンネル」 ノードの下に表示されます。アイコンの上の X は、そのチャンネルが現在切断されていることを示します。

**注意：** チャンネルは BPEL PM Server により管理されているため、チャンネルを開始しないでください。テストやデバッグ目的でチャンネルを開始した場合は、ランタイム前に停止してください。

### 6.5.1.2 イベント通知のための WSDL の生成

チャンネルを作成し、そのチャンネルが開始されていないことを検証した後で、アプリケーション・エクスプローラを使用してイベント用の WSDL を生成する必要があります。

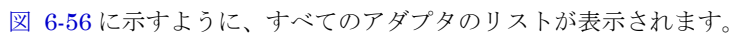
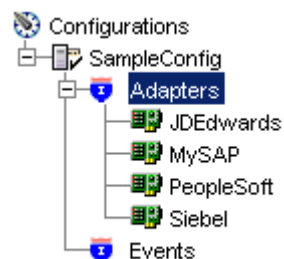
1. アプリケーション・エクスプローラを起動します。
2. 「アダプタ」 ノードを展開します。  


図 6-56 「アダプタ」 ノード





次のようにします。

- a. 「MySAP」ノードを展開します。

図 6-57 に示すように、使用可能なターゲットのリストが表示されます。

図 6-57 切断されている「MySAP Target」ノード



- b. 「MySAP」ノードで、「SAPTarget」などのターゲット名をクリックします。  
保存されているパラメータが「接続」ダイアログに表示されます。

3. 接続パラメータを検証します。
4. ターゲット名を右クリックし、「接続」を選択します。

図 6-58 に示すように、X アイコンが消えてノードが接続されたことが示されます。

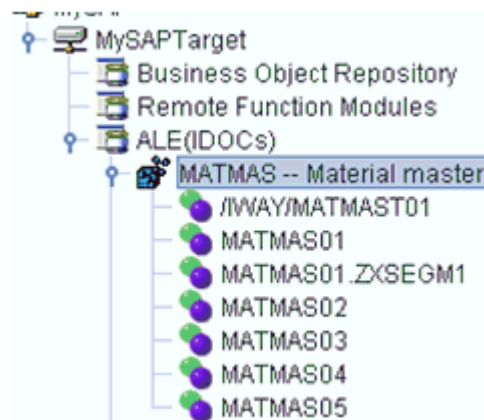
図 6-58 接続されている MySAP Target



5. 「ALE(IDOCs)」ノードを展開し、「MATMAS - Material Master」を選択します。

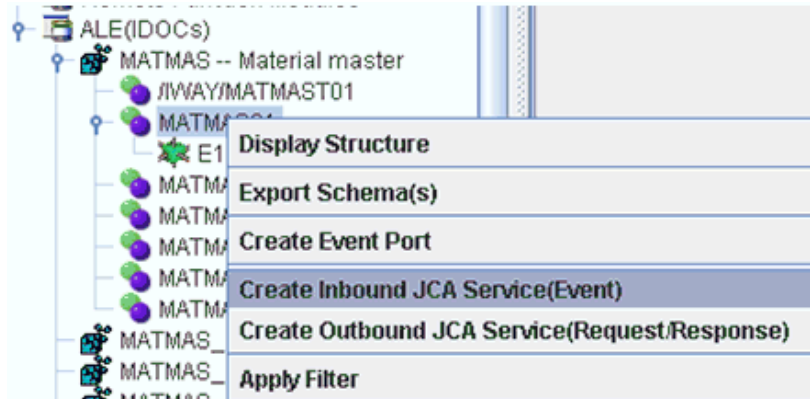
図 6-59 に示すように、「MATMAS」リストが表示されます。

図 6-59 「MATMAS Material Master」リスト



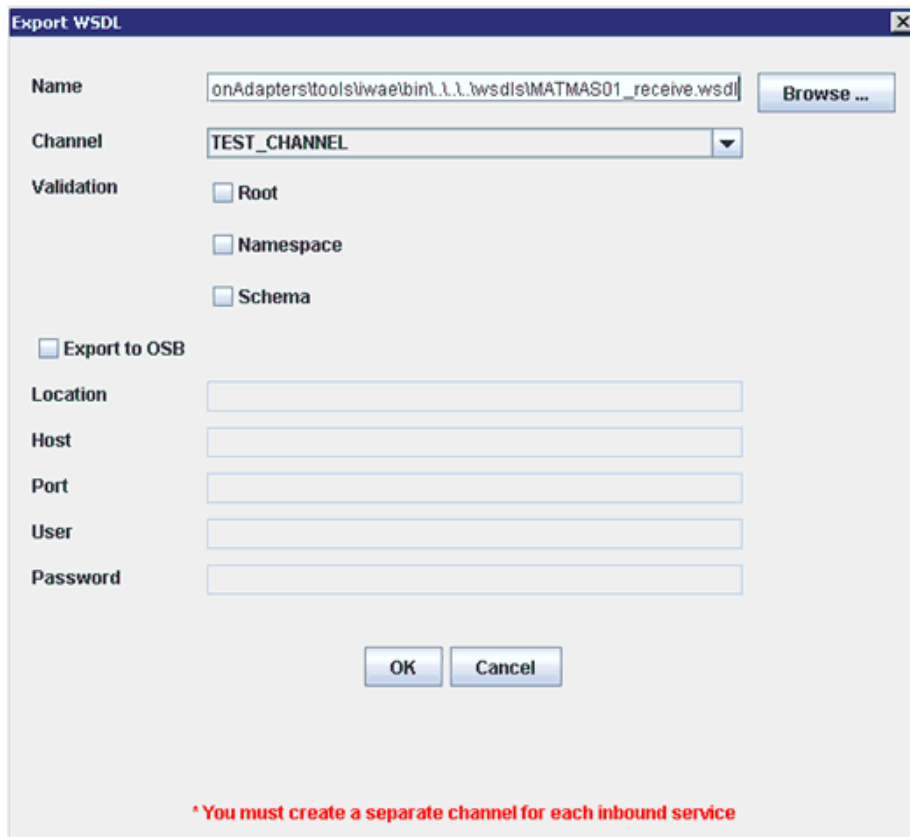
6. 「MATMAS」リストの「MATMAS01」を右クリックします。

図 6-60 インバウンド JCA サービス (イベント) の作成オプション



7.  6-60 に示すように、インバウンド JCA サービス (イベント) の作成を選択します。  
 6-61 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

図 6-61 「WSDL のエクスポート」ダイアログ



次のようにします。

- a. 「名前」フィールドで、WSDL ファイル名を指定します。
- b. 「チャンネル」リストで、このインバウンド・サービスのために作成したチャンネルを選択します。

**重要:** イベントごとに個別にチャンネルを作成する必要があります。ランタイムの前にチャンネルが停止されていることを検証します。

- c. 「ルート」、「ネームスペース」および「スキーマ」の各検証の3つのチェック・ボックスも使用可能です。複数の検証オプションを選択できます。

- ルート検証は、インバウンド XML ドキュメント内のルート要素の検証に使用しません。

- ネームスペース検証は、インバウンド XML ドキュメント内のネームスペースの検証に使用します。

- スキーマ検証は、WSDL ドキュメント内のスキーマを使用したインバウンド XML ドキュメントの検証に使用します。

ランタイム中、検証は選択された検証オプションに基づいて処理されます。複数の検証オプションが選択されている場合、ランタイム中に最初の検証オプションが失敗すると、残りの検証オプションは処理されません。ルートとネームスペースの検証は、適度な検証レベルであるとみなされます。スキーマの検証は、より厳密な検証レベルになります。ルート要素とネームスペースが SAP 環境内の IDOC 間で異なっている場合を除いて、ルートとネームスペースの検証オプションを使用することをお勧めします。

8. 「OK」をクリックします。

これで、JDeveloper で BPEL インバウンド・プロセスを定義するために必要な最初の手順として、SOA 用の空のコンポジットを作成できるようになりました。

## 6.5.2 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

1. 新規 SOA アプリケーションを作成します。
2. 新規 SOA アプリケーションの名前を入力し、「次へ」をクリックします。  
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクト名を入力し、「次へ」をクリックします。  
SOA 設定の構成ページが表示されます。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

詳細は、6-11 ページの [6.4.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

## 6.5.3 BPEL インバウンド・プロセスの定義

この項では、BPEL インバウンド・プロセスを定義する方法について、次の項目で説明します。

- [6.5.3.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの作成」](#)
- [6.5.3.2 項「インバウンド BPEL プロセス・コンポーネントの作成」](#)

### 6.5.3.1 サード・パーティ・アダプタ・サービス・コンポーネントの作成

サード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

1. 図 6-62 に示すように、「サード・パーティ・アダプタ」コンポーネントを「サービス・アダプタ」ペインから「公開されたサービス」ペインにドラッグ・アンド・ドロップします。

図 6-62 サード・パーティ・アダプタ・コンポーネント

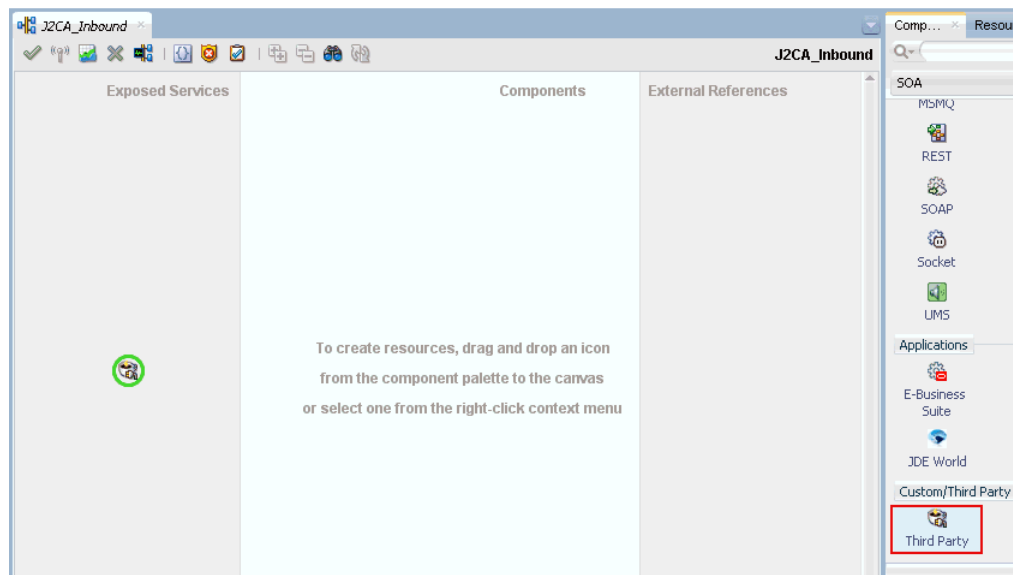
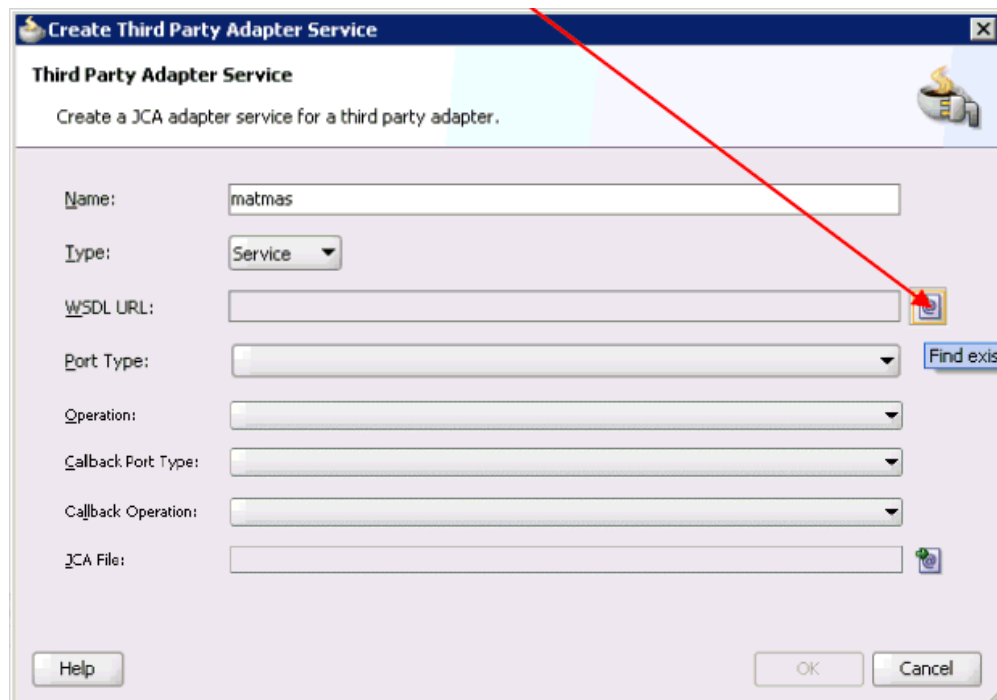


図 6-63 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

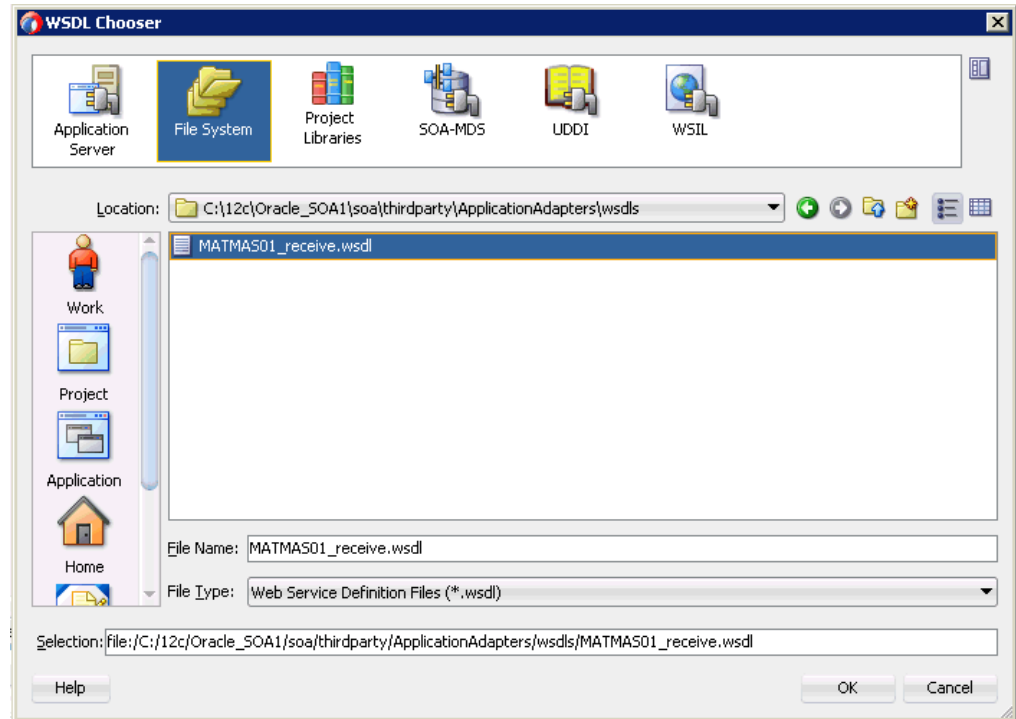
図 6-63 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



2. サード・パーティ・アダプタ・サービスの名前を入力します。

3. 「タイプ」リストで「サービス」が選択されていることを確認します (デフォルト)。
4. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。  
図 6-64 に示すように、「WSDL の選択」ダイアログが表示されます。

図 6-64 「WSDL の選択」ダイアログ

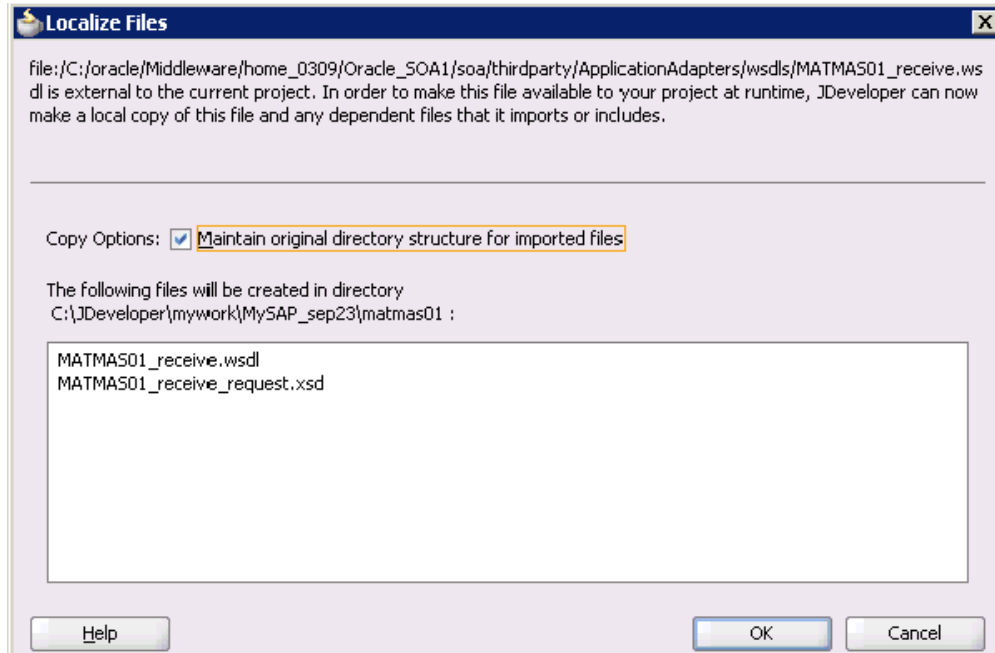


5. 次のディレクトリでインバウンド WSDL ファイルを参照して選択します。  
<ADAPTER\_HOME>\wsdl\

6. 「OK」をクリックします。

☒ 6-65 に示すように、「ファイルのローカライズ」ダイアログが表示されます。

図 6-65 「ファイルのローカライズ」ダイアログ



## 7. 「OK」をクリックします。

インバウンド WSDL ファイルと関連する受信/リクエスト XML スキーマ・ファイル (.xsd) が、作成されたプロジェクト・フォルダにインポートされます。

図 6-66 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

図 6-66 「サード・パーティ・アダプタ・サービスの作成」ダイアログ

**Create Third Party Adapter Service**

**Third Party Adapter Service**

Create a JCA adapter service for a third party adapter.

Name: matmas

Type: Service

WSDL URL: MATMAS01\_receive.wsdl

Port Type: MATMAS01PortType

Operation: MATMAS01

Callback Port Type: ----- No Callback -----

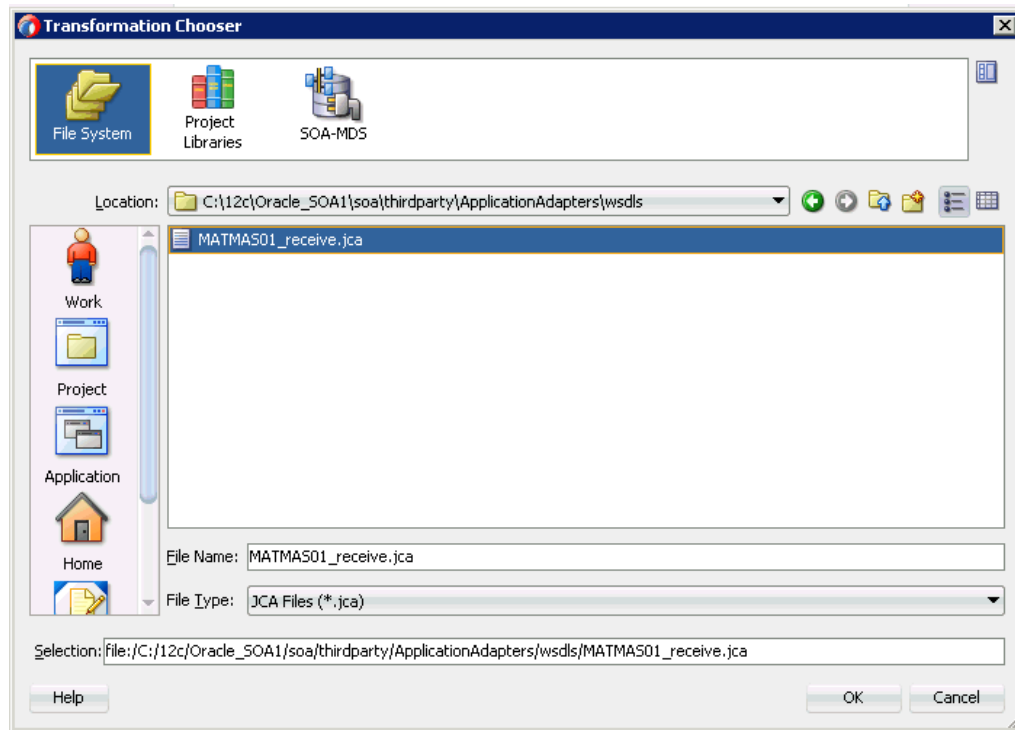
Callback Operation:

JCA File:

Help OK Cancel

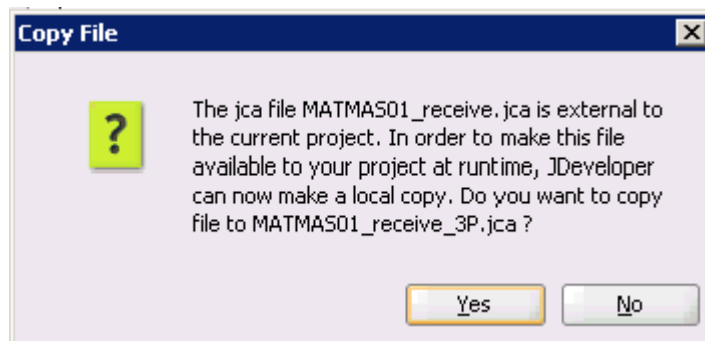
8. 「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。  
 6-67 に示すように、トランスフォーメーション・チューザ・ダイアログが表示されま  
 ず。

図 6-67 トランスフォーメーション・チューザ・ダイアログ



9. 次のディレクトリで JCA プロパティ・ファイルを参照して選択します。  
`<ADAPTER_HOME>\wsdls`
10. 「OK」をクリックします。  
 6-68 に示すように、「ファイルのコピー」メッセージが表示されます。

図 6-68 「ファイルのコピー」確認メッセージ



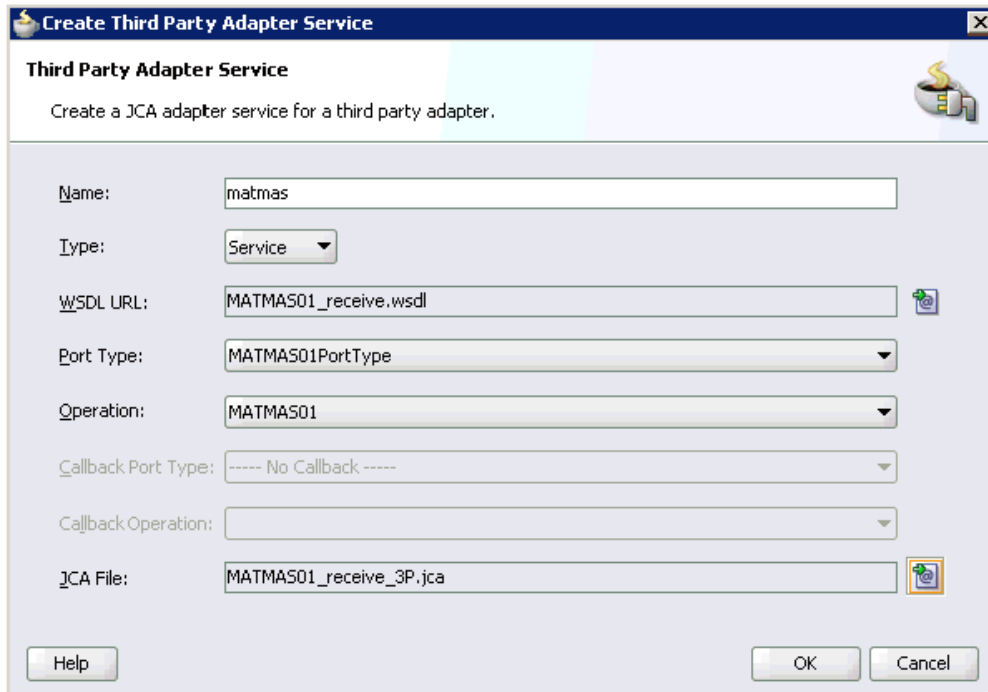


## 11. 「はい」 をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

図 6-69 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

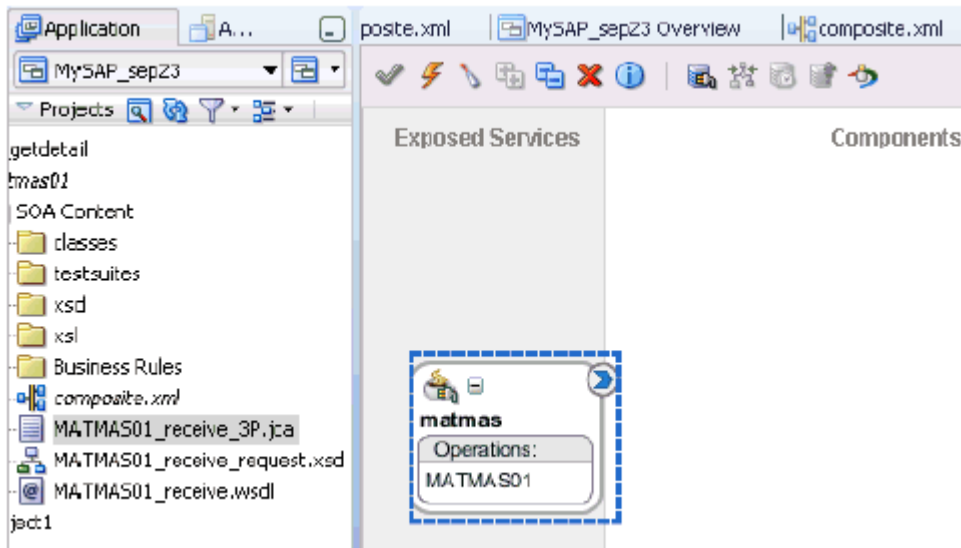
図 6-69 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



## 12. 「OK」 をクリックします。

図 6-70 に示すように、サード・パーティ・アダプタ・サービス・コンポーネント (matmas) が作成され、「公開されたサービス」ペインに表示されます。

図 6-70 サード・パーティ・アダプタ・サービス・コンポーネント



これで、インバウンド BPEL プロセス・コンポーネントを構成する準備ができました。

### 6.5.3.2 インバウンド BPEL プロセス・コンポーネントの作成

インバウンド BPEL プロセス・コンポーネントを作成するには、次のようにします。

1. 図 6-71 に示すように、「BPEL プロセス」コンポーネントを「サービス・コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

図 6-71 BPEL プロセス・コンポーネント

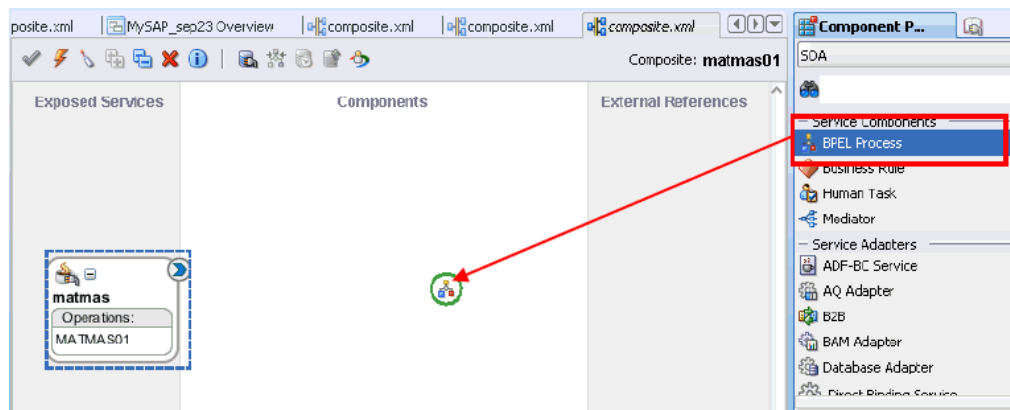
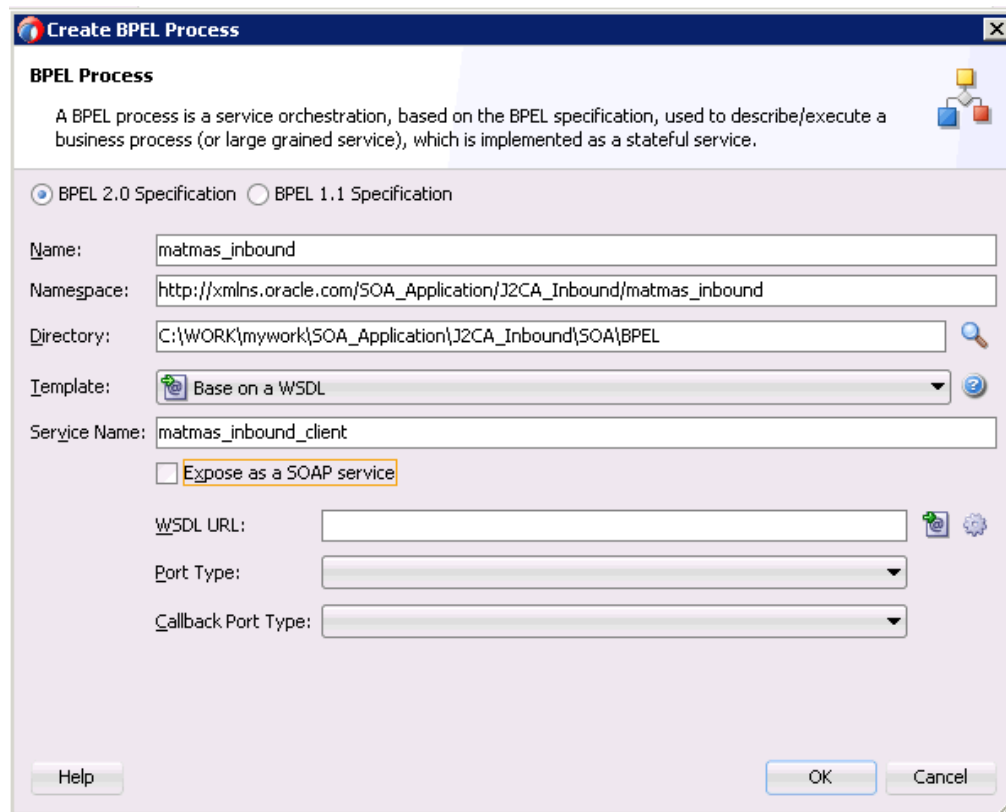


図 6-72 に示すように、「BPEL プロセスの作成」ダイアログが表示されます。

図 6-72 「BPEL プロセスの作成」ダイアログ



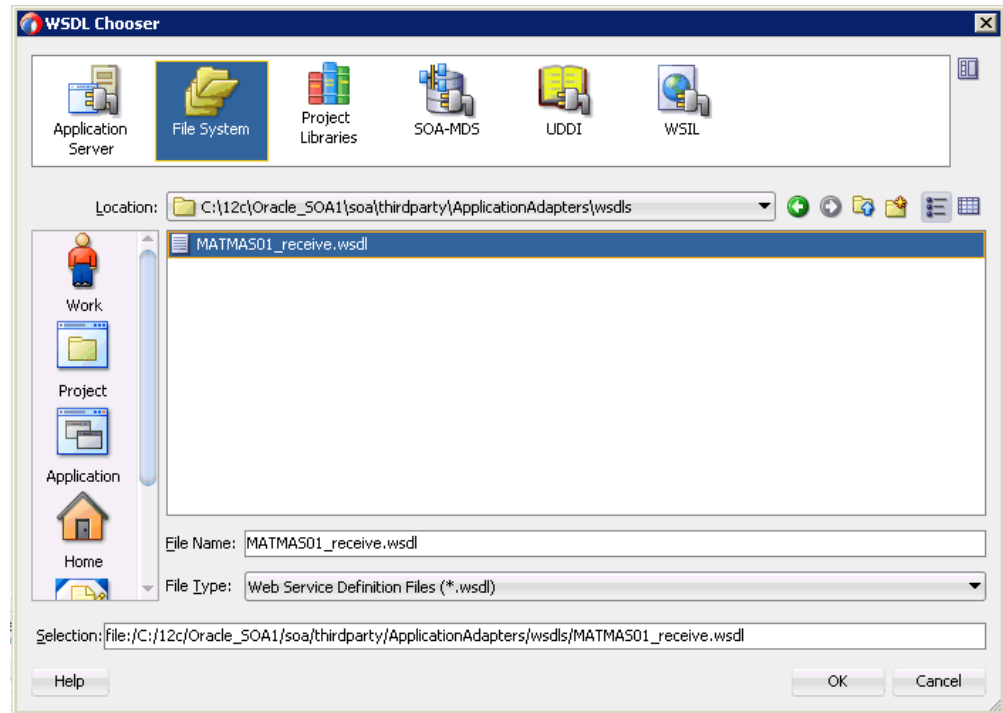
2. 「名前」フィールドに、新規インバウンド BPEL プロセス・コンポーネントを識別する名前を入力します (matmas\_inbound など)。

デフォルトでは、「BPEL 2.0 仕様」オプションが選択されています。

3. 「テンプレート」リストで「WSDLに基づく」を選択します。
4. SOAP サービスとして公開チェック・ボックスの選択を解除します。
5. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。

図 6-73 に示すように、「WSDL の選択」ダイアログが表示されます。

図 6-73 「WSDL の選択」ダイアログ



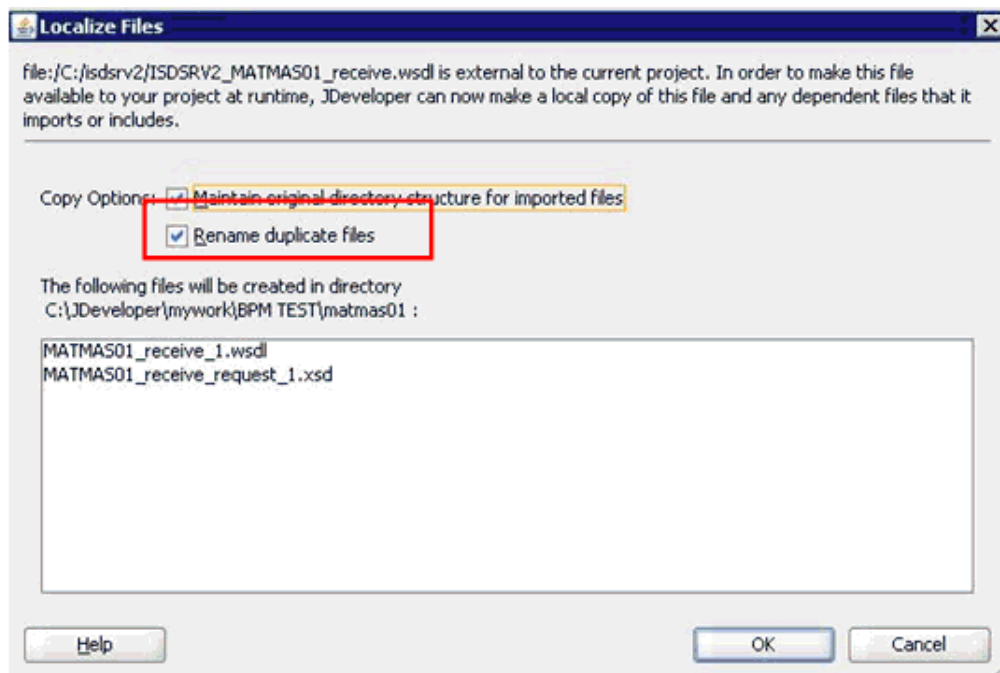
6. 次のディレクトリからインバウンド WSDL ファイルを選択します。

<ADAPTER\_HOME>\wsdls

7. 「OK」をクリックします。

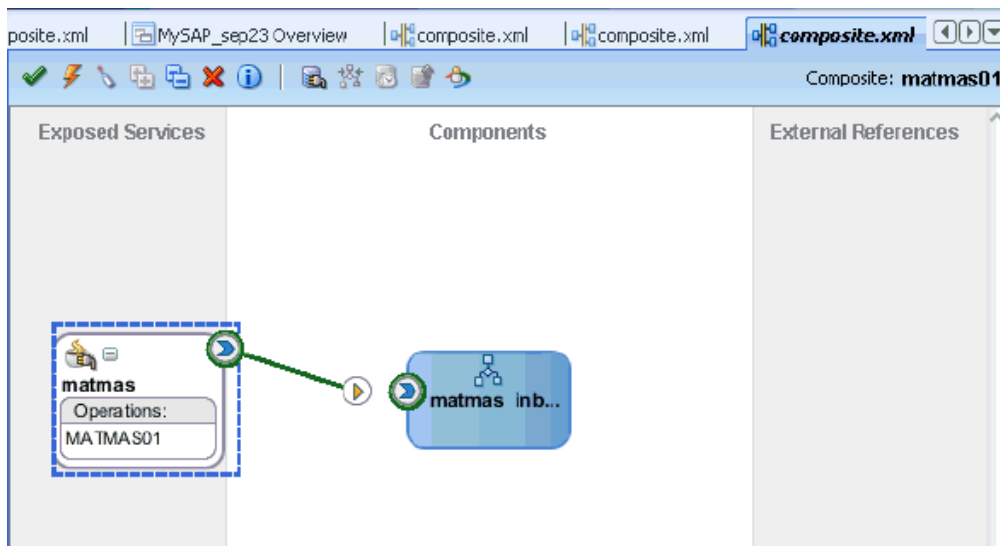
図 6-74 に示すように、「ファイルのローカライズ」ダイアログが表示されます。

図 6-74 「ファイルのローカライズ」 ダイアログ



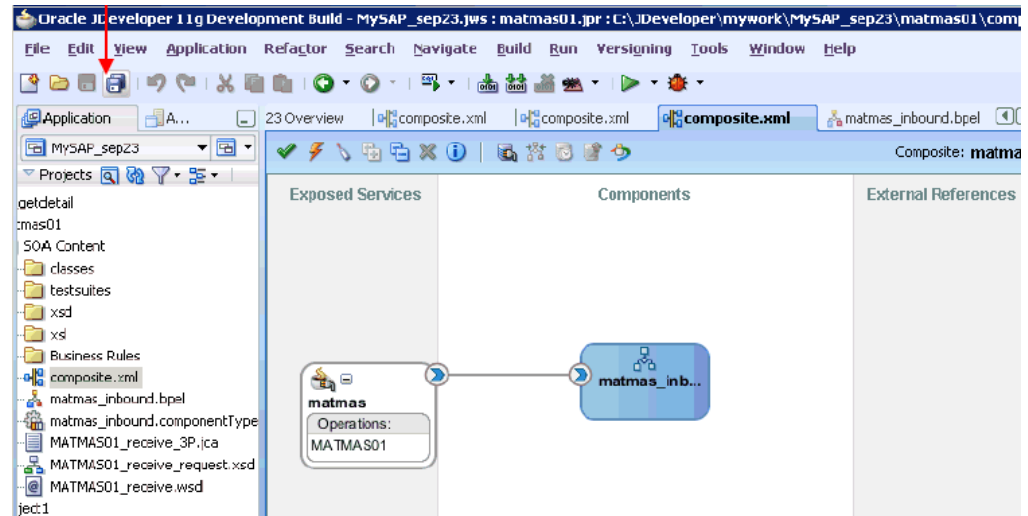
8. 「重複ファイルの名前変更」オプションの選択を解除します。
9. 「OK」をクリックします。  
「BPEL プロセスの作成」ダイアログに戻ります。
10. 「OK」をクリックします。

図 6-75 作成された接続



11. 図 6-75 に示すように、サード・パーティ・アダプタ・サービス・コンポーネント (matmas) とインバウンド BPEL プロセス・コンポーネント (matmas\_inbound) 間の接続を作成します。
12. 左ペインで「J2CA\_Inbound」をダブルクリックします。

図 6-76 「すべて保存」アイコン



13. 図 6-76 に示すように、メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規インバウンド BPEL プロセス・コンポーネントを保存します。

これで、BPEL インバウンド・プロセスをデプロイする準備ができました。

### 6.5.3.3 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの 6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」を参照してください。

## 6.5.4 BPEL インバウンド・プロセスのデプロイ

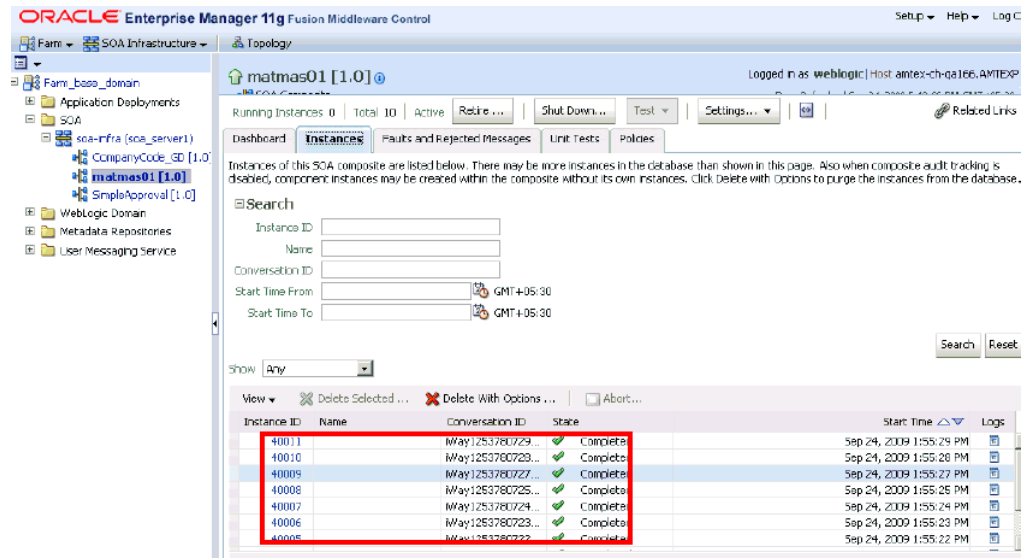
BPEL インバウンド・プロセスをデプロイするには、次のようにします。

1. 左ペインのプロジェクト名を右クリックし、「デプロイ」を選択してから「matmas01」をクリックします。  
「デプロイメント・アクション」ページが表示されます。
2. 「アプリケーション・サーバーにデプロイ」が選択されていることを確認します。
3. 「次へ」をクリックします。  
「構成のデプロイ」ページが表示されます。
4. デフォルト値を選択済のままにして、「次へ」をクリックします。  
「サーバーの選択」ページが表示されます。
5. 構成済の使用可能なアプリケーション・サーバーを選択し、「次へ」をクリックします。  
「SOA サーバー」ページが表示されます。
6. ターゲット SOA サーバーを選択し、「次へ」をクリックします。  
「サマリー」ページが表示されます。
7. プロジェクトに関する使用可能なすべてのデプロイメント情報を確認および検証し、「終了」をクリックします。  
プロセスが正常にデプロイされます。

詳細は、6-34 ページの 6.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」を参照してください。

SAP GUI を通じてイベント・メッセージがトリガーされると、[図 6-77](#) に示すように、Oracle Enterprise Manager コンソールに正常なインスタンスが受信されます。

図 6-77 受信されたインスタンス



## 6.5.5 SAP R/3 でのイベントのトリガー

イベントはアプリケーション・システムのアクティビティによって生成されます。たとえば、SAP R/3 では、システムで顧客情報が更新されるとイベントが生成される場合があります。詳細は、4-19 ページの「[イベント・アダプタの構成](#)」を参照してください。

この項では、次の項目について説明します。

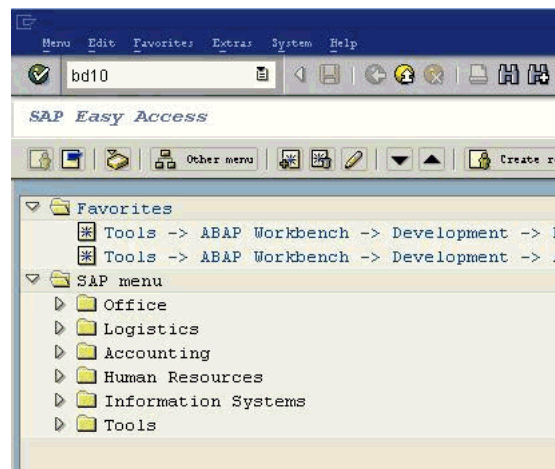
### ■ 6.5.5.1 項「結果の確認」

ここでは、SAP R/3 でイベントをトリガーする方法、および Oracle Application Adapter for SAP R/3 を使用してイベント統合を検証する方法について説明します。

SAP R/3 でイベントをトリガーするには、次のようにします。

1. [図 6-78](#) に示すように、SAP Workbench を開始して SAP R/3 システムにログインします。

図 6-78 SAP Workbench



2. **bd10** トランザクションを実行します。

図 6-79 「Send Material」 ウィンドウ

The screenshot shows the 'Send Material' dialog box. The 'Material' field is set to '1' and 'to' is '100'. The 'Class' field is empty. The 'Message type (R/3 Standard)' is 'MATMAS'. The 'Logical system' is 'UMALOG'. The 'Send material in full' checkbox is unchecked. The 'Parallel processing' section has 'Server group' empty and 'Number of materials per proces' set to '20'.

「Send Material」 ウィンドウに次の情報を入力します。

- a. 図 6-79 に示すように、「Material」フィールドに 1 から 100 の範囲の Material の番号を入力します。
  - b. 「Logical system」フィールドに、SAP R/3 で使用している論理システムを指定します。
3. 「Execute」 ボタンをクリックします。

Material マスター・データが指定された論理システムに送信されます。アプリケーション・エクスプローラのチャンネルでプログラム ID が同じ値で定義されている場合、そのチャンネルは SAP R/3 からこの Material マスター・データを受信します。

### 6.5.5.1 結果の確認

結果を確認するには、次のようにします。

1. 次の URL を使用して、Oracle Enterprise Manager コンソールにログオンします。  
<http://localhost:7001/em>
2. 左ペインで、ドメインを展開してから、「SOA」フォルダを展開します。
3. 使用可能なインバウンド BPEL プロセスを選択します。
4. 「インスタンス」タブをクリックします。  
 最近受信したランタイム・イベント・メッセージが「インスタンス」タブに表示されます。
5. インスタンス ID をクリックして、受信したイベント・メッセージを表示します。

## 6.6 BPEL を使用した同期 SAP R/3 イベントの処理 (J2CA 構成)

この項では、BPEL を使用して同期 SAP R/3 イベントを処理する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\J2CA\Synchronous_Events
```

この項では、次の項目について説明します。

- 6.6.1 項「概要」
- 6.6.2 項「アダプタ・ターゲットの構成」
- 6.6.3 項「チャンネルの構成」
- 6.6.4 項「メッセージの WSDL の生成」
- 6.6.5 項「JDeveloper での BPEL プロセスの作成」

### 6.6.1 概要

Oracle Application Adapter for SAP R/3 は、次の 2 通りの方法で SAP R/3 イベントの受信をサポートします。これらは、リモート関数呼出し (RFC) と Intermediate Document (IDoc) です。イベント処理中に、アダプタは RFC および IDoc を SAP R/3 から直接受信します。SAP メッセージ・オブジェクトは SAP R/3 イベントを定義するもので、SAP R/3 からパブリッシュされます。SAP R/3 Integration Broker は、SAP R/3 メッセージ・オブジェクトを他のサーブド・パーティ・アプリケーションにパブリッシュするためのプロセスの構成に使用されます。Oracle Application Adapter for SAP R/3 は、SAP R/3 メッセージ・オブジェクトにサブスクライブできます。

Oracle Application Adapter for SAP R/3 は、非同期または同期のいずれかのイベントをサポートします。同期イベント・モデルでは、エンタープライズ・アプリケーションはイベントごとに出力をパブリッシュし、メッセージの公開に対するレスポンスを受信するまで待機します。たとえば、ユーザーがバンキング・アプリケーションに顧客を追加すると、バンキング・アプリケーションからリクエストが送信されてイベントが開始され、信用調査アプリケーションによりその顧客のクレジット・スコアが検査されます。信用調査アプリケーションはその顧客のクレジット・スコアを返します。バンキング・アプリケーションはクレジット・スコアの詳細を受信し、バンキング・アプリケーション内の顧客レコードを更新します。これらのやりとり全体が単一のトランザクションとなります。

Oracle Application Adapter for SAP R/3 は、RFC および IDoc を介したインバウンド・トランザクション、および SAP R/3 との同期イベントをサポートします。

SAP R/3 との同期イベントでは、SAP R/3 がイベントをパブリッシュする場合、SAP R/3 はクライアントになります。Oracle Application Adapter for SAP R/3 はサーバーとして機能し、リクエストを受信すると他のアプリケーションを起動して処理を実行し、SAP にレスポンスを返します。その時点までは、SAP R/3 アプリケーションはアダプタからのレスポンスを受信するために待機します。SAP R/3 の RFC オブジェクトは同期イベントをサポートしますが、IDoc は同期イベントをサポートしません。

このユースケース・シナリオでは、Oracle Application Adapter for SAP R/3 が同期的に機能する仕組みについて説明します。ここでは 2 つの SAP R/3 システムを構成し、これらのシステムが同期イベントを使用して相互に通信する方法を示します。一方の SAP R/3 システムはソース・システムとして機能し、イベントを開始します。もう一方の SAP R/3 システムはターゲット・システムとして機能し、ここから詳細が抽出されます。このユースケース・シナリオは他の EIS でも使用できます。ただし、ソース・システムは SAP R/3 システムである必要があります。たとえば同期イベントの開始は、SAP R/3 システムからのみ実行可能です。ターゲット・システムは SAP 以外のシステム (Siebel, PeopleSoft, J.D. Edwards など) でもかまいません。

---

**注意：** 同期イベントは、返信スキーマを含むオブジェクトについてのみサポートされており、IDoc についてはサポートされていません。

---

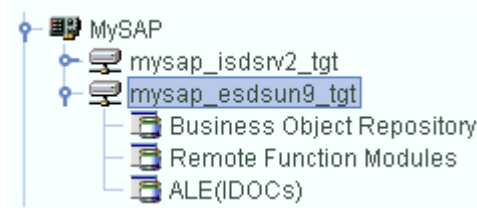


## 6.6.2 アダプタ・ターゲットの構成

次の項では、アダプタ・ターゲットを構成する方法について説明します。

1. 図 6-80 に示すように、アプリケーション・エクスプローラを開き、J2CA 構成を使用して SAP R/3 の 2 つのターゲットを作成します。

図 6-80 SAP ターゲット



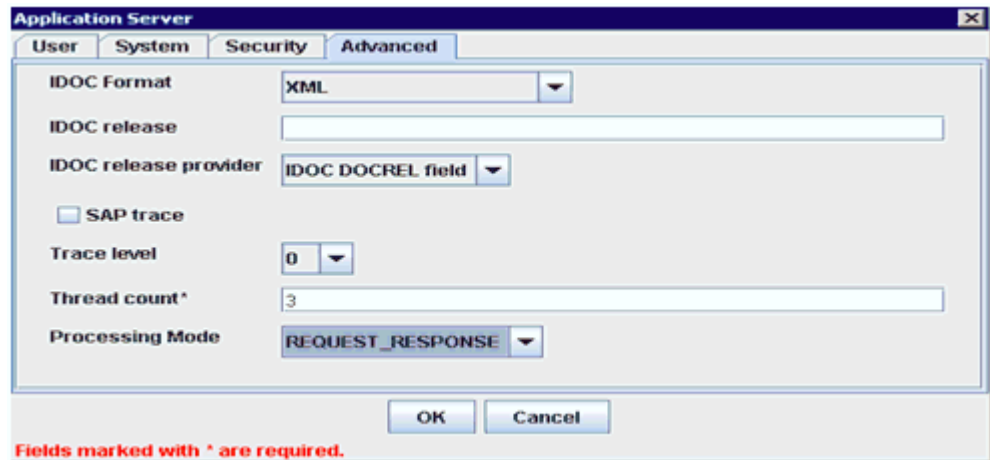
アプリケーション・エクスプローラの起動、J2CA 構成の作成および SAP R/3 へのターゲットの定義の詳細は、第 4 章「構成 Oracle Application Adapter for SAP R/3」を参照してください。

## 6.6.3 チャンネルの構成

この項では、チャンネルを構成する方法について説明します。

1. 図 6-81 に示すように、チャンネルを作成し、「詳細」タブの「処理モード」リストで「REQUEST\_RESPONSE」が選択されていることを確認します。

図 6-81 「アプリケーション・サーバー」ダイアログ



チャンネル作成の詳細は、4.9 項「イベント・アダプタの構成」を参照してください。

作成されたチャンネルは、アプリケーション・エクスプローラに表示されます。チャンネルは、左ペインの「チャンネル」ノードの下に表示されます。図 6-82 に示すように、アイコン上の X は、そのチャンネルが現在切断されていることを示しています。

図 6-82 新規チャンネル



**注意：**チャンネルは BPEL PM Server により管理されているため、チャンネルを開始しないでください。テストやデバッグ目的でチャンネルを開始した場合は、ランタイム前に停止してください。

## 6.6.4 メッセージの WSDL の生成

メッセージの WSDL を生成するには、次のようにします。

1. WSDL を生成する前に、チャンネルが開始されていないことを確認します。
2. 最初の SAP アダプタ・ターゲット (target1) に接続します。
3. BAPI\_COMPANYCODE GETLIST のインバウンド WSDL を生成します (BAPI\_COMPANYCODE GETLIST は、「リモート関数モジュール」→「Financial Accounting」→「0002 - Company Code Business Object」を使用してナビゲート)。
4. 2 番目の SAP アダプタ・ターゲット (target2) に接続します。
5. BAPI\_COMPANYCODE GETLIST のアウトバウンド WSDL を生成します (BAPI\_COMPANYCODE GETLIST は、「リモート関数モジュール」→「Financial Accounting」→「0002 - Company Code Business Object」を使用してナビゲート)。

詳細は、6.4.1 項「リクエスト/レスポンス・サービスの WSDL の生成」および 6.5.1 項「イベント統合のための WSDL の生成」を参照してください。

## 6.6.5 JDeveloper での BPEL プロセスの作成

この項では、JDeveloper で BPEL プロセスを作成する方法について説明します。

### 6.6.5.1 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

1. JDeveloper で、SOA アプリケーションを選択し、新規プロジェクトを作成します。
2. 「新規ギャラリ」ウィンドウで、「SOA プロジェクト」を選択して「OK」をクリックします。
3. プロジェクト名を入力し、「次へ」をクリックします。
4. テンプレートとして「空のコンポジット」を選択し、「終了」をクリックします。

### 6.6.5.2 インバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスの作成

インバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスを作成するには、次のようにします。

1. 「サード・パーティ・アダプタ」コンポーネントを「カスタム/サード・パーティ」領域から「公開されたサービス」ペインにドラッグ・アンド・ドロップします。

「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

2. 「名前」フィールドに名前を入力し、「WSDL URL」フィールドの横にあるアイコンをクリックします。

「WSDL の選択」ダイアログが表示されます。

3. 「ファイルシステム」タブを選択し、デフォルトの WSDL の場所で作成済のインバウンド WSDL を参照して選択し、「はい」をクリックします。

インバウンド WSDL と、リクエストおよびレスポンス XML スキーマ・ファイル (.xsd) が、作成したプロジェクト・フォルダにインポートされます。

4. 「JCA ファイルを検索します。」アイコンをクリックして、特定のインバウンド JCA プロパティ・ファイルを選択します。

5. WSDL の場所から JCA プロパティ・ファイルを選択し、「OK」をクリックします。

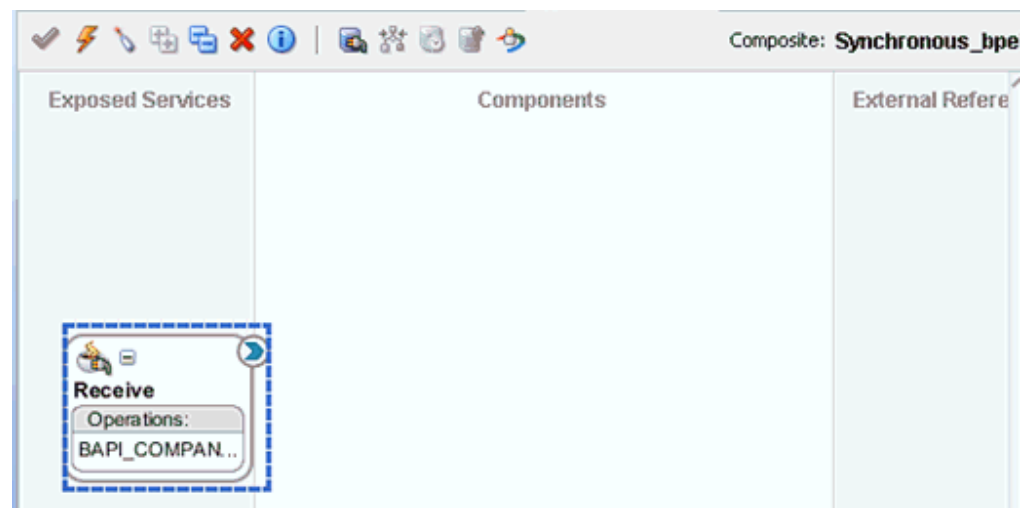
6. 「はい」をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

7. 「OK」をクリックします。

図 6-83 に示すように、「公開されたサービス」ペインにインバウンド WSDL のサード・パーティ・アダプタ・サービスが作成されます。

図 6-83 「公開されたサービス」ペイン



### 6.6.5.3 アウトバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスの作成

アウトバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスを作成するには、次のようにします。

1. 「サード・パーティ・アダプタ」コンポーネントを「カスタム / サード・パーティ」領域から「外部参照」ペインにドラッグ・アンド・ドロップします。

「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

2. 「名前」フィールドに名前を入力し、「WSDL URL」フィールドの横にあるアイコンをクリックします。

「WSDL の選択」ダイアログが表示されます。

3. 「ファイルシステム」タブを選択し、デフォルトの WSDL の場所で作成済のアウトバウンド WSDL を参照して選択し、「はい」をクリックします。

アウトバウンド WSDL と、リクエストおよびレスポンス XML スキーマ・ファイル (.xsd) が、作成したプロジェクト・フォルダにインポートされます。

4. 「JCA ファイルを検索します。」アイコンをクリックして、特定のアウトバウンド JCA プロパティ・ファイルを選択します。

5. WSDL の場所から JCA プロパティ・ファイルを選択し、「OK」をクリックします。

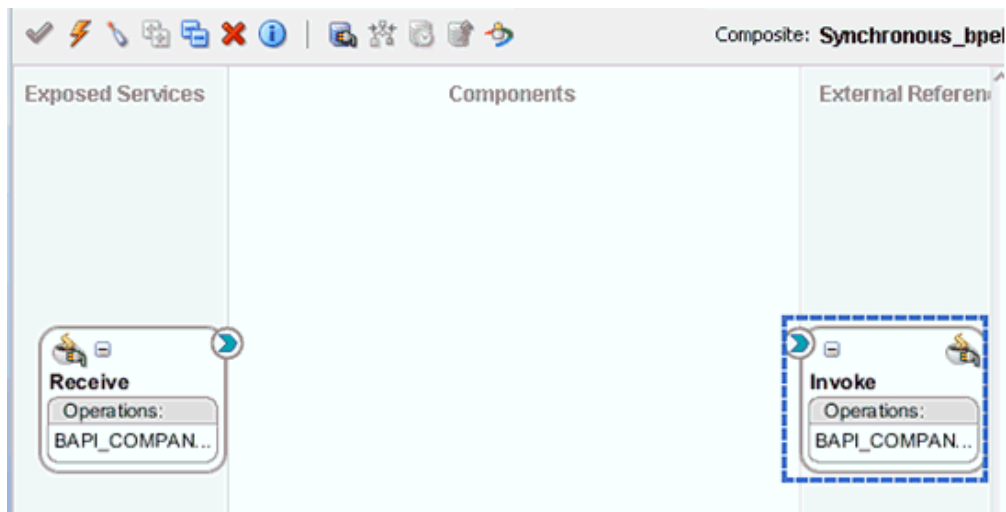
6. 「はい」をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

7. 「OK」をクリックします。

図 6-84 に示すように、「外部参照」ペインにアウトバウンド WSDL のサード・パーティ・アダプタ・サービスが作成されます。

図 6-84 「外部参照」ペイン



### 6.6.5.4 同期 BPEL プロセスの作成

同期 BPEL プロセスを作成するには、次のようにします。

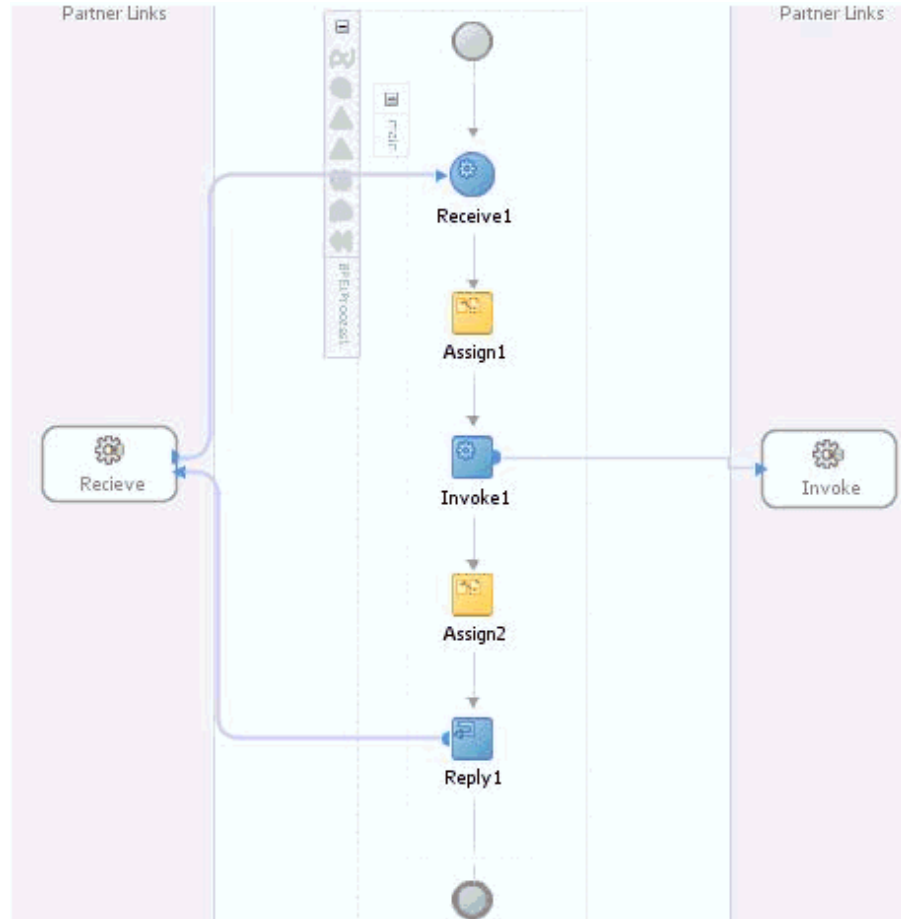
1. 「**BPEL プロセス**」コンポーネントを「コンポーネント」領域から「コンポーネント」ペインにドラッグ・アンド・ドロップします。  
「BPEL プロセスの作成」ウィンドウが表示されます。
2. 「名前」フィールドに BPEL プロセスの名前を入力し、「テンプレート」リストで「**サービスを後で定義**」を選択します。
3. 「**OK**」をクリックします。  
BPEL プロセス・コンポーネントが「コンポーネント」ペインに作成されます。
4. サード・パーティ・アダプタ・サービス「**受信**」を BPEL プロセス・コンポーネント「**Synchronous\_Events**」に接続します。
5. 「**OK**」をクリックします。
6. BPEL プロセス・コンポーネント「**Synchronous\_Events**」をサード・パーティ・アダプタ・サービス・コンポーネント「**起動**」に接続します。
7. 「コンポーネント」ペインで、BPEL プロセス・コンポーネントをダブルクリックします。
8. 「**受信**」アクティビティをビジュアル・エディタにドラッグ・アンド・ドロップして、**ここにアクティビティをドロップ**というラベルの付いた指定済のプレースホルダ内に配置します。
9. 「**受信**」アクティビティをサード・パーティ・アダプタ・サービス「**受信**」に接続します。  
「受信の編集」ダイアログが表示されます。
10. 「変数」フィールドの右側にある**プラス・アイコン**をクリックします。  
「変数の作成」ダイアログが表示されます。
11. 「**OK**」をクリックします。  
「受信の編集」ダイアログに戻ります。
12. 「**インスタンスの作成**」オプションを選択します。
13. 「**適用**」をクリックしてから、「**OK**」をクリックします。
14. 「**割当て**」アクティビティをビジュアル・エディタにドラッグ・アンド・ドロップして、「**Receive1**」アイコンの下に配置します。
15. 「**起動**」アクティビティをビジュアル・エディタにドラッグ・アンド・ドロップして、「**Assign1**」アイコンの下に配置します。
16. 「**起動**」アクティビティをサード・パーティ・アダプタ・サービス「**起動**」に接続します。  
「起動の編集」ダイアログが表示されます。
17. 「入力」タブで、「**入力**」フィールドの右側にある**プラス・アイコン**をクリックします。  
「変数の作成」ダイアログが表示されます。
18. 「**OK**」をクリックします。  
「起動の編集」ダイアログに戻ります。
19. 「出力」タブを選択し、「**出力**」フィールドの右側にある**プラス・アイコン**をクリックします。  
「変数の作成」ダイアログが表示されます。
20. 「**OK**」をクリックします。  
「起動の編集」ダイアログに戻ります。
21. 「**適用**」をクリックしてから、「**OK**」をクリックします。

22. 「割当て」 アクティビティをビジュアル・エディタにドラッグ・アンド・ドロップして、「**Invoke1**」 アイコンの下に配置します。
23. 「返信」 アクティビティをビジュアル・エディタにドラッグ・アンド・ドロップして、「**Assign2**」 アイコンの下に配置します。
24. 「返信」 アクティビティをサード・パーティ・アダプタ・サービス「**受信**」に接続します。  
「返信の編集」 ダイアログが表示されます。
25. 「変数」 フィールドで**プラス**・アイコンをクリックします。  
「変数の作成」 ダイアログが表示されます。
26. 「**OK**」 をクリックします。  
「返信の編集」 ダイアログに戻ります。
27. 「**適用**」 をクリックしてから、「**OK**」 をクリックします。
28. 「**Assign1**」 アイコンをダブルクリックして編集モードで開きます。
29. この「割当て」 アクティビティでは、「受信」 アクティビティの入力変数を「起動」 アクティビティの入力変数に割り当てる必要があります。
30. 「**適用**」 をクリックしてから、「**OK**」 をクリックします。  
ビジュアル・エディタに戻ります。
31. 「**Assign2**」 アクティビティをダブルクリックして編集モードで開きます。
32. この「割当て」 アクティビティでは、「起動」 アクティビティの出力変数を「返信」 アクティビティの出力変数に割り当てる必要があります。

33. 「適用」をクリックしてから、「OK」をクリックします。

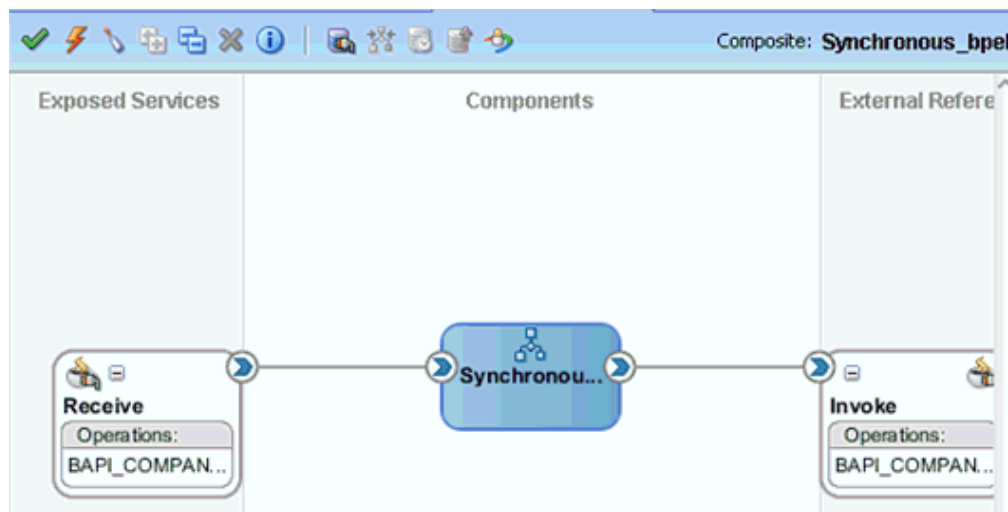
ビジュアル・エディタに戻ります。図 6-85 に示すように、同期イベントの BPEL プロセスが完成しました。

図 6-85 ビジュアル・エディタ



34. 図 6-86 に示すように、「ファイル」メニューからプロセスをすべて保存します。

図 6-86 完成した BPEL プロセス



#### 6.6.5.5 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの 6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」を参照してください。

#### 6.6.5.6 BPEL プロセスのデプロイ

BPEL プロセスをデプロイする方法の詳細は、6-34 ページの 6.4.4 項「BPEL アウトバウン  
ド・プロセスのデプロイ」を参照してください。

## 6.7 サービス統合のためのアウトバウンド BPEL プロセスの設計 (BSE 構成)

この項では、サービス統合のためのアウトバウンド BPEL プロセスを設計する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユース  
ケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPEL\BSE\Outbound_Project
```

アウトバウンド・デザインタイム構成を完了するには、次のツールが必要です。

- Oracle Adapter Application Explorer (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPEL Designer (JDeveloper)

この項では、次の項目について説明します。

- 6.7.1 項「Web サービスを使用したリクエストおよびレスポンス・サービスの WSDL ファイルの生成」
- 6.7.2 項「SOA 用の空のコンポジットの作成」
- 6.7.3 項「BPEL アウトバウンド・プロセスの定義」



## 6.7.1 Web サービスを使用したリクエストおよびレスポンス・サービスの WSDL ファイルの生成

BPEL プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。

Web サービスを使用してリクエストおよびレスポンス・サービスの WSDL ファイルを生成するには、次のようにします。

1. アプリケーション・エクスプローラを起動し、定義済の MySAP ターゲット (BSE 構成) に接続します。  
SAP R/3 へのターゲットの定義および接続の詳細は、4.4.1 項「SAP R/3 へのターゲットの定義」を参照してください。
2. 接続先の MySAP ターゲットを展開します。
3. 「リモート関数モジュール」 → 「Financial Accounting」 → 「0002 – Company Code Business Object」を展開し、「BAPI\_COMPANYCODE\_GETDETAIL」を選択します。
4. 図 6-87 に示すように、「GetDetail」ノードを右クリックし、メニューの「Web サービスの作成」を選択します。

図 6-87 「GetDetail」ノード

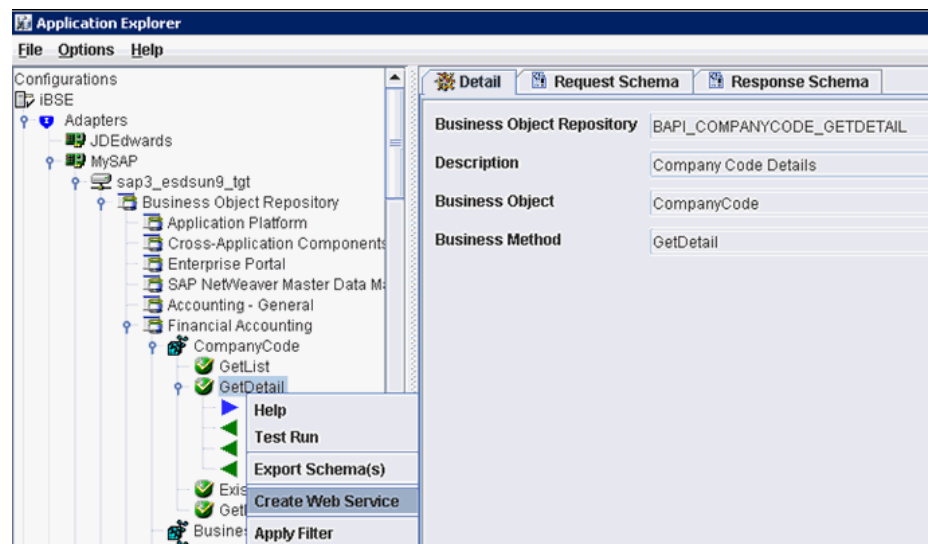
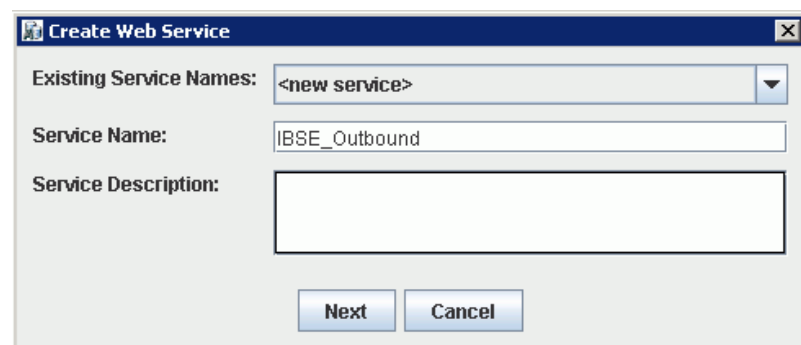


図 6-88 に示すように、「Web サービスの作成」ダイアログが表示されます。

図 6-88 「Web サービスの作成」ダイアログ



5. サービス名を入力し、「次へ」をクリックします。
6. 表示される次のダイアログで、「OK」をクリックします。  
アプリケーション・エクスペローラの表示が「ビジネス・サービス」ノードに切り替わり、新規 Web サービスが左ペインに表示されます。
7. 新規 Web サービスを右クリックして、メニューの「WSDL の保存」を選択します。
8. 「wsdls」フォルダに WSDL を保存し、「保存」をクリックします。  
これで、JDeveloper で BPEL アウトバウンド・プロセスを定義するために必要な最初の手順として、SOA 用の空のコンポジットを作成できるようになりました。

## 6.7.2 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

1. 新規 SOA アプリケーションを作成します。
2. SOA アプリケーションの名前を入力し、「次へ」をクリックします。  
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクト名を入力し、「次へ」をクリックします。  
SOA 設定の構成ページが表示されます。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。  
詳細は、6-11 ページの第 6.4.2 項「SOA 用の空のコンポジットの作成」を参照してください。

## 6.7.3 BPEL アウトバウンド・プロセスの定義

この項では、BPEL アウトバウンド・プロセス・コンポーネントを構成する方法について説明します。

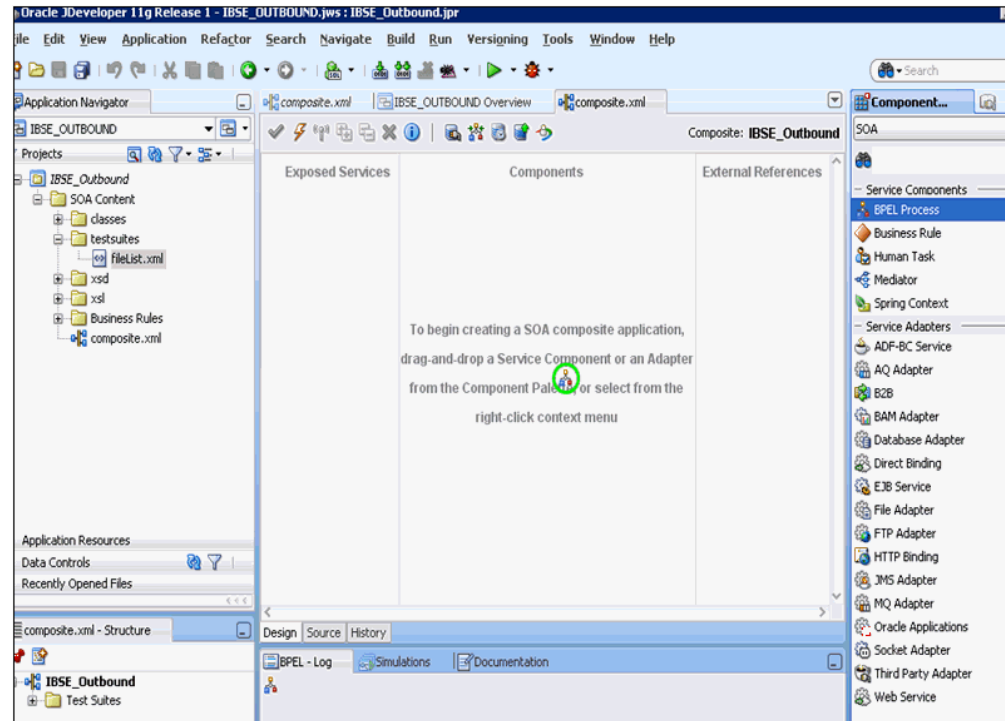
この項では、次の項目について説明します。

- 6.7.3.1 項「パートナ・リンクの作成」
- 6.7.3.2 項「BPEL アクティビティの作成および作成済パートナ・リンクとのマッピング」

BPEL アウトバウンド・プロセスを定義するには、次のようにします。

1. 図 6-89 に示すように、「BPEL プロセス」コンポーネントを「サービス・コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

図 6-89 BPEL プロセス・コンポーネント



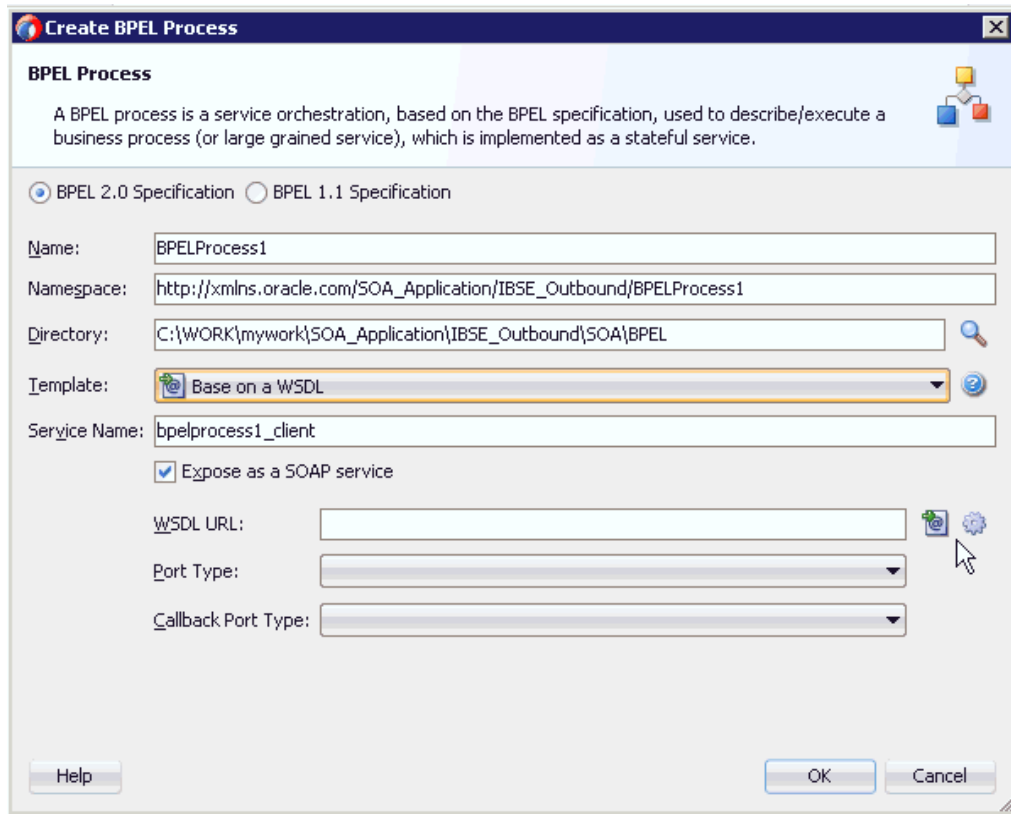
2. 「名前」フィールドに、新規アウトバウンド BPEL プロセス・コンポーネントを識別する名前を入力します。

デフォルトでは、「BPEL 2.0 仕様」オプションが選択されています。

3. 「テンプレート」ドロップダウン・リストで「WSDL に基づく」を選択します。

4. 図 6-90 に示すように、「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。

図 6-90 「既存の WSDL を検索します。」アイコン



「WSDL の選択」 ダイアログが表示されます。


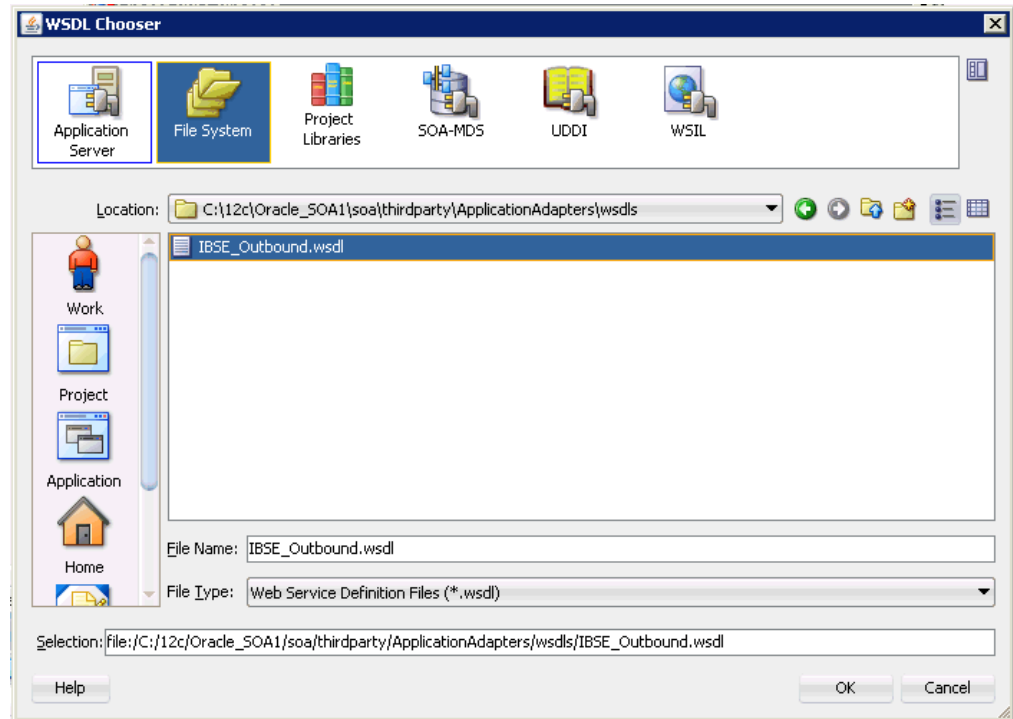
5.  6-91 に示すように、アプリケーション・エクスプローラから WSDL がエクスポートされる場所に移動し、WSDL を選択して「OK」をクリックします。

図 6-91 「WSDL の選択」 ダイアログ



「ファイルのローカライズ」ウィンドウが表示されます。


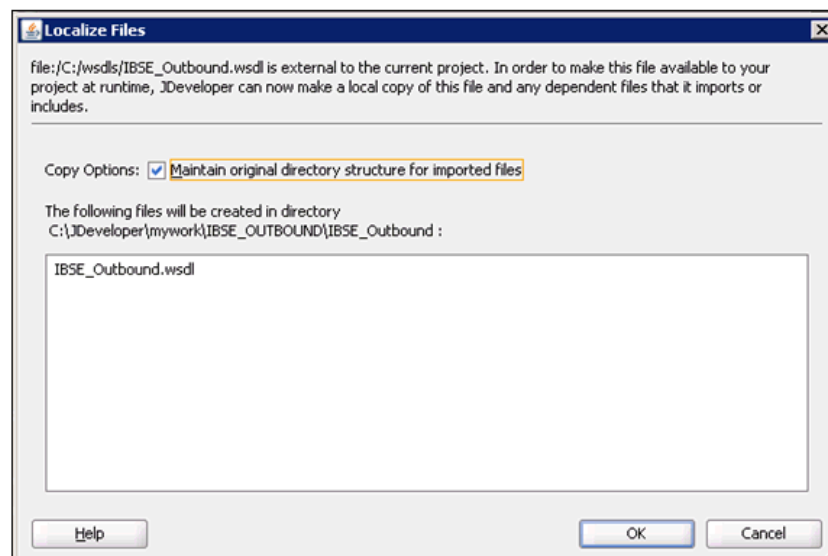
6. 表示される「ファイルのローカライズ」ウィンドウで、「OK」をクリックします。これにより、 6-92 に示すように、WSDL ファイルがプロジェクト・フォルダにインポートされます。

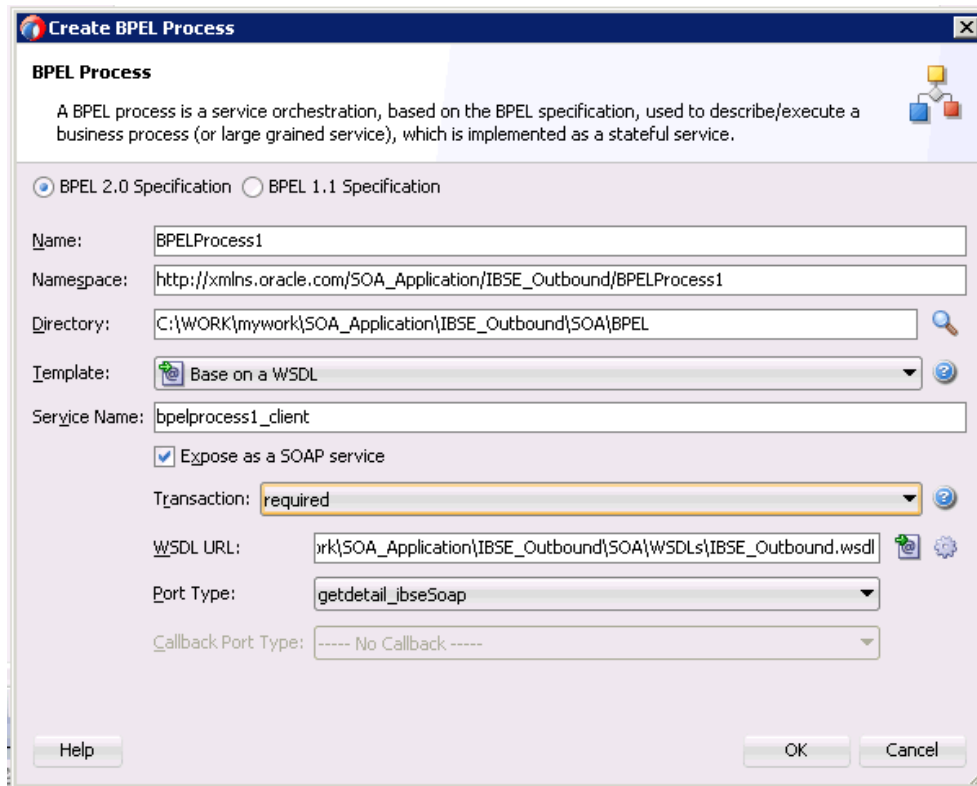
図 6-92 「ファイルのローカライズ」ウィンドウ



「BPEL プロセスの作成」ウィンドウが表示されます。

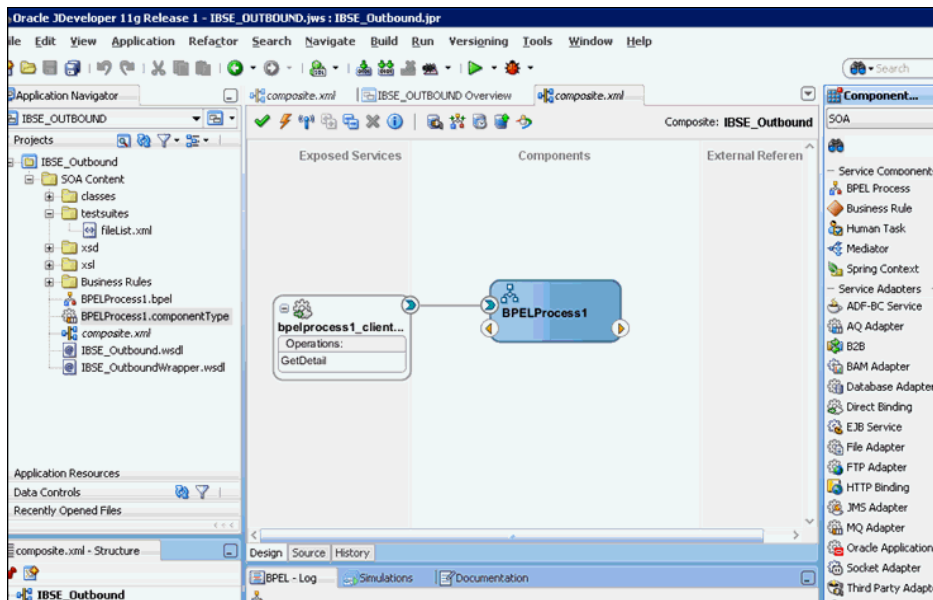
7. 図 6-93 に示すように、「BPEL プロセス」ペインで、「OK」をクリックします。

図 6-93 「BPEL プロセス」ペイン



- 図 6-94 に示すように、「BPEL プロセス」コンポーネントが作成され、表示されます。

図 6-94 BPEL プロセス・コンポーネント



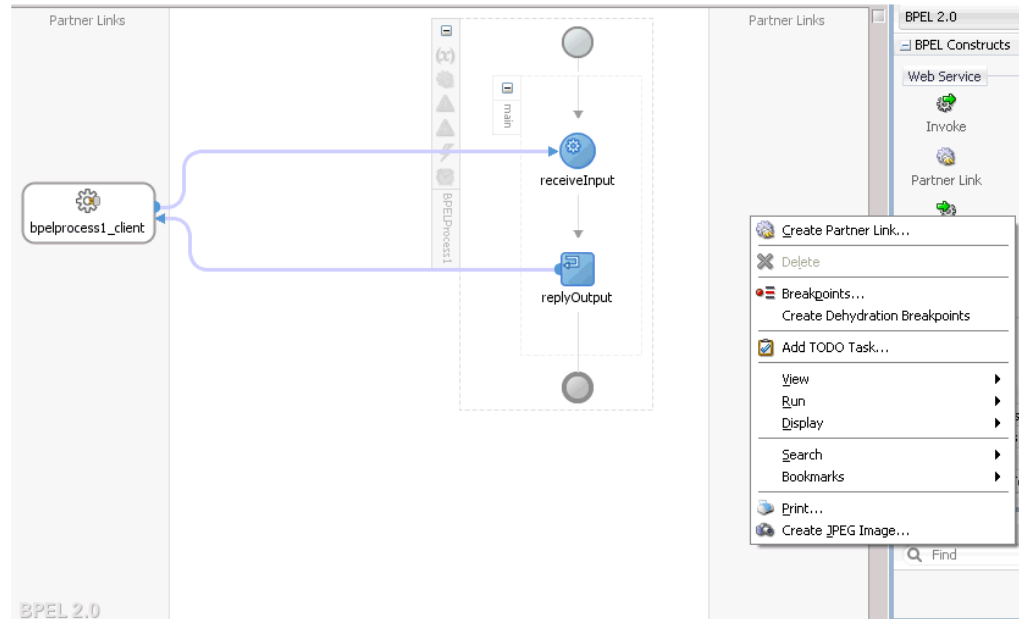
### 6.7.3.1 パートナ・リンクの作成

この項では、パートナ・リンクを作成する方法について説明します。

パートナ・リンクを作成するには、次のようにします。

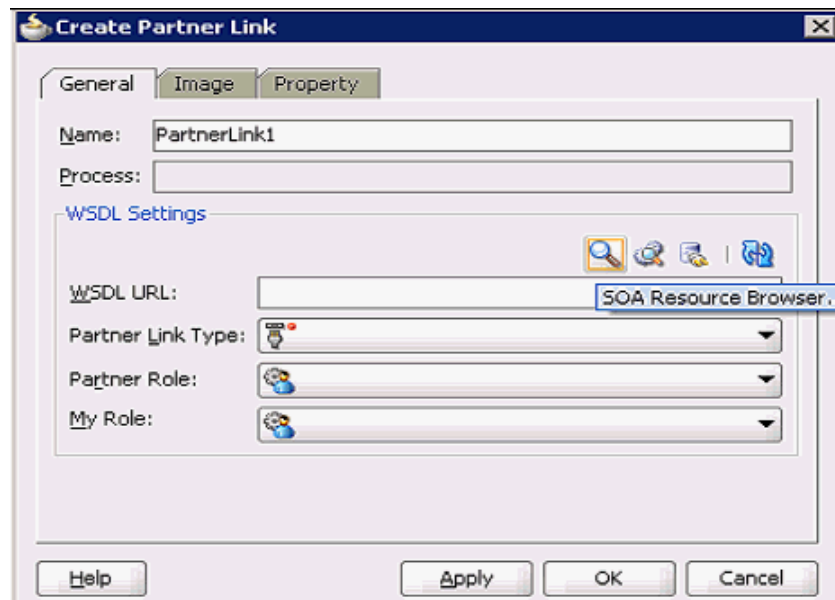
1. 「コンポーネント」ペインで、アウトバウンド BPEL プロセス・コンポーネントをダブルクリックします。
2. 図 6-95 に示すように、「パートナ・リンク」ペインで右クリックして、「パートナ・リンクの作成」を選択します。

図 6-95 パートナ・リンクの作成



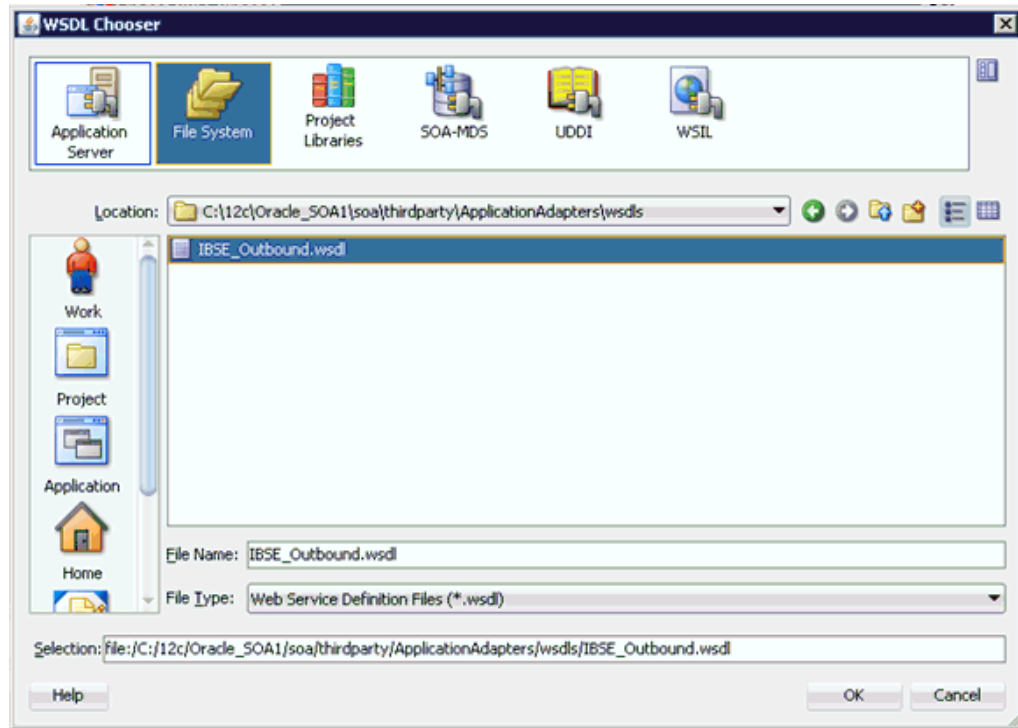
3. 表示される「パートナ・リンクの作成」ウィンドウで、図 6-96 に示すように、適切な名前を指定し、「SOA リソース・ブラウザ」ツールをクリックします。

図 6-96 「SOA リソース・ブラウザ」ツール



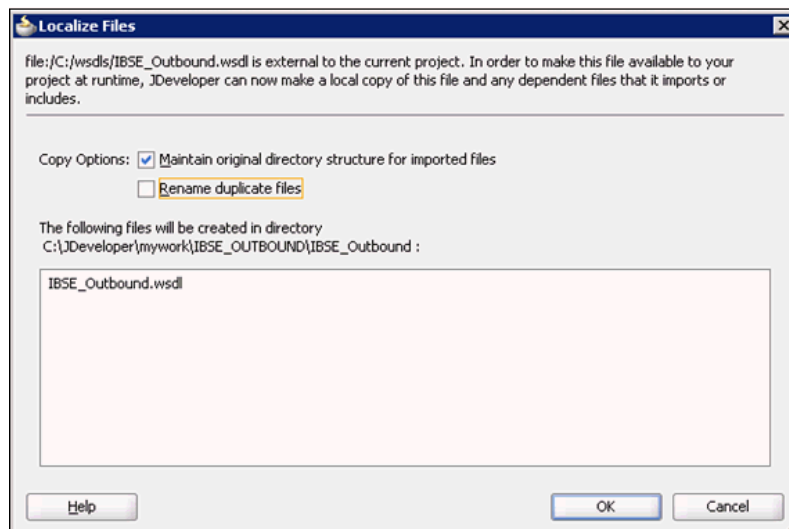
4. 図 6-97 に示すように、表示される「WSDL の選択」ダイアログで、アプリケーション・エクスプローラから WSDL がエクスポートされる場所に移動し、WSDL を選択して「OK」をクリックします。

図 6-97 「WSDL の選択」ダイアログ



5. 図 6-98 に示すように、表示される「ファイルのローカライズ」ウィンドウで、「重複ファイルの名前変更」チェック・ボックスの選択を解除し、「OK」をクリックします。

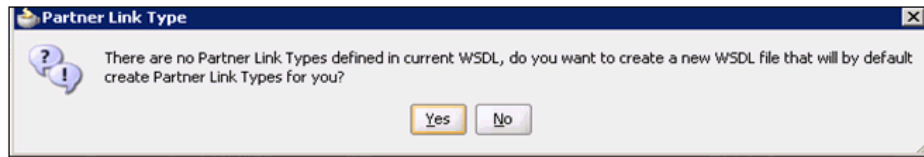
図 6-98 「ファイルのローカライズ」ウィンドウ





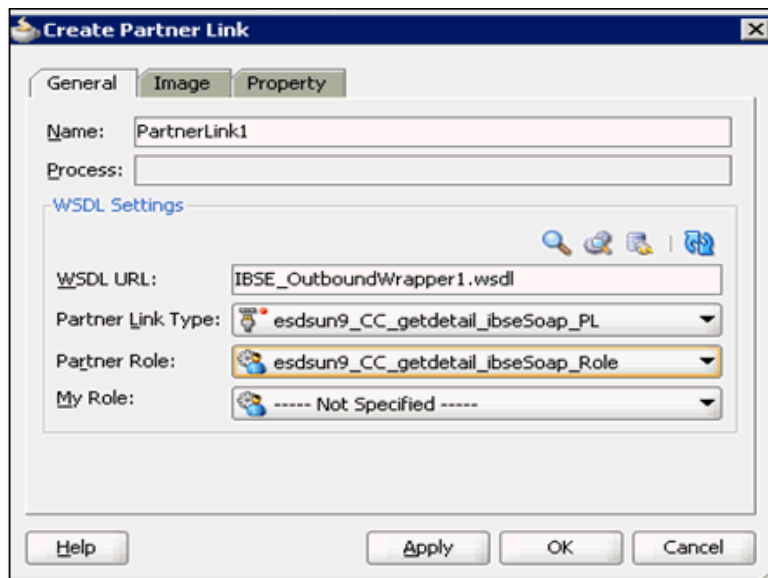
6. 図 6-99 に示すように、表示される「パートナ・リンク・タイプ」ウィンドウで「はい」をクリックします。

図 6-99 パートナ・リンク・タイプ



7. 表示される「パートナ・リンクの作成」ウィンドウで、「パートナ・ロール」ドロップダウン・リストを展開して、使用可能なパートナ・ロールを選択します。
8. 図 6-100 に示すように、「適用」をクリックしてから、「OK」をクリックします。

図 6-100 パートナ・リンクの作成



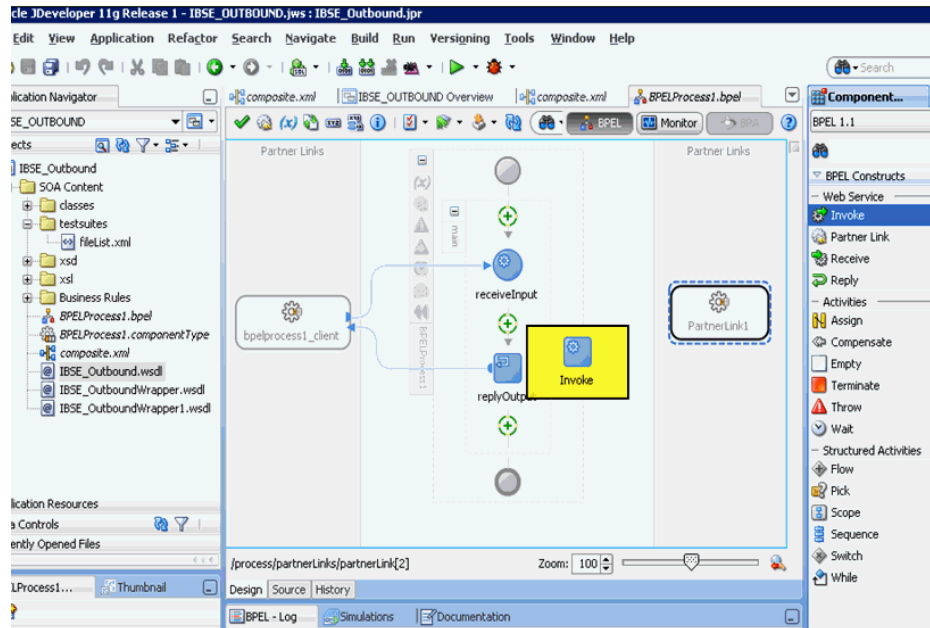
### 6.7.3.2 BPEL アクティビティの作成および作成済パートナ・リンクとのマッピング

この項では、BPEL アクティビティを作成して、作成済のパートナ・リンクとマッピングする方法について説明します。

BPEL アクティビティを作成して作成済パートナ・リンクとマッピングするには、次のようにします。

1. 「起動」アクティビティ・コンポーネントを「BPEL コンストラクト」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。これを図 6-101 に示すように、**receiveInput** アクティビティ・コンポーネントと **replyOutput** アクティビティ・コンポーネントの間に配置します。

図 6-101 「起動」 アクティビティ・コンポーネント




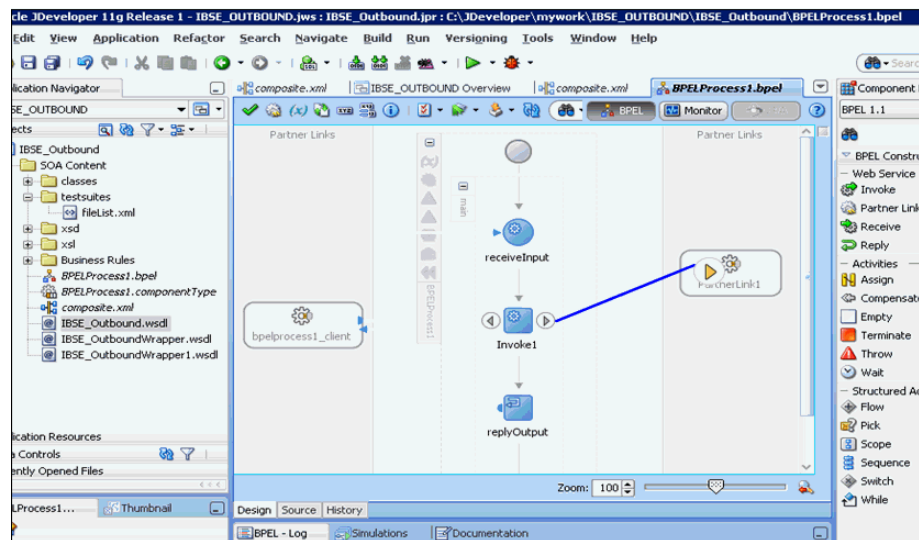
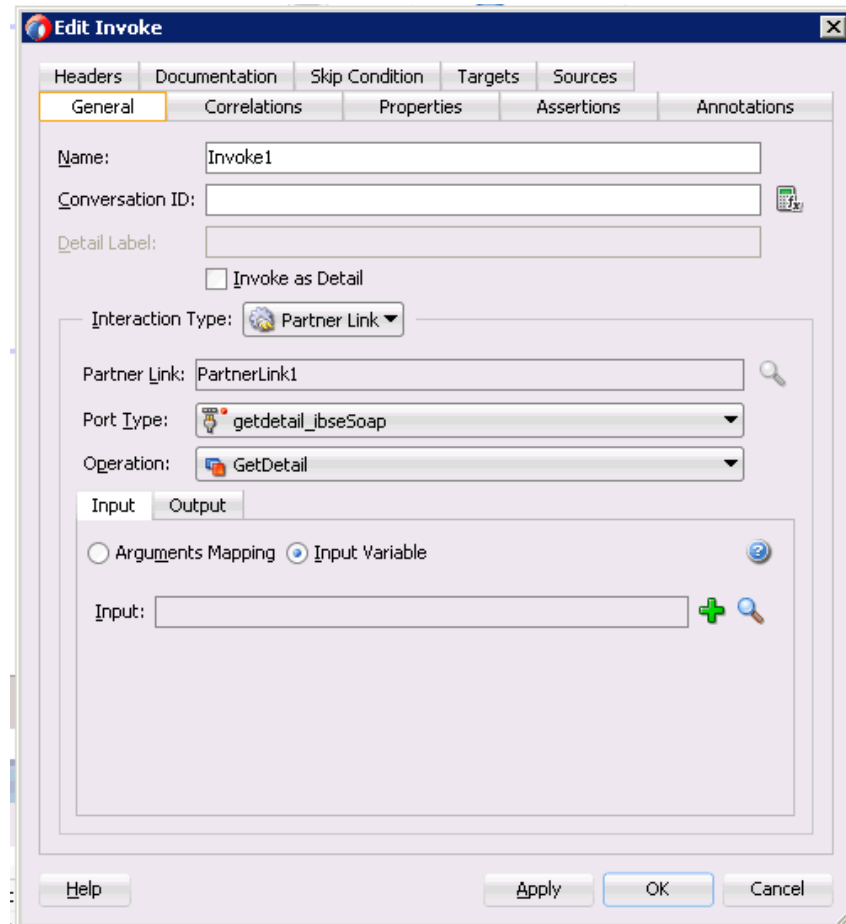
2.  6-102 に示すように、新規「起動」アクティビティ・コンポーネント (Invoke1) と「パートナ・リンク」コンポーネント (Partner link1) 間の接続を作成します。

図 6-102 パートナ・リンク・コンポーネント



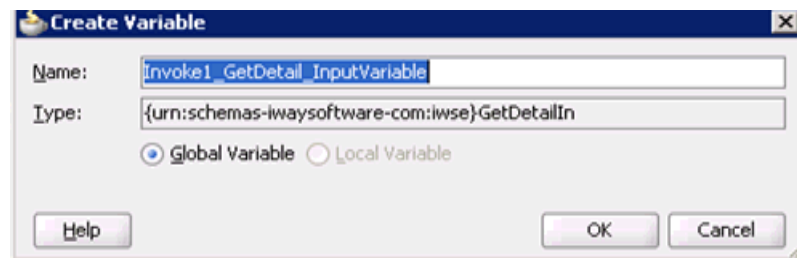
3. 図 6-103 に示すように、表示される「起動の編集」ウィンドウで、「入力」フィールドの右側にあるプラス (+) アイコンをクリックして、新規入力変数を構成します。

図 6-103 「起動の編集」ウィンドウ



4. 図 6-104 に示すように、新規入力変数として提供されるデフォルト値を受け入れ、「OK」をクリックします。

図 6-104 新規入力変数



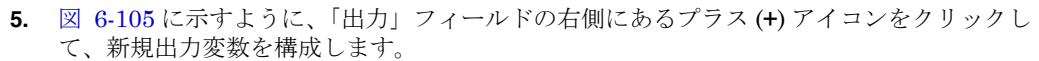
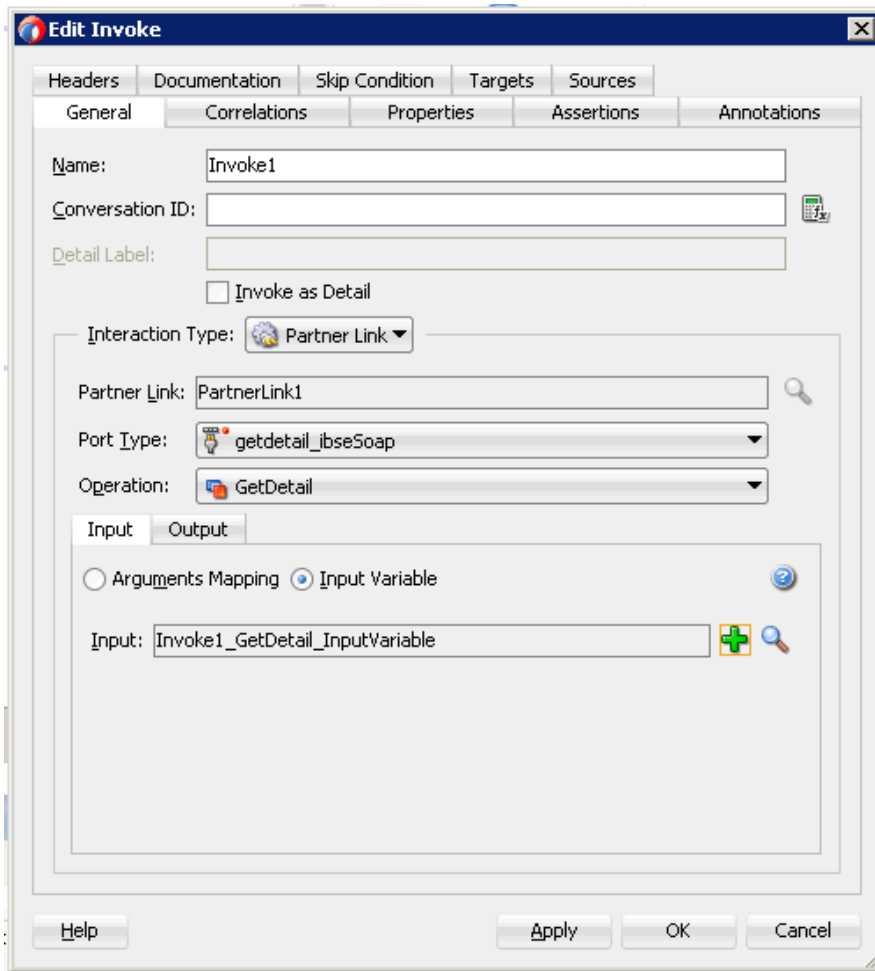
5.  6-105 に示すように、「出力」フィールドの右側にあるプラス (+) アイコンをクリックして、新規出力変数を構成します。

図 6-105 「起動の編集」 ウィンドウ



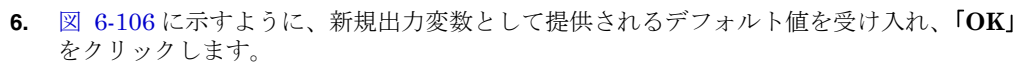
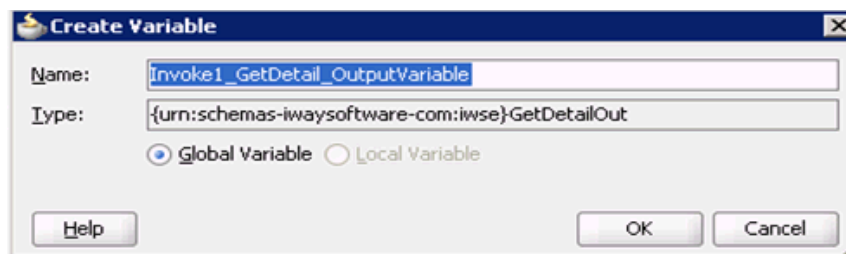
6.  6-106 に示すように、新規出力変数として提供されるデフォルト値を受け入れ、「OK」をクリックします。

図 6-106 新規出力変数



7. 図 6-107 に示すように、「適用」をクリックしてから、「OK」をクリックします。

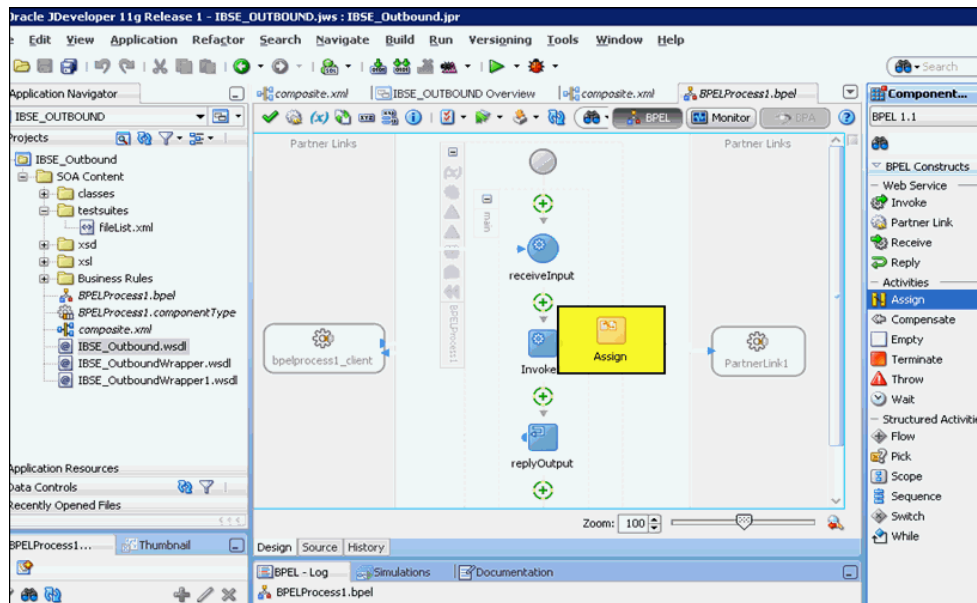
図 6-107 「起動の編集」ウィンドウ

The screenshot shows the 'Edit Invoke' dialog box with the following configuration:

- General Tab:**
  - Name: Invoke1
  - Conversation ID: (empty)
  - Detail Label: (empty)
  - Invoke as Detail
  - Interaction Type: Partner Link
  - Partner Link: PartnerLink1
  - Port Type: getdetail\_ibseSoap
  - Operation: GetDetail
- Output Section:**
  - Arguments Mapping
  - Output Variable
  - Output: Invoke1\_GetDetail\_OutputVariable

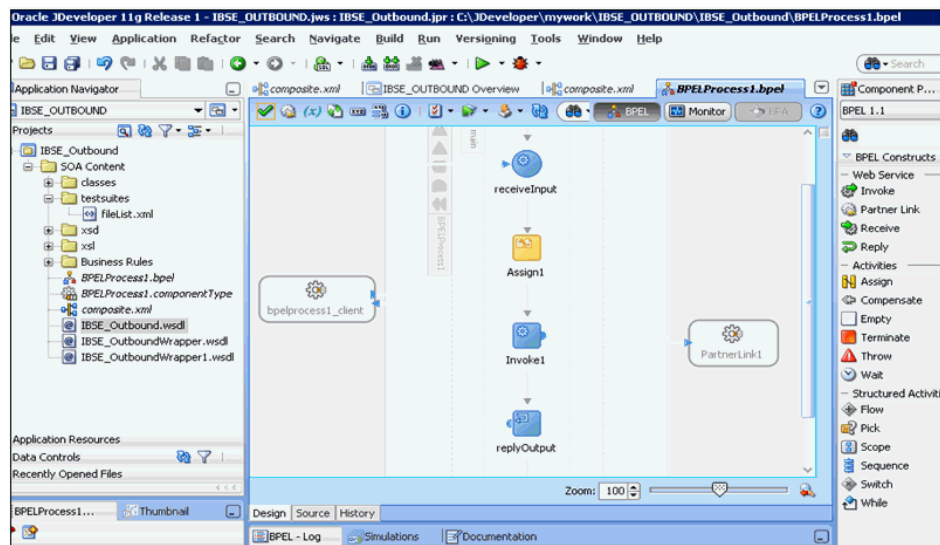
- 「割当て」アクティビティ・コンポーネントを「BPEL コンストラクト」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。これを図 6-108 に示すように、「受信」アクティビティ・コンポーネント (receiveInput) と「起動」アクティビティ・コンポーネント (Invoke1) の間に配置します。

図 6-108 「割当て」アクティビティ・コンポーネント



- 図 6-109 に示すように、新規「割当て」アクティビティ・コンポーネント (Assign1) をダブルクリックします。

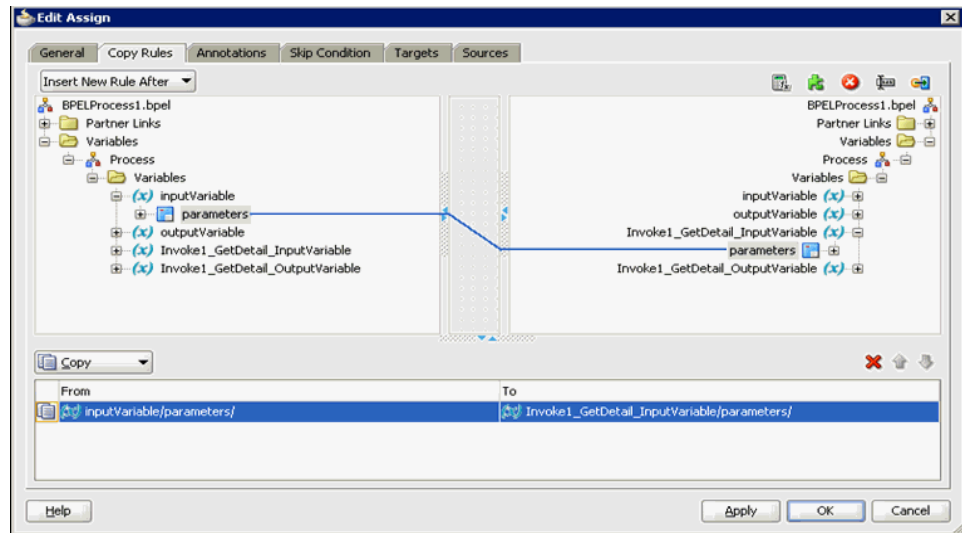
図 6-109 「割当て」アクティビティ・コンポーネント



- 左ペインで、「変数」の下の「InputVariable」を展開し、「パラメータ」を選択します。
- 右ペインで、「変数」の下の「Invoke1\_GetDetail\_InputVariable」を展開し、「パラメータ」を選択します。

12. 図 6-110 に示すように、「InputVariable」パラメータを「Invoke1\_GetDetail\_InputVariable」パラメータにドラッグしてマップします。

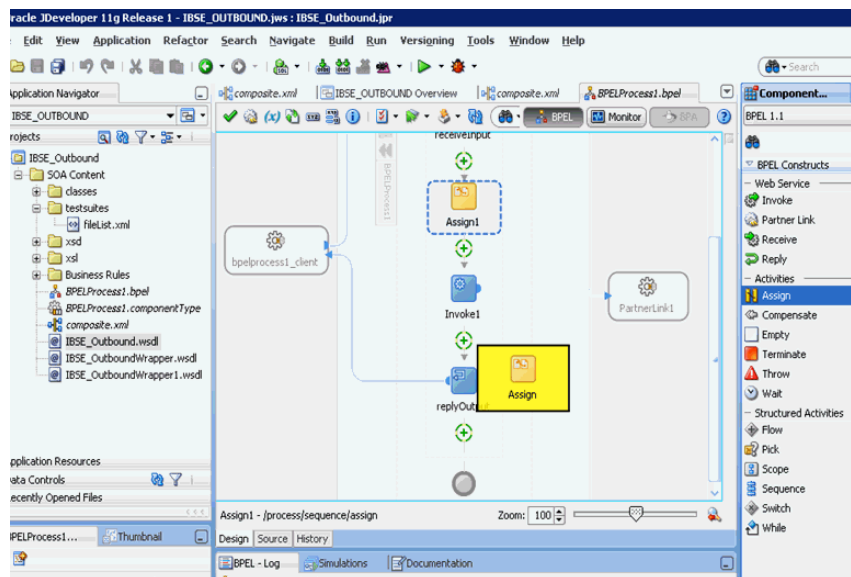
図 6-110 InputVariable パラメータ



13. 「適用」をクリックしてから、「OK」をクリックします。

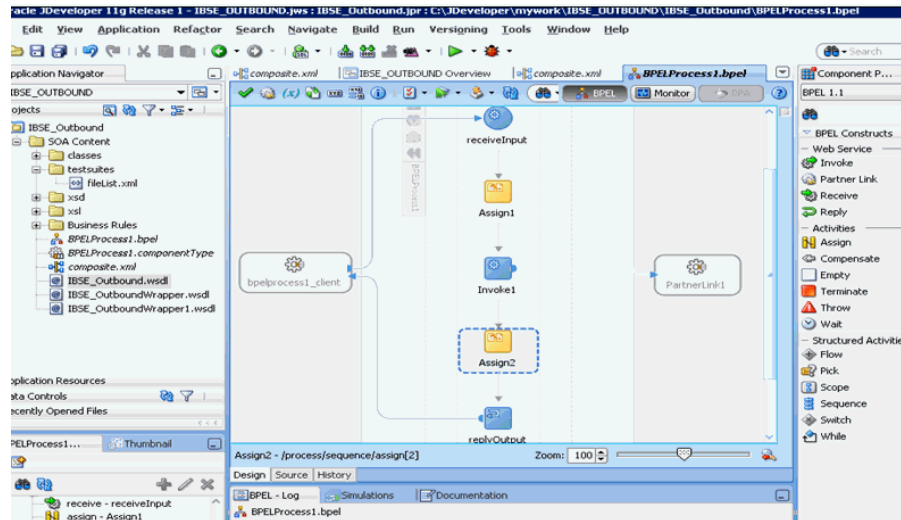
14. 図 6-111 に示すように、「割当て」アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップし、「起動」アクティビティ (Invoke1) と「返信」アクティビティ (replyOutput) の間に配置します。

図 6-111 「割当て」アクティビティ・コンポーネント



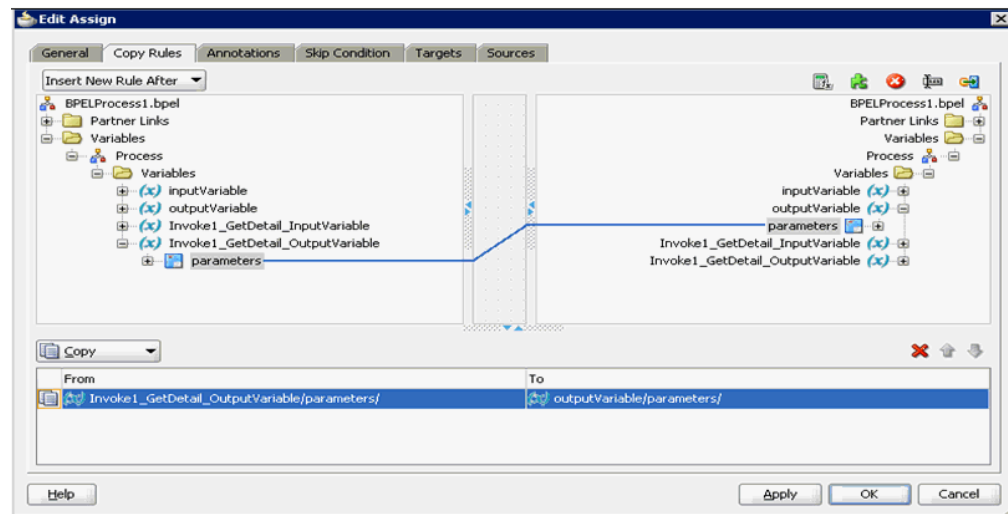
15. 図 6-112 に示すように、新規「割当て」アクティビティ・コンポーネント (Assign2) をダブルクリックします。

図 6-112 新規「割当て」アクティビティ・コンポーネント



16. 左ペインで、「変数」の下の「Invoke1\_GetDetail\_OutputVariable」を展開し、「パラメータ」を選択します。
17. 右ペインで、「変数」の下の「outputVariable」を展開し、「パラメータ」を選択します。
18. 図 6-113 に示すように、「Invoke1\_GetDetail\_OutputVariable」パラメータを「outputVariable」パラメータにドラッグしてマップします。

図 6-113 outputVariable パラメータ

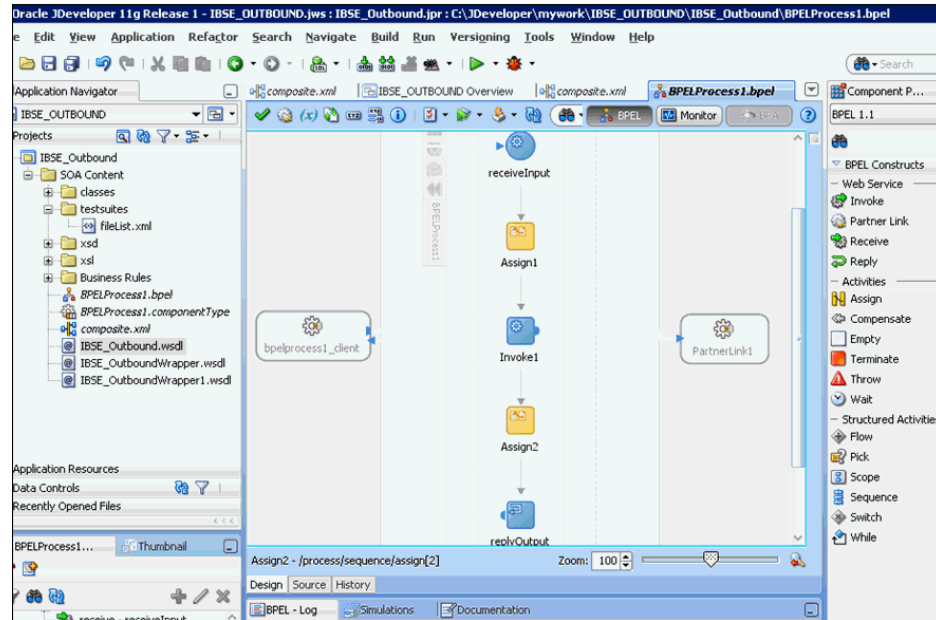




19. 「適用」をクリックしてから、「OK」をクリックします。

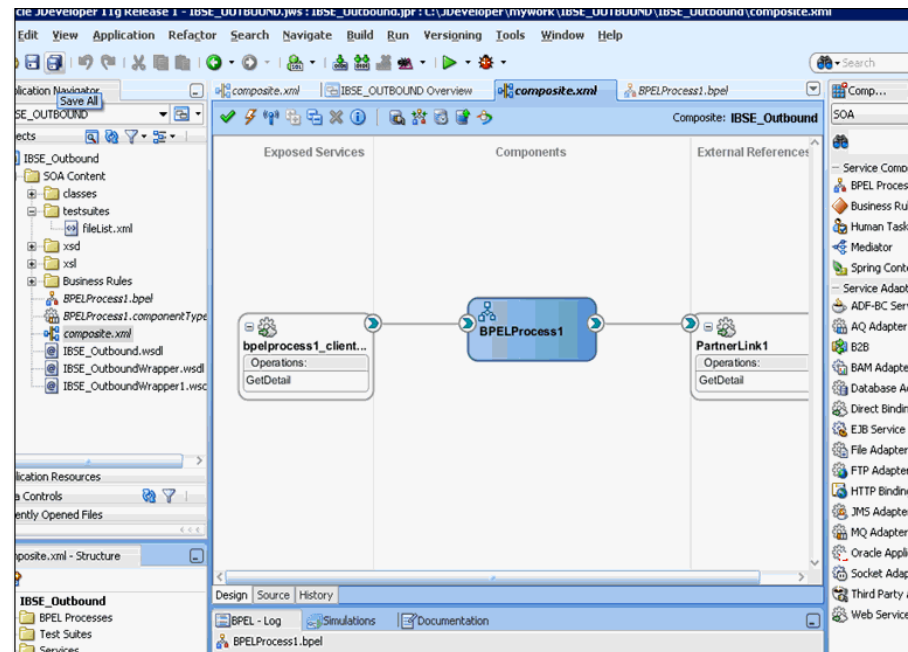
図 6-114 に示すように、「コンポーネント」ペインに戻ります。

図 6-114 「コンポーネント」ペイン



20. 図 6-115 に示すように、メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド BPEL プロセス・コンポーネントを保存します。

図 6-115 「すべて保存」アイコン



これで、BPEL アウトバウンド・プロセスをデプロイする準備ができました。6.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」と同じ手順を実行できます。

デプロイが完了したら、[6.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」](#) で定義されているように入力 XML を呼び出すことができます。

---

---

## Oracle SOA Suite でのメディエータ・サービス・コンポーネントとの統合

この章では、Oracle SOA Suite でのメディエータ・サービス・コンポーネントとの統合について説明します。次の項で構成されています。

- 7.1 項「新規アプリケーション・サーバー接続の構成」
- 7.2 項「メディエータ・アウトバウンド・プロセスの構成 (J2CA 構成)」
- 7.3 項「メディエータ・インバウンド・プロセスの構成 (J2CA 構成)」
- 7.4 項「メディエータ・アウトバウンド・プロセスの構成 (BSE 構成)」

この章で示されるシナリオには次の前提条件が必要です。

### 前提条件

インストールおよび構成の要件は次のとおりです。

- Oracle Application Adapter for SAP R/3 は、Oracle WebLogic Server にインストールされている必要があります。
- SAP R/3 は、インバウンドおよびアウトバウンド処理用に構成されている必要があります。詳細は、付録 A「リモート処理での SAP のシステム・ロールについて」を参照してください。

この章の例では、サービスおよびイベントの SAP R/3 との統合を説明するために必要な構成手順を示しています。この情報を使用する前に、次のことをよく理解しておく必要があります。

- サービスおよびイベント用に Oracle Application Adapter for SAP R/3 を構成する方法。詳細は、第 4 章「構成 Oracle Application Adapter for SAP R/3」を参照してください。
- Oracle JDeveloper を構成する方法。詳細は、第 6 章「Oracle SOA Suite での BPEL サービス・コンポーネントとの統合」を参照してください。

### メディエータ統合の概要

メディエータは包括的なアプリケーション統合フレームワークを提供します。Oracle Application Adapter for SAP R/3 をメディエータとともに使用することで、カスタム・コードを使用することなく、エンタープライズ・ソフトウェアをシームレスに統合できます。カスタム・コーディング・ソリューションと異なり、機能的なモデリングのため、ソフトウェアを再利用でき、ソフトウェア・ライフサイクルにわたって発生する複雑さや管理上の問題を軽減します。この統合モデルには、高レベルの統合ロジックおよび低レベルのプラットフォーム・サービスの 2 つの構成要素があります。

アダプタと Oracle メディエータとの統合は 2 ステップのプロセスです。

1. **デザインタイム**：第 4 章「構成 Oracle Application Adapter for SAP R/3」の説明に従って、アプリケーション・エクスポーラで Oracle Application Adapter for SAP R/3 をサービスおよびイベント用に構成します。統合ロジックはメディエータでモデリングされます。

2. **ランタイム**: 基礎となるプラットフォームでは、このメタデータをランタイム手順として扱い、関与しているアプリケーション間の通信を可能にします。

## 7.1 新規アプリケーション・サーバー接続の構成

Oracle JDeveloper で新規アプリケーション・サーバー接続を構成する方法の詳細は、[6.3 項「新規アプリケーション・サーバー接続の構成」](#)を参照してください。

## 7.2 メディエータ・アウトバウンド・プロセスの構成 (J2CA 構成)

この項では、Oracle JDeveloper でメディエータ・プロジェクトを使用して SAP R/3 システムへのメディエータ・アウトバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\Mediator\J2CA\Outbound_Project
```

ここでは、次の項目について説明します。

- [7.2.1 項「SOA 用の空のコンポジットの作成」](#)
- [7.2.2 項「メディエータ・アウトバウンド・プロセスの定義」](#)
- [7.2.3 項「メディエータ・アウトバウンド・プロセスのデプロイ」](#)
- [7.2.4 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」](#)

### 前提条件

メディエータ・アウトバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、[6-9 ページの「リクエスト/レスポンス・サービスの WSDL の生成」](#)を参照してください。

### 7.2.1 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

1. 新規 SOA アプリケーションを作成します。
2. 新規 SOA アプリケーションの名前を入力し、「**次へ**」をクリックします。  
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクト名を入力し、「**次へ**」をクリックします。  
SOA 設定の構成ページが表示されます。
4. 「コンポジット・テンプレート」リストで「**空のコンポジット**」を選択し、「**終了**」をクリックします。

詳細は、[6-11 ページの 6.4.2 項「SOA 用の空のコンポジットの作成」](#)を参照してください。

## 7.2.2 メディエータ・アウトバウンド・プロセスの定義

この項では、メディエータ・アウトバウンド・プロセスを定義する方法について、次の項目で説明します。

- 7.2.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」
- 7.2.2.2 項「アウトバウンド・メディエータ・プロセス・コンポーネントの構成」
- 7.2.2.3 項「ルーティング・ルールの構成」
- 7.2.2.4 項「12c でのデプロイメントに関する既知の問題の調整」

### 7.2.2.1 サード・パーティ・アダプタ・サービス・コンポーネントの構成

サード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

1. 「サード・パーティ・アダプタ」コンポーネントを「サービス・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。
2. サード・パーティ・アダプタ・サービスの名前を入力します。
3. 「タイプ」ドロップダウン・リストから「参照」が選択されていることを確認します (デフォルト)。
4. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。
5. 次のディレクトリでアウトバウンド WSDL ファイルを参照して選択します。

```
<ADAPTER_HOME>\wsdls
```

6. 「OK」をクリックします。
7. 「OK」をクリックします。

アウトバウンド WSDL ファイル、および関連するリクエストとレスポンスの XML スキーマ・ファイル (.xsd) が、作成されたプロジェクト・フォルダにインポートされます。

8. 「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。
9. 次のディレクトリで JCA プロパティ・ファイルを参照して選択します。

```
<ADAPTER_HOME>\wsdls
```

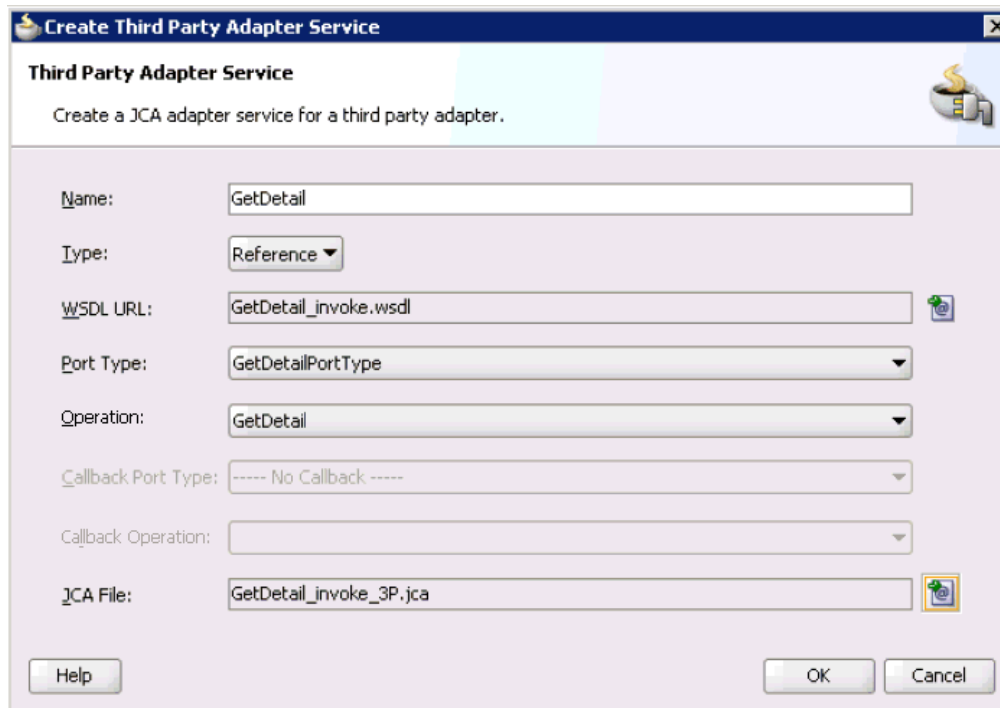
10. 「OK」をクリックします。

「ファイルのコピー」確認メッセージが表示されます。

11. 「はい」をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

図 7-1 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



12. 「OK」をクリックします。

「外部参照」 ペインにサード・パーティ・アダプタ・サービス・コンポーネント (GetDetail) が作成されます。

これで、アウトバウンド・メディエータ・プロセス・コンポーネントを構成する準備ができました。

詳細は、6-14 ページの [6.4.3.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」](#)を参照してください。

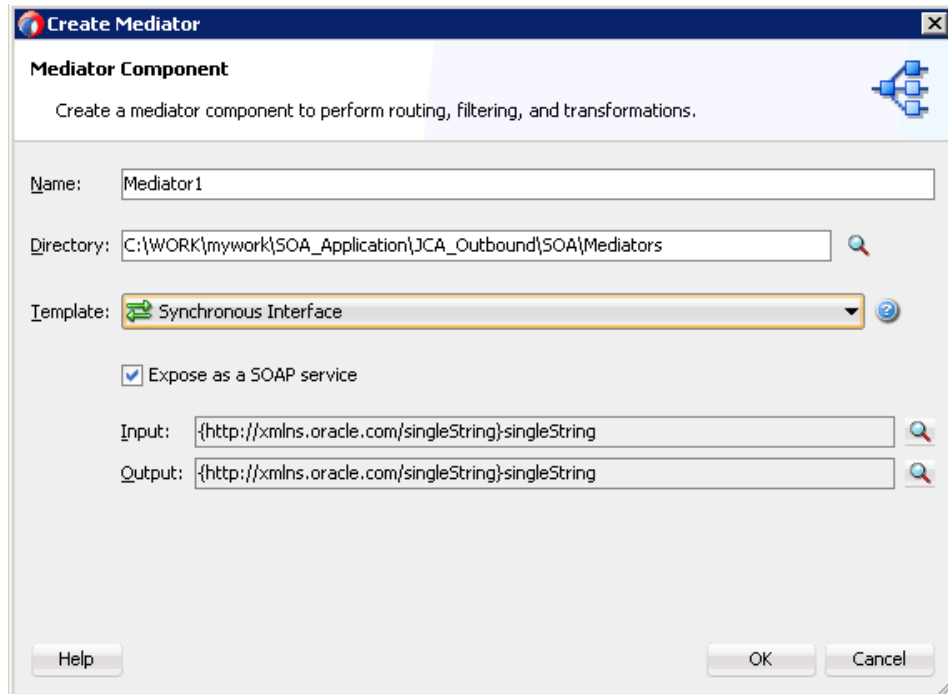
### 7.2.2.2 アウトバウンド・メディエータ・プロセス・コンポーネントの構成

アウトバウンド・メディエータ・プロセス・コンポーネントを構成するには、次のようにします。

1. **メディエータ・プロセス・コンポーネント**を「コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

図 7-2 に示すように、「メディエータの作成」ダイアログが表示されます。

図 7-2 「メディエータの作成」ダイアログ

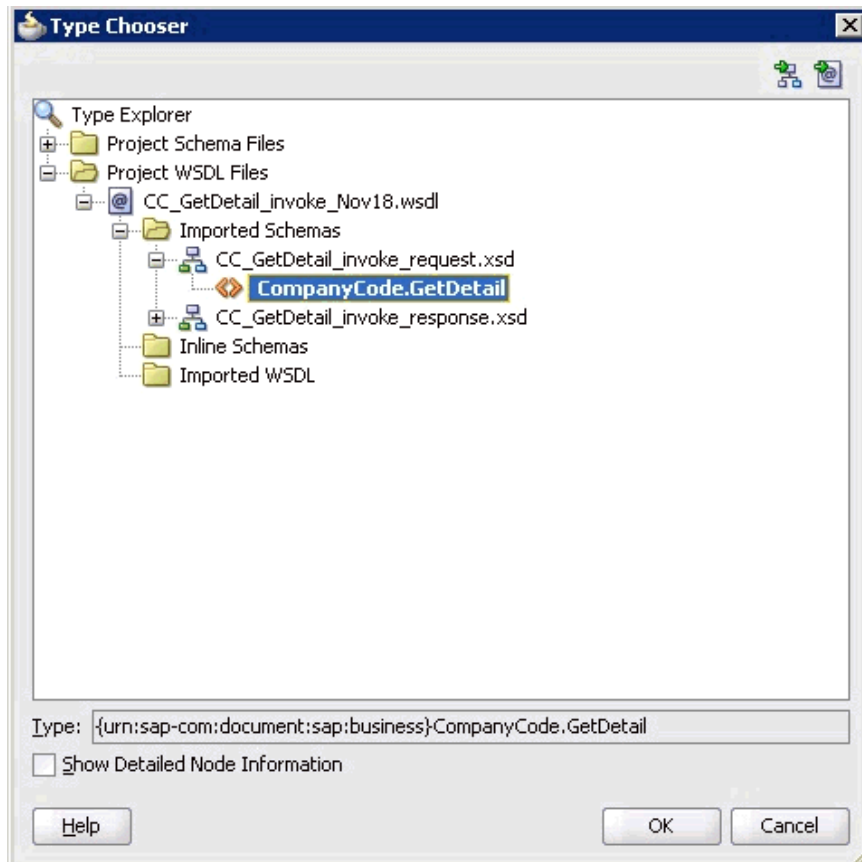


2. 「名前」フィールドに、新規アウトバウンド・メディエータ・プロセス・コンポーネントを識別する名前を入力します。
3. 「テンプレート」ドロップダウン・リストで「同期インタフェース」を選択します。

4. 「入力」フィールドの右側にある「参照」アイコンをクリックし、関連する XML リクエスト・スキーマ・ファイルを選択します。

図 7-3 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 7-3 「タイプ・チューザ」ダイアログ



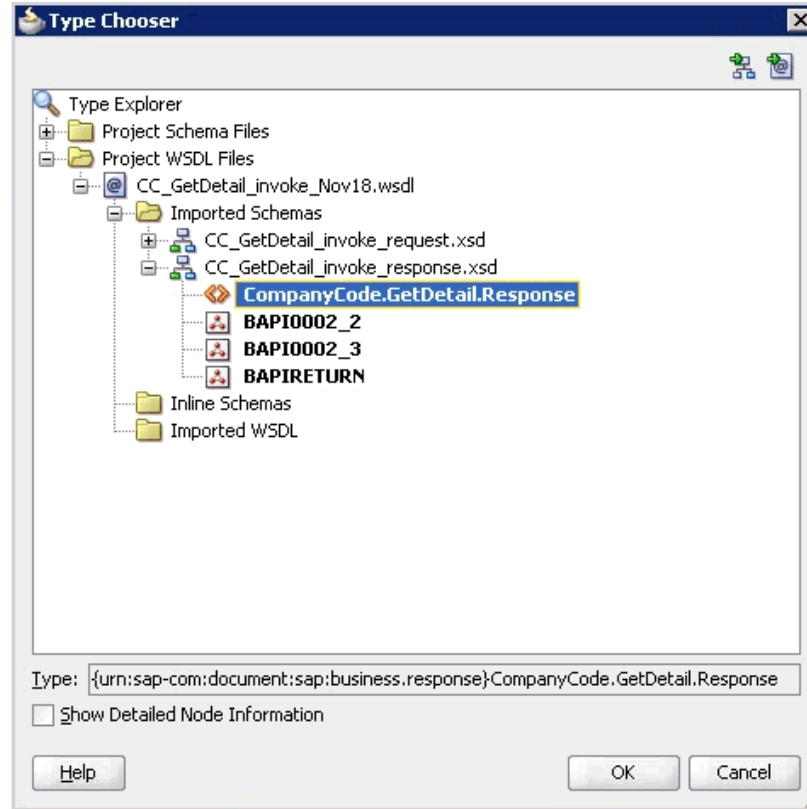
5. 「プロジェクトの WSDL ファイル」 → 「CC\_GetDetail\_invoke\_Nov18.wsdl」 → 「インポートしたスキーマ」 → 「CC\_GetDetail\_invoke\_request.xsd」を展開し、「CompanyCode.GetDetail」を選択します。
6. 「OK」をクリックします。  
「メディアータの作成」ダイアログに戻ります。



- 「出力」フィールドの右側にある「参照」アイコンをクリックし、関連する XML レスポンス・スキーマ・ファイルを選択します。

図 7-4 に示すように、「タイプ・チューザ」ダイアログが表示されます。

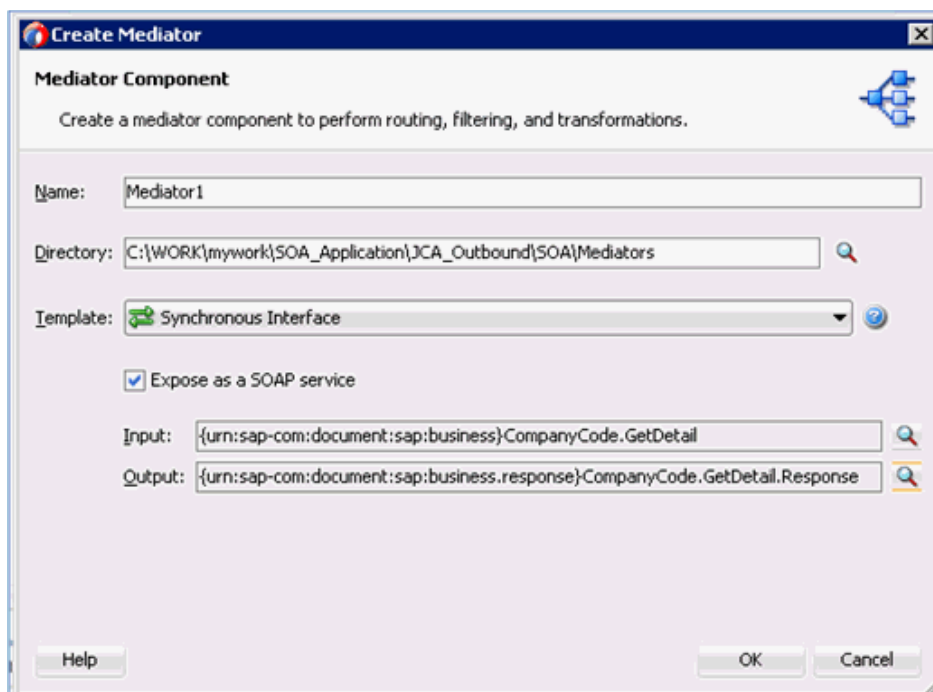
図 7-4 「タイプ・チューザ」ダイアログ



- 「プロジェクトの WSDL ファイル」 → 「CC\_GetDetail\_invoke\_Nov18.wsdl」 → 「インポートしたスキーマ」 → 「CC\_GetDetail\_invoke\_response.xsd」を展開し、「CompanyCode.GetDetailResponse」を選択します。

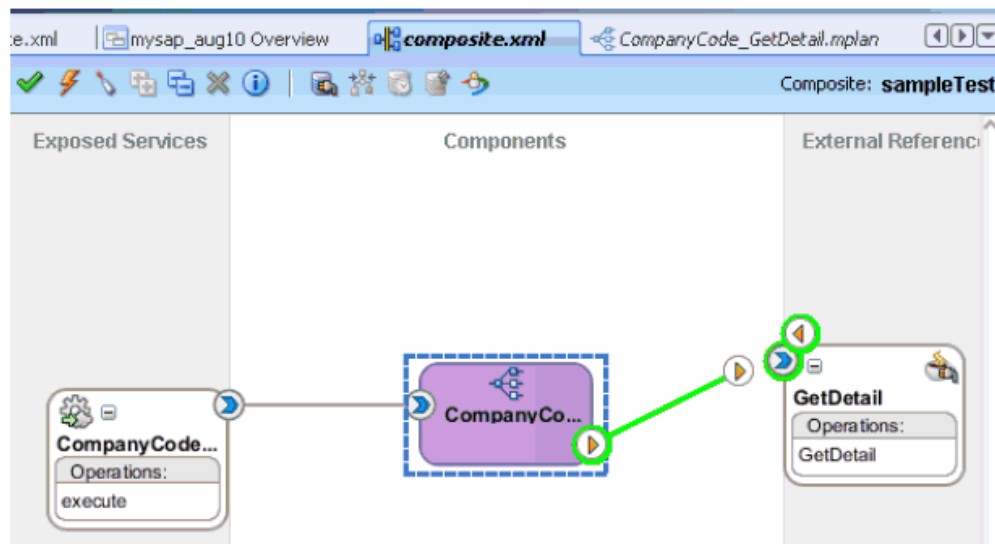
9. 「OK」をクリックします。  
 図 7-5 に示すように、「メディエータの作成」ダイアログに戻ります。

図 7-5 「メディエータの作成」ダイアログ



10. 「OK」をクリックします。
11. 図 7-6 に示すように、アウトバウンド・メディエータ・プロセス・コンポーネントとサード・パーティ・アダプタ・サービス・コンポーネント (GetDetail) 間の接続を作成します。

図 7-6 作成された接続



これで、ルーティング・ルールを構成する準備ができました。

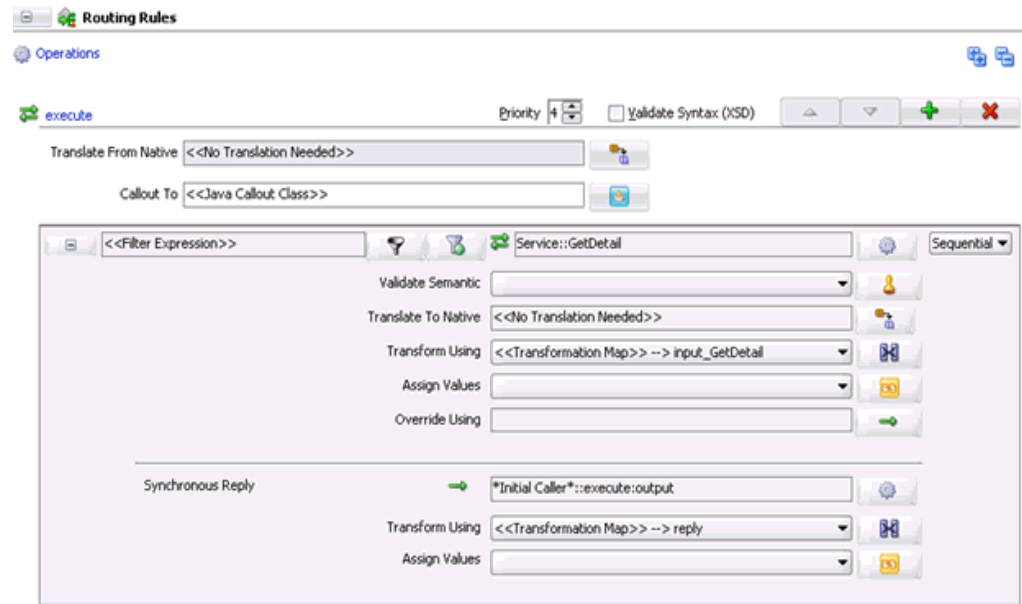
### 7.2.2.3 ルーティング・ルール構成

メディエータ・アウトバウンド・プロセス・コンポーネント用のルーティング・ルールを構成するには、次のようにします。

1. 「コンポーネント」 ペインで、アウトバウンド・メディエータ・プロセス・コンポーネントをダブルクリックします。

図 7-7 に示すように、「ルーティング・ルール」 ダイアログが表示されます。

図 7-7 「ルーティング・ルール」 ダイアログ



2. 「<< フィルタ式 >>」 領域で、「次を使用して変換」 フィールドの右側にあるアイコンをクリックします。

図 7-8 に示すように、「リクエスト・トランスフォーメーション・マップ」 ダイアログが表示されます。

図 7-8 「リクエスト・トランスフォーメーション・マップ」 ダイアログ

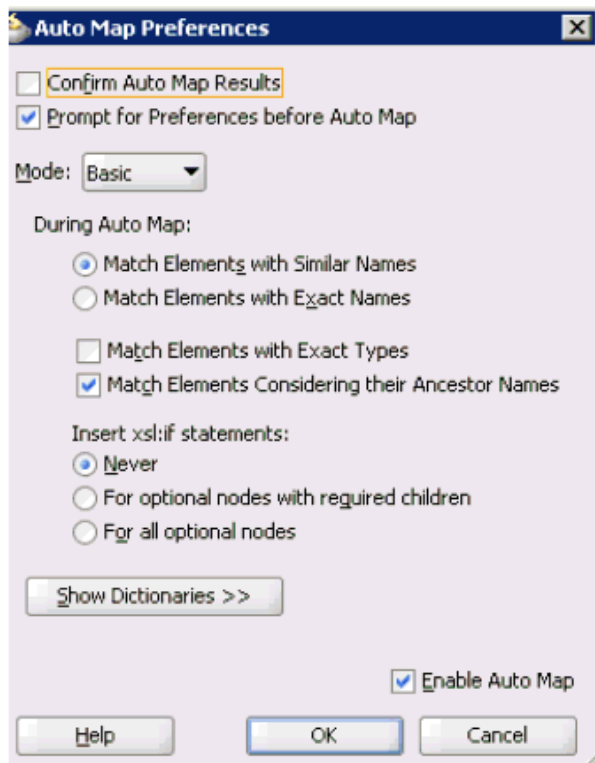


3. 「追加」 (+) アイコンをクリックします。  
「トランスフォーメーション・マップの作成」 ページが表示されます。
4. 選択されている「タイプ」が「XSLT」であることを確認し、「OK」をクリックします。
5. 「OK」をクリックします。

6. `ns0:CompanyCode.GetDetail.Response` ソース要素を  
`ns0:CompanyCode.GetDetail.Response` ターゲット要素にマップします。

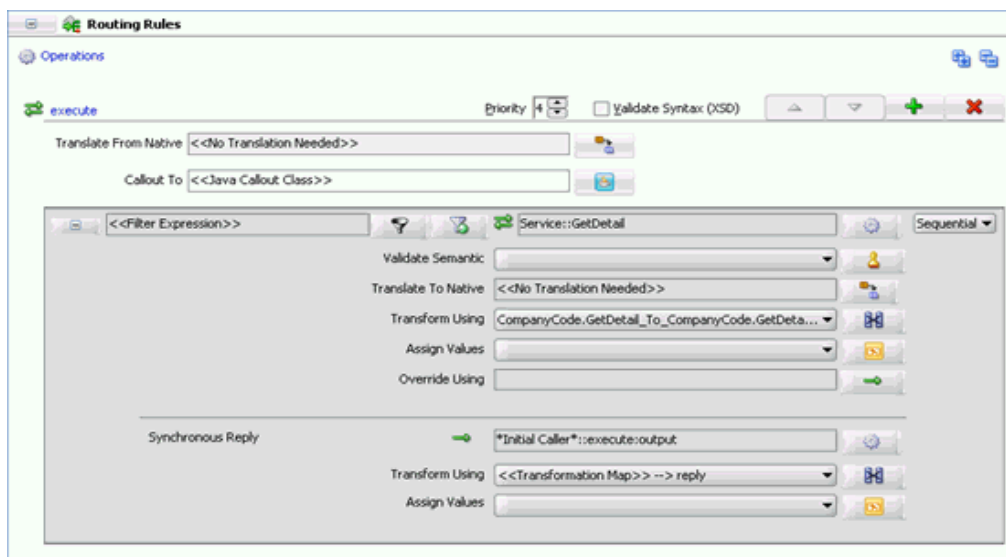
図 7-9 に示すように、「自動マップ・プリファレンス」ダイアログが表示されます。

図 7-9 「自動マップ・プリファレンス」ダイアログ



7. デフォルト値を保持し、「OK」をクリックします。
8. 図 7-10 に示すように、「ルーティング・ルール」ダイアログに戻ります。

図 7-10 「ルーティング・ルール」ダイアログ



- 「同期リプライ」領域で、「次を使用して変換」フィールドの右側にあるアイコンをクリックします。

「リプライ・トランスフォーメーション・マップ」ダイアログが表示されます。

- 「追加」(+) アイコンをクリックします。

トランスフォーメーションの作成ページが表示されます。

- 選択されている「タイプ」が「XSLT」であることを確認し、「OK」をクリックします。

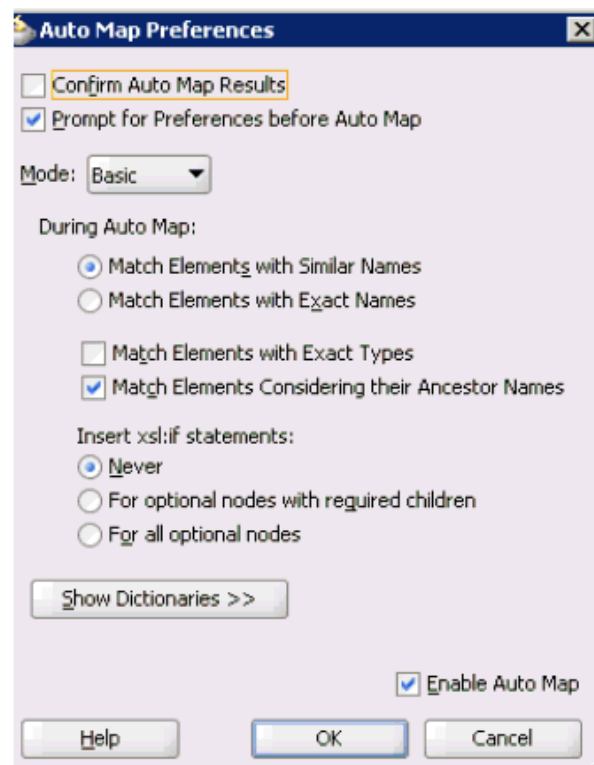
マッピング・ページが表示されます。

- 「OK」をクリックします。

- ns0:CompanyCode.GetDetail.Response** ソース要素を  
**ns0:CompanyCode.GetDetail.Response** ターゲット要素にマップします。

図 7-11 に示すように、「自動マップ・プリファレンス」ダイアログが表示されます。

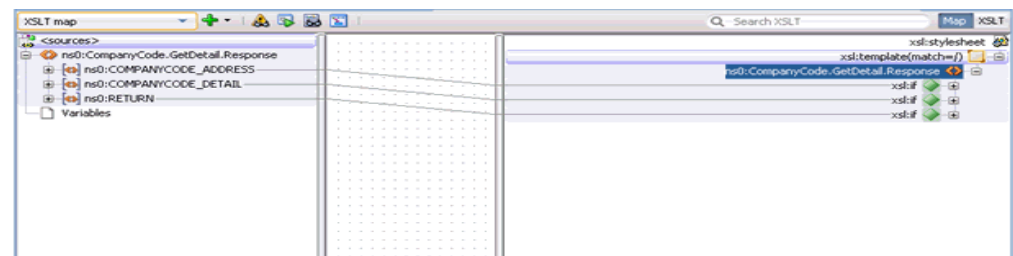
図 7-11 「自動マップ・プリファレンス」ダイアログ



- デフォルト値を保持し、「OK」をクリックします。

図 7-12 に示すように、マッピングが完了しました。

図 7-12 完了したマッピング



15. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド・メディエータ・プロセス・コンポーネントを保存します。

これで、メディエータ・アウトバウンド・プロセスをデプロイする準備ができました。

#### 7.2.2.4 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの 6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」を参照してください。

### 7.2.3 メディエータ・アウトバウンド・プロセスのデプロイ

メディエータ・アウトバウンド・プロセスをデプロイするには、次のようにします。

1. 左ペインのプロジェクト名を右クリックし、「デプロイ」を選択してから「J2CA\_Outbound」をクリックします。  
「デプロイメント・アクション」ページが表示されます。
2. 「アプリケーション・サーバーにデプロイ」が選択されていることを確認します。
3. 「次へ」をクリックします。  
「構成のデプロイ」ページが表示されます。
4. デフォルト値を選択済のままにして、「次へ」をクリックします。  
「サーバーの選択」ページが表示されます。
5. 構成済の使用可能なアプリケーション・サーバーを選択し、「次へ」をクリックします。  
「SOA サーバー」ページが表示されます。
6. ターゲット SOA サーバーを選択し、「次へ」をクリックします。  
「サマリー」ページが表示されます。
7. プロジェクトに関する使用可能なすべてのデプロイメント情報を確認および検証し、「終了」をクリックします。  
詳細は、6-34 ページの 6.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」を参照してください。

### 7.2.4 Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し

詳細は、6-39 ページの 6.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」を参照してください。

## 7.3 メディエータ・インバウンド・プロセスの構成 (J2CA 構成)

この項では、Oracle JDeveloper でメディエータ・プロジェクトを使用して SAP R/3 システムへのメディエータ・インバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\Mediator\J2CA\Inbound_Project
```

ここでは、次の項目について説明します。

- [7.3.1 項「SOA 用の空のコンポジットの作成」](#)
- [7.3.2 項「メディエータ・インバウンド・プロセスの定義」](#)

### 前提条件

メディエータ・インバウンド・プロセスを設計する前に、アプリケーション・エクスポージャーを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-42 ページの「[イベント統合のための WSDL の生成](#)」を参照してください。

### 7.3.1 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

1. 新規 SOA アプリケーションを作成します。
2. 新規 SOA アプリケーションの名前を入力し、「次へ」をクリックします。  
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクト名を入力し、「次へ」をクリックします。  
SOA 設定の構成ページが表示されます。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。  
詳細は、6-11 ページの [6.4.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

### 7.3.2 メディエータ・インバウンド・プロセスの定義

この項では、メディエータ・インバウンド・プロセスを定義する方法について、次の項目で説明します。

- [7.3.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」](#)
- [7.3.2.2 項「ファイル・アダプタを使用したインバウンド・メディエータ・プロセス・コンポーネントの構成」](#)
- [7.3.2.3 項「ルーティング・ルールの構成」](#)
- [7.3.2.4 項「12c でのデプロイメントに関する既知の問題の調整」](#)

#### 7.3.2.1 サード・パーティ・アダプタ・サービス・コンポーネントの構成

サード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

1. 「サード・パーティ・アダプタ」コンポーネントを「サービス・アダプタ」ペインから「公開されたサービス」ペインにドラッグ・アンド・ドロップします。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。
2. サード・パーティ・アダプタ・サービスの名前を入力します。
3. 「タイプ」ドロップダウン・リストで「サービス」が選択されていることを確認します (デフォルト)。

4. 「WSDL URL」フィールドの右側にある「**既存の WSDL を検索します。**」アイコンをクリックします。  
「WSDL の選択」ダイアログが表示されます。
5. 次のディレクトリでインバウンド WSDL ファイルを参照して選択します。  
<ADAPTER\_HOME>\wsdl.s
6. 「OK」をクリックします。  
「ファイルのローカライズ」ダイアログが表示されます。
7. 「OK」をクリックします。  
インバウンド WSDL ファイルと関連する受信 / リクエスト・スキーマ・ファイル (.xsd) が、作成されたプロジェクト・フォルダにインポートされます。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。
8. 「JCA ファイル」フィールドの右側にある「**JCA ファイルを検索します。**」アイコンをクリックします。  
トランスフォーメーション・チューザ・ダイアログが表示されます。
9. 次のディレクトリで JCA プロパティ・ファイルを参照して選択します。  
<ADAPTER\_HOME>\wsdl.s
10. 「OK」をクリックします。  
「ファイルのコピー」確認メッセージが表示されます。
11. 「はい」をクリックします。  
プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。
12. 「OK」をクリックします。  
「公開されたサービス」ペインにサード・パーティ・アダプタ・サービス・コンポーネントが作成されます。  
これで、インバウンド・メディアータ・プロセス・コンポーネントを構成する準備ができました。  
詳細は、6-50 ページの [6.5.3.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの作成」](#) を参照してください。



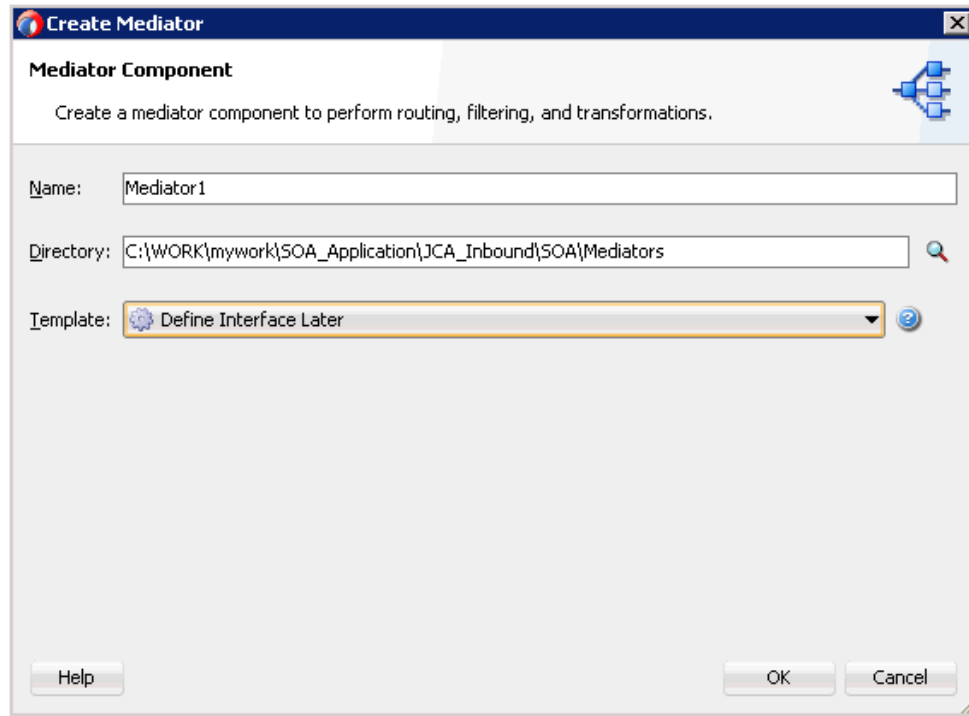
### 7.3.2.2 ファイル・アダプタを使用したインバウンド・メディエータ・プロセス・コンポーネントの構成

ファイル・アダプタを使用してインバウンド・メディエータ・プロセス・コンポーネントを構成するには、次のようにします。

1. **メディエータ・プロセス・コンポーネント**を「サービス・コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

図 7-13 に示すように、「メディエータの作成」ダイアログが表示されます。

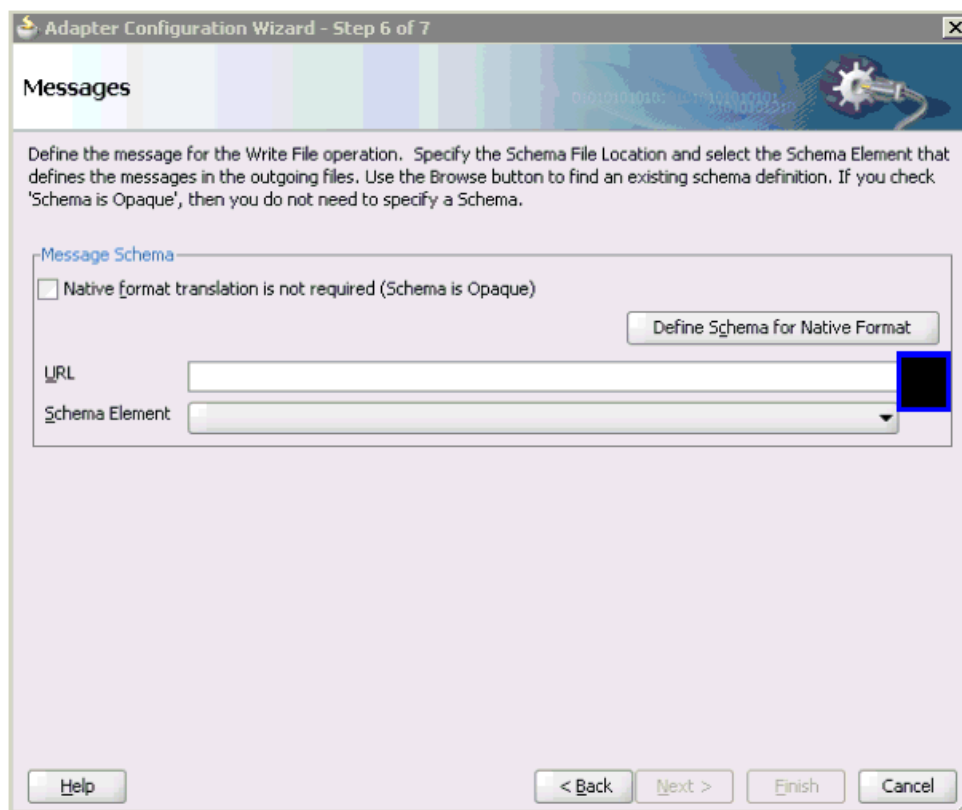
図 7-13 「メディエータの作成」ダイアログ



2. 「名前」フィールドに、新規インバウンド・メディエータ・プロセス・コンポーネントを識別する名前を入力します。
3. 「テンプレート」ドロップダウン・リストで「**インタフェースを後で定義**」を選択します。
4. 「**OK**」をクリックします。  
新規メディエータ・プロセス・コンポーネントが「コンポーネント」ペインに追加されます。
5. 「**ファイル**」コンポーネントを「テクノロジー・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。  
ファイル・アダプタ構成ウィザードが表示されます。
6. 新規ファイル・アダプタの名前を入力し、「**次へ**」をクリックします。  
「アダプタ・インタフェース」ページが表示されます。
7. 「**操作およびスキーマから定義 (後で指定)**」オプションが選択されていることを確認してください。
8. 「**次へ**」をクリックします。  
「操作」ページが表示されます。
9. 「**次へ**」をクリックします。

10. 「操作タイプ」オプションのリストで「ファイルの書込み」を選択し、「操作名」を指定します (Write など)。
11. 「次へ」をクリックします。  
「ファイル構成」ページが表示されます。
12. 出力ファイルを書き込むファイル・システム上の場所を指定します。
13. ファイル・ネーミング規則フィールドで、出力ファイルの名前を指定します。
14. 「次へ」をクリックします。  
図 7-14 に示すように、「メッセージ」ページが表示されます。

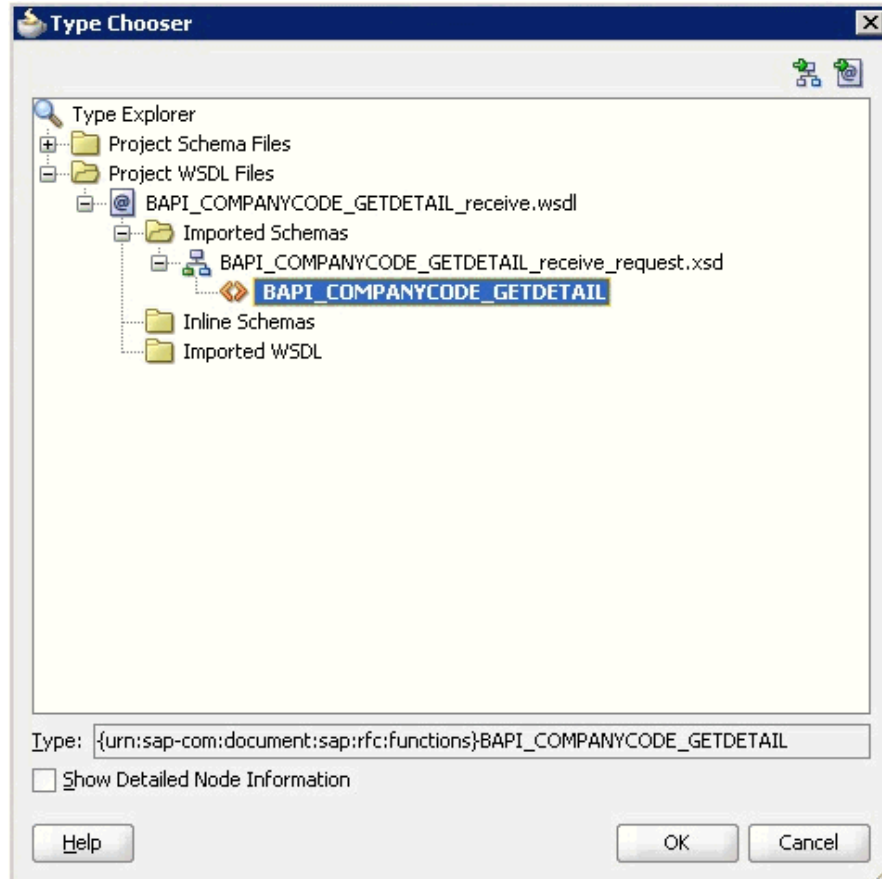
図 7-14 「メッセージ」ページ



15. 「URL」 フィールドの右側にある「参照」をクリックします。

図 7-15 に示すように、「タイプ・チューザ」ダイアログが表示されます。

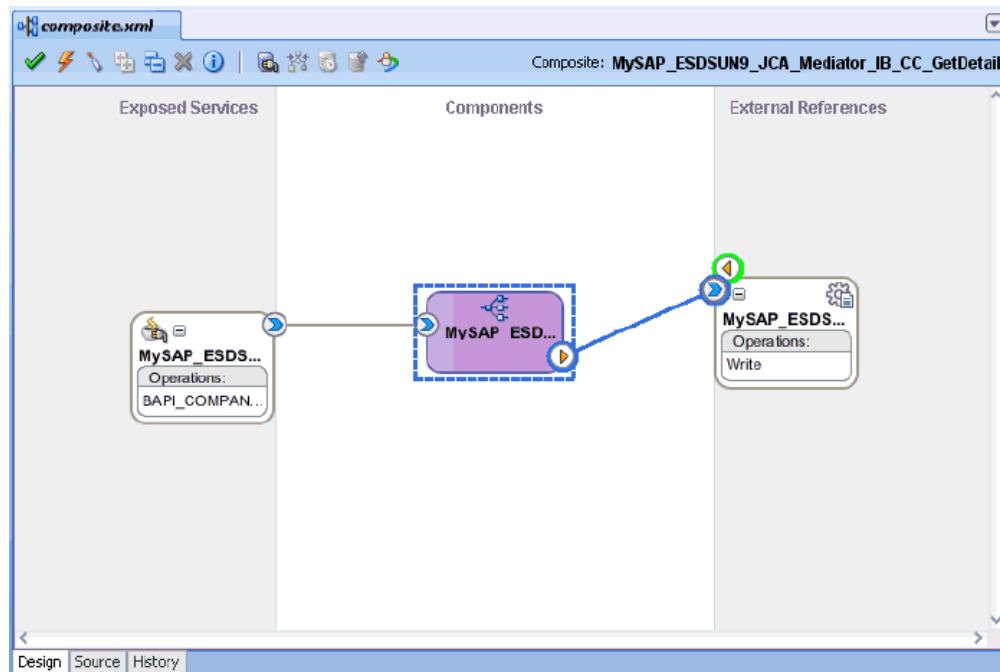
図 7-15 「タイプ・チューザ」ダイアログ



16. 「プロジェクトの WSDL ファイル」 → 「BAPI\_COMPANYCODE\_GETDETAIL\_receive.wSDL」 → 「インポートしたスキーマ」 → 「BAPI\_COMPANYCODE\_GETDETAIL\_receive\_request.xsd」を展開し、「BAPI\_COMPANYCODE\_GETDETAIL」を選択します。
17. 「OK」をクリックします。  
「メッセージ」ページに戻ります。
18. 「次へ」をクリックします。  
「終了」ページが表示されます。
19. 「終了」をクリックします。
20. インバウンド・メディエータ・プロセス・コンポーネントとサード・パーティ・アダプタ・サービス・コンポーネント間の接続を作成します。

21. 図 7-16 に示すように、インバウンド・メディエータ・プロセス・コンポーネントとファイル・アダプタ・コンポーネント間の接続を作成します。

図 7-16 作成された接続



これで、ルーティング・ルールを構成する準備ができました。

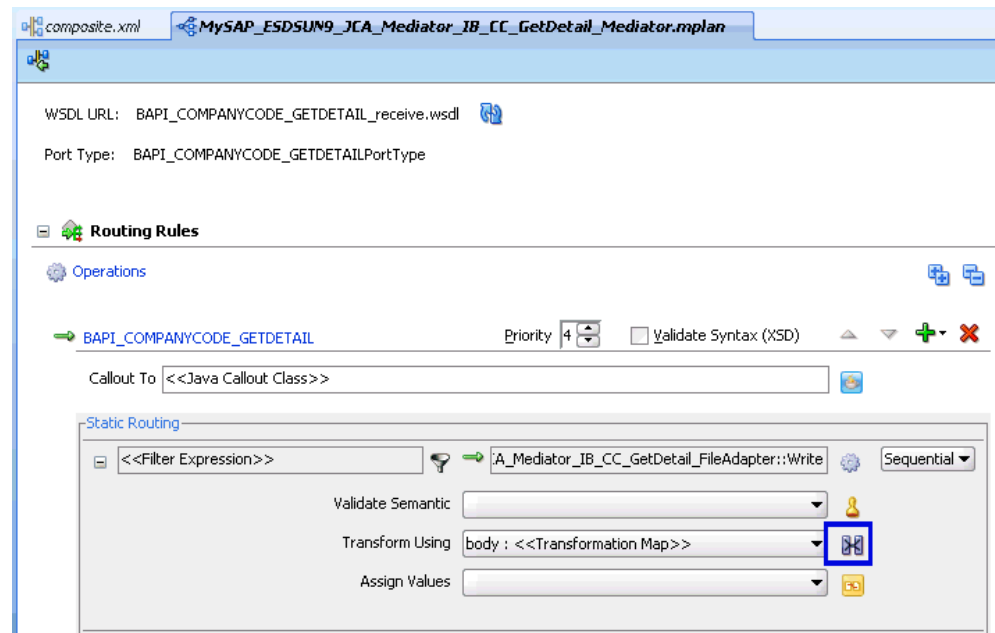
### 7.3.2.3 ルーティング・ルール

メディエータ・インバウンド・プロセス・コンポーネント用のルーティング・ルールを構成するには、次のようにします。

1. 「コンポーネント」 ページで、インバウンド・メディエータ・プロセス・コンポーネントをダブルクリックします。

図 7-17 に示すように、「ルーティング・ルール」 ダイアログが表示されます。

図 7-17 「ルーティング・ルール」 ダイアログ



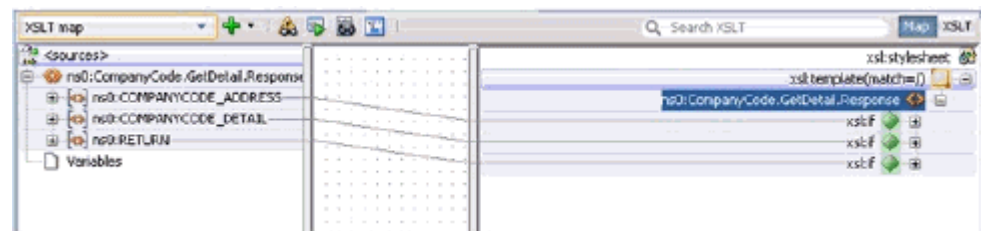
2. 「<<フィルタ式>>」 領域で、「次を使用して変換」 フィールドの右側にあるアイコンをクリックします。

「リクエスト・トランスフォーメーション・マップ」 ダイアログが表示されます。

3. 「追加 (+) アイコンをクリックし、選択されている「タイプ」が「XSLT」であることを確認して、「OK」をクリックします。
4. 「OK」をクリックします。

図 7-18 に示すように、マッピング・ページが表示されます。

図 7-18 マッピング・ページ



5. 「OK」をクリックします。
6. ns0:BAPI\_COMPANYCODE\_GETDETAIL ソース要素を ns0:BAPI\_COMPANYCODE\_GETDETAIL ターゲット要素にマップします。

「自動マップ・プリファレンス」 ダイアログが表示されます。

7. デフォルト値を保持し、「OK」をクリックします。  
これでマッピングが完了しました。
8. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規インバウンド・メディエータ・プロセス・コンポーネントを保存します。

#### 7.3.2.4 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの [6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」](#) を参照してください。

これで、メディエータ・インバウンド・プロセスをデプロイする準備ができました。6-59 ページの「[BPEL インバウンド・プロセスのデプロイ](#)」と同じ手順を実行できます。

SAP GUI を通じてイベント・メッセージがトリガーされると、ファイル・アダプタ・コンポーネント用に指定された場所へ出力 XML が受信されます。SAP R/3 でのイベントのトリガーの詳細は、6-60 ページの「[SAP R/3 でのイベントのトリガー](#)」を参照してください。

## 7.4 メディエータ・アウトバウンド・プロセスの構成 (BSE 構成)

この項では、Oracle JDeveloper でメディエータ・プロジェクトを使用して SAP R/3 システムへのメディエータ・アウトバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\Mediator\BSE\Outbound_Project
```

ここでは、次の項目について説明します。

- [7.4.1 項「SOA 用の空のコンポジットの作成」](#)
- [7.4.2 項「メディエータ・アウトバウンド・プロセスの定義」](#)

#### 前提条件

メディエータ・アウトバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、[6.7.1 項「Web サービスを使用したリクエストおよびレスポンス・サービスの WSDL ファイルの生成」](#) を参照してください。

### 7.4.1 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

1. 新規 SOA アプリケーションを作成します。
2. SOA アプリケーションの名前を入力し (IBSE\_OUTBOUND など)、「次へ」をクリックします。
3. プロジェクト名を入力し (IBSE\_Outbound など)、「次へ」をクリックします。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

詳細は、6-72 ページの [6.7.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

## 7.4.2 メディエータ・アウトバウンド・プロセスの定義

この項では、メディエータ・アウトバウンド・プロセスを定義する方法について説明します。ここでは、次の項目について説明します。

- 7.4.2.1 項「SOAP サービスの構成」
- 7.4.2.2 項「メディエータ・コンポーネントの作成」
- 7.4.2.3 項「ルーティング・ルールの構成」

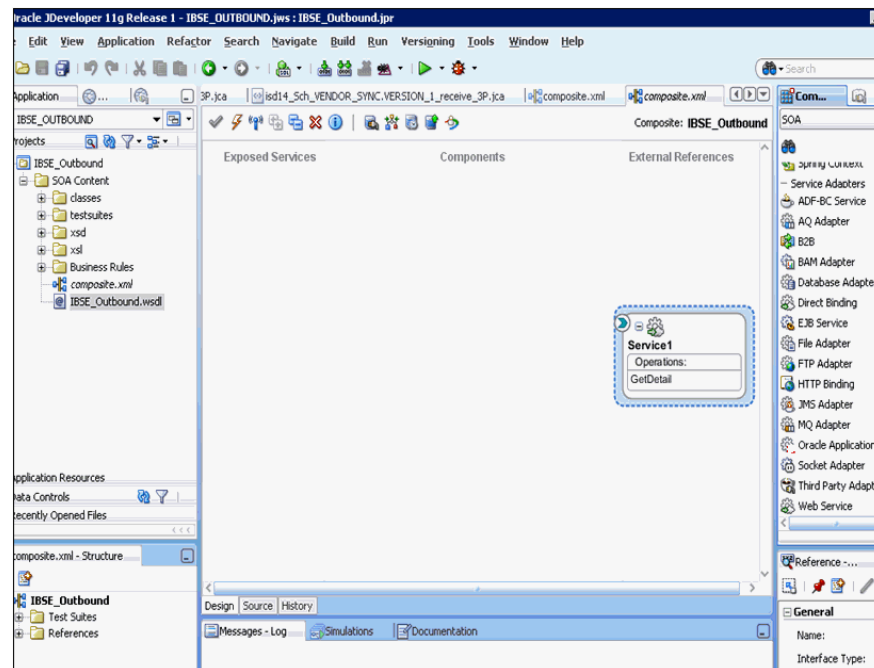
### 7.4.2.1 SOAP サービスの構成

SOAP サービスを構成するには、次のようにします。

1. 「SOAP」ノードを「テクノロジー・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。
2. SOAP サービスの適切な名前を入力し、「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。
3. 表示される「SOA リソース・ブラウザ」ウィンドウで「ファイルシステム」タブを選択し、アプリケーション・エクスプローラから WSDL がエクスポートされる場所に移動して WSDL を選択し、「OK」をクリックします。
4. 「Web サービスの作成」ウィンドウで、「OK」をクリックします。
5. 表示される「ファイルのローカライズ」ウィンドウで、「OK」をクリックします。これにより、WSDL ファイルがプロジェクト・フォルダにインポートされます。

図 7-19 に示すように、Web サービスが作成されて表示されます。

図 7-19 作成された Web サービス

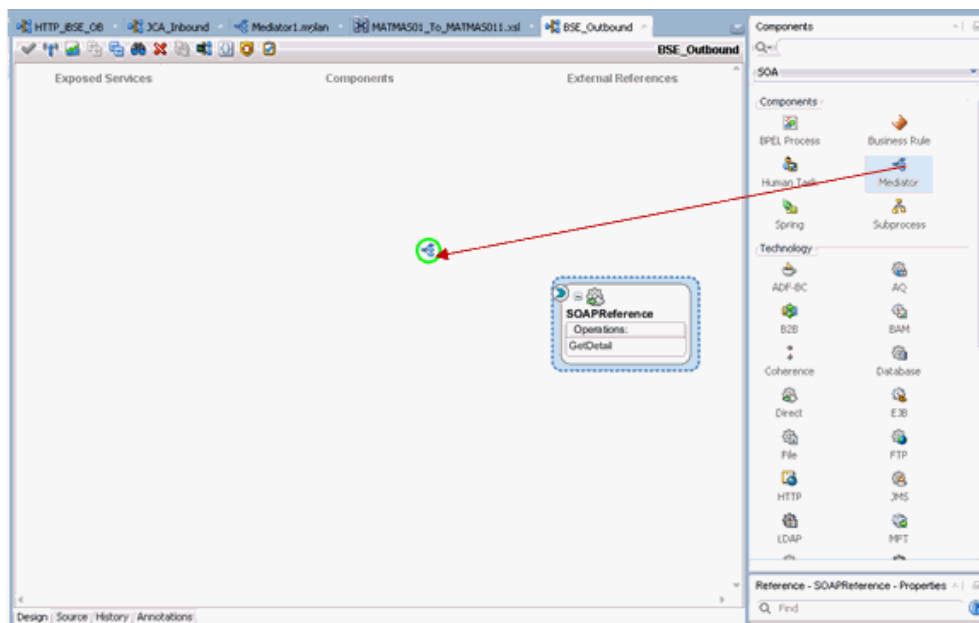


### 7.4.2.2 メディエータ・コンポーネントの作成

メディエータ・コンポーネントを作成するには、次のようにします。

1. 図 7-20 に示すように、「メディエータ」コンポーネントを「コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

図 7-20 「メディエータ」コンポーネント・ダイアログ

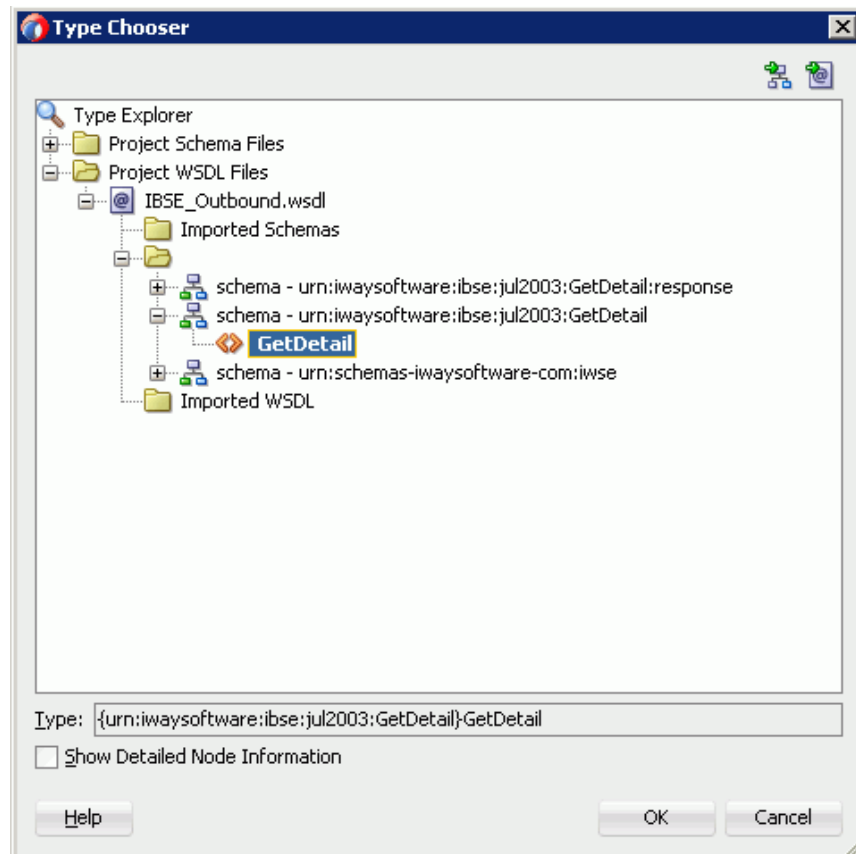


2. 「名前」フィールドに、新規アウトバウンド・メディエータ・プロセス・コンポーネントを識別する名前を入力します。
3. 「テンプレート」ドロップダウン・リストで「同期インタフェース」を選択します。
4. 「入力」フィールドの右側にある「参照」アイコンをクリックし、関連する XML リクエスト・スキーマ・ファイルを選択します。



5. 図 7-21 に示すように、「タイプ・チューザ」ダイアログで、「プロジェクトの WSDL ファイル」を展開して「IBSE\_Outbound.wsdl」を選択し、「GetDetail」をクリックします。

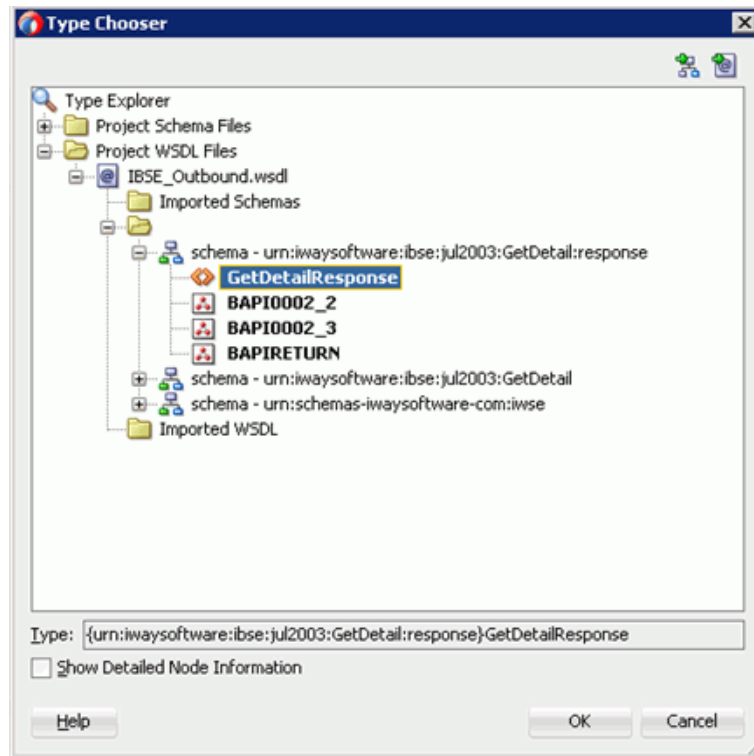
図 7-21 「タイプ・チューザ」ダイアログ



6. 「OK」をクリックします。
7. 「出力」フィールドの右側にある「参照」アイコンをクリックし、関連する XML レスポンス・スキーマ・ファイルを選択します。

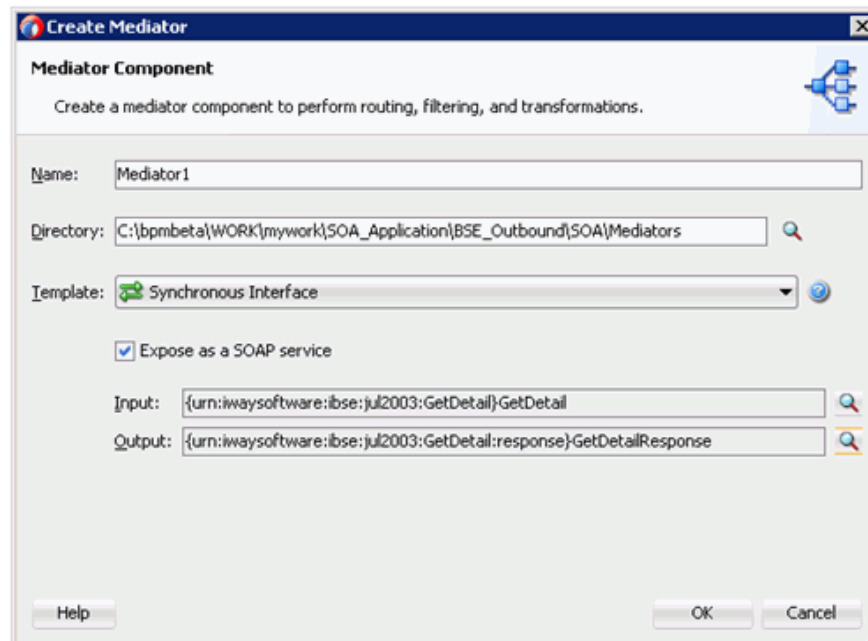
8. 図 7-22 に示すように、「タイプ・チューザ」ダイアログで、「プロジェクトの WSDL ファイル」を展開して「IBSE\_Outbound.wsdl」を選択し、「GetDetailResponse」をクリックします。

図 7-22 「タイプ・チューザ」ダイアログ



9. 図 7-23 に示すように、「OK」をクリックします。

図 7-23 「メディエータの作成」ダイアログ

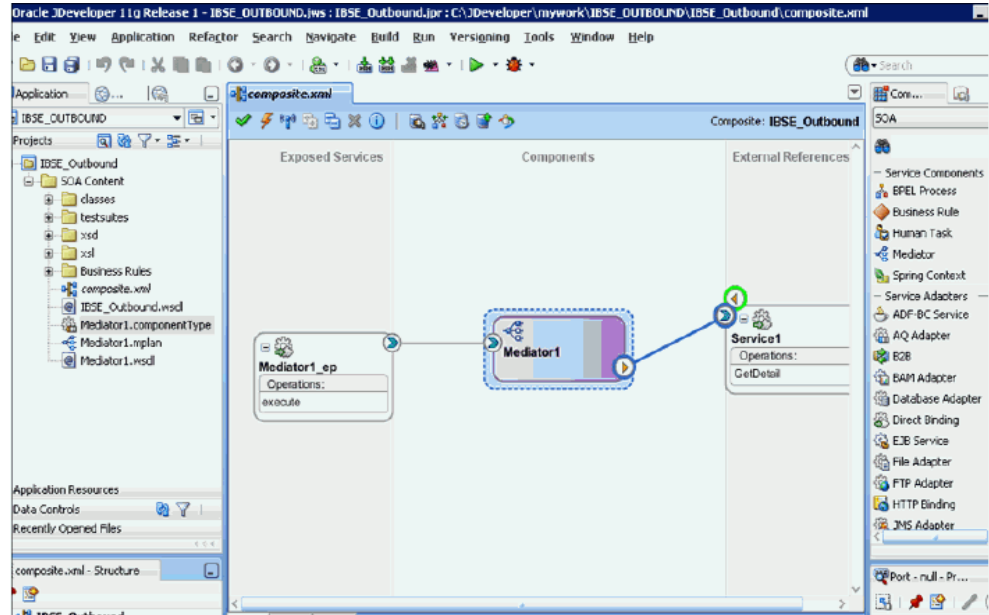


10. 「OK」をクリックします。

メディエータ・コンポーネントが作成されて表示されます。

11. 図 7-24 に示すように、「メディエータ」コンポーネントと「SOAP サービス」コンポーネント間の接続を作成します。

図 7-24 作成された接続

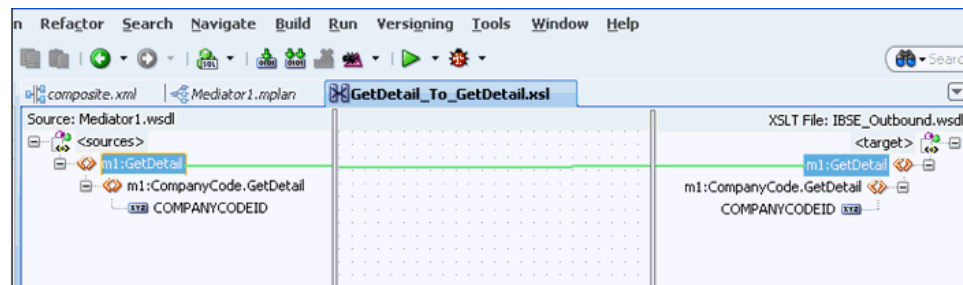


### 7.4.2.3 ルーティング・ルールの構成

ルーティング・ルールを構成するには、次のようにします。

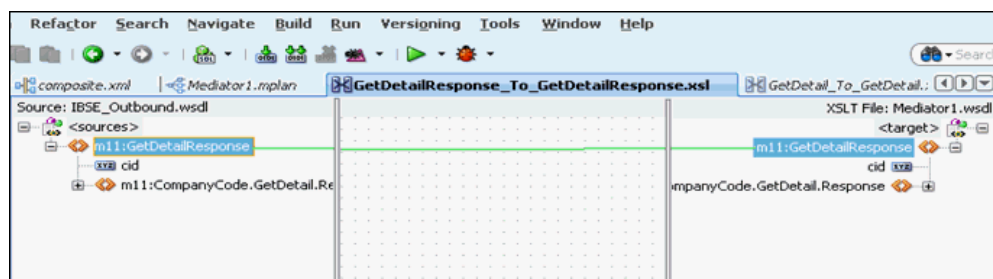
1. 「コンポーネント」ペインで、「メディエータ」コンポーネントをダブルクリックします。
2. 「静的ルーティング」セクションの「<<フィルタ式>>」領域で、「次を使用して変換」フィールドの右側にあるアイコンをクリックします。
3. 表示される「リクエスト・トランスフォーメーション・マップ」ウィンドウで「追加」(+) アイコンをクリックし、「トランスフォーメーション・マップの作成」ダイアログ・ボックスで、選択されている「タイプ」が「XSLT」であることを確認して「OK」をクリックします。
4. 「OK」をクリックします。
5. 図 7-25 に示すように、ns0:GetDetail ソース要素を ns0:GetDetail ターゲット要素にマップします。

図 7-25 「GetDetail\_To\_GetDetail.xsl」タブ



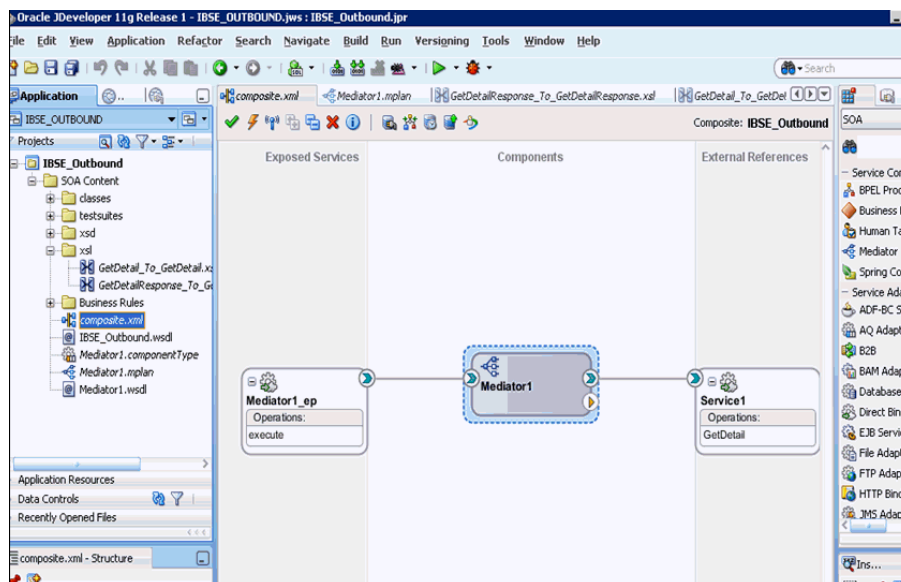
6. 表示される「自動マップ・プリファレンス」ウィンドウで、デフォルト値を保持して「OK」をクリックします。
7. 「同期リプライ」領域で、「次を使用して変換」フィールドの右側にあるアイコンをクリックします。
8. 表示される「リプライ・トランスフォーメーション・マップ」ウィンドウで「追加」(+ )アイコンをクリックし、「トランスフォーメーション・マップの作成」ダイアログ・ボックスで、選択されている「タイプ」が「XSLT」であることを確認して「OK」をクリックします。
9. 図 7-26 に示すように、**ns0:GetDetailResponse** ソース要素を **ns0:GetDetailResponse** ターゲット要素にマップします。

図 7-26 ソース要素とターゲット要素



10. 表示される「自動マップ・プリファレンス」ウィンドウで、デフォルト値を保持して「OK」をクリックします。
11. 図 7-27 に示すように、メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド・メディエータ・コンポーネントを保存します。

図 7-27 「すべて保存」アイコン



これで、メディエータ IBSE アウトバウンド・プロセスをデプロイする準備ができました。7.2.3 項「メディエータ・アウトバウンド・プロセスのデプロイ」と同じ手順を実行できます。

デプロイが完了したら、7.2.4 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」の説明に従って入力 XML を呼び出すことができます。

---

---

## Oracle SOA Suite での BPM サービス・コンポーネントとの統合

Oracle Application Adapter for SAP R/3 は Business Process Management (BPM) とシームレスに統合され、これにより Web サービス統合が簡略化されます。Oracle BPM は、サービス指向アーキテクチャ (SOA) に基づいています。これは Web Service Definition Language (WSDL) ドキュメントとして公開されているアダプタ・サービスを消費します。

この章は次の項で構成されています。

- 8.1 項「概要」
- 8.2 項「アダプタのデプロイメント」
- 8.3 項「新規アプリケーション・サーバー接続の構成」
- 8.4 項「トランスフォーメーションを使用したサービス統合のためのアウトバウンド BPM プロセスの設計 (J2CA 構成)」
- 8.5 項「トランスフォーメーションを使用したイベント統合のためのインバウンド BPM プロセスの設計 (J2CA 構成)」
- 8.6 項「BPM を使用した同期 SAP R/3 イベントの処理 (J2CA 構成)」
- 8.7 項「トランスフォーメーションを使用したサービス統合のためのアウトバウンド BPM プロセスの設計 (BSE 構成)」

### 8.1 概要

Oracle BPM と統合するには、Oracle Application Adapter for SAP R/3 は、Oracle BPM と同じ WLS コンテナにデプロイされている必要があります。基礎となるアダプタ・サービスは WSDL ファイルとして公開されている必要があり、これらのファイルは、アダプタのリクエスト/レスポンス (アウトバウンド) とイベント通知 (インバウンド) の両方のサービスについて、Oracle Adapter Application Explorer (アプリケーション・エクスプローラ) でデザインタイムに生成されます。詳細は、4-15 ページの「WSDL の生成 (J2CA 構成のみ)」を参照してください。

生成された WSDL ファイルは、インバウンドまたはアウトバウンドのアダプタ・サービスに適した BPM プロセスを設計するために使用されます。完成した BPM プロセスは、JDeveloper で正常にコンパイルされ、BPM サーバーにデプロイされる必要があります。BPM サーバーへのデプロイメント時に、新しく作成されたプロセスはすべて自動的に Oracle Enterprise Manager コンソールにデプロイされます。このコンソールで BPM プロセスの実行、監視および管理や、アダプタ・イベントのリスニングを行います。

## 8.2 アダプタのデプロイメント

インストール時に、Oracle Application Adapter for SAP R/3 は、J2CA 1.0 リソース・アダプタとして WLS コンテナ内にデプロイされます。このアダプタは Oracle BPM と同じ WLS コンテナにデプロイされる必要があります。

## 8.3 新規アプリケーション・サーバー接続の構成

Oracle JDeveloper で新規アプリケーション・サーバー接続を構成する方法の詳細は、6.3 項「新規アプリケーション・サーバー接続の構成」を参照してください。

## 8.4 トランスフォーメーションを使用したサービス統合のためのアウトバウンド BPM プロセスの設計 (J2CA 構成)

この項では、サービス統合のためにトランスフォーメーションを使用してアウトバウンド BPM プロセスを設計する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPM\J2CA\Outbound_Project
```

アウトバウンド・デザインタイム構成を完了するには、次のツールが必要です。

- Oracle Adapter Application Explorer (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPM Designer (JDeveloper)

---

**注意：** この章の例では、JDeveloper を使用しています。

---

ここでは、次の項目について説明します。

- 8.4.1 項「BPM 用の空のコンポジットの作成」
- 8.4.2 項「BPM アウトバウンド・プロセスの定義」
- 8.4.3 項「12c でのデプロイメントに関する既知の問題の調整」
- 8.4.4 項「BPM アウトバウンド・プロセスのデプロイ」
- 8.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」

BPM プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-9 ページの「リクエスト/レスポンス・サービスの WSDL の生成」を参照してください。

### 8.4.1 BPM 用の空のコンポジットの作成

BPM 用の空のコンポジットを作成するには、次のようにします。

1. 新規 BPM アプリケーションを作成します。
2. 新規 BPM アプリケーションの名前を入力し、「次へ」をクリックします。  
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクト名を入力し、プロジェクト機能で「BPM」を選択して、「次へ」をクリックします。  
SOA 設定の構成ページが表示されます。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

## 8.4.2 BPM アウトバウンド・プロセスの定義

この項では、BPM アウトバウンド・プロセスを定義する方法について、次の項目で説明します。

- 8.4.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」
- 8.4.2.2 項「アウトバウンド BPM プロセス・コンポーネントの構成」
- 8.4.2.3 項「書込み操作のファイル・アダプタの作成」

### 8.4.2.1 サード・パーティ・アダプタ・サービス・コンポーネントの構成

サード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

1. 作成されたプロジェクトをダブルクリックして、コンポーネントをロードします。
2. 「サード・パーティ・アダプタ」コンポーネントを「カスタム/サード・パーティ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。
3. サード・パーティ・アダプタ・サービスの名前を入力します。
4. 「タイプ」リストで「参照」が選択されていることを確認します(デフォルト)。
5. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。  
「WSDL の選択」ダイアログが表示されます。
6. 次のディレクトリでアウトバウンド WSDL ファイルを参照して選択します。  
<ADAPTER\_HOME>\wsdls
7. 「OK」をクリックします。  
「ファイルのローカライズ」ダイアログが表示されます。
8. 「OK」をクリックします。  
アウトバウンド WSDL ファイル、および関連するリクエストとレスポンスの XML スキーマ・ファイル(.xsd)が、作成されたプロジェクト・フォルダにインポートされます。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。
9. 「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。  
トランスフォーメーション・マップ・ダイアログが表示されます。
10. 次のディレクトリで JCA プロパティ・ファイルを参照して選択します。  
<ADAPTER\_HOME>\wsdls
11. 「OK」をクリックします。  
「ファイルのコピー」メッセージが表示されます。
12. 「はい」をクリックします。  
プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。
13. 「OK」をクリックします。  
サード・パーティ・アダプタ・サービス・コンポーネントが作成され、「外部参照」ペインに表示されます。  
これで、アウトバウンド BPM プロセス・コンポーネントを構成する準備ができました。  
スクリーン・ショットなどの詳細は、6-14 ページの 6.4.3.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」を参照してください。

### 8.4.2.2 アウトバウンド BPM プロセス・コンポーネントの構成

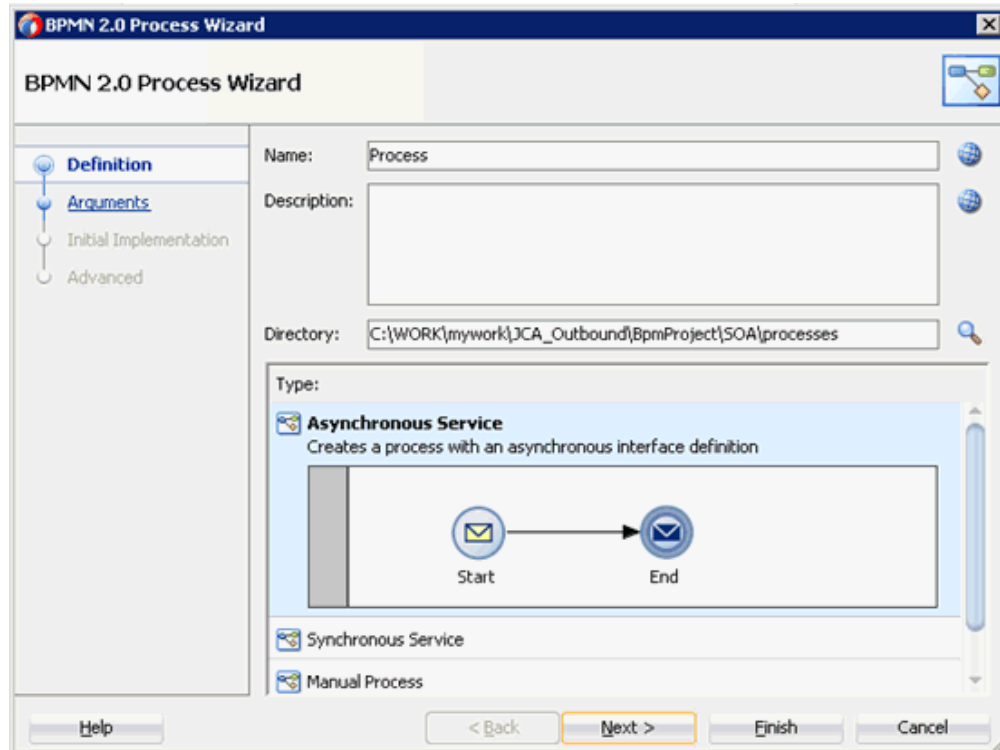
この項では、アウトバウンド BPM プロセス・コンポーネントを構成する方法について説明します。

アウトバウンド BPM プロセス・コンポーネントを構成するには、次のようにします。

1. 「BPMN プロセス」コンポーネントを「コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

☒ 8-1 に示すように、BPMN プロセスの作成のダイアログが表示されます。

図 8-1 BPMN プロセスの作成のダイアログ

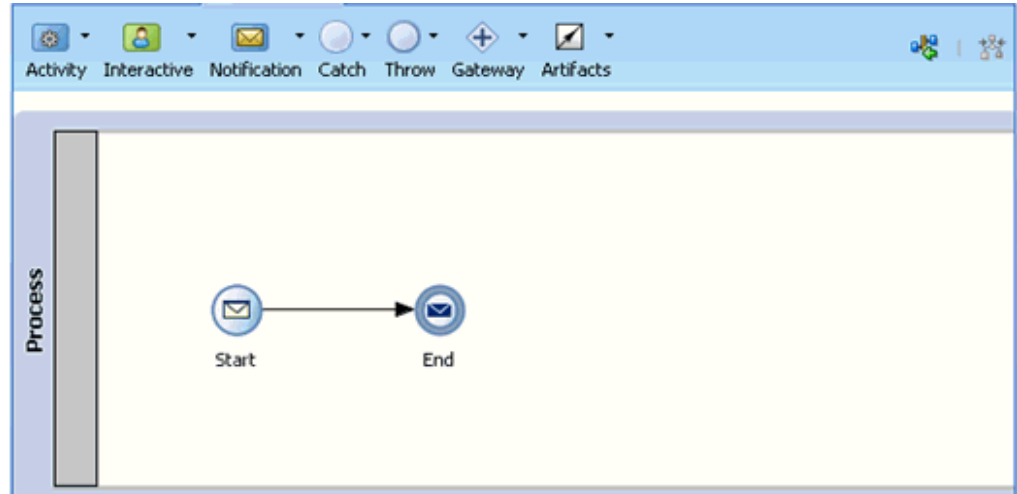




2. 「タイプ」領域で選択されているデフォルト・オプション (非同期サービス) を受け入れ、「終了」をクリックします。

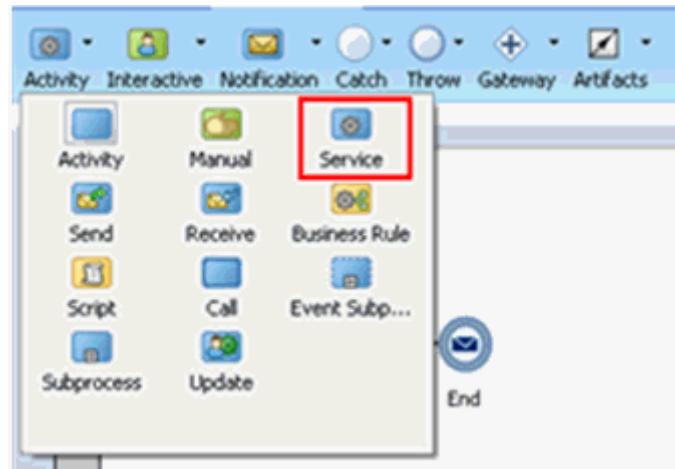
図 8-2 に示すように、BPMN プロセスが表示されます。

図 8-2 BPMN プロセス



3. 図 8-3 に示すように、「アクティビティ」ドロップダウン・メニューをクリックして「サービス」を選択します。

図 8-3 「アクティビティ」ドロップダウン・メニュー



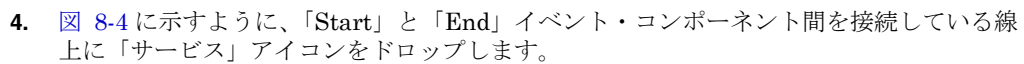
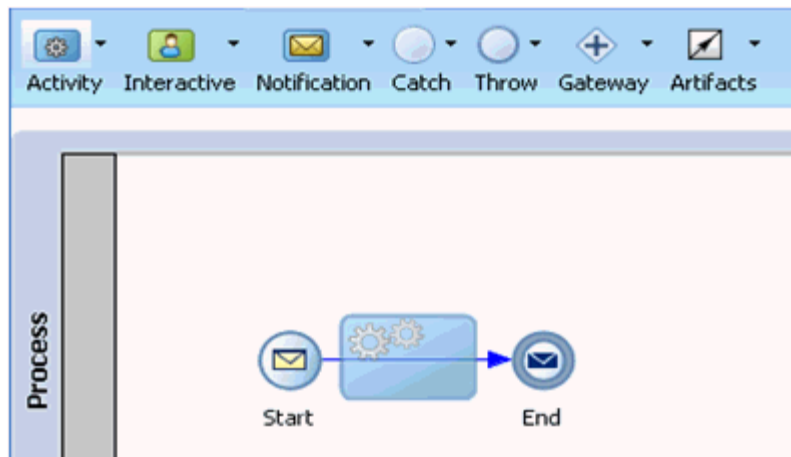
4.  図 8-4 に示すように、「Start」と「End」イベント・コンポーネント間を接続している線の上に「サービス」アイコンをドロップします。

図 8-4 「アクティビティ」アイコン



「プロパティ - ServiceTask」ウィンドウが表示されます。

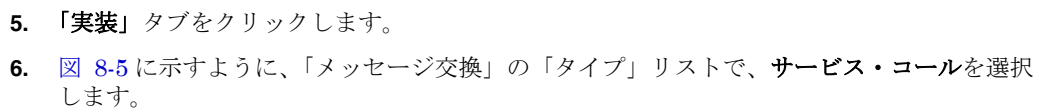
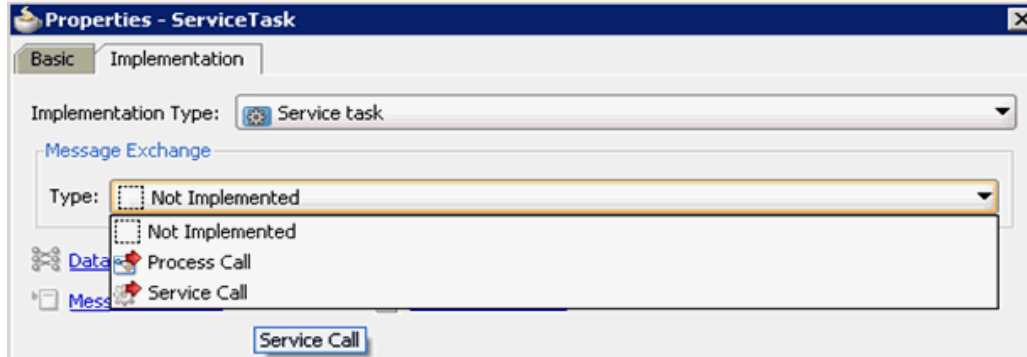
5. 「実装」タブをクリックします。
6.  図 8-5 に示すように、「メッセージ交換」の「タイプ」リストで、サービス・コールを選択します。

図 8-5 サービス・コール



7. 図 8-6 に示すように、「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 8-6 「参照」アイコン

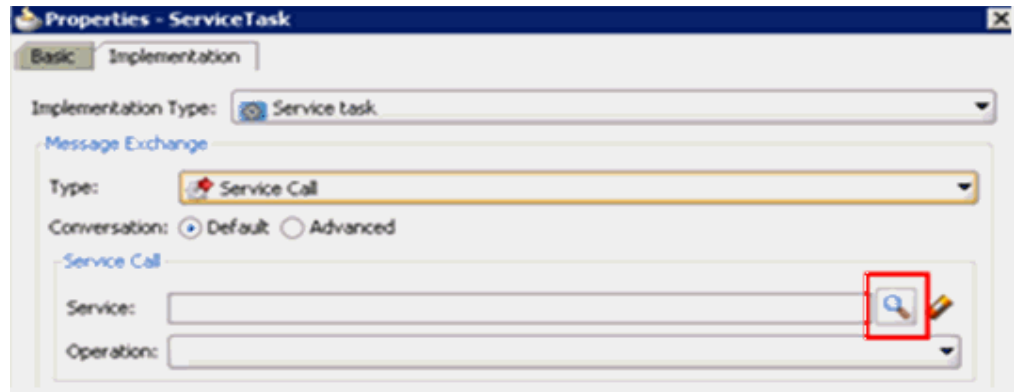


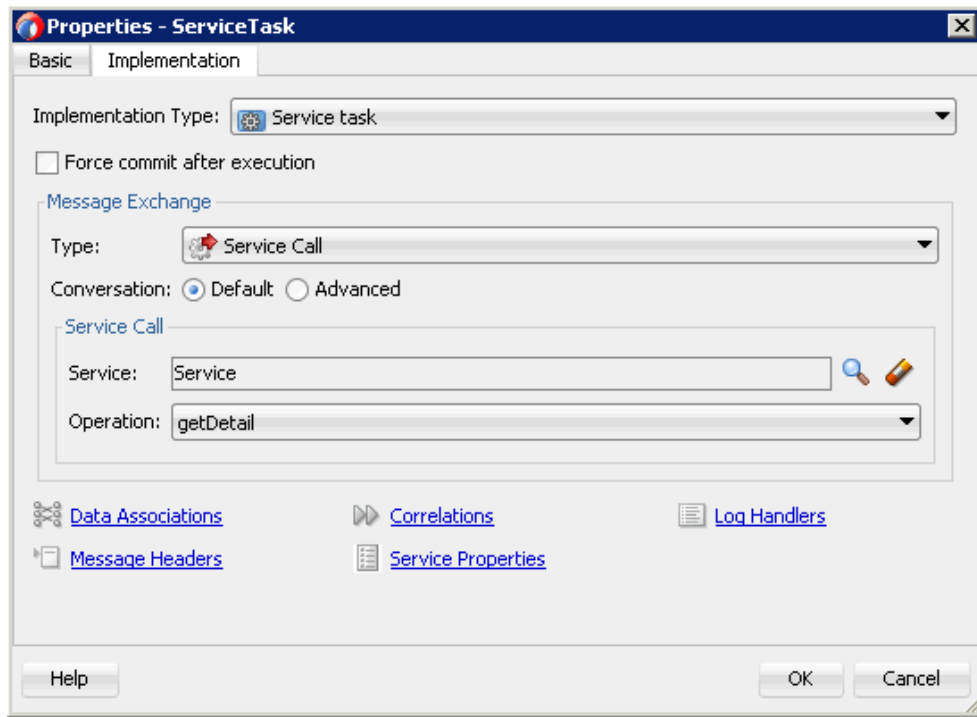
図 8-7 に示すように、「サービス」ダイアログが表示されます。

図 8-7 「サービス」ダイアログ



8. 作成したサード・パーティ・サービスを選択し、「OK」をクリックします。  
☒ 8-8 に示すように、「プロパティ - ServiceTask」ダイアログに戻ります。

図 8-8 「プロパティ - ServiceTask」ダイアログ



9. データ・アソシエーション・ハイパーリンクをクリックします。  
データ・アソシエーション・ダイアログが表示されます。
10. ☒ 8-9 に示すように、左ペインで、「プロセス」の下の「データ・オブジェクト」ノードを右クリックし、「新規」を選択します。

図 8-9 「新規」オプション

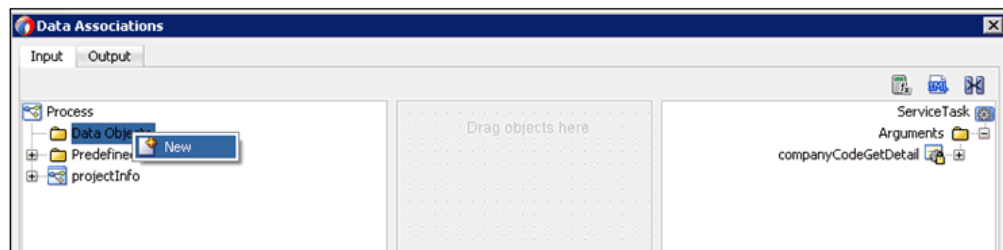
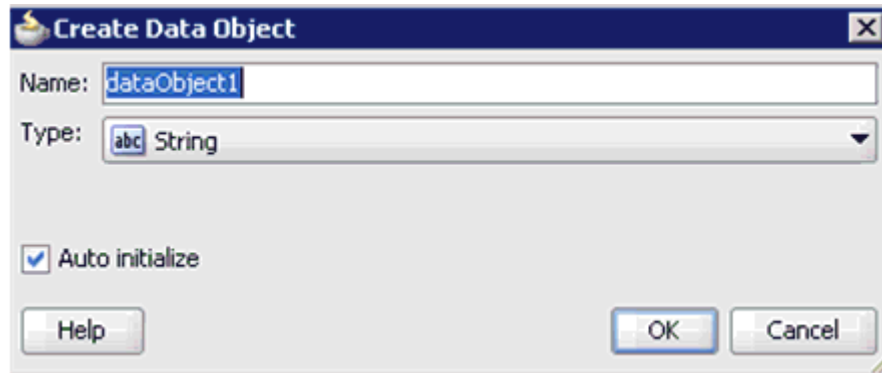


図 8-10 に示すように、データ・オブジェクトの作成ダイアログが表示されます。

図 8-10 データ・オブジェクトの作成ダイアログ



11. 図 8-11 に示すように、「名前」フィールドに名前を入力し (Request など)、「タイプ」フィールドのドロップダウン・ボタンをクリックして、リストで「参照」を選択します。

図 8-11 データ・オブジェクトの作成ダイアログ

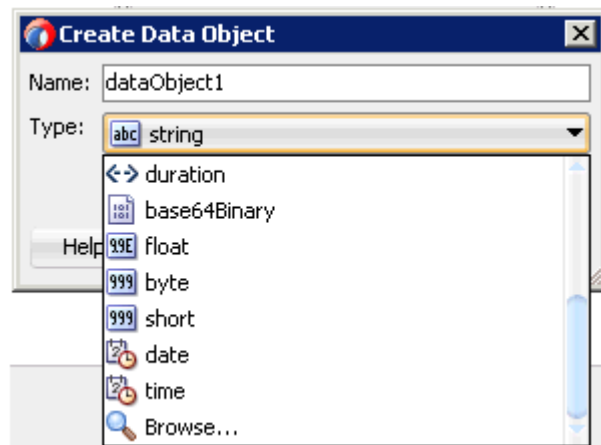
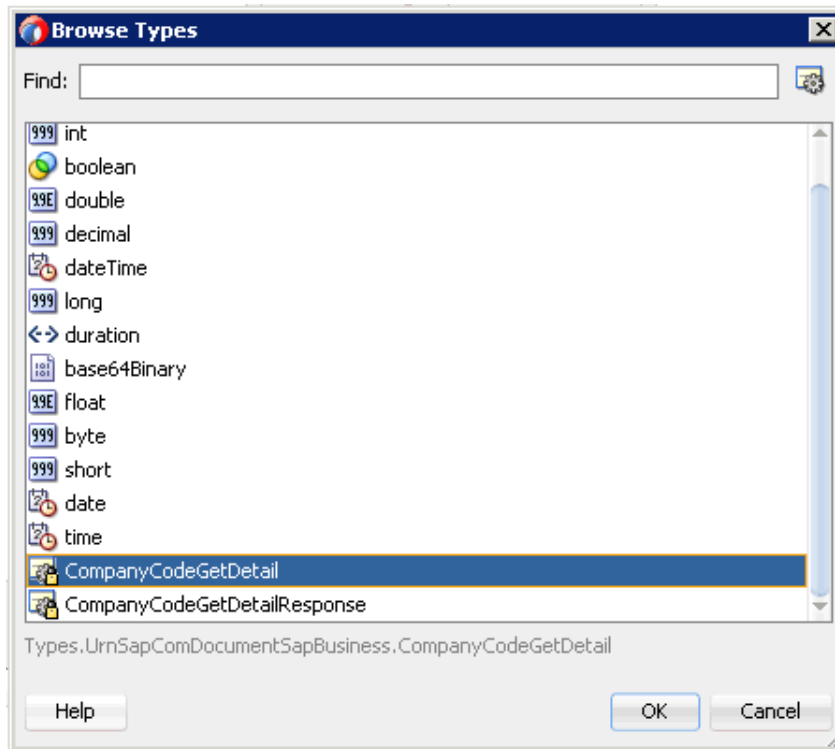


図 8-12 に示すように、「タイプの参照」ダイアログが表示されます。

図 8-12 「タイプの参照」ダイアログ



- 最初のコンポーネント (CompanyCodeGetDetail など) を選択し、「OK」をクリックします。

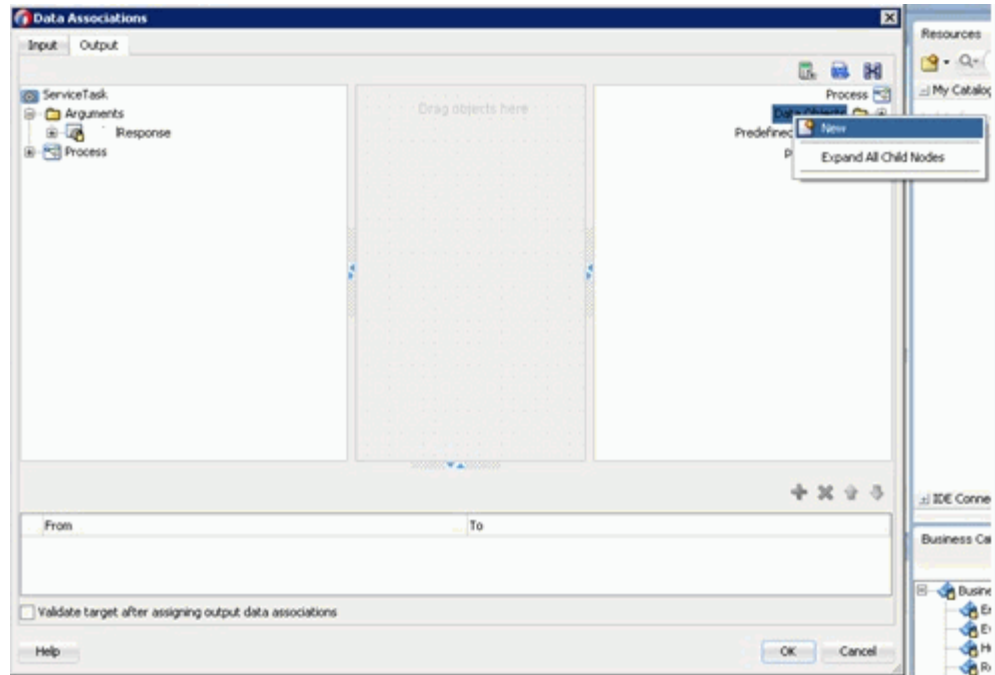
データ・オブジェクトの作成ダイアログに戻ります。

- 「OK」をクリックします。

作成したデータ・オブジェクト (Request など) が、データ・アソシエーション・ダイアログの「データ・オブジェクト」ノードの下に表示されます。

14. 図 8-13 に示すように、「出力」タブの右ペインにある「データ・オブジェクト」ノードを右クリックして「新規」を選択し、別のデータ・オブジェクトを作成します。

図 8-13 データ・アソシエーション・ダイアログ

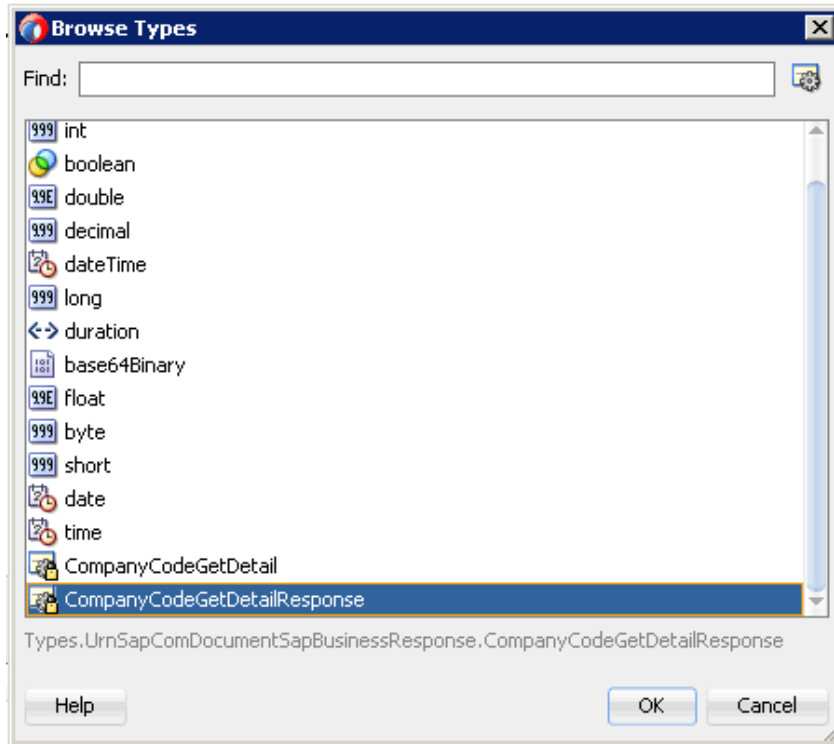


データ・オブジェクトの作成ダイアログが表示されます。

15. 「名前」フィールドに名前を入力し (Response など)、「タイプ」フィールドのドロップダウン・ボタンをクリックして、リストで「参照」を選択します。

図 8-14 に示すように、「タイプの参照」ダイアログが表示されます。

図 8-14 「タイプの参照」ダイアログ



16. 2 番目のコンポーネント (CompanyCodeGetDetailResponse など) を選択し、「OK」をクリックします。

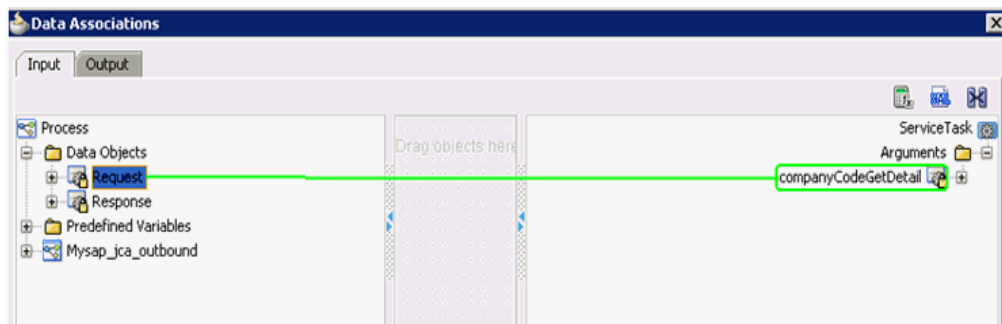
データ・オブジェクトの作成ダイアログに戻ります。

17. 「OK」をクリックします。

作成したデータ・オブジェクト (Response など) が、データ・アソシエーション・ダイアログの「プロセス」ノードの下に表示されます。

18. 図 8-15 に示すように、「入力」タブの左ペインで、「データ・オブジェクト」ノードの下の「Request」データ・オブジェクトを選択してドラッグし、右ペインの「引数」ノードの下の「CompanyCodeGetDetail」に接続します。

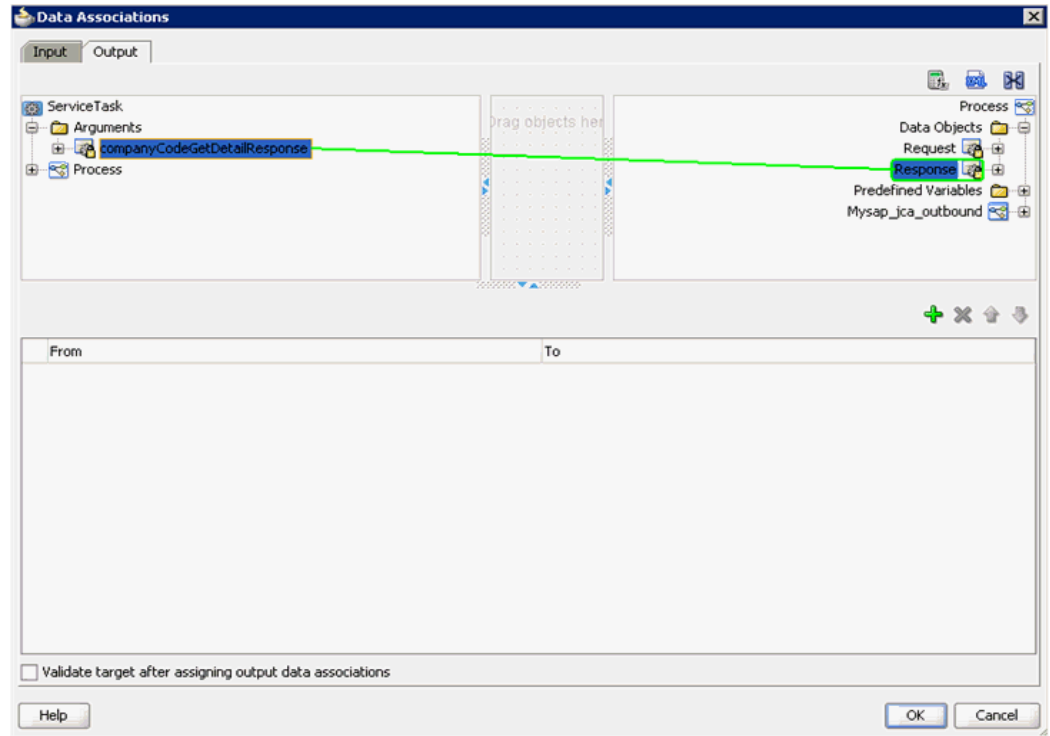
図 8-15 「Request」データ・オブジェクト





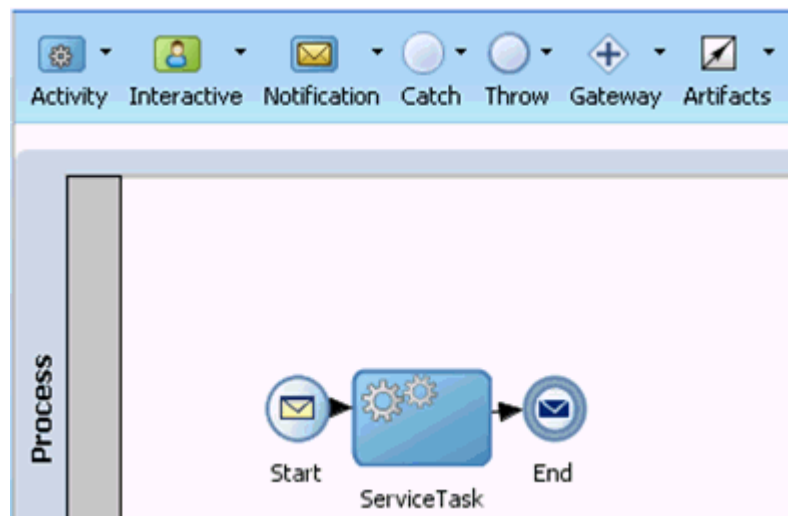
19. 図 8-16 に示すように、「出力」タブをクリックし、左ペインで「引数」ノードの下の「CompanyCodeGetDetailResponse」を選択してドラッグし、「データ・オブジェクト」ノードの下の「Response」データ・オブジェクトに接続します。

図 8-16 「Response」データ・オブジェクト



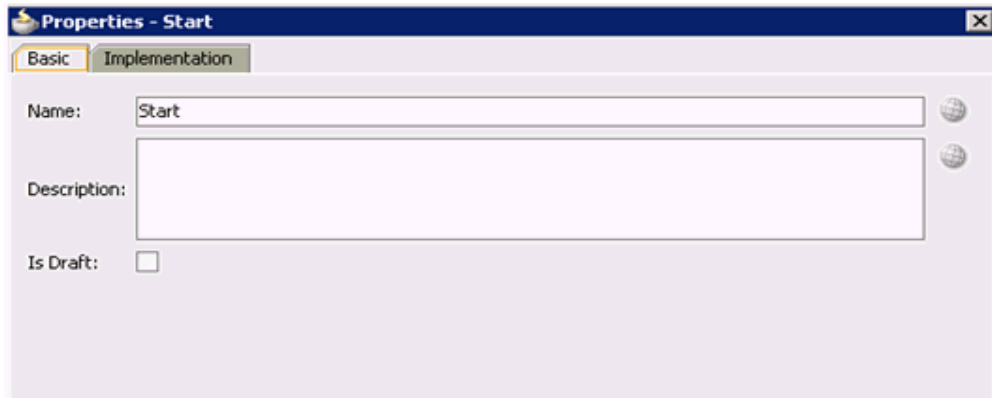
20. 「OK」をクリックします。  
「プロパティ - ServiceTask」ダイアログに戻ります。
21. 「OK」をクリックします。  
図 8-17 に示すように、「Start」と「End」イベント・コンポーネントの間に「ServiceTask」が作成されます。

図 8-17 ServiceTask



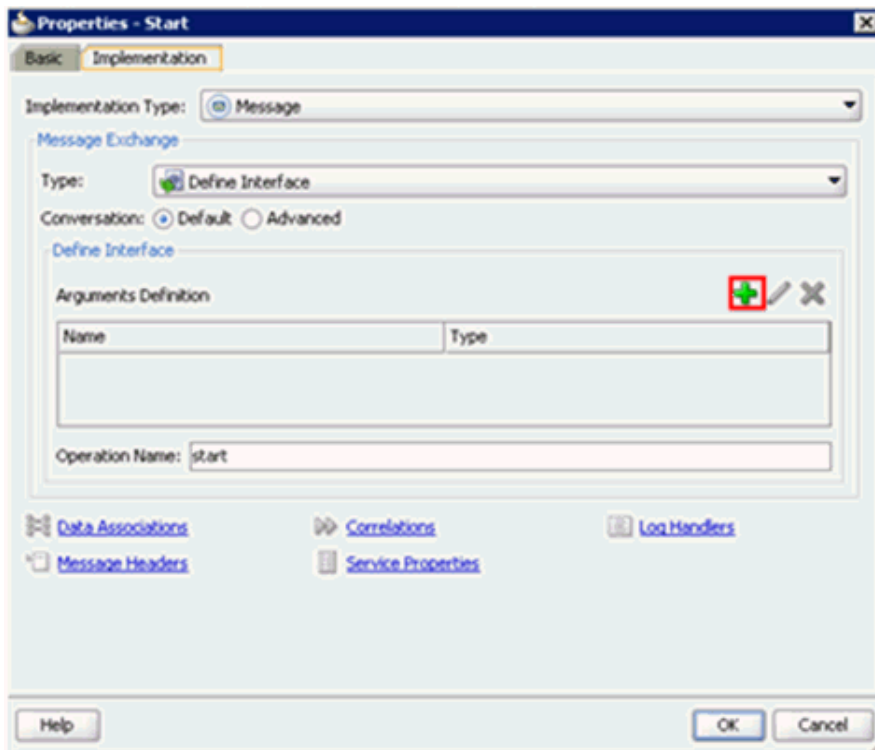
22. プロセスを保存し、「Start」 イベント・コンポーネントをダブルクリックします。  
図 8-18 に示すように、「プロパティ - Start」 ダイアログが表示されます。

図 8-18 「プロパティ - Start」 ダイアログ



23. 図 8-19 に示すように、「実装」タブをクリックします。

図 8-19 「実装」タブ



24. 引数定義フィールドの右側にあるプラス・アイコンをクリックします。  
「引数の作成」 ダイアログが表示されます。

25. 図 8-20 に示すように、「名前」フィールドに名前を入力し(デフォルトでは argument1)、「タイプ」フィールドのドロップダウン・ボタンをクリックして、リストで「参照」を選択します。

図 8-20 「引数の作成」 ダイアログ

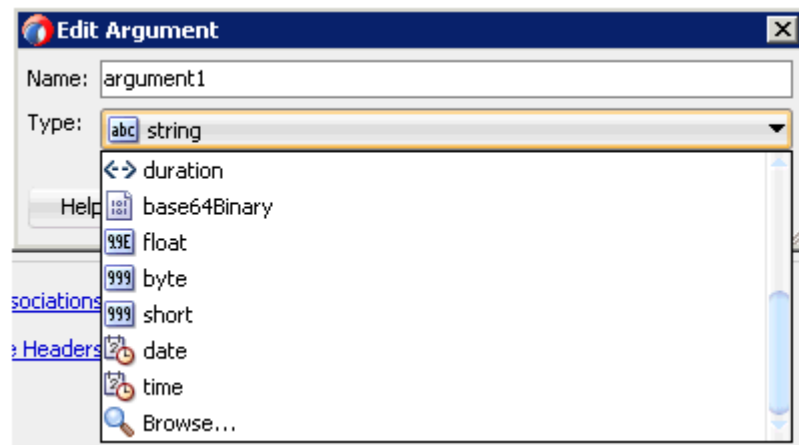
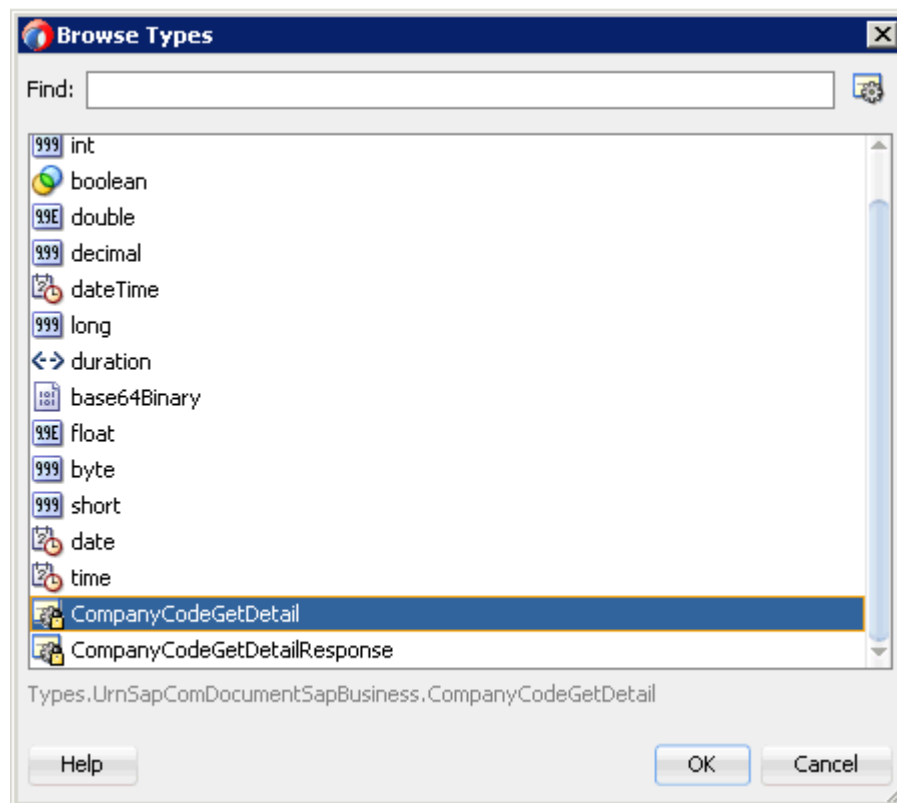


図 8-21 に示すように、「タイプの参照」ダイアログが表示されます。

図 8-21 「タイプの参照」ダイアログ

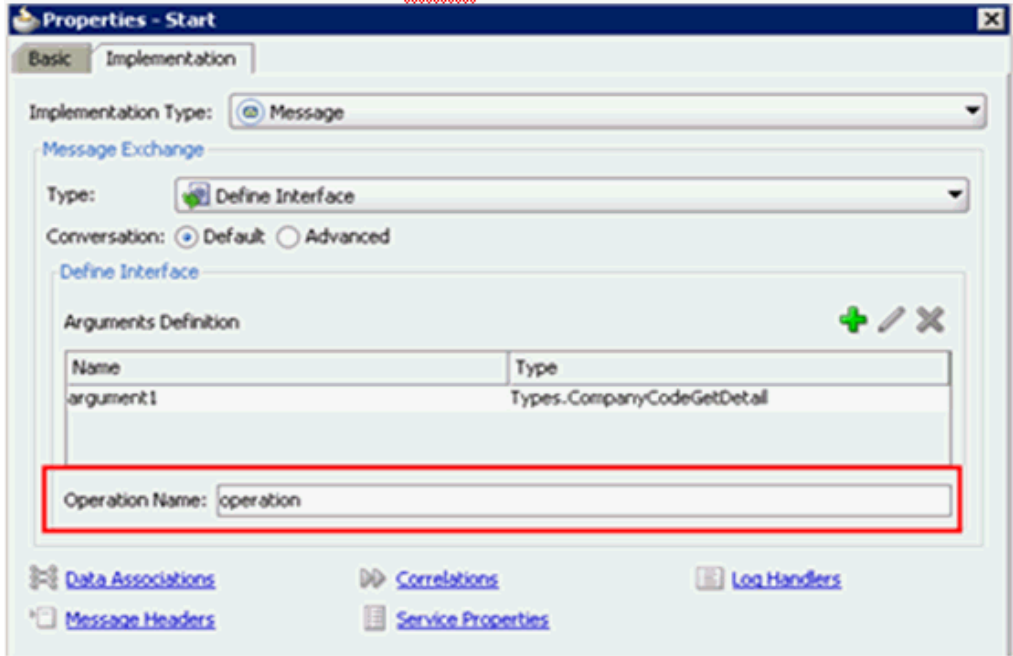


26. 最初のコンポーネント (CompanyCodeGetDetail など) を選択し、「OK」をクリックします。

「引数の作成」ダイアログに戻ります。

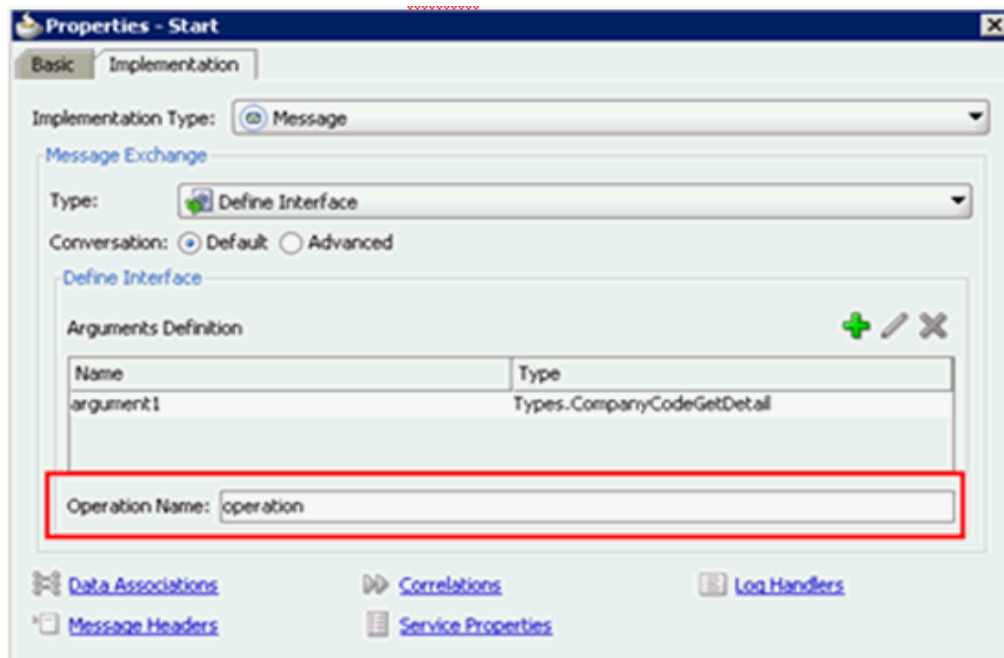
27. 「OK」をクリックします。

「プロパティ - Start」ダイアログに戻ります。

28.  8-22 に示すように、「操作名」フィールドで、**start** (デフォルト) を **operation** に変更します。

**注意:** この変更は古い BPM ペイロードを処理するために必要です。

図 8-22 「操作名」フィールド



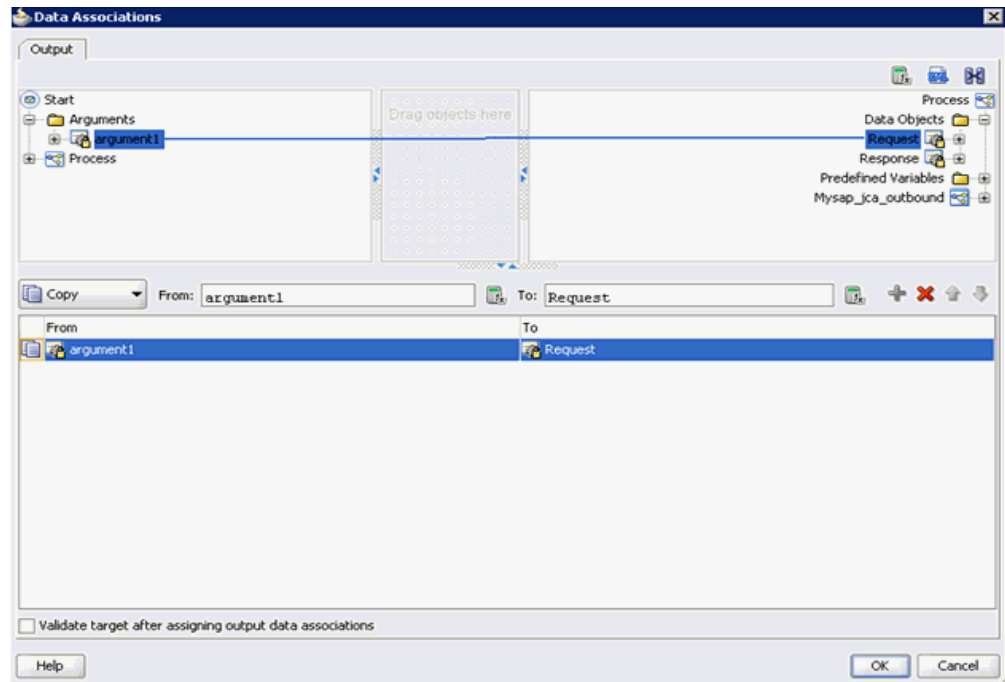
29. データ・アソシエーション・ハイパーリンクをクリックします。

データ・アソシエーション・ダイアログが表示されます。

30. 左ペインで「引数」ノードの下の「**arguments1**」を選択してドラッグし、右ペインの「データ・オブジェクト」の下の「**Request**」データ・オブジェクトに接続します。

31. 図 8-23 に示すように、「OK」をクリックします。

図 8-23 「OK」 ボタン

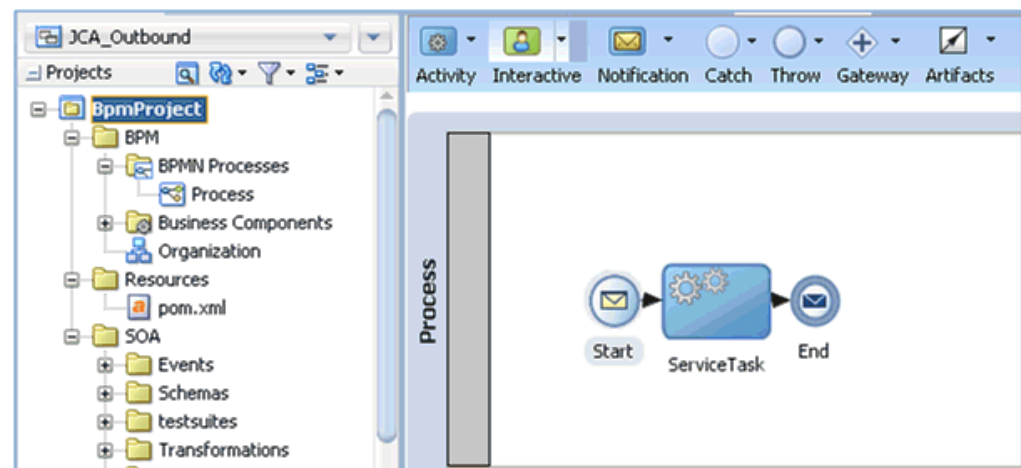


「プロパティ - Start」ダイアログに戻ります。

32. 「OK」をクリックします。

図 8-24 に示すように、「プロセス」ワークスペース領域に戻ります。

図 8-24 「プロセス」ワークスペース領域



33. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド BPM プロセス・コンポーネントを保存します。

これで、書込み操作用のファイル・アダプタを作成する準備ができました。

### 8.4.2.3 書込み操作のファイル・アダプタの作成

この項では、書込み操作のファイル・アダプタを作成する方法について説明します。

書込み操作のファイル・アダプタを作成するには、次のようにします。

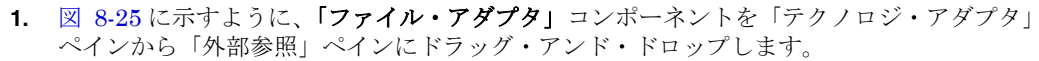
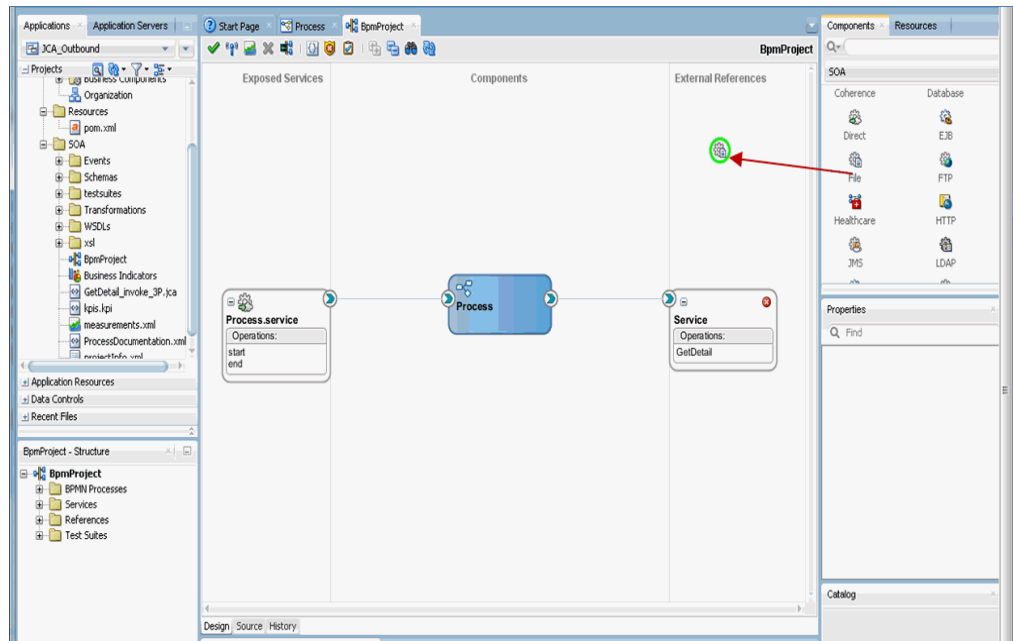
1.  8-25 に示すように、「ファイル・アダプタ」コンポーネントを「テクノロジー・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。

図 8-25 「ファイル・アダプタ」コンポーネント



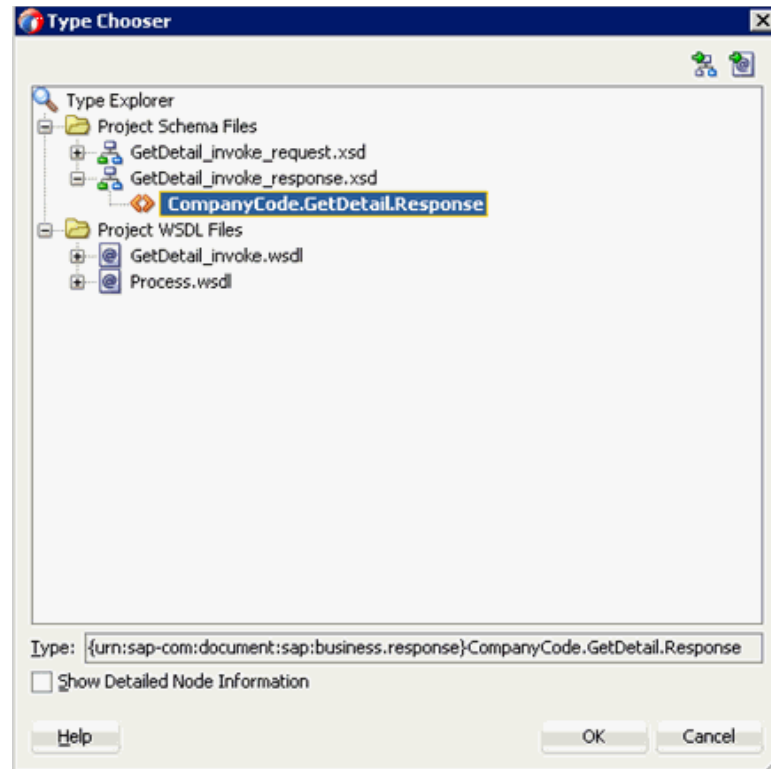
アダプタ構成ウィザードが表示されます。

2. 参照名 (FileWrite など) を指定します。
3. 「次へ」をクリックします。  
「アダプタ・インタフェース」ページが表示されます。
4. 「操作およびスキーマから定義 (後で指定)」オプションが選択されていることを確認してください。
5. 「次へ」をクリックします。  
「ファイル・サーバーの接続」ページが表示されます。
6. 「次へ」をクリックします。  
「操作」ページが表示されます。
7. 「操作タイプ」オプションのリストで「ファイルの書込み」を選択し、「操作名」を指定します (Write など)。
8. 「次へ」をクリックします。  
「ファイル構成」ページが表示されます。
9. 出力ファイルを書き込むファイル・システム上の場所を指定します。
10. ファイル・ネーミング規則フィールドで、出力ファイルの名前を指定します。
11. 「次へ」をクリックします。  
「メッセージ」ページが表示されます。

12. 「URL」 フィールドの右側にある「参照」をクリックします。

図 8-26 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 8-26 「タイプ・チューザ」ダイアログ

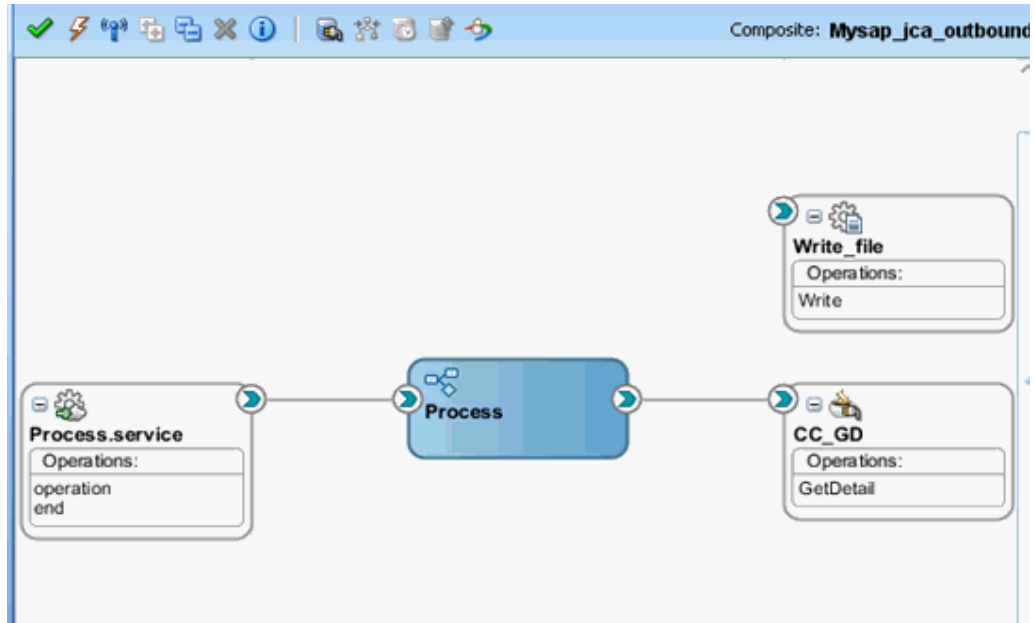


13. 「プロジェクトのスキーマ・ファイル」 → 「GetDetail\_invoke\_response.xsd」を展開します。
14. 使用可能なスキーマを選択します (CompanyCode.GetDetail.Response など)。
15. 「OK」をクリックします。  
「メッセージ」ページに戻ります。
16. 「次へ」をクリックします。  
「終了」ページが表示されます。

17. 「終了」をクリックします。

☒ 8-27 に示すように、「外部参照」ペインにファイル・アダプタ・サービスが作成されます。

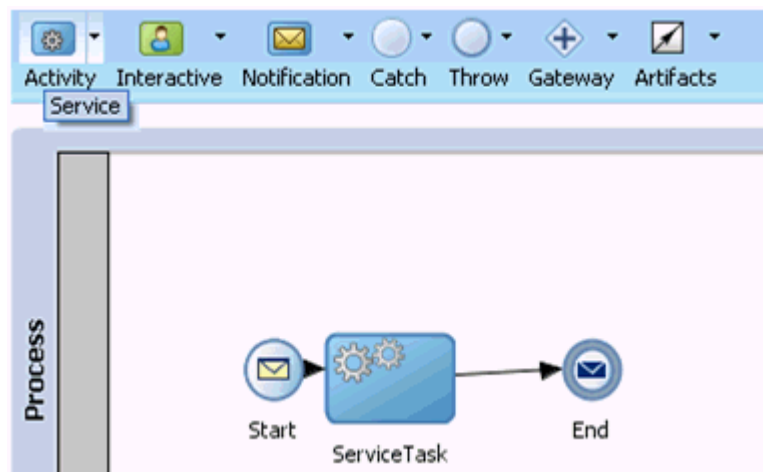
図 8-27 ファイル・アダプタ・サービス



18. BPMN プロセス・コンポーネントをダブルクリックします。

☒ 8-28 に示すように、BPMN プロセスが表示されます。

図 8-28 BPMN プロセス



19. 「アクティビティ」アイコンをクリックし、「サービス」を選択します。



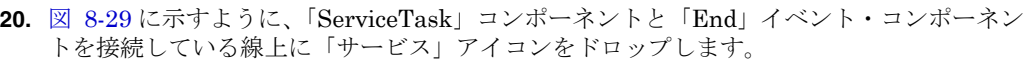
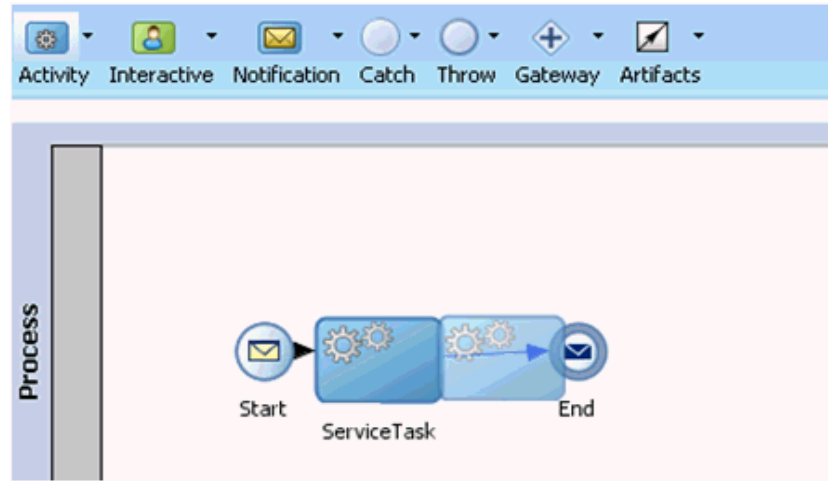
20.  8-29 に示すように、「ServiceTask」コンポーネントと「End」イベント・コンポーネントを接続している線上に「サービス」アイコンをドロップします。

図 8-29 「アクティビティ」アイコン



「プロパティ - ServiceTask1」ダイアログが表示されます。

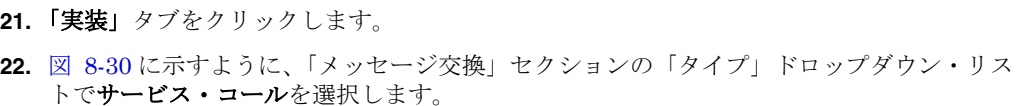
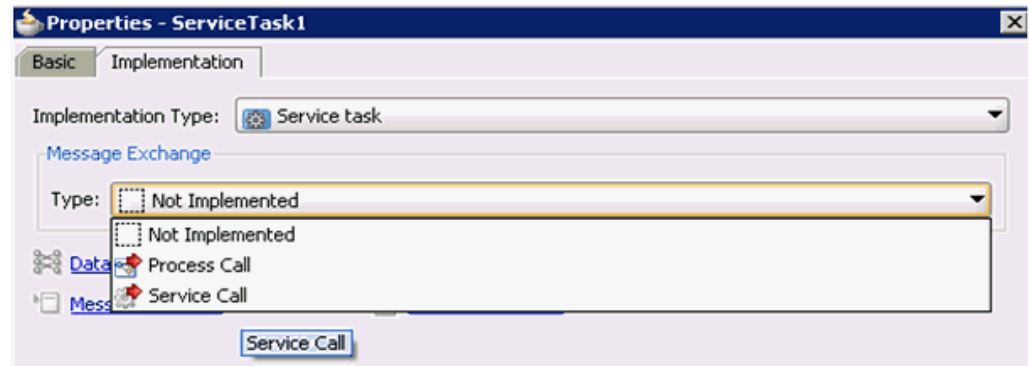
21. 「実装」タブをクリックします。
22.  8-30 に示すように、「メッセージ交換」セクションの「タイプ」ドロップダウン・リストでサービス・コールを選択します。

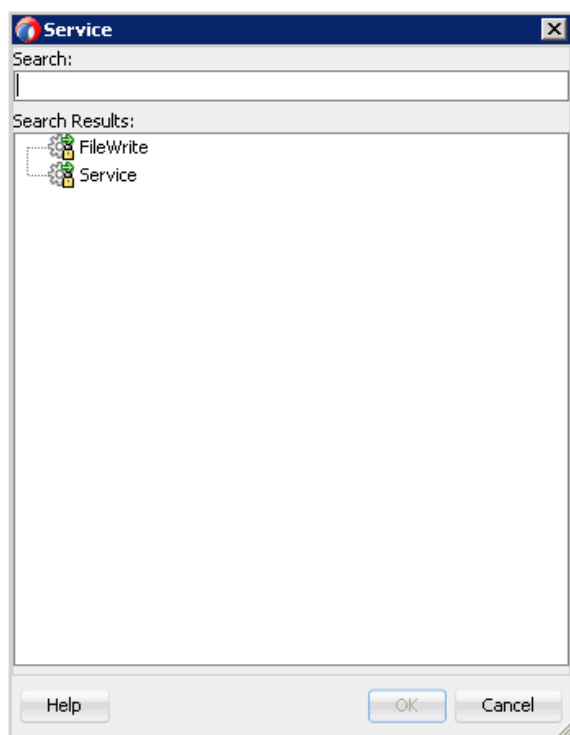
図 8-30 サービス・コール



23. 「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 8-31 に示すように、「サービス」ダイアログが表示されます。

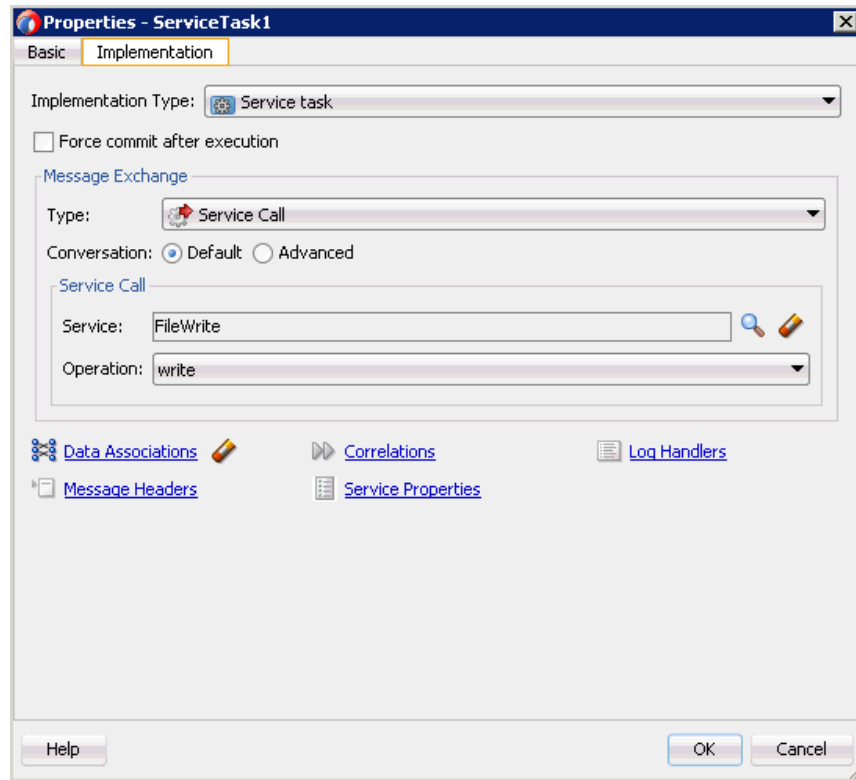
図 8-31 「サービス」ダイアログ



24. 作成した書込み操作のサービスを選択し (FileWrite など)、「OK」をクリックします。

図 8-32 に示すように、「プロパティ - ServiceTask1」ダイアログに戻ります。

図 8-32 「プロパティ - ServiceTask1」ダイアログ



25. データ・アソシエーション・ハイパーリンクをクリックします。

図 8-33 に示すように、データ・アソシエーション・ダイアログが表示されます。

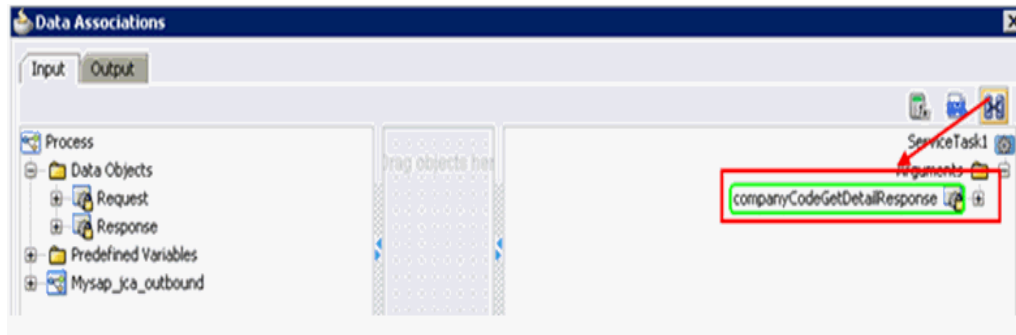
図 8-33 データ・アソシエーション・ダイアログ



26. 「入力」タブで、右ペイン上部にある XSL トランスフォーメーション・アイコンをクリックします。

27. 図 8-34 に示すように、XSL トランスフォーメーション・アイコンを「companyCodeGetDetailResponse」ノードにドラッグ・アンド・ドロップします。

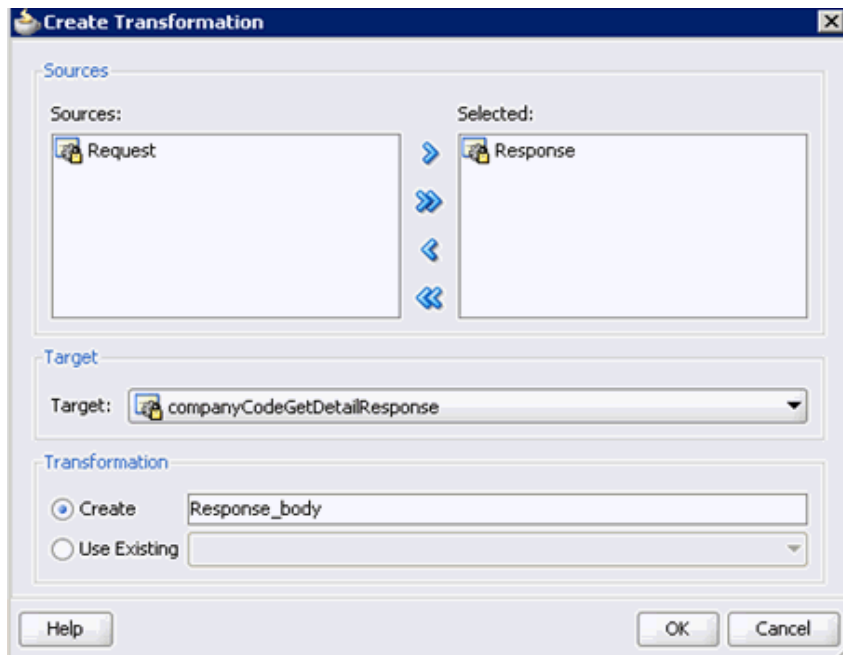
図 8-34 「CompanyCodeGetDetailResponse」ノード



トランスフォーメーションの作成ダイアログが表示されます。

28. 「ソース」セクションで「Response」を選択し、右矢印記号をクリックします。  
図 8-35 に示すように、「Response」オブジェクトが「選択済」要素領域に追加されます。

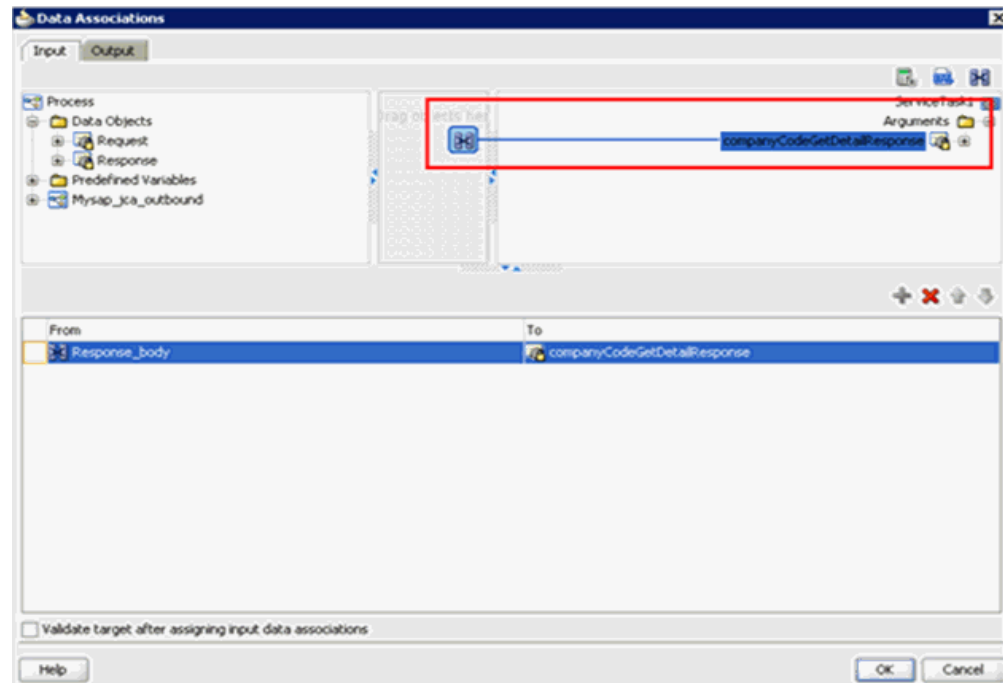
図 8-35 「Response」オブジェクト



29. 「ターゲット」 ドロップダウン・リストで選択されているデフォルト値および「作成」フィールドのデフォルト名を、「OK」をクリックして受け入れます。

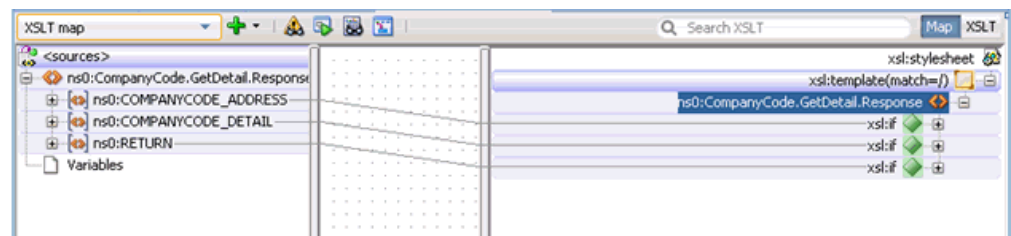
図 8-36 に示すように、XSL トランスフォーメーションが作成されたデータ・アソシエーション・ダイアログ・ウィンドウに戻ります。

図 8-36 データ・アソシエーション・ダイアログ



30. 「OK」をクリックします。  
「プロパティ - ServiceTask1」ダイアログに戻ります。
31. 「OK」をクリックします。  
「Response\_body.xml」タブが表示されます。
32. ソース要素とターゲット要素を自動マップします。  
「自動マップ・プリファレンス」ダイアログが表示されます。
33. デフォルト値を受け入れて、「OK」をクリックします。  
図 8-37 に示すように、トランスフォーメーションが完了しました。

図 8-37 完了したトランスフォーメーション



34. トランスフォーメーションを保存します。

35. 「プロセス」ワークスペース領域に戻ります。  
「ServiceTask1」コンポーネントが「ServiceTask」コンポーネントと「End」イベント・コンポーネントの間に作成されます。
36. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド BPM プロセス・コンポーネントを保存します。  
これで、アウトバウンド BPM プロセスをデプロイする準備ができました。

### 8.4.3 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの [6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」](#) を参照してください。

### 8.4.4 BPM アウトバウンド・プロセスのデプロイ

メディアエータ・アウトバウンド・プロセスをデプロイするには、次のようにします。

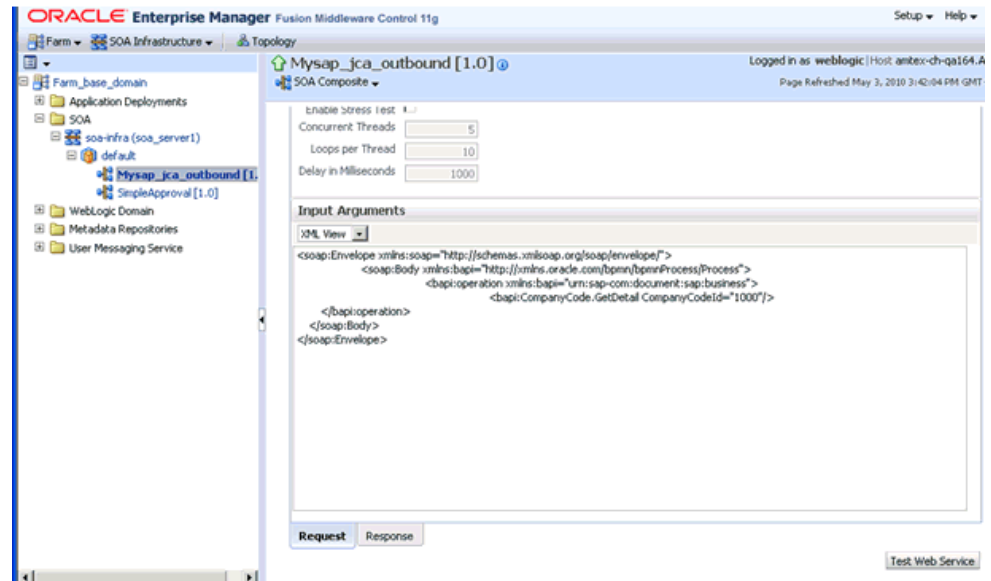
1. 左ペインのプロジェクト名を右クリックし、「デプロイ」を選択してから「**J2CA\_Outbound**」をクリックします。  
「デプロイメント・アクション」ページが表示されます。
2. 「**アプリケーション・サーバーにデプロイ**」が選択されていることを確認します。
3. 「次へ」をクリックします。  
「構成のデプロイ」ページが表示されます。
4. デフォルト値を選択済のままにして、「次へ」をクリックします。  
「サーバーの選択」ページが表示されます。
5. 構成済の使用可能なアプリケーション・サーバーを選択し、「次へ」をクリックします。  
「SOA サーバー」ページが表示されます。
6. ターゲット SOA サーバーを選択し、「次へ」をクリックします。  
「サマリー」ページが表示されます。
7. プロジェクトに関する使用可能なすべてのデプロイメント情報を確認および検証し、「終了」をクリックします。  
詳細は、6-34 ページの [6.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」](#) を参照してください。

### 8.4.5 Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し

Oracle Enterprise Manager コンソールで入力 XML ドキュメントを呼び出すには、次のようにします。

1. Oracle Enterprise Manager コンソールにログオンします。
2. 左ペインで、ドメインを展開してから、「**SOA**」フォルダを展開します。
3. 使用可能なプロジェクトを選択します (Mysap\_jca\_outbound など)。
4. 「テスト」をクリックします。
5. 「リクエスト」タブをクリックします。

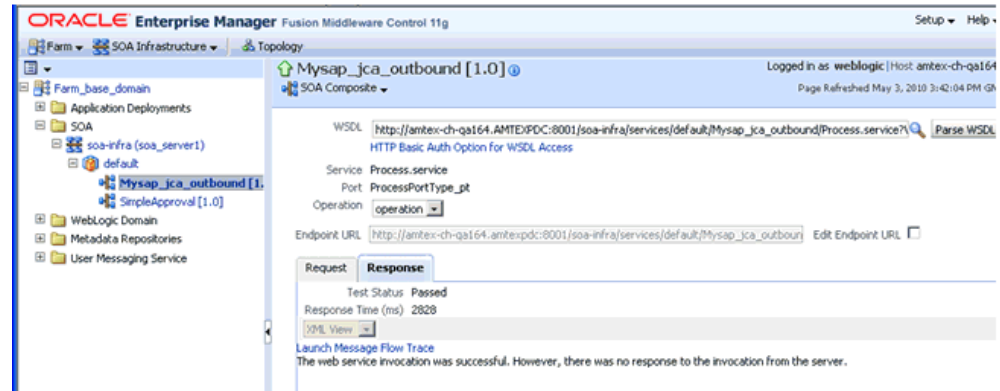
図 8-38 「リクエスト」タブ



6. 図 8-38 に示すように、「値」フィールドに適切な入力値を指定して、「Web サービスのテスト」をクリックします。

図 8-39 に示すように、「レスポンス」タブにレスポンスが受信されます。これは、Oracle Enterprise Manager コンソールで呼出しが成功したことを示します。

図 8-39 受信されたレスポンス



7. ファイル・システムの定義済出力ディレクトリに移動して、受信された XML レスポンス・ドキュメントを開きます。

XML レスポンス・ドキュメントには、値を持つ生成済の出力が含まれています。次に例を示します。

```
<?xml version="1.0" encoding="UTF-8" ?><bapi:CompanyCode.GetDetail.Response
xmlns:bapi="urn:sap-com:document:sap:business:response"
xmlns="urn:sap-com:document:sap:business:response">
  <bapi:COMPANYCODE_ADDRESS>
    <bapi:ADDR_NO>000000121</bapi:ADDR_NO>
    <bapi:FORMOFADDR>Firma</bapi:FORMOFADDR>
    <bapi:NAME>Ides AG</bapi:NAME>
    <bapi:NAME_2/>
    <bapi:NAME_3/>
    <bapi:NAME_4/>
    <bapi:C_O_NAME/>
  </bapi:COMPANYCODE_ADDRESS>
</bapi:CompanyCode.GetDetail.Response>
```

```

    <bapi:CITY>Frankfurt</bapi:CITY>
    <bapi:DISTRICT/>
    <bapi:CITY_NO/>
    <bapi:POSTL_COD1>60441</bapi:POSTL_COD1>
    <bapi:POSTL_COD2>60070</bapi:POSTL_COD2>
    <bapi:POSTL_COD3/>
    <bapi:PO_BOX>160529</bapi:PO_BOX>
    <bapi:PO_BOX_CIT/>
    <bapi:DELIV_DIS/>
    <bapi:STREET>Lyoner Stern 231</bapi:STREET>
    <bapi:STREET_NO/>
    <bapi:STR_ABBR/>
    <bapi:HOUSE_NO/>
    <bapi:STR_SUPPL1/>
    <bapi:STR_SUPPL2/>
    <bapi:LOCATION/>
    <bapi:BUILDING/>
    <bapi:FLOOR/>
    <bapi:ROOM_NO/>
    <bapi:COUNTRY>DE</bapi:COUNTRY>
    <bapi:LANGU>D</bapi:LANGU>
    <bapi:REGION>06</bapi:REGION>
    <bapi:SORT1>IDES</bapi:SORT1>
    <bapi:SORT2/>
    <bapi:TIME_ZONE>CET</bapi:TIME_ZONE>
    <bapi:TAXJURCODE/>
    <bapi:ADR_NOTES/>
    <bapi:COMM_TYPE/>
    <bapi:TEL1_NUMBR/>
    <bapi:TEL1_EXT/>
    <bapi:FAX_NUMBER/>
    <bapi:FAX_EXTENS/>
  </bapi:COMPANYCODE_ADDRESS>
  <bapi:COMPANYCODE_DETAIL>
    <bapi:COMP_CODE>1000</bapi:COMP_CODE>
    <bapi:COMP_NAME>IDES AG</bapi:COMP_NAME>
    <bapi:CITY>Frankfurt</bapi:CITY>
    <bapi:COUNTRY>DE</bapi:COUNTRY>
    <bapi:CURRENCY>EUR</bapi:CURRENCY>
    <bapi:LANGU>D</bapi:LANGU>
    <bapi:CHRT_ACCTS>INT</bapi:CHRT_ACCTS>
    <bapi:FY_VARIANT>K4</bapi:FY_VARIANT>
    <bapi:VAT_REG_NO>DE123456789</bapi:VAT_REG_NO>
    <bapi:COMPANY>001000</bapi:COMPANY>
    <bapi:ADDR_NO>0000000121</bapi:ADDR_NO>
    <bapi:COUNTRY_ISO>DE</bapi:COUNTRY_ISO>
    <bapi:CURRENCY_ISO>EUR</bapi:CURRENCY_ISO>
    <bapi:LANGU_ISO>DE</bapi:LANGU_ISO>
  </bapi:COMPANYCODE_DETAIL>
  <bapi:RETURN>
    <bapi:TYPE/>
    <bapi:CODE/>
    <bapi:MESSAGE/>
    <bapi:LOG_NO/>
    <bapi:LOG_MSG_NO>000000</bapi:LOG_MSG_NO>
    <bapi:MESSAGE_V1/>
    <bapi:MESSAGE_V2/>
    <bapi:MESSAGE_V3/>
    <bapi:MESSAGE_V4/>
  </bapi:RETURN>
</bapi:CompanyCode.GetDetail.Response>

```



## 8.5 トランスフォーメーションを使用したイベント統合のためのインバウンド BPM プロセスの設計 (J2CA 構成)

この項では、Oracle Application Adapter for SAP R/3 を SAP R/3 と統合し、イベント・データを受信する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

<ADAPTER\_HOME>\etc\sample\SAP\_Samples.zip\SAP\_Samples\BPM\J2CA\Inbound\_Project  
アダプタ・デザインタイム構成を完了するには、次のツールが必要です。

- Oracle Adapter Application Explorer (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPM Designer (JDeveloper)

---

**注意:** この章の例では、JDeveloper を使用しています。

---

ここでは、次の項目について説明します。

- [8.5.1 項「BPM 用の空のコンポジットの作成」](#)
- [8.5.2 項「BPM インバウンド・プロセスの定義」](#)

BPM プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-42 ページの「[イベント統合のための WSDL の生成](#)」を参照してください。

### 8.5.1 BPM 用の空のコンポジットの作成

Oracle JDeveloper で新規アプリケーション・サーバー接続を構成する方法の詳細は、6-2 ページの [6.3 項「新規アプリケーション・サーバー接続の構成」](#) を参照してください。

### 8.5.2 BPM インバウンド・プロセスの定義

この項では、BPM インバウンド・プロセスを定義する方法について、次の項目で説明します。

- [8.5.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」](#)
- [8.5.2.2 項「インバウンド BPM プロセス・コンポーネントの構成」](#)
- [8.5.2.3 項「書込み操作用のファイル・アダプタの作成」](#)
- [8.5.2.4 項「12c でのデプロイメントに関する既知の問題の調整」](#)

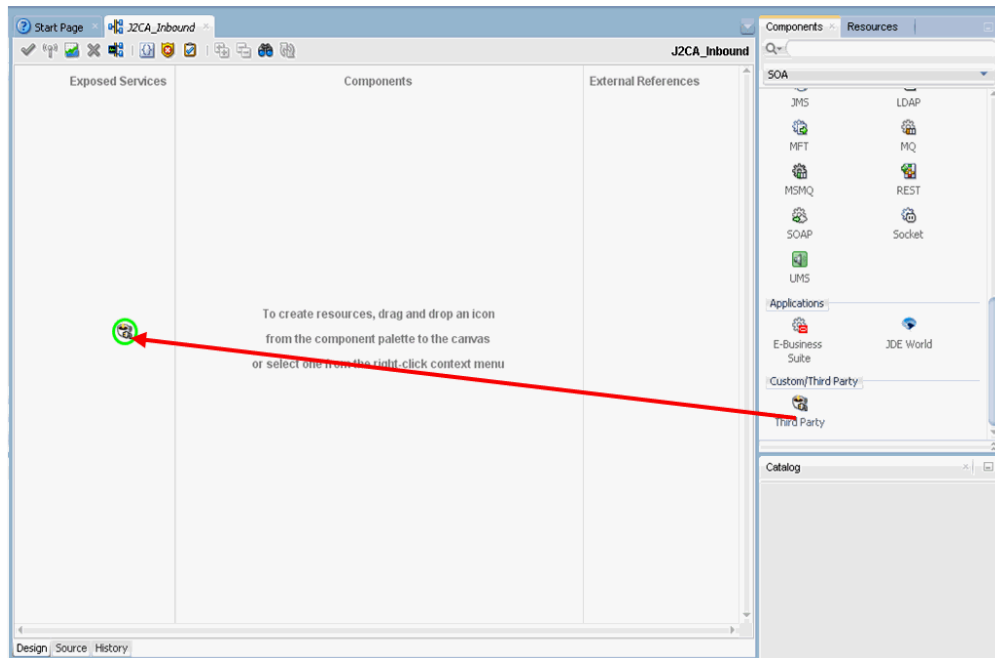
#### 8.5.2.1 サード・パーティ・アダプタ・サービス・コンポーネントの構成

サード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

1. 作成されたプロジェクトをダブルクリックして、コンポーネントをロードします。

2. 図 8-40 に示すように、「サード・パーティ・アダプタ」コンポーネントを「カスタム / サード・パーティ」ペインから「公開されたサービス」ペインにドラッグ・アンド・ドロップします。

図 8-40 サード・パーティ・アダプタ・コンポーネント



「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

3. サード・パーティ・アダプタ・サービスの名前を入力します。
4. 「タイプ」リストで「サービス」が選択されていることを確認します (デフォルト)。
5. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。  
「WSDL の選択」ダイアログが表示されます。
6. 「ファイルシステム」を選択し、次のディレクトリでインバウンド WSDL ファイルを参照して選択します。  
`<ADAPTER_HOME>\wsdls`
7. 「OK」をクリックします。  
「ファイルのローカライズ」ダイアログが表示されます。
8. 「OK」をクリックします。  
インバウンド WSDL ファイルと関連する receive\_request XML スキーマ・ファイル (.xsd) が、作成されたプロジェクト・フォルダにインポートされます。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。
9. 「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。  
トランスフォーメーション・チューザ・ダイアログが表示されます。
10. 「ファイルシステム」を選択し、次のディレクトリで JCA プロパティ・ファイルを参照して選択します。  
`<ADAPTER_HOME>\wsdls`

11. 「OK」をクリックします。

「ファイルのコピー」メッセージが表示されます。

12. 「はい」をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

13. 「OK」をクリックします。

「公開されたサービス」ペインにサード・パーティ・アダプタ・サービス・コンポーネント (matmas) が作成されます。

これで、インバウンド BPM プロセス・コンポーネントを構成する準備ができました。

詳細は、6-50 ページの 6.5.3.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの作成」を参照してください。

### 8.5.2.2 インバウンド BPM プロセス・コンポーネントの構成

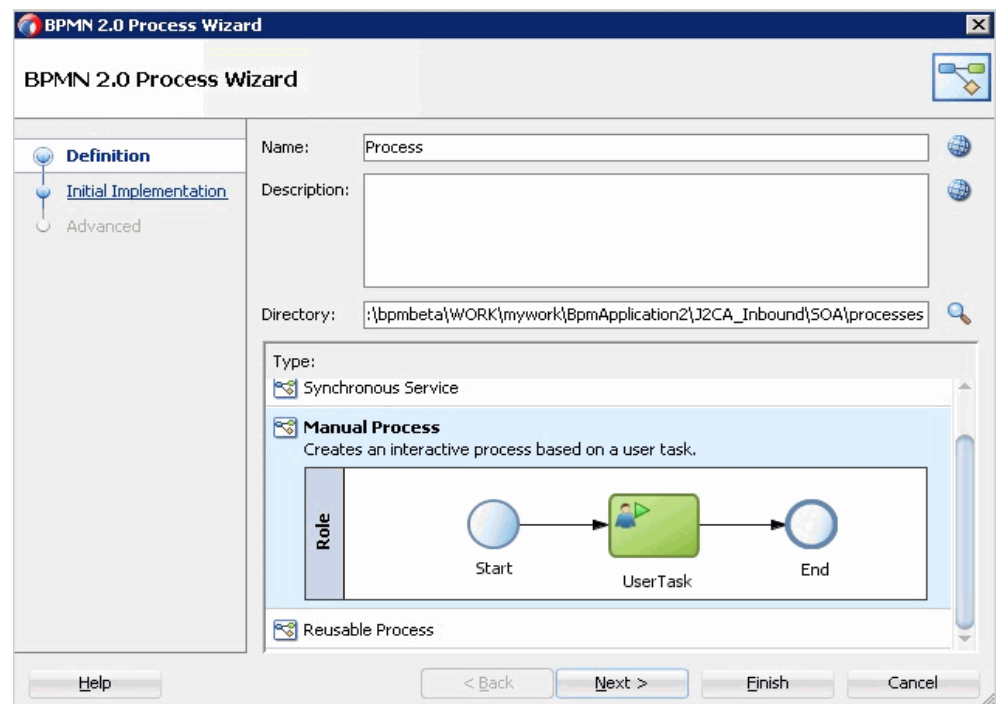
この項では、インバウンド BPM プロセス・コンポーネントを構成する方法について説明します。

インバウンド BPM プロセス・コンポーネントを構成するには、次のようにします。

1. 「BPMN プロセス」コンポーネントを「コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

図 8-41 に示すように、BPMN プロセスの作成のダイアログが表示されます。

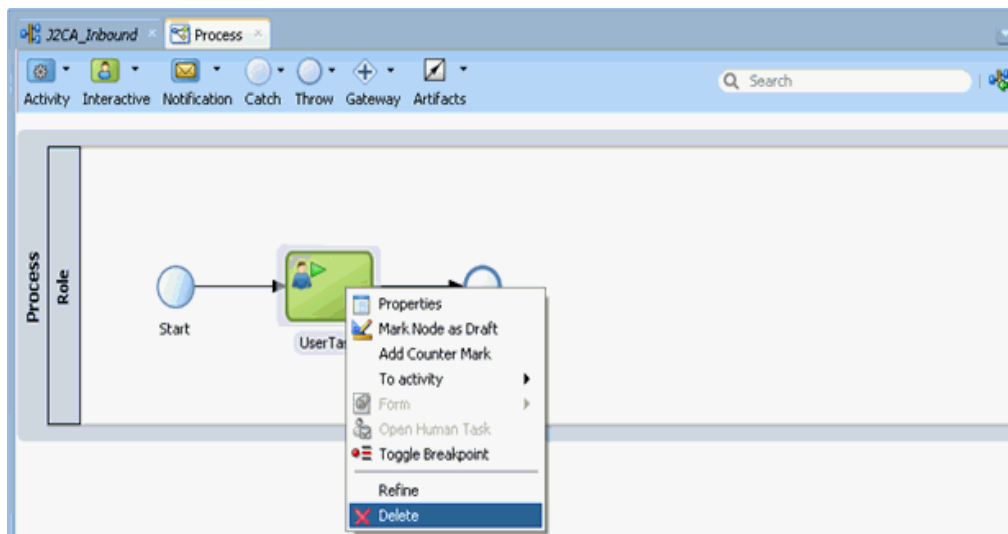
図 8-41 BPMN プロセスの作成のダイアログ



2. 「タイプ」セクションで手動プロセスを選択します。

3. 「終了」をクリックします。  
☒ 8-42 に示すように、BPMN プロセスが表示されます。

図 8-42 BPMN プロセス



4. 「UserTask」を右クリックして、メニューの「削除」を選択します。
5. 「Start」イベント・コンポーネントをダブルクリックします。  
「プロパティ - Start」ダイアログが表示されます。
6. 「実装」タブをクリックします。
7. 「実装タイプ」リストで「メッセージ」を選択します。
8. 「メッセージ交換」の「タイプ」ドロップダウン・リストで「インターフェースの使用」を選択します。

9. 図 8-43 に示すように、「参照」フィールドの右側にある「参照」アイコンをクリックします。

図 8-43 「参照」アイコン

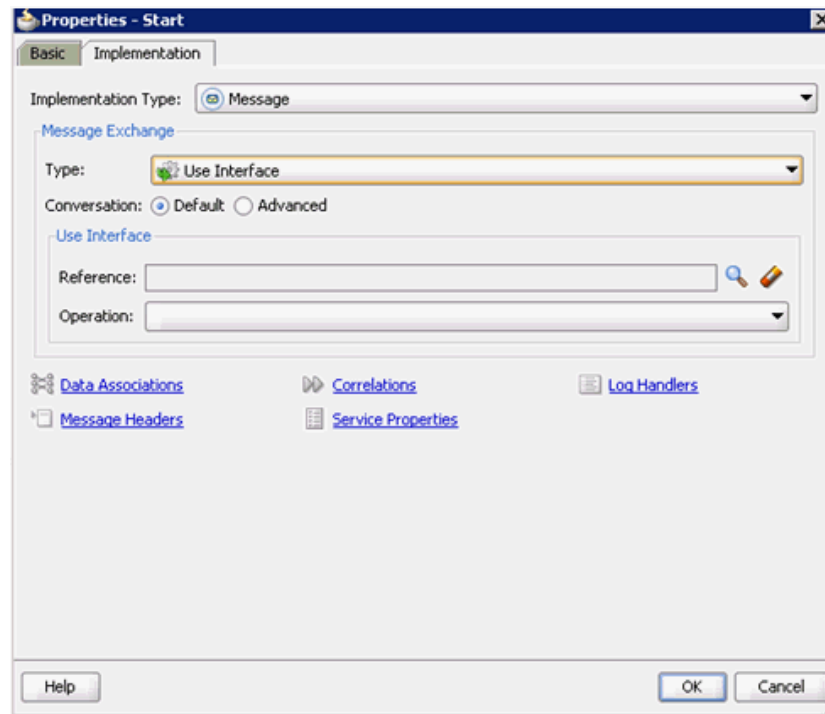
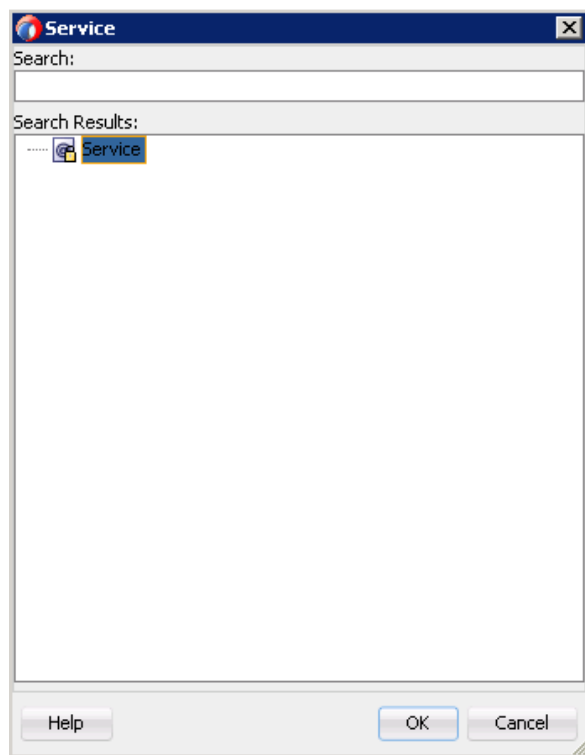


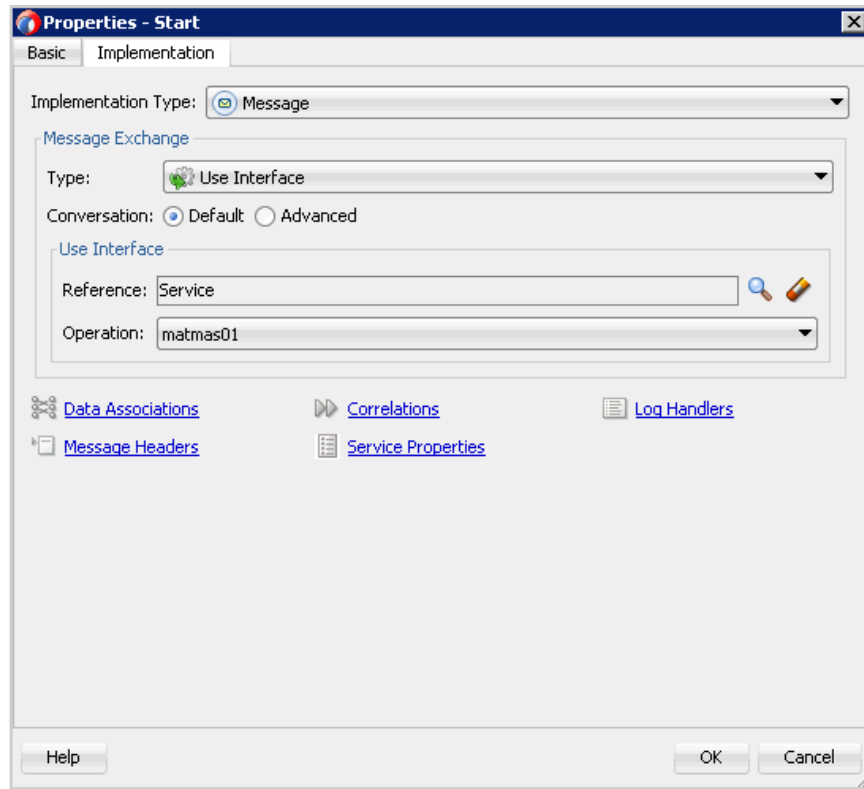
図 8-44 に示すように、「サービス」ダイアログが表示されます。

図 8-44 「サービス」ダイアログ



10. 作成したサード・パーティ・サービスを選択し、「OK」をクリックします。  
図 8-45 に示すように、「プロパティ - Start」ダイアログに戻ります。

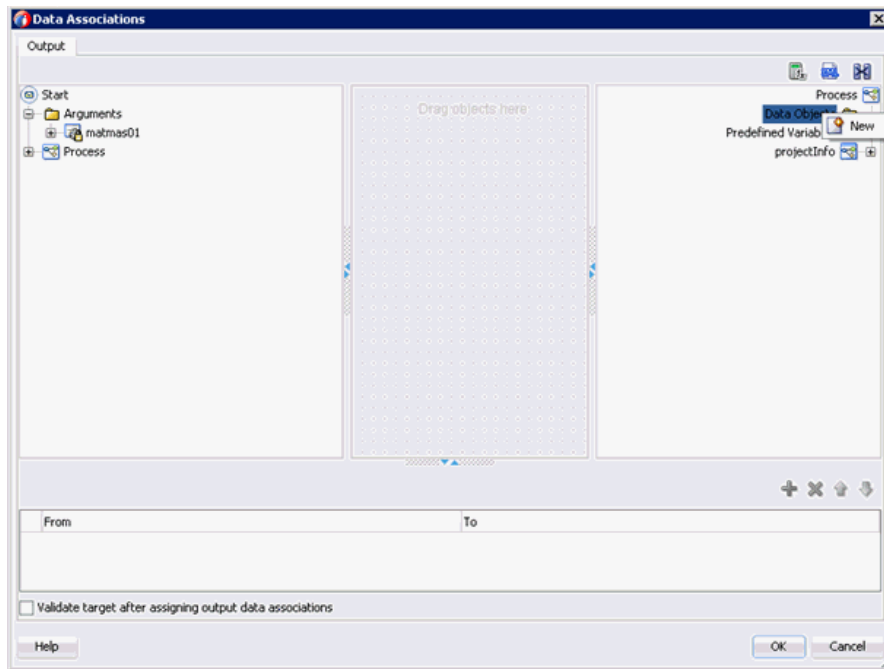
図 8-45 「プロパティ - Start」ダイアログ



11. データ・アソシエーション・アイコンをクリックします。

図 8-46 に示すように、データ・アソシエーション・ダイアログが表示されます。

図 8-46 データ・アソシエーション・ダイアログ



12. 右ペインの「データ・オブジェクト」ノードを右クリックして、「新規」を選択します。

データ・オブジェクトの作成ダイアログが表示されます。

13. 図 8-47 に示すように、「名前」フィールドに名前を入力し、「タイプ」フィールドのドロップダウン・ボタンをクリックして、リストで「参照」を選択します。

図 8-47 データ・オブジェクトの作成ダイアログ

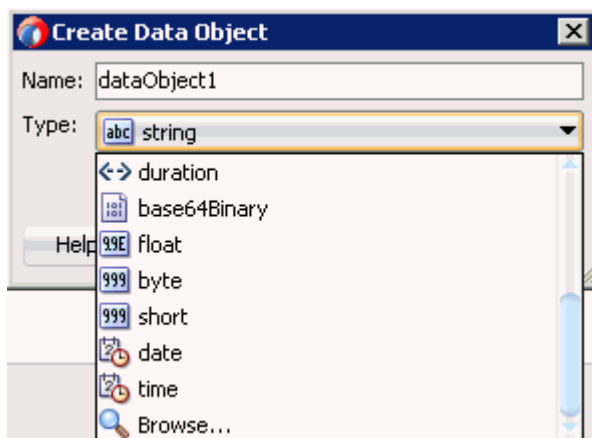
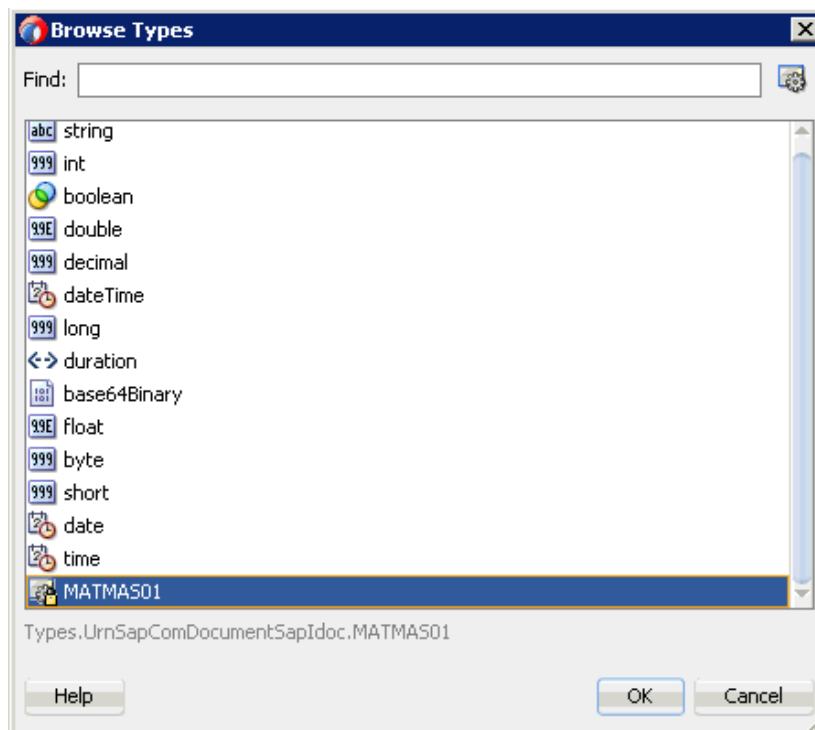




図 8-48 に示すように、「タイプの参照」ダイアログが表示されます。

図 8-48 「タイプの参照」ダイアログ

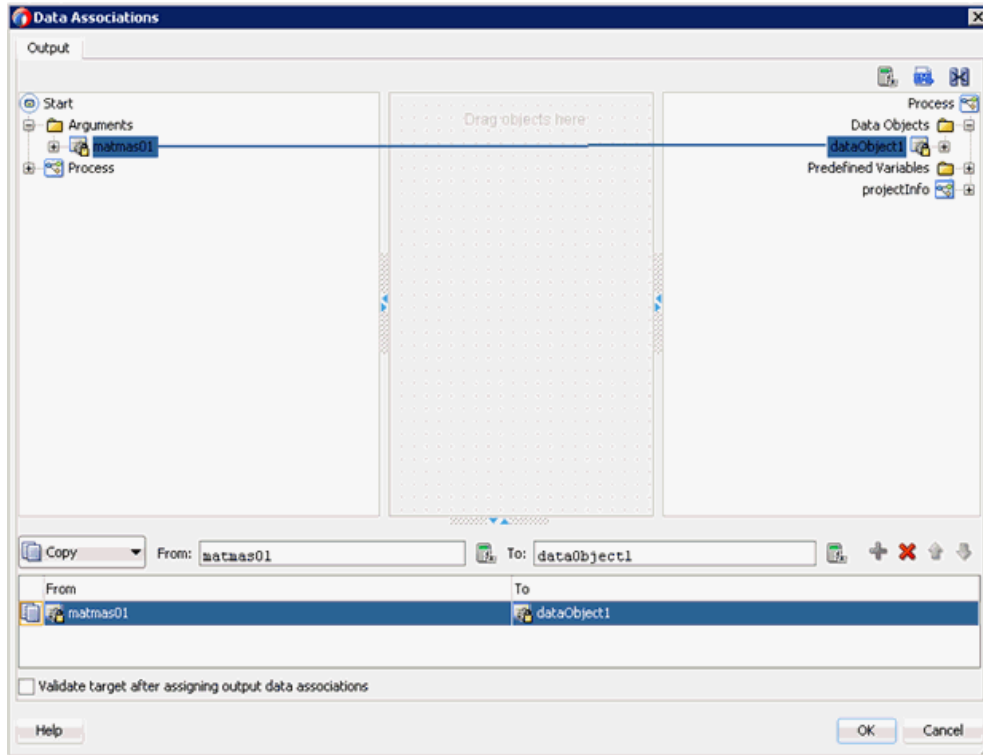


14. コンポーネントを選択し、「OK」をクリックします。  
データ・オブジェクトの作成ダイアログに戻ります。

15. 「OK」 をクリックします。

図 8-49 に示すように、作成されたデータ・オブジェクトは、データ・アソシエーション・ダイアログの「データ・オブジェクト」ノードの下に表示されます。

図 8-49 データ・アソシエーション・ダイアログ



16. 左ペインの「Start」ノードの下の「matmas01」引数を選択して、右ペインのデータ・オブジェクトにドラッグします。

17. 「OK」 をクリックします。

「プロパティ - Start」ダイアログに戻ります。

18. 「OK」 をクリックします。

「プロセス」ワークスペース領域に戻ります。

19. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規インバウンド BPM プロセス・コンポーネントを保存します。

これで、書込み操作のファイル・アダプタを作成する準備ができました。

### 8.5.2.3 書込み操作のファイル・アダプタの作成

この項では、書込み操作のファイル・アダプタを作成する方法について説明します。

書込み操作のファイル・アダプタを作成するには、次のようにします。

1. 「ファイル・アダプタ」コンポーネントを「テクノロジー・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。

アダプタ構成ウィザードが表示されます。

2. 「名前」フィールドで、新規ファイル・アダプタの名前を入力し、「次へ」をクリックします。

「アダプタ・インタフェース」ページが表示されます。

3. 「操作およびスキーマから定義 (後で指定)」オプションが選択されていることを確認してください。

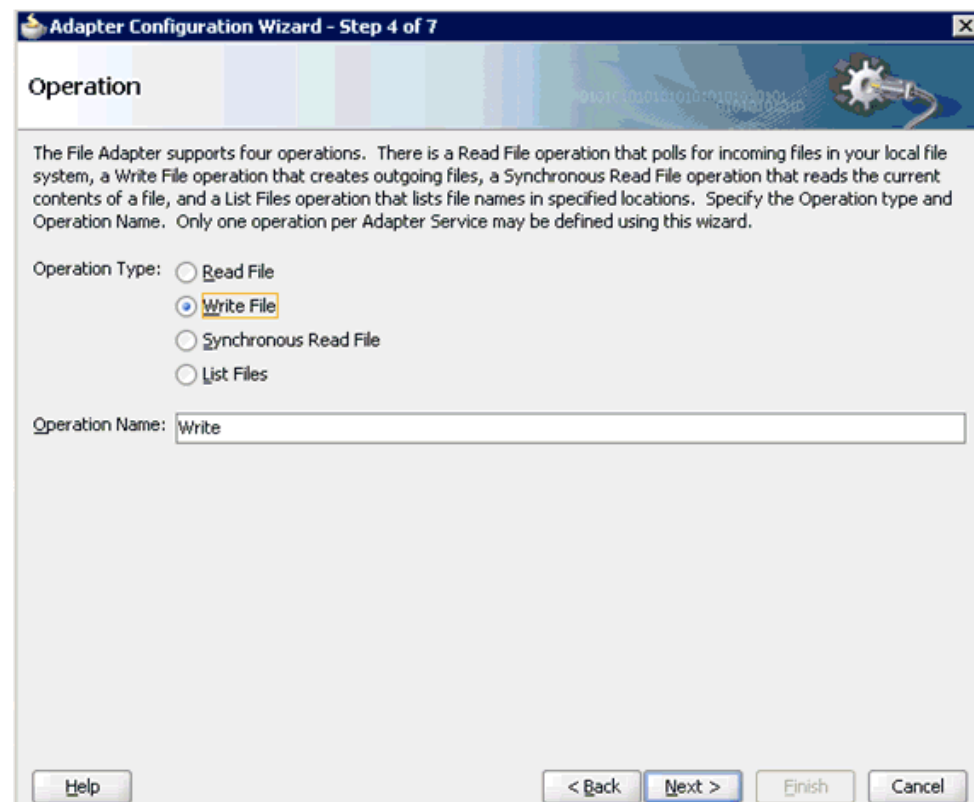
4. 「次へ」をクリックします。

「ファイル・サーバーの接続」ページが表示されます。

5. 「次へ」をクリックします。

図 8-50 に示すように、「操作」ページが表示されます。

図 8-50 「操作」ページ



6. 「操作タイプ」オプションのリストで「ファイルの書込み」を選択し、「操作名」を指定します (Write など)。

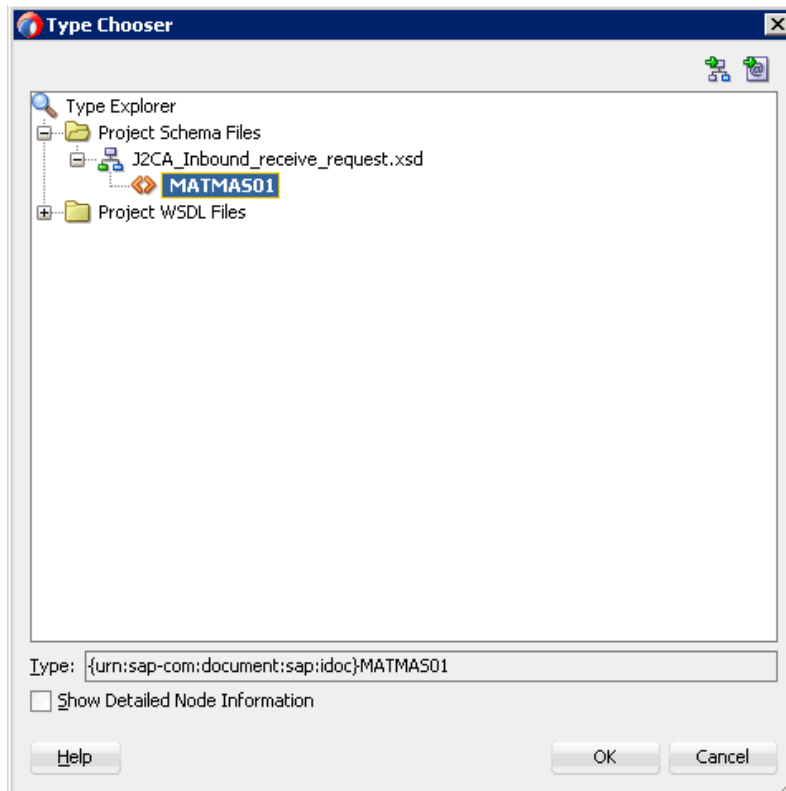
7. 「次へ」をクリックします。

「ファイル構成」ページが表示されます。

8. 出力ファイルを書き込むファイル・システム上の場所を指定します。

9. ファイル・ネーミング規則フィールドで、出力ファイルの名前を指定します。
10. 「次へ」をクリックします。  
「メッセージ」ページが表示されます。
11. 「URL」フィールドの右側にある「参照」をクリックします。  
図 8-51 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 8-51 「タイプ・チューザ」ダイアログ

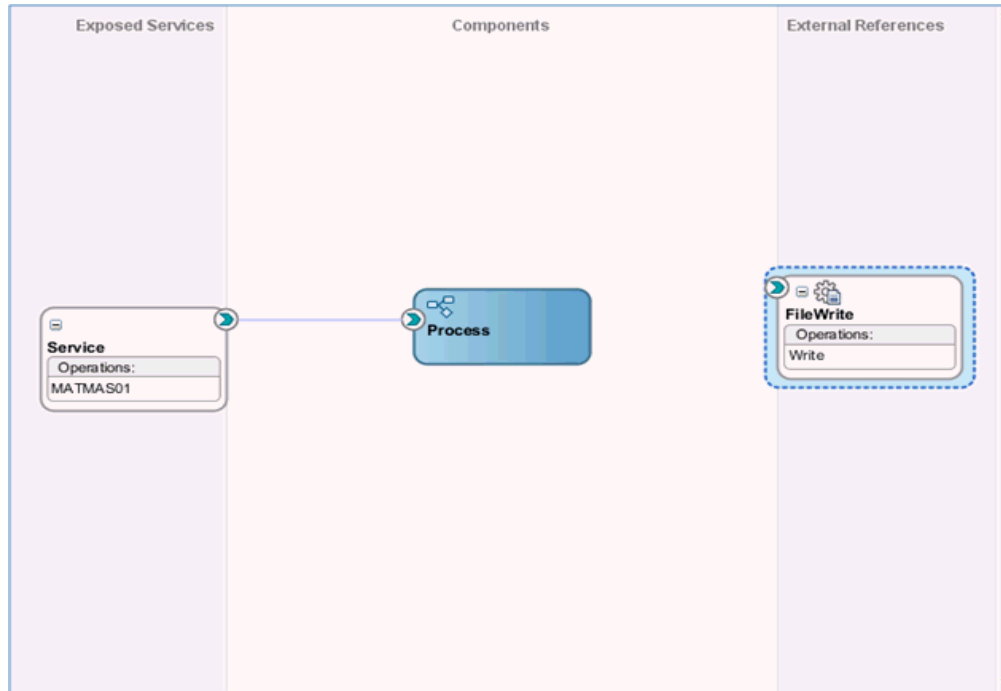


12. 「プロジェクトのスキーマ・ファイル」 → 「J2CA\_Inbound\_receive\_request.xsd」を展開します。
13. 使用可能なスキーマを選択します。
14. 「OK」をクリックします。  
「メッセージ」ページに戻ります。
15. 「次へ」をクリックします。  
「終了」ページが表示されます。

16. 「終了」をクリックします。

図 8-52 に示すように、「外部参照」ペインにファイル・アダプタ・サービスが作成されます。

図 8-52 ファイル・アダプタ・サービス



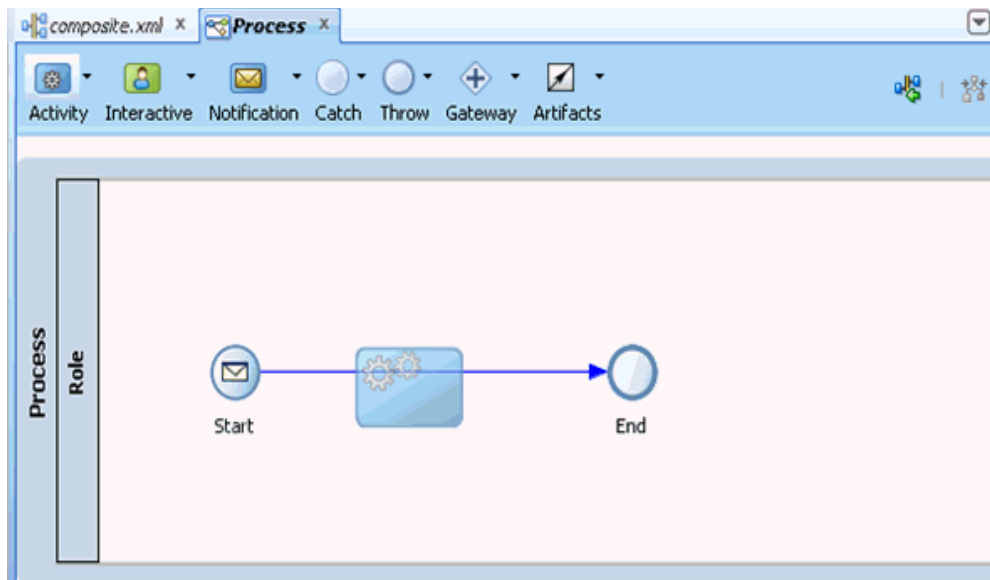
17. BPMN プロセス・コンポーネントをダブルクリックします。

BPMN プロセスが表示されます。

18. 「アクティビティ」アイコンをクリックし、「サービス」を選択します。

19. 図 8-53 に示すように、「Start」と「End」イベント・コンポーネント間を接続している線の上に「サービス」アイコンをドロップします。

図 8-53 「アクティビティ」アイコン



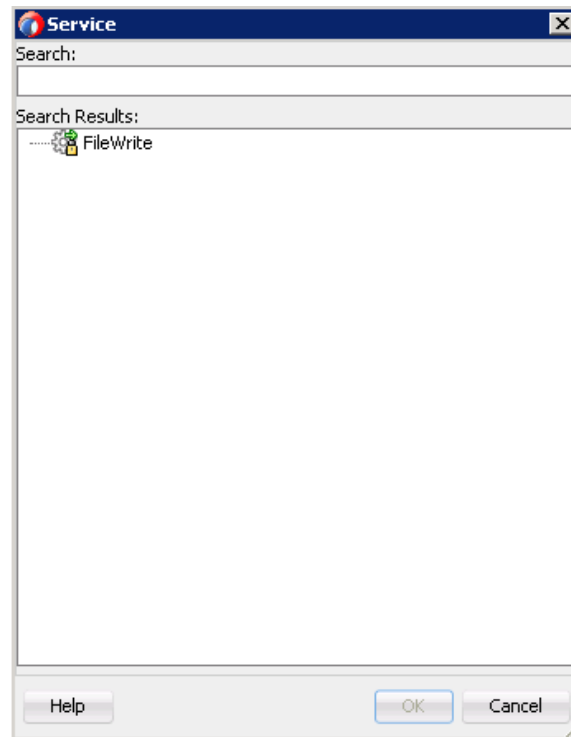
「プロパティ - ServiceTask」ダイアログが表示されます。

20. 「実装」タブをクリックします。
21. 「実装タイプ」リストで「ServiceTask」を選択します。
22. 「メッセージ交換」の「タイプ」リストで、サービス・コールを選択します。

23. 「サービス」フィールドの右側にある「参照」アイコンをクリックします。

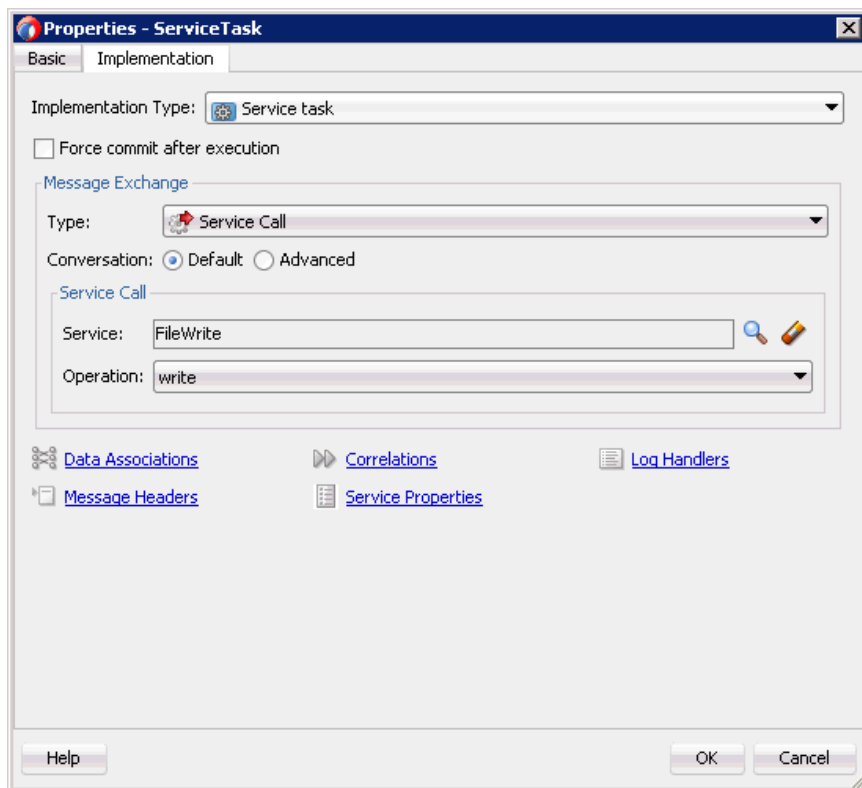
図 8-54 に示すように、「タイプ」ダイアログが表示されます。

図 8-54 「タイプ」ダイアログ



24. 作成された書き込み操作のサービスを選択し、「OK」をクリックします。  
図 8-55 に示すように、「プロパティ - ServiceTask」ダイアログに戻ります。

図 8-55 「プロパティ - ServiceTask」ダイアログ

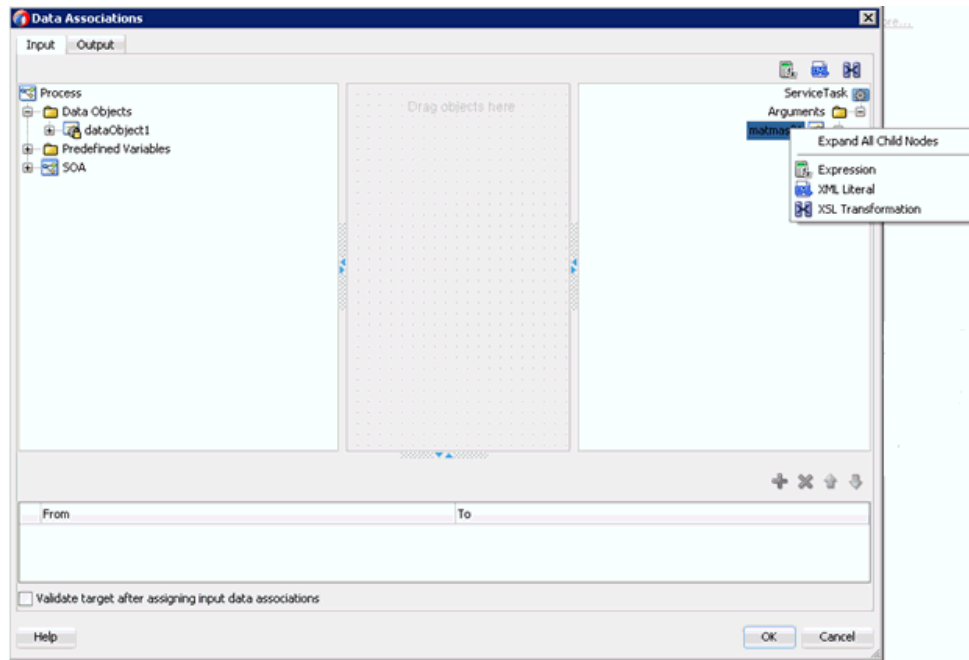


25. データ・アソシエーション・ハイパーリンクをクリックします。  
データ・アソシエーション・ダイアログが表示されます。



26. 図 8-56 に示すように、右ペインの「matmas01」引数を右クリックして XSL トランスフォーメーションを選択します。

図 8-56 XSL トランスフォーメーション



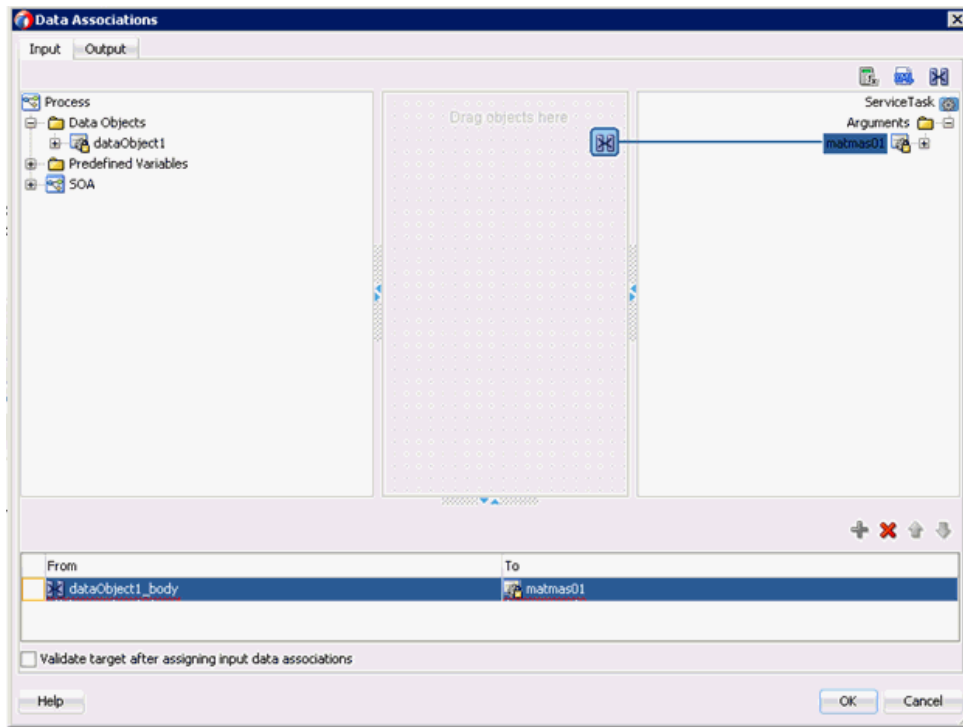
トランスフォーメーションの作成ダイアログが表示されます。

27. 作成したデータ・オブジェクトを「ソース」領域で選択し、右矢印アイコンをクリックします。これにより、作成したデータ・オブジェクトが「選択済」要素領域に追加されます。

28. 「OK」をクリックします。

☒ 8-57 に示すように、データ・アソシエーション・ダイアログに戻ります。

図 8-57 データ・アソシエーション・ダイアログ



29. 「OK」をクリックします。

「プロパティ - ServiceTask」ダイアログに戻ります。

30. 「OK」をクリックします。

「dataobject1\_body.xsl」タブが表示されます。

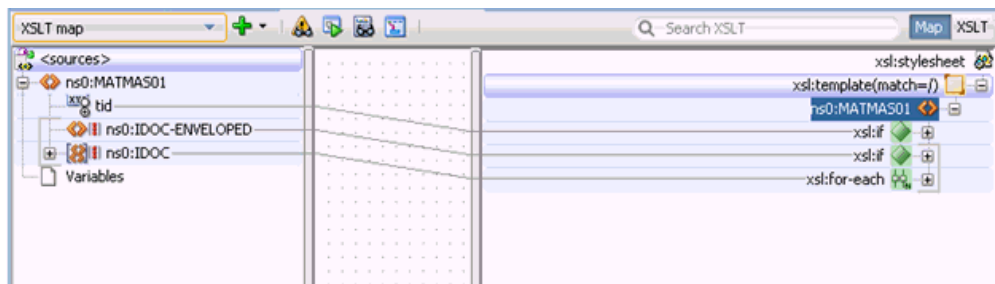
31. ソース要素とターゲット要素を自動マップします。

「自動マップ・プリファレンス」ダイアログが表示されます。

32. デフォルト値を受け入れて、「OK」をクリックします。

☒ 8-58 に示すように、トランスフォーメーションが完了しました。

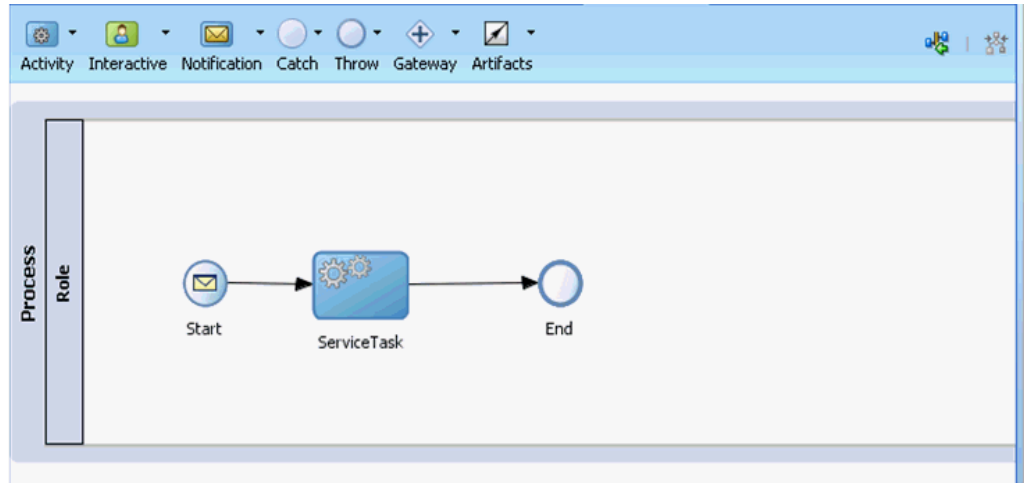
図 8-58 完了したトランスフォーメーション



33. トランスフォーメーションを保存します。

34. 図 8-59 に示すように、「プロセス」ワークスペース領域に戻ります。

図 8-59 「プロセス」ワークスペース領域



「ServiceTask」コンポーネントが「Start」イベント・コンポーネントと「End」イベント・コンポーネントの間に作成されます。

35. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規インバウンド BPM プロセス・コンポーネントを保存します。

#### 8.5.2.4 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの 6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」を参照してください。

これで、インバウンド BPM プロセスをデプロイする準備ができました。6-59 ページの「BPEL インバウンド・プロセスのデプロイ」と同じ手順を実行できます。SAP R/3 でイベントをトリガーする方法の詳細は、6-60 ページの「SAP R/3 でのイベントのトリガー」を参照してください。

## 8.6 BPM を使用した同期 SAP R/3 イベントの処理 (J2CA 構成)

この項では、BPM を使用して同期 SAP R/3 イベントを処理する方法について説明します。同期 SAP R/3 イベントの詳細は、6.6.1 項「概要」を参照してください。

---

**注意：** 同期イベントは、返信スキーマを含むオブジェクトについてのみサポートされており、IDoc についてはサポートされていません。

---

アプリケーション・アダプタ・インストールの次のフォルダに、このユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPM\J2CA\Synchronous_Events
```

この項では、次の項目について説明します。

- 8.6.1 項「アダプタ・ターゲットおよびチャネルの構成」
- 8.6.2 項「メッセージの WSDL ファイルの生成」
- 8.6.3 項「JDeveloper での BPM プロセスの作成」

## 8.6.1 アダプタ・ターゲットおよびチャネルの構成

アプリケーション・エクスペローラを開き、J2CA 構成を使用して SAP 用に 2 つのターゲットおよび 1 つのチャネルを作成します。

詳細は、第 4 章「構成 Oracle Application Adapter for SAP R/3」を参照してください。

## 8.6.2 メッセージの WSDL ファイルの生成

メッセージの WSDL ファイルを生成するには、次のようにします。

1. WSDL ファイルを生成する前に、チャネルが開始されていないことを確認します。
2. 最初の SAP アダプタ・ターゲット (target1) に接続します。
3. BAPI\_COMPANYCODE\_GETLIST のインバウンド WSDL ファイルを生成します (BAPI\_COMPANYCODE\_GETLIST は、「リモート関数モジュール」→「Financial Accounting」→「0002 - Company Code Business Object」を使用してナビゲート)。
4. 2 番目の SAP アダプタ・ターゲット (target2) に接続します。
5. BAPI\_COMPANYCODE\_GETLIST のアウトバウンド WSDL ファイルを生成します (BAPI\_COMPANYCODE\_GETLIST は、「リモート関数モジュール」→「Financial Accounting」→「0002 - Company Code Business Object」を使用してナビゲート)。

詳細は、6.4.1 項「リクエスト/レスポンス・サービスの WSDL の生成」および 6.5.1 項「イベント統合のための WSDL の生成」を参照してください。

## 8.6.3 JDeveloper での BPM プロセスの作成

この項では、JDeveloper で BPM プロセスを作成する方法について説明します。

### 8.6.3.1 BPM 用の空のコンポジットの作成

BPM 用の空のコンポジットを作成するには、次のようにします。

1. 新規 BPM アプリケーションを作成します。
2. 新規 BPM アプリケーションの名前を入力し、「次へ」をクリックします。  
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクト名を入力し、「次へ」をクリックします。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

詳細は、6-11 ページの 6.4.2 項「SOA 用の空のコンポジットの作成」を参照してください。

### 8.6.3.2 インバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスの作成

インバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスを作成するには、次のようにします。

1. 「サード・パーティ・アダプタ」コンポーネントを「カスタム/サード・パーティ」領域から「公開されたサービス」ペインにドラッグ・アンド・ドロップします。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。
2. 「名前」フィールドに名前を入力し、「WSDL URL」フィールドの横にあるアイコンをクリックします。  
「WSDL の選択」ダイアログが表示されます。
3. 「ファイルシステム」タブを選択し、WSDL の場所でインバウンド WSDL ファイルを参照して選択し、「OK」をクリックします。

4. 「ファイルのローカライズ」ダイアログで、「OK」をクリックします。  
インバウンド WSDL、リクエストおよびレスポンス XML スキーマ・ファイル (.xsd) が、作成したプロジェクト・フォルダにインポートされます。
5. 「JCA ファイルを検索します。」アイコンをクリックして、インバウンド JCA プロパティ・ファイルを選択します。  
トランスフォーメーション・チューザ・ダイアログが表示されます。
6. WSDL の場所から JCA プロパティ・ファイルを選択し、「OK」をクリックします。  
「ファイルのコピー」メッセージ・ウィンドウが表示されます。
7. 「はい」をクリックします。  
プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。
8. 「OK」をクリックします。  
図 8-60 に示すように、「公開されたサービス」ペインにインバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスが作成されます。

図 8-60 「公開されたサービス」ペイン



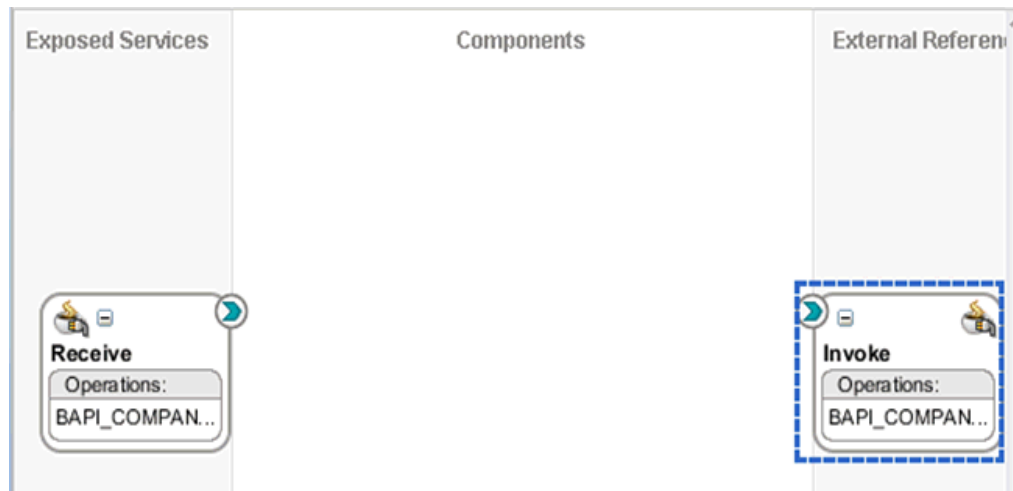
### 8.6.3.3 アウトバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスの作成

アウトバウンド WSDL ファイルのサード・パーティ・アダプタ・サービスを作成するには、次のようにします。

1. 「サード・パーティ・アダプタ」コンポーネントを「カスタム/サード・パーティ」領域から「外部参照」ペインにドラッグ・アンド・ドロップします。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。
2. 「名前」フィールドに名前を入力し、「WSDL URL」フィールドの横にあるアイコンをクリックします。  
「WSDL の選択」ダイアログが表示されます。
3. WSDL の場所で作成済アウトバウンド WSDL を参照して選択し、「OK」をクリックします。  
「ファイルのローカライズ」ダイアログが表示されます。
4. 「OK」をクリックします。  
アウトバウンド WSDL、リクエストおよびレスポンス XML スキーマ・ファイル (.xsd) が、作成したプロジェクト・フォルダにインポートされます。

5. 「JCA ファイルを検索します。」アイコンをクリックして、アウトバウンド JCA プロパティ・ファイルを選択します。
6. WSDL の場所から JCA プロパティ・ファイルを選択し、「OK」をクリックします。
7. 「ファイルのコピー」メッセージ・ウィンドウが表示されます。
8. 「はい」をクリックします。  
プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。
9. 「OK」をクリックします。  
 8-61 に示すように、「外部参照」ペインにアウトバウンド WSDL 用のサード・パーティ・アダプタ・サービスが作成されます。

図 8-61 「外部参照」ペイン



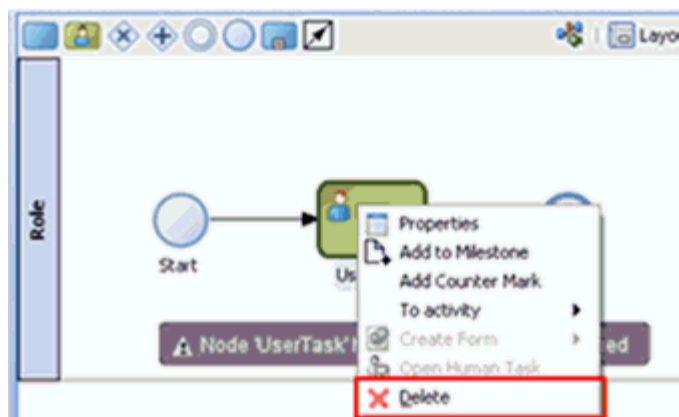
#### 8.6.3.4 同期 BPM プロセスの作成

同期 BPMN プロセスを作成するには、次のようにします。

1. 「BPMN プロセス」コンポーネントを「コンポーネント」領域から「コンポーネント」ペインにドラッグ・アンド・ドロップします。  
BPMN2.0 プロセス・ウィザード・ウィンドウが表示されます。
2. 「タイプ」セクションで**手動プロセス**を選択し、「終了」をクリックします。  
BPM プロセス・ワークスペース領域が表示されます。

3. 図 8-62 に示すように、「UserTask」アイコンを右クリックして、メニューの「削除」を選択します。

図 8-62 「削除」オプション



4. 「Start」 イベント・コンポーネントをダブルクリックします。  
「プロパティ - Start」 ダイアログが表示されます。
5. 「実装」 タブをクリックします。
6. 「実装タイプ」 リストで「メッセージ」をクリックし、「メッセージ交換」タブの「タイプ」リストで「インタフェースの使用」を選択します。
7. インタフェースの使用タブの下の「参照」フィールドの右側にある「参照」アイコンをクリックします。  
「サービス」ダイアログが表示されます。
8. インバウンド WSDL 用に作成されたサード・パーティ・サービスを選択し、「OK」をクリックします。  
「プロパティ - Start」ダイアログに戻ります。
9. データ・アソシエーションをクリックします。  
データ・アソシエーション・ダイアログが表示されます。
10. 右ペインの「プロセス」ノードの下の「データ・オブジェクト」を右クリックして、「新規」を選択します。  
データ・オブジェクトの作成ダイアログが表示されます。
11. 「名前」フィールドに名前を入力します (receive\_in など)。
12. 「タイプ」フィールドのドロップダウン・ボタンをクリックして、リストで「参照」を選択します。  
「タイプの参照」ダイアログが表示されます。
13. 対応する「リクエスト」コンポーネントを選択し、「OK」をクリックします。  
データ・オブジェクトの作成ダイアログに戻ります。
14. 「OK」をクリックします。  
作成したデータ・オブジェクト (receive\_in など) は、データ・アソシエーション・ダイアログの「データ・オブジェクト」の下に表示されます。
15. 左ペインの「引数」ノードの下に表示されているコンポーネントを選択してドラッグし、右ペインの「データ・オブジェクト」の下の作成済データ・オブジェクトに接続します。

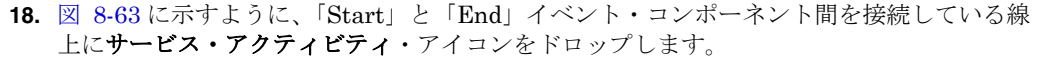
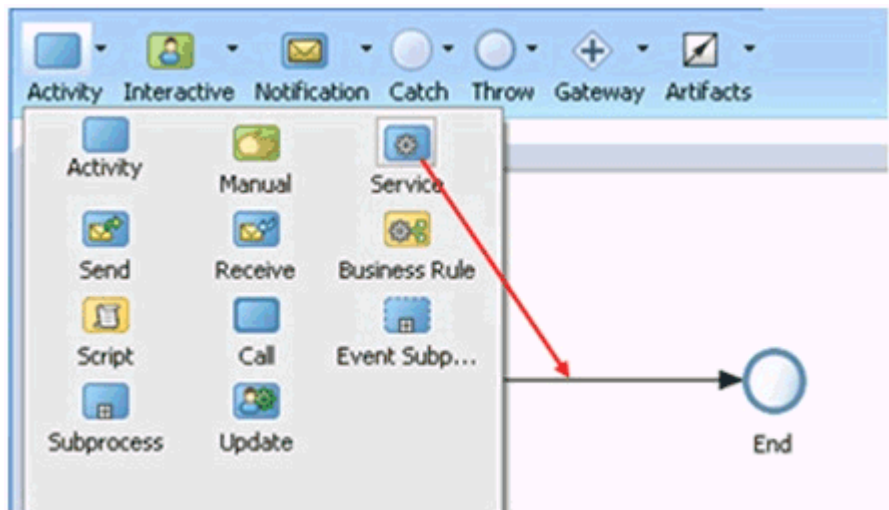
16. 「OK」をクリックします。  
「プロパティ - Start」ダイアログに戻ります。
17. 「OK」をクリックします。  
「プロセス」ワークスペース領域に戻ります。
18.  8-63 に示すように、「Start」と「End」イベント・コンポーネント間を接続している線の上にサービス・アクティビティ・アイコンをドロップします。

図 8-63 サービス・アクティビティ・アイコン



- 「プロパティ - ServiceTask」ダイアログが表示されます。
19. 「実装」タブをクリックし、「メッセージ交換」タブの「タイプ」リストで、サービス・コールを選択します。
  20. 「サービス」フィールドの右側にある「参照」アイコンをクリックします。  
「サービス」ダイアログが表示されます。
  21. アウトバウンド WSDL 用に作成されたサード・パーティ・サービスを選択し、「OK」をクリックします。  
「プロパティ - Start」ダイアログ・ウィンドウに戻ります。
  22. データ・アソシエーションをクリックします。  
データ・アソシエーション・ダイアログが表示されます。
  23. 入力領域で、左ペインの「プロセス」ノードの下の「データ・オブジェクト」を右クリックして、「新規」を選択します。  
データ・オブジェクトの作成ダイアログが表示されます。
  24. 「名前」フィールドに名前を入力し (invoke\_in など)、「タイプ」フィールドのドロップダウン・ボタンをクリックして、リストで「参照」を選択します。  
「タイプの参照」ダイアログが表示されます。
  25. 対応する「リクエスト」コンポーネントを選択し、「OK」をクリックします。  
データ・オブジェクトの作成ダイアログに戻ります。
  26. 「OK」をクリックします。  
作成したデータ・オブジェクト (invoke\_in など) は、データ・アソシエーション・ダイアログの「データ・オブジェクト」の下に表示されます。

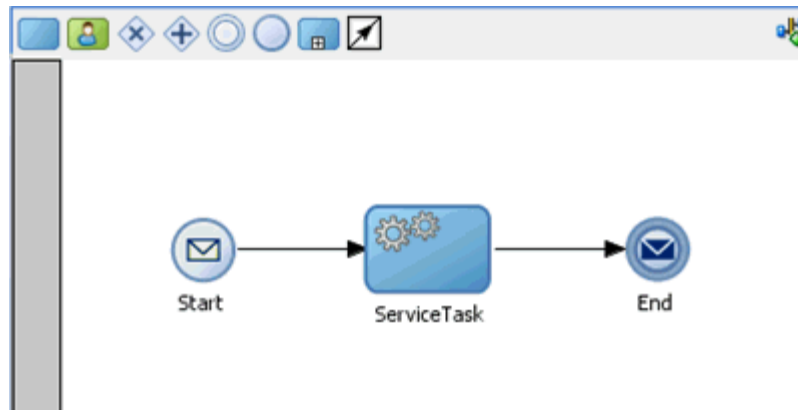


27. 左ペインの「データ・オブジェクト」の下で作成済データ・オブジェクトを選択してドラッグし、右ペインの「引数」ノードの下に表示されているコンポーネントに接続します。
28. 出力領域を選択し、右ペインの「プロセス」ノードの下に「データ・オブジェクト」を右クリックして、「新規」を選択します。  
データ・オブジェクトの作成ダイアログが表示されます。
29. 「名前」フィールドに名前を入力し (invoke\_out など)、「タイプ」フィールドのドロップダウン・ボタンをクリックして、リストで「参照」を選択します。  
「タイプの参照」ダイアログが表示されます。
30. 対応する「レスポンス」コンポーネントを選択し、「OK」をクリックします。  
データ・オブジェクトの作成ダイアログに戻ります。
31. 「OK」をクリックします。  
作成したデータ・オブジェクト (invoke\_out など) は、データ・アソシエーション・ダイアログの「データ・オブジェクト」の下に表示されます。
32. 左ペインの「引数」ノードの下に表示されているコンポーネントを選択してドラッグし、右ペインの「データ・オブジェクト」の下で作成済データ・オブジェクト (Response) に接続します。
33. 「OK」をクリックします。  
「プロパティ - Start」ダイアログに戻ります。
34. 「OK」をクリックします。  
「プロセス」ワークスペース領域に戻ります。
35. 「End」イベント・コンポーネントをダブルクリックします。  
「プロパティ - End」ダイアログが表示されます。
36. 「実装」タブをクリックします。
37. 「実装タイプ」リストで「メッセージ」を選択し、「メッセージ交換」タブの「タイプ」リストでインタフェースの使用を選択します。
38. インタフェースの使用の「タイプ」フィールドで、「同期」ボタンを選択します。
39. インタフェースの使用タブの下に「参照」フィールドの右側にある「参照」アイコンをクリックします。  
「サービス」ダイアログが表示されます。
40. インバウンド WSDL 用に作成されたサード・パーティ・サービスを選択し、「OK」をクリックします。
41. データ・アソシエーションをクリックします。  
データ・アソシエーション・ダイアログが表示されます。
42. 左ペインの「データ・オブジェクト」の下に表示されているレスポンス (invoke\_out など) の作成済データ・オブジェクトを選択してドラッグし、右ペインの「引数」の下に表示されているコンポーネントに接続します。
43. 「OK」をクリックします。  
「プロパティ - End」ダイアログに戻ります。

44. 「OK」をクリックしてから、「すべて保存」をクリックします。

図 8-64 に示すように、「プロセス」ワークスペース領域に戻ります。

図 8-64 「プロセス」ワークスペース領域



これで、同期イベントの BPM プロセスの作成が完了しました。

#### 8.6.3.5 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの 6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」を参照してください。

#### 8.6.3.6 BPM プロセスのデプロイ

BPM プロセスをデプロイする方法の詳細は、8-26 ページの 8.4.4 項「BPM アウトバウンド・プロセスのデプロイ」を参照してください。

## 8.7 トランスフォーメーションを使用したサービス統合のためのアウトバウンド BPM プロセスの設計 (BSE 構成)

この項では、Oracle JDeveloper で BPM プロジェクトを使用して SAP R/3 システムへの BPM アウトバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\BPM\BSE\Outbound_Project
```

アウトバウンド・デザインタイム構成を完了するには、次のツールが必要です。

- Oracle Adapter Application Explorer (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPM Designer (JDeveloper)

ここでは、次の項目について説明します。

- 8.7.1 項「BPM 用の空のコンポジットの作成」
- 8.7.2 項「BPM アウトバウンド・プロセスの定義」

#### 前提条件

BPM アウトバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6.7.1 項「Web サービスを使用したリクエストおよびレスポンス・サービスの WSDL ファイルの生成」を参照してください。

## 8.7.1 BPM 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには、次のようにします。

1. 新規 BPM アプリケーションを作成します。
2. BPM アプリケーションの名前を入力し、「次へ」をクリックします。
3. 「プロジェクト名」フィールドに名前を入力し、「次へ」をクリックします。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

詳細は、6-11 ページの 6.4.2 項「SOA 用の空のコンポジットの作成」を参照してください。

## 8.7.2 BPM アウトバウンド・プロセスの定義

この項では、BPM アウトバウンド・プロセスを定義する方法について説明します。ここでは、次の項目について説明します。

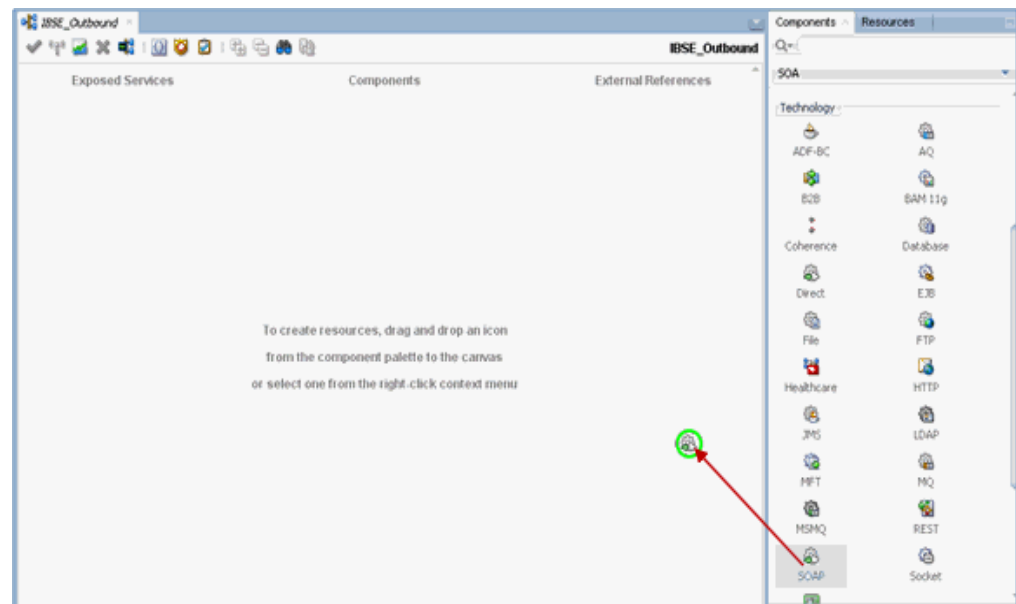
- 8.7.2.1 項「Web サービス・コンポーネントの構成」
- 8.7.2.2 項「BPM プロセス・コンポーネントの構成」
- 8.7.2.3 項「書込み操作作用のファイル・アダプタの作成」

### 8.7.2.1 Web サービス・コンポーネントの構成

Web サービス・コンポーネントを構成するには、次のようにします。

1. 作成されたプロジェクトをダブルクリックして、コンポーネントをロードします。
2. 図 8-65 に示すように、「Web サービス」ノードを「テクノロジー・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップします。

図 8-65 「Web サービス」ノード



3. Web サービスの適切な名前を入力し、「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。
4. 表示される「WSDL の選択」ウィンドウで、アプリケーション・エクスプローラから WSDL がエクスポートされる場所に移動し、WSDL を選択します。

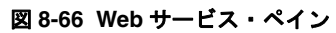
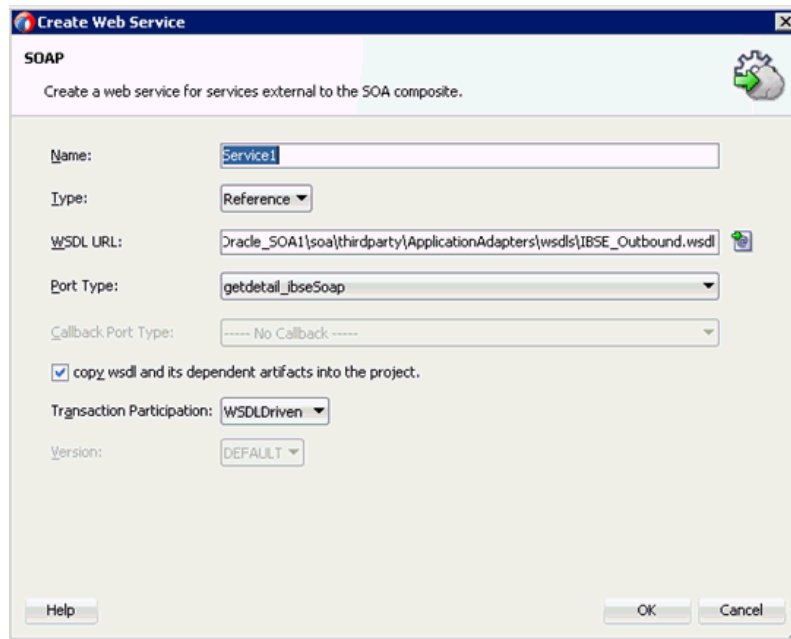
5. 「OK」をクリックします。
6.  8-66 に示すように、Web サービス・ペインで「OK」をクリックします。

図 8-66 Web サービス・ペイン



7. 表示される「ファイルのローカライズ」ウィンドウで、「OK」をクリックします。  
これにより、WSDL ファイルがプロジェクト・フォルダにインポートされます。

### 8.7.2.2 BPM プロセス・コンポーネントの構成

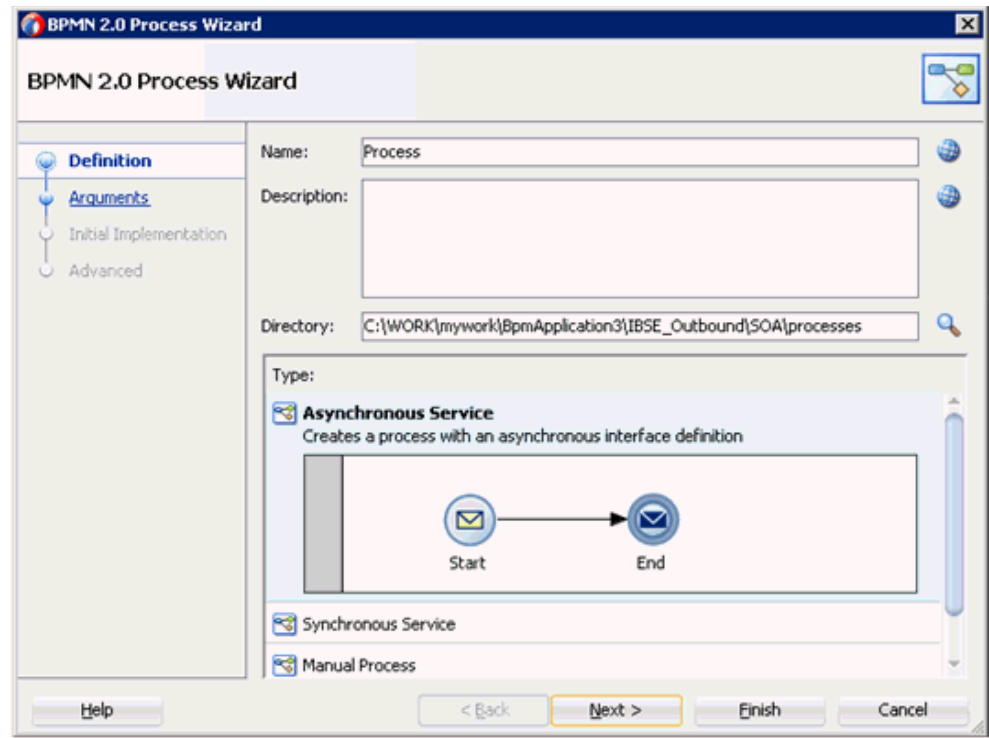
この項では、アウトバウンド BPM プロセス・コンポーネントを構成する方法について説明します。

BPM コンポーネントを構成するには、次のようにします。

1. 「BPMN プロセス」コンポーネントを「コンポーネント」ペインから「コンポーネント」ペインにドラッグ・アンド・ドロップします。

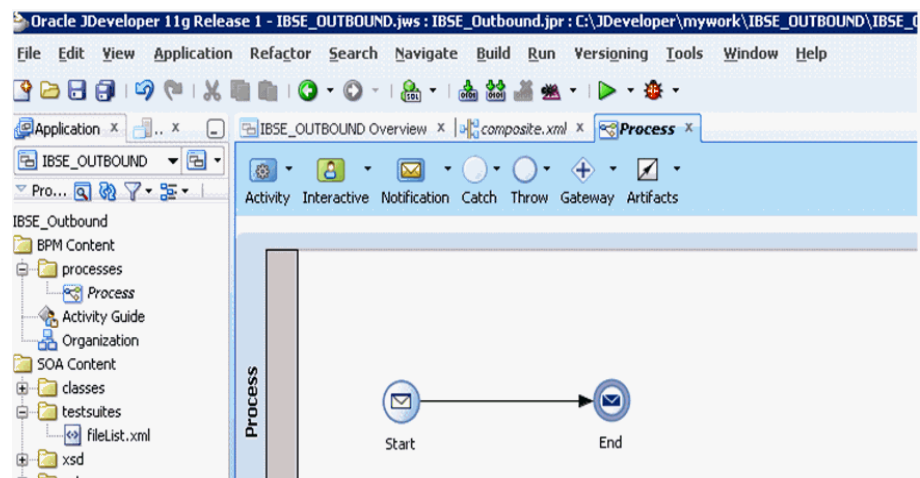
2. 図 8-67 に示すように、「タイプ」領域で選択されているデフォルト・オプション (非同期サービス) を受け入れ、「終了」をクリックします。

図 8-67 「タイプ」領域



3. 図 8-68 に示すように、「Start」イベント・コンポーネントをダブルクリックします。

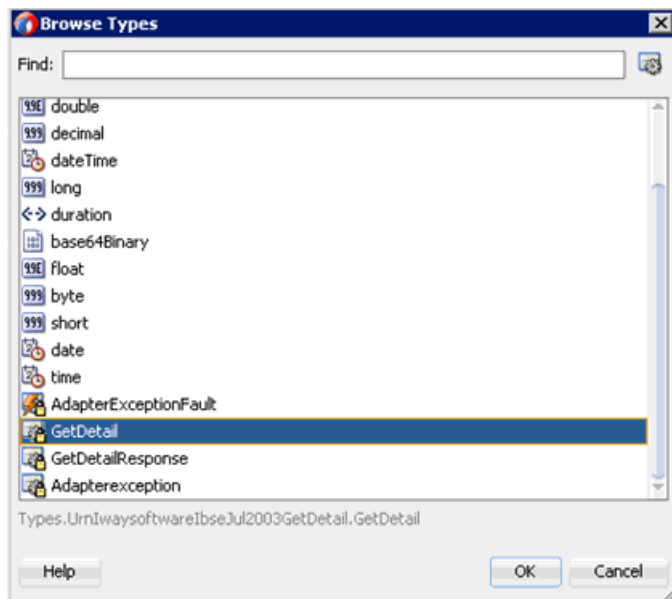
図 8-68 「Start」イベント・コンポーネント



4. 表示される「プロパティ - Start」ウィンドウで、「実装」タブをクリックします。
5. 引数定義フィールドの右側にあるプラス (+) アイコンをクリックします。  
「引数の編集」ウィンドウが表示されます。
6. 「名前」フィールドに名前を入力し、「タイプ」ドロップダウン・リストをクリックして「参照」を選択します。

7. 図 8-69 に示すように、「リクエスト」コンポーネント (GetDetail など) を選択して「OK」をクリックします。

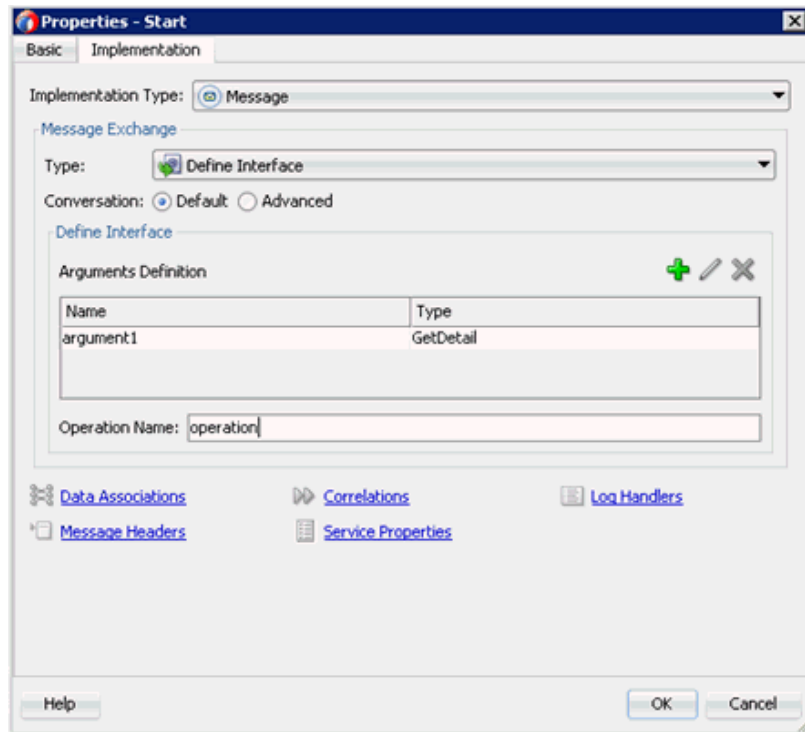
図 8-69 「リクエスト」コンポーネント



8. 表示される「引数の編集」ウィンドウで「OK」をクリックします。  
「プロパティ - Start」ウィンドウが表示されます。
9. 操作名フィールドで、デフォルト・エントリを **start** から **operation** に変更します。

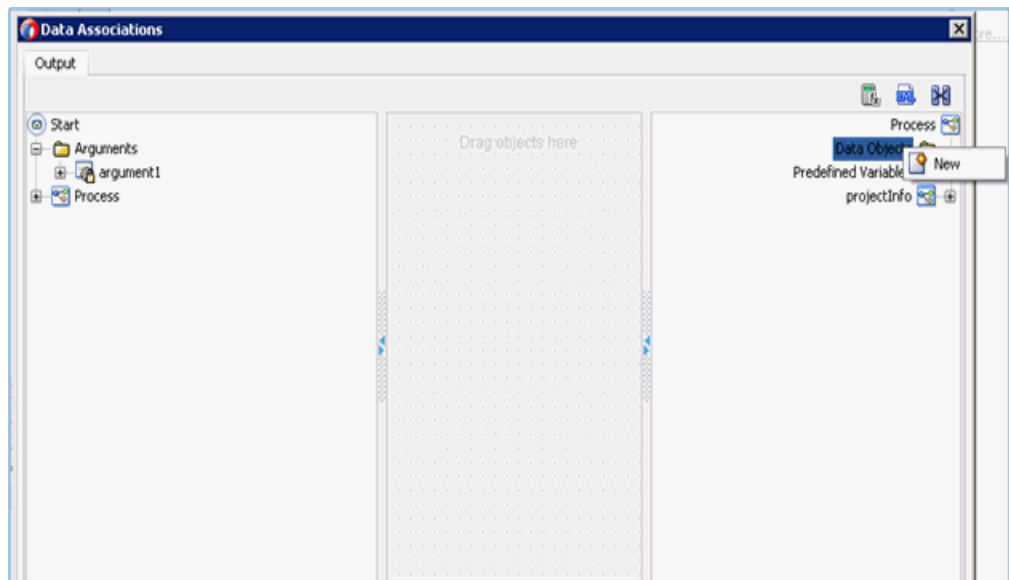
10. 図 8-70 に示すように、データ・アソシエーション・ハイパーリンクをクリックします。

図 8-70 「プロパティ - Start」 ウィンドウ



11. 図 8-71 に示すように、右ペインで、「プロセス」の下の「データ・オブジェクト」ノードを右クリックし、「新規」を選択します。

図 8-71 「データ・オブジェクト」ノード

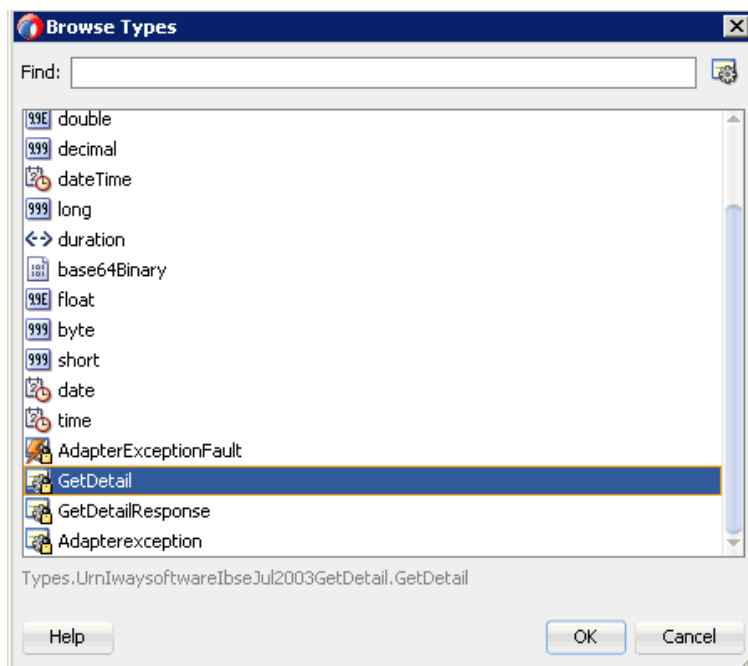


データ・オブジェクトの作成ウィンドウが表示されます。

12. 「名前」フィールドに名前を入力し、「タイプ」ドロップダウン・リストをクリックして「参照」を選択します。

13. 図 8-72 に示すように、「リクエスト」コンポーネント (GetDetail など) を選択して「OK」をクリックします。

図 8-72 「リクエスト」コンポーネント (GetDetail)

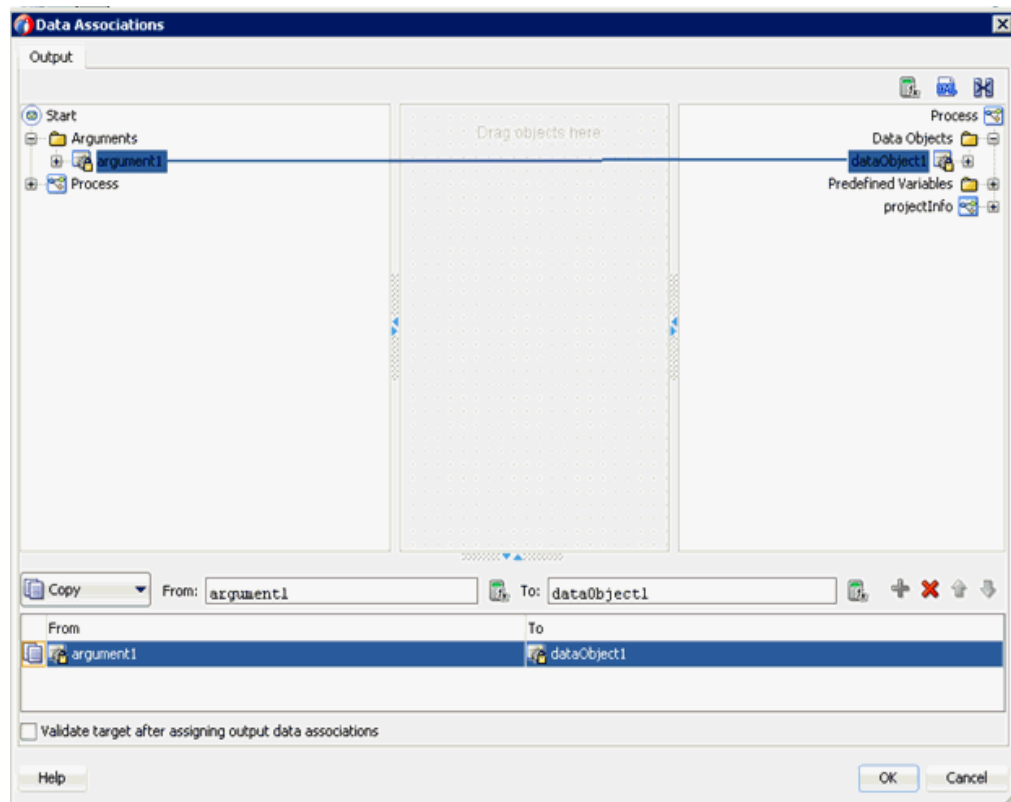


14. データ・オブジェクトの作成ウィンドウで、「OK」をクリックします。  
データ・アソシエーション・ウィンドウが表示されます。
15. 左ペインで「引数」ノードの下の「argument1」を選択してドラッグし、右ペインの「データ・オブジェクト」の下の「dataObject1」に接続します。



16. 図 8-73 に示すように、「OK」をクリックします。

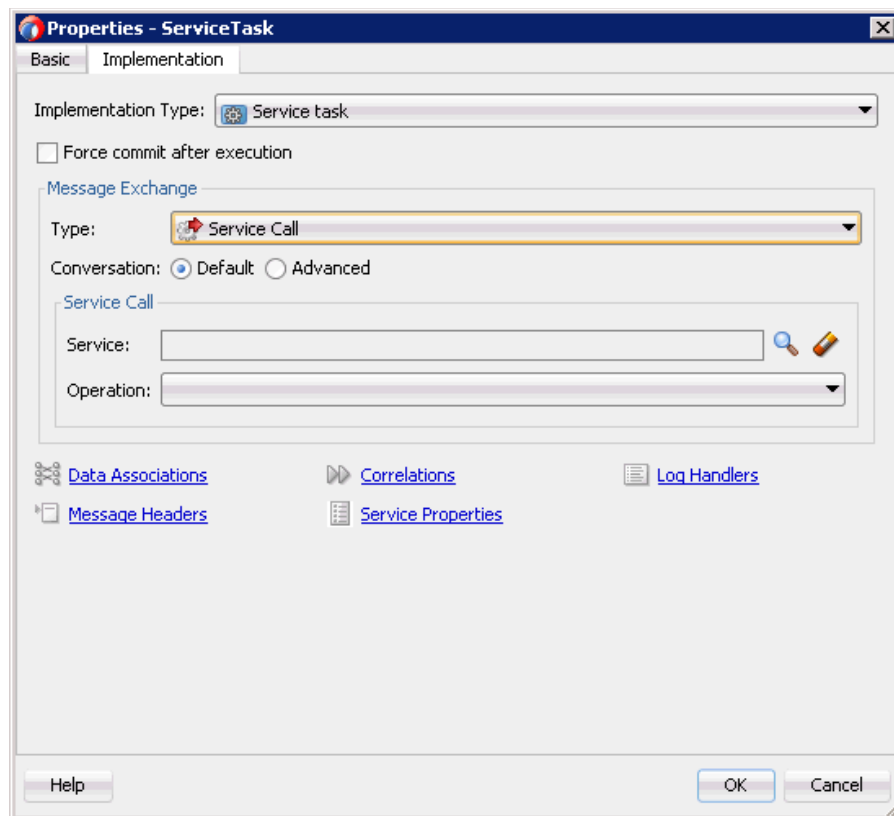
図 8-73 データ・アソシエーション



17. 表示される「プロパティ - Start」ウィンドウで「OK」をクリックします。  
「プロセス」ワークスペース領域に戻ります。
18. 「アクティビティ」ドロップダウン・メニューをクリックし、「サービス」を選択します。
19. 「サービス」アイコンを「Start」と「End」イベント・コンポーネント間を接続している線の上にドロップします。
20. 表示される「プロパティ - ServiceTask」ウィンドウで、「実装」タブをクリックします。
21. 「メッセージ交換」の「タイプ」リストで、サービス・コールを選択します。

22. 図 8-74 に示すように、「サービス」フィールドの右側にある「参照」アイコンをクリックします。

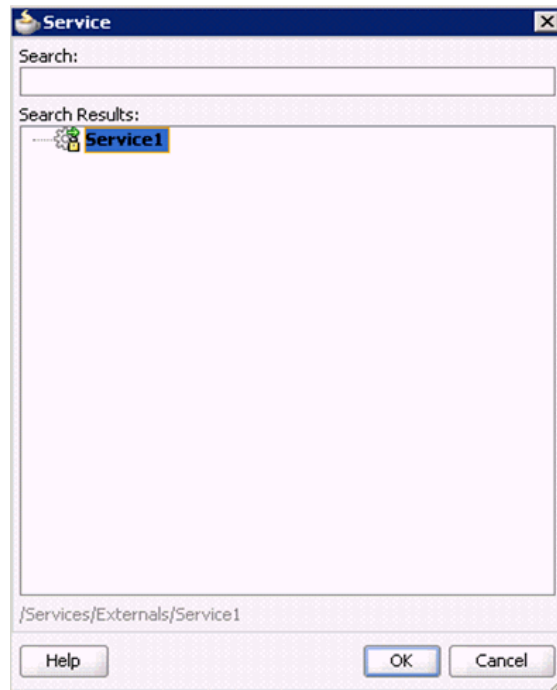
図 8-74 「参照」アイコン



「サービス」ウィンドウが表示されます。

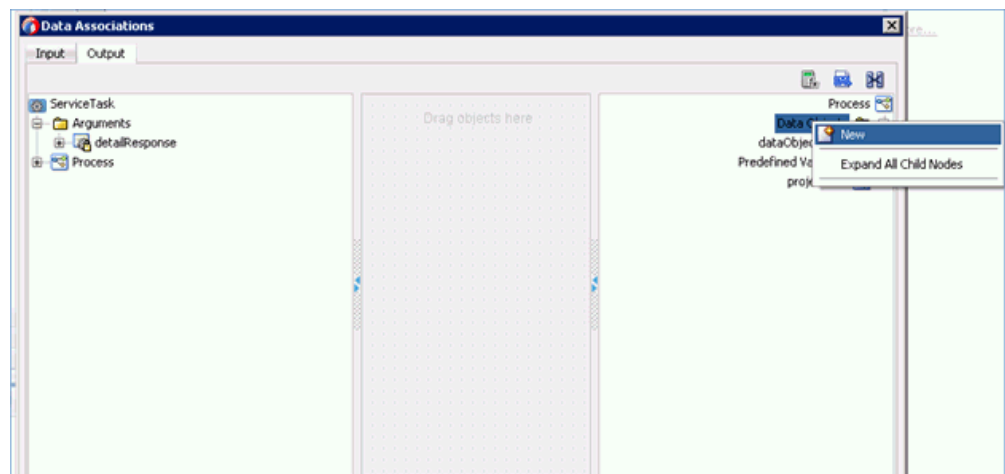
23. 図 8-75 に示すように、作成された Web サービスを選択し、「OK」をクリックします。

図 8-75 作成された Web サービス



24. 表示される「プロパティ - ServiceTask」ウィンドウで、データ・アソシエーション・ハイパーリンクをクリックします。  
データ・アソシエーション・ウィンドウが表示されます。
25. 図 8-76 に示すように、「出力」タブの右ペインにある「データ・オブジェクト」ノードを右クリックして「新規」を選択し、レスポンス・データ・オブジェクトを作成します。

図 8-76 「データ・オブジェクト」ノード



データ・オブジェクトの作成ウィンドウが表示されます。

26. 「名前」フィールドに名前を入力し、「タイプ」ドロップダウン・リストをクリックして「参照」を選択します。

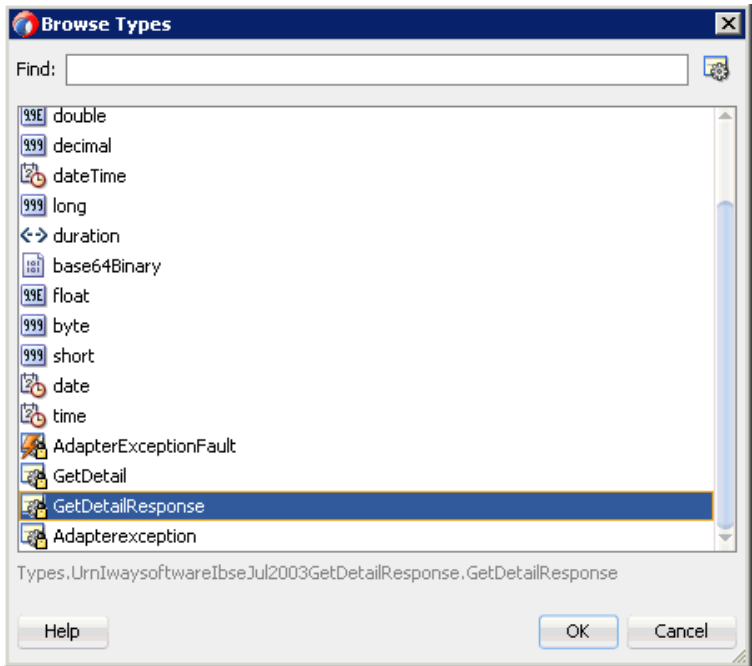
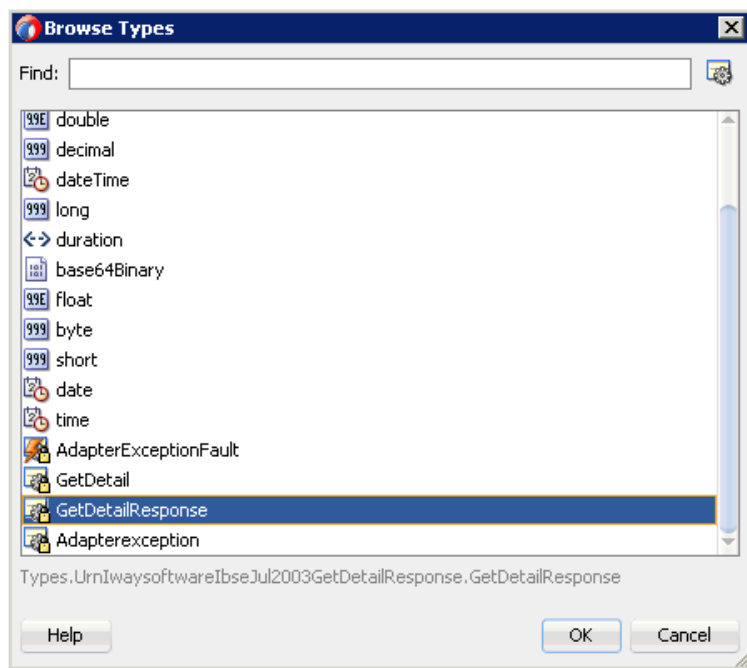
27.  8-77 に示すように、「レスポンス」コンポーネント (GetDetailResponse など) を選択して「OK」をクリックします。

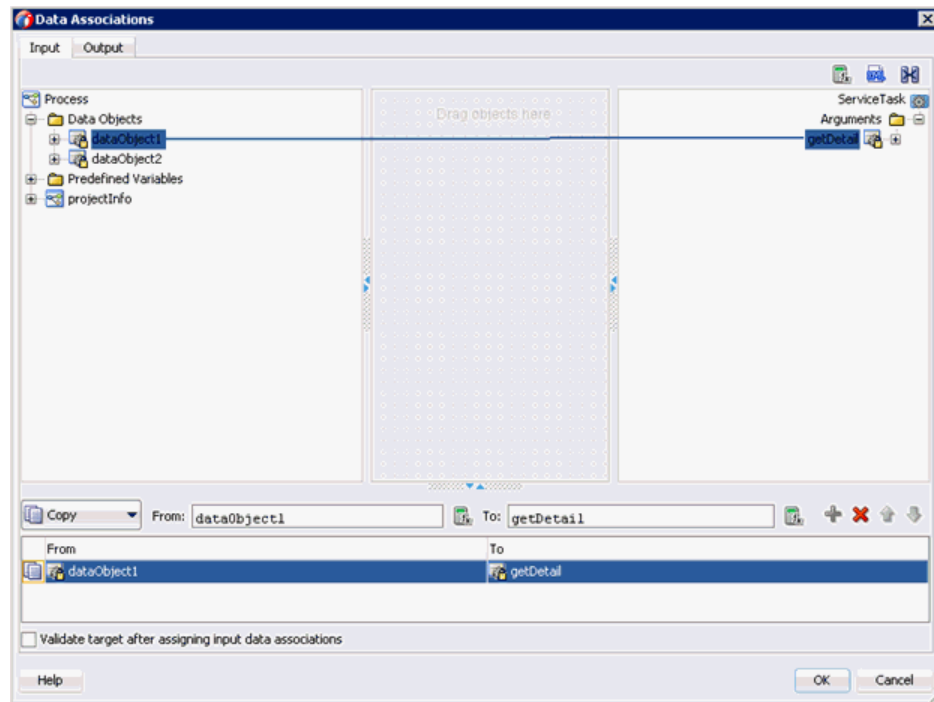
図 8-77 「レスポンス」コンポーネント (GetDetailResponse)



28. データ・オブジェクトの作成ウィンドウで、「OK」をクリックします。  
データ・アソシエーション・ウィンドウが表示されます。

29. 図 8-78 に示すように、「入力」タブの左ペインで、「データ・オブジェクト」ノードの下の「dataObject1」を選択してドラッグし、右ペインの「引数」ノードの下の「getDetail」ノードに接続します。

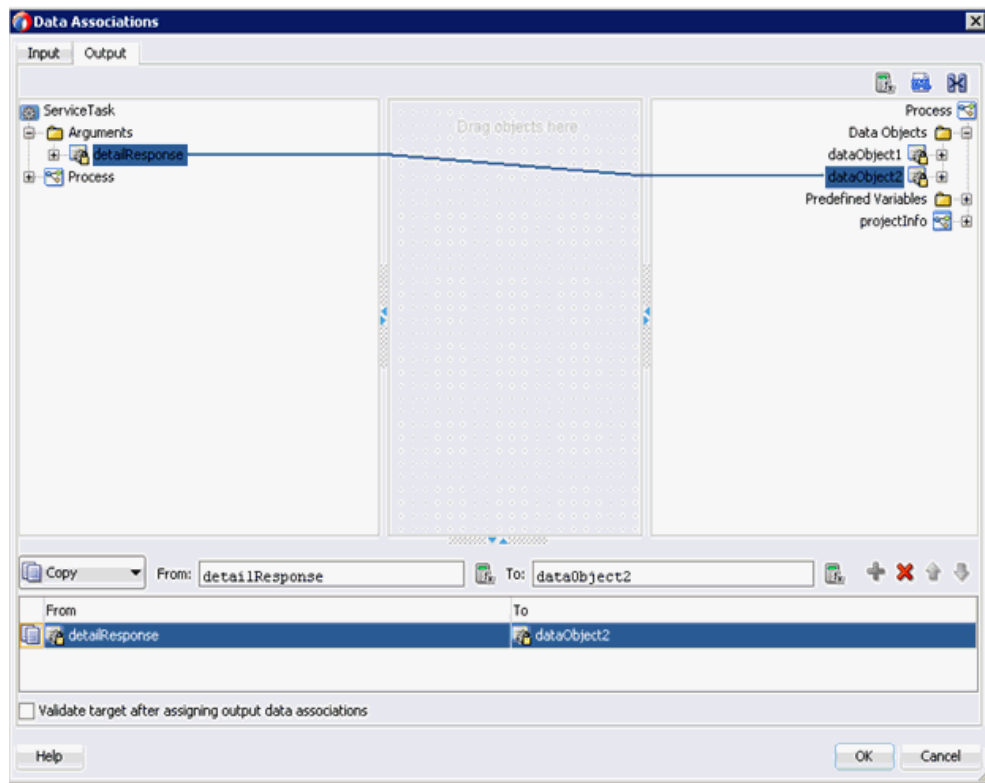
図 8-78 データ・アソシエーション



30. 「出力」タブをクリックし、左ペインの「引数」ノードの下の「detailResponse」を選択してドラッグし、「データ・オブジェクト」ノードの下の「dataObject2」に接続します。

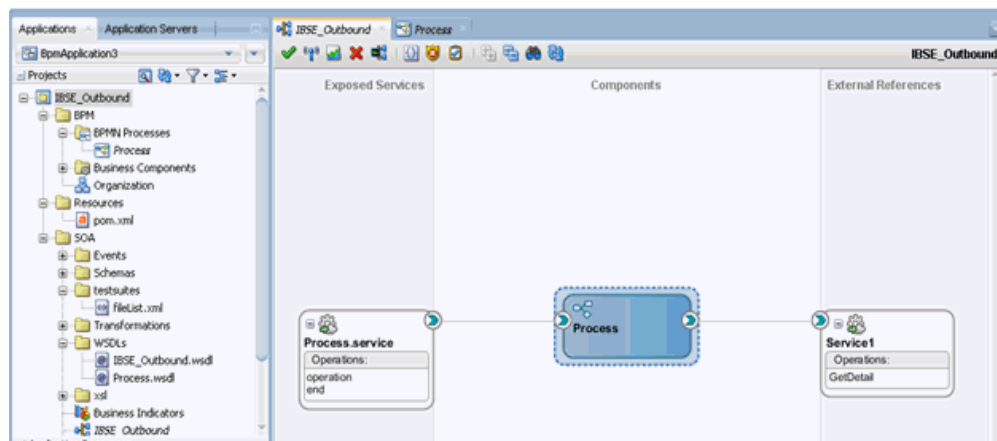
31. 図 8-79 に示すように、「OK」をクリックします。

図 8-79 「出力」タブ



32. 表示される「プロパティ - ServiceTask」ウィンドウで「OK」をクリックします。
33. メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド BPM プロセス・コンポーネントを保存します。
34. 図 8-80 に示すように、左ペインにある「IBSE\_Outbound」ノードをダブルクリックします。

図 8-80 「Composite.xml」ノード



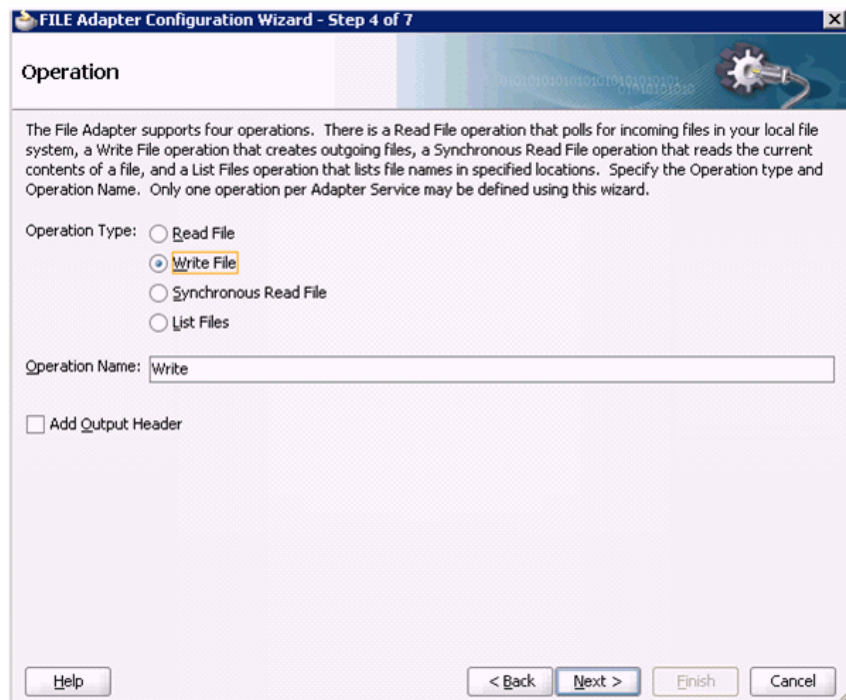
### 8.7.2.3 書込み操作のファイル・アダプタの作成

この項では、書込み操作のファイル・アダプタを作成する方法について説明します。

書込み操作のファイル・アダプタを作成するには、次のようにします。

1. 「ファイル・アダプタ」コンポーネントを「テクノロジー・アダプタ」ペインから「外部参照」ペインにドラッグ・アンド・ドロップし、ファイル・アダプタの名前を指定します。
2. 表示される「アダプタ・インタフェース」ペインで、「操作およびスキーマから定義(後で指定)」オプションが選択されていることを確認し、「次へ」をクリックします。
3. 「次へ」をクリックします。
4. 図 8-81 に示すように、表示される「操作」ペインで、「操作タイプ」オプションのリストから「ファイルの書込み」を選択し、「次へ」をクリックします。

図 8-81 「操作」ペイン

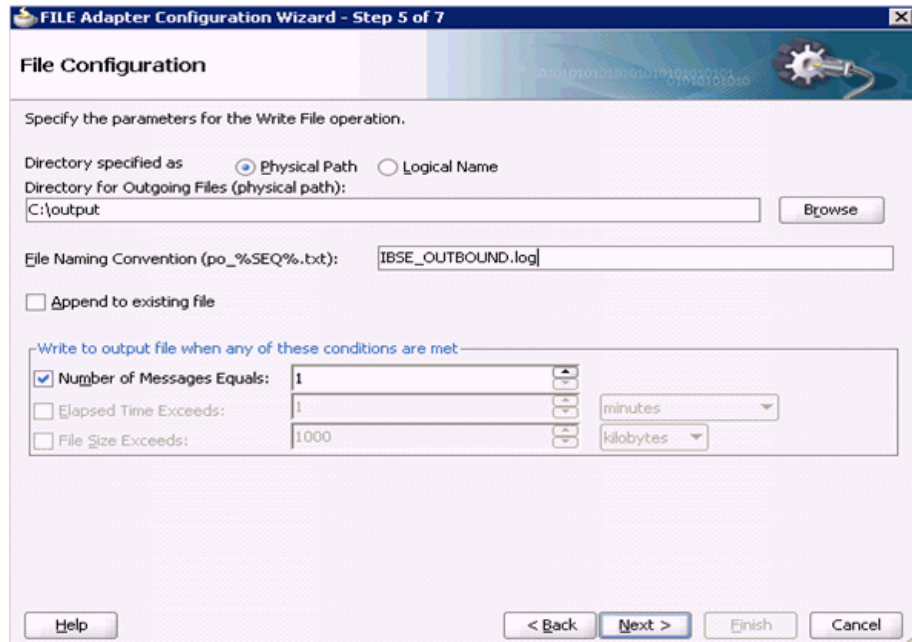


「ファイル構成」ペインが表示されます。

5. 「発信ファイルのディレクトリ(物理パス)」フィールドで、出力ファイルが書き込まれるファイル・システム上の場所を指定します。
6. ファイル・ネーミング規則フィールドで、出力ファイルの名前を指定します。

7. 図 8-82 に示すように、「次へ」をクリックします。

図 8-82 「ファイル構成」 ペイン



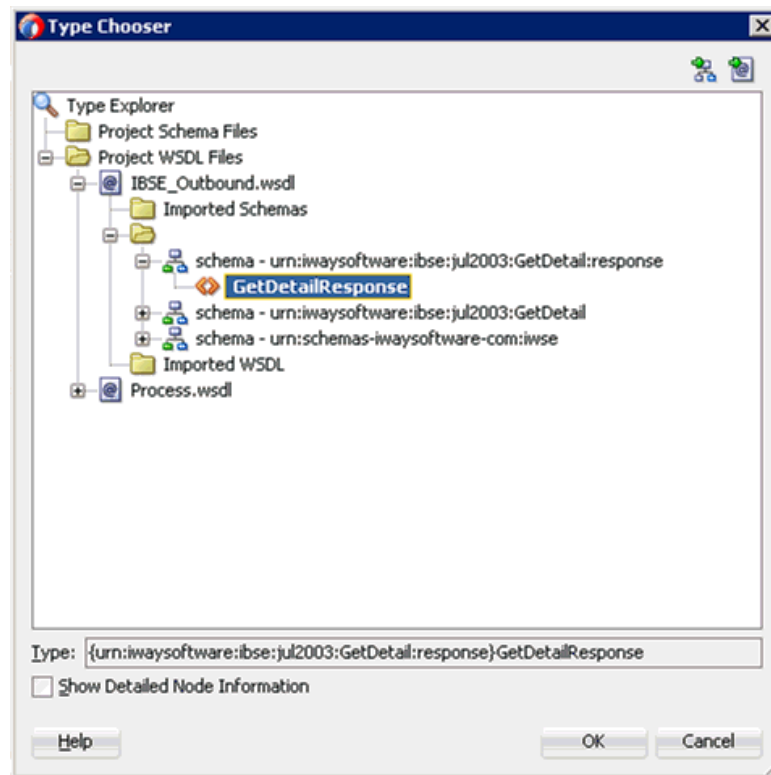
「メッセージ」 ペインが表示されます。

8. 「URL」 フィールドの右側にある「参照」をクリックします。
9. 表示される「タイプ・チューザ」 ウィンドウで、「プロジェクトの WSDL ファイル」 → 「IBSE\_Outbound.wsdl」 → 「インライン・スキーマ」を展開し、「GetDetailresponse」を選択します。



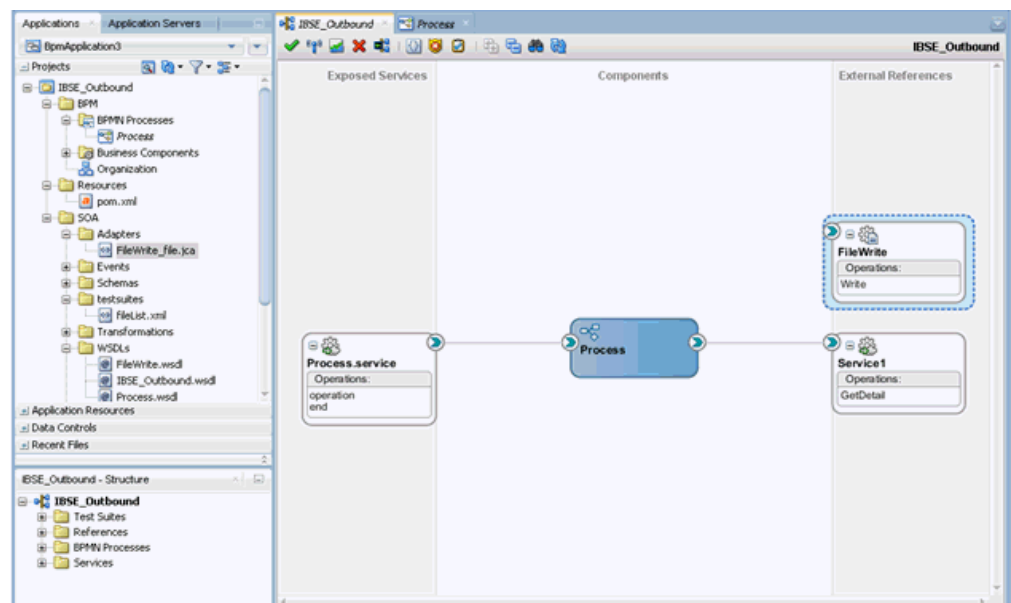
10. 図 8-83 に示すように、「OK」をクリックします。

図 8-83 「タイプ・チューザ」 ウィンドウ



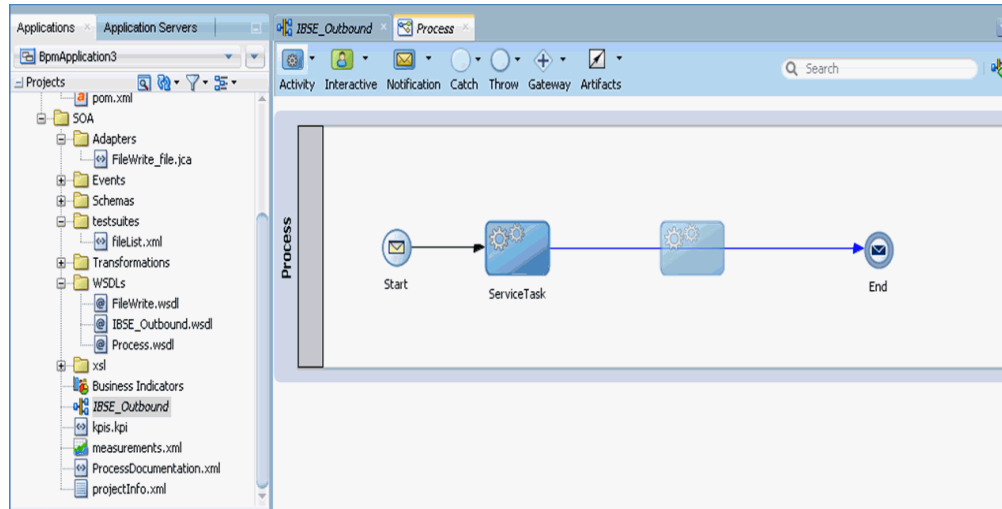
11. 「メッセージ」ペインで「次へ」をクリックします。
12. 表示される「終了」ペインで「終了」をクリックします。
13. 図 8-84 に示すように、「BPMN プロセス」コンポーネントをダブルクリックします。

図 8-84 「Composite.xml」 タブ



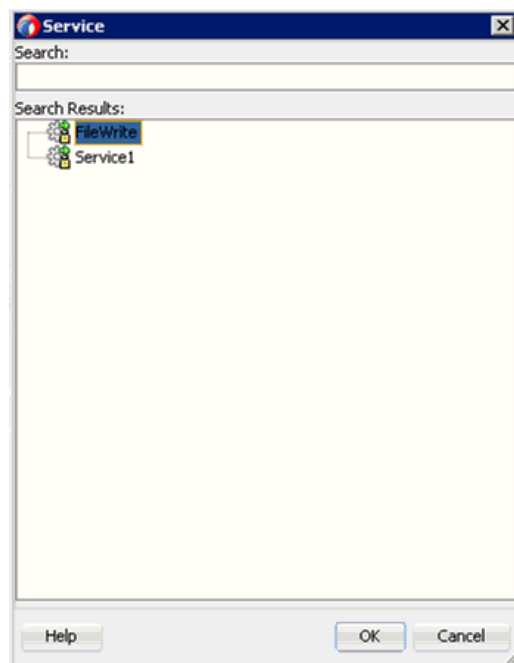
- 「アクティビティ」アイコンをクリックします。
- 図 8-85 に示すように、「アクティビティ」アイコンを「ServiceTask」と「End」イベント・コンポーネント間を接続している線上にドロップします。

図 8-85 「アクティビティ」アイコン



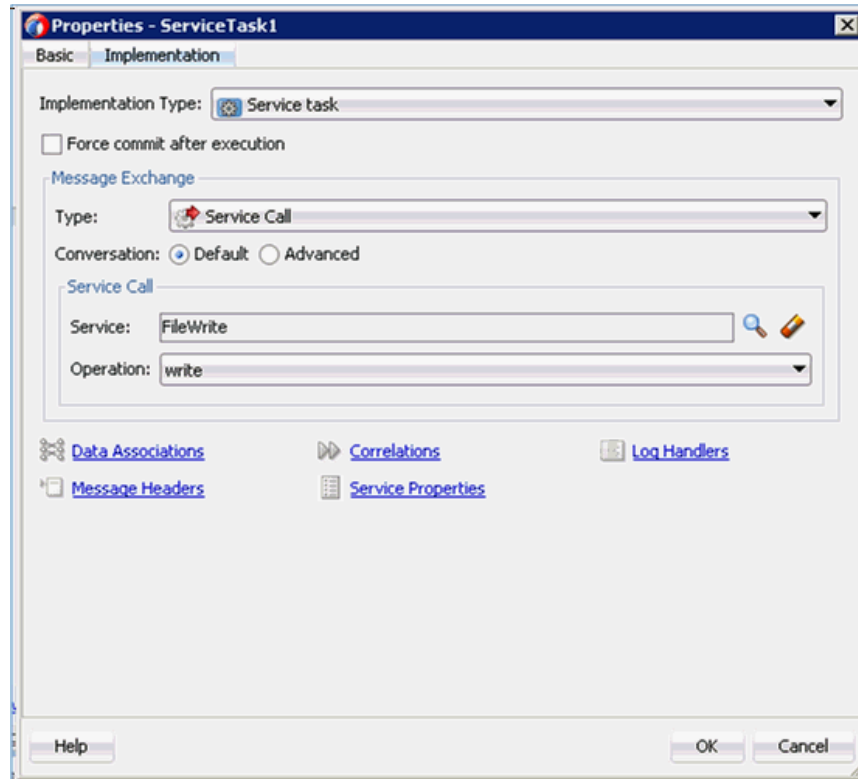
- 表示される「プロパティ - ServiceTask1」ウィンドウで、「実装」タブをクリックします。
- 「メッセージ交換」セクションの「タイプ」ドロップダウン・リストでサービス・コールを選択します。
- 「サービス」フィールドの右側にある「参照」アイコンをクリックします。
- 図 8-86 に示すように、作成された書込み操作のサービスを選択し、「OK」をクリックします。

図 8-86 「サービス」ウィンドウ



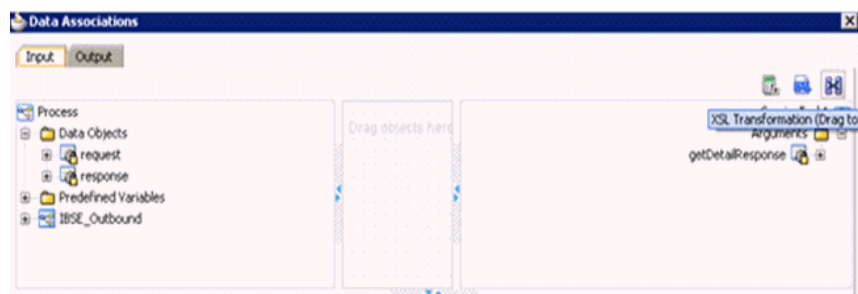
20. 図 8-87 に示すように、「プロパティ - ServiceTask1」ウィンドウで、データ・アソシエーション・ハイパーリンクをクリックします。

図 8-87 データ・アソシエーション



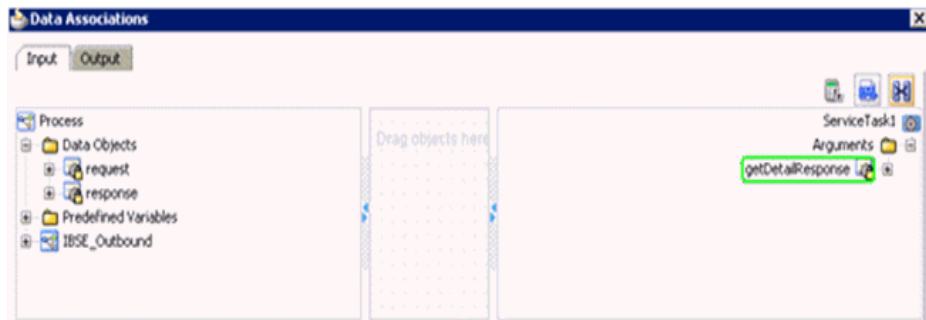
21. 図 8-88 に示すように、「入力」タブで、右ペイン上部にある XSL トランスフォーメーション・アイコンをクリックします。

図 8-88 XSL トランスフォーメーション・アイコン



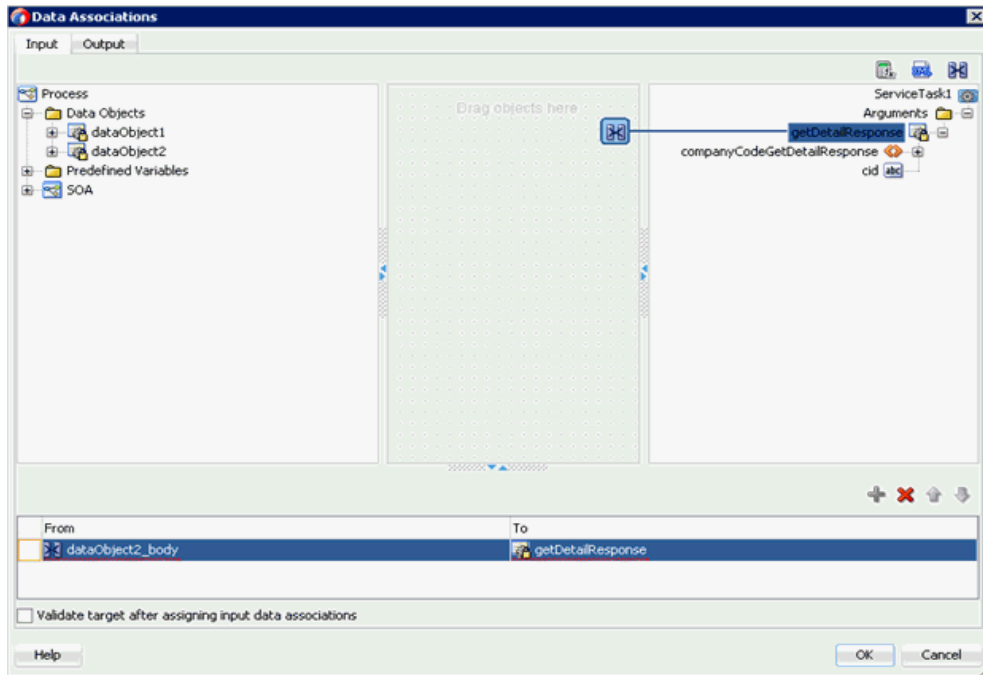
22. 図 8-89 に示すように、**XSL** トランスフォーメーション・アイコンを「getDetailResponse」ノードにドラッグ・アンド・ドロップします。

図 8-89 「GetDetailResponse」ノード



23. 表示される「トランスフォーメーションの作成」ウィンドウで、「ソース」セクションの「dataObject2」を選択し、右矢印記号をクリックします。
24. 「ターゲット」ドロップダウン・リストで選択されているデフォルト値および「作成」フィールドのデフォルト名を、「OK」をクリックして受け入れます。
25. 図 8-90 に示すように、データ・アソシエーション・ウィンドウで「OK」をクリックします。

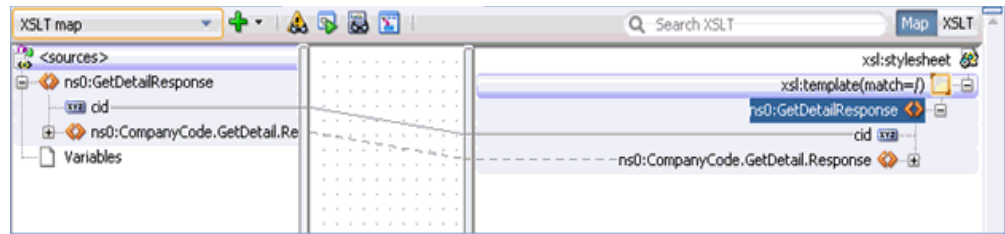
図 8-90 データ・アソシエーション・ウィンドウ



26. 「プロパティ - ServiceTask1」ウィンドウで「OK」をクリックします。
27. 「response\_body.xml」タブで、「ns:GetDetailResponse」ソース要素を「ns:GetDetailResponse」ターゲット要素にマップします。

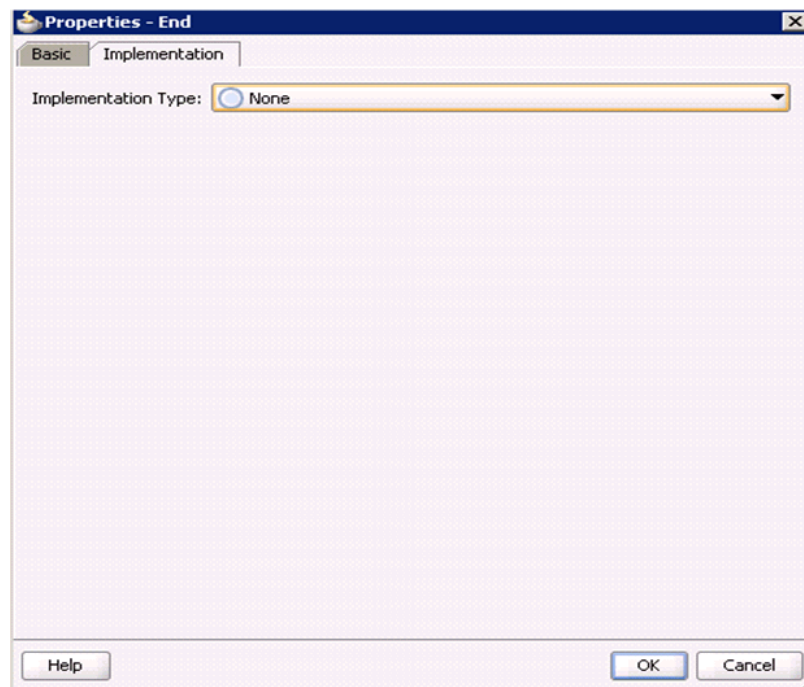
28. 図 8-91 に示すように、表示される「自動マップ・プリファレンス」ウィンドウで、デフォルト値を保持して「OK」をクリックします。

図 8-91 自動マップ・プリファレンス



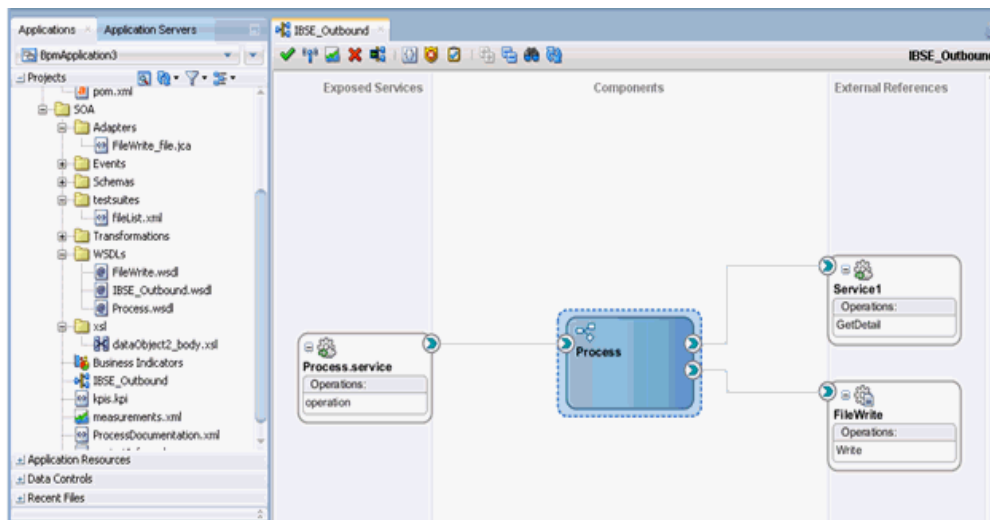
29. 「プロセス」ワークスペース領域に戻り、「End」イベント・コンポーネントをダブルクリックします。
30. 表示される「プロパティ - End」ウィンドウで、「実装」タブをクリックします。
31. 「実装タイプ」ドロップダウン・リストで「なし」を選択します。
32. 図 8-92 に示すように、「OK」をクリックします。

図 8-92 「実装」タブ



33. 図 8-93 に示すように、メニュー・バーの「すべて保存」アイコンをクリックし、構成した新規アウトバウンド BPM コンポーネントを保存します。

図 8-93 「すべて保存」アイコン



これで、BPM BSE アウトバウンド・プロセスをデプロイする準備ができました。8.4.4 項「BPM アウトバウンド・プロセスのデプロイ」と同じ手順を実行できます。

デプロイが完了したら、8.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」で定義されているように入力 XML を呼び出すことができます。

---

---

# sbconsole を使用した Oracle Service Bus の アウトバウンドおよびインバウンド・プロセス の構成

Oracle Application Adapter for SAP R/3 は Oracle Service Bus (OSB) とシームレスに統合され、これにより Web サービス統合が簡略化されます。OSB はサービス指向アーキテクチャ (SOA) に基づいています。これは Web Service Definition Language (WSDL) ドキュメントとして公開されているアダプタ・サービスを消費します。

この章は次の項で構成されています。

- 9.1 項「アプリケーション・アダプタと Oracle Service Bus との統合の概要」
- 9.2 項「sbconsole を使用したアウトバウンド・プロセスの構成 (J2CA 構成)」
- 9.3 項「sbconsole を使用したインバウンド・プロセスの構成 (J2CA 構成)」
- 9.4 項「sbconsole を使用したアウトバウンド・プロセスの構成 (BSE 構成)」
- 9.5 項「Oracle Service Bus を使用した同期 SAP R/3 イベントの使用 (J2CA 構成)」
- 9.6 項「Oracle Service Bus を使用した JMS プロキシ・サービスの構成 (J2CA 構成)」
- 9.7 項「Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成)」

## 9.1 アプリケーション・アダプタと Oracle Service Bus との統合の概要

Oracle Service Bus (OSB) と統合するには、Oracle Application Adapter for SAP R/3 は OSB と同じ Oracle WebLogic Server 内にデプロイされている必要があります。基礎となるアダプタ・サービスは WSDL ファイルとして公開されている必要があり、これらのファイルは、アダプタのリクエスト/レスポンス (アウトバウンド) とイベント通知 (インバウンド) の両方のサービスについて、Oracle Adapter Application Explorer (アプリケーション・エクスプローラ) でデザインタイムに生成されます。

## 9.2 sbconsole を使用したアウトバウンド・プロセスの構成 (J2CA 構成)

この項では、sbconsole を使用して、J2CA 構成用にアウトバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\J2CA\SAP_Sample_J2CA_OSB_Outbound_Project
```

この項では、次の項目について説明します。

- 9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」
- 9.2.2 項「アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定」
- 9.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ」
- 9.2.4 項「WSDL ベースのビジネス・サービスの構成」
- 9.2.5 項「ファイル・タイプ・ビジネス・サービスの構成」
- 9.2.6 項「プロキシ・サービスでのパイプラインの構成」

## 9.2.1 Oracle Service Bus の起動およびプロジェクト・フォルダの作成

この項では、Oracle Service Bus (OSB) を起動してプロジェクト・フォルダを作成する方法について説明します。

Oracle Service Bus を起動してプロジェクト・フォルダを作成するには、次のようにします。

1. 構成した Oracle WebLogic Server ドメインの Oracle WebLogic Server を起動します。
2. Web ブラウザで次の URL を入力し、Oracle Service Bus コンソールを開きます。

`http://hostname:port/sbconsole`

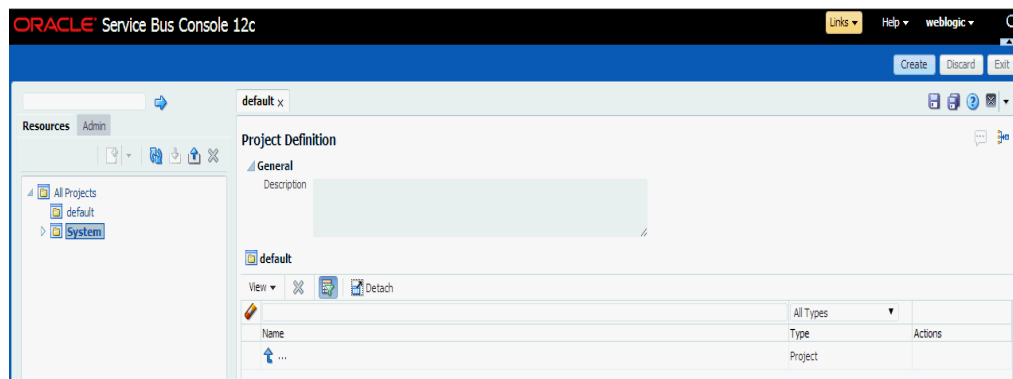
ここで、*hostname* は Oracle WebLogic Server を実行しているマシンの名前、*port* は使用しているドメインのポートです。

Oracle Service Bus コンソールのログオン・ページが表示されます。

3. 有効なユーザー名とパスワードを使用して、Oracle Service Bus コンソールにログオンします。

☒ 9-1 に示すように、Oracle Service Bus コンソールのホーム・ページが表示されます。

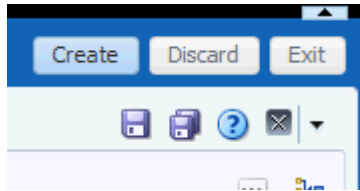
図 9-1 Oracle Service Bus コンソールのホーム・ページ





4. 図 9-2 に示すように、Oracle Service Bus セッションの右ペインの「作成」をクリックします。

図 9-2 Oracle Service Bus セッション



5. 図 9-3 に示すように、「すべてのプロジェクト」を選択し、左ペインで下矢印をクリックして「プロジェクト」を選択します。

図 9-3 「すべてのプロジェクト」フォルダ

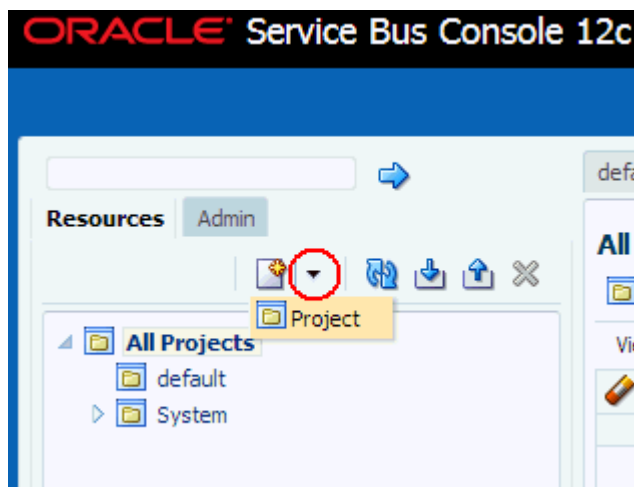
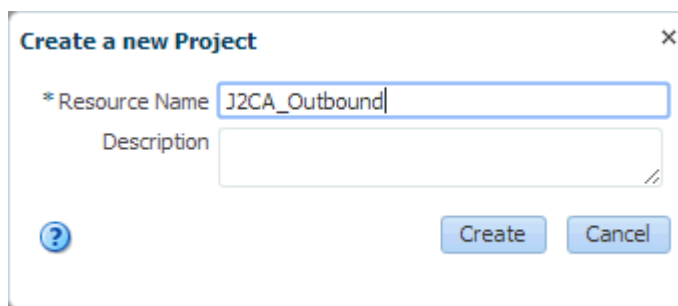


図 9-4 に示すように、「新規プロジェクトの作成」ウィンドウが表示されます。

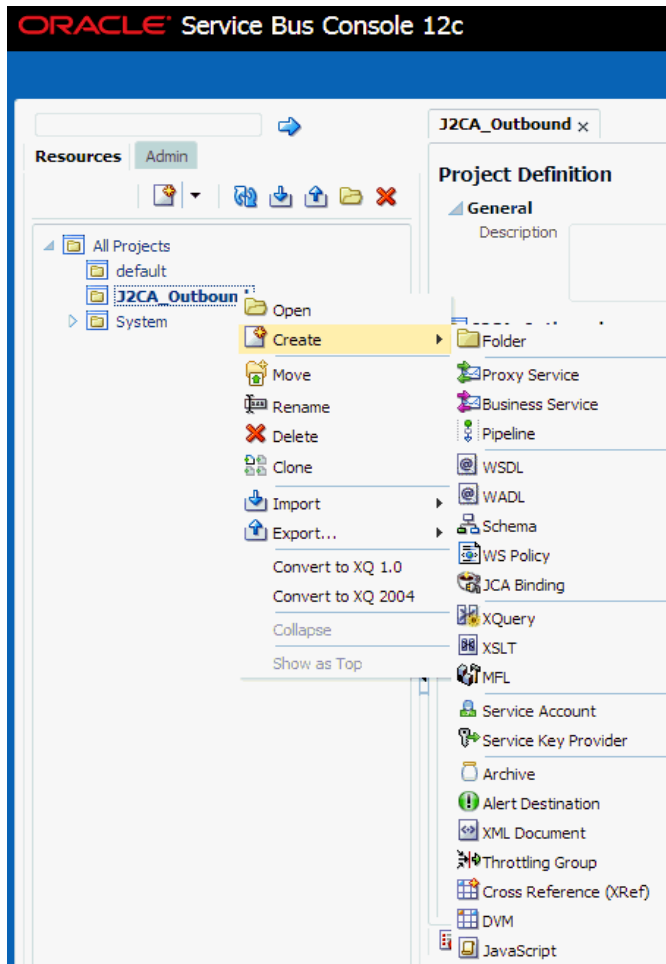
図 9-4 「新規プロジェクトの作成」ウィンドウ



6. 「リソース名」フィールドで、新しいプロジェクトの有効な名前 (J2CA\_Outbound など) を指定し、「作成」をクリックします。  
新しいプロジェクトが正常に作成されて表示されます。

7. 図 9-5 に示すように、新規作成されたプロジェクトを右クリックし、「作成」を選択して、「フォルダ」をクリックします。

図 9-5 「作成」オプション



「新規フォルダの作成」ウィンドウが表示されます。

8. 「リソース名」フィールドに **Business Service** と入力し、「作成」をクリックします。

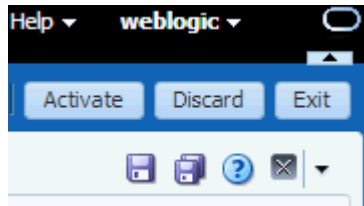
9. 手順 7 と 8 を繰り返して、**Proxy Service** および **Wsdls** という名前のフォルダを作成します。
- 図 9-6 に示すように、「Business Service」、「Proxy Service」および「Wsdls」の各フォルダが、左ペインのプロジェクト・ノードの下に表示されます。

図 9-6 プロジェクト・ノード



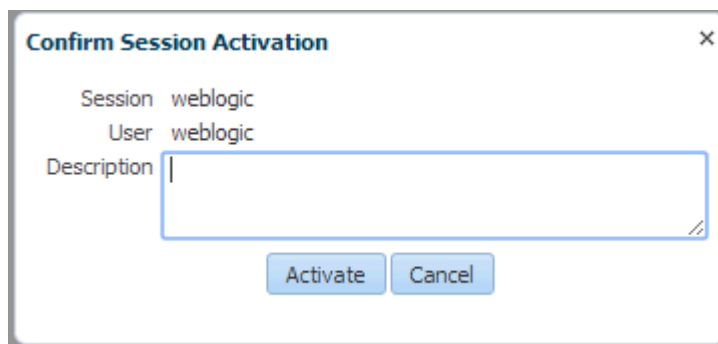
10. 図 9-7 に示すように、Oracle Service Bus セッションの右ペインの「アクティブ化」をクリックします。

図 9-7 「アクティブ化」ボタン



11. 図 9-8 に示すように、「セッションのアクティブ化の確認」ページで、「アクティブ化」をクリックして変更を保存します。

図 9-8 「セッションのアクティブ化の確認」ウィンドウ



## 9.2.2 アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定

アプリケーション・エクスプローラを起動し、これを使用して WSDL を Oracle Service Bus (OSB) コンソール (プロジェクト/フォルダ) に直接パブリッシュする前に、OSB ユーザーは次のようにする必要があります。

1. コマンド・プロンプト・ウィンドウを開きます。
2. 次のディレクトリにナビゲートします。  
`<ORACLE_HOME>\user_projects\domains\base_domain\bin`
3. **setDomainEnv.cmd** (Windows) または **./setDomainEnv.sh** (UNIX/Linux) を実行します。  
このコマンドにより、WSDL を OSB コンソールにパブリッシュするために、アプリケーション・エクスプローラが Oracle WebLogic Server API にアクセスするためのクラス・パスが設定されます。
4. コマンド・プロンプト・ウィンドウを開いたままにしておきます。
5. 次のディレクトリにナビゲートします。  
`<ADAPTER_HOME>\tools\iwae\bin`
6. **ae.bat** (Windows) または **iwae.sh** (UNIX/Linux) を実行して、アプリケーション・エクスプローラを起動します。  
これで、アプリケーション・エクスプローラから OSB コンソールに WSDL をパブリッシュする準備ができました。

## 9.2.3 アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ

アプリケーション・エクスプローラから Oracle Service Bus に WSDL をパブリッシュするには、次のようにします。

1. アプリケーション・エクスプローラを起動し、J2CA 構成に接続して、SAP R/3 ターゲットに接続します。  
詳細は、[第 4 章「構成 Oracle Application Adapter for SAP R/3」](#) を参照してください。
2. 「ビジネス・オブジェクト・リポジトリ」 → 「Financial Accounting」 → 「CompanyCode」を展開します。

3. 図 9-9 に示すように、**GetDetail** メソッドを右クリックして、メニューのアウトバウンド JCA サービス (リクエスト/レスポンス) の作成を選択します。

図 9-9 アウトバウンド JCA サービス (リクエスト/レスポンス) の作成オプション

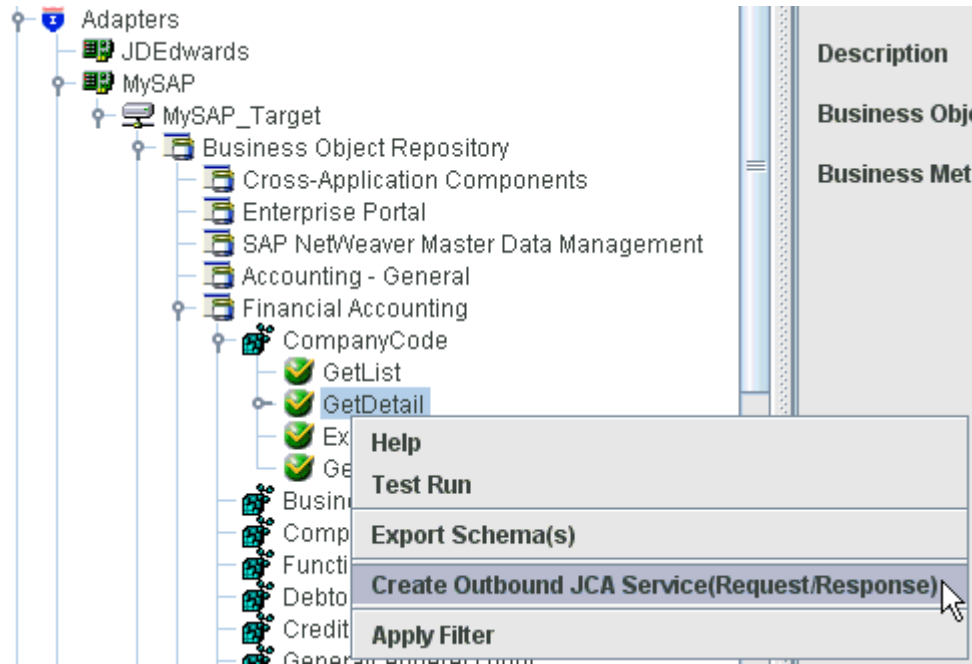
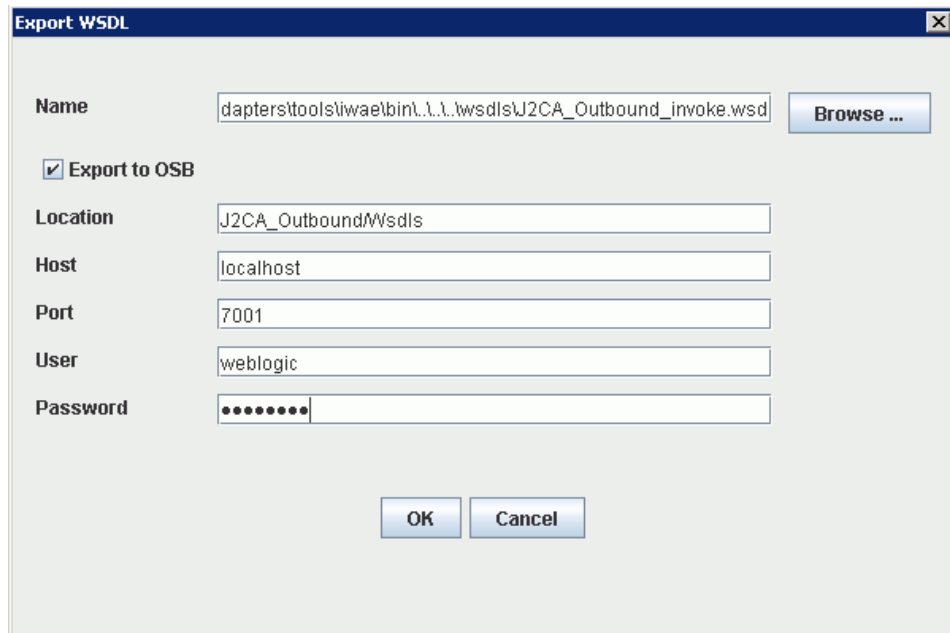


図 9-10 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

図 9-10 「WSDL のエクスポート」ダイアログ



4. 「名前」フィールドには、WSDL ファイルのデフォルトのファイル名が指定されています。デフォルトを受け入れることも独自に指定することもできます。
5. OSB にエクスポート・オプションを選択します。

6. 「ロケーション」フィールドで、WSDL ドキュメントのパブリッシュ先となる Oracle Service Bus 内のフォルダ名を入力します。  
この場所は、Oracle Service Bus プロジェクト名と、オプションで 1 つ以上のフォルダ名で構成されます。プロジェクト名およびすべてのフォルダ名は、スラッシュ文字 / で区切る必要があります。
7. 「ホスト」フィールドで、Oracle Service Bus がインストールされているマシンの名前を入力します。
8. 「ポート」フィールドで、Oracle Service Bus で使用されているポートを入力します。
9. 「ユーザー」フィールドで、Oracle Service Bus にアクセスする際のユーザー名を入力します。
10. 「パスワード」フィールドで、Oracle Service Bus にアクセスする際のパスワードを入力します。
11. 「OK」をクリックします。  
WSDL が「WSDL のエクスポート」ダイアログで指定された場所にパブリッシュされ、Oracle Service Bus でビジネス・サービスまたはプロキシ・サービスに使用できるようになりました。

## 9.2.4 WSDL ベースのビジネス・サービスの構成

WSDL ベースのプロキシ・サービスを構成するには、次のようにします。

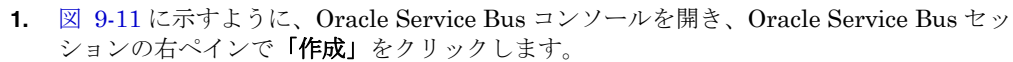
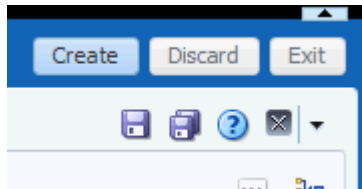
1.  9-11 に示すように、Oracle Service Bus コンソールを開き、Oracle Service Bus セッションの右ペインで「作成」をクリックします。

図 9-11 「作成」 ボタン



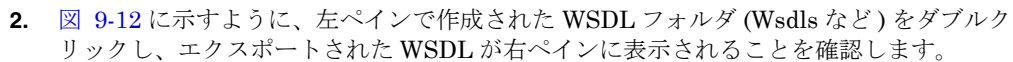
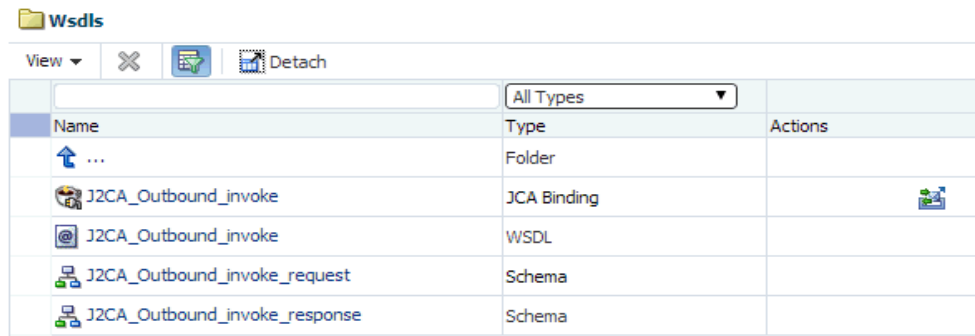
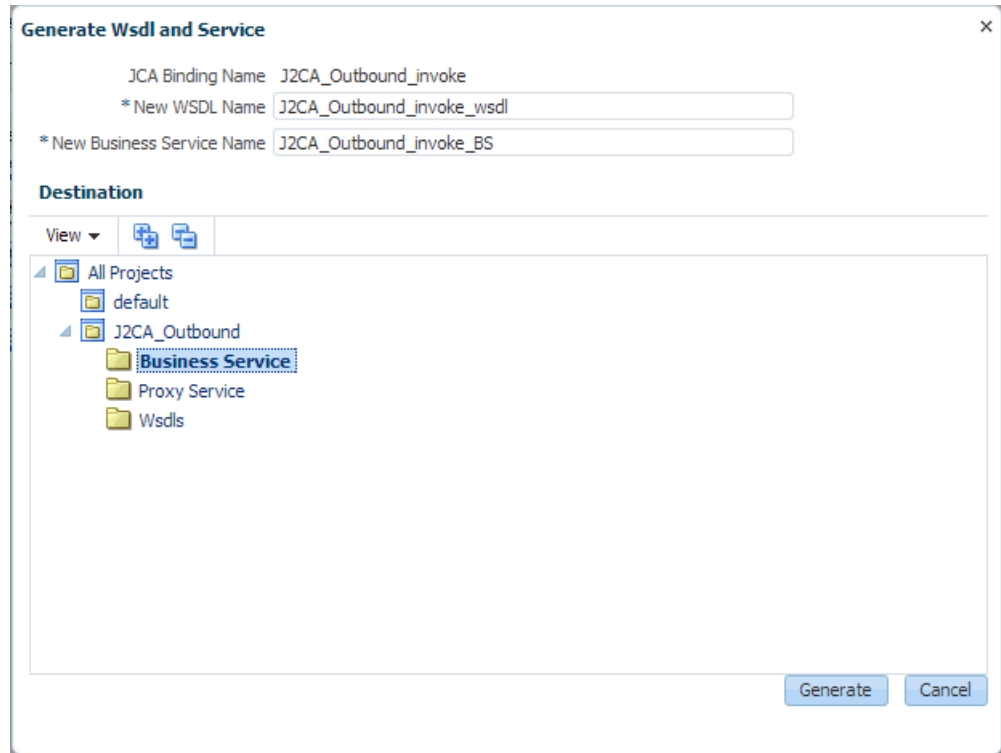
2.  9-12 に示すように、左ペインで作成された WSDL フォルダ (Wsdls など) をダブルクリックし、エクスポートされた WSDL が右ペインに表示されることを確認します。

図 9-12 「Wsdls」 フォルダ



3. 「JCA バインド」に対応する「アクション」列内のアイコンをクリックします。  
図 9-13 に示すように、「WSDL とサービスの生成」ウィンドウが表示されます。

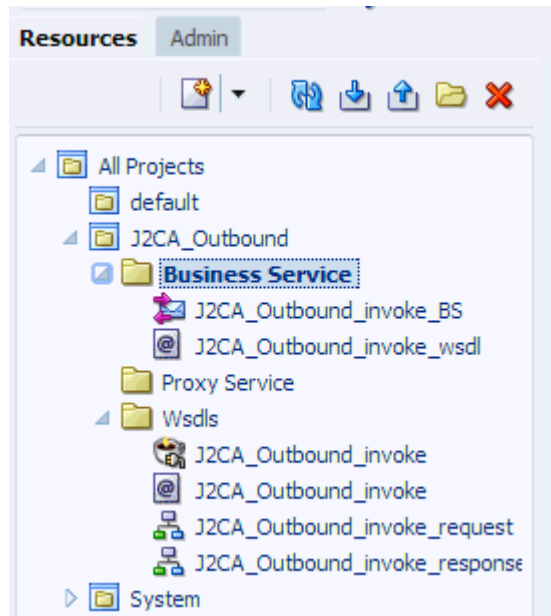
図 9-13 「WSDL とサービスの生成」ウィンドウ



4. 対応するフィールドに、新しい WSDL 名および新しいビジネス・サービス名を指定します。
5. 「宛先」領域で、使用可能なプロジェクトおよびビジネス・サービス用に指定されているサブフォルダを選択します。
6. 「生成」をクリックします。

7. 図 9-14 に示すように、プロジェクト・フォルダの下の「**Business Service**」を展開し、生成された WSDL およびビジネス・サービスが表示されているかどうかを確認します。

図 9-14 「Business Service」 フォルダ



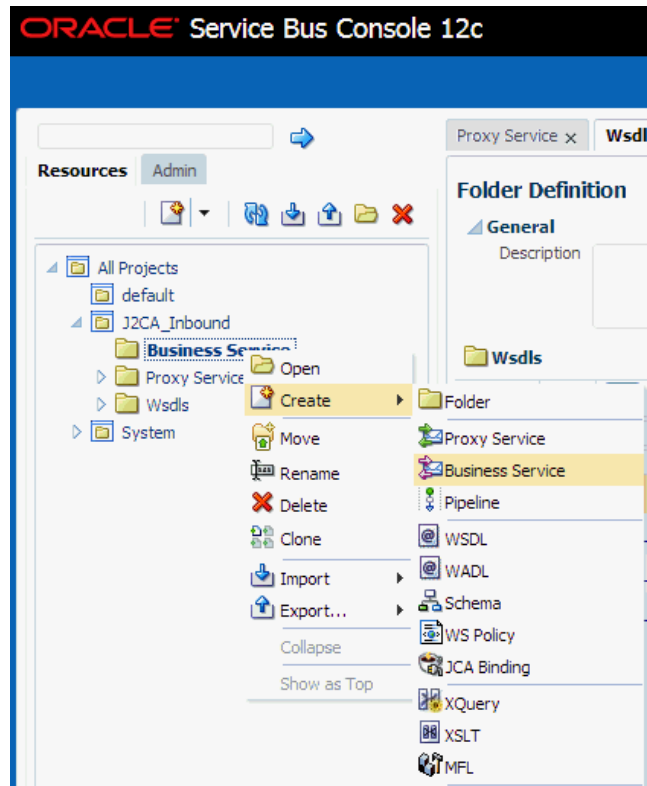


## 9.2.5 ファイル・タイプ・ビジネス・サービスの構成

ファイル・タイプ・ビジネス・サービスを構成するには、次のようにします。

1. 図 9-15 に示すように、作成した「**Business Service**」フォルダを左ペインで右クリックし、「作成」を選択して「ビジネス・サービス」をクリックします。

図 9-15 「Business Service」フォルダ



「ビジネス・サービスの作成」ウィンドウが表示されます。

2. 図 9-16 に示すように、「リソース名」フィールドでビジネス・サービスの名前を指定し、「サービス定義」の下の「トランスポート」セクションで「ファイル」オプションを選択して、「次」をクリックします。

図 9-16 サービス定義

Create Business Service

Create Type Transport

Create Service

\* Resource Name File\_Out

Description

Service Definition

WSDL Based Service

Name

Path

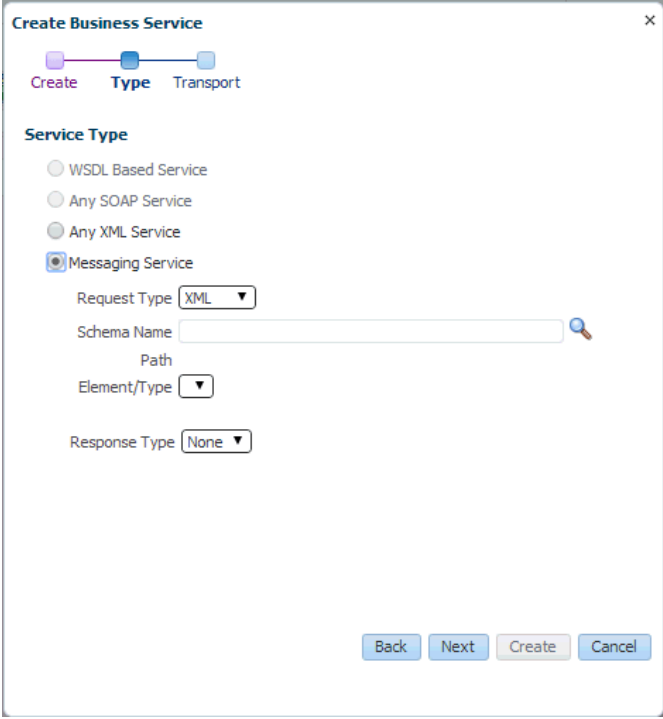
Port/Binding

Transport file

Back Next Create Cancel

- 「サービス・タイプ」セクションで、「メッセージ・サービス」を選択します。デフォルトでは、「リクエスト・タイプ」は「XML」に、「レスポンス・タイプ」は「なし」に設定されています。その後、[図 9-17](#) に示すように、「次」をクリックします。

図 9-17 サービス・タイプの構成ページ



The screenshot shows a dialog box titled "Create Business Service" with a close button (X) in the top right corner. At the top, there is a progress indicator with three steps: "Create", "Type", and "Transport". The "Type" step is currently active and highlighted in blue. Below this, the "Service Type" section contains four radio button options: "WSDL Based Service", "Any SOAP Service", "Any XML Service", and "Messaging Service". The "Messaging Service" option is selected. Underneath, there are several input fields: "Request Type" is a dropdown menu set to "XML"; "Schema Name" is a text input field with a search icon; "Path" is a text input field; "Element/Type" is a dropdown menu; and "Response Type" is a dropdown menu set to "None". At the bottom of the dialog, there are four buttons: "Back", "Next", "Create", and "Cancel".

- 「エンドポイント URI」フィールドで、ファイル・システム上の宛先フォルダへのパスを入力します。

5. 図 9-18 に示すように、「作成」をクリックします。

図 9-18 「トランスポート」 ページ

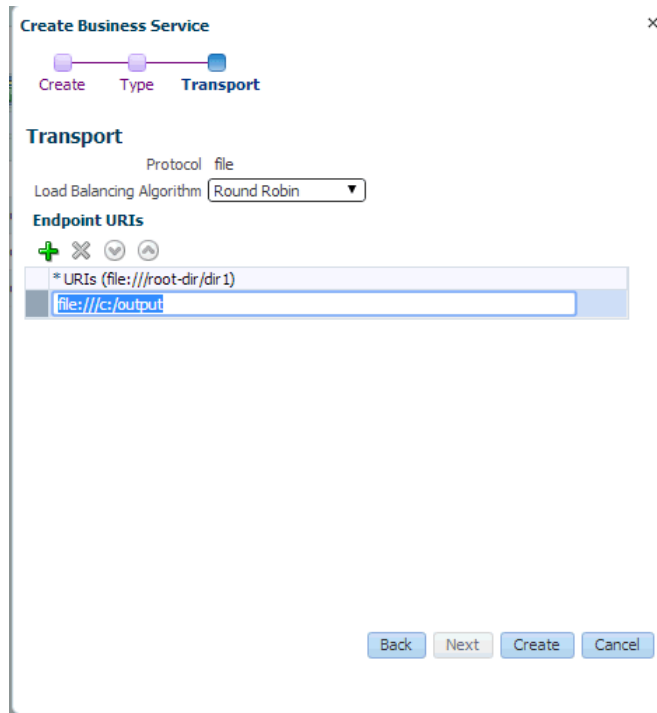
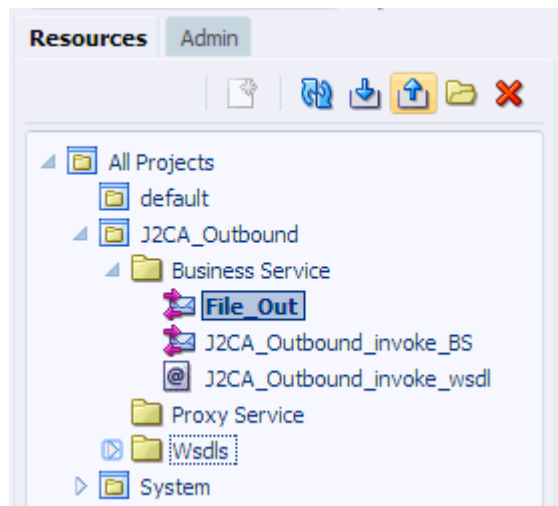


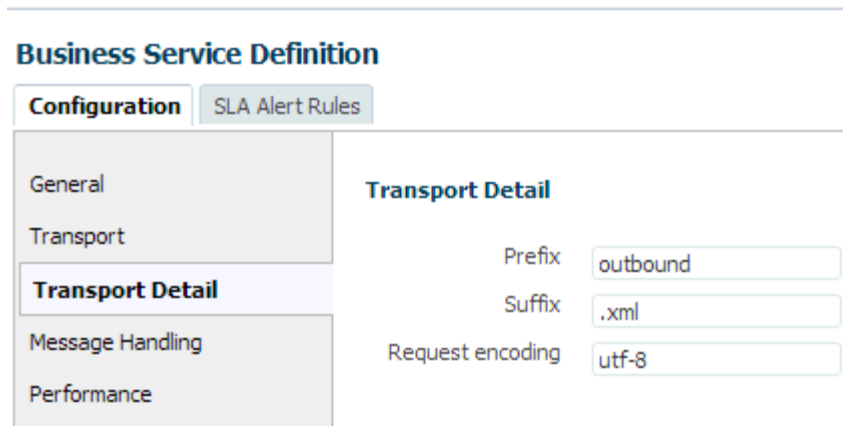
図 9-19 に示すように、「Business Service」の下に、ビジネス・サービス **File\_Out** が作成されて表示されます。

図 9-19 File\_Out ビジネス・サービス



6. 「File\_Out」をダブルクリックして、図 9-20 に示すように、左ペインの「トランスポートの詳細」をクリックし、受信する出力ファイルの接頭辞と接尾辞を入力します。

図 9-20 トランスポートの詳細



7. 図 9-21 に示すように、右隅の「保存」または「すべて保存」アイコンをクリックします。

図 9-21 「保存」 / 「すべて保存」アイコン

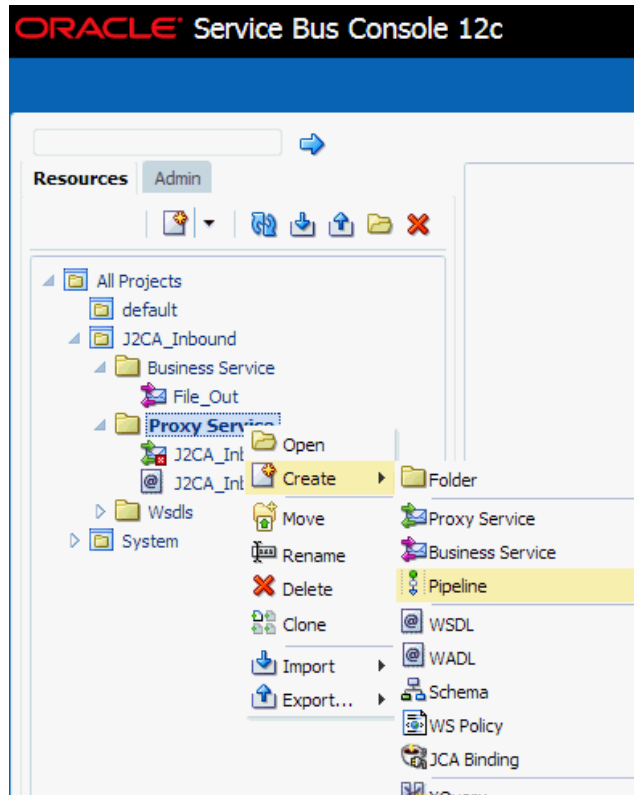


## 9.2.6 プロキシ・サービスでのパイプラインの構成

パイプラインを構成するには、次のようにします。

1. 図 9-22 に示すように、「Proxy Service」フォルダを右クリックし、「作成」を選択して、「パイプライン」をクリックします。

図 9-22 「パイプライン」オプション



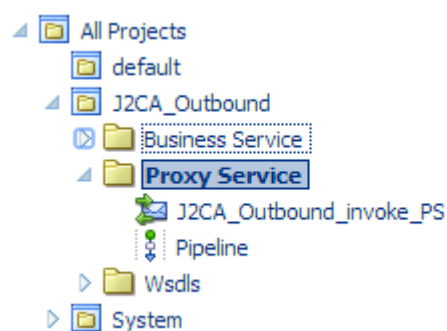
「パイプラインの作成」ウィンドウが表示されます。

- 「パイプライン名」フィールドに名前を入力します。デフォルトでは、「プロキシ・サービスとして公開」が選択されています。プロキシ・サービス名を変更する必要がある場合は、[図 9-23](#) に示すように、名前を変更して「トランスポート」を「file」に設定し、「作成」をクリックします。

図 9-23 「パイプラインの作成」 ウィンドウ

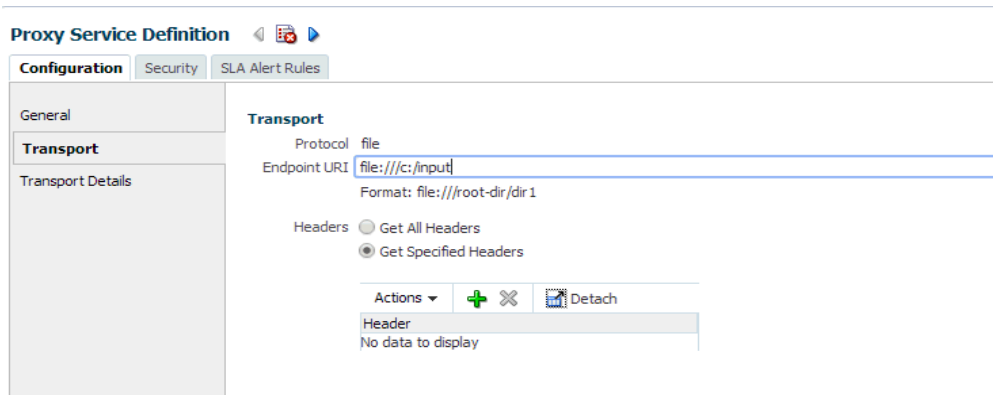
[図 9-24](#) に示すように、「Proxy Service」の下に、作成されたパイプラインおよびプロキシ・サービスが表示されます。

図 9-24 「パイプライン」 ノード



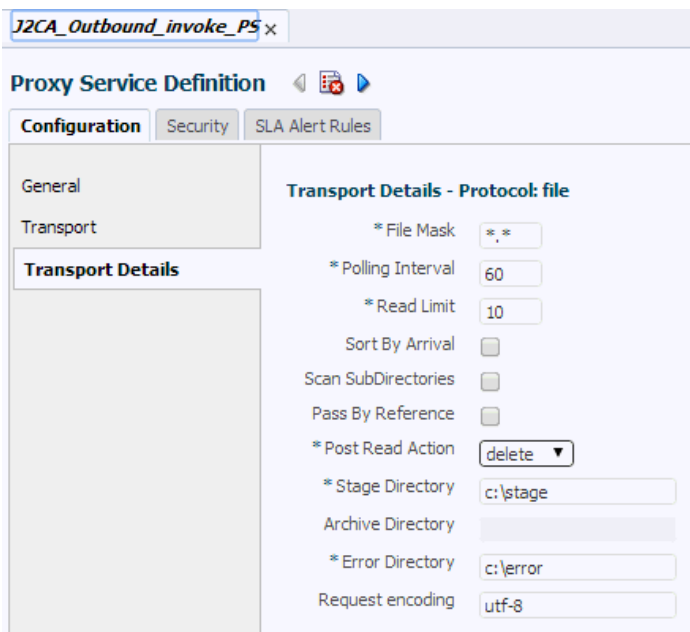
- 作成されたプロキシ・サービスをダブルクリックし、左ペインの「トランスポート」をクリックします。図 9-25 に示すように、「エンドポイント URI」フィールドで入力場所を指定します。

図 9-25 トランスポート



- 図 9-26 に示すように、左ペインの「トランスポートの詳細」をクリックして、「ステージ・ディレクトリ」および「エラー・ディレクトリ」の各フィールドに場所を指定します。

図 9-26 トランスポートの詳細



- 図 9-27 に示すように、右隅の「すべて保存」アイコンをクリックします。

図 9-27 「すべて保存」アイコン





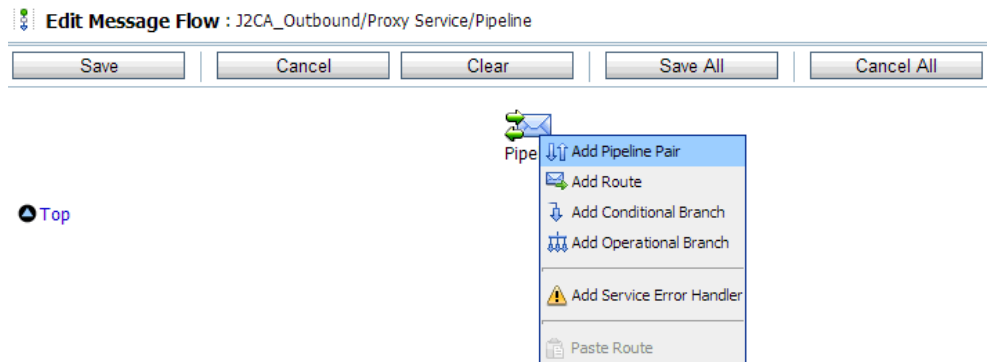
- 「Pipeline」ノードをダブルクリックして、図 9-28 に示すように、右ペインの「メッセージ・フローを開く」アイコンをクリックしてメッセージ・フローを開きます。

図 9-28 「メッセージ・フローを開く」アイコン



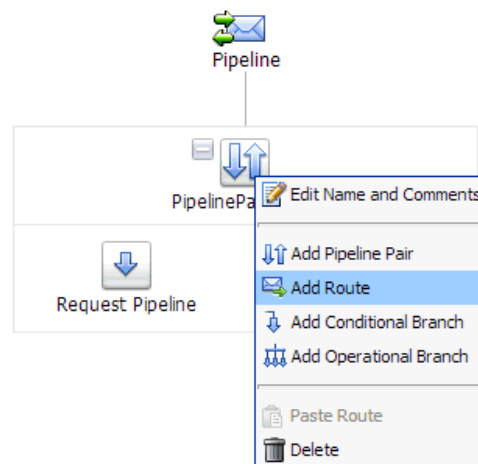
- 図 9-29 に示すように、「プロキシ・サービス」アイコンをクリックして、メニューの「パイプライン・ペアの追加」を選択します。

図 9-29 「パイプライン・ペアの追加」オプション



- 図 9-30 に示すように、「PipelinePairNode1」アイコンをクリックして、メニューの「ルートの追加」を選択します。

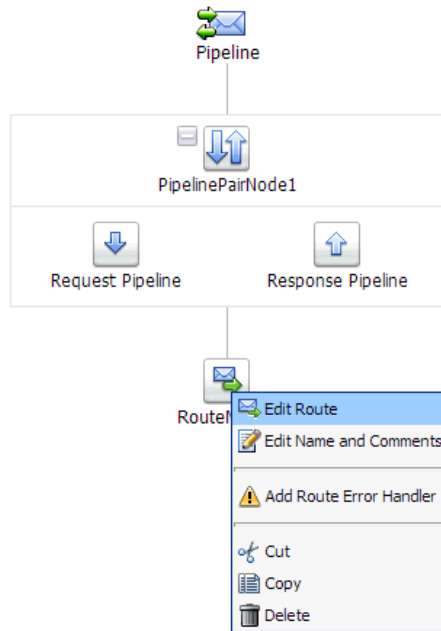
図 9-30 「ルートの追加」オプション



「RouteNode1」アイコンが「PipelinePairNode1」アイコンの下に追加されます。

9. 図 9-31 に示すように、「RouteNode1」アイコンをクリックして、メニューの「ルート  
の編集」を選択します。

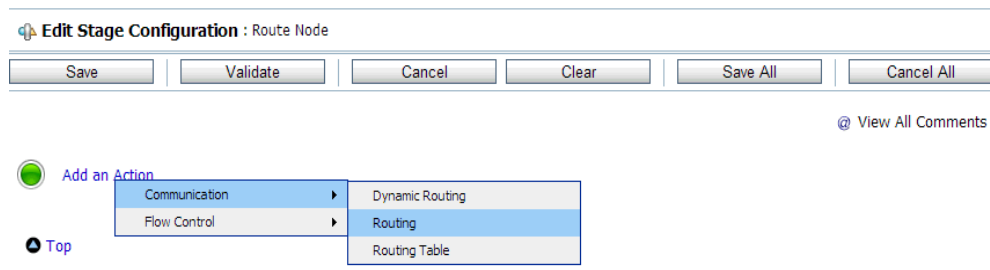
図 9-31 「ルート of の編集」オプション



「ステージ構成の編集」ワークスペース領域が表示されます。

10. 図 9-32 に示すように、「アクションの追加」をクリックして「通信」を選択し、「ルー  
ティング」をクリックします。

図 9-32 「ステージ構成の編集」ワークスペース領域



11. 図 9-33 に示すように、<Service> をクリックします。

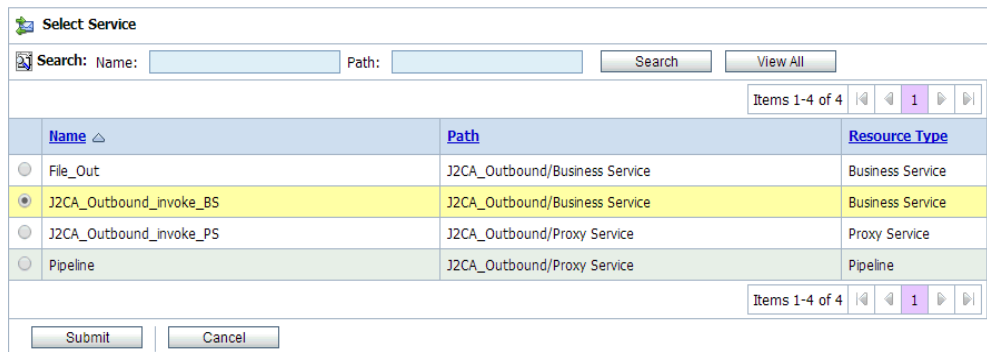
図 9-33 アクション



「サービスの選択」ダイアログが表示されます。

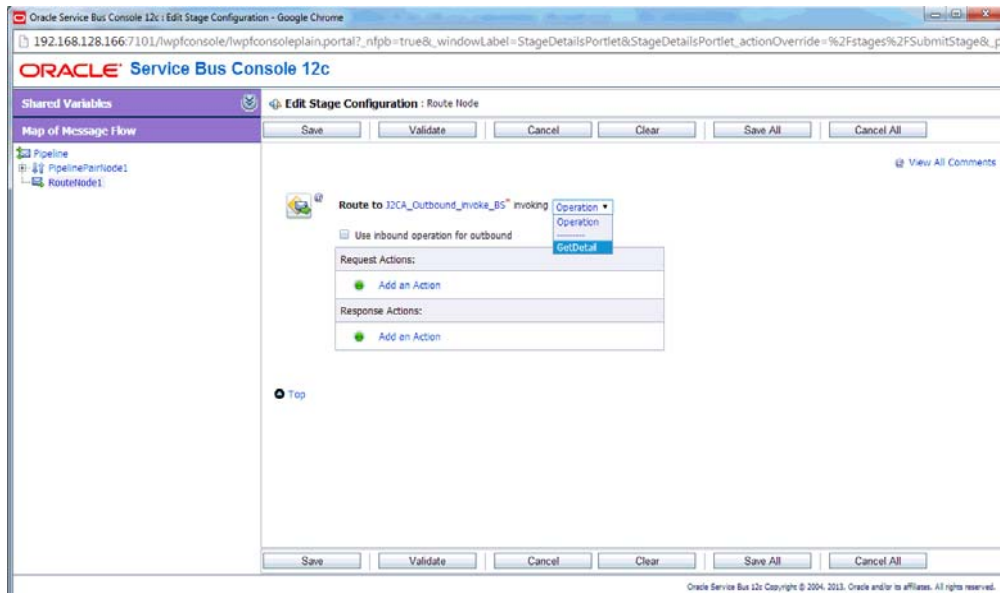
12. 図 9-34 に示すように、SAP R/3 用に構成された WSDL タイプのビジネス・サービスを選択して、「発行」をクリックします。

図 9-34 「サービスの選択」ダイアログ



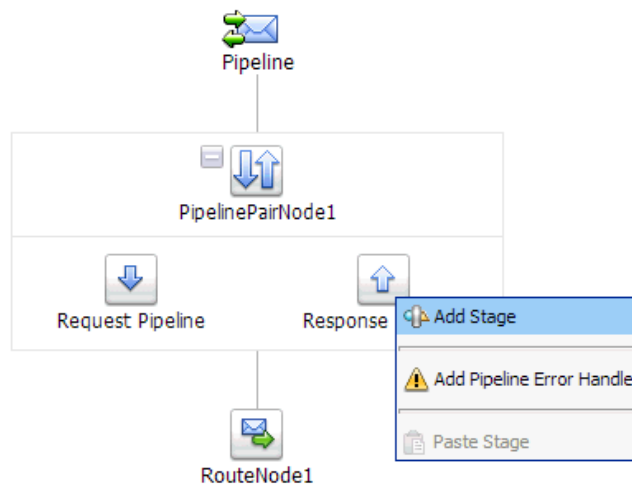
13. 図 9-35 に示すように、リストから、操作属性として SAP R/3 ビジネス・オブジェクトの名前 (GetDetail など) を選択し、「保存」をクリックします。

図 9-35 ステージ構成の編集



14. 図 9-36 に示すように、「レスポンス・パイプライン」アイコンをクリックして、メニューの「ステージの追加」を選択します。

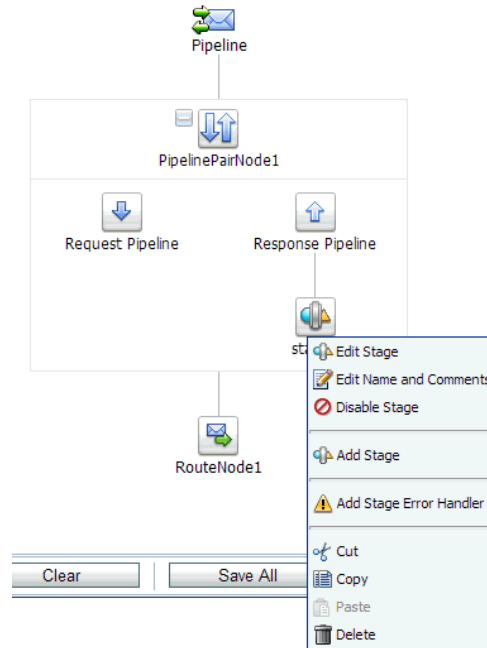
図 9-36 「レスポンス・パイプライン」アイコン



「ステージ 1」アイコンが「レスポンス・パイプライン」アイコンの下に追加されます。

15. 図 9-37 に示すように、「ステージ 1」アイコンをクリックして、メニューの「ステージの編集」を選択します。

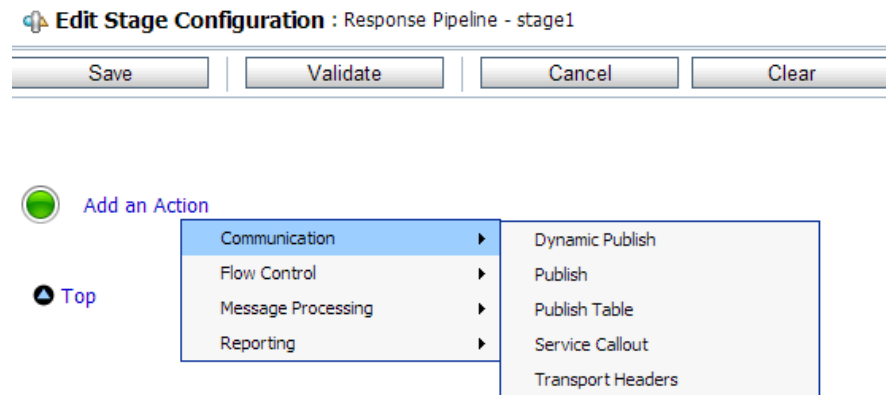
図 9-37 「ステージの編集」オプション



「ステージ構成の編集」ワークスペース領域が表示されます。

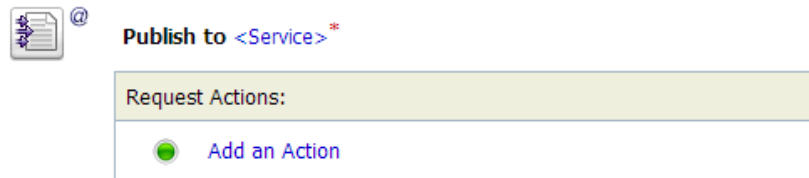
16. 図 9-38 に示すように、「アクションの追加」をクリックして「通信」を選択し、「パブリッシュ」をクリックします。

図 9-38 「ステージ構成の編集」ワークスペース領域



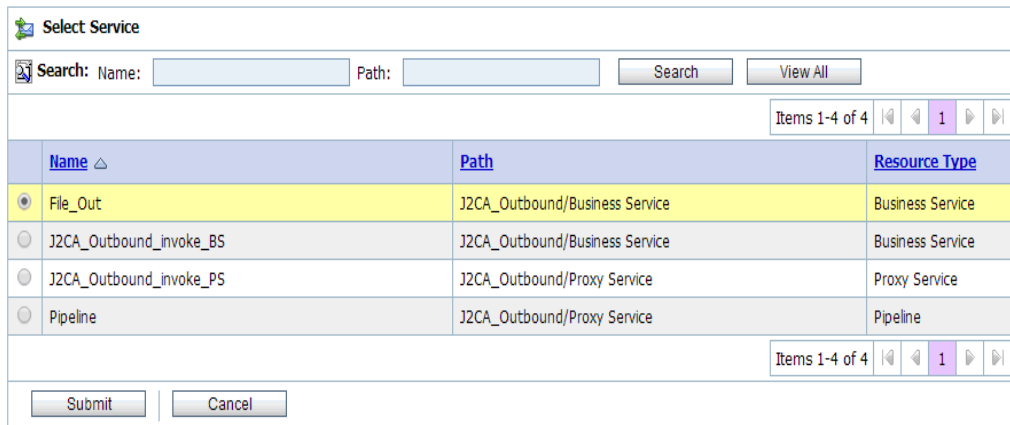
17. 図 9-39 に示すように、<Service> をクリックします。

図 9-39 <Service> アクション



18. 図 9-40 に示すように、「サービスの選択」ダイアログでファイル・タイプ・ビジネス・サービスを選択し、「発行」をクリックします。

図 9-40 「サービスの選択」ダイアログ

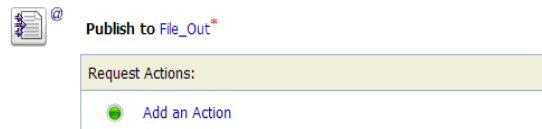


19. 図 9-41 に示すように、「すべて保存」をクリックします。

図 9-41 「すべて保存」ボタン

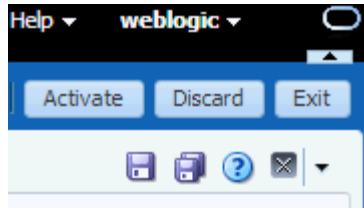


@ View



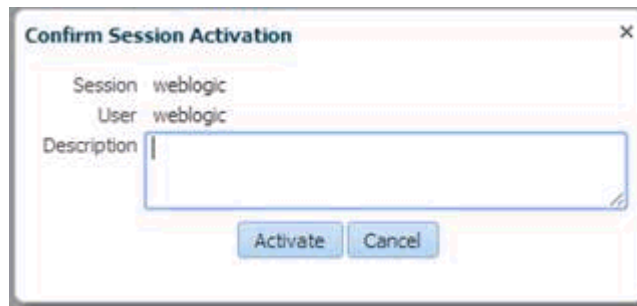
20. 図 9-42 に示すように、Oracle Service Bus セッションの右ペインの「アクティブ化」をクリックします。

図 9-42 「アクティブ化」ボタン



21. 図 9-43 に示すように、「アクティブ化」をクリックして変更を保存します。

図 9-43 セッションのアクティブ化の確認



22. 構成済の入力フォルダ (C:\input など) に、入力 XML ファイルをコピーして貼り付けます。出力は、構成済の出力場所 (C:\output など) で受信されます。

## 9.3 sbconsole を使用したインバウンド・プロセスの構成 (J2CA 構成)

この項では、sbconsole を使用して、J2CA 構成用にインバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\J2CA\SAP_Sample_J2CA_OSB_Inbound_Project
```

この項では、次の項目について説明します。

- 9.3.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」
- 9.3.2 項「アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定」
- 9.3.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ」
- 9.3.4 項「WSDL ベースのプロキシ・サービスの構成」
- 9.3.5 項「ファイル・タイプ・ビジネス・サービスの構成」
- 9.3.6 項「パイプラインの構成」

### 9.3.1 Oracle Service Bus の起動およびプロジェクト・フォルダの作成

Oracle Service Bus の起動およびプロジェクト・フォルダの作成の詳細は、9-2 ページの 9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」を参照してください。

### 9.3.2 アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定

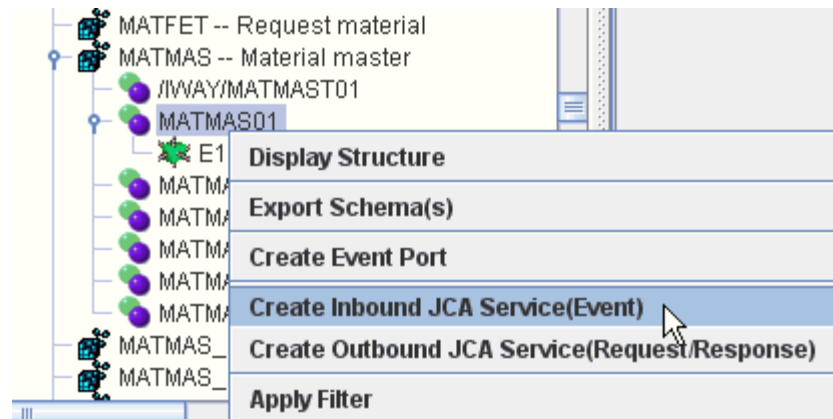
アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定の詳細は、9-6 ページの 9.2.2 項「アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定」を参照してください。

### 9.3.3 アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ

アプリケーション・エクスプローラから Oracle Service Bus に WSDL をパブリッシュするには、次のようにします。

1. アプリケーション・エクスプローラを起動し、J2CA 構成に接続して、SAP R/3 ターゲットに接続します。  
詳細は、第 4 章「構成 Oracle Application Adapter for SAP R/3」を参照してください。
2. SAP R/3 チャンネルを作成します。  
詳細は、4-19 ページの 4.9 項「イベント・アダプタの構成」を参照してください。
3. 「ALE (IDOCs)」ノードを展開して、ターゲットの中から「MATMAS – Material master」ノードを展開します。

図 9-44 インバウンド JCA サービス (イベント) の作成オプション



4. 図 9-44 に示すように、「MATMAS01」を右クリックし、メニューのインバウンド JCA サービス (イベント) の作成を選択します。



図 9-45 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

図 9-45 「WSDL のエクスポート」ダイアログ

5. 「名前」フィールドには、WSDL ファイルのデフォルトのファイル名が指定されています。デフォルトを受け入れることも独自に指定することもできます。
6. 「チャンネル」リストで、このインバウンド・サービスのために作成したチャンネルを選択します。
7. 「ルート」、「ネームスペース」および「スキーマ」の各検証の 3 つのチェック・ボックスも使用可能です。複数の検証オプションを選択できます。
8. **OSB にエクスポート・オプション**を選択します。
9. 「ロケーション」フィールドで、WSDL ドキュメントのパブリッシュ先となる **Oracle Service Bus** 内のフォルダ名を入力します。  
この場所は、**Oracle Service Bus** プロジェクト名と、オプションで 1 つ以上のフォルダ名で構成されます。プロジェクト名およびすべてのフォルダ名は、スラッシュ文字 / で区切る必要があります。
10. 「ホスト」フィールドで、**Oracle Service Bus** がインストールされているマシンの名前を入力します。
11. 「ポート」フィールドで、**Oracle Service Bus** で使用されているポートを入力します。
12. 「ユーザー」フィールドで、**Oracle Service Bus** にアクセスする際のユーザー名を入力します。
13. 「パスワード」フィールドで、**Oracle Service Bus** にアクセスする際のパスワードを入力します。
14. 「OK」をクリックします。

インバウンド WSDL が「WSDL のエクスポート」ダイアログで指定された場所にパブリッシュされ、Oracle Service Bus でプロキシ・サービスに使用できるようになりました。

### 9.3.4 WSDL ベースのプロキシ・サービスの構成

WSDL ベースのプロキシ・サービスを構成するには、次のようにします。

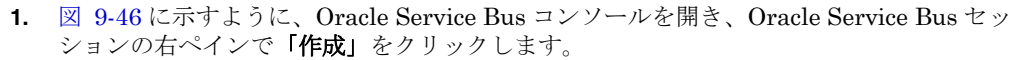
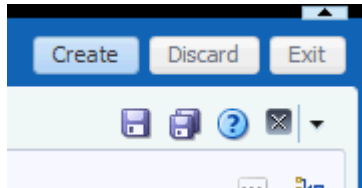
1.  図 9-46 に示すように、Oracle Service Bus コンソールを開き、Oracle Service Bus セッションの右ペインで「作成」をクリックします。

図 9-46 「作成」 ボタン



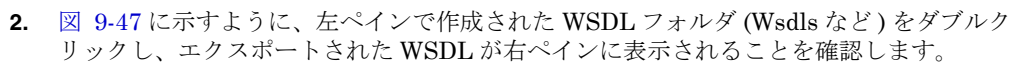
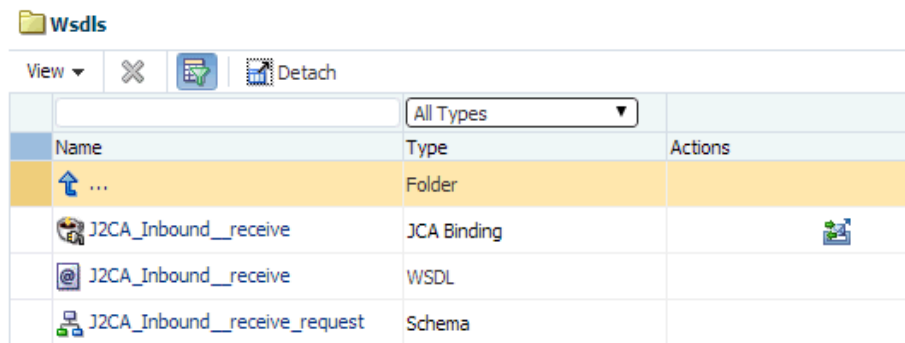
2.  図 9-47 に示すように、左ペインで作成された WSDL フォルダ (Wsdls など) をダブルクリックし、エクスポートされた WSDL が右ペインに表示されることを確認します。

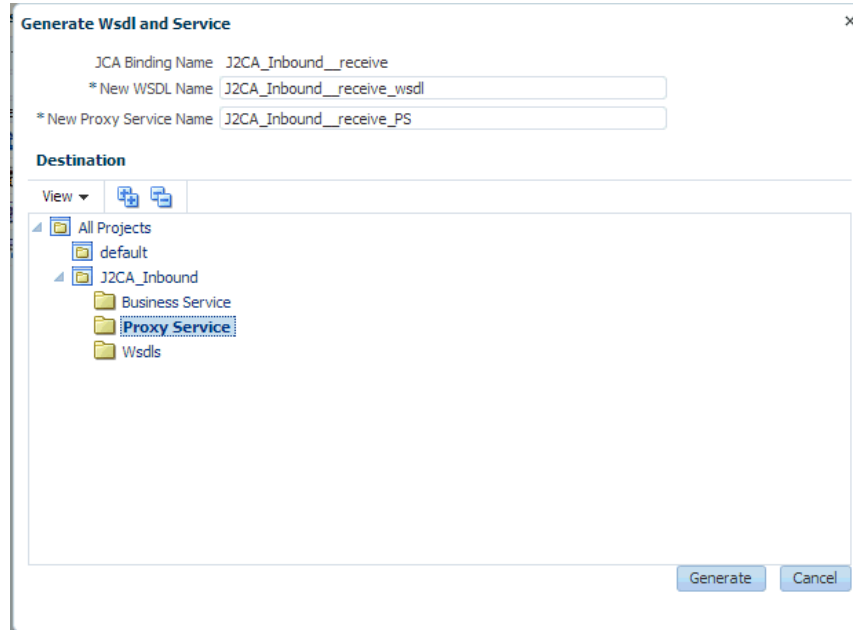
図 9-47 エクスポートされた WSDL



- 「JCA バインド」に対応する「アクション」列内のアイコンをクリックします。

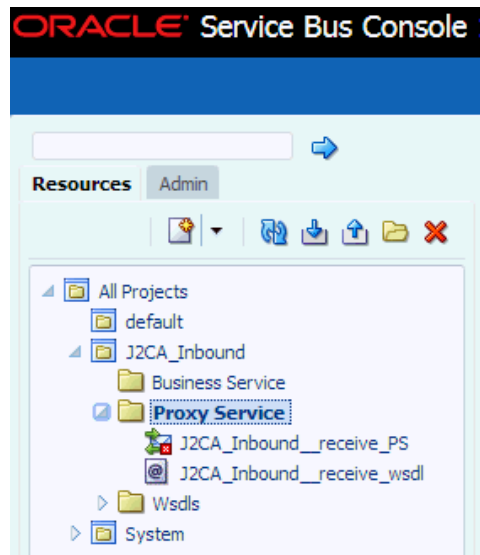
図 9-48 に示すように、「WSDL とサービスの生成」ページが表示されます。

図 9-48 「WSDL とサービスの生成」ページ



- 対応するフィールドに、新しい WSDL 名および新しいプロキシ・サービス名を指定します。
- 「宛先」領域で、使用可能なプロジェクトおよびプロキシ・サービス用に指定されているサブフォルダを選択します。
- 「生成」をクリックします。
- 図 9-49 に示すように、プロジェクト・エクスプローラで「Proxy Service」を展開し、生成された WSDL およびプロキシ・サービスが表示されているかどうかを確認します。

図 9-49 生成された WSDL

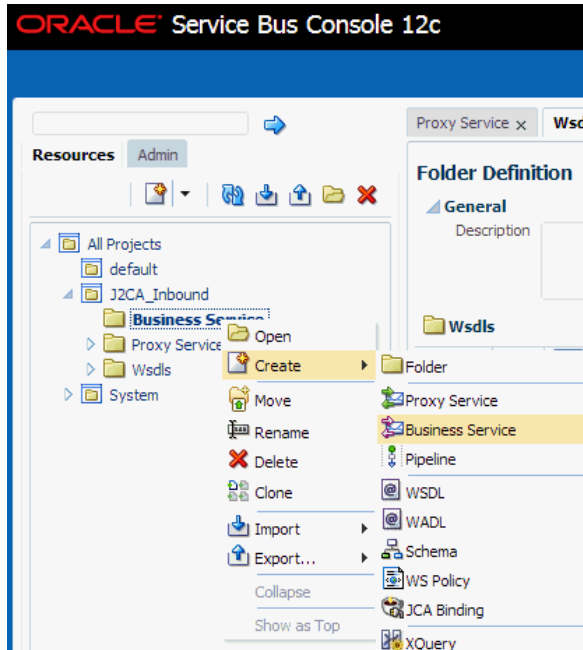


### 9.3.5 ファイル・タイプ・ビジネス・サービスの構成

ファイル・タイプ・ビジネス・サービスを構成するには、次のようにします。

1. 図 9-50 に示すように、作成した「Business Service」フォルダを左ペインで右クリックし、「作成」を選択して「ビジネス・サービス」をクリックします。

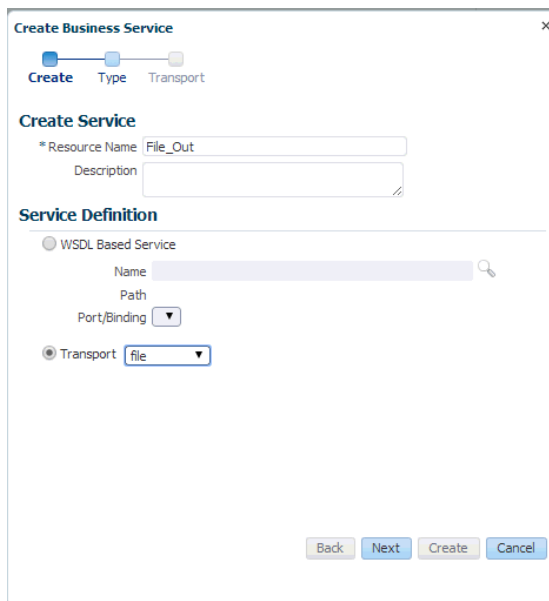
図 9-50 「Business Service」フォルダ



「ビジネス・サービスの作成」ウィンドウが表示されます。

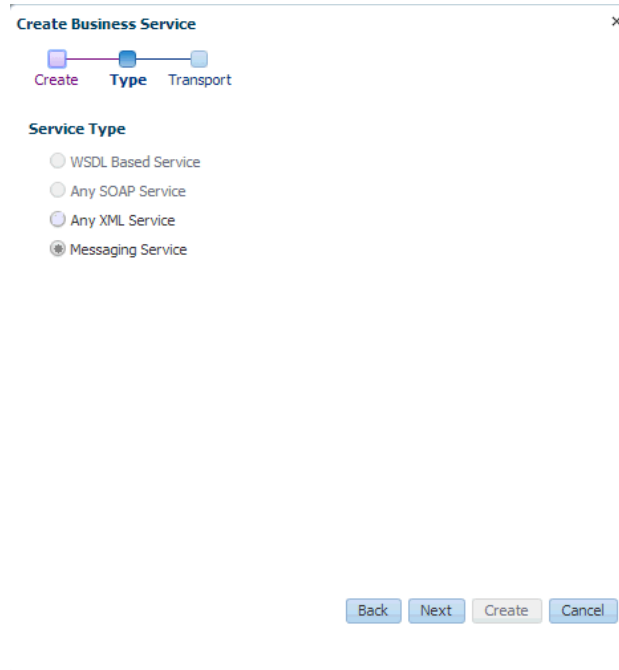
2. 図 9-51 に示すように、「リソース名」フィールドでビジネス・サービスの名前を指定し、「サービス定義」領域の「トランスポート」ドロップダウン・リストで「ファイル」オプションを選択します。

図 9-51 「ビジネス・サービスの作成」ウィンドウ



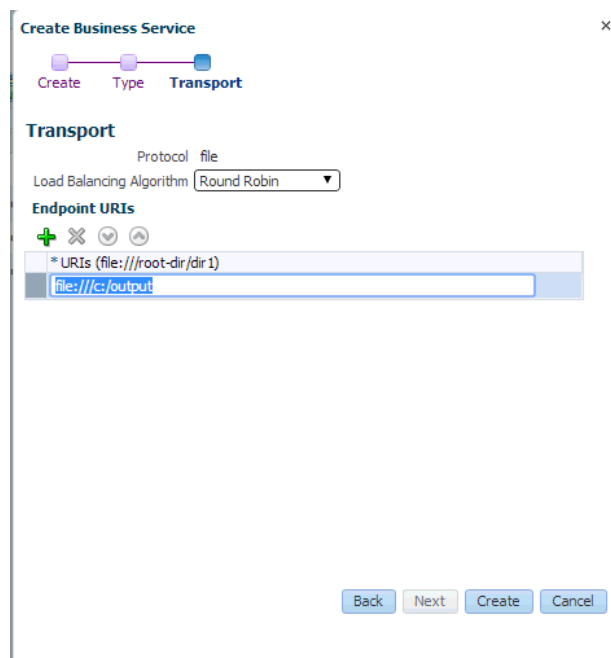
3. 「次」をクリックします。
4. 図 9-52 に示すように、「サービス・タイプ」領域で、サービス・タイプとして「メッセージ・サービス」を選択します。

図 9-52 「サービス・タイプ」領域



5. 「次」をクリックします。  
図 9-53 に示すように、「トランスポート」ページが表示されます。

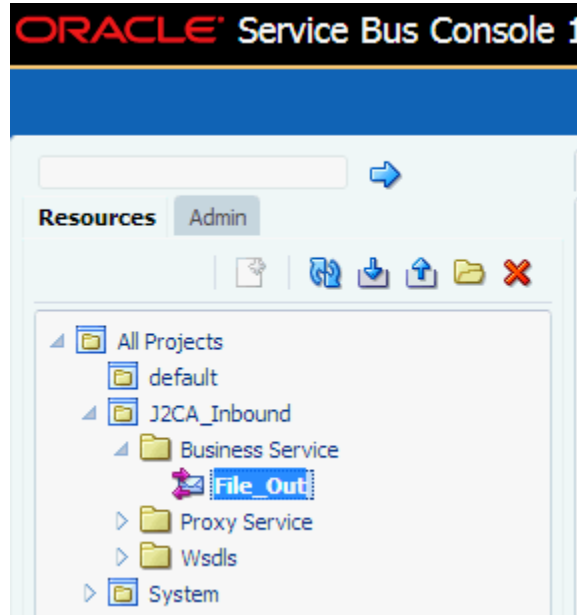
図 9-53 「トランスポート」ページ



- 「エンドポイント URI」フィールドで、ファイル・システム上の宛先フォルダへのパスを入力し、「作成」をクリックします。

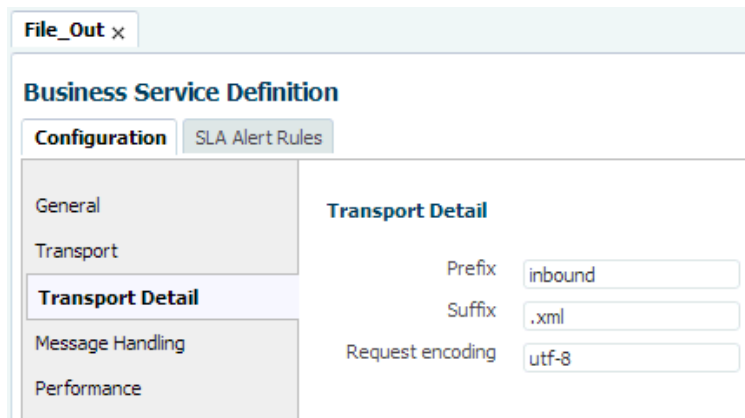
図 9-54 に示すように、「Business Service」の下に、ビジネス・サービス File\_Out が作成されて表示されます。

図 9-54 File\_Out ビジネス・サービス



- 「File\_Out」をダブルクリックして、図 9-55 に示すように、左ペインの「トランスポートの詳細」をクリックし、受信する出力ファイルの接頭辞と接尾辞を入力します。

図 9-55 「トランスポートの詳細」ページ



8. 図 9-56 に示すように、右隅の「保存」または「すべて保存」アイコンをクリックします。

図 9-56 保存の各アイコン

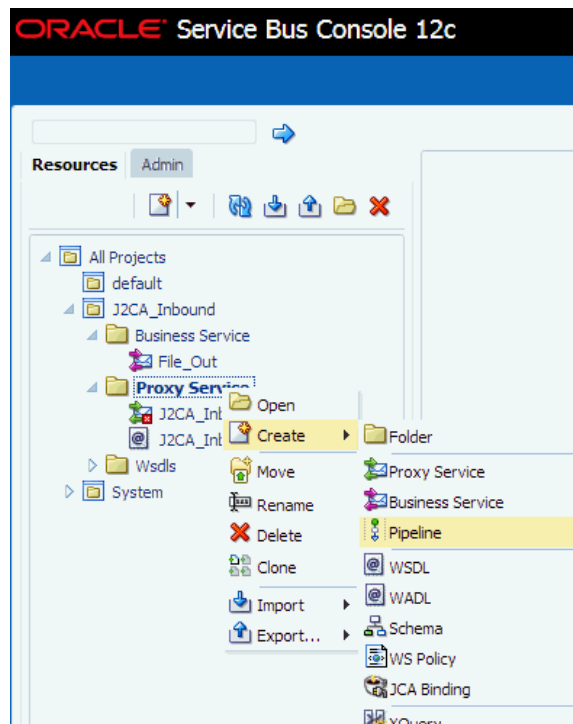


### 9.3.6 パイプラインの構成

パイプラインを構成するには、次のようにします。

1. 図 9-57 に示すように、作成したプロキシ・サービスを右クリックし、「作成」を選択して、「パイプライン」をクリックします。

図 9-57 「パイプライン」オプション



「パイプラインの作成」ウィンドウが表示されます。

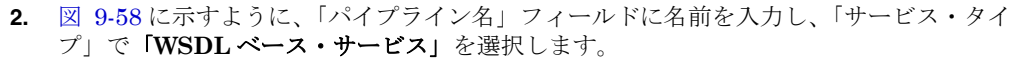
2.  9-58 に示すように、「パイプライン名」フィールドに名前を入力し、「サービス・タイプ」で「WSDL ベース・サービス」を選択します。

図 9-58 「パイプラインの作成」ウィンドウ

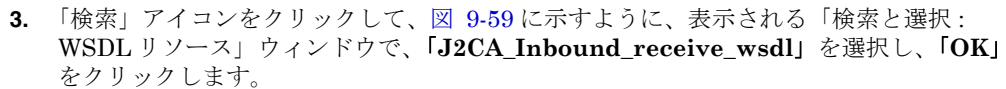
3. 「検索」アイコンをクリックして、 9-59 に示すように、表示される「検索と選択 : WSDL リソース」ウィンドウで、「J2CA\_Inbound\_receive\_wsdl」を選択し、「OK」をクリックします。

図 9-59 「検索と選択 : WSDL リソース」ウィンドウ

Name	Path	Namespace
J2CA_Inbound_receive_wsdl	J2CA_Inbound/Proxy Service	http://xmlns.oracle...
J2CA_Inbound_receive	J2CA_Inbound/Wsdl	http://xmlns.oracle...

「パイプラインの作成」ウィンドウが開きます。

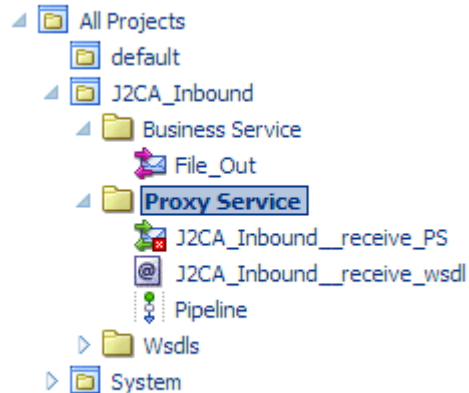


4. 図 9-60 に示すように、「プロキシ・サービスとして公開」チェック・ボックスの選択を解除して、「作成」をクリックします。

図 9-60 「パイプラインの作成」 ウィンドウ

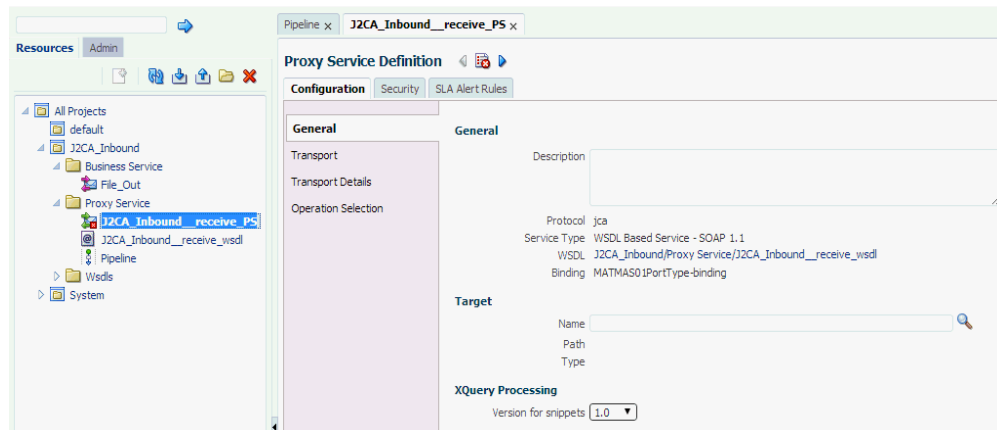
図 9-61 に示すように、「Proxy Service」の下にパイプラインが作成されて表示されます。

図 9-61 プロキシ・サービスのパイプライン



5. 図 9-62 に示すように、左ペインの「Proxy Service」の下で「J2CA\_Inbound\_receive\_PS」ノードをダブルクリックし、右ペインの「ターゲット」領域で「検索」アイコンをクリックします。

図 9-62 「プロキシ・サービス定義」ウィンドウ

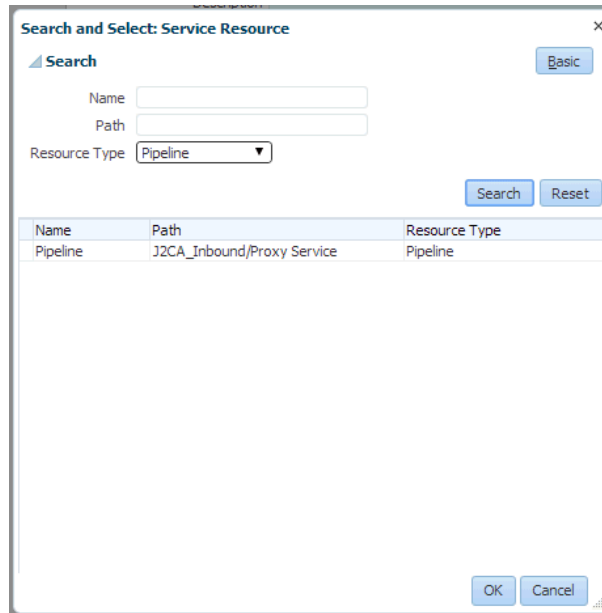


「検索と選択：サービス・リソース」ウィンドウが表示されます。

6. 「リソース・タイプ」ドロップダウン・リストで「パイプライン」を選択し、「検索」ボタンをクリックします。

図 9-63 に示すように、パイプラインが表示されます。

図 9-63 「検索と選択：サービス・リソース」ウィンドウ



7. 「パイプライン」を選択して、「OK」をクリックします。

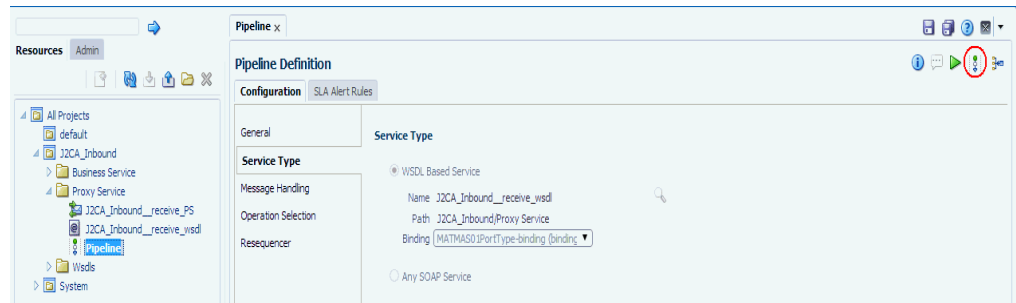
8. 図 9-64 に示すように、右隅の「保存」または「すべて保存」アイコンをクリックします。

図 9-64 「保存」および「すべて保存」アイコン



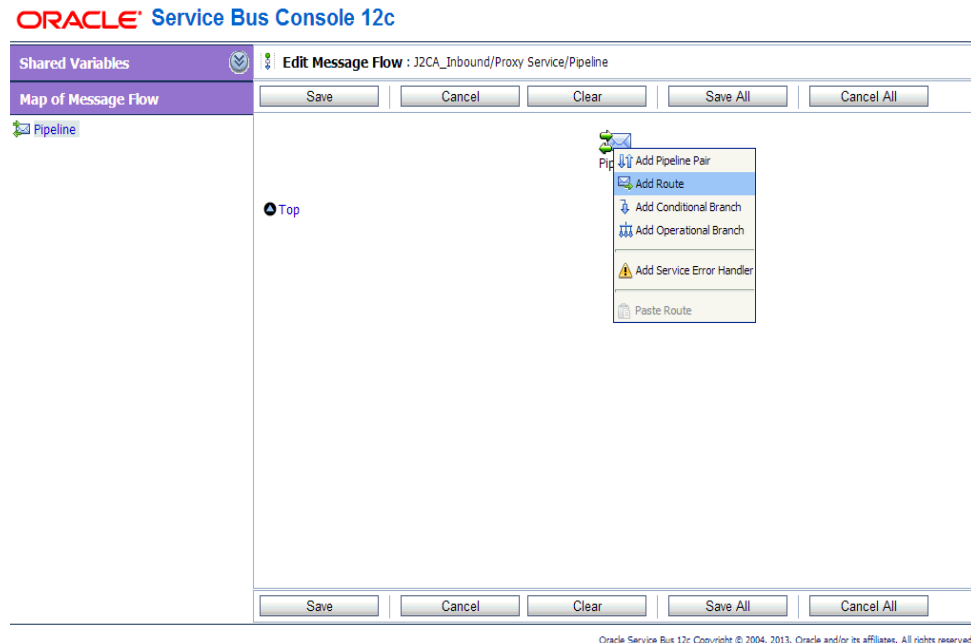
9. 図 9-65 に示すように、左ペインで、「Proxy Service」フォルダの下の「パイプライン」をダブルクリックし、右ペインの下向きアイコンをクリックしてメッセージ・フローを開きます。

図 9-65 メッセージ・フロー



10. 図 9-66 に示すように、表示される「プロキシ・サービス」アイコンをクリックし、メニューの「ルートの追加」を選択します。

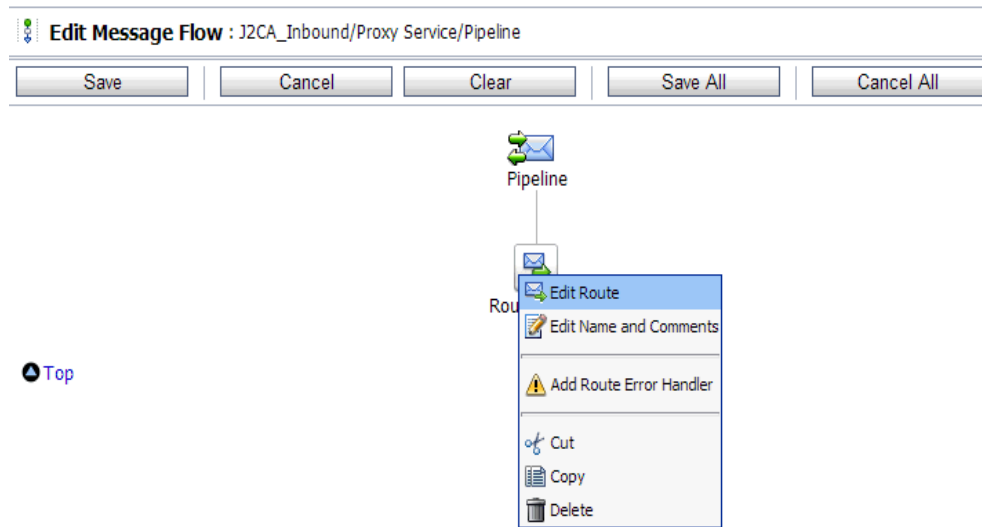
図 9-66 「ルートの追加」オプション



「RouteNode1」アイコンが追加されます。

11. 図 9-67 に示すように、「RouteNode1」アイコンをクリックして、メニューの「ルート  
の編集」を選択します。

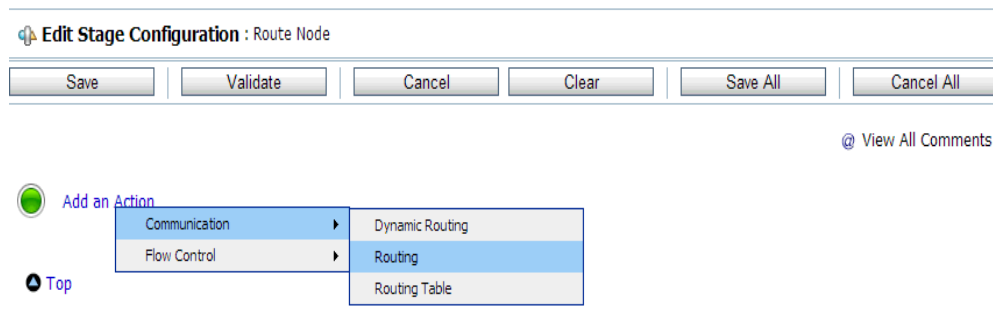
図 9-67 「ルート of の編集」オプション



「ステージ構成 of の編集」ワークスペース領域が表示されます。

12. 図 9-68 に示すように、「アクション of の追加」をクリックして、メニュー of の「通信」を選択し、「ルーティング」をクリックします。

図 9-68 「ステージ構成 of の編集」ワークスペース




13.  9-69 に示すように、<Service> をクリックします。

図 9-69 Service ルート・アクション



「サービスの選択」ダイアログが表示されます。


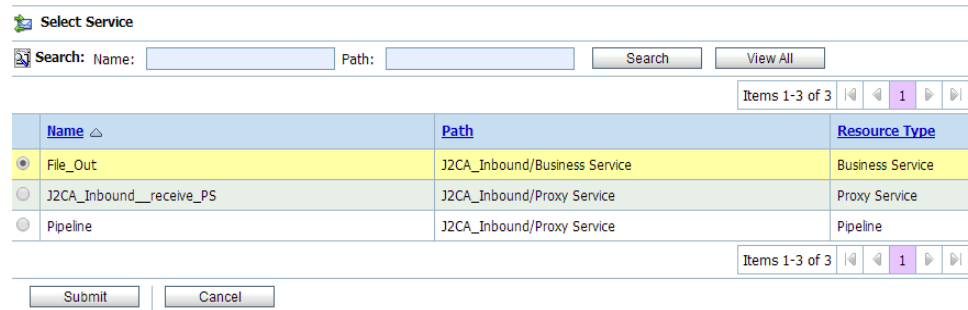
14.  9-70 に示すように、「File\_Out」ビジネス・サービスを選択して、「発行」をクリックします。

図 9-70 「サービスの選択」ダイアログ



「ステージ構成の編集」ワークスペース領域に戻ります。


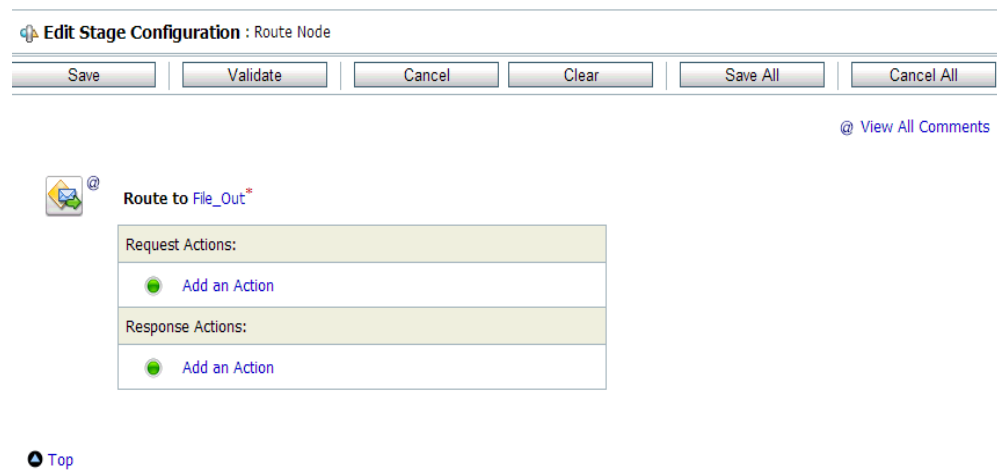
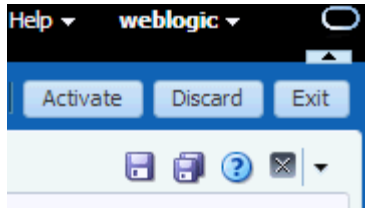
15.  9-71 に示すように、「すべて保存」をクリックします。

図 9-71 「ステージ構成の編集」ワークスペース領域



16. 図 9-72 に示すように、Oracle Service Bus セッションの右ペインの「アクティブ化」をクリックします。

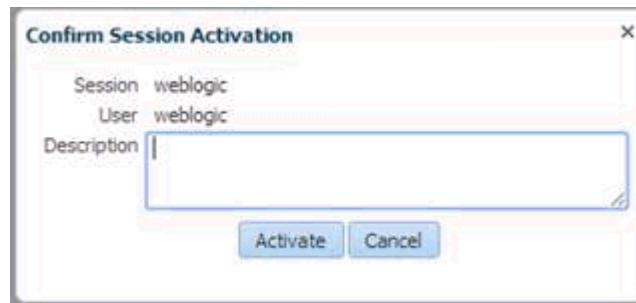
図 9-72 「アクティブ化」ボタン



「セッションのアクティブ化の確認」ウィンドウが表示されます。

17. 図 9-73 に示すように、「アクティブ化」をクリックして変更を保存します。

図 9-73 「セッションのアクティブ化の確認」ウィンドウ



18. SAP R/3 システムからイベントをトリガーし、構成済の出力場所では出力が受信されるかどうかを確認します。

## 9.4 sbconsole を使用したアウトバウンド・プロセスの構成 (BSE 構成)

この項では、sbconsole を使用して、BSE 構成用にアウトバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\BSE\SAP_Sample_BSE_OSB_Outbound_Project
```

この項では、次の項目について説明します。

- 9.4.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」
- 9.4.2 項「アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定」
- 9.4.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ」
- 9.4.4 項「ファイル・タイプ・ビジネス・サービスの構成」
- 9.4.5 項「WSDL ベースのビジネス・サービスの構成」
- 9.4.6 項「プロキシ・サービスでのパイプラインの構成」

## 9.4.1 Oracle Service Bus の起動およびプロジェクト・フォルダの作成

Oracle Service Bus の起動およびプロジェクト・フォルダの作成の詳細は、9-2 ページの 9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」を参照してください。

## 9.4.2 アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定

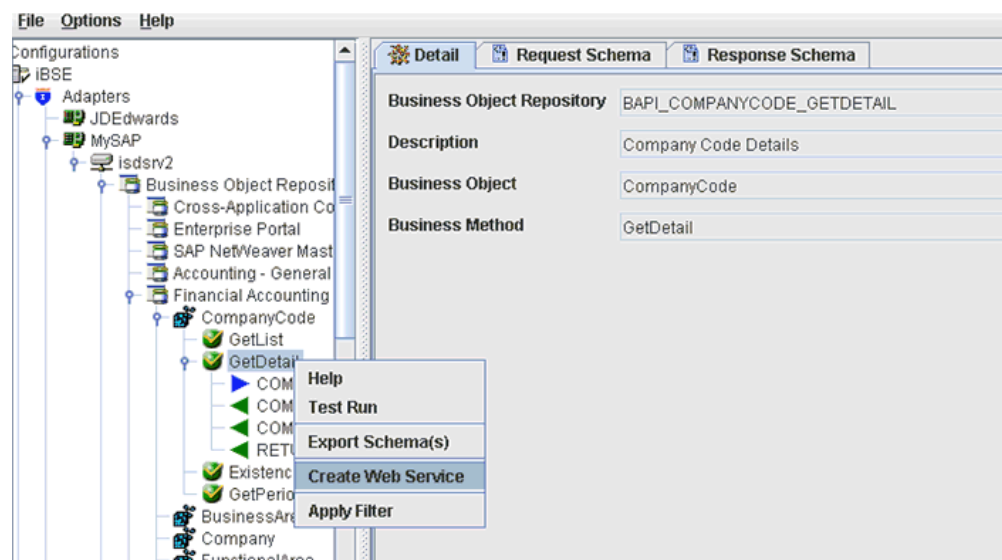
アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定の詳細は、9-6 ページの 9.2.2 項「アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定」を参照してください。

## 9.4.3 アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ

この項では、アプリケーション・エクスプローラ (BSE 構成) から Oracle Service Bus に WSDL をパブリッシュする方法について説明します。

1. アプリケーション・エクスプローラを起動し、BSE 構成に接続して、SAP R/3 ターゲットに接続します。

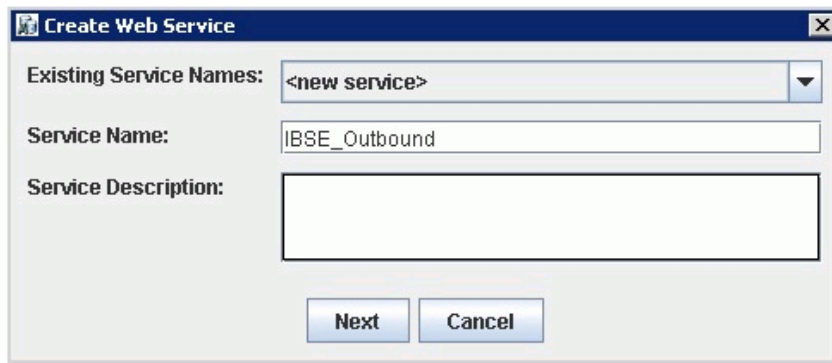
図 9-74 「Web サービスの作成」オプション



2. 「ビジネス・オブジェクト・リポジトリ」ノード→「Financial Accounting」ノード→「CompanyCode」ビジネス・オブジェクトを展開します。
3. 図 9-74 に示すように、GetDetail メソッドを右クリックし、メニューの「Web サービスの作成」を選択します。

図 9-75 に示すように、「Web サービスの作成」ダイアログが表示されます。

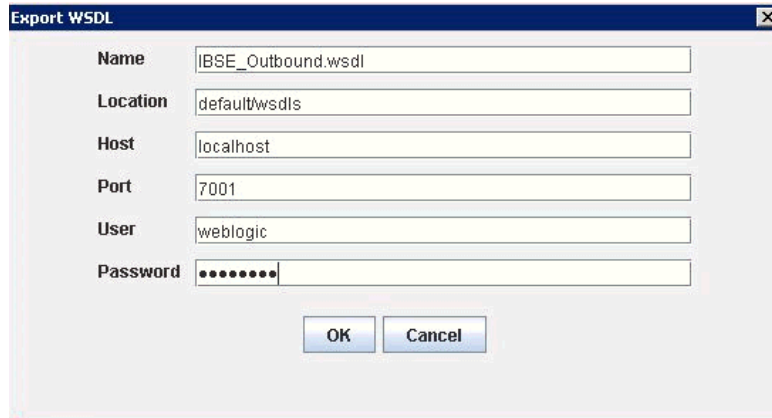
図 9-75 「Web サービスの作成」ダイアログ



4. サービス名を入力し、「次へ」をクリックします。
5. 表示される次のダイアログで、「OK」をクリックします。  
アプリケーション・エクスプローラの表示が「ビジネス・サービス」ノードに切り替わり、新規 Web サービスが左ペインに表示されます。
6. 新規 Web サービスを右クリックして、メニューの「WSDL のエクスポート」を選択します。

図 9-76 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

図 9-76 「WSDL のエクスポート」ダイアログ



7. 「名前」フィールドには、WSDL ファイルのデフォルトのファイル名が指定されています。デフォルトを受け入れることも独自に指定することもできます。
8. 「ロケーション」フィールドで、WSDL ドキュメントのパブリッシュ先となる場所を入力します。  
この場所は、Oracle Service Bus プロジェクト名と、オプションで 1 つ以上のフォルダ名で構成されます。プロジェクト名およびすべてのフォルダ名は、スラッシュ文字 / で区切る必要があります。
9. 「ホスト」フィールドで、Oracle WebLogic Server が実行されているマシンの名前を入力します。
10. 「ポート」フィールドで、使用しているドメインのポートを入力します。
11. 「ユーザー」フィールドで、Oracle Service Bus にアクセスする際のユーザー名を入力します。



12. 「パスワード」フィールドで、Oracle Service Bus にアクセスする際のパスワードを入力します。

13. 「OK」をクリックします。

WSDL が「WSDL のエクスポート」ダイアログで指定された場所にパブリッシュされ、Oracle Service Bus でビジネス・サービスまたはプロキシ・サービスに使用できるようになりました。

## 9.4.4 ファイル・タイプ・ビジネス・サービスの構成

ファイル・タイプ・ビジネス・サービスの構成の詳細は、9-11 ページの 9.2.5 項「ファイル・タイプ・ビジネス・サービスの構成」を参照してください。

## 9.4.5 WSDL ベースのビジネス・サービスの構成

この項では、Oracle Service Bus コンソールを使用して、WSDL タイプのビジネス・サービスを構成する方法について説明します。

WSDL ベースのプロキシ・サービスを構成するには、次のようにします。

1. 左ペインで「Business Service」フォルダを右クリックし、「ビジネス・サービス」を選択します。

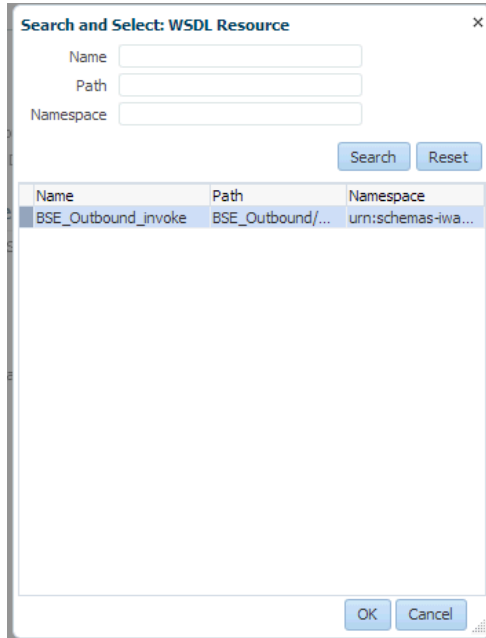
図 9-77 に示すように、「ビジネス・サービスの作成」ウィンドウが表示されます。

図 9-77 「ビジネス・サービスの作成」ウィンドウ

- ビジネス・サービスの名前を指定し、「サービス定義」領域で「WSDL ベース・サービス」オプションを選択して「検索」アイコンをクリックします。

図 9-78 に示すように、「検索と選択 : WSDL リソース」ウィンドウが表示されます。

図 9-78 「検索と選択 : WSDL リソース」ウィンドウ



- 「検索」ボタンをクリックし、BSE アウトバウンド WSDL を選択して「OK」をクリックします。  
「ビジネス・サービスの作成」ウィンドウに戻ります。
- 「次」をクリックします。

5. 図 9-79 に示すように、デフォルト値を受け入れて、「作成」ボタンをクリックします。

図 9-79 「ビジネス・サービスの作成」ウィンドウ

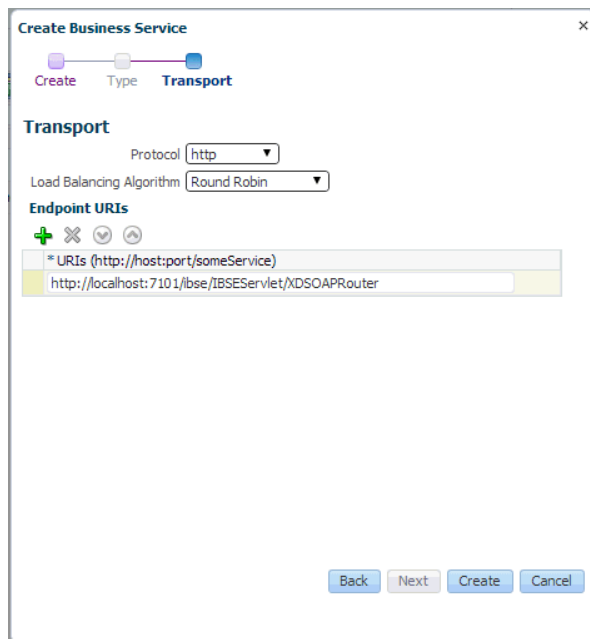
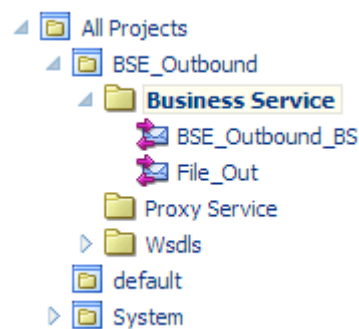


図 9-80 に示すように、「Business Service」フォルダの下に、作成された WSDL ベースのビジネス・サービスが表示されます。

図 9-80 WSDL ベースのビジネス・サービス



## 9.4.6 プロキシ・サービスでのパイプラインの構成

この項では、Oracle Service Bus コンソールを使用して、プロキシ・サービスを構成する方法について説明します。

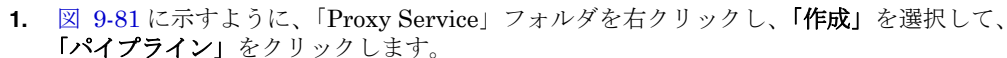
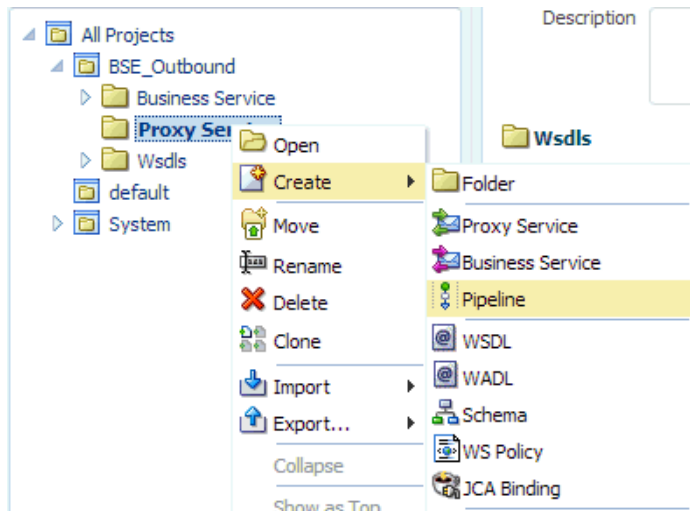
1.  9-81 に示すように、「Proxy Service」フォルダを右クリックし、「作成」を選択して、「パイプライン」をクリックします。

図 9-81 「パイプライン」オプション



「パイプラインの作成」ウィンドウが表示されます。

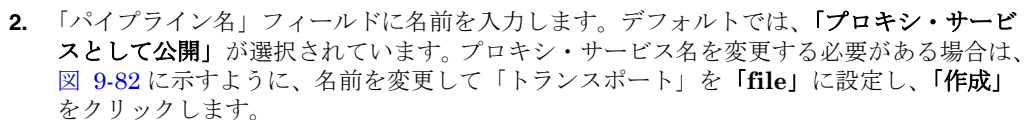
2. 「パイプライン名」フィールドに名前を入力します。デフォルトでは、「プロキシ・サービスとして公開」が選択されています。プロキシ・サービス名を変更する必要がある場合は、 9-82 に示すように、名前を変更して「トランスポート」を「file」に設定し、「作成」をクリックします。

図 9-82 「パイプラインの作成」ウィンドウ

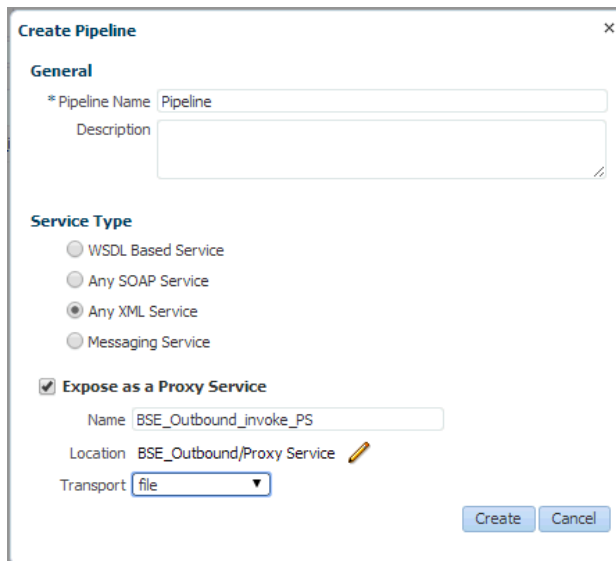
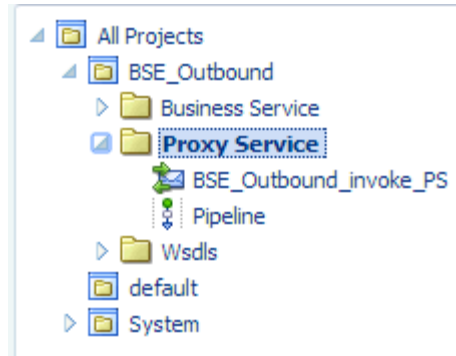


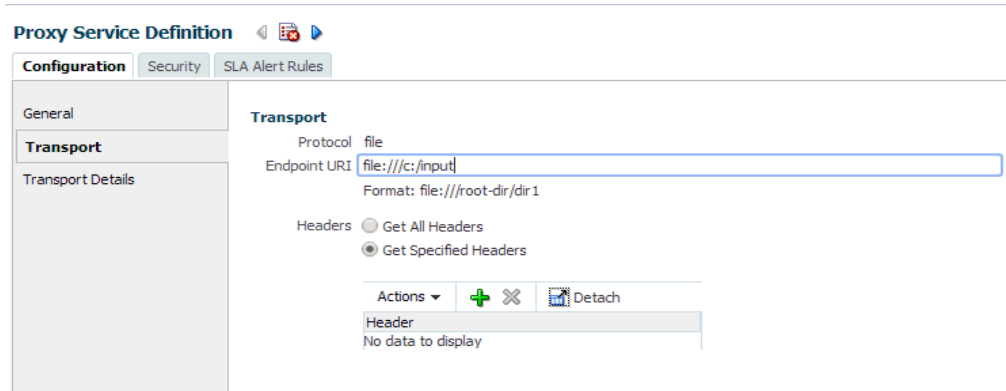
図 9-83 に示すように、「Proxy Service」の下に、作成されたパイプラインおよびプロキシ・サービスが表示されます。

図 9-83 「パイプライン」ノード



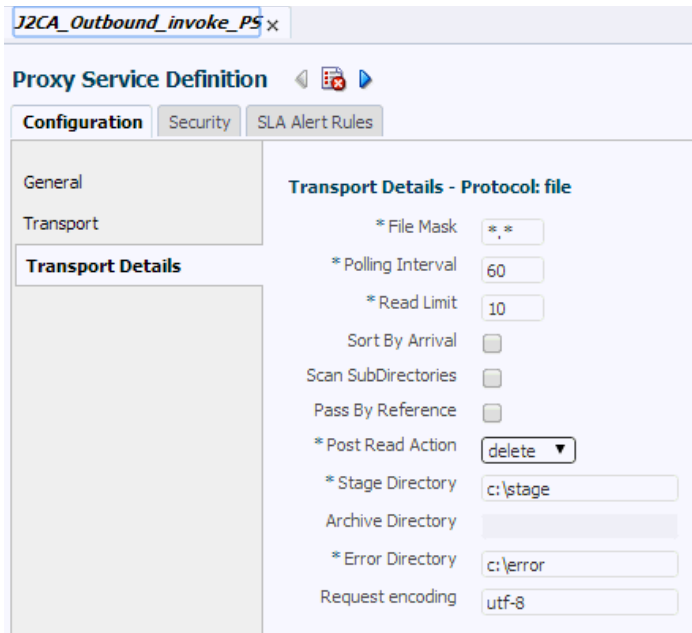
3. 作成されたプロキシ・サービスをダブルクリックし、左ペインの「トランスポート」をクリックします。図 9-84 に示すように、「エンドポイント URI」フィールドで入力場所を指定します。

図 9-84 トランスポート



4. 図 9-85 に示すように、左ペインの「トランスポートの詳細」をクリックして、「ステージ・ディレクトリ」および「エラー・ディレクトリ」の各フィールドに場所を指定します。

図 9-85 トランスポートの詳細



5. 図 9-86 に示すように、右隅の「すべて保存」アイコンをクリックします。

図 9-86 「すべて保存」アイコン



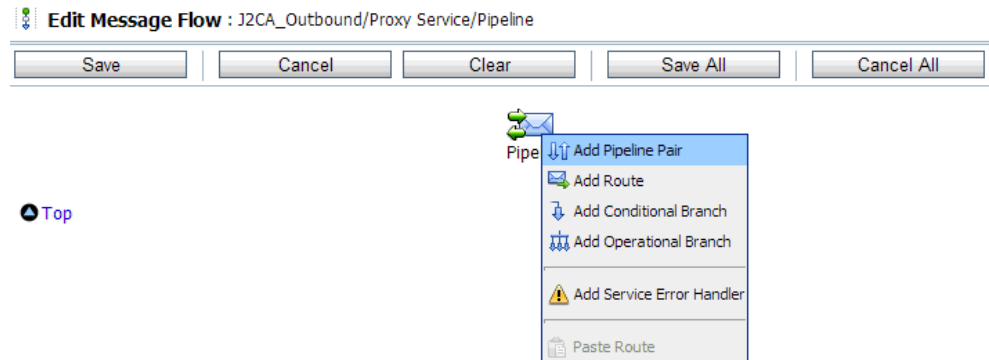
6. 「パイプライン」ノードをダブルクリックして、図 9-87 に示すように、右ペインの「メッセージ・フローを開く」アイコンをクリックしてメッセージ・フローを開きます。

図 9-87 「メッセージ・フローを開く」アイコン



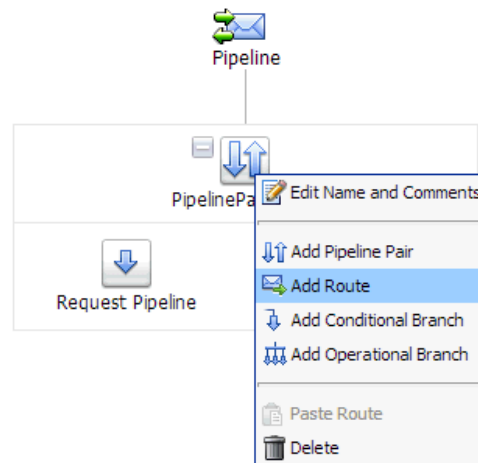
7. 図 9-88 に示すように、「プロキシ・サービス」アイコンをクリックして、メニューの「パイプライン・ペアの追加」を選択します。

図 9-88 「パイプライン・ペアの追加」オプション



8. 図 9-89 に示すように、「PipelinePairNode1」アイコンをクリックして、メニューの「ルートの追加」を選択します。

図 9-89 「ルートの追加」オプション



「RouteNode1」アイコンが「PipelinePairNode1」アイコンの下に追加されます。


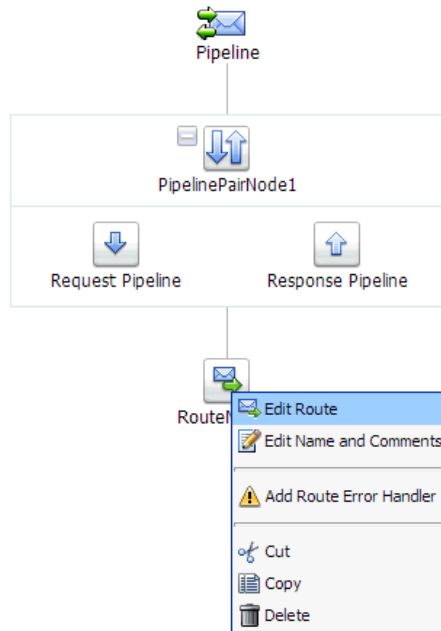
9.  9-90 に示すように、「RouteNode1」アイコンをクリックして、メニューの「ルート  
の編集」を選択します。

図 9-90 「ルート of の編集」オプション



「ステージ構成の編集」ワークスペース領域が表示されます。


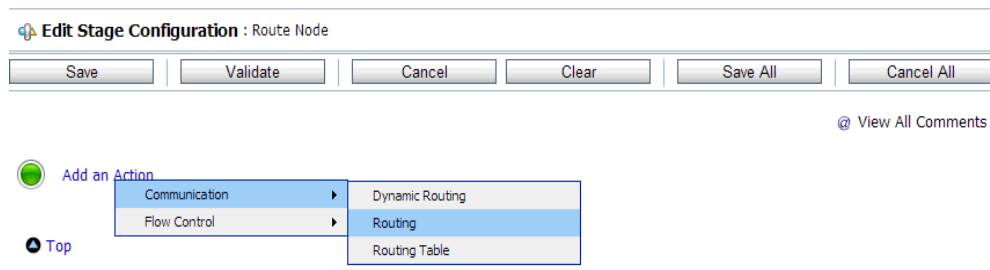
10.  9-91 に示すように、「アクションの追加」をクリックして「通信」を選択し、「ルー  
ティング」をクリックします。

図 9-91 「ステージ構成の編集」ワークスペース領域





11. 図 9-92 に示すように、<Service> をクリックします。

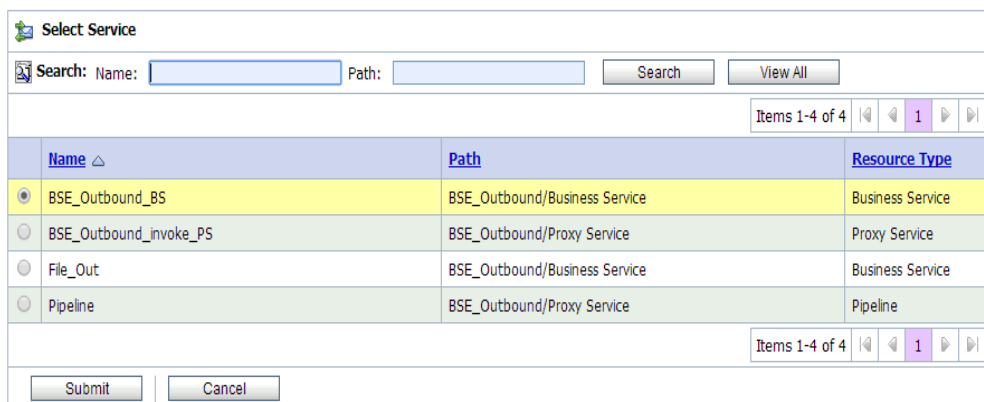
図 9-92 アクション



「サービスの選択」ダイアログが表示されます。

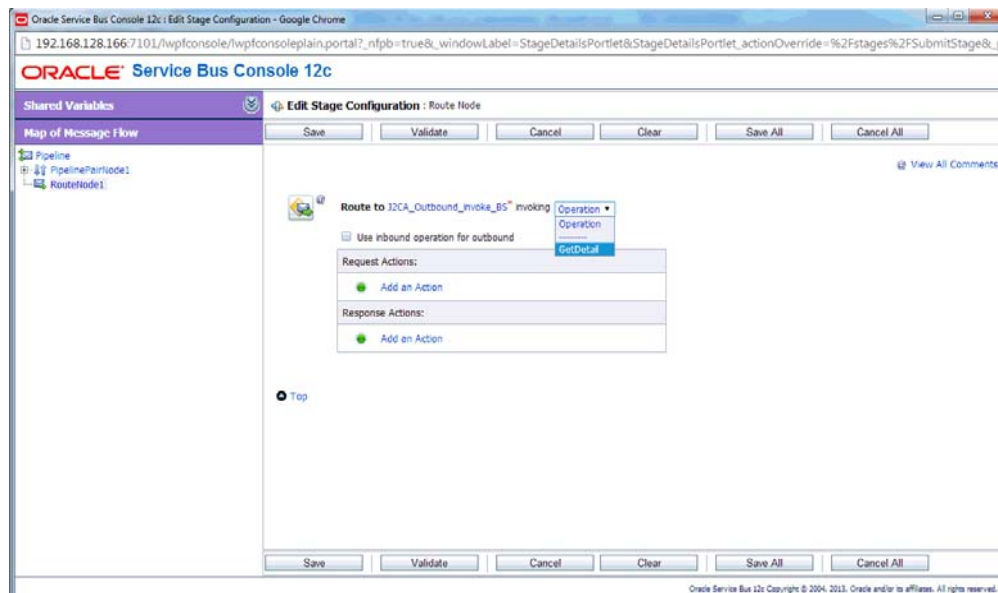
12. 図 9-93 に示すように、SAP R/3 用に構成された WSDL タイプのビジネス・サービスを選択して、「発行」をクリックします。

図 9-93 「サービスの選択」ダイアログ



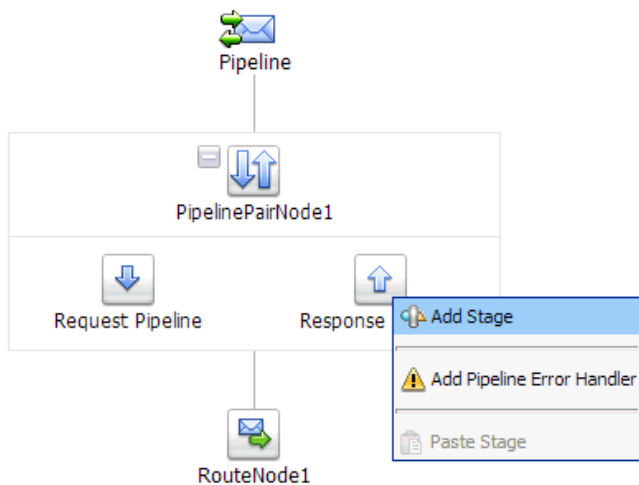
13. 図 9-94 に示すように、リストから、操作属性として SAP R/3 ビジネス・オブジェクトの名前 (GetDetail など) を選択し、「保存」をクリックします。

図 9-94 ステージ構成の編集



14. 図 9-95 に示すように、「レスポンス・パイプライン」アイコンをクリックして、メニューの「ステージの追加」を選択します。

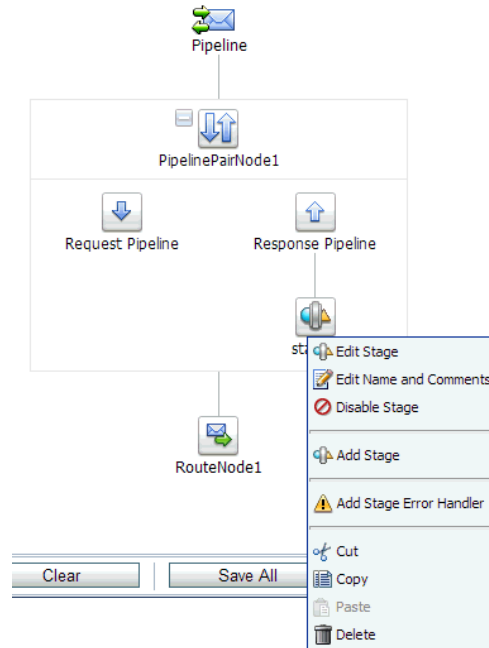
図 9-95 「レスポンス・パイプライン」アイコン



「ステージ 1」アイコンが「レスポンス・パイプライン」アイコンの下に追加されます。

15. 図 9-96 に示すように、「ステージ 1」アイコンをクリックして、メニューの「ステージの編集」を選択します。

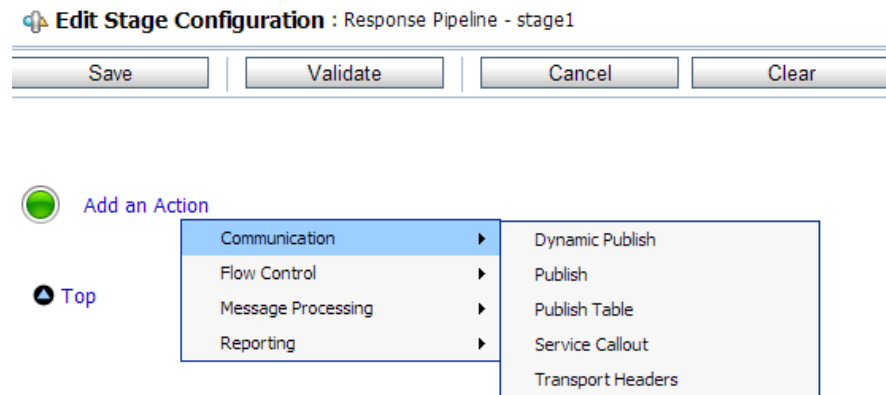
図 9-96 「ステージの編集」オプション



「ステージ構成の編集」ワークスペース領域が表示されます。

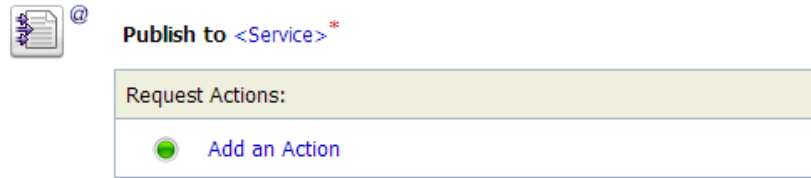
16. 図 9-97 に示すように、「アクションの追加」をクリックして「通信」を選択し、「パブリッシュ」をクリックします。

図 9-97 「ステージ構成の編集」ワークスペース領域



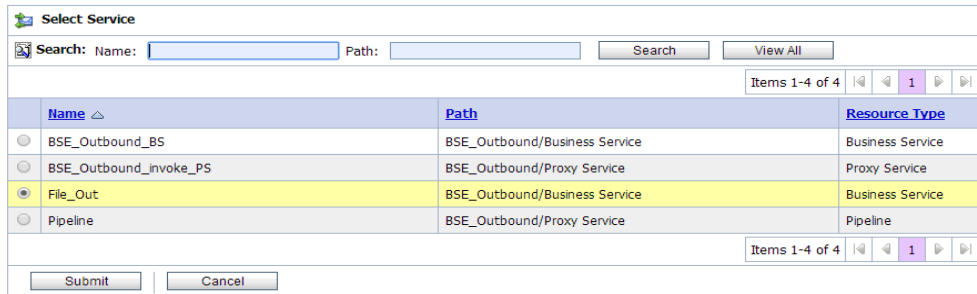
17. 図 9-98 に示すように、<Service> をクリックします。

図 9-98 <Service> アクション



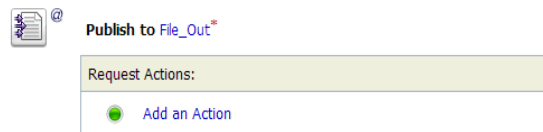
18. 図 9-99 に示すように、「サービスの選択」ダイアログでファイル・タイプ・ビジネス・サービスを選択し、「発行」をクリックします。

図 9-99 「サービスの選択」ダイアログ



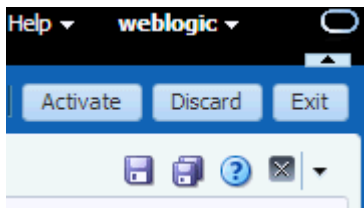
19. 図 9-100 に示すように、「すべて保存」をクリックします。

図 9-100 「すべて保存」ボタン



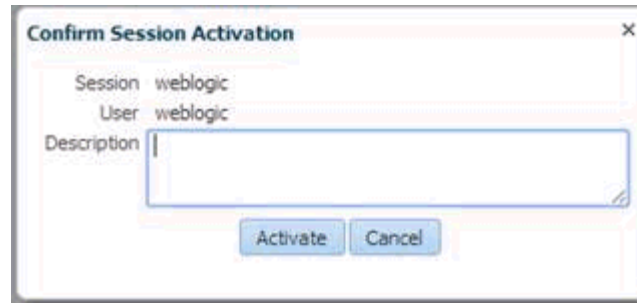
20. 図 9-101 に示すように、Oracle Service Bus セッションの右ペインの「アクティブ化」をクリックします。

図 9-101 「アクティブ化」ボタン



21. 図 9-102 に示すように、「アクティブ化」をクリックして変更を保存します。

図 9-102 セッションのアクティブ化の確認



22. 構成済の入力フォルダ (C:\input など) に、入力 XML ファイルをコピーして貼り付けます。

出力は、構成済の出力場所 (C:\output など) で受信されます。

## 9.5 Oracle Service Bus を使用した同期 SAP R/3 イベントの使用 (J2CA 構成)

この項では、Oracle Service Bus を使用して同期 SAP R/3 イベントを処理する方法について説明します。同期 SAP R/3 イベントの詳細は、6.6.1 項「概要」を参照してください。

---

**注意：** 同期イベントは、返信スキーマを含むオブジェクトについてのみサポートされており、IDoc についてはサポートされていません。

---

アプリケーション・アダプタ・インストールの次のフォルダに、このユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB\J2CA\SAP_Sample_J2CA_OSBSynchronous_Events
```

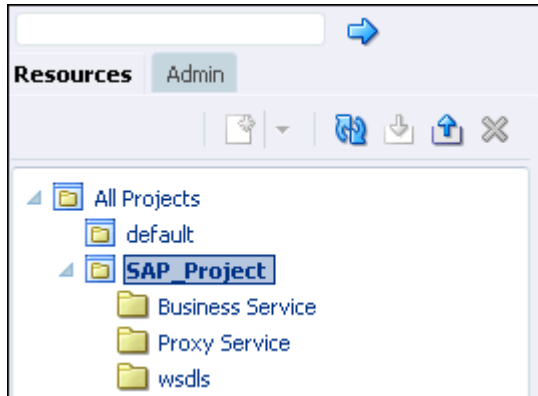
この項では、次の項目について説明します。

- 9.5.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」
- 9.5.2 項「アダプタ・ターゲットおよびチャネルの構成」
- 9.5.3 項「メッセージの WSDL ファイルの生成」
- 9.5.4 項「Oracle Service Bus コンソールでのビジネス・サービスの作成」
- 9.5.5 項「Oracle Service Bus コンソールでのプロキシ・サービスの作成」
- 9.5.6 項「プロキシ・サービスでのパイプラインの構成」

## 9.5.1 Oracle Service Bus の起動およびプロジェクト・フォルダの作成

図 9-103 に示すように、Oracle Service Bus を起動して、必要なプロジェクト・フォルダを作成します。

図 9-103 プロジェクト・エクスプローラ



詳細は、9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」を参照してください。

## 9.5.2 アダプタ・ターゲットおよびチャネルの構成

アプリケーション・エクスプローラを起動し、J2CA 構成を使用して SAP R/3 用に 2 つのアダプタ・ターゲットおよび 1 つのチャネルを作成します。

詳細は、第 4 章「構成 Oracle Application Adapter for SAP R/3」を参照してください。

## 9.5.3 メッセージの WSDL ファイルの生成

メッセージの WSDL ファイルを生成するには、次のようにします。

1. WSDL ファイルを生成する前に、チャネルが開始されていないことを確認します。
2. 最初の SAP アダプタ・ターゲット (target1) に接続します。
3. BAPI\_COMPANYCODE\_GETLIST のインバウンド WSDL ファイルを生成します (「リモート関数モジュール」→「Financial Accounting」→「0002 - Company Code」ビジネス・オブジェクトを選択)。
4. 2 番目の SAP アダプタ・ターゲット (target2) に接続します。
5. BAPI\_COMPANYCODE\_GETLIST のアウトバウンド WSDL ファイルを生成します (「リモート関数モジュール」→「Financial Accounting」→「0002 - Company Code」ビジネス・オブジェクトを選択)。

詳細は、9.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ」および 9.3.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ」を参照してください。

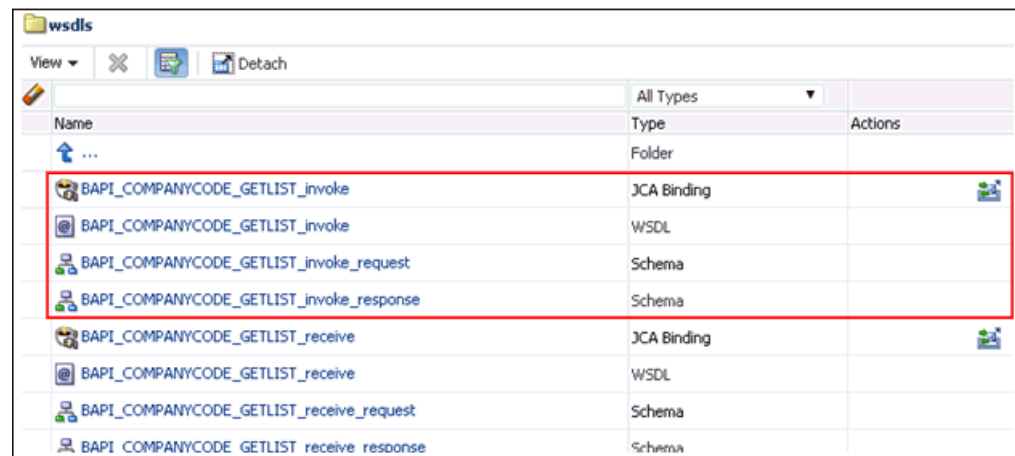
## 9.5.4 Oracle Service Bus コンソールでのビジネス・サービスの作成

この項では、Oracle Service Bus コンソールでビジネス・サービスを作成する方法について説明します。

1. Oracle Service Bus コンソールで、上部にある「作成」をクリックし、左ペインで作成された WSDL フォルダを選択します。

☒ 9-104 に示すように、WSDL ファイル、JCA バインディング・ファイルおよび XSD ファイルが右ペインに表示されます。

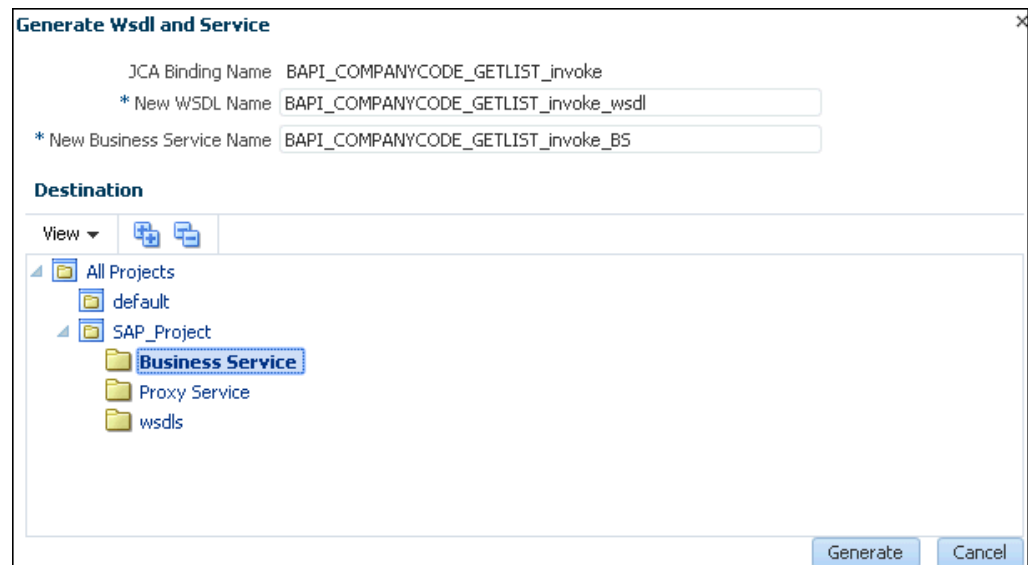
図 9-104 WSDL フォルダのコンテンツ



2. エクスポートされたアウトバウンド JCA バインドに対応するアイコンをクリックします。

☒ 9-105 に示すように、「WSDL とサービスの生成」ページが開きます。

図 9-105 「WSDL とサービスの生成」ページ

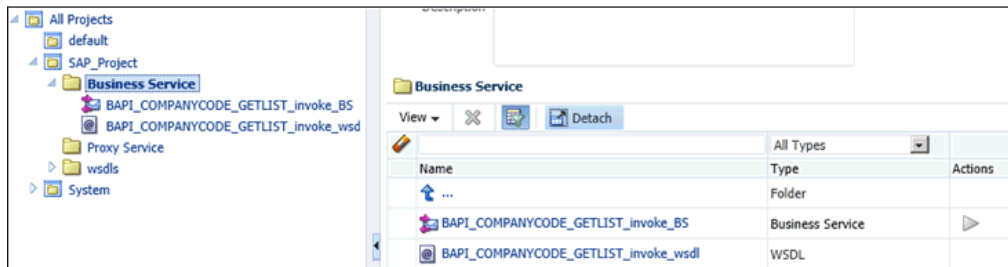


3. 対応するフィールドに、新しい WSDL 名および新しいビジネス・サービス名を指定します。
4. 「宛先」領域で、対応するプロジェクトから「**Business Service**」サブフォルダを選択します。

5. 「生成」をクリックします。

図 9-106 に示すように、左ペインのプロジェクト・フォルダ (SAP\_Project など) の下で「Business Service」フォルダを選択し、WSDL およびサービス・リソースが正常に作成されていることを確認します。

図 9-106 WSDL およびサービス・リソースの確認



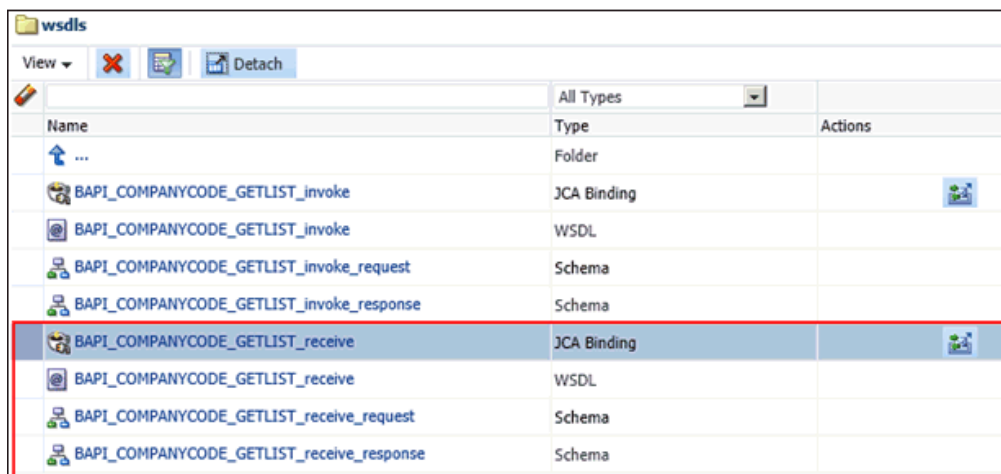
### 9.5.5 Oracle Service Bus コンソールでのプロキシ・サービスの作成

この項では、Oracle Service Bus コンソールでプロキシ・サービスを作成する方法について説明します。

1. Oracle Service Bus コンソールの左ペインで、作成された WSDL フォルダを選択します。

図 9-107 に示すように、WSDL ファイル、JCA バインディング・ファイルおよび XSD ファイルが右ペインに表示されます。

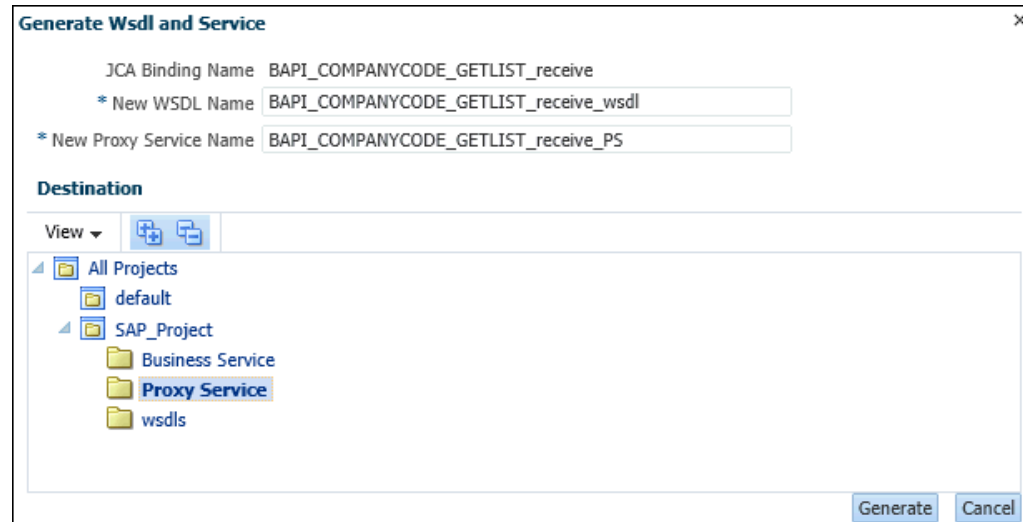
図 9-107 WSDL フォルダのコンテンツ





2. エクスポートされたインバウンド JCA バインドに対応するアイコンをクリックします。  
 図 9-108 に示すように、「WSDL とサービスの生成」ページが表示されます。

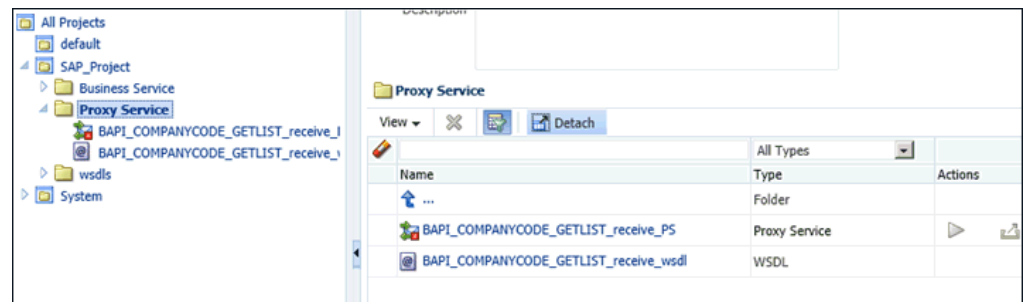
図 9-108 「WSDL とサービスの生成」 ページ



3. 対応するフィールドに、新しい WSDL 名および新しいプロキシ・サービス名を指定します。
4. 「宛先」領域で、対応するプロジェクトから「**Proxy Service**」サブフォルダを選択します。
5. 「生成」をクリックします。

図 9-109 に示すように、左ペインのプロジェクト・フォルダ (SAP\_Project など) の下で「Proxy Service」フォルダを選択し、WSDL およびサービス・リソースが正常に作成されていることを確認します。

図 9-109 WSDL およびサービス・リソースの確認



6. 「Proxy Service」を右クリックし、「作成」を選択して、「パイプライン」をクリックします。  
 図 9-110 に示すように、「パイプラインの作成」ウィンドウが表示されます。

図 9-110 「パイプラインの作成」ウィンドウ

7. パイプライン名を入力して、「サービス・タイプ」として「WSDL ベース・サービス」を選択し、「名前」フィールドの横にある「検索」アイコンをクリックします。  
 「検索と選択 : WSDL リソース」ウィンドウが表示されます。

- 「検索」をクリックし、表示されるリストから、プロキシ・サービスで使用可能な「BAPI\_COMPANYCODE\_GETLIST\_receive\_wsd1」を選択して、「OK」をクリックします。

図 9-111 に示すように、「パイプラインの作成」ウィンドウに戻ります。

図 9-111 「パイプラインの作成」ウィンドウ

**Create Pipeline**

**General**

\* Pipeline Name

Description

**Service Type**

WSDL Based Service

Name  🔍

Path

Binding  ▼

Any SOAP Service

Any XML Service

Messaging Service

Expose as a Proxy Service

Name

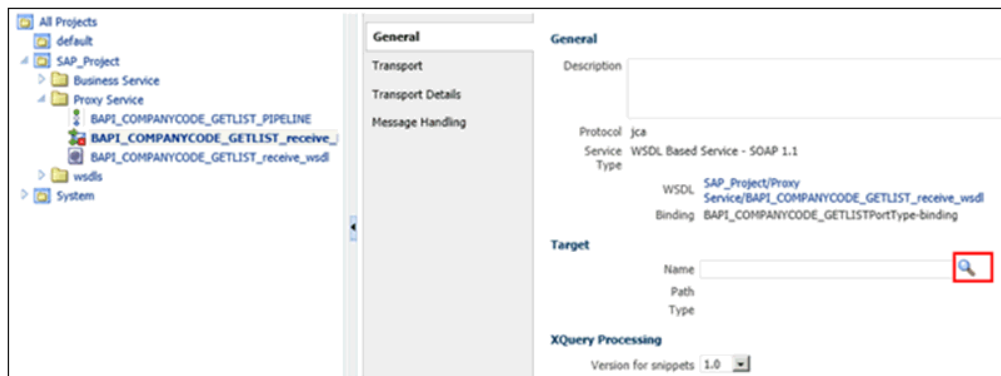
Location  ✎

Transport  ▼

- 「プロキシ・サービスとして公開」オプションの選択を解除し、「作成」をクリックします。「Proxy Service」フォルダの下に、パイプラインが作成されてリストされます。

10. 図 9-112 に示すように、「Proxy Service」フォルダの下で「BAPI\_COMPANYCODE\_GETLIST\_receive\_PS」をダブルクリックし、右ペインの「ターゲット」領域で「検索」アイコンをクリックします。

図 9-112 「検索」アイコン



「検索と選択：サービス・リソース」ウィンドウが表示されます。

11. 「検索」をクリックし、表示されるリストから、プロキシ・サービスで使用可能な「BAPI\_COMPANYCODE\_GETLIST\_PIPELINE」を選択して、「OK」をクリックします。
12. 左隅の「保存」/「すべて保存」をクリックします。

## 9.5.6 プロキシ・サービスでのパイプラインの構成

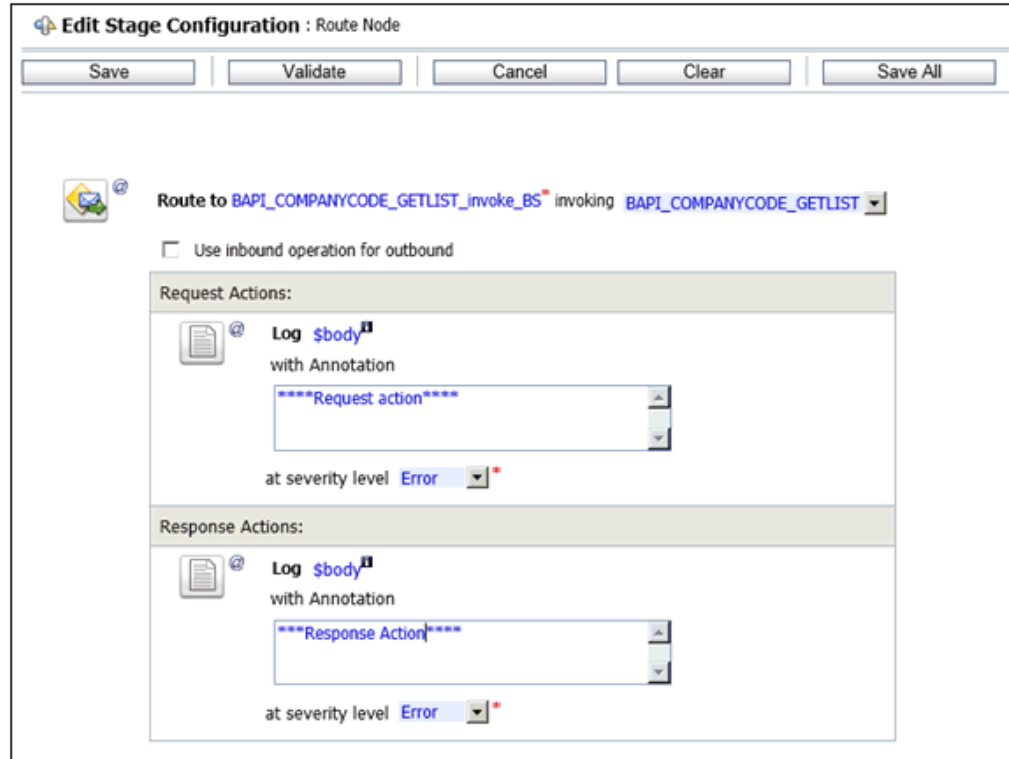
この項では、プロキシ・サービスでパイプラインを構成する方法について説明します。

1. 「Proxy Service」フォルダの下で作成されたパイプラインをクリックし、右ペインの上部にある「メッセージ・フローを開く」アイコンをクリックします。
2. 「メッセージ・フローの編集」ワークスペース領域で、表示される「プロキシ・サービス」アイコンをクリックし、メニューから「ルートの追加」を選択します。
3. 「RouteNode1」アイコンをクリックして、メニューの「ルートの編集」を選択します。
4. 「アクションの追加」をクリックして、メニューの「通信」を選択し、「ルーティング」をクリックします。
5. <Service> をクリックします。
6. 作成されたビジネス・サービスを選択して、「発行」をクリックします。
7. リストから、「BAPI\_COMPANYCODE\_GETLIST」操作を選択します。
8. 「リクエスト・アクション」の下で、「アクションの追加 - レポート」をクリックし、「ログ」を選択します。
9. ログ <Expression> をクリックします。
10. 「XQuery/XSLT 式エディタ」ページで \$body と入力し、「検証」をクリックしてから「保存」をクリックします。
11. 「リクエスト・アクション」セクションの「ログ」領域の下にある「重大度レベル」リストから「エラー」を選択します。
12. 「レスポンス・アクション」の下で、「アクションの追加 - レポート」をクリックし、「ログ」を選択します。
13. ログ <Expression> をクリックします。
14. 「XQuery/XSLT 式エディタ」ページで \$body と入力し、「検証」をクリックしてから「保存」をクリックします。

15. 「レスポンス・アクション」セクションの「ログ」領域の下にある「重大度レベル」リストから「エラー」を選択します。

図 9-113 に示すように、最終的なパイプライン構造が表示されます。

図 9-113 最終的なパイプライン構造



16. 「保存」をクリックします。  
「メッセージ・フローの編集」ワークスペース領域に戻ります。
17. 「保存」をクリックします。
18. 画面の上部にある「アクティブ化」をクリックします。  
表示される「セッションのアクティブ化の確認」ウィンドウで、「アクティブ化」をクリックします。

## 9.6 Oracle Service Bus を使用した JMS プロキシ・サービスの構成 (J2CA 構成)

この項では、J2CA 構成の Oracle Service Bus を使用して JMS プロキシ・サービスを構成する方法について説明します。

1. Oracle Service Bus を起動して、必要なプロジェクト・フォルダを作成します。

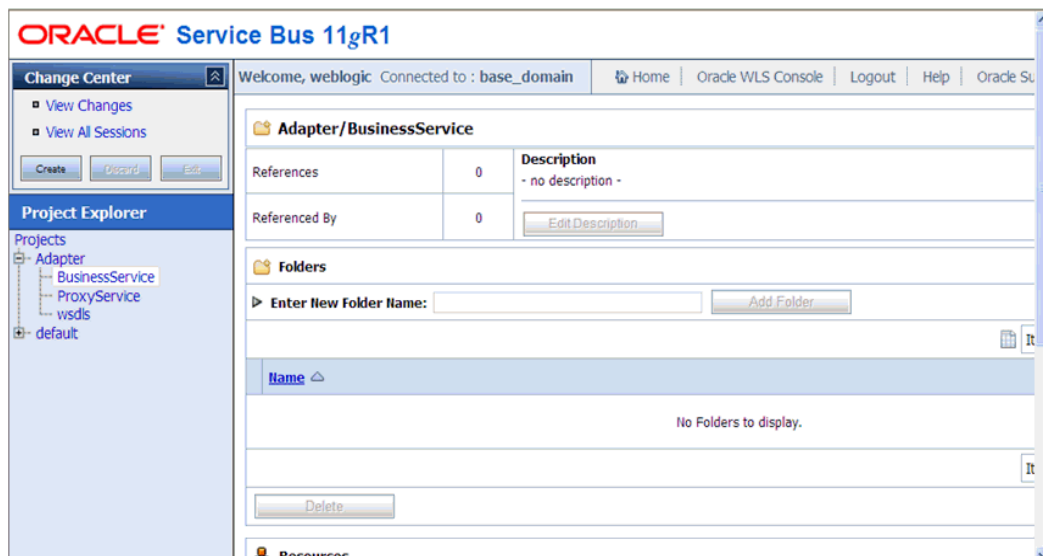
詳細は、9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」を参照してください。

2. WSDL を生成して、アプリケーション・エクスプローラから作成されたプロジェクト・フォルダにパブリッシュします。パブリッシュされた WSDL を使用して、ビジネス・サービスを作成します。

詳細は、9.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ」を参照してください。

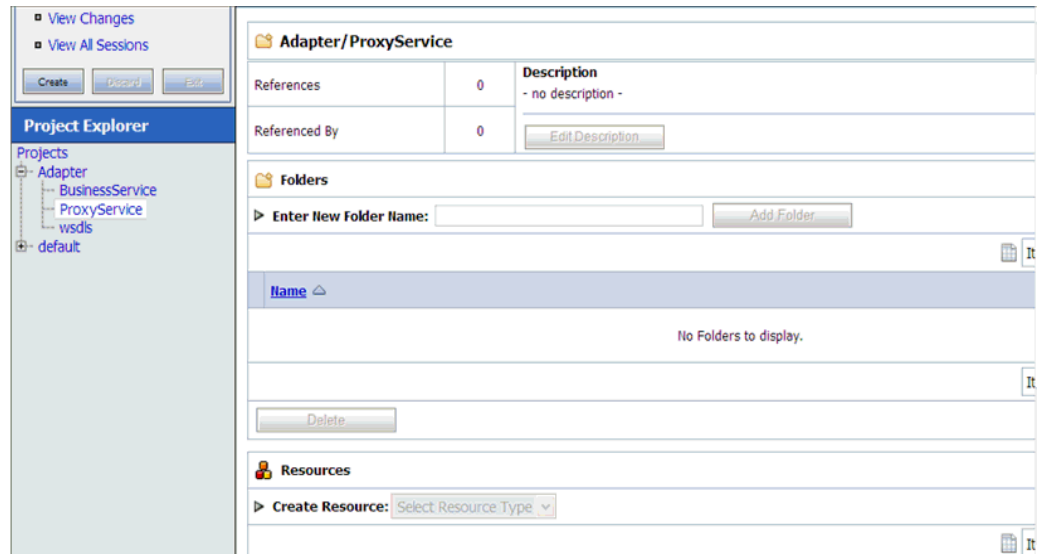
3. 図 9-114 に示すように、Service Bus コンソール・ページを開きます。

図 9-114 Service Bus コンソール



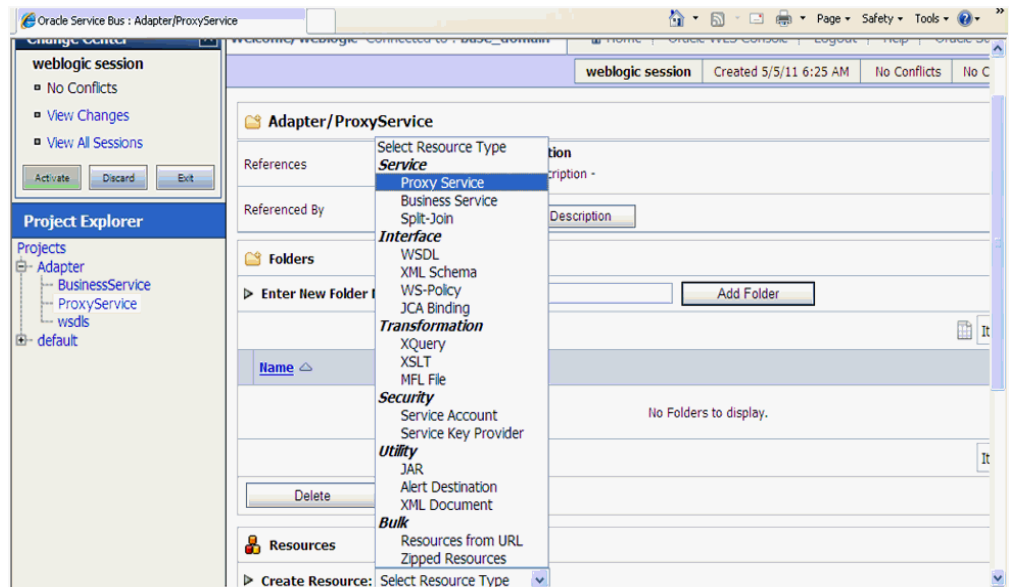
4. 図 9-115 に示すように、左ペインの「ProxyService」フォルダを選択し、「作成」をクリックします。

図 9-115 プロキシ・サービス



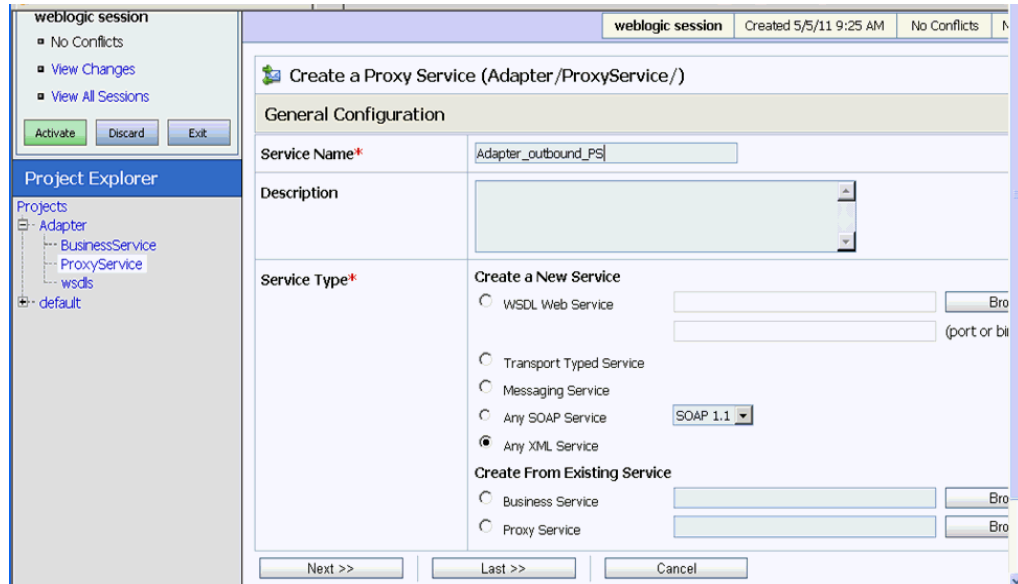
5. 図 9-116 に示すように、右ペインで、「リソースの作成」リストから「プロキシ・サービス」を選択します。

図 9-116 「リソースの作成」メニュー



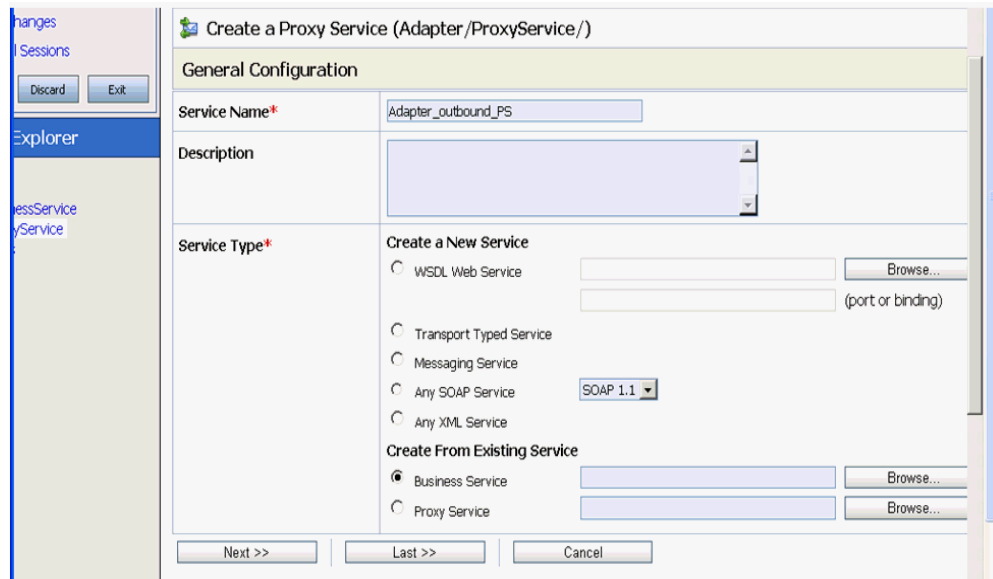
6. 図 9-117 に示すように、「サービス名」フィールドに適切な名前を入力します。

図 9-117 サービス名



7. 図 9-118 に示すように、「サービス・タイプ」セクションの「既存のサービスから作成」の下で、「ビジネス・サービス」ラジオ・ボタンを選択し、「参照」をクリックします。

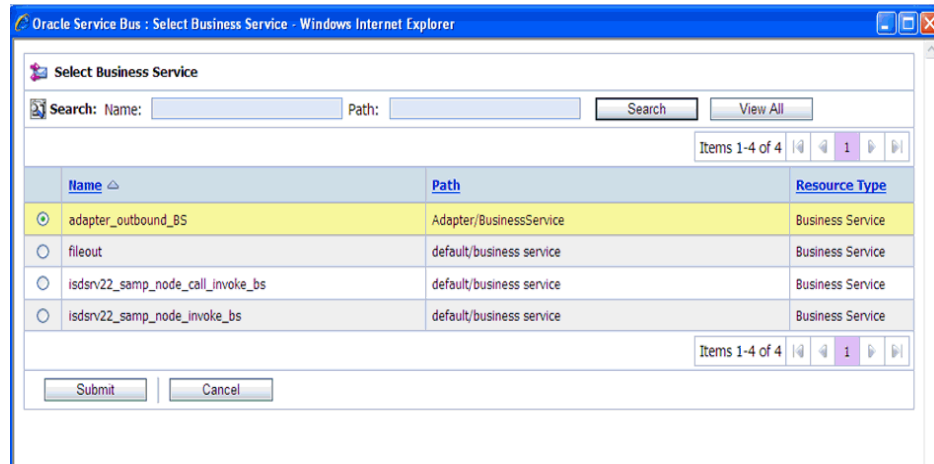
図 9-118 ビジネス・サービス





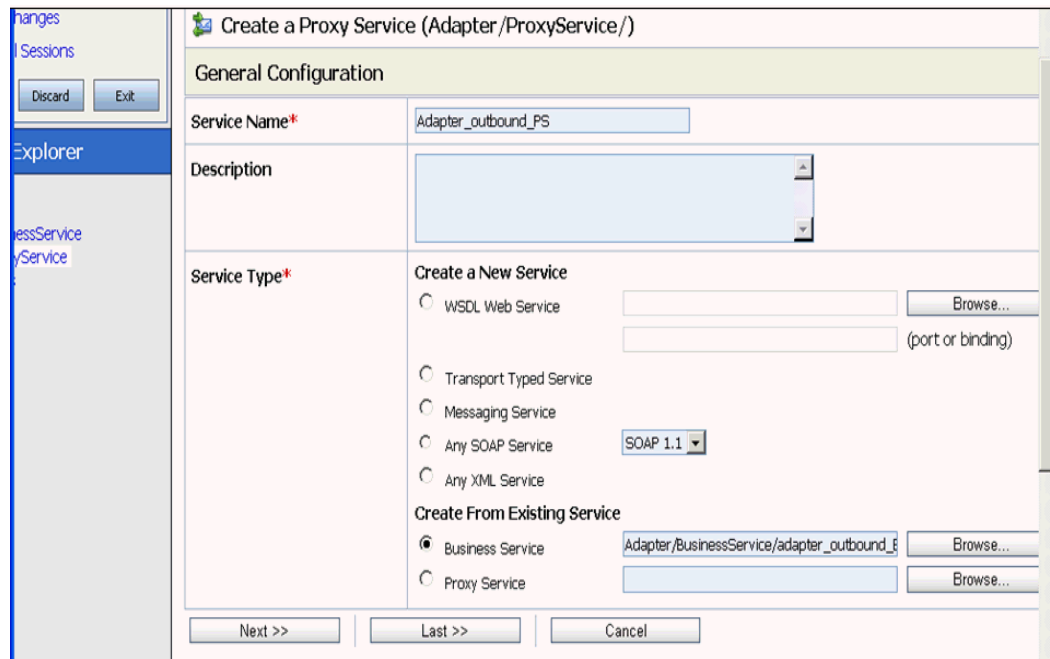
8. 図 9-119 に示すように、既存のビジネス・サービスを選択し、「発行」をクリックします。

図 9-119 既存のビジネス・サービス



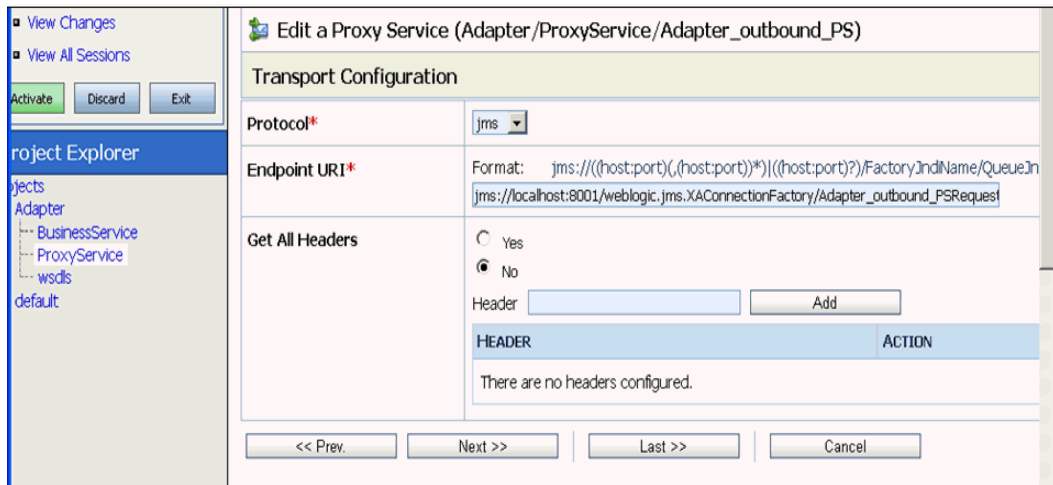
9. 図 9-120 に示すように、「次」をクリックします。

図 9-120 次



10. 図 9-121 に示すように、「プロトコル」リストから「jms」を選択し、「次」をクリックします。

図 9-121 「プロトコル」 リスト



11. 図 9-122 に示すように、次のパラメータを指定します。
- 「宛先タイプ」セクションで、「キュー」を選択します。
  - 「レスポンスが必要」チェック・ボックスを有効にします。
  - 「レスポンス・メッセージ・タイプ」セクションで、「テキスト」を選択します。
  - 「レスポンス URI」フィールドで、「トランスポート構成」で使用するエンドポイント URI を指定し、Request を Response に変更します。

次に例を示します。

```
jms://localhost:8001/weblogic.jms.XAConnectionFactory/Adapter_outbound_PSResponse
```

図 9-122 プロキシ・サービスの編集

12. 「次」をクリックします。

図 9-123 に示すように、「操作選択構成」ペインが表示されます。

図 9-123 「操作選択構成」ペイン

13. 「SOAP 本体タイプ」が選択されていることを確認して、「次」をクリックします。

14. 図 9-124 に示すように、「トランザクションが必要」ボックスを有効にし、「次」をクリックします。

図 9-124 メッセージ処理

Message Handling	
Transaction Required	<input checked="" type="checkbox"/> Enabled
Same Transaction For Response	<input type="checkbox"/> Enabled
Content Streaming	<input type="checkbox"/> Enabled
	<b>Buffer Type</b> <input type="radio"/> Memory Buffer <input type="radio"/> Disk Buffer
	<b>Compression</b> <input type="checkbox"/> Enabled

<< Prev.      Next >>      Last >>      Cancel

15. 図 9-125 に示すように、「保存」をクリックします。

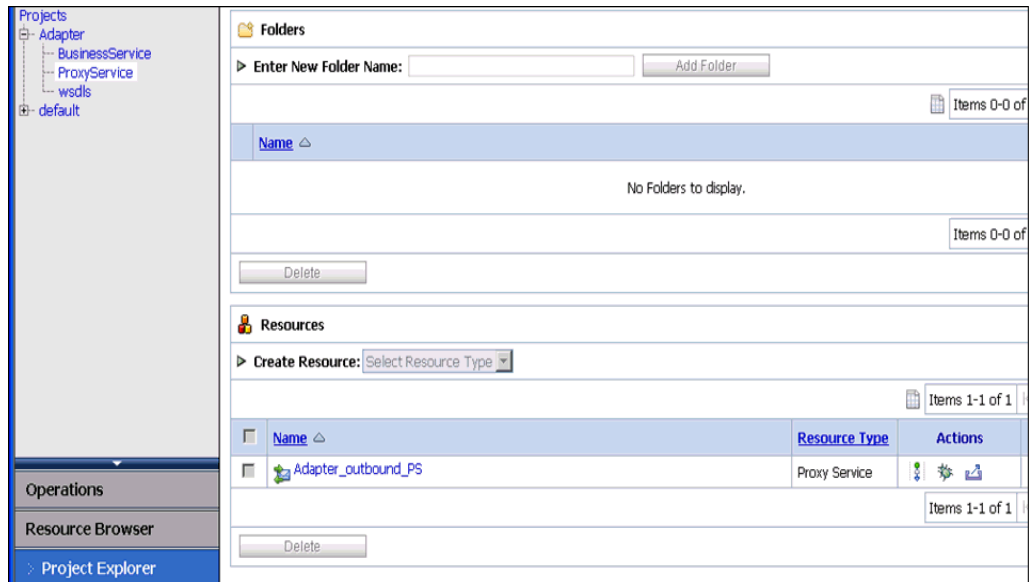
図 9-125 保存

Use SSL	Disabled
Is Response Required	Enabled
Request Encoding	UTF-8
Response Encoding	UTF-8
Response Pattern	JMSCorrelationID
JNDI Timeout	0
Response URI	jms://localhost:8001/weblogic.jms.XAConnectionFactory/Adapter_outbound_BSResponse
Response Message Type	Text
Client Response Timeout	300
Is XA Required	False
Operation Selection Configuration	
Selection Algorithm	SOAP Body Type
Message Handling Configuration	
Transaction Required	Enabled
Same Transaction For Response	Disabled
Content Streaming	Disabled

<< Prev.      Save      Cancel

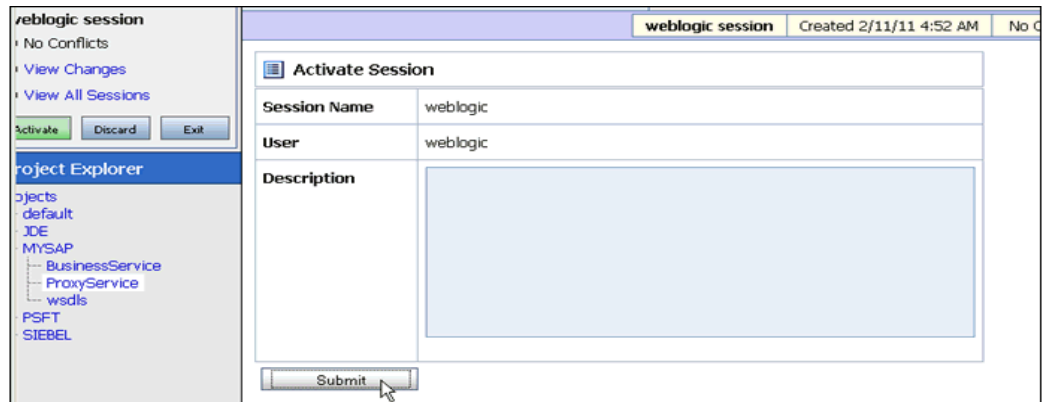
図 9-126 に示すように、作成されたプロキシ・サービスが保存されます。

図 9-126 プロキシ・サービス



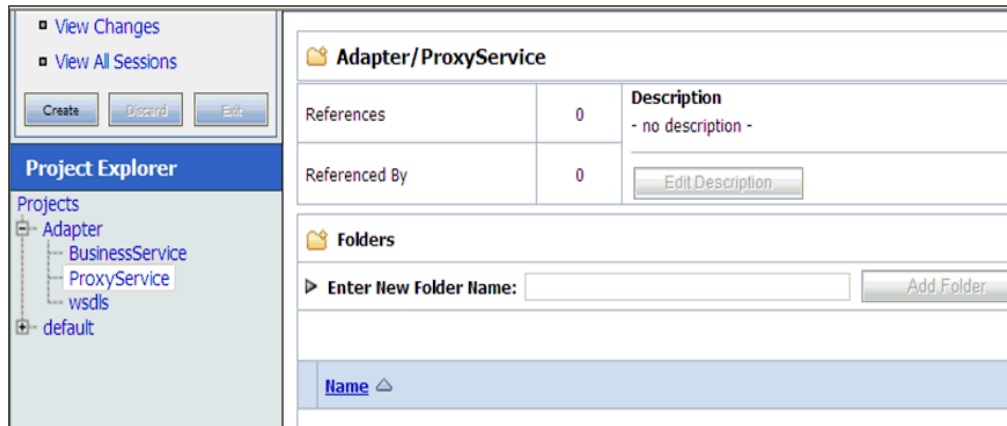
16. 図 9-127 に示すように、左ペインで「アクティブ化」をクリックしてから、「発行」をクリックします。

図 9-127 セッションのアクティブ化



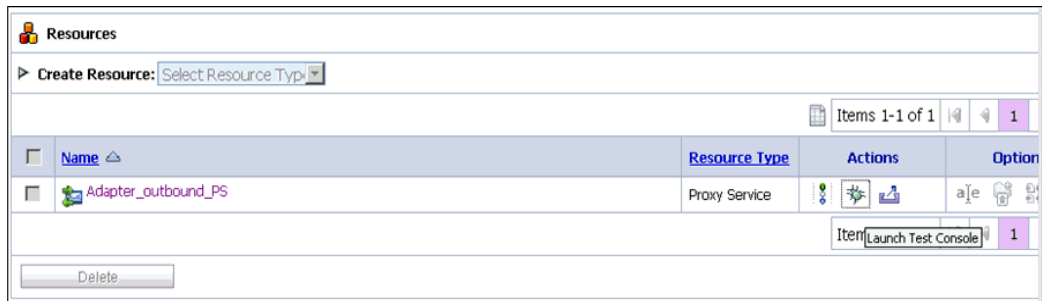
17. 図 9-128 に示すように、左ペインの「プロジェクト」フォルダの下で「ProxyService」をクリックします。

図 9-128 アダプタ /ProxyService



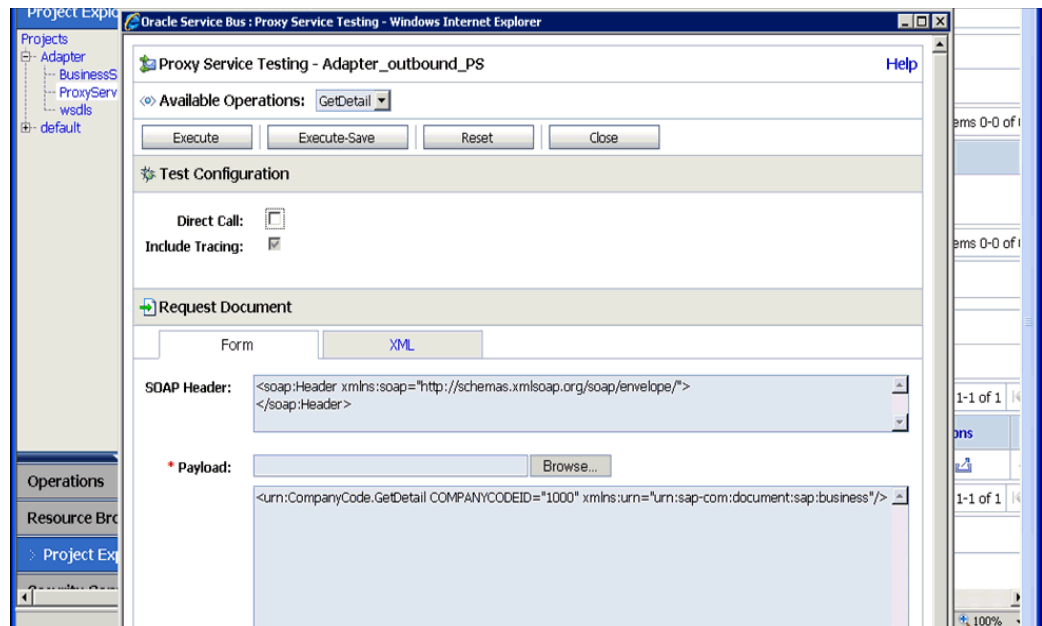
18. 図 9-129 に示すように、作成されたプロキシ・サービスの「テスト・コンソールの起動」アイコンをクリックします。

図 9-129 「テスト・コンソールの起動」アイコン



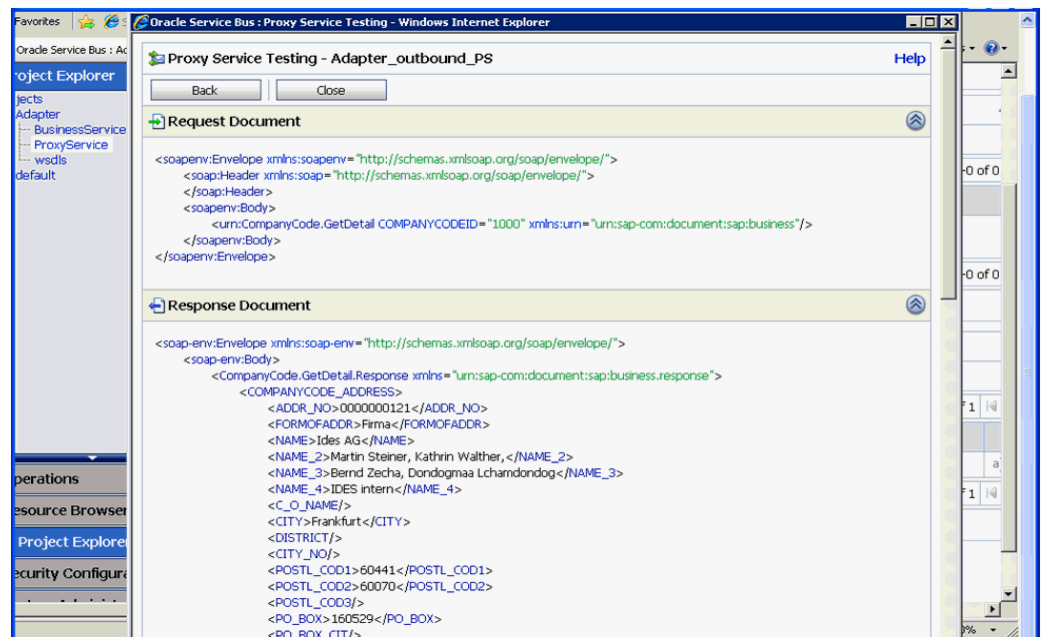
19. 図 9-130 に示すように、「ペイロード」に入力値を指定し、「直接呼出し」ボックスの選択を解除して、「実行」をクリックします。

図 9-130 プロキシ・サービス・テスト



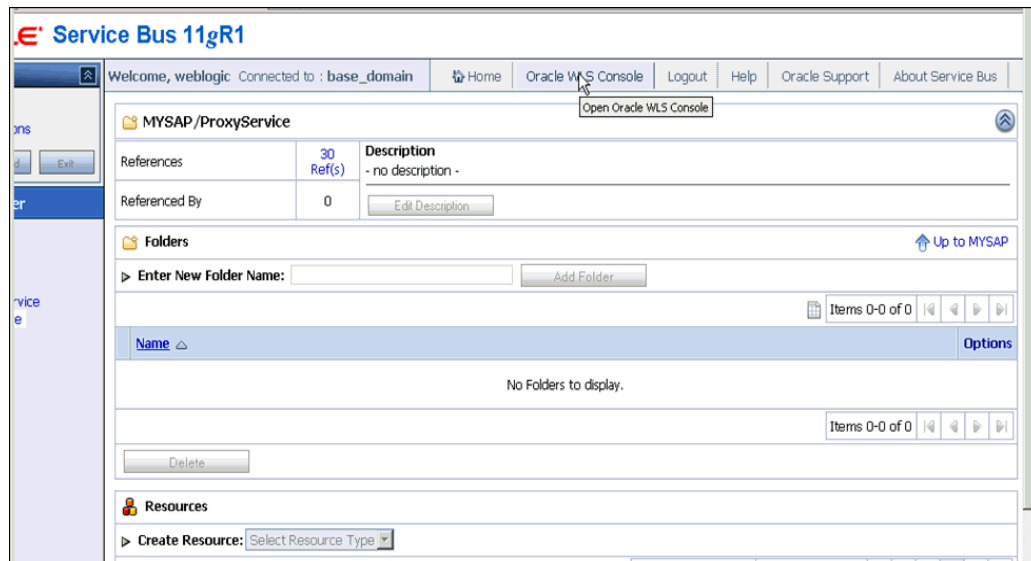
20. 図 9-131 に示すように、レスポンス・ドキュメントを確認してから、「閉じる」をクリックします。

図 9-131 レスポンス・ドキュメント



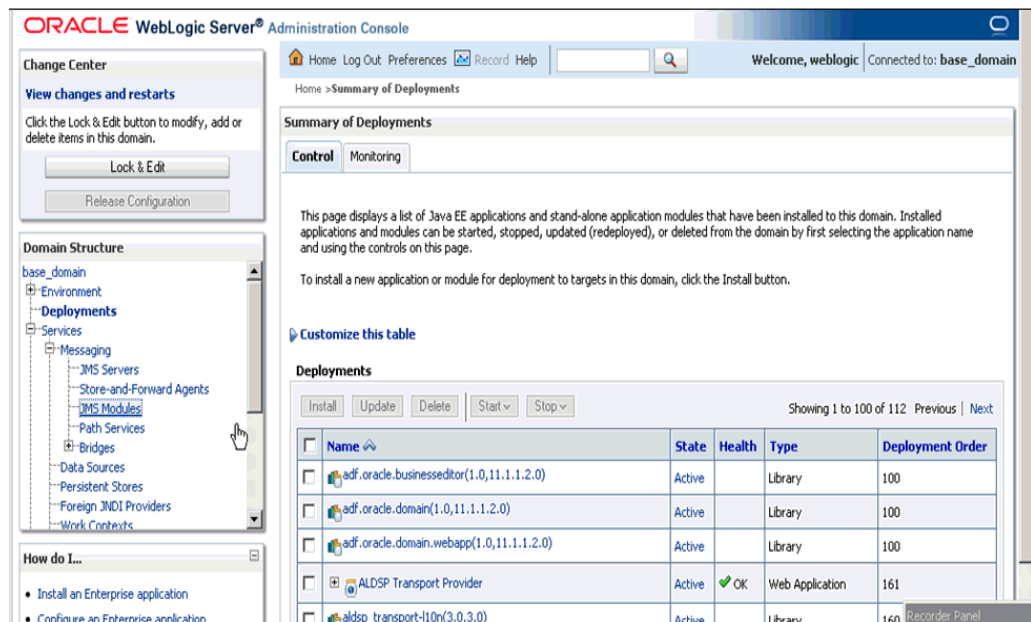
21. 図 9-132 に示すように、「Oracle WLS コンソール」タブをクリックします。

図 9-132 ProxyService



22. 図 9-133 に示すように、Oracle WLS コンソールで、「サービス」→「メッセージング」を展開し、「JMS モジュール」をクリックします。

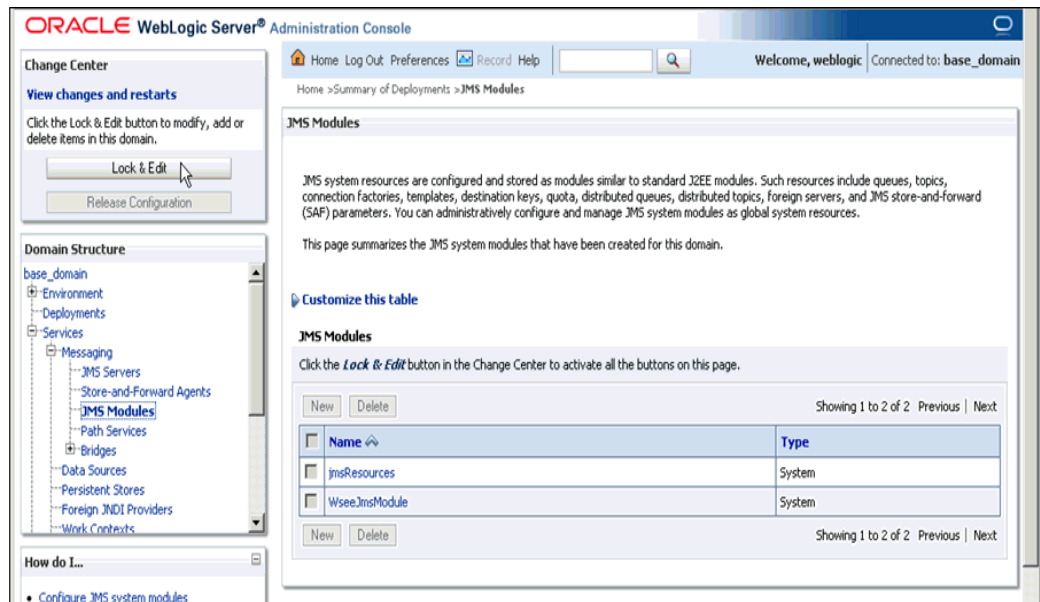
図 9-133 Oracle WLS コンソール





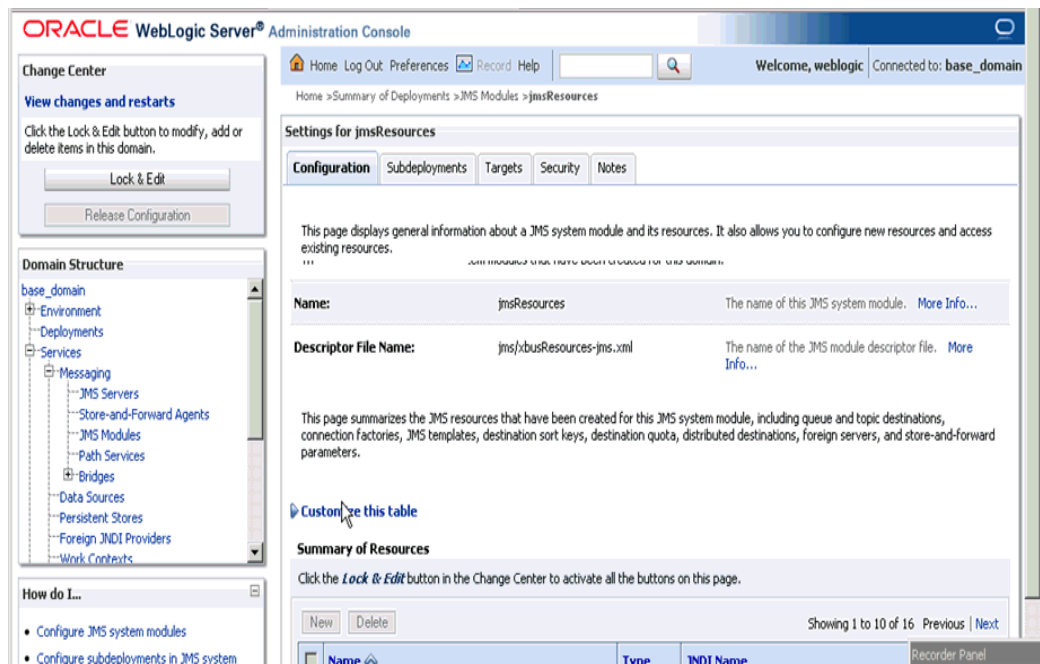
23. 図 9-134 に示すように、「jmsResources」をクリックします。

図 9-134 JMS モジュール



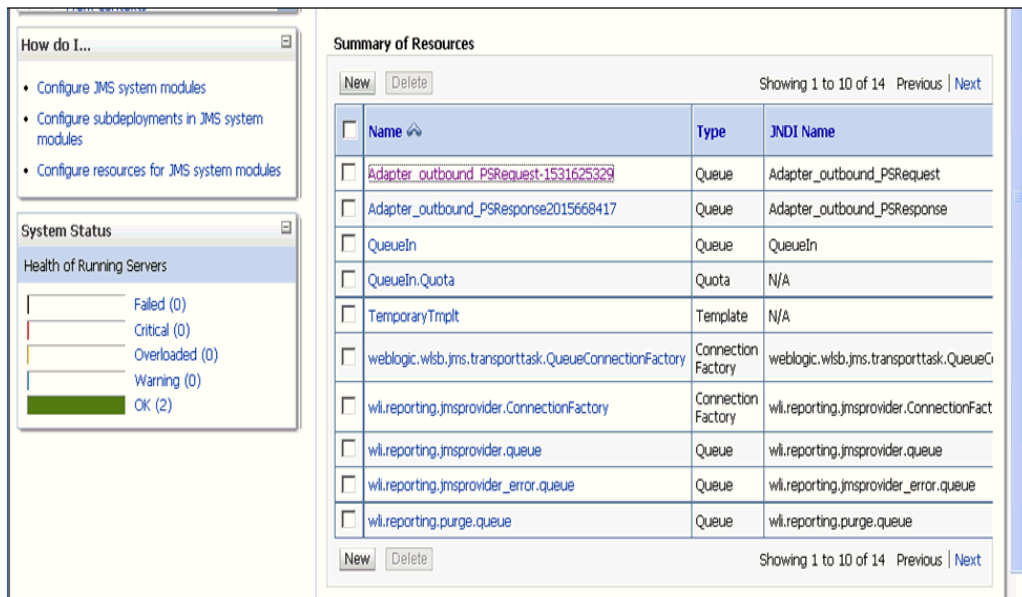
24. 図 9-135 に示すように、「ロックして編集」をクリックします。

図 9-135 構成設定



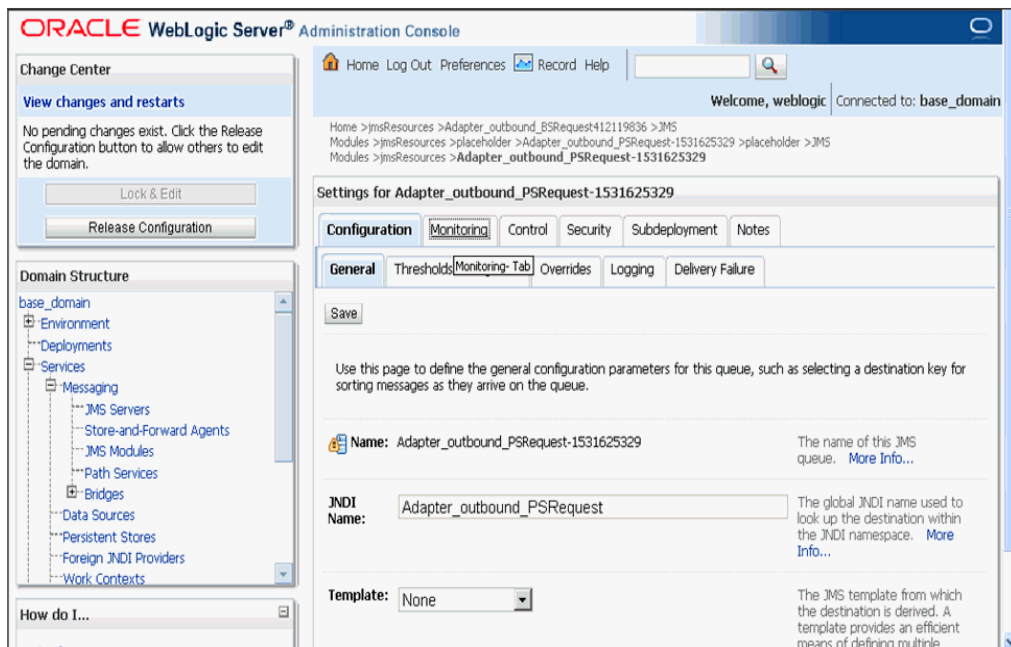
25. 図 9-136 に示すように、適切なリクエスト・リンク (**Adapter\_outbound\_PSRequest** など) をクリックします。

図 9-136 Adapter\_outbound\_PSRequest



26. 図 9-137 に示すように、「モニタリング」タブをクリックします。

図 9-137 「モニタリング」タブ



27. 図 9-138 に示すように、チェック・ボックスを有効にして「メッセージの表示」をクリックします。

図 9-138 アダプタ設定

Home > jmsResources > Adapter\_outbound\_PSRequest412119836 > JMS  
 Modules > jmsResources > placeholder > Adapter\_outbound\_PSRequest-1531625329 > placeholder > JMS  
 Modules > jmsResources > Adapter\_outbound\_PSRequest-1531625329

Settings for Adapter\_outbound\_PSRequest-1531625329

Configuration **Monitoring** Control Security Subdeployment Notes

A JMS destination identifies a queue (Point-To-Point) or a topic (Pub/Sub) that is targeted to a JMS server.  
 This page summarizes the active JMS destinations that have been created for this JMS module.

Customize this table

Destinations (Filtered - More Columns Exist)

Show Messages Showing 1 to 1 of 1 Previous | Next

<input checked="" type="checkbox"/>	Name	Messages Current	Messages Pending	Messages Total	Consumers Current	Consumers High	Consumers Total
<input checked="" type="checkbox"/>	.jmsResources!Adapter_outbound_PSRequest-1531625329	0	0	0	16	16	16

Show Messages Showing 1 to 1 of 1 Previous | Next

28. 図 9-139 に示すように、「新規」をクリックします。

図 9-139 JMS メッセージ

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Lock & Edit  
Release Configuration

Domain Structure

base\_domain  
 Environment  
 Deployments  
 Services  
 Messaging  
 JMS Servers  
 Store-and-Forward Agents  
 JMS Modules  
 Path Services  
 Bridges  
 Data Sources  
 Persistent Stores  
 Foreign JNDI Providers  
 Work Contexts

How do I...  
 Manage queue messages

Summary of JMS Messages

This page summarizes the available messages for a stand-alone queue, a distributed queue, or a topic durable subscriber. Use this page to view message details, create new messages, delete selected messages, move messages to another destination, export message contents in XML format to another file, import XML formatted message contents from another file, or drain all the messages from a destination.

Click on a message to view its contents.

Message Selector:  Apply

Customize this table

JMS Messages (Filtered - More Columns Exist)

New Delete Move Import Export Showing 1 to 0 of 0 Previous | Next

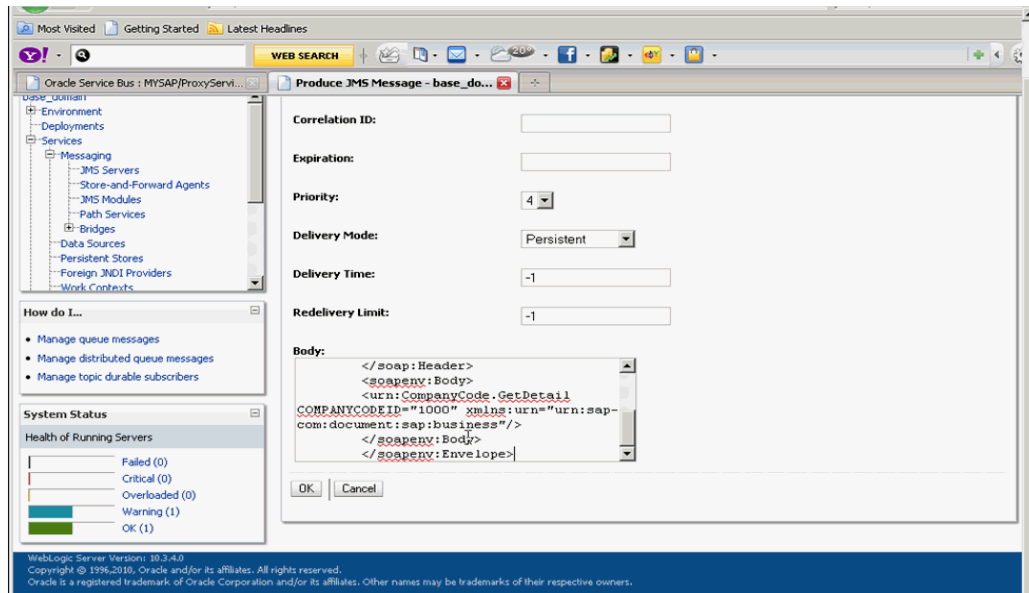
<input type="checkbox"/>	ID	CorrId	Time Stamp	State String	JMS Delivery Mode	Message Size
There are no items to display						

New Delete Move Import Export Showing 1 to 0 of 0 Previous | Next

Recorder Panel

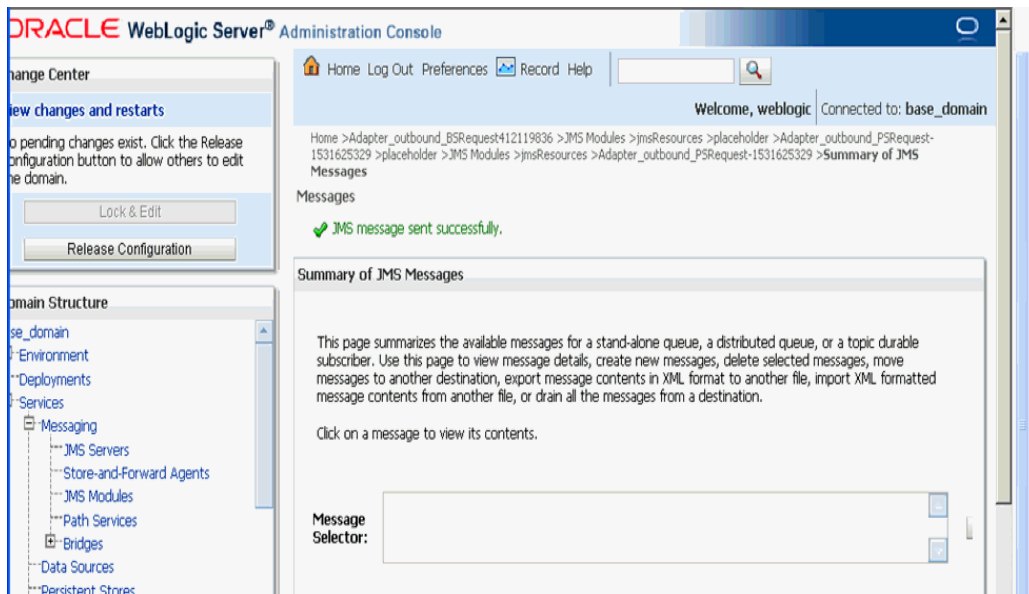
29. 図 9-140 に示すように、「本文」フィールドに入力ペイロードを指定し、「OK」をクリックします。

図 9-140 入力ペイロード



- 図 9-141 に示すように、成功メッセージが表示されます。

図 9-141 JMS 成功メッセージ



30. 図 9-142 に示すように、Oracle WLS コンソールで、「サービス」→「メッセージング」を展開し、「JMS モジュール」をクリックします。

図 9-142 JMS モジュール

The screenshot shows the Oracle WebLogic Server Administration Console. The left sidebar displays the Domain Structure tree, where 'JMS Modules' is selected under the 'Messaging' folder. The main content area is titled 'Summary of Deployments' and includes a 'Control' tab. Below the tab, there is a table of deployments. The table has columns for Name, State, Health, Type, and Deployment Order. The following table represents the data shown in the screenshot:

Name	State	Health	Type	Deployment Order
adf.oracle.businesseditor(1.0,11.1.1.2.0)	Active		Library	100
adf.oracle.domain(1.0,11.1.1.2.0)	Active		Library	100
adf.oracle.domain.webapp(1.0,11.1.1.2.0)	Active		Library	100
ALDSP Transport Provider	Active	OK	Web Application	161
aldsp_transport-110n(3.0,3.0)	Active		Library	160

31. 図 9-143 に示すように、「jmsResources」をクリックします。

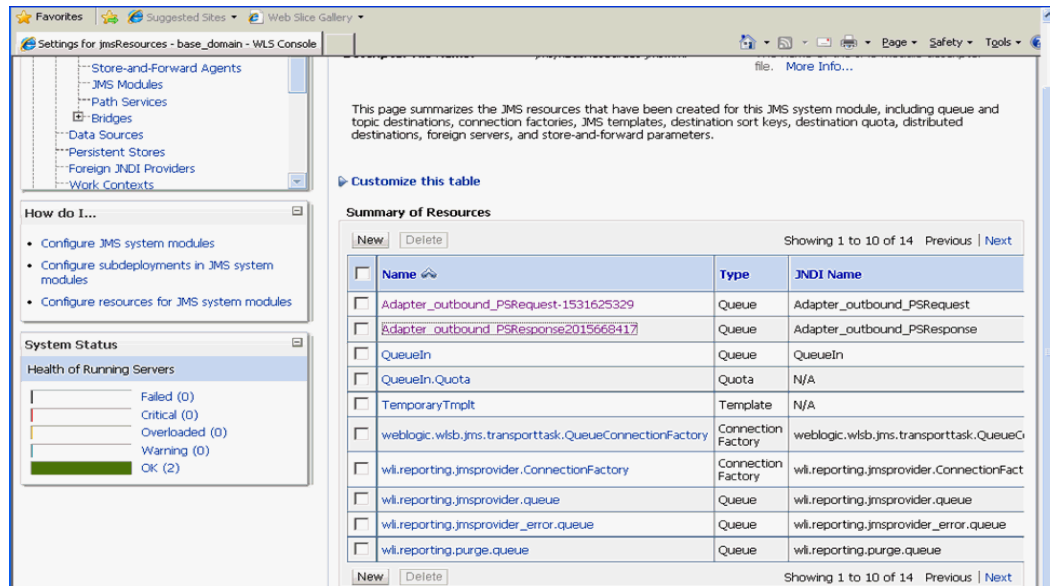
図 9-143 jmsResources

The screenshot shows the Oracle WebLogic Server Administration Console. The left sidebar displays the Domain Structure tree, where 'JMS Modules' is selected. The main content area is titled 'JMS Modules' and includes a 'Lock & Edit' button. Below the button, there is a table of JMS Modules. The table has columns for Name and Type. The following table represents the data shown in the screenshot:

Name	Type
jmsResources	System
WseeJmsModule	System

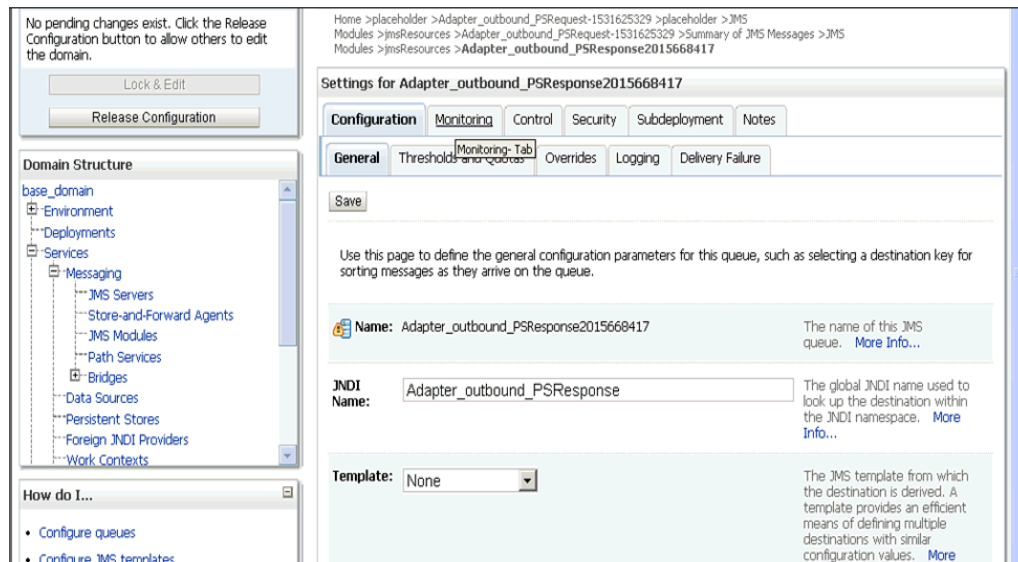
32. 図 9-144 に示すように、適切なレスポンス・リンク (**Adapter\_outbound\_PSResponse** など) をクリックします。

図 9-144 リソースのサマリー



33. 図 9-145 に示すように、「モニタリング」タブをクリックします。

図 9-145 「モニタリング」タブ



34. 図 9-146 に示すように、チェック・ボックスを有効にして「メッセージの表示」をクリックします。

図 9-146 宛先メッセージ

ORACLE WebLogic Server<sup>®</sup> Administration Console

Home Log Out Preferences Record Help

Welcome, weblogic Connected to: base\_domain

Home > placeholder > Adapter\_outbound\_PSRequest-1531625329 > placeholder > JMS Modules > jmsResources > Adapter\_outbound\_PSRequest-1531625329 > Summary of JMS Messages > JMS Modules > jmsResources > Adapter\_outbound\_PSResponse2015668417

Settings for Adapter\_outbound\_PSResponse2015668417

Configuration **Monitoring** Control Security Subdeployment Notes

A JMS destination identifies a queue (Point-To-Point) or a topic (Pub/Sub) that is targeted to a JMS server. This page summarizes the active JMS destinations that have been created for this JMS module.

Customize this table

Destinations (Filtered - More Columns Exist)

Show Messages Showing 1 to 1 of 1 Previous Next

<input checked="" type="checkbox"/>	Name	Messages Current	Messages Pending	Messages Total	Consumers Current	Consumers High
<input checked="" type="checkbox"/>	jmsResources! Adapter_outbound_PSResponse2015668417	1	0	1	16	16

Show Messages Showing 1 to 1 of 1 Previous Next

35. 図 9-147 に示すように、「ID」をクリックします。

図 9-147 JMS メッセージ

Click on a message to view its contents.

Message Selector:

Customize this table

JMS Messages (Filtered - More Columns Exist)

New Delete Move Import Export Showing 1 to 1 of 1 Previous Next

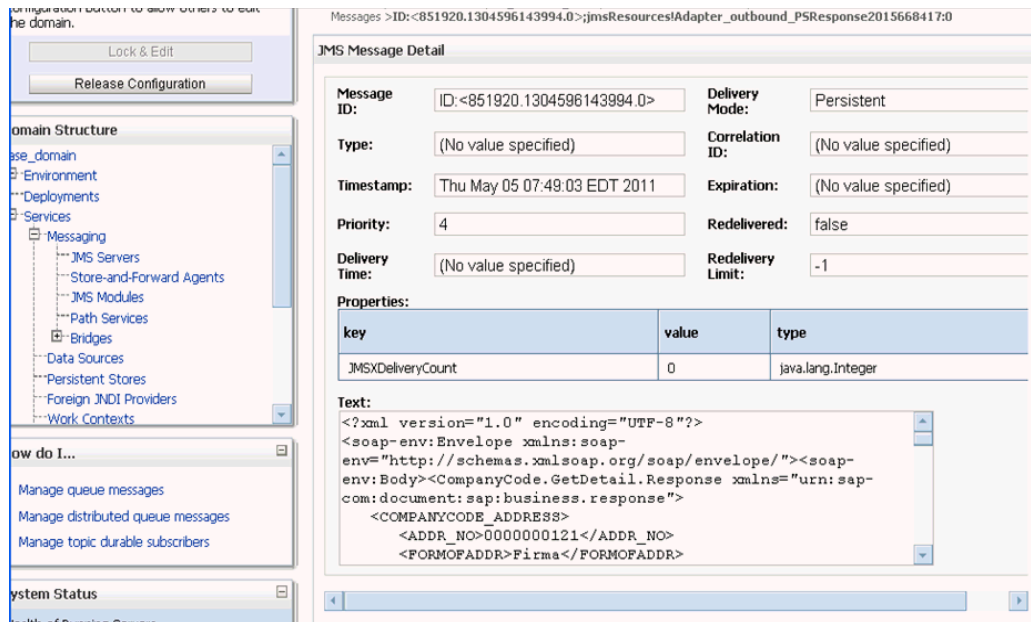
<input type="checkbox"/>	ID	Corrid	Time Stamp	State String	JMS Delivery Mode	Message Size
<input type="checkbox"/>	<a href="#">ID:&lt;851920.1304596143994.0&gt;</a>		Thu May 05 07:49:03 EDT 2011	visible	Persistent	2043

New Delete Move Import Export Showing 1 to 1 of 1 Previous Next



図 9-148 に示すように、「テキスト」フィールドの下にレスポンス・ドキュメントが表示されます。

図 9-148 レスポンス・ドキュメント



## 9.7 Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成)

この項では、J2CA 構成の Oracle Service Bus を使用して HTTP プロキシ・サービスを構成する方法について説明します。

1. Oracle Service Bus を起動して、必要なプロジェクト・フォルダを作成します。

詳細は、[9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」](#)を参照してください。

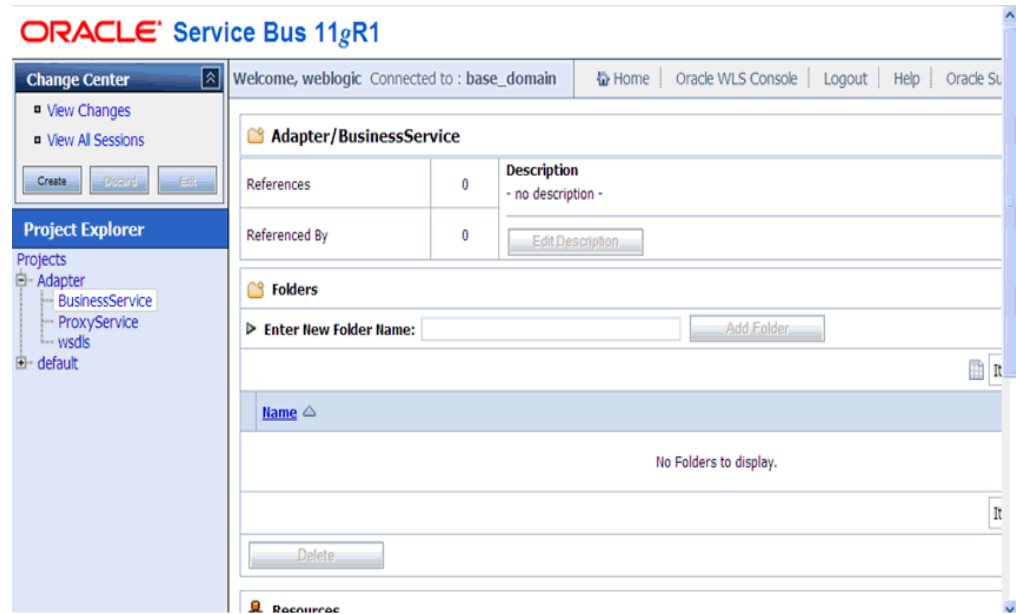
2. WSDL を生成して、アプリケーション・エクスプローラから作成されたプロジェクト・フォルダにパブリッシュし、パブリッシュされた WSDL を使用してビジネス・サービスを作成します。

詳細は、[9.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ」](#)を参照してください。



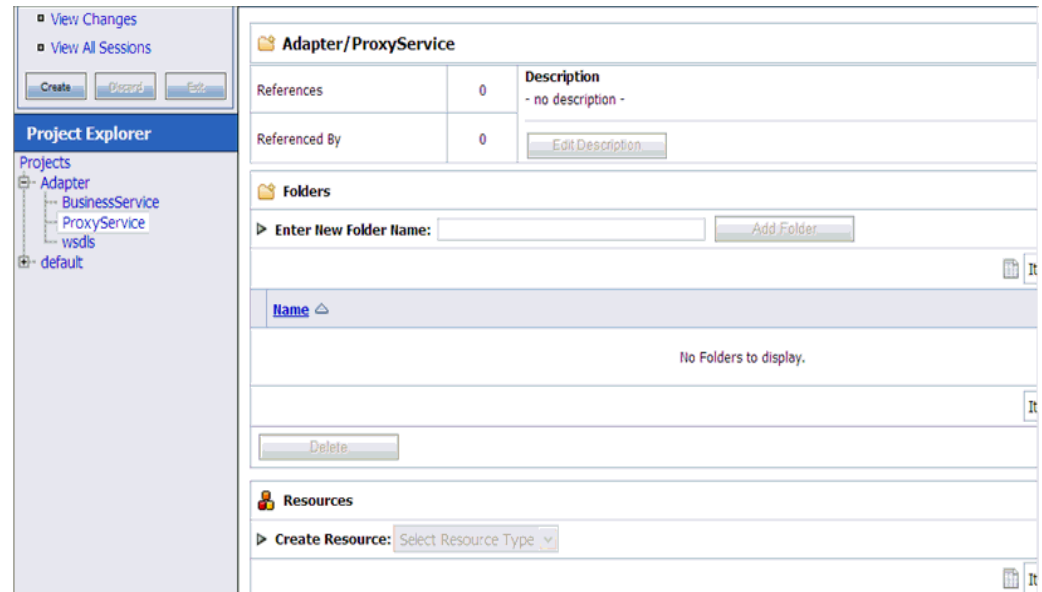
3. 図 9-149 に示すように、Service Bus コンソール・ページを開きます。

図 9-149 Service Bus コンソール・ページ



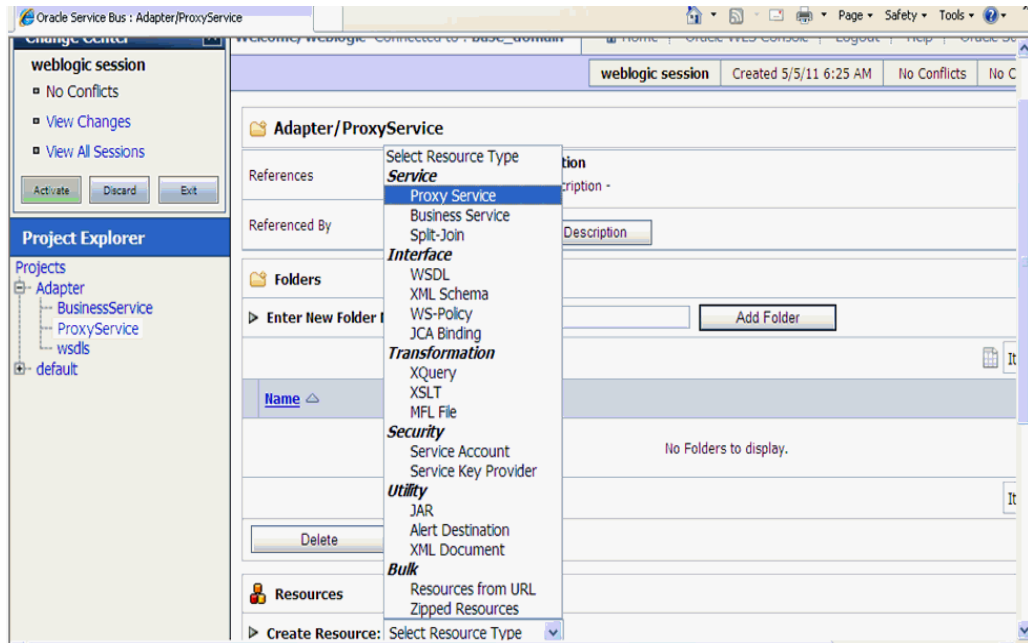
4. 図 9-150 に示すように、プロジェクト・エクスプローラで、「ProxyService」プロジェクト・フォルダを選択して「作成」をクリックします。

図 9-150 プロジェクト・エクスプローラ



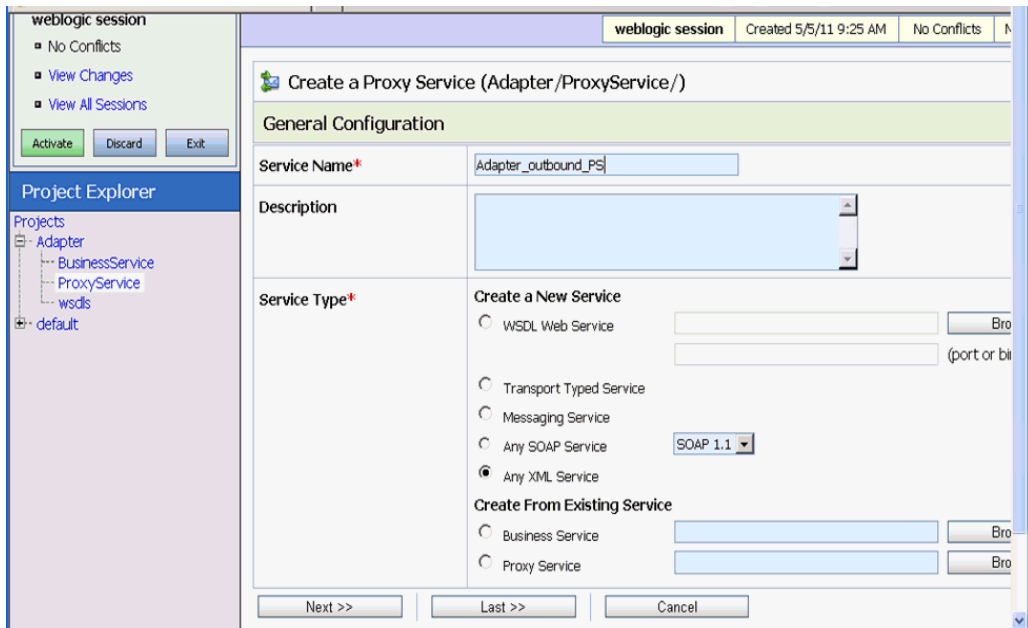
5. 図 9-151 に示すように、右ペインで、「リソースの作成」リストから「プロキシ・サービス」を選択します。

図 9-151 プロキシ・サービス



6. 図 9-152 に示すように、「サービス名」フィールドに適切な名前を入力します。

図 9-152 サービス名



7. 図 9-153 に示すように、「サービス・タイプ」セクションの「既存のサービスから作成」の下で、「ビジネス・サービス」ラジオ・ボタンを選択し、「参照」をクリックします。

図 9-153 一般構成

The screenshot shows the 'Create a Proxy Service (Adapter/ProxyService/)' dialog box. The 'General Configuration' section includes the following fields:

- Service Name\***: Adapter\_outbound\_PS
- Description**: (Empty text area)
- Service Type\***:
  - Create a New Service**:
    - WSDL Web Service
    - Transport Typed Service
    - Messaging Service
    - Any SOAP Service (SOAP 1.1)
    - Any XML Service
  - Create From Existing Service**:
    - Business Service
    - Proxy Service

Buttons at the bottom include 'Next >>', 'Last >>', and 'Cancel'.

8. 図 9-154 に示すように、既存のビジネス・サービスを選択し、「発行」をクリックします。

図 9-154 ビジネス・サービス

The screenshot shows the 'Select Business Service' dialog box. It features a search bar and a table of available services. The 'Adapter\_outbound\_BS' service is selected.

Name	Path	Resource Type
<input checked="" type="radio"/> Adapter_outbound_BS	Adapter/BusinessService	Business Service
<input type="radio"/> fileout	default/business service	Business Service
<input type="radio"/> isdsrv22_samp_node_call_invoke_bs	default/business service	Business Service
<input type="radio"/> isdsrv22_samp_node_invoke_bs	default/business service	Business Service

Buttons at the bottom include 'Submit' and 'Cancel'.

9. 図 9-155 に示すように、「次」をクリックします。

図 9-155 一般構成

**Create a Proxy Service (Adapter/ProxyService/)**

**General Configuration**

Service Name\*

Description

Service Type\*

**Create a New Service**

WSDL Web Service    
 (port or binding)

Transport Typed Service

Messaging Service

Any SOAP Service

Any XML Service

**Create From Existing Service**

Business Service

Proxy Service

10. 図 9-156 に示すように、「プロトコル」リストから「http」を選択し、「次」をクリックします。

図 9-156 トランスポート構成

**Create a Proxy Service (Adapter/ProxyService/Adapter\_outbound\_PS)**

**Transport Configuration**

Protocol\*

Endpoint URI\* Format: /someName

Get All Headers

Yes

No

Header

HEADER	ACTION
There are no headers configured.	

11. 図 9-157 に示すように、「次」をクリックします。

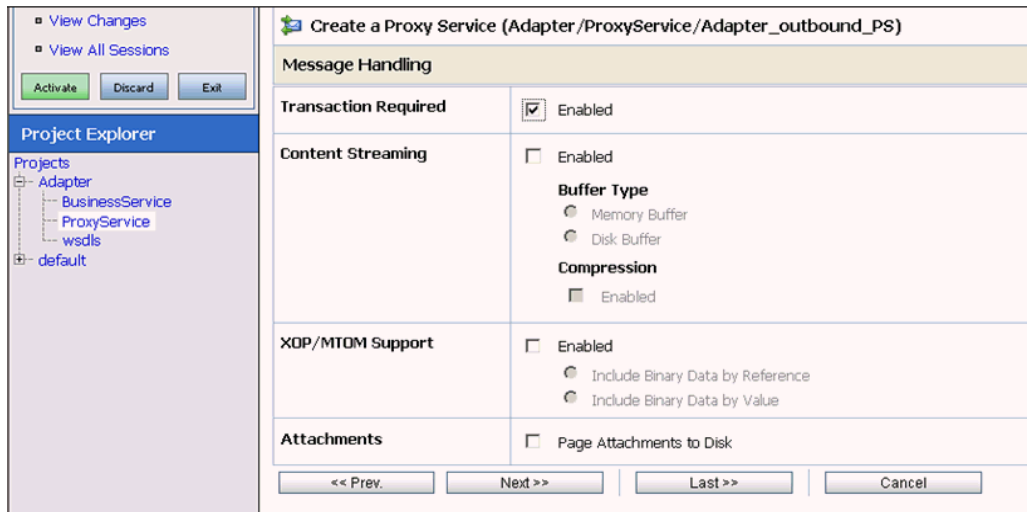
図 9-157 HTTP トランスポート構成

12. 図 9-158 に示すように、「次」をクリックします。

図 9-158 操作選択構成

13. 図 9-159 に示すように、「トランザクションが必要」チェック・ボックスを有効にし、「次」をクリックします。

図 9-159 メッセージ処理



14. 図 9-160 に示すように、「保存」をクリックします。

図 9-160 保存

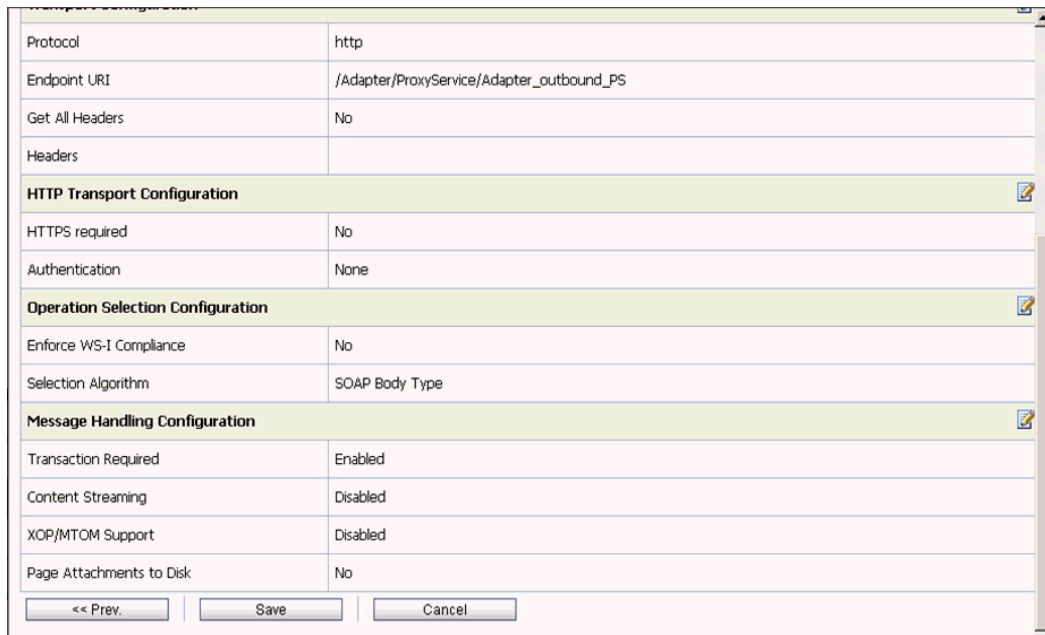
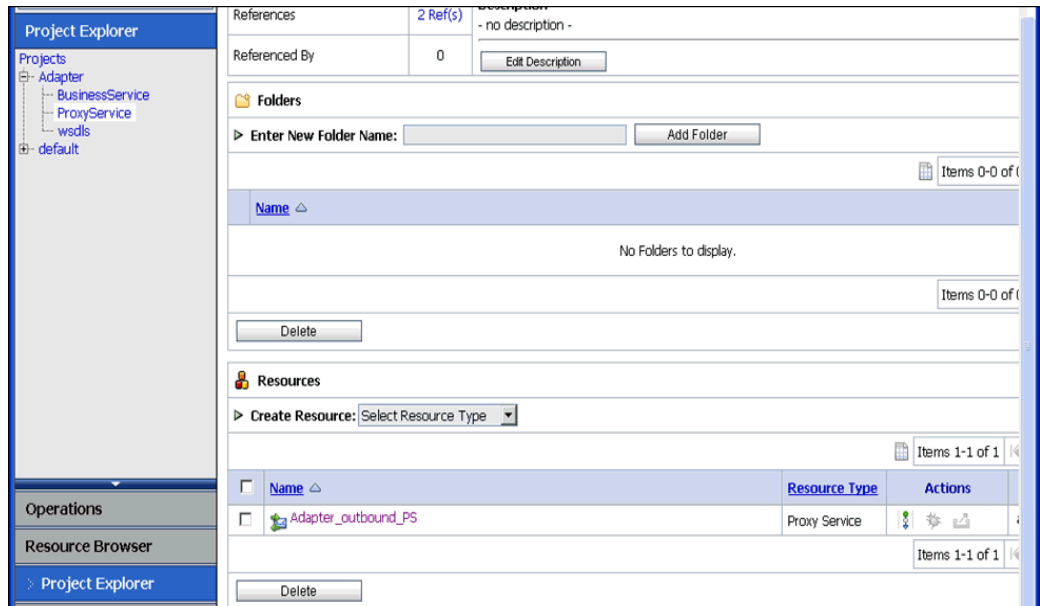


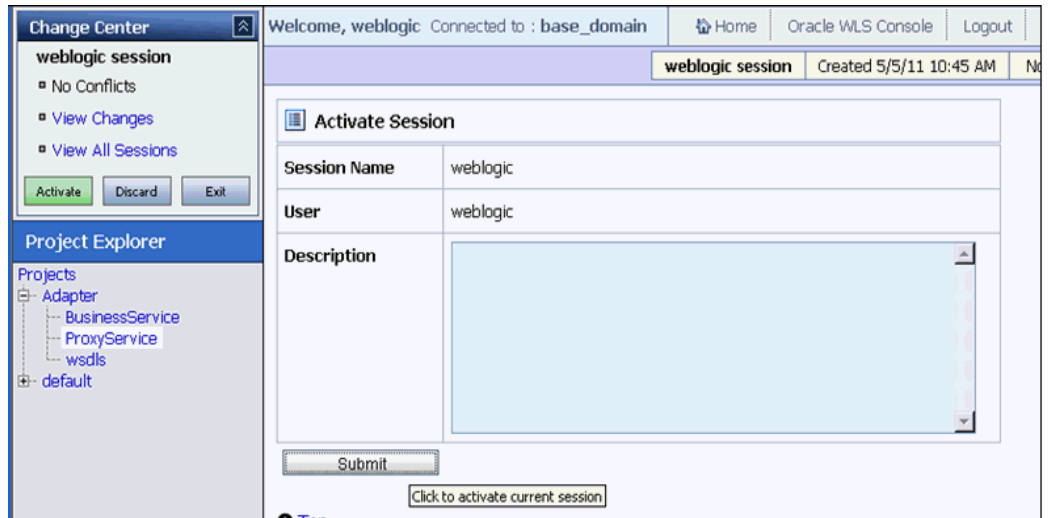
図 9-161 に示すように、作成されたプロキシ・サービスが保存されます。

図 9-161 プロキシ・サービス



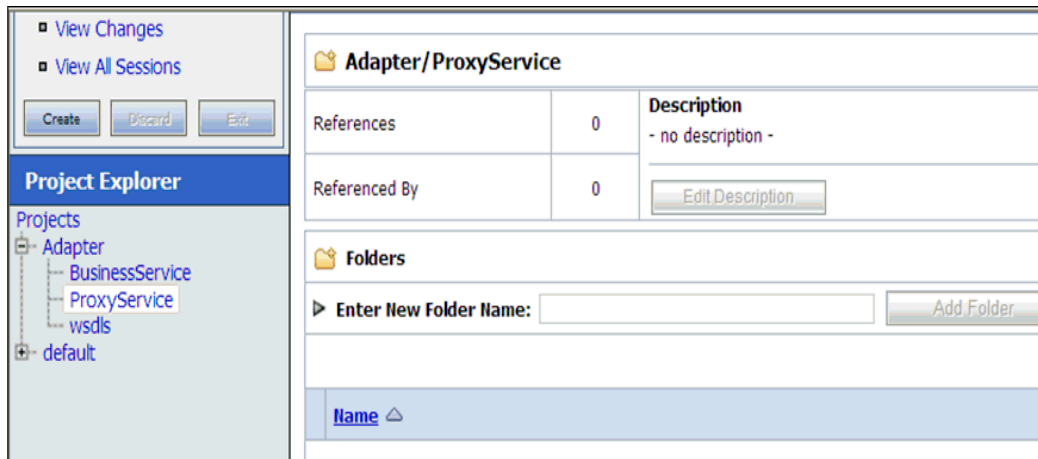
15. 図 9-162 に示すように、左ペインで「アクティブ化」をクリックしてから、右ペインで「発行」をクリックします。

図 9-162 セッションのアクティブ化



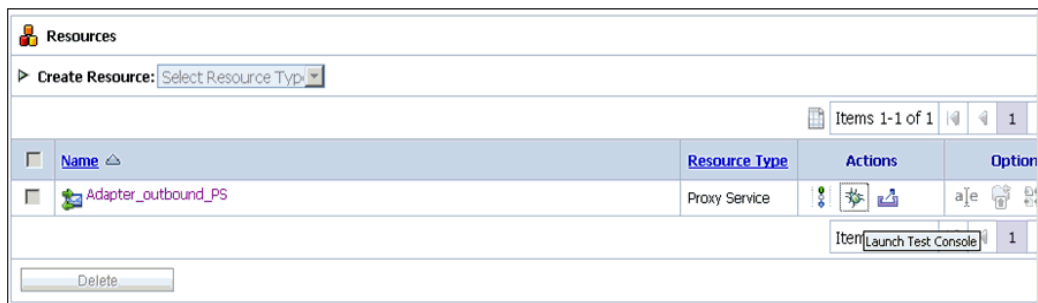
16. 図 9-163 に示すように、左ペインの「プロジェクト」フォルダで「ProxyService」をクリックします。

図 9-163 ProxyService



17. 図 9-164 に示すように、作成されたプロキシ・サービスの「テスト・コンソールの起動」アイコンをクリックします。

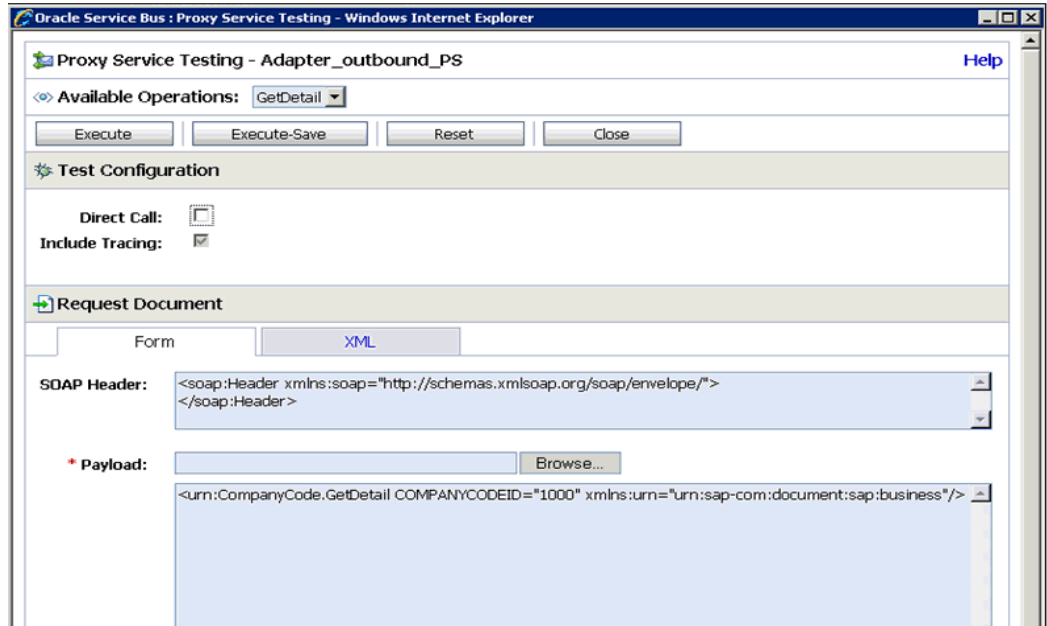
図 9-164 テスト・コンソールの起動





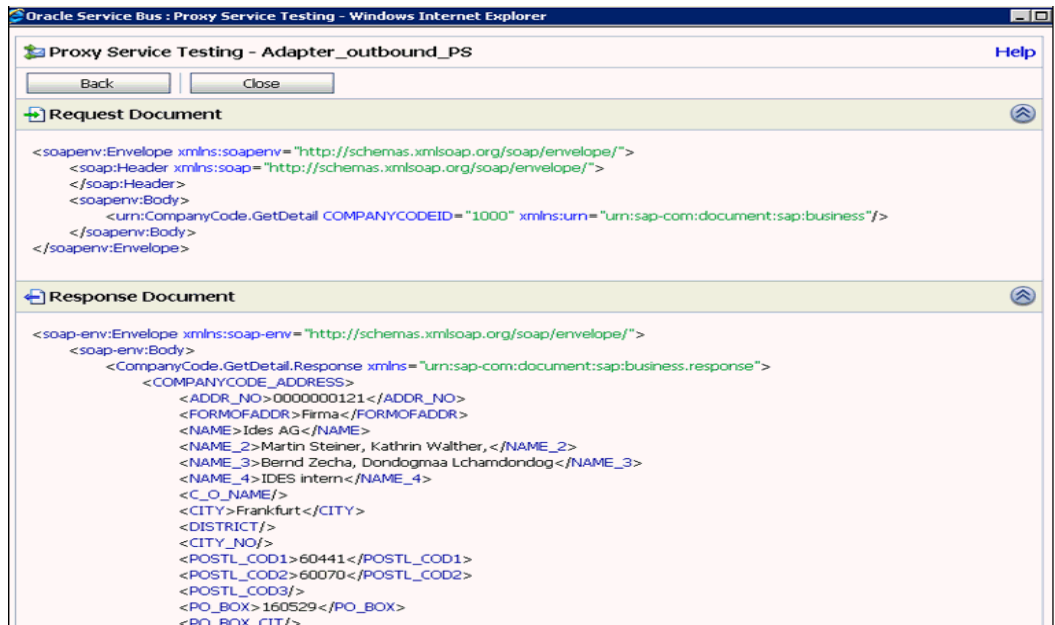
18. 図 9-165 に示すように、「直接呼出し」チェック・ボックスの選択を解除して「ペイロード」に入力値を指定し、「実行」をクリックします。

図 9-165 プロキシ・サービス・テスト



19. 図 9-166 に示すように、「レスポンス・ドキュメント」を確認します。

図 9-166 レスポンス・ドキュメント





---

# JDeveloper を使用した Oracle Service Bus のアウトバウンドおよびインバウンド・プロセス の構成

Oracle Application Adapter for SAP R/3 は Oracle JDeveloper とシームレスに統合され、これにより Web サービス統合が簡略化されます。

この章は次の項で構成されています。

- 10.1 項「JDeveloper を使用した OSB アウトバウンド・プロセスの構成 (J2CA 構成)」
- 10.2 項「JDeveloper を使用した OSB インバウンド・プロセスの構成 (J2CA 構成)」
- 10.3 項「JDeveloper を使用した OSB アウトバウンド・プロセスの構成 (BSE 構成)」
- 10.4 項「JDeveloper を使用した Oracle Service Bus の同期 SAP R/3 イベントの処理 (J2CA 構成)」
- 10.5 項「JDeveloper を使用した JMS インバウンド・プロセスの構成 (J2CA 構成)」
- 10.6 項「JDeveloper を使用した JMS アウトバウンド・プロセスの構成 (J2CA 構成)」
- 10.7 項「JDeveloper を使用した HTTP アウトバウンド・プロセスの構成 (J2CA 構成)」

## 10.1 JDeveloper を使用した OSB アウトバウンド・プロセスの 構成 (J2CA 構成)

この項では、Oracle JDeveloper を使用して SAP R/3 システムへの OSB アウトバウンド・プロセスを構成する方法 (J2CA 構成用) について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\J2CA\SAP_Sample_
J2CA_OSB_Outbound_Project
```

この項では、次の項目について説明します。

- 10.1.1 項「OSB の Service Bus アプリケーションの作成」
- 10.1.2 項「OSB アウトバウンド・プロセスの定義」
- 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」

### 前提条件

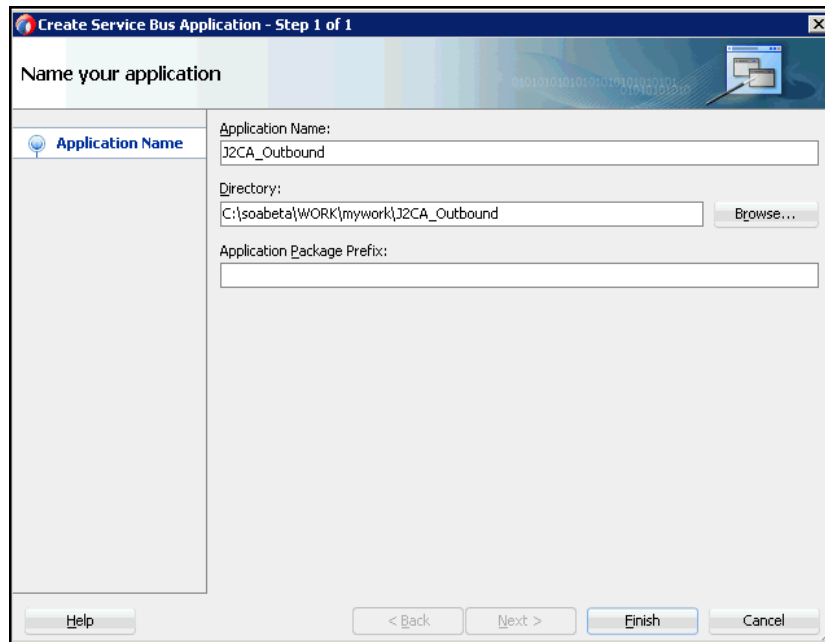
OSB アウトバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、4-15 ページの 4.7、「WSDL の生成 (J2CA 構成のみ)」を参照してください。

## 10.1.1 OSB の Service Bus アプリケーションの作成

OSB の Service Bus アプリケーションを作成するには、JDeveloper で次のようにします。

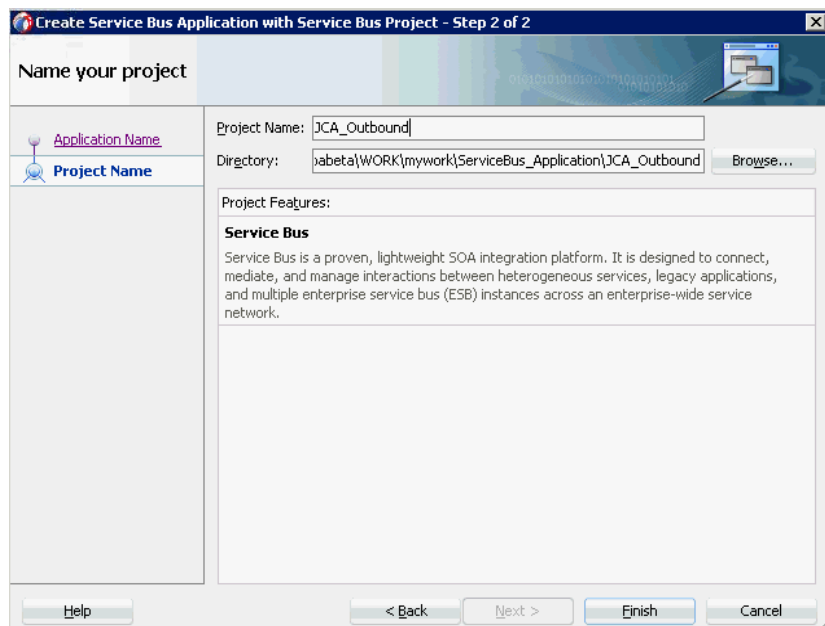
1. 新しい OSB アプリケーションを作成します。
2. [図 10-1](#) に示すように、OSB アプリケーションの名前 (J2CA\_Outbound など) を入力し、「終了」をクリックします。

図 10-1 「アプリケーションの名前付け」 ペイン



3. [図 10-2](#) に示すように、プロジェクト名 (JCA\_Outbound など) を入力し、「終了」をクリックします。

図 10-2 「プロジェクトの名前付け」 ペイン



## 10.1.2 OSB アウトバウンド・プロセスの定義

この項では、OSB アウトバウンド・プロセスを定義する方法について説明します。ここでは、次の項目について説明します。

- 10.1.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」
- 10.1.2.2 項「ファイル・トランスポート・タイプ・ビジネス・サービスの構成」
- 10.1.2.3 項「パイプラインを含むプロキシ・サービスの作成」
- 10.1.2.4 項「ルーティング・ルールの構成」

### 10.1.2.1 サード・パーティ・アダプタ・サービス・コンポーネントの構成

ビジネス・サービスとともにサード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

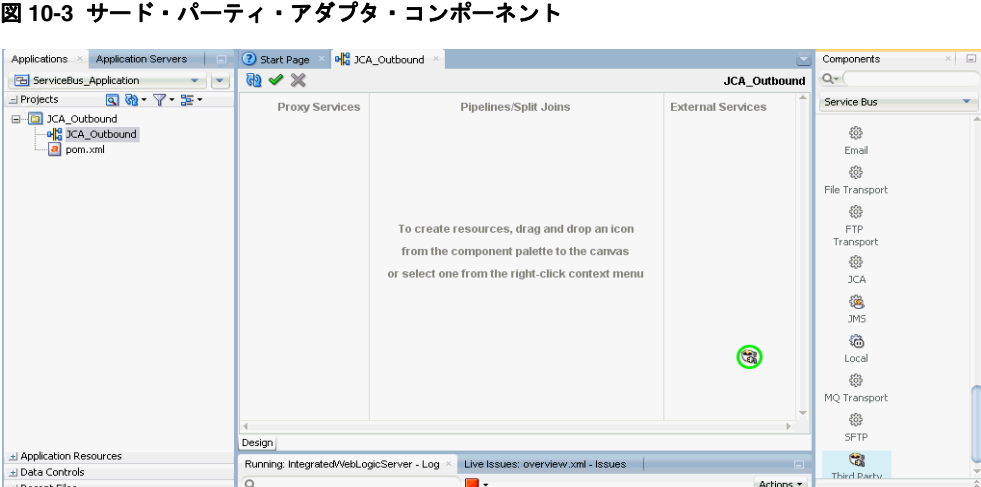
1.  10-3 に示すように、「サード・パーティ」アダプタ・コンポーネントを「コンポーネント」の「Service Bus」ペインから「外部サービス」ペインにドラッグ・アンド・ドロップします。

図 10-3 サード・パーティ・アダプタ・コンポーネント

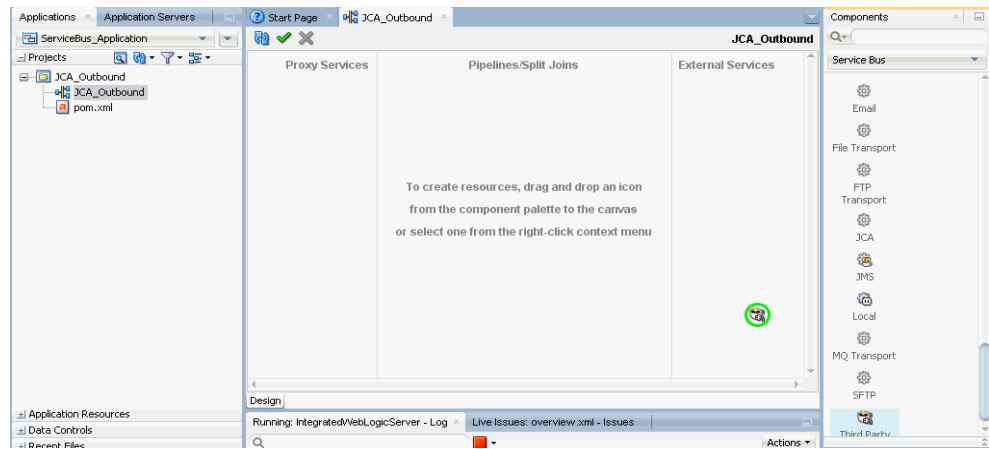
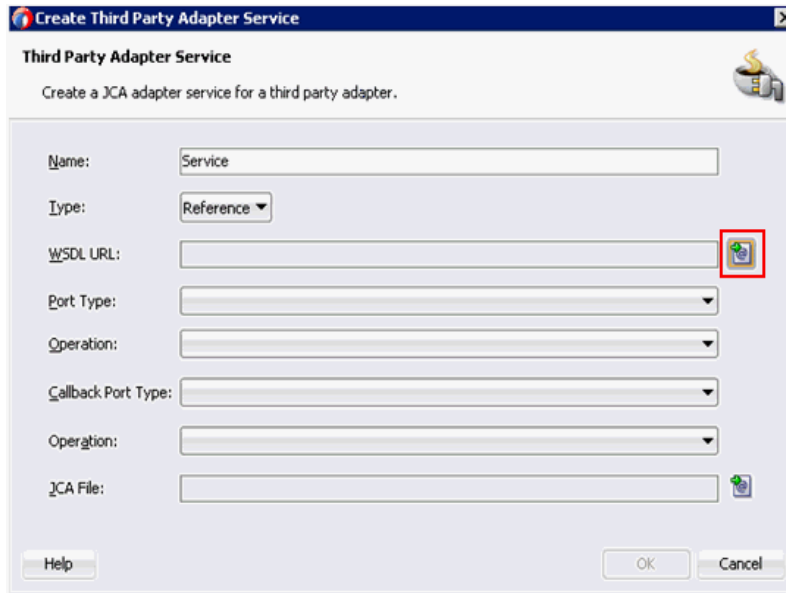


図 10-4 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

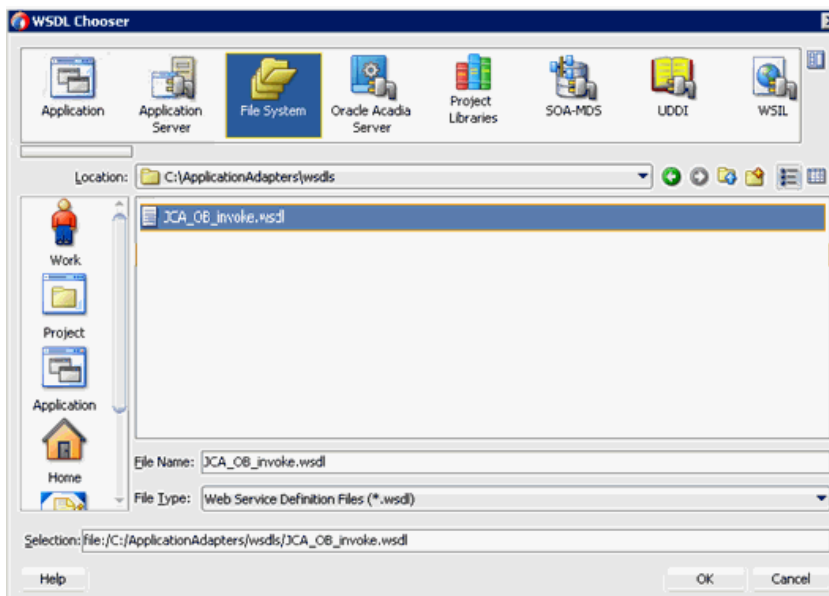
図 10-4 「サード・パーティ・アダプタ・サービスの作成」 ペイン



2. ビジネス・サービス名として使用するサード・パーティ・アダプタ・サービスの適切な名前を入力します。
3. 「タイプ」ドロップダウン・リストで「参照」が選択されていることを確認します (デフォルト)。
4. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。

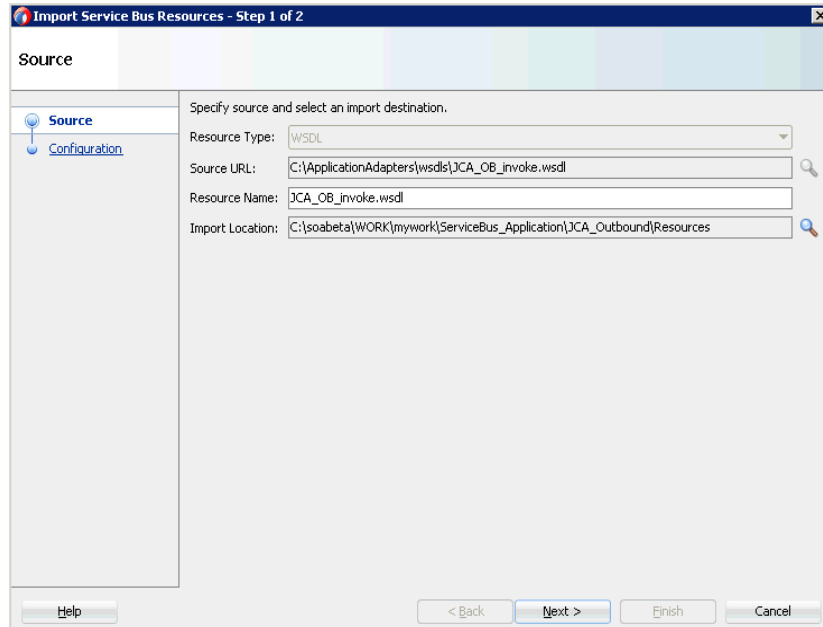
図 10-5 に示すように、「WSDL の選択」ダイアログが表示されます。

図 10-5 「WSDL の選択」ダイアログ



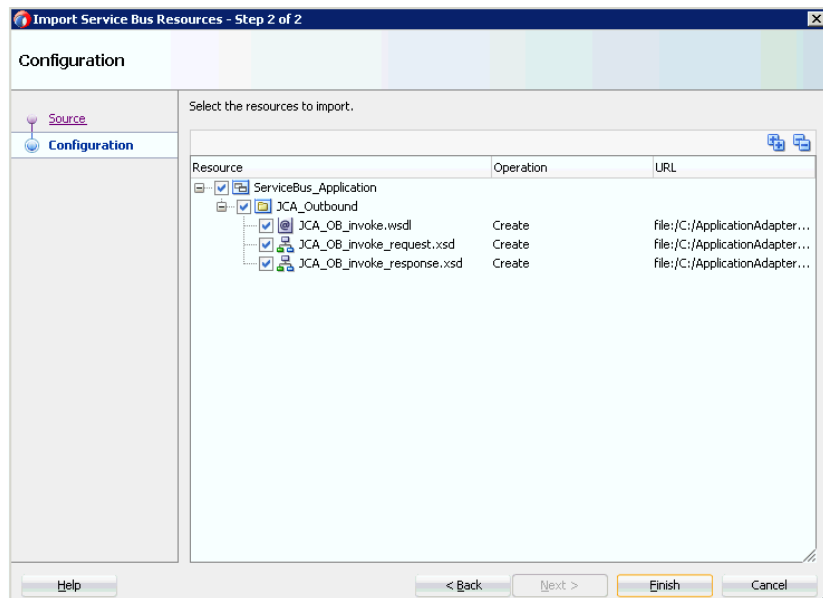
5. 「ファイルシステム」タブを選択し、WSDLディレクトリでアウトバウンド WSDL ファイルを参照して選択します。
6. 「OK」をクリックします。  
「Service Bus リソースのインポート」ダイアログが表示されます。
7. 図 10-6 に示すように、「次」をクリックします。

図 10-6 「ソース」 ペイン



8. 図 10-7 に示すように、「構成」ペインで「終了」をクリックします。

図 10-7 「構成」 ペイン



「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

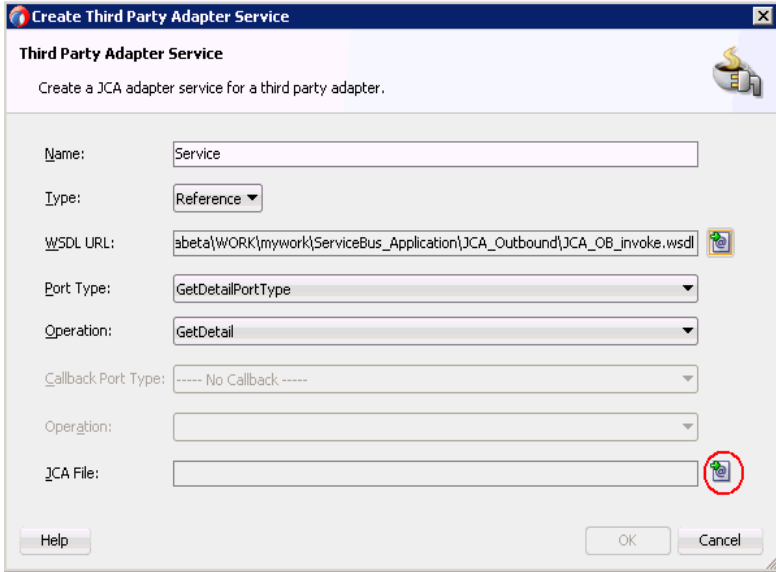
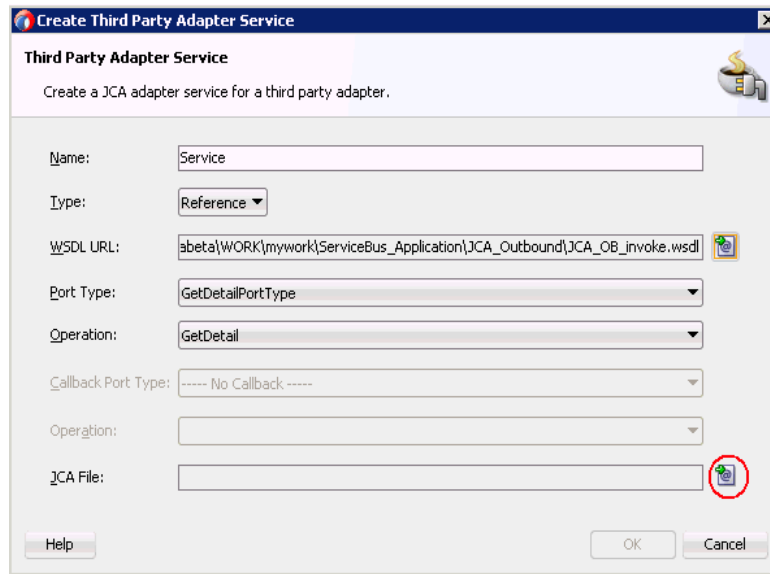
9.  10-8 に示すように、「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。

図 10-8 「JCA ファイルを検索します。」アイコン



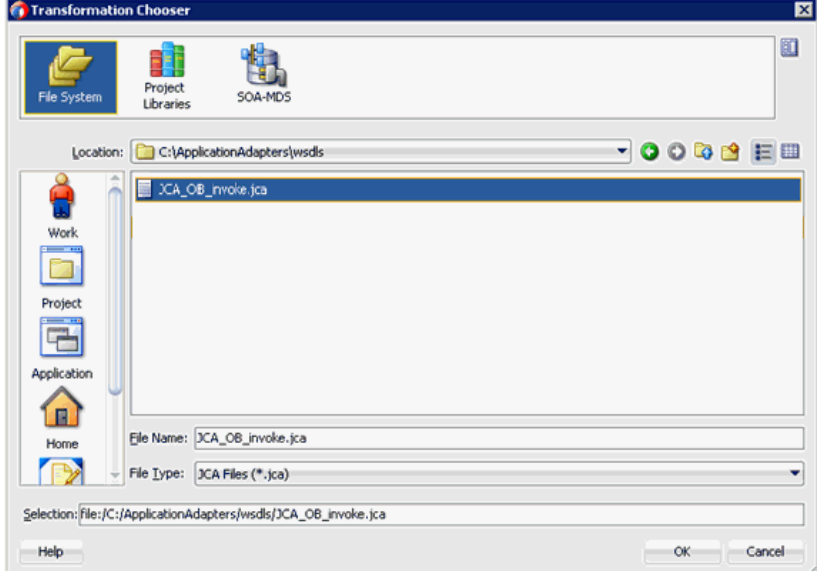
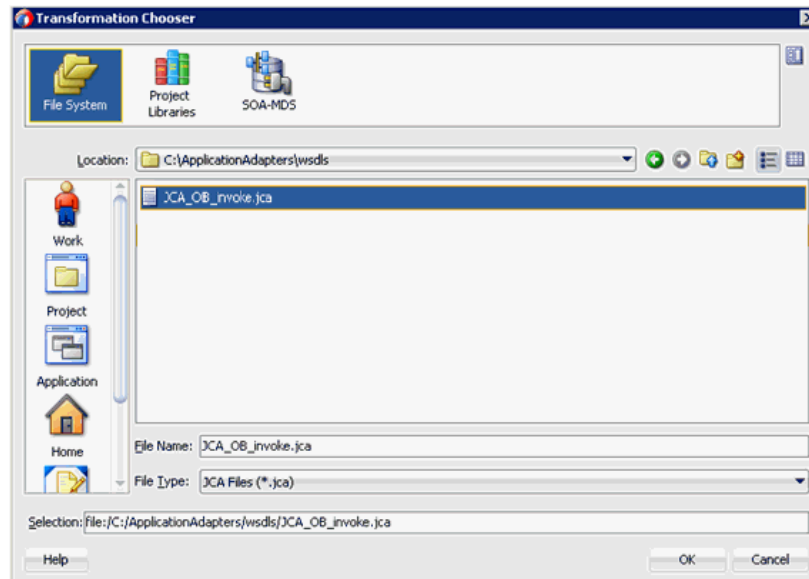
10.  10-9 に示すように、トランスフォーメーション・チューザ・ダイアログが表示されます。

図 10-9 トランスフォーメーション・チューザ・ダイアログ

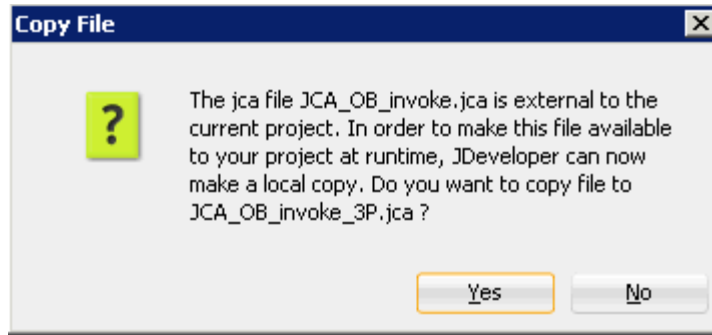


10. WSDL ディレクトリから JCA プロパティ・ファイルを選択します。



11. 「OK」をクリックします。図 10-10 に示すように、「ファイルのコピー」メッセージが表示されます。

図 10-10 「ファイルのコピー」メッセージ

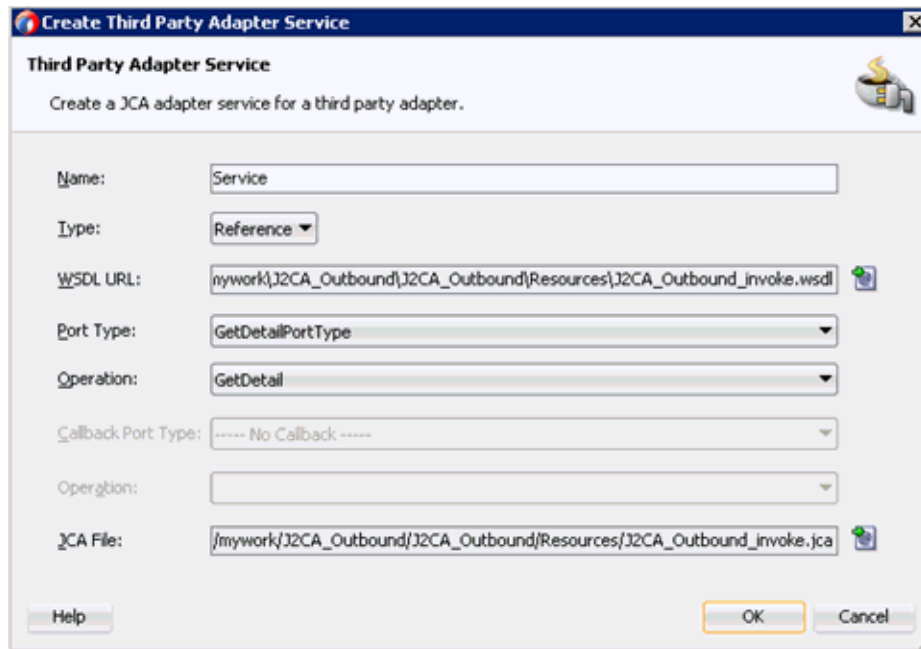


12. 「はい」をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

図 10-11 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

図 10-11 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



13. 「OK」をクリックします。

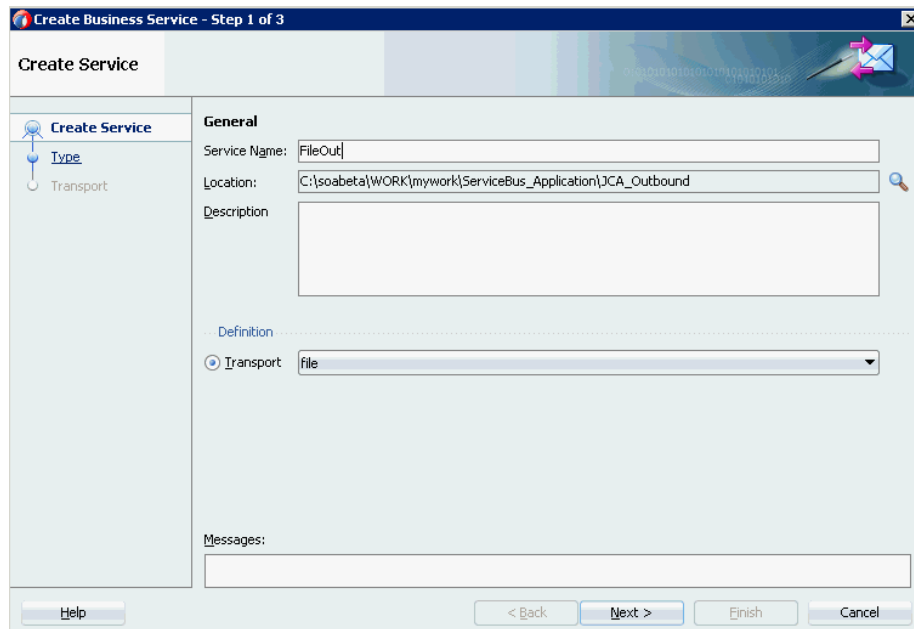
「外部サービス」ペインにビジネス・サービス・コンポーネントが作成されます。

### 10.1.2.2 ファイル・トランスポート・タイプ・ビジネス・サービスの構成

ファイル・トランスポート・ビジネス・サービスを作成するには、次のようにします。

1. 「ファイル・トランスポート」コンポーネントを「詳細」ペインから「外部サービス」ペインにドラッグ・アンド・ドロップします。  
「ビジネス・サービスの作成」ダイアログが表示されます。
2. 図 10-12 に示すように、「サービス名」フィールドにビジネス・サービスに使用する任意の名前 (FileOut など) を入力し、「次」をクリックします。

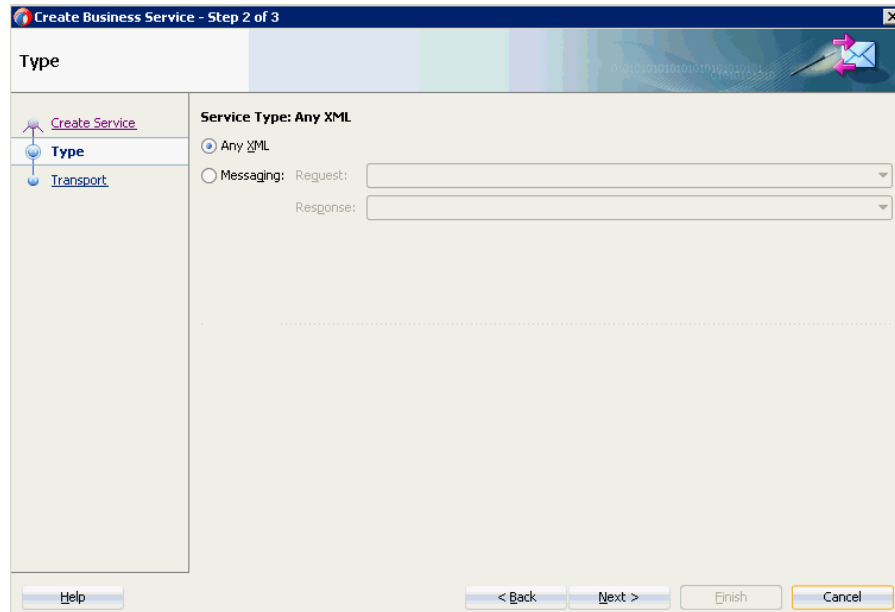
図 10-12 「サービスの作成」ペイン



「タイプ」ペインが表示されます。デフォルトでは、「任意の XML」オプションが選択されています。

3. 図 10-13 に示すように、「次」をクリックします。

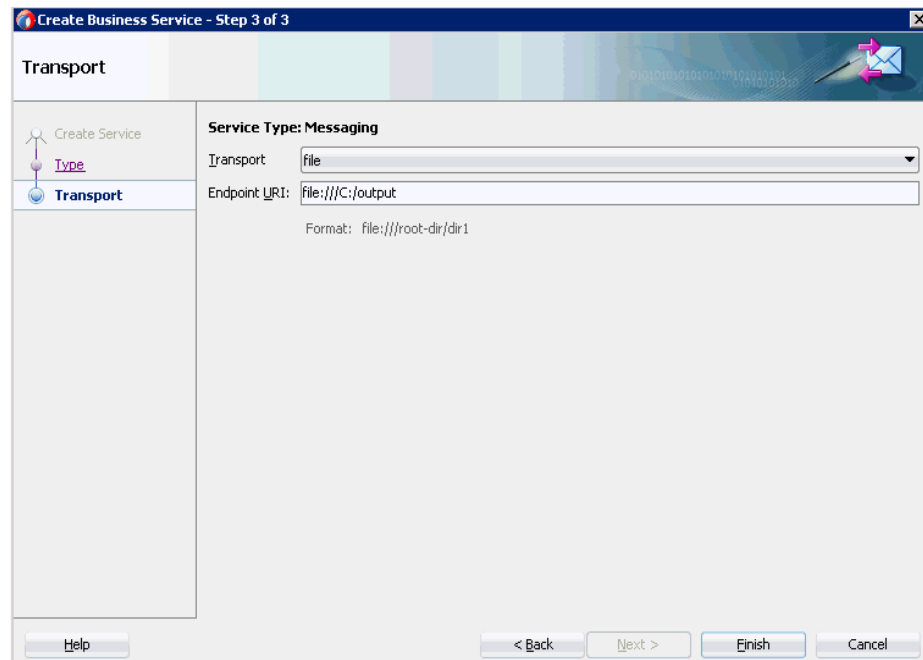
図 10-13 「タイプ」 ペイン



「トランスポート」ペインが表示されます。

4. 図 10-14 に示すように、「エンドポイント URI」フィールドに出力場所 (c:/output など) を指定し、「終了」をクリックします。

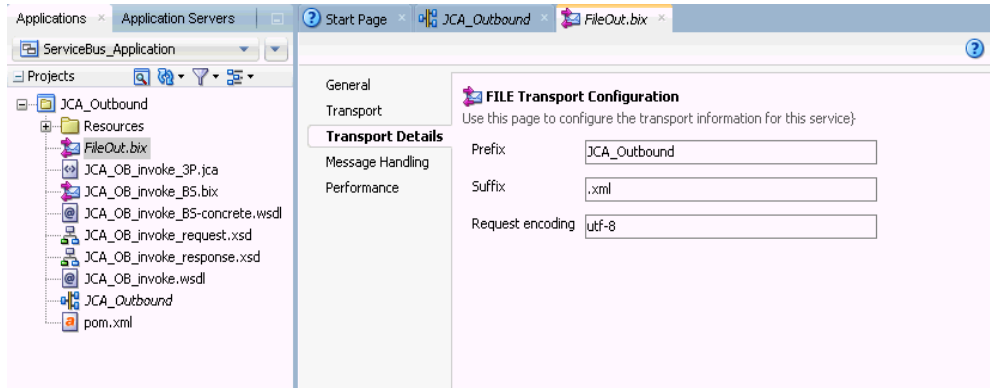
図 10-14 「トランスポート」 ペイン



ファイル・トランスポート・ビジネス・サービス Fileout が作成されて表示されます。

5. 図 10-15 に示すように、作成されたビジネス・サービス **Fileout** をダブルクリックし、「トランスポートの詳細」タブで「接頭辞」と「接尾辞」の各フィールドに値を指定します。

図 10-15 トランスポートの詳細

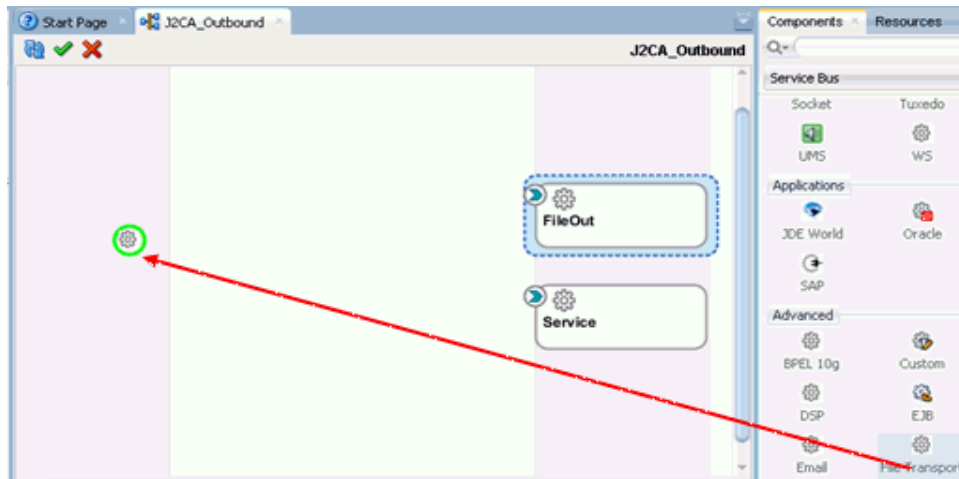


### 10.1.2.3 パイプラインを含むプロキシ・サービスの作成

パイプラインを含むプロキシ・サービスを作成するには、次のようにします。

1. 図 10-16 に示すように、「ファイル・トランスポート」コンポーネントを「コンポーネント」の「詳細」ペインから「プロキシ・サービス」ペインにドラッグ・アンド・ドロップします。

図 10-16 ファイル・トランスポート・コンポーネント

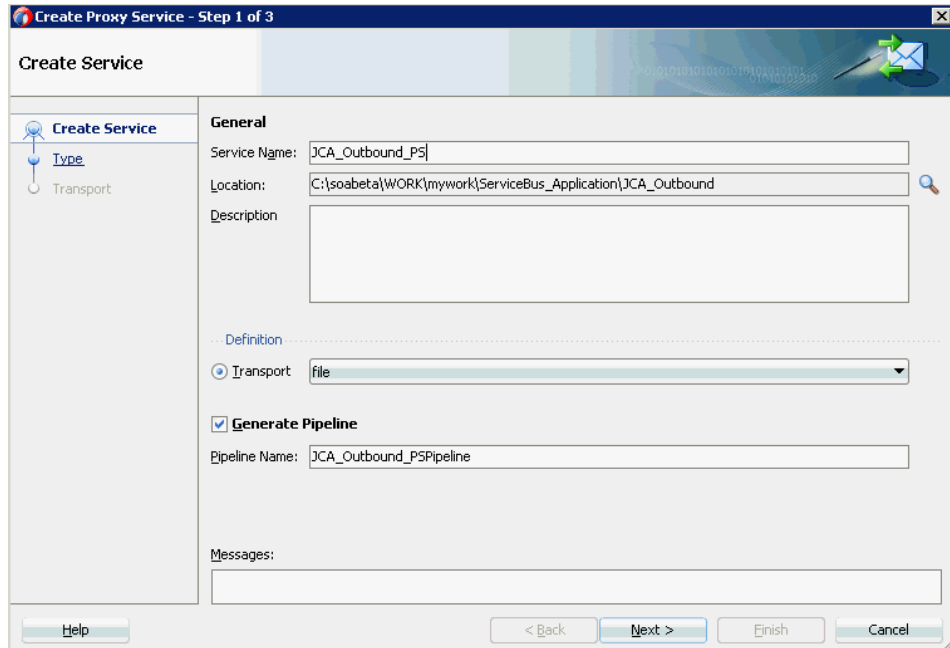


「プロキシ・サービスの作成」ペインが表示されます。

2. 「サービス名」フィールドに、プロキシ・サービスに使用する任意の名前 (JCA\_Outbound\_PS など) を入力します。デフォルトでは、「パイプラインの生成」が選択されています。

3. 図 10-17 に示すように、「次」をクリックします。

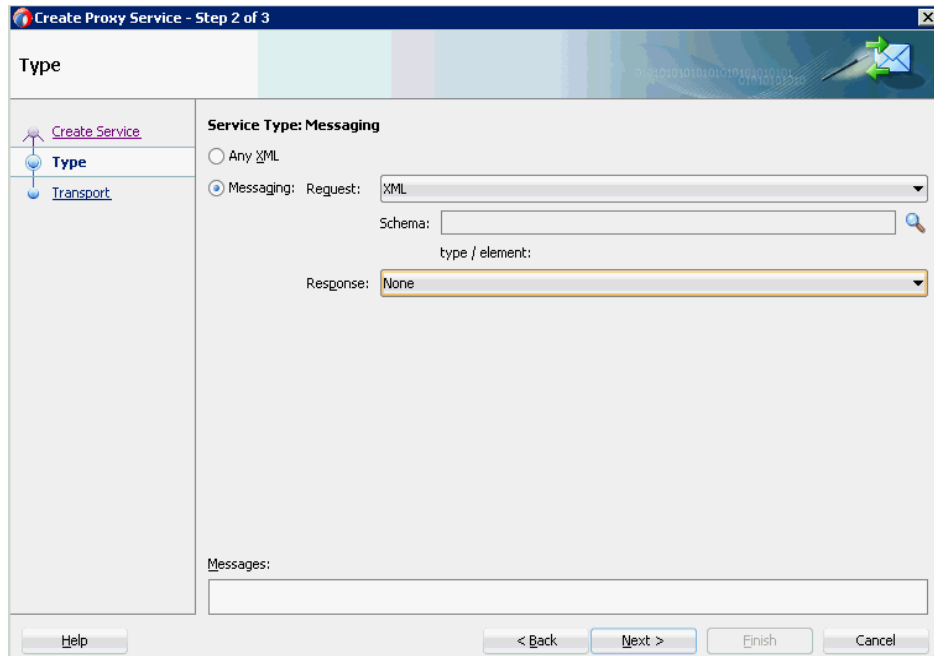
図 10-17 「サービスの作成」 ペイン



「タイプ」 ペインが表示されます。

4. 図 10-18 に示すように、「メッセージング」 オプションを選択し、「リクエスト」を「XML」に、「レスポンス」を「なし」に設定してから、「次」をクリックします。

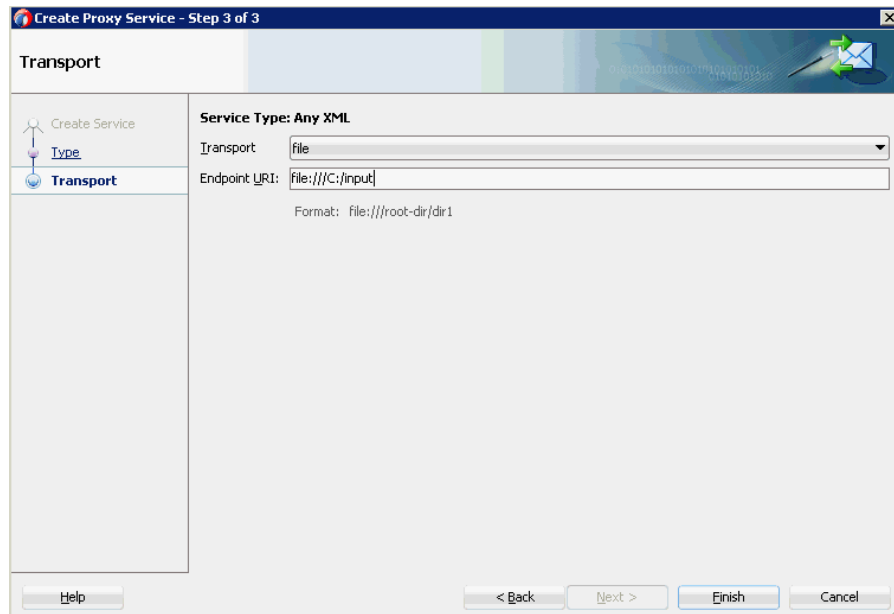
図 10-18 「タイプ」 ペイン



「トランスポート」 ウィンドウが表示されます。

5. 図 10-19 に示すように、「エンドポイント URI」フィールドに入力場所 (c:/input など) を指定し、「終了」をクリックします。

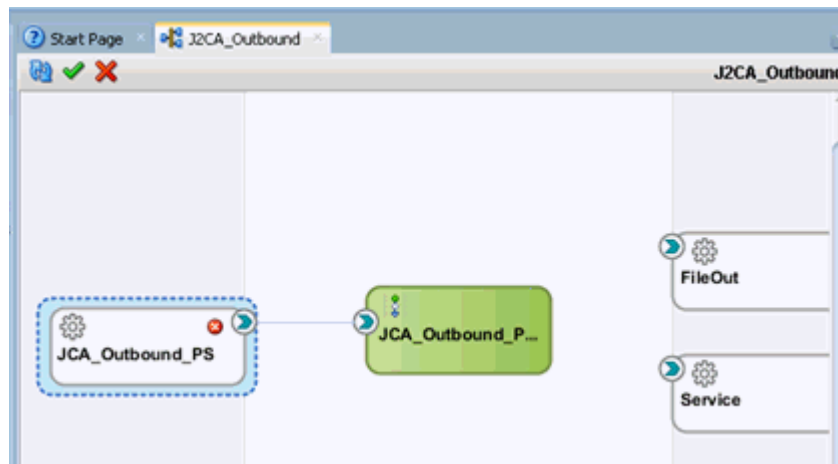
図 10-19 「トランスポート」ウィンドウ



プロキシ・サービスが、パイプラインとともに作成されて表示されます。

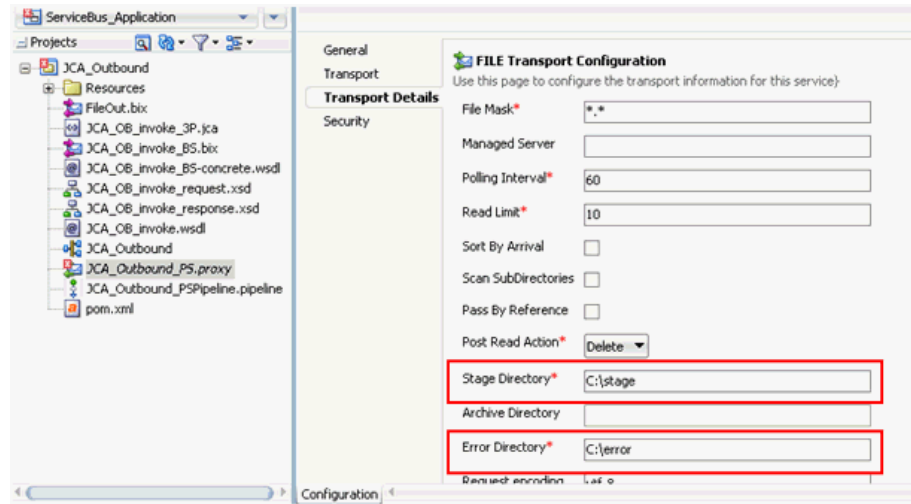
6. 図 10-20 に示すように、作成されたプロキシ・サービス (JCA\_Outbound\_PS など) をダブルクリックします。

図 10-20 プロキシ・サービスの編集



7. 図 10-21 に示すように、表示されるプロキシ・サービスの構成ページで「トランスポートの詳細」を選択し、「ステージ・ディレクトリ」と「エラー・ディレクトリ」に値を指定します。

図 10-21 「ファイル・トランスポート」の構成



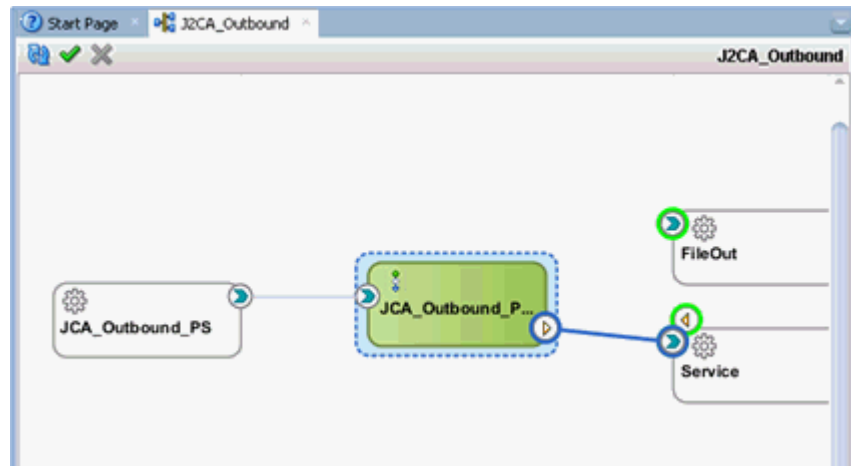
8. プロキシ・サービスの構成ページを保存して閉じます。

#### 10.1.2.4 ルーティング・ルールの構成

ルーティング・ルールを構成するには、次のようにします。

1. 図 10-22 に示すように、パイプラインをビジネス・サービス (Service など) に接続します。

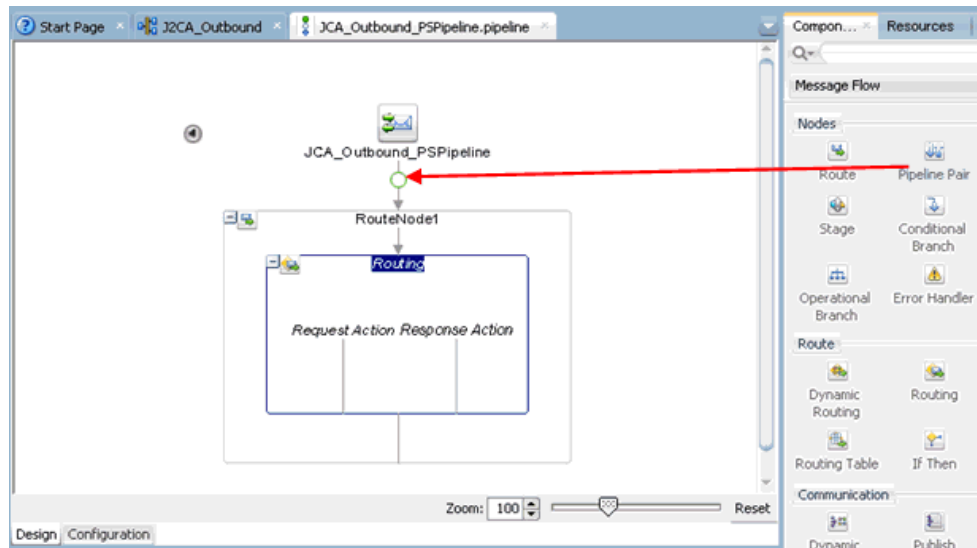
図 10-22 ビジネス・サービス・パイプライン



2. 「パイプライン/分割結合」ペインで、パイプライン (JCA\_Outbound\_PSPipeline など) をダブルクリックします。  
パイプラインの構成ページが表示されます。

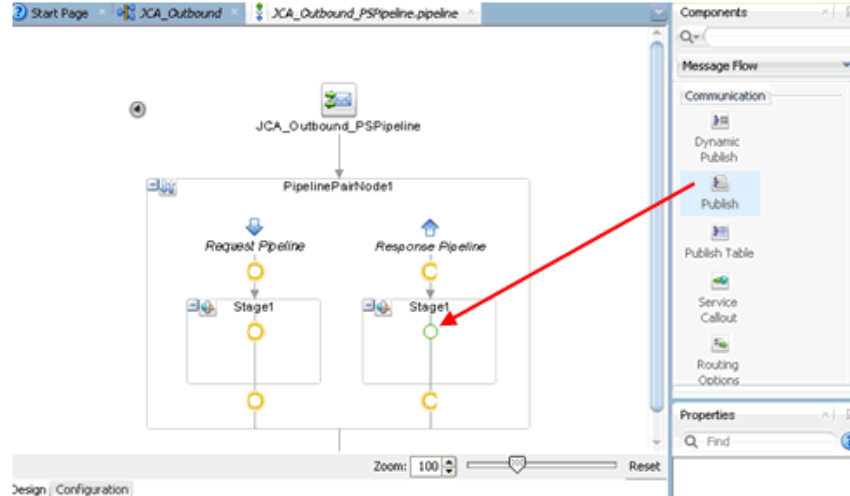
3. 図 10-23 に示すように、「パイプライン・ペア」ノードを「ノード」ペインからパイプライン (JCA\_Outbound\_PSPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 10-23 「パイプライン・ペア」ノード



4. 図 10-24 に示すように、「パブリッシュ」ノードを「通信」ペインから「レスポンス・パイプライン」の「ステージ 1」の下領域にドラッグ・アンド・ドロップします。

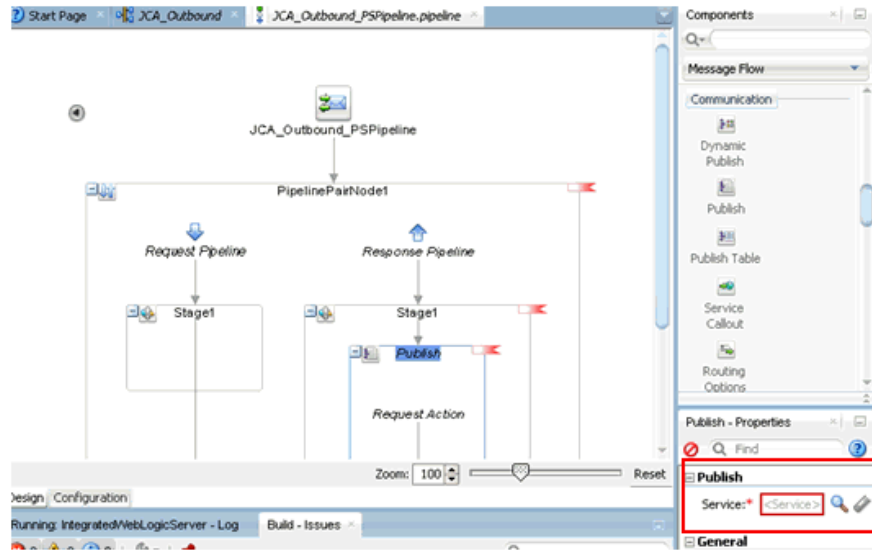
図 10-24 「パブリッシュ」ノード





5. 図 10-25 に示すように、「パブリッシュ - プロパティ」の右ペインで、「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 10-25 「参照」アイコン



6. 図 10-26 に示すように、表示される「リソース・チューザ」ウィンドウで、「Fileout.bix」ファイル・トランスポート・ビジネス・サービスを選択して「OK」をクリックします。

図 10-26 リソース・チューザ

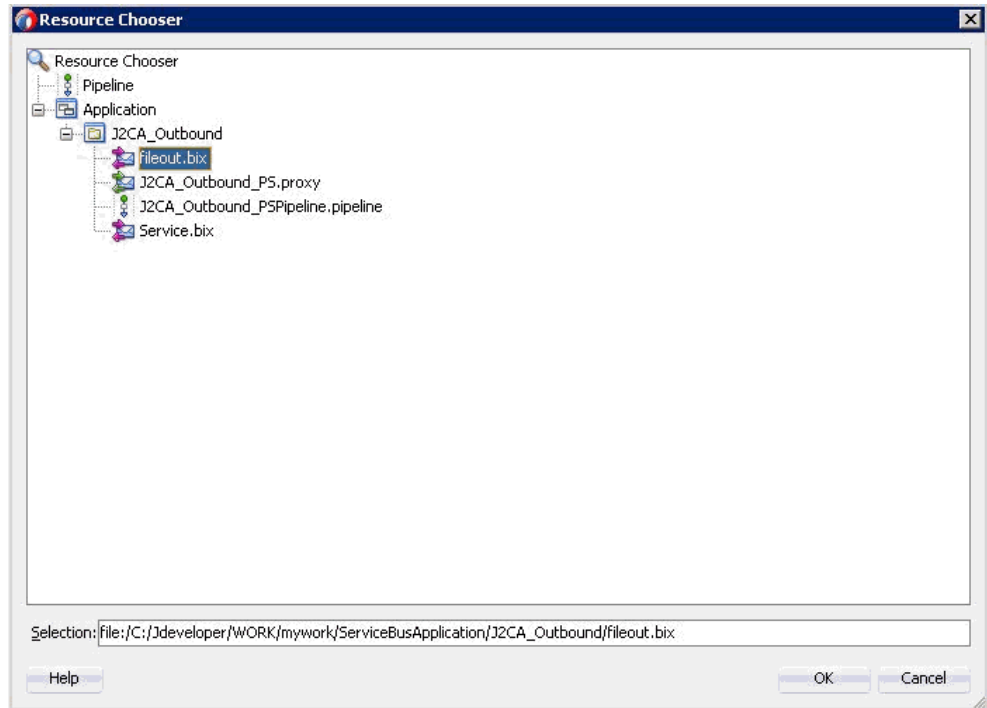
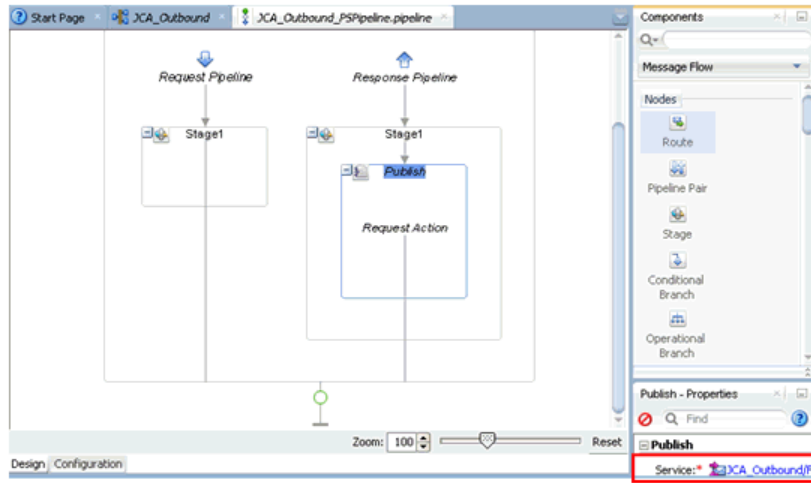


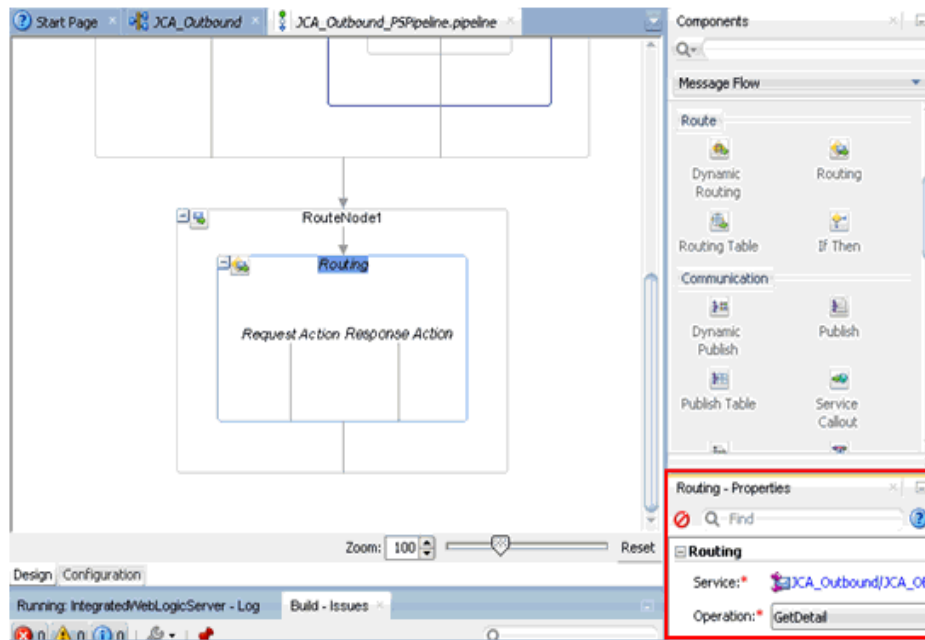
図 10-27 に示すように、選択したサービスが右ペインの「パブリッシュ」ペインに構成されます。

図 10-27 「パブリッシュ」ペイン



7. 図 10-28 に示すように、「ルーティング」をクリックして、サービスが正しく選択されていることを確認します。

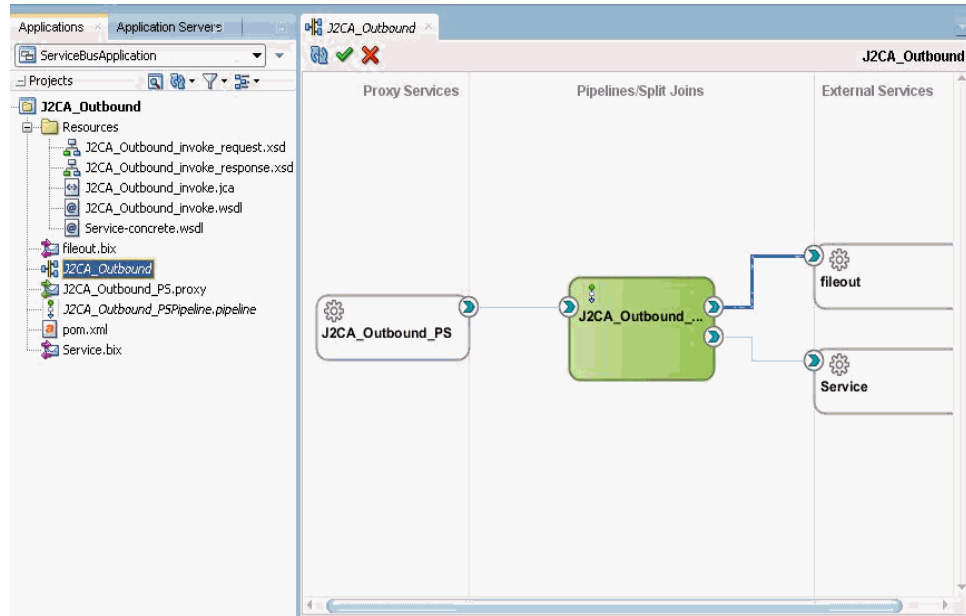
図 10-28 パイプラインの構成



8. パイプラインの構成ページを保存して閉じます。

9. 図 10-29 に示すように、overview.xml ファイル (JCA\_Outbound など) をダブルクリックし、メニュー・バーの「すべて保存」をクリックして OSB プロセスを保存します。

図 10-29 「すべて保存」アイコン

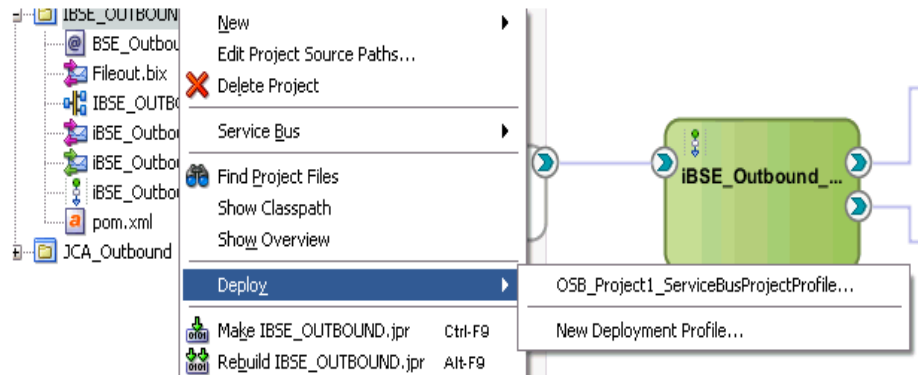


### 10.1.3 OSB アウトバウンド・プロセスのデプロイ

OSB アウトバウンド・プロセスをデプロイするには、次のようにします。

1. 図 10-30 に示すように、OSB プロジェクトを右クリックして「デプロイ」を選択し、「OSB\_Project1\_ServiceBusProjectProfile...」を選択します。

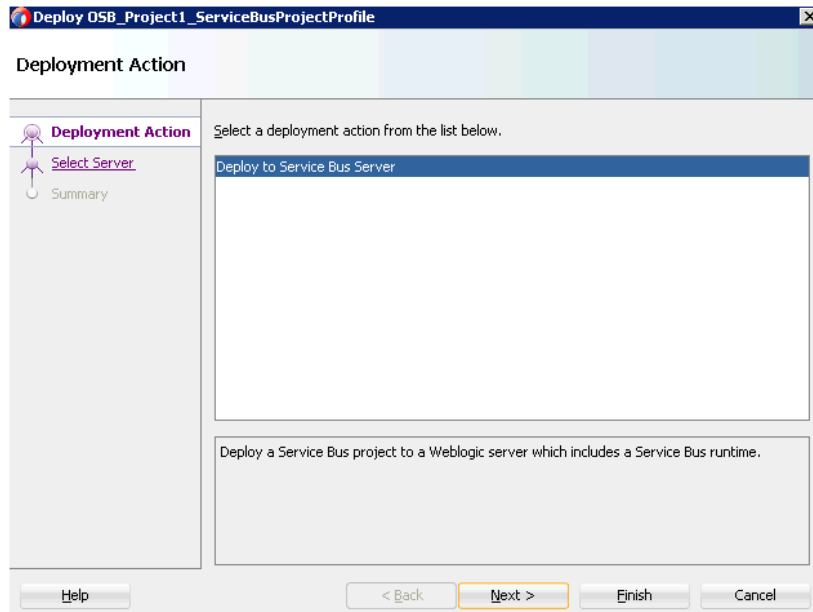
図 10-30 「デプロイ」オプション



「デプロイメント・アクション」ページが表示されます。

2. 図 10-31 に示すように、「次」をクリックします。

図 10-31 「デプロイメント・アクション」 ページ



「サーバーの選択」 ページが表示されます。

3. 図 10-32 に示すように、構成済の使用可能なアプリケーション・サーバーを選択し、「次」をクリックします。

図 10-32 「サーバーの選択」 ページ

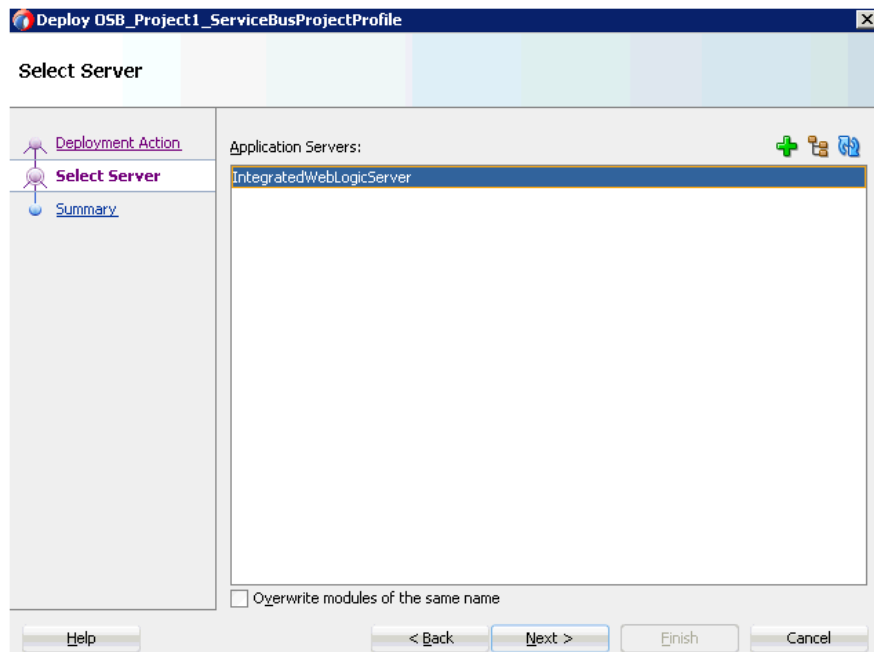
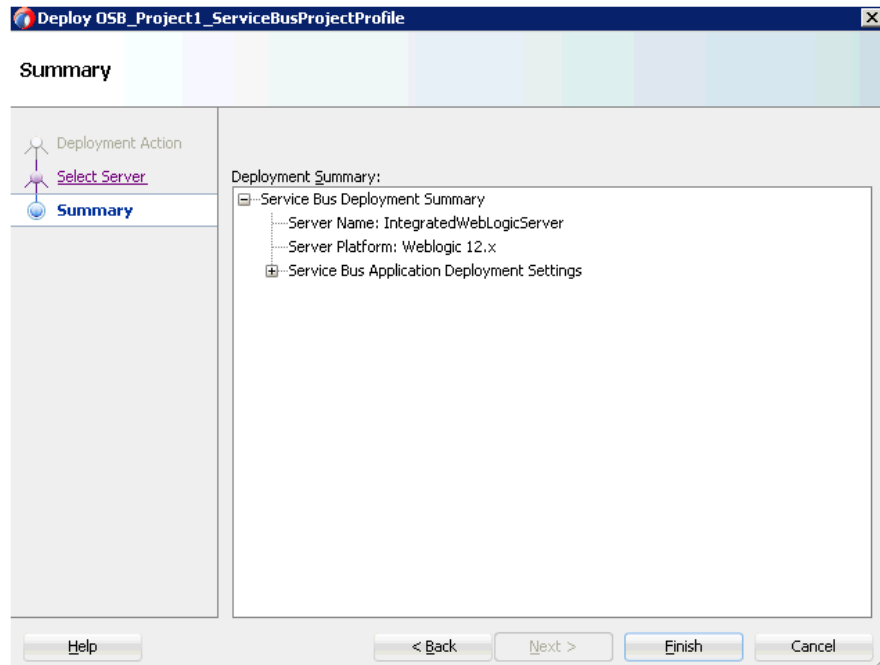


図 10-33 に示すように、「サマリー」ページが表示されます。

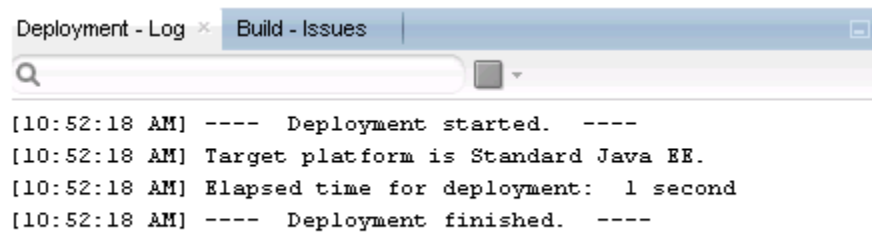
図 10-33 「サマリー」ページ



- プロジェクトに関する使用可能なすべてのデプロイメント情報を確認および検証し、「終了」をクリックします。

図 10-34 に示すように、プロセスが正常にデプロイされます。

図 10-34 成功したデプロイメントのメッセージ



- 構成済の入力フォルダ (C:\input など) に、入力 XML ファイルをコピーして貼り付けます。

出力は、構成済の出力場所 (C:\output など) で受信されます。

## 10.2 JDeveloper を使用した OSB インバウンド・プロセスの構成 (J2CA 構成)

この項では、Oracle JDeveloper を使用して、J2CA 構成用に SAP R/3 システムに対する OSB インバウンド・プロセスを構成する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\J2CA\SAP_Sample_J2CA_OSB_Inbound_Project
```

この項では、次の項目について説明します。

- [10.2.1 項「OSB の Service Bus アプリケーションの作成」](#)
- [10.2.2 項「OSB インバウンド・プロセスの定義」](#)
- [10.2.3 項「OSB インバウンド・プロセスのデプロイ」](#)

### 前提条件

OSB インバウンド・プロセスを設計する前に、アプリケーション・エクスペローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-42 ページの [6.5.1 項「イベント統合のための WSDL の生成」](#) を参照してください。

### 10.2.1 OSB の Service Bus アプリケーションの作成

OSB インバウンド・プロセスを構成するには、OSB の Service Bus アプリケーションを作成する必要があります。詳細は、10-2 ページの [10.1.1 項「OSB の Service Bus アプリケーションの作成」](#) を参照してください。

### 10.2.2 OSB インバウンド・プロセスの定義

この項では、OSB インバウンド・プロセスを定義する方法について説明します。ここでは、次の項目について説明します。

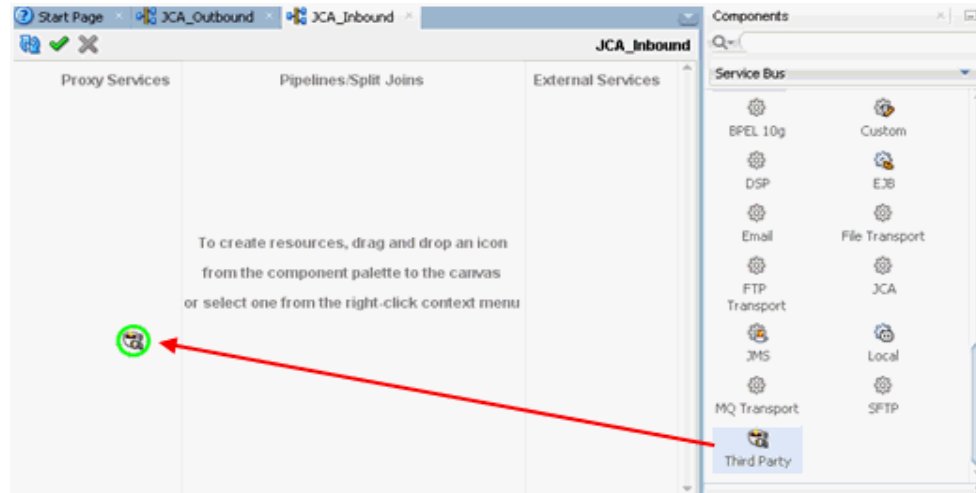
- [10.2.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」](#)
- [10.2.2.2 項「パイプラインの作成」](#)
- [10.2.2.3 項「ファイル・トランスポート・タイプ・ビジネス・サービスの構成」](#)
- [10.2.2.4 項「ルーティング・ルールの構成」](#)

### 10.2.2.1 サード・パーティ・アダプタ・サービス・コンポーネントの構成

サード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

1. 図 10-35 に示すように、「サード・パーティ」アダプタ・コンポーネントを「コンポーネント」の「Service Bus」ペインから「プロキシ・サービス」にドラッグ・アンド・ドロップします。

図 10-35 サード・パーティ・アダプタ・サービス・コンポーネント



「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

2. サード・パーティ・アダプタ・サービスに使用する任意の名前 (CC\_GetDetail など) を入力します。
3. 「タイプ」ドロップダウン・リストで「サービス」が選択されていることを確認します。

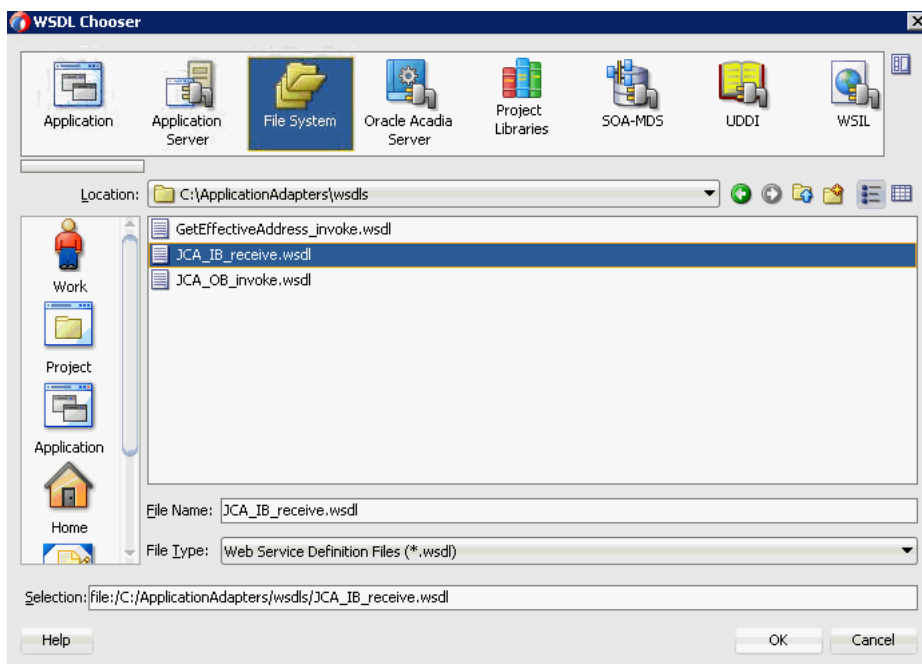
4. 図 10-36 に示すように、「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。

図 10-36 「サード・パーティ・アダプタ・サービス」ダイアログ



図 10-37 に示すように、「WSDL の選択」ダイアログが表示されます。

図 10-37 「WSDL の選択」ダイアログ

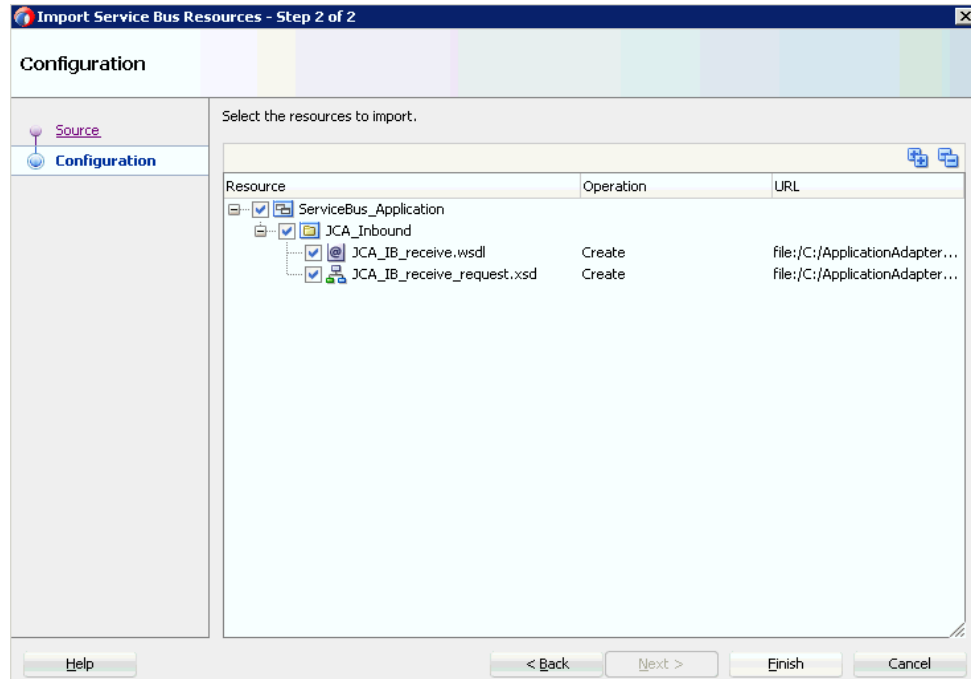


5. 「ファイルシステム」フォルダを選択し、WSDL ディレクトリでインバウンド WSDL ファイルを参照して選択します。



6. 「OK」をクリックします。  
「Service Bus リソースのインポート」ダイアログが表示されます。
7. 「次」をクリックします。
8. 図 10-38 に示すように、「構成」ウィンドウで「終了」をクリックします。

図 10-38 「構成」ウィンドウ



「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

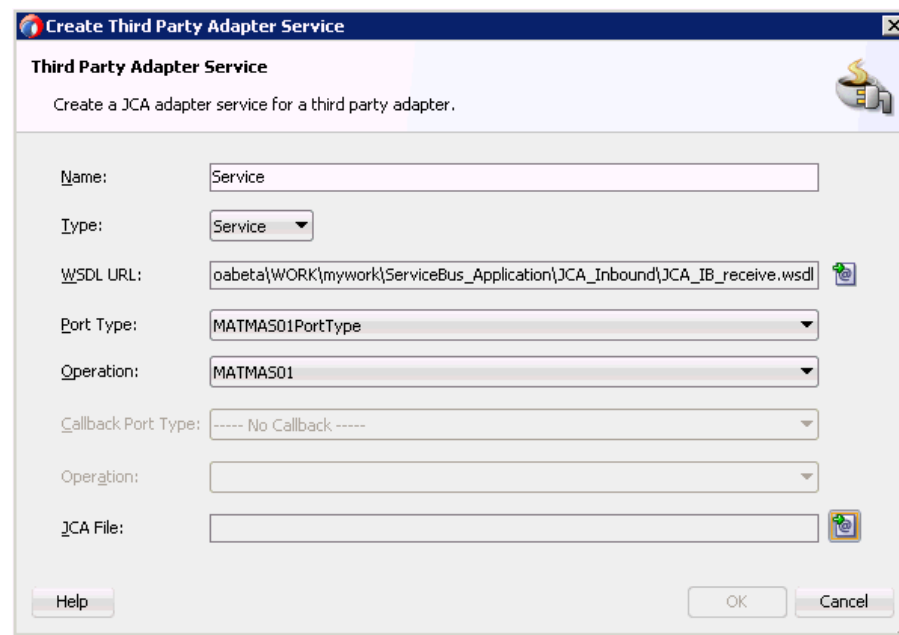
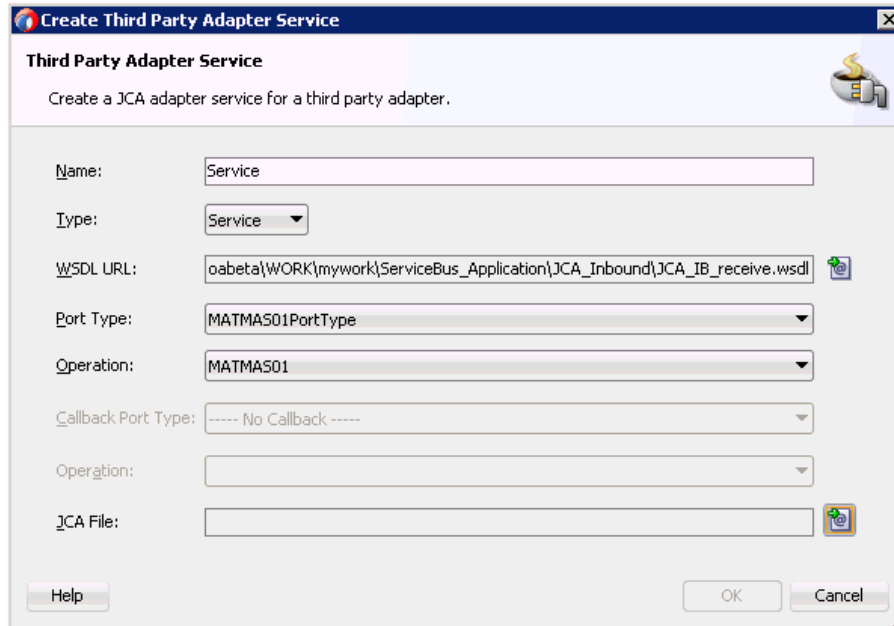
9.  10-39 に示すように、「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。

図 10-39 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



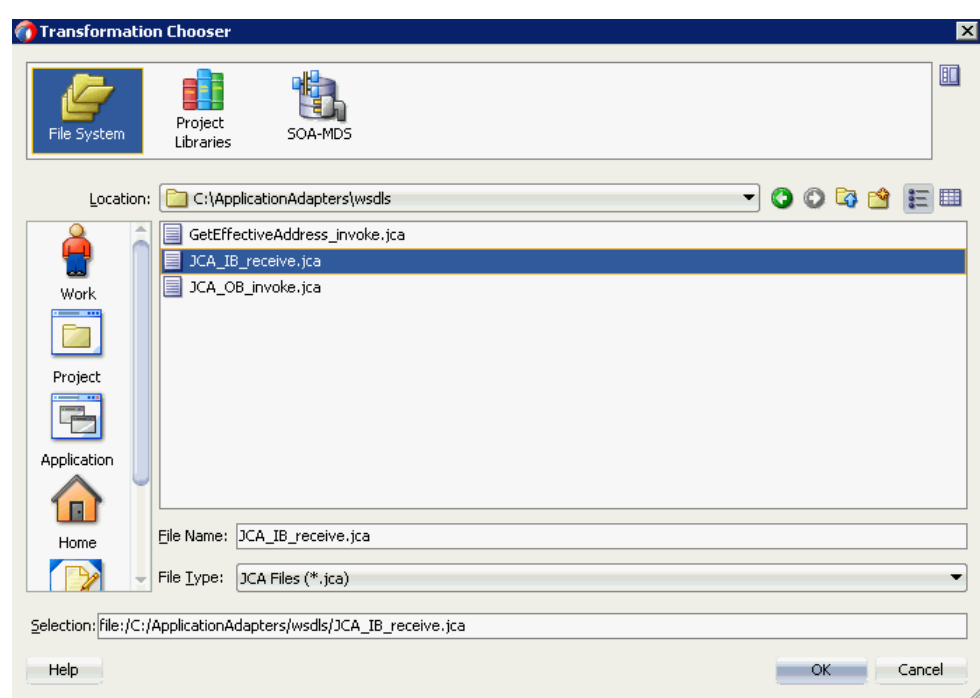
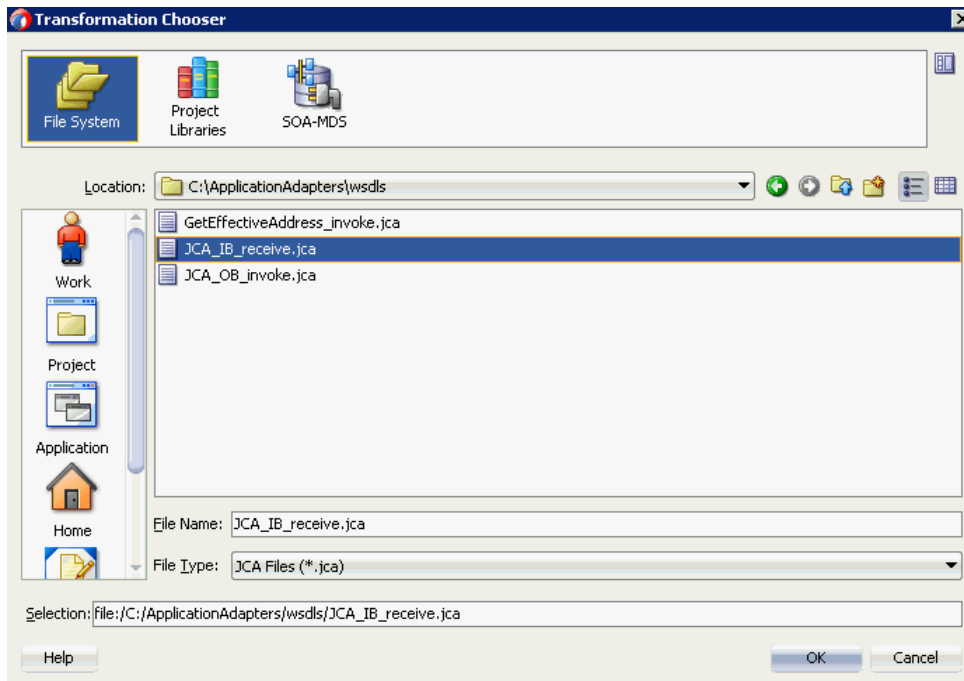
10.  10-40 に示すように、トランスフォーメーション・チューザ・ダイアログが表示されます。

図 10-40 トランスフォーメーション・チューザ・ダイアログ

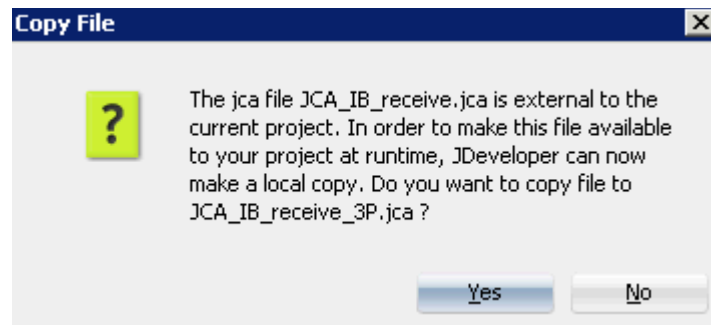


10. WSDL ディレクトリから JCA プロパティ・ファイルを選択します。

11. 「OK」をクリックします。

図 10-41 に示すように、「ファイルのコピー」メッセージが表示されます。

図 10-41 「ファイルのコピー」メッセージ

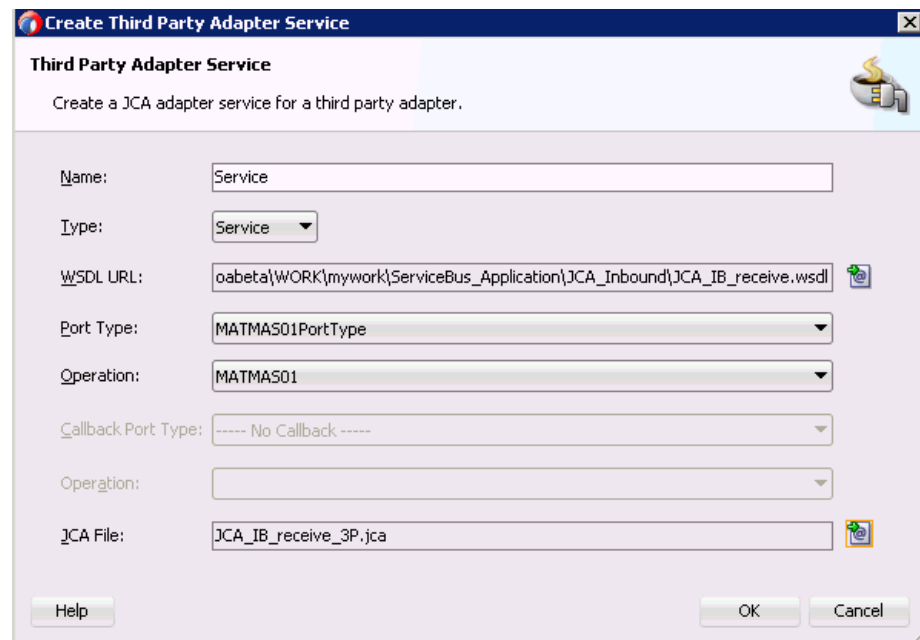


12. 「はい」をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

図 10-42 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

図 10-42 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



13. 「OK」をクリックします。

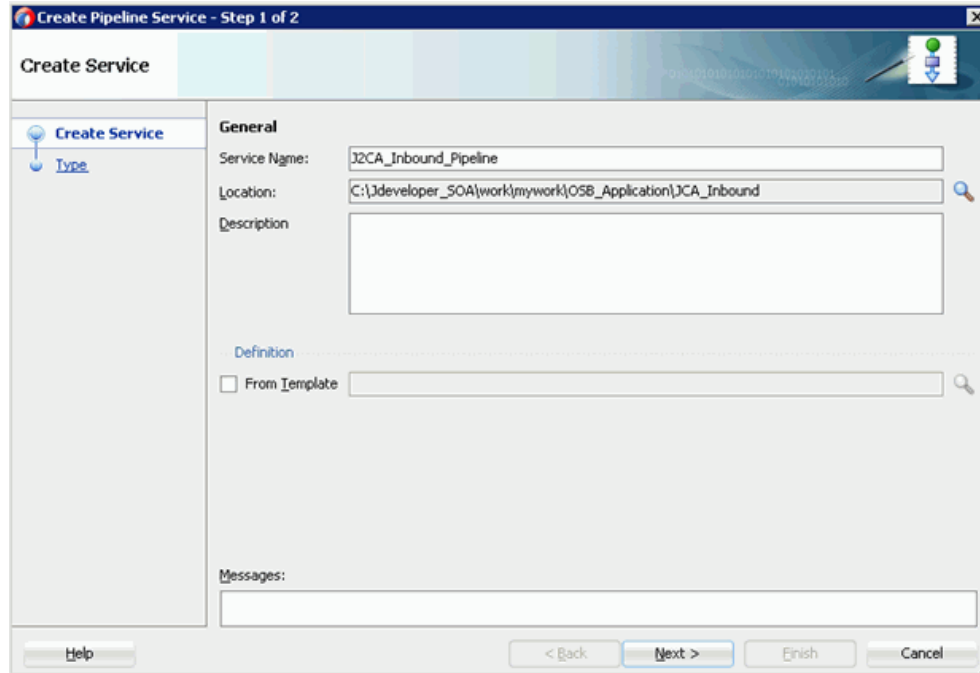
「プロキシ・サービス」ペインにサード・パーティ・アダプタ・サービス・コンポーネントが作成されます。

### 10.2.2.2 パイプラインの作成

パイプラインを含むインバウンド・プロキシ・サービスを生成するには、次のようにします。

1. 「Service Bus」の下で「リソース」をクリックします。
2. パイプラインを「パイプライン/分割結合」ペインにドラッグ・アンド・ドロップします。
3. [図 10-43](#) に示すように、パイプラインの名前を指定して「次」をクリックします。

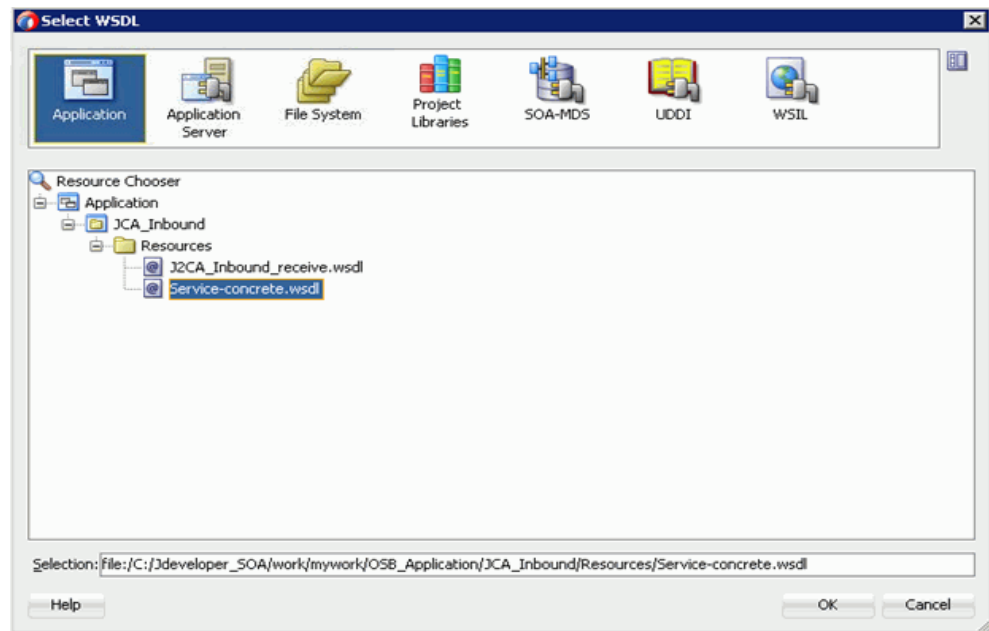
図 10-43 「サービスの作成」 ページ



4. 「パイプライン・サービスの作成」 ウィンドウで、「WSDL」を選択して「WSDL URL」をクリックします。

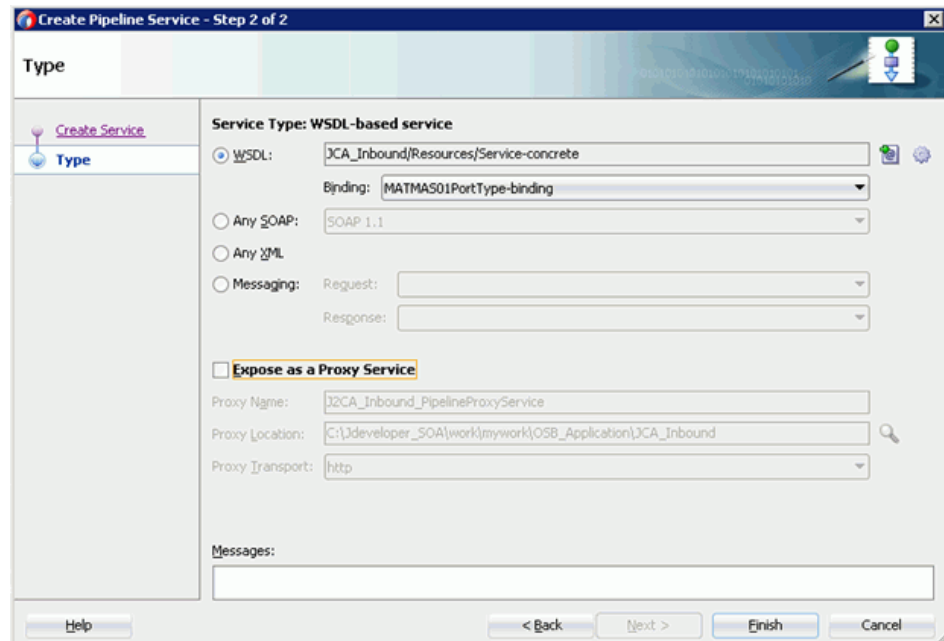
5. 図 10-44 に示すように、「WSDL の選択」 ウィンドウで「アプリケーション」を選択し、適切な OSB オブジェクトの「service-concrete.wsdl」を選択して、「OK」をクリックします。

図 10-44 「WSDL の選択」 ページ



6. 図 10-45 に示すように、「プロキシ・サービスとして公開」チェック・ボックスの選択を解除して、「終了」をクリックします。

図 10-45 「タイプ」 ページ



7. プロキシ・サービスを「パイプライン/分割結合」ペインにドラッグ・アンド・ドロップします。

### 10.2.2.3 ファイル・トランスポート・タイプ・ビジネス・サービスの構成

ファイル・トランスポート・タイプ・ビジネス・サービスを作成するには、次のようにします。

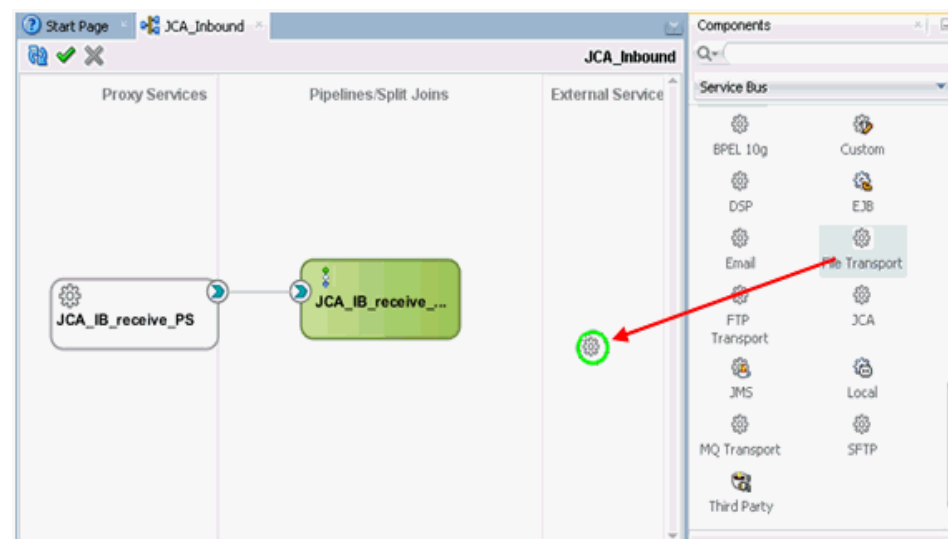
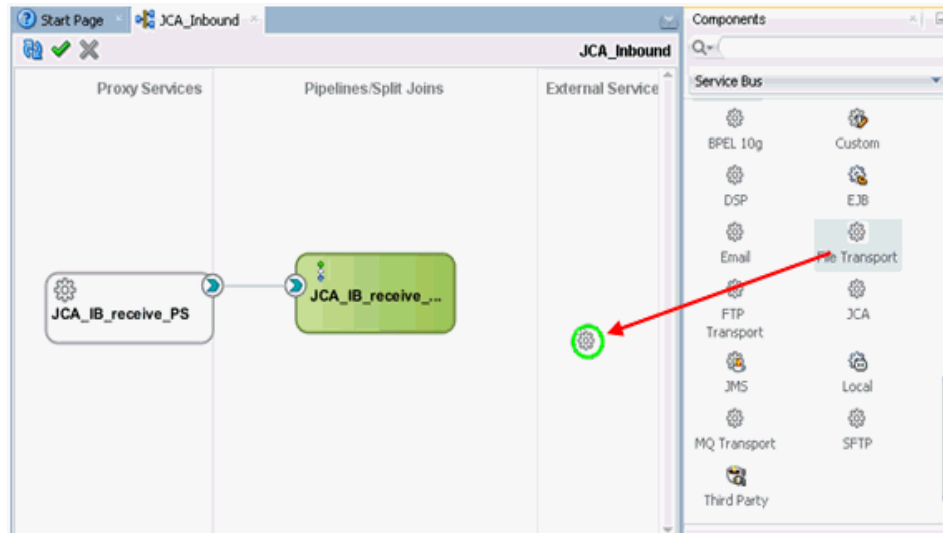
1.  10-46 に示すように、「ファイル・トランスポート」コンポーネントを「詳細」ペインから「外部サービス」ペインにドラッグ・アンド・ドロップします。

図 10-46 「ファイル・トランスポート」ノード



「ビジネス・サービスの作成」ダイアログが表示されます。

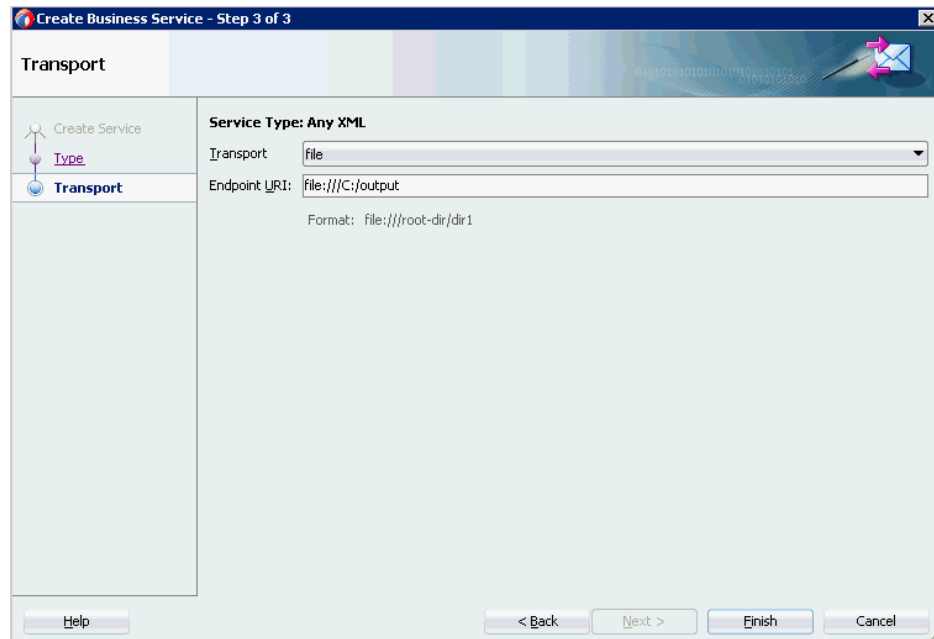
2. 「サービス名」フィールドにビジネス・サービスに使用する任意の名前 (FileOut など) を入力し、「次」をクリックします。

表示される「タイプ」ウィンドウでは、「任意の XML」オプションがデフォルトで選択されています。

3. 「次」をクリックします。

4. 図 10-47 に示すように、表示される「トランスポート」ウィンドウで、「エンドポイント URI」フィールドに出力場所 (c:\output など) を指定し、「終了」をクリックします。

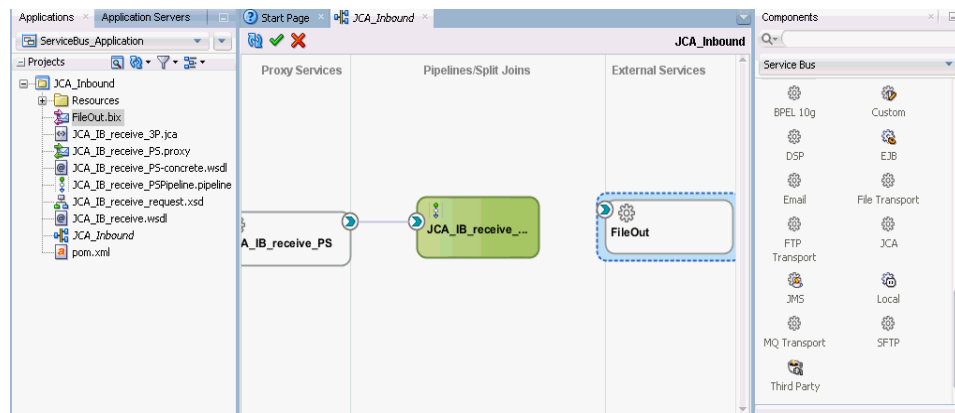
図 10-47 「トランスポート」 ペイン



FileOut ビジネス・サービスが作成されます。

5. 図 10-48 に示すように、FileOut ビジネス・サービスをダブルクリックします。

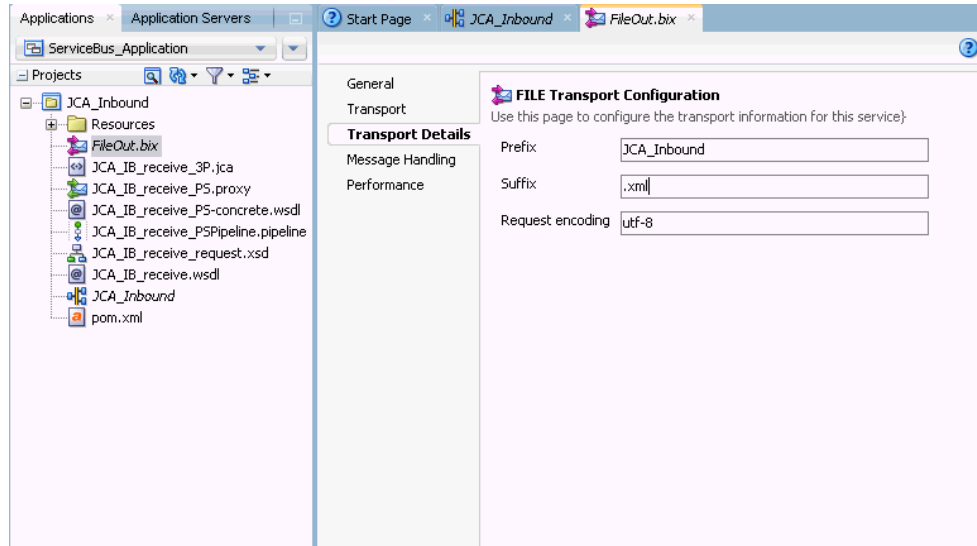
図 10-48 FileOut ビジネス・サービス



構成ページが表示されます。

- 図 10-49 に示すように、「トランスポートの詳細」タブにナビゲートし、「接頭辞」と「接尾辞」の各フィールドに値を指定します。

図 10-49 「ファイル・トランスポート」の構成



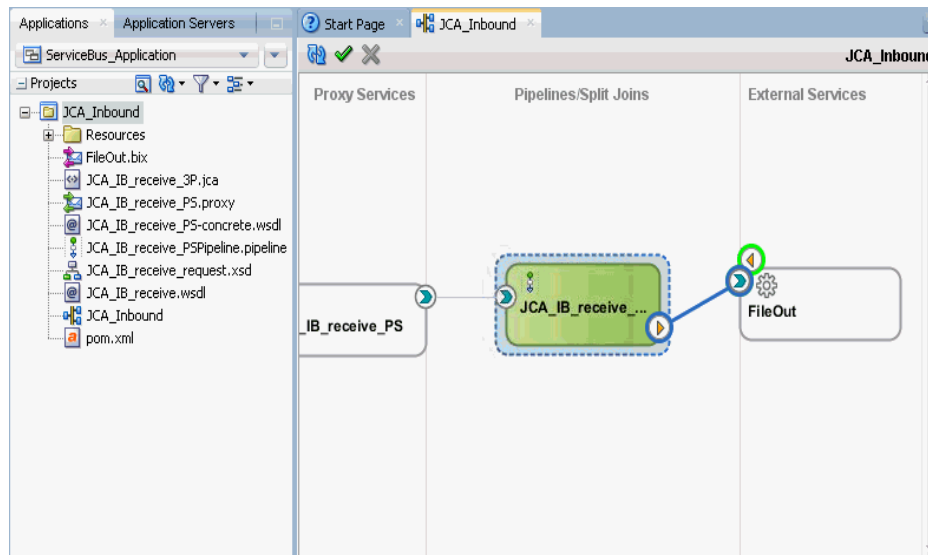
- 構成ページを保存して閉じます。

### 10.2.2.4 ルーティング・ルール of 構成

ルーティング・ルールを構成するには、次のようにします。

- 図 10-50 に示すように、パイプライン (JCA\_IB\_receive\_PSPipeline など) とファイル・タイプ・ビジネス・サービス (FileOut など) との間に接続を作成します。

図 10-50 プロキシと FileOut のマッピング



- パイプライン (JCA\_IB\_receive\_PSPipeline など) をダブルクリックします。



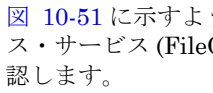
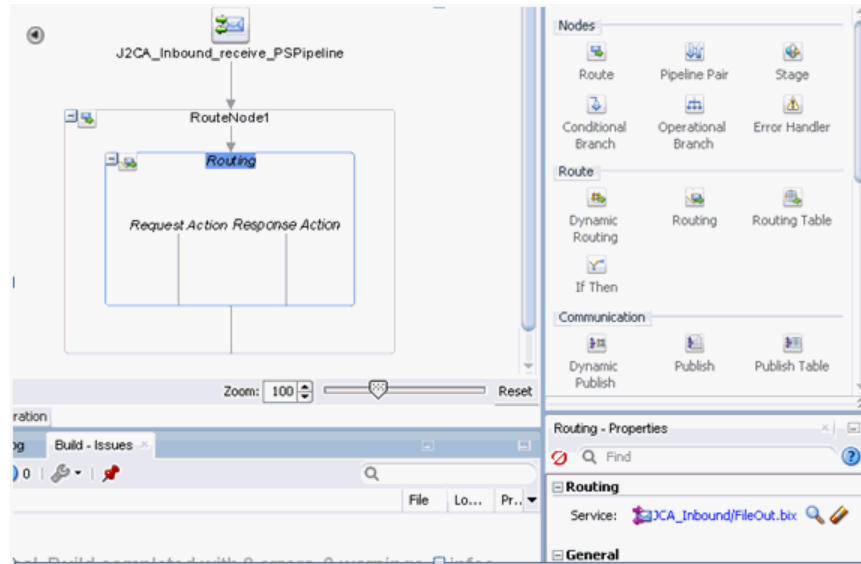
3.  10-51 に示すように、「ルーティング」ペインをクリックし、ファイル・タイプ・ビジネス・サービス (FileOut など) が「サービス」フィールドに正しく構成されていることを確認します。

図 10-51 「ルーティング」ペイン




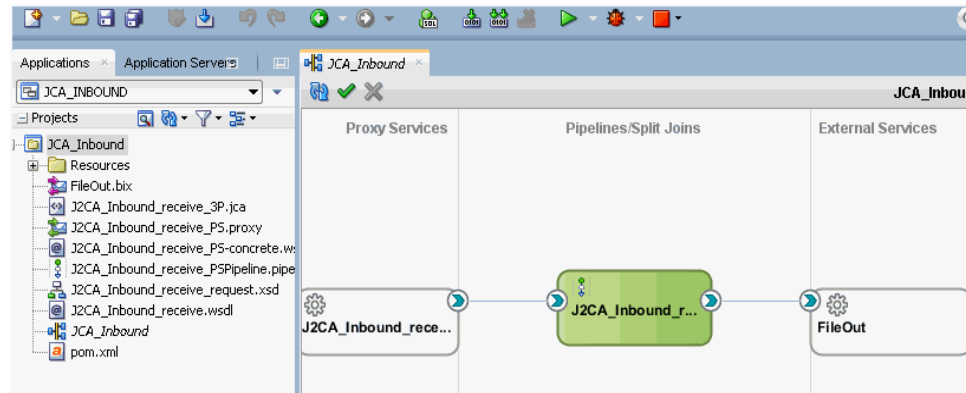
4. パイプラインの構成ページを保存して閉じます。
5.  10-52 に示すように、overview.xml ファイル (JCA\_Inbound など) をダブルクリックし、メニュー・バーの「すべて保存」をクリックして OSB プロセスを保存します。

図 10-52 すべて保存



## 10.2.3 OSB インバウンド・プロセスのデプロイ

作成された OSB インバウンド・プロセスをデプロイするには、10-17 ページの [10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」](#)に記載されている手順 1 から 4 を参照してください。

OSB インバウンド・プロセスが正常にデプロイされたら、SAP R/3 システムからイベントをトリガーし、出力が構成済の出力場所 (C:\output など) で受信されることを確認します。

## 10.3 JDeveloper を使用した OSB アウトバウンド・プロセスの構成 (BSE 構成)

この項では、Oracle JDeveloper を使用して、SAP R/3 システムへの OSB アウトバウンド・プロセスを構成する方法 (BSE 構成) について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\BSE\SAP_Sample_BSE_OSB_Outbound_Project
```

この項では、次の項目について説明します。

- [10.3.1 項「OSB の Service Bus アプリケーションの作成」](#)
- [10.3.2 項「OSB アウトバウンド・プロセスの定義」](#)
- [10.3.3 項「OSB アウトバウンド・プロセスのデプロイ」](#)

### 前提条件

OSB アウトバウンド・プロセスを設計する前に、アプリケーション・エクスペローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-71 ページの [6.7.1 項「Web サービスを使用したリクエストおよびレスポンス・サービスの WSDL ファイルの生成」](#)を参照してください。

### 10.3.1 OSB の Service Bus アプリケーションの作成

OSB アウトバウンド・プロセスを構成するには、OSB の Service Bus アプリケーションを作成する必要があります。詳細は、10-2 ページの [10.1.1 項「OSB の Service Bus アプリケーションの作成」](#)を参照してください。

### 10.3.2 OSB アウトバウンド・プロセスの定義

この項では、OSB アウトバウンド・プロセスを定義する方法について説明します。ここでは、次の項目について説明します。

- [10.3.2.1 項「WSDL ベースのビジネス・サービスの構成」](#)
- [10.3.2.2 項「パイプラインを含むプロキシ・サービスの作成」](#)
- [10.3.2.3 項「ファイル・トランスポート・タイプ・ビジネス・サービスの構成」](#)
- [10.3.2.4 項「ルーティング・ルールの構成」](#)

### 10.3.2.1 WSDL ベースのビジネス・サービスの構成

WSDL ベースのビジネス・サービスを構成するには、次のようにします。

1. 図 10-53 に示すように、「HTTP」コンポーネントを「コンポーネント」の「テクノロジー」ペインから「外部サービス」領域にドラッグ・アンド・ドロップします。

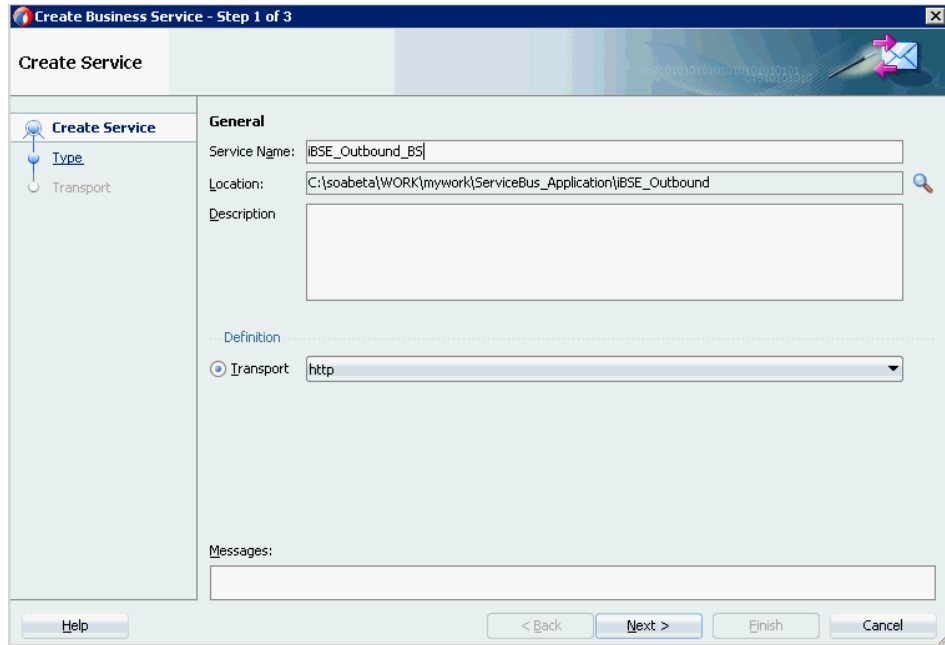
図 10-53 HTTP コンポーネント



「ビジネス・サービスの作成」ウィンドウが表示されます。

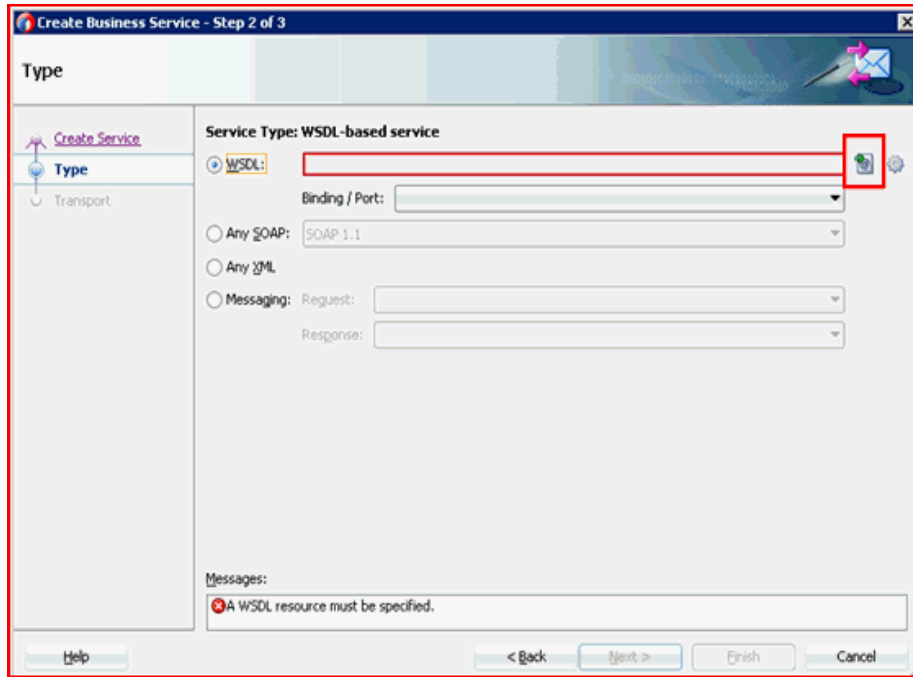
2. 図 10-54 に示すように、「サービス名」フィールドにビジネス・サービスに使用する任意の名前を入力し、「次」をクリックします。

図 10-54 ビジネス・サービスの作成



3. 図 10-55 に示すように、表示される「サービス・タイプ」ウィンドウで「WSDL」オプションを選択し、「WSDL の選択」アイコンをクリックします。

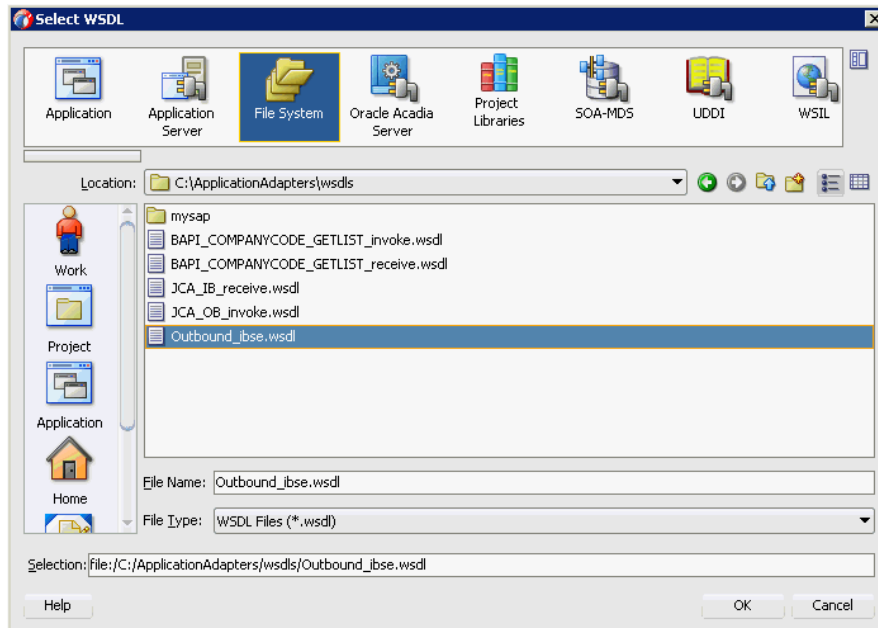
図 10-55 「タイプ」 ペイン



「WSDL の選択」ウィンドウが表示されます。

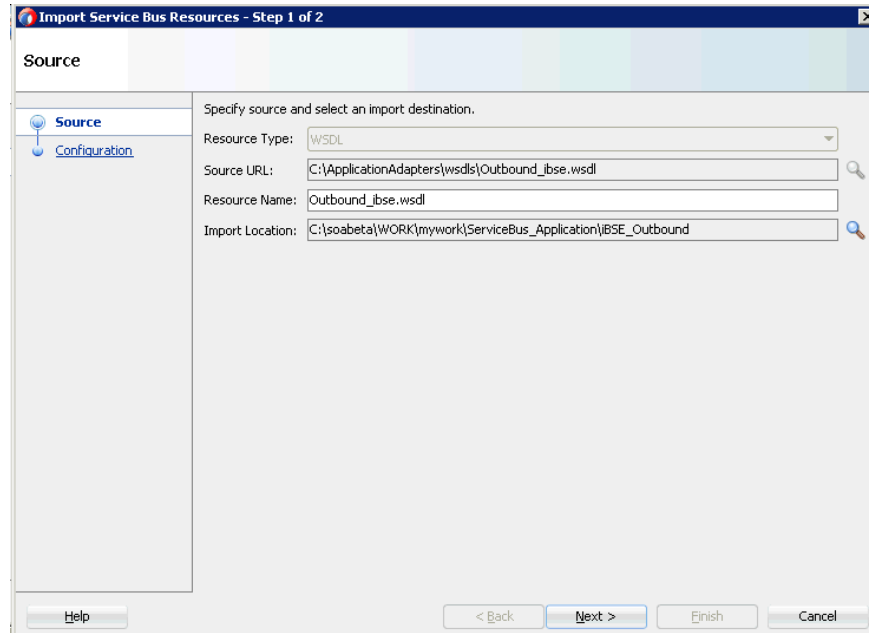
4. 図 10-56 に示すように、「ファイルシステム」フォルダ・アイコンを選択し、WSDL の場所から iBSE WSDL ファイルを参照して選択し、「OK」をクリックします。

図 10-56 「WSDL の選択」ウィンドウ



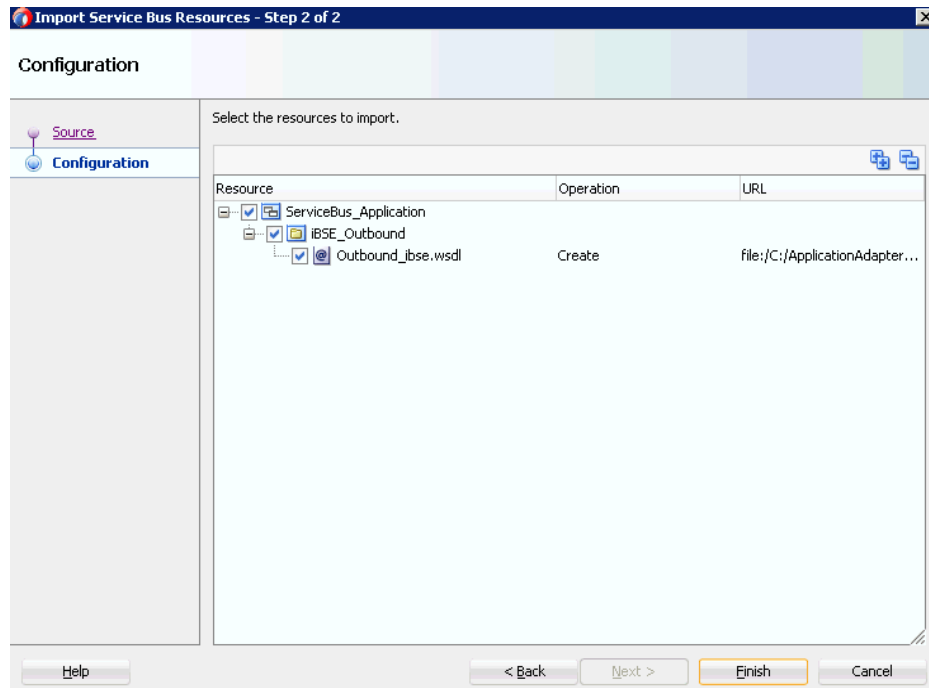
5. 図 10-57 に示すように、表示される「ソース」ペインで「次」をクリックします。

図 10-57 「ソース」ペイン



6. 図 10-58 に示すように、表示される「構成」ペインで「終了」をクリックします。

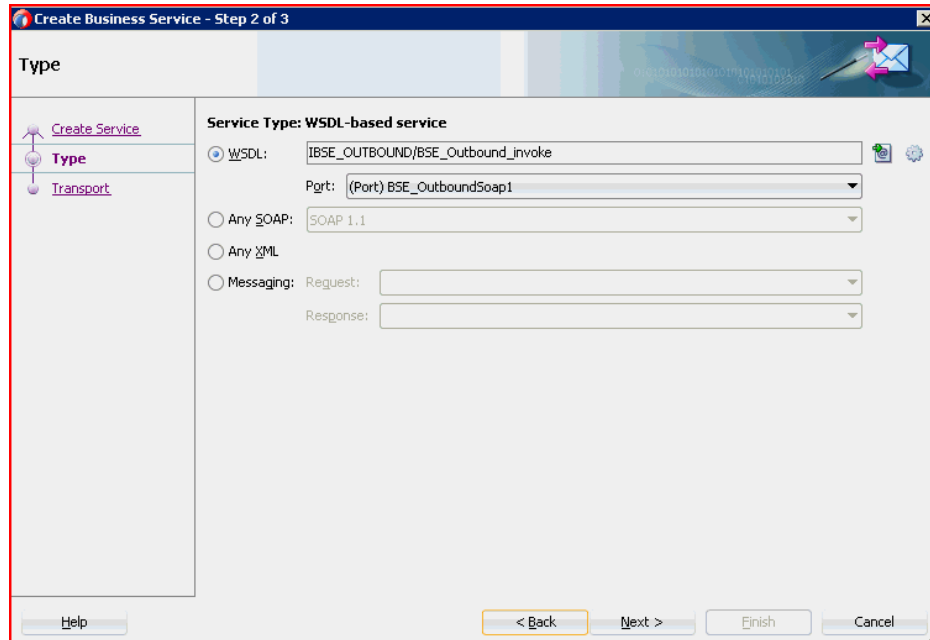
図 10-58 「構成」ペイン



「ビジネス・サービスの作成」ウィンドウに戻ります。

7. 図 10-59 に示すように、表示される「タイプ」ペインで「次」をクリックします。

図 10-59 「タイプ」ペイン



8. 図 10-60 に示すように、ホスト名とポート番号が異なる場合には、表示される「トランスポート」ウィンドウで「エンドポイント URI」フィールドを変更し、「終了」をクリックします。

図 10-60 「トランスポート」ペイン

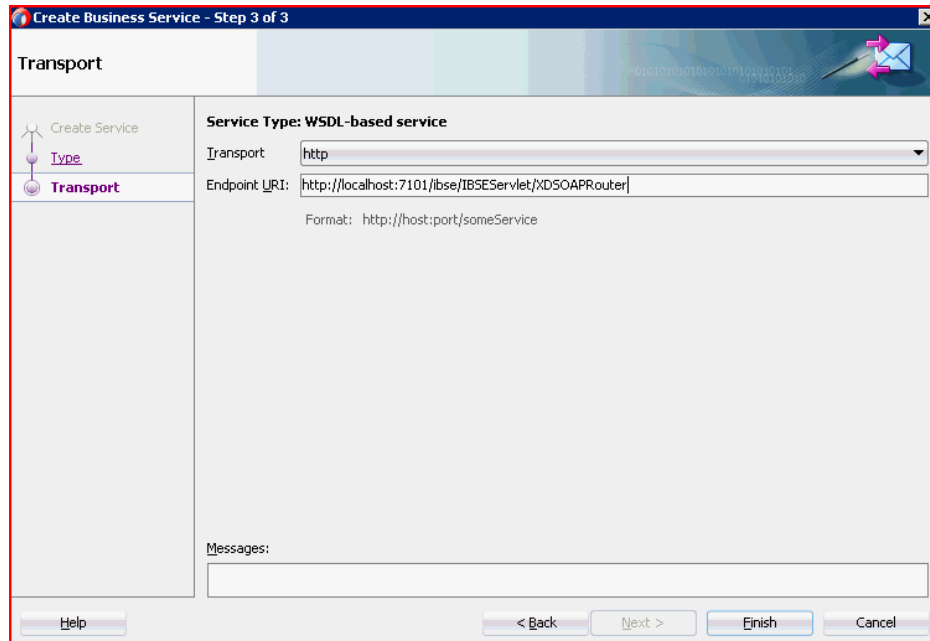
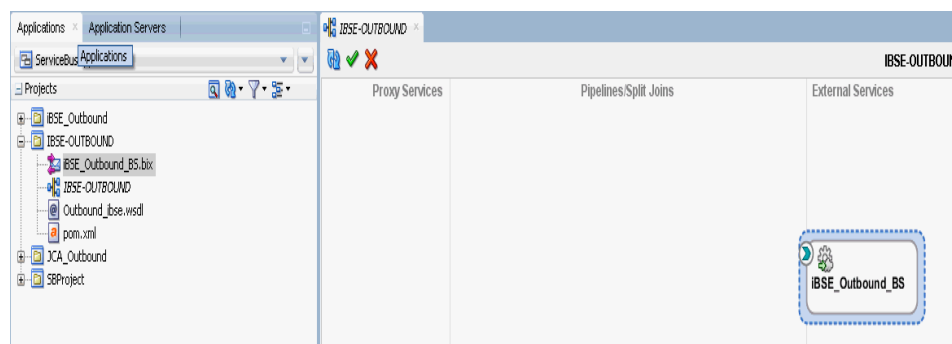


図 10-61 に示すように、「外部サービス」ペインにビジネス・サービスが作成されて表示されます。

図 10-61 「外部サービス」ペイン

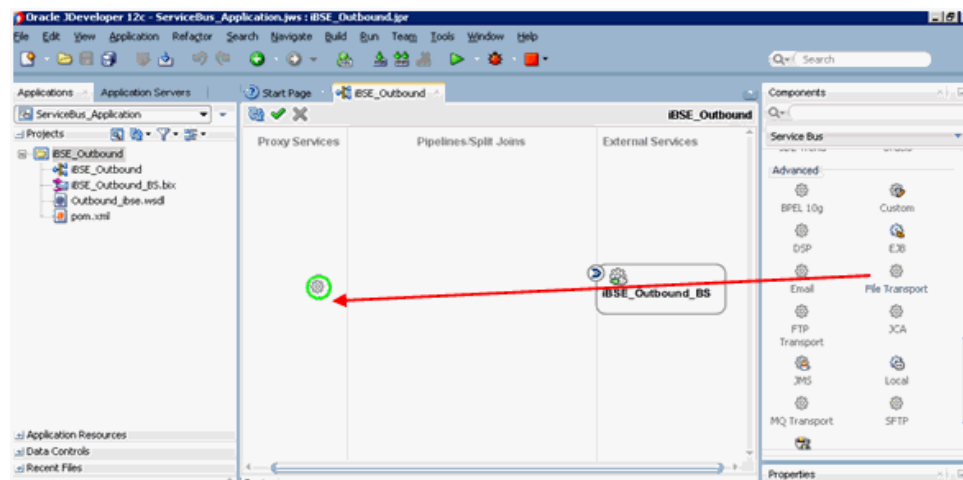


### 10.3.2.2 パイプラインを含むプロキシ・サービスの作成

パイプラインを含むプロキシ・サービスを作成するには、次のようにします。

1. 図 10-62 に示すように、「ファイル・トランスポート」コンポーネントを「コンポーネント」の「詳細」ペインから「プロキシ・サービス」ペインにドラッグ・アンド・ドロップします。

図 10-62 ファイル・トランスポート・コンポーネント

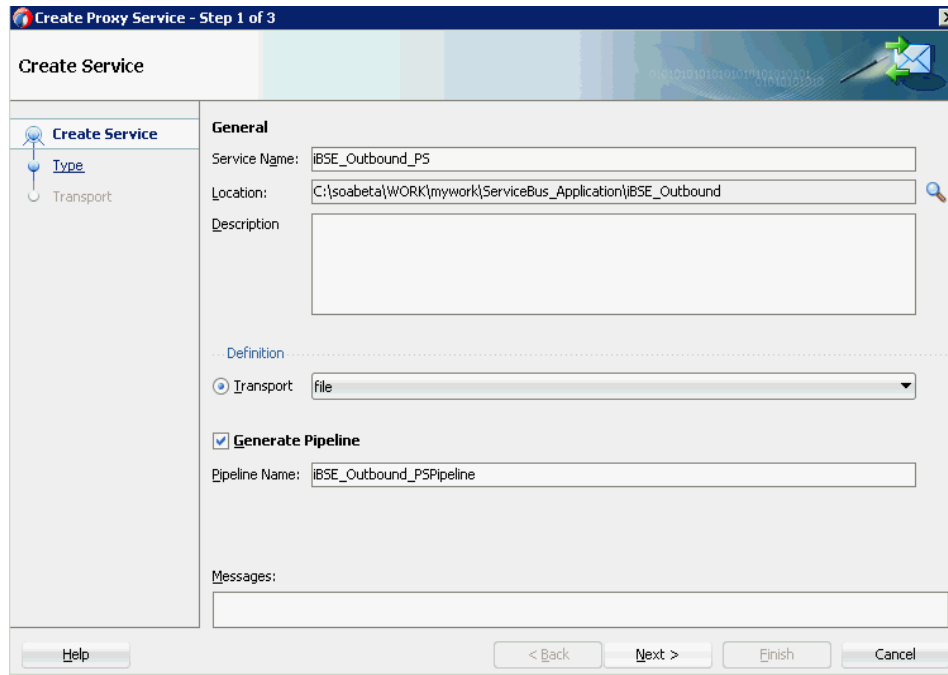


「プロキシ・サービスの作成」ペインが表示されます。

2. 「サービス名」フィールドに、プロキシ・サービスに使用する任意の名前 (JCA\_Outbound\_PS など) を入力します。デフォルトでは、「パイプラインの生成」が選択されています。

3. 図 10-63 に示すように、「次」をクリックします。

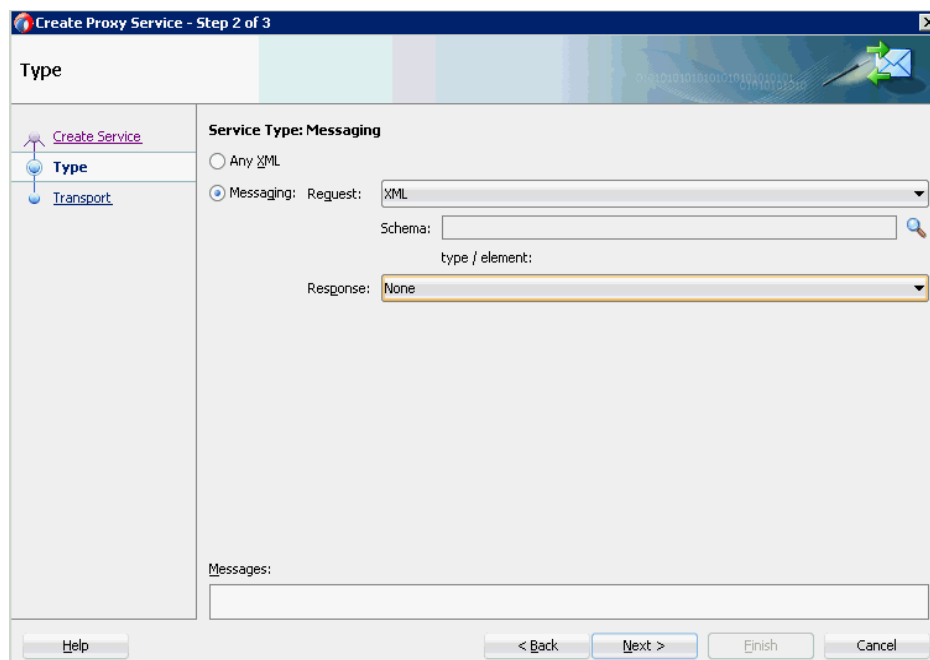
図 10-63 「サービスの作成」 ペイン



「タイプ」 ペインが表示されます。

4. 図 10-64 に示すように、「メッセージング」 オプションを選択し、「リクエスト」を「XML」に、「レスポンス」を「なし」に設定してから、「次」をクリックします。

図 10-64 「タイプ」 ペイン

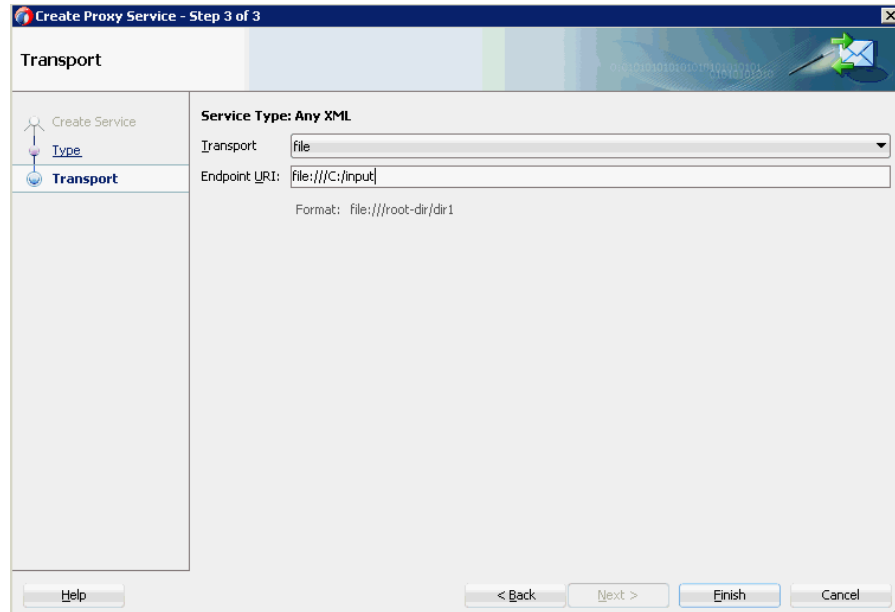


「トランスポート」 ウィンドウが表示されます。



5. 図 10-65 に示すように、「エンドポイント URI」フィールドに入力場所 (c:/input など) を指定し、「終了」をクリックします。

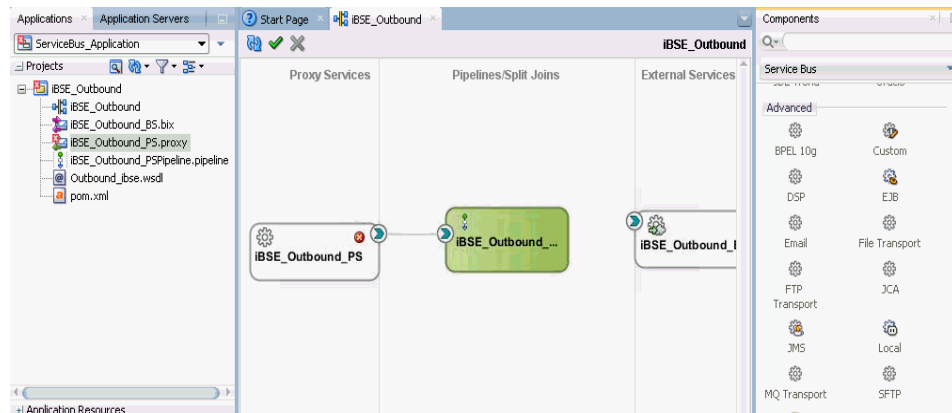
図 10-65 「トランスポート」ウィンドウ



プロキシ・サービスが、パイプラインとともに作成されて表示されます。

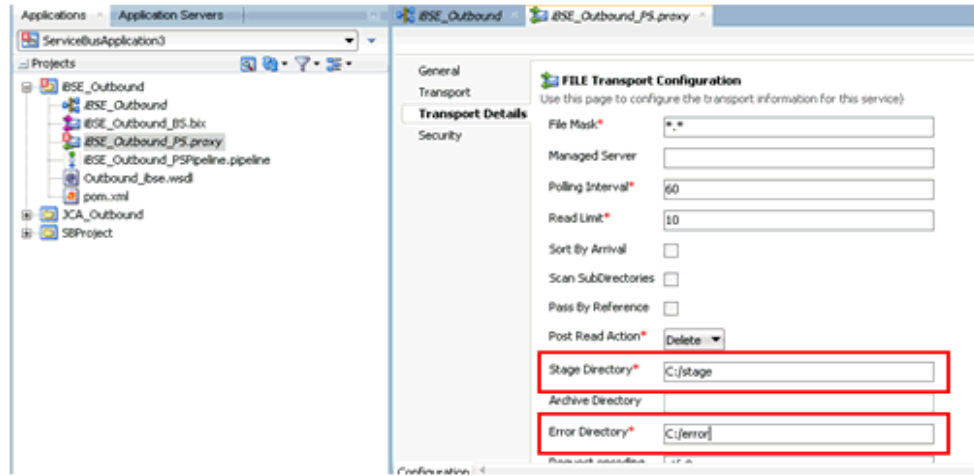
6. 図 10-66 に示すように、作成されたプロキシ・サービス (iBSE\_Outbound\_PS など) をダブルクリックします。

図 10-66 プロキシ・サービスの編集



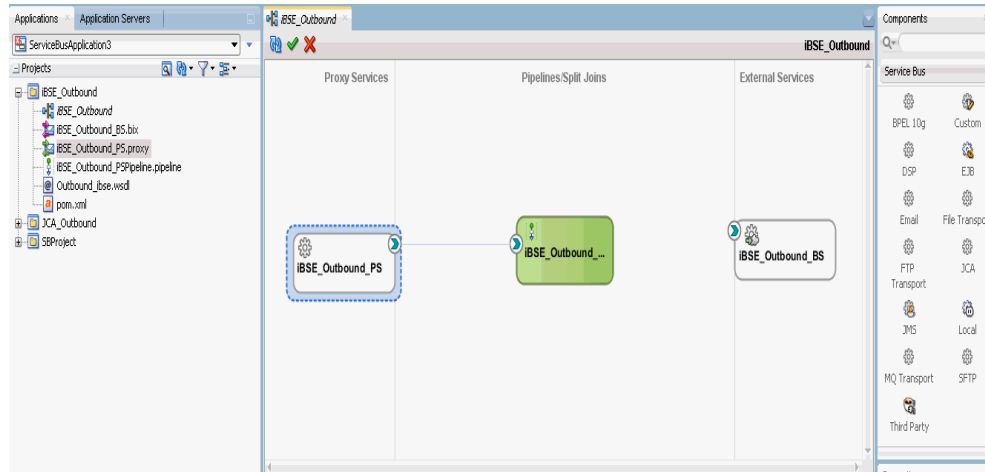
7. 図 10-67 に示すように、表示されるプロキシ・サービスの構成ページで「**トランスポートの詳細**」を選択し、「**ステージ・ディレクトリ**」と「**エラー・ディレクトリ**」に値を指定します。

図 10-67 「ファイル・トランスポート」の構成



8. プロキシ・サービスの構成ページを保存して閉じます。
9. overview.xml ファイル (iBSE\_Outbound など) をダブルクリックします。  
図 10-68 に示すように、プロキシ・サービスが更新されて表示されます。

図 10-68 プロキシ・サービス



### 10.3.2.3 ファイル・トランスポート・タイプ・ビジネス・サービスの構成

ファイル・トランスポート・タイプ・ビジネス・サービスを作成するには、次のようにします。


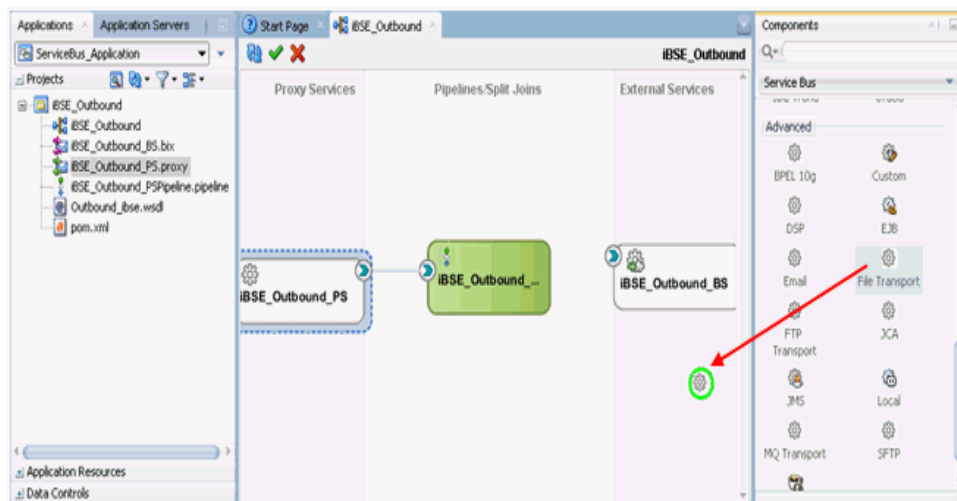
1.  10-69 に示すように、「ファイル・トランスポート」コンポーネントを「詳細」ペインから「外部サービス」ペインにドラッグ・アンド・ドロップします。

図 10-69 ファイル・トランスポート・コンポーネント



「ビジネス・サービスの作成」ダイアログが表示されます。


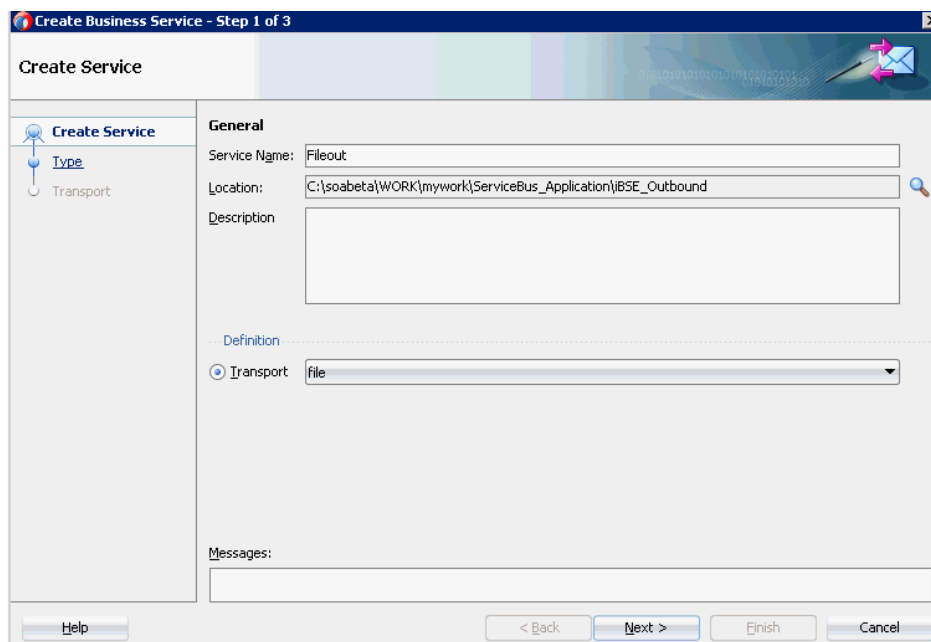
2.  10-70 に示すように、「サービス名」フィールドにビジネス・サービスに使用する任意の名前 (FileOut など) を入力し、「次」をクリックします。

図 10-70 「サービスの作成」ペイン



「タイプ」ペインが表示されます。デフォルトでは、「任意の XML」オプションが選択されています。

3. 「次」をクリックします。  
「トランスポート」ペインが表示されます。
4. 図 10-71 に示すように、「エンドポイント URI」フィールドに出力場所 (c:/output など) を指定し、「終了」をクリックします。

図 10-71 「トランスポート」ペイン

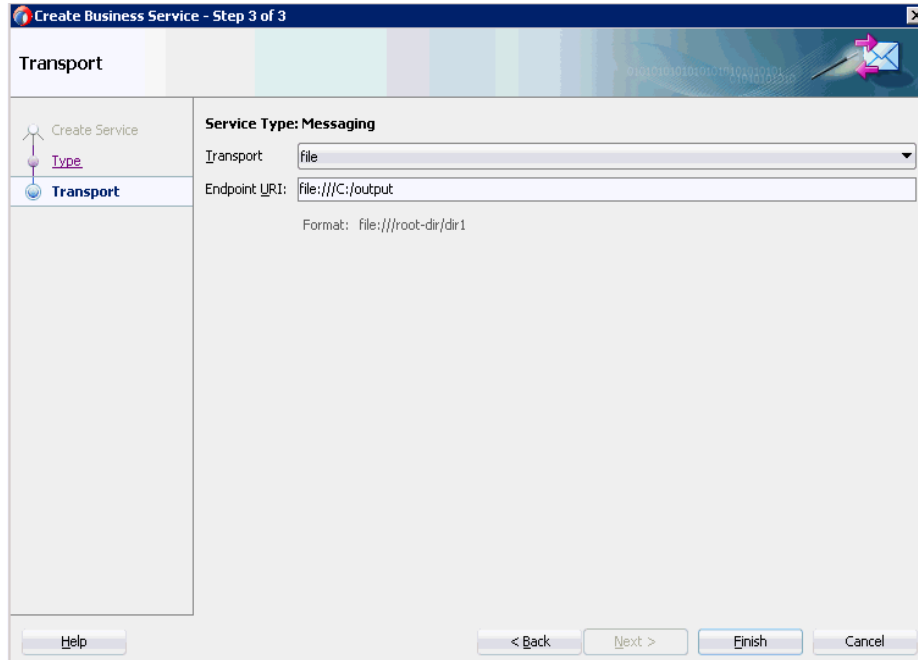
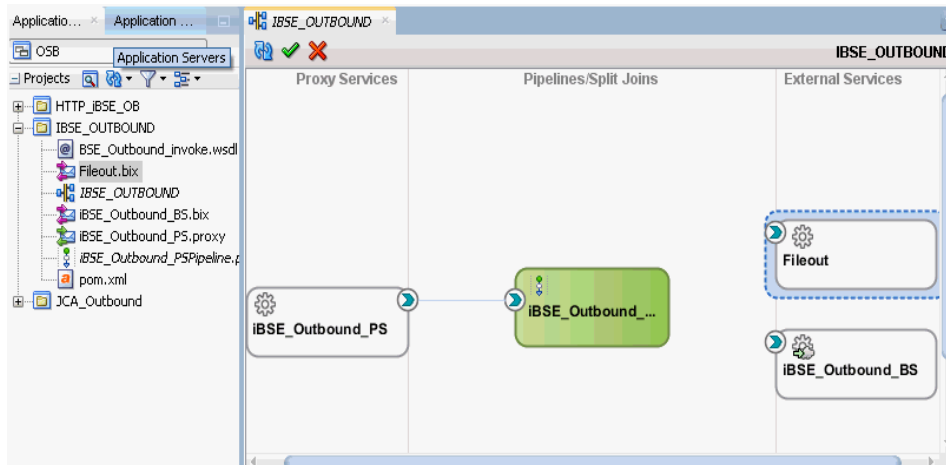


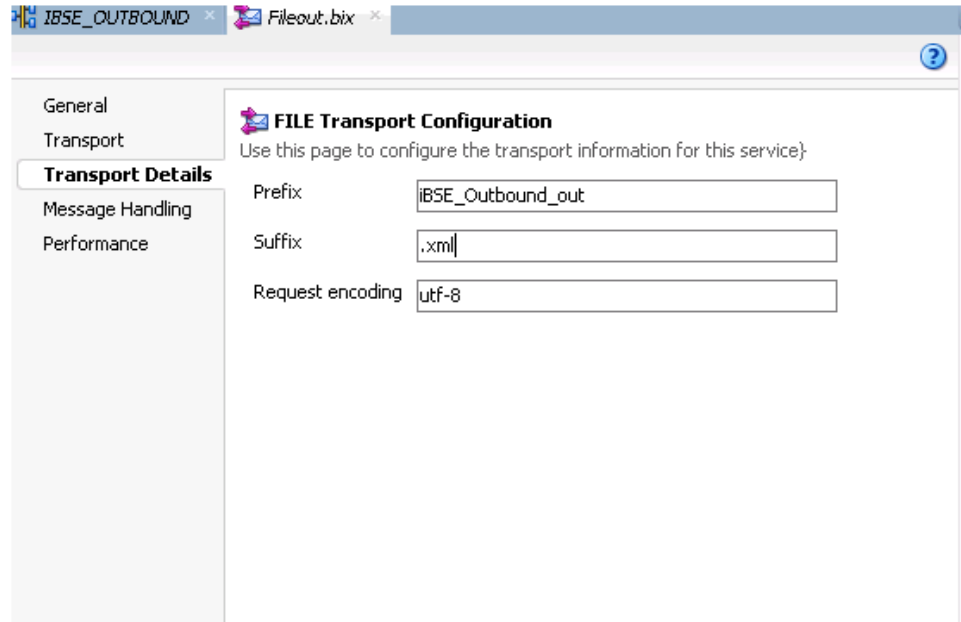
図 10-72 に示すように、ファイル・トランスポート・ビジネス・サービス Fileout が作成されて表示されます。

図 10-72 Fileout ビジネス・サービス



5. 図 10-73 に示すように、作成されたビジネス・サービス **Fileout** をダブルクリックし、「トランスポートの詳細」タブで「接頭辞」と「接尾辞」の各フィールドに値を指定します。

図 10-73 トランスポートの詳細



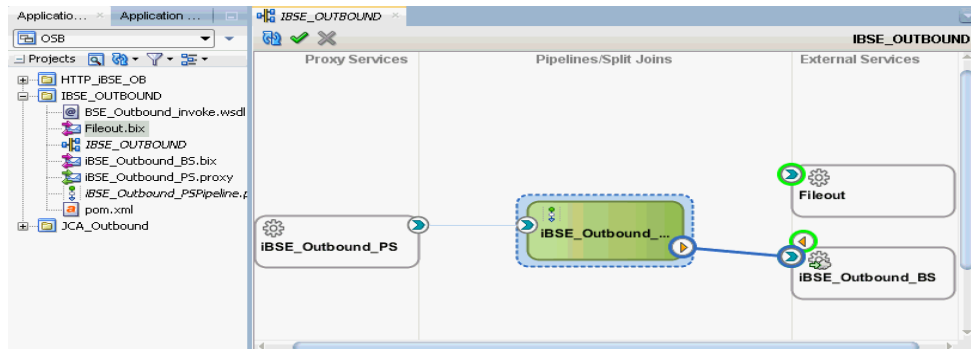
6. 構成ページを保存して閉じ、overview.xml (iBSE\_Outbound など) をダブルクリックします。

### 10.3.2.4 ルーティング・ルール構成

ルーティング・ルールを構成するには、次のようにします。

1. 図 10-74 に示すように、パイプライン・コンポーネント (iBSE\_Outbound\_PSPipeline など) と WSDL ベースのビジネス・サービス (iBSE\_Outbound\_BS など) との間に接続を作成します。

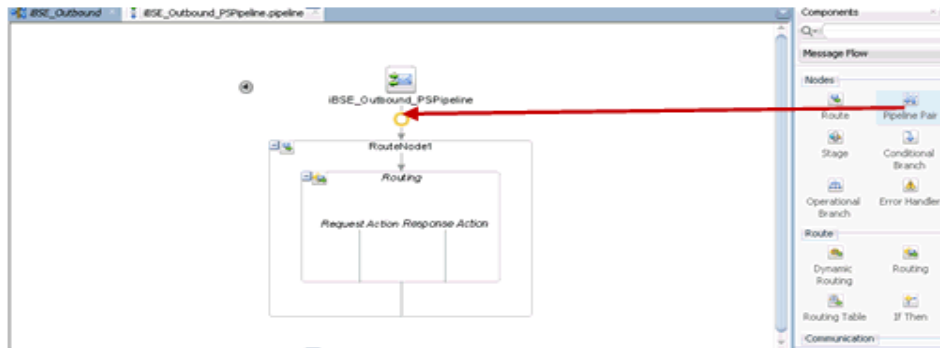
図 10-74 パイプライン・コンポーネント



2. 「パイプライン/分割結合」ペインで、パイプライン・コンポーネント (iBSE\_Outbound\_PSPipeline など) をダブルクリックします。

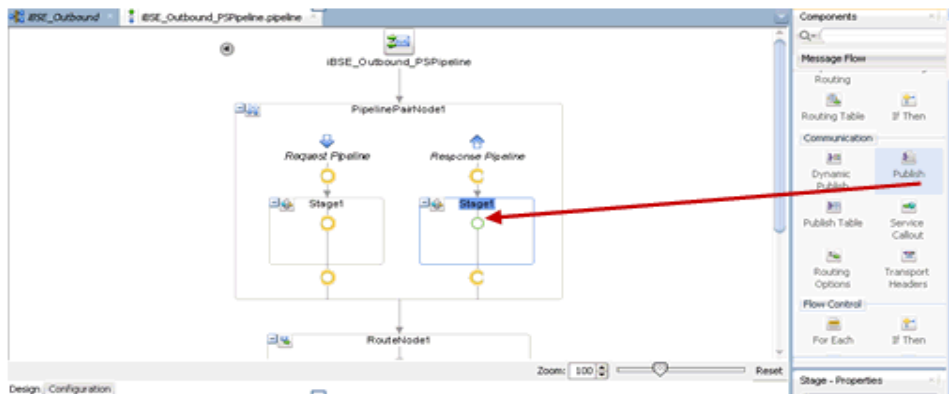
3. 図 10-75 に示すように、「パイプライン・ペア」ノードを「ノード」ペインからパイプライン (iBSE\_Outbound\_PSPipeline など) と「RouteNode1」の間の領域にドラッグ・アンド・ドロップします。

図 10-75 「パイプライン・ペア」ノード



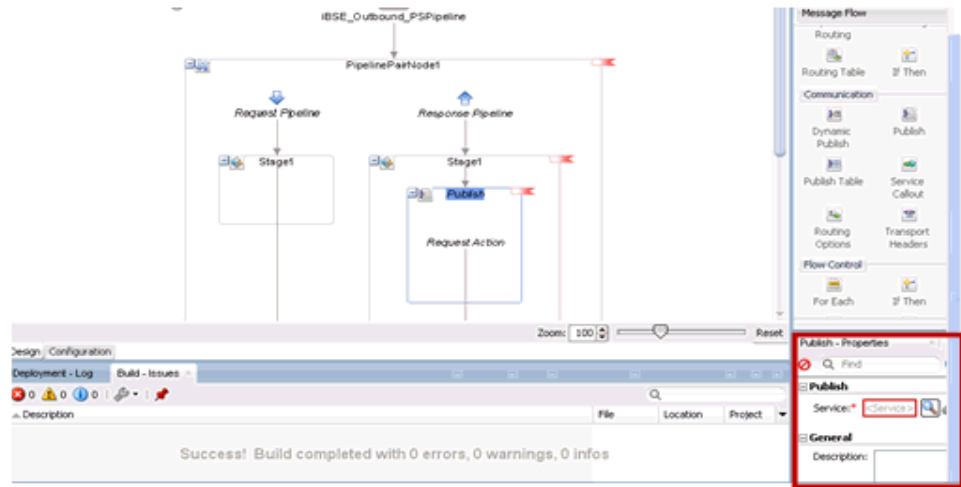
4. 図 10-76 に示すように、「パブリッシュ」ノードを「通信」ペインから「レスポンス・パイプライン」の「ステージ 1」の下の領域にドラッグ・アンド・ドロップします。

図 10-76 「パブリッシュ」ノード



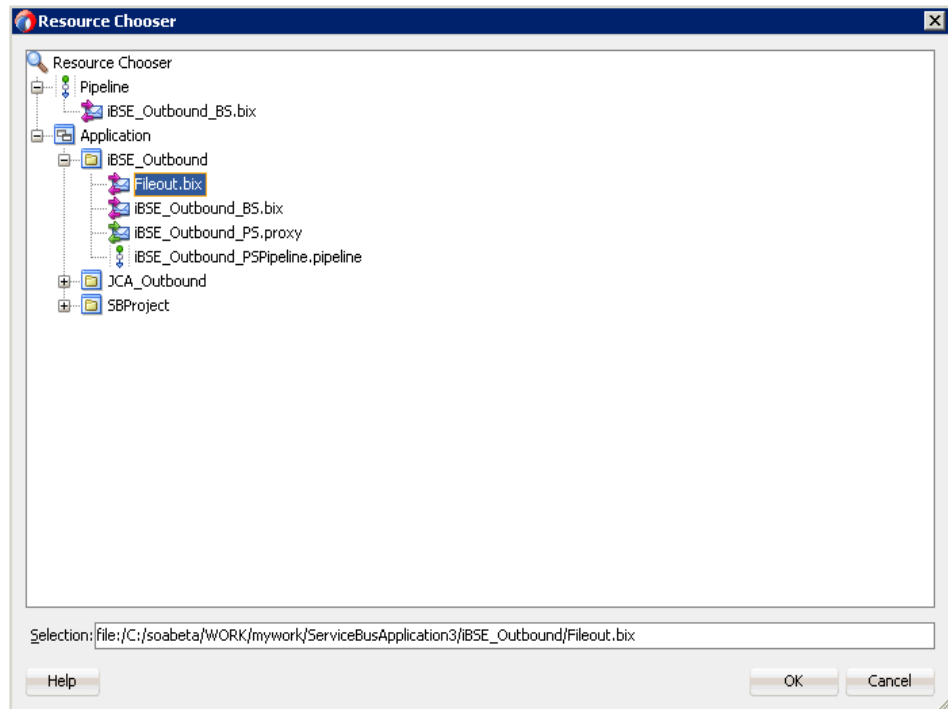
5. 図 10-77 に示すように、「パブリッシュ - プロパティ」の右ペインで、「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 10-77 「参照」アイコン



6. 図 10-78 に示すように、表示される「リソース・チューザ」ウィンドウで、「Fileout.bix」ファイル・トランスポート・ビジネス・サービスを選択して「OK」をクリックします。

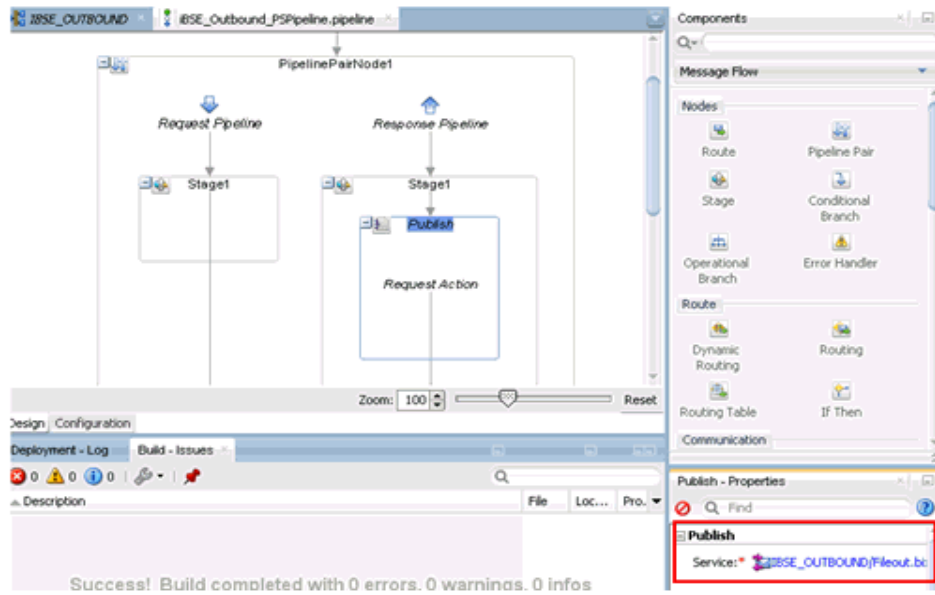
図 10-78 リソース・チューザ



パイプラインの構成ページに戻ります。

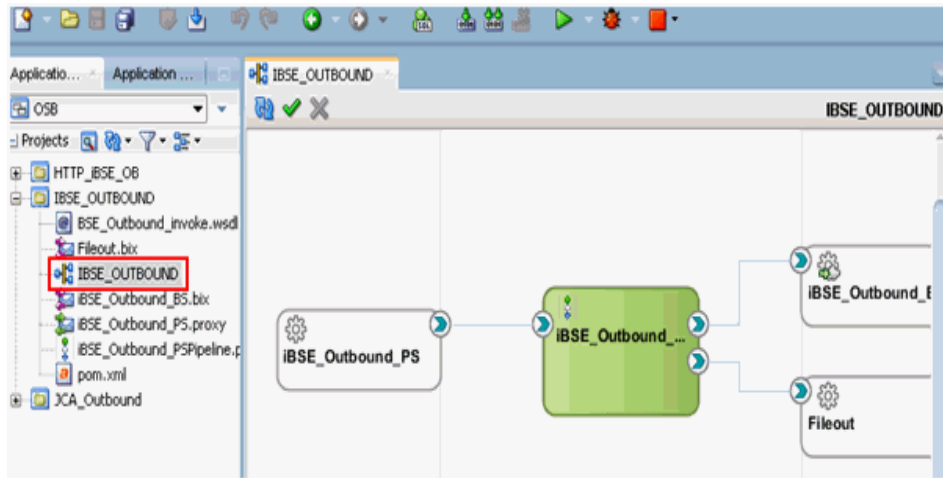
図 10-79 に示すように、選択したサービスが右ペインの「パブリッシュ」ペインに構成されます。

図 10-79 「パブリッシュ」ペイン



7. パイプラインの構成ページを保存して閉じます。
8. 図 10-80 に示すように、overview.xml ファイル (iBSE\_Outbound など) をダブルクリックし、メニュー・バーの「すべて保存」をクリックして OSB プロセスを保存します。

図 10-80 「すべて保存」アイコン





### 10.3.3 OSB アウトバウンド・プロセスのデプロイ

作成された OSB アウトバウンド・プロセスをデプロイして、入力 XML ドキュメントを呼び出すには、[10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」](#)を参照してください。

## 10.4 JDeveloper を使用した Oracle Service Bus の同期 SAP R/3 イベントの処理 (J2CA 構成)

この項では、J2CA 構成の Oracle JDeveloper を使用して Oracle Service Bus の同期 SAP R/3 イベントを処理する方法について説明します。

アプリケーション・アダプタ・インストールの次のフォルダに、このユースケース・シナリオ用のサンプル・プロジェクトが用意されています。

```
<ADAPTER_HOME>\etc\sample\SAP_Samples.zip\SAP_Samples\OSB_Jdeveloper\J2CA\SAP_Sample_
J2CA_OSB_Synchronous_Events
```

この項では、次の項目について説明します。

- [10.4.1 項「OSB の Service Bus アプリケーションの作成」](#)
- [10.4.2 項「OSB プロセスの定義」](#)
- [10.4.3 項「OSB プロセスのデプロイ」](#)

#### 前提条件

OSB プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。

メッセージの WSDL ファイルを生成するには、次のようにします。

1. 最初の SAP アダプタ・ターゲット (target1 など) およびチャネルを作成します。WSDL ファイルを生成する前に、チャネルが開始されていないことを確認します。
2. 最初の SAP アダプタ・ターゲット (target1 など) に接続します。
3. BAPI\_COMPANYCODE\_GETLIST のインバウンド WSDL ファイルを生成するには、「リモート関数モジュール」をクリックして「**Financial Accounting**」を選択し、「**0002 - Company Code**」ビジネス・オブジェクトをクリックします。
4. 2 番目の SAP アダプタ・ターゲット (target2 など) に接続します。
5. BAPI\_COMPANYCODE\_GETLIST のアウトバウンド WSDL ファイルを生成するには、「リモート関数モジュール」をクリックして「**Financial Accounting**」を選択し、「**0002 - Company Code**」ビジネス・オブジェクトをクリックします。

ターゲット、チャネルおよび WSDL ファイルの作成の詳細は、[第 4 章「構成 Oracle Application Adapter for SAP R/3」](#)を参照してください。

### 10.4.1 OSB の Service Bus アプリケーションの作成

OSB の Service Bus アプリケーションの作成の詳細は、10-2 ページの [10.1.1 項「OSB の Service Bus アプリケーションの作成」](#)を参照してください。

### 10.4.2 OSB プロセスの定義

この項では、OSB プロセスを定義する方法について説明します。ここでは、次の項目について説明します。

- [10.4.2.1 項「アウトバウンドおよびインバウンド WSDL に対するサード・パーティ・アダプタ・サービス・コンポーネントの構成」](#)
- [10.4.2.2 項「パイプラインの作成」](#)
- [10.4.2.3 項「ルーティング・ルールの構成」](#)

### 10.4.2.1 アウトバウンドおよびインバウンド WSDL に対するサード・パーティ・アダプタ・サービス・コンポーネントの構成

アウトバウンドおよびインバウンド WSDL に対してサード・パーティ・アダプタ・サービス・コンポーネントを作成するには、次のようにします。

1. 「サード・パーティ」アダプタ・コンポーネントを「コンポーネント」の「Service Bus」ペインから「外部サービス」ペインにドラッグ・アンド・ドロップします。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。
2. ビジネス・サービスに適切な名前を入力します。
3. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。  
「WSDL の選択」ダイアログが表示されます。
4. 「ファイルシステム」タブを選択し、WSDL ディレクトリでアウトバウンド WSDL ファイルを参照して選択します。
5. 「OK」をクリックします。  
「Service Bus リソースのインポート」ダイアログが表示されます。
6. 「次」をクリックします。
7. 「構成」ペインで「終了」をクリックします。  
「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。
8. 「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。  
トランスフォーメーション・チューザ・ダイアログが表示されます。
9. WSDL ディレクトリからアウトバウンド JCA プロパティ・ファイルを選択します。
10. 「OK」をクリックします。  
「ファイルのコピー」ダイアログ・ウィンドウが表示されます。
11. 「はい」をクリックします。  
プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

図 10-81 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

図 10-81 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



12. 「OK」をクリックします。

「外部サービス」にビジネス・サービスが作成されます。

13. 「サード・パーティ」アダプタ・コンポーネントを「コンポーネント」の「Service Bus」ペインから「プロキシ・サービス」ペインにドラッグ・アンド・ドロップします。

「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

14. プロキシ・サービスに適切な名前を入力します。

15. 「WSDL URL」フィールドの右側にある「既存の WSDL を検索します。」アイコンをクリックします。

「WSDL の選択」ダイアログが表示されます。

16. 「ファイルシステム」タブで、WSDL ディレクトリからインバウンド WSDL ファイルを選択します。

17. 「OK」をクリックします。

「Service Bus リソースのインポート」ダイアログが表示されます。

18. 「次」をクリックします。

19. 「構成」ペインで「終了」をクリックします。

「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

20. 「JCA ファイル」フィールドの右側にある「JCA ファイルを検索します。」アイコンをクリックします。

トランスフォーメーション・チューザ・ダイアログが表示されます。

21. WSDL ディレクトリからインバウンド JCA プロパティ・ファイルを選択します。

22. 「OK」をクリックします。

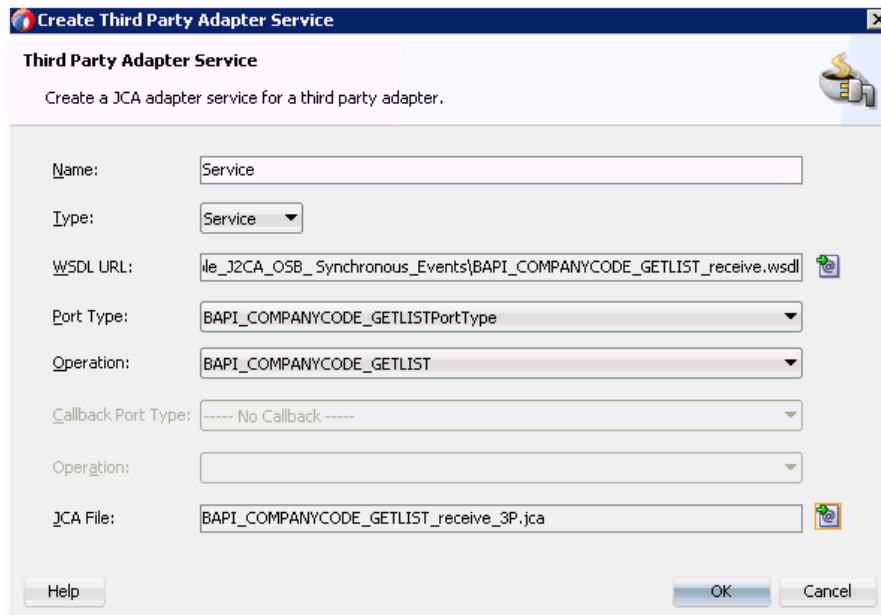
「ファイルのコピー」ダイアログ・ウィンドウが表示されます。

23. 「はい」 をクリックします。

プロジェクト・フォルダに JCA プロパティ・ファイルのコピーが作成されます。

☒ 10-82 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

図 10-82 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



24. 「OK」 をクリックします。

「プロキシ・サービス」 ペインにプロキシ・サービスが作成されます。

---

**注意：** 構成済パイプラインとマッピングした後に削除される作成されたプロキシ・サービスには、赤色のマークが表示されます。

---

### 10.4.2.2 パイプラインの作成

パイプラインを作成するには、次のようにします。

1. パイプライン・コンポーネントを「リソース」ペインから「パイプライン/分割結合」ペインにドラッグ・アンド・ドロップします。  
「パイプライン・サービスの作成」ウィンドウが表示されます。
2. パイプラインの任意の名前を入力し、「次」をクリックします。
3. 「WSDL」 ボタンを選択し、フィールドの横にある「WSDLの参照」アイコンをクリックします。  
「WSDLの選択」ウィンドウが表示されます。
4. 「アプリケーション」 タブを選択します。
5. 適切なプロジェクト・フォルダをクリックし、インバウンド具象 WSDL (ProxyService\_Name-concrete.wsdl など) を選択して「OK」 をクリックします。  
「パイプライン・サービスの作成」ウィンドウに戻ります。
6. 「プロキシ・サービスとして公開」のチェック・ボックスの選択を解除します。

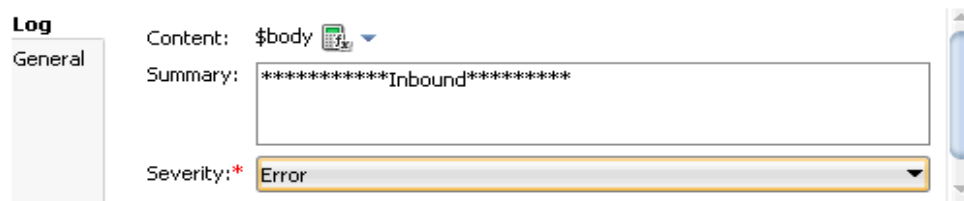
7. 「終了」をクリックします。  
プロキシ・サービス作成されます。
8. プロキシ・サービスとパイプラインを接続してから、変更を保存します。

### 10.4.2.3 ルーティング・ルール構成

ルーティング・ルールを構成するには、次のようにします。

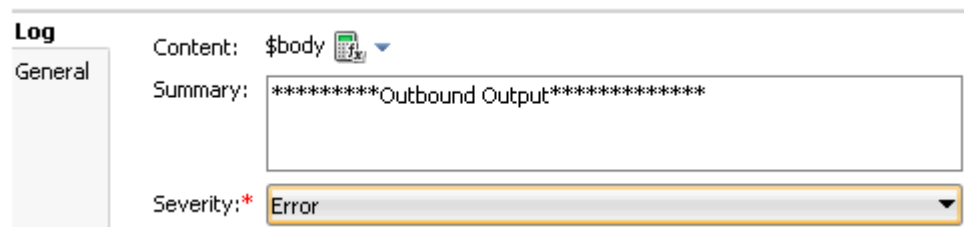
1. パイプラインとビジネス・サービスを接続します。
2. 「パイプライン/分割結合」ペインで、パイプラインをダブルクリックします。  
パイプラインの構成ページが表示されます。
3. 「ログ」ノードを「コンポーネント」の「レポート」ペインから「リクエスト・アクション」の下領域にドラッグ・アンド・ドロップします。
4. 「ログ・プロパティ」ペインで、「コンテンツ」フィールドの <expression> の近くにあるアイコンをクリックします。
5. 「XQuery 式ビルダー」ページで、「式」フィールドの下領域のテキスト領域に \$body と入力し、「OK」をクリックします。
6. 図 10-83 に示すように、「サマリー」フィールドに必要な任意のテキスト・メッセージを入力し(オプション)、「ログ」領域の下領域の「重大度」レベルを示すドロップダウン・リストで「エラー」を選択します。

図 10-83 「重大度」レベルを示すドロップダウン・リスト



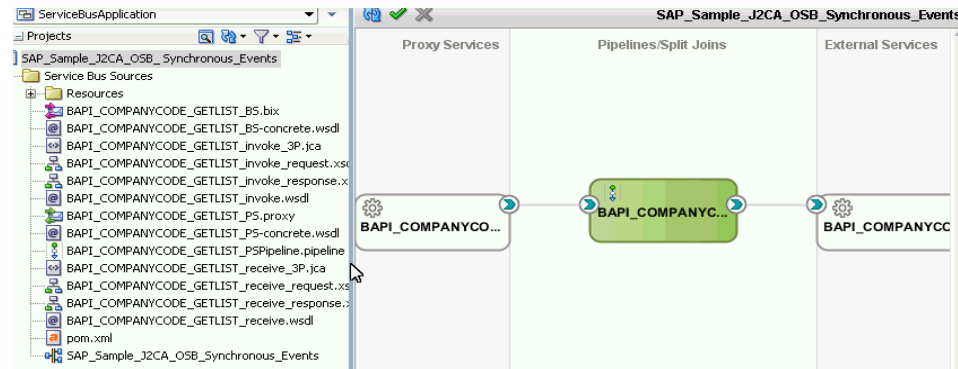
7. 「ログ」ノードを「コンポーネント」の「レポート」ペインから「レスポンス・アクション」の下領域にドラッグ・アンド・ドロップします。
8. 「ログ・プロパティ」ペインで、「コンテンツ」フィールドの <expression> の近くにあるアイコンをクリックします。
9. 「XQuery 式ビルダー」ページで、「式」フィールドの下領域のテキスト領域に \$body と入力し、「OK」をクリックします。
10. 図 10-84 に示すように、「サマリー」フィールドに必要な任意のテキスト・メッセージを入力し(オプション)、「ログ」領域の下領域の「重大度」レベルを示すドロップダウン・リストで「エラー」を選択します。

図 10-84 「重大度」レベルを示すドロップダウン・リスト



11. 図 10-85 に示すように、パイプラインの構成ページを保存して閉じます。

図 10-85 パイプラインの構成ページ



これで、OSB プロセスをデプロイする準備ができました。

### 10.4.3 OSB プロセスのデプロイ

OSB プロセスのデプロイの詳細は、10-17 ページの 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」を参照してください。

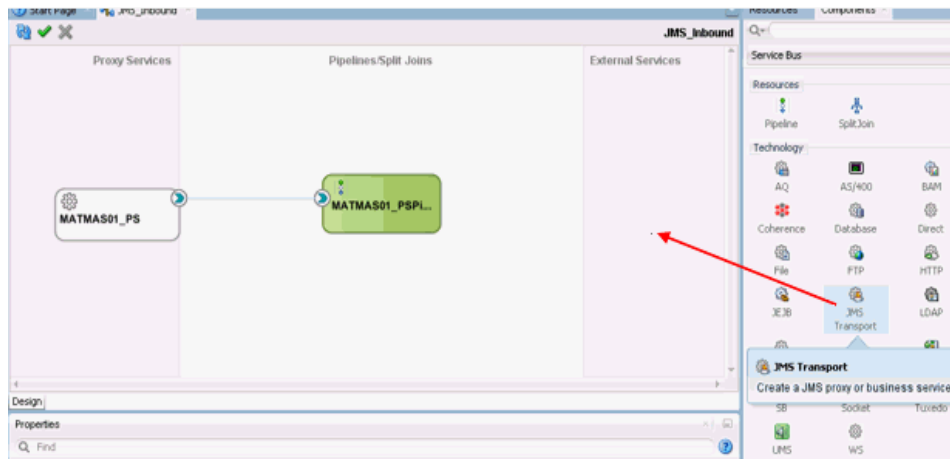
## 10.5 JDeveloper を使用した JMS インバウンド・プロセスの構成 (J2CA 構成)

この項では、Oracle JDeveloper を使用して、J2CA 構成用に SAP R/3 システムに対する JMS インバウンド・プロセスを構成する方法について説明します。

1. JMS プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-42 ページの 6.5.1 項「イベント統合のための WSDL の生成」を参照してください。
2. Oracle JDeveloper を開始し、OSB に Service Bus アプリケーションを作成します。詳細は、10-2 ページの 10.1.1 項「OSB の Service Bus アプリケーションの作成」を参照してください。
3. サード・パーティ・アダプタ・サービス・コンポーネントを作成します。詳細は、7-13 ページの 7.3.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」を参照してください。
4. JCA バインディング・ファイルから、プロキシ・サービスをパイプラインとともに作成します。詳細は、10-26 ページの 10.2.2.2 項「パイプラインの作成」を参照してください。

5. JMS トランスポート・ビジネス・サービスを作成し、次のようにします。
  - a. 図 10-86 に示すように、「JMS トランスポート」コンポーネントを「コンポーネント」の「テクノロジー」ペインから「外部サービス」ペインにドラッグ・アンド・ドロップします。

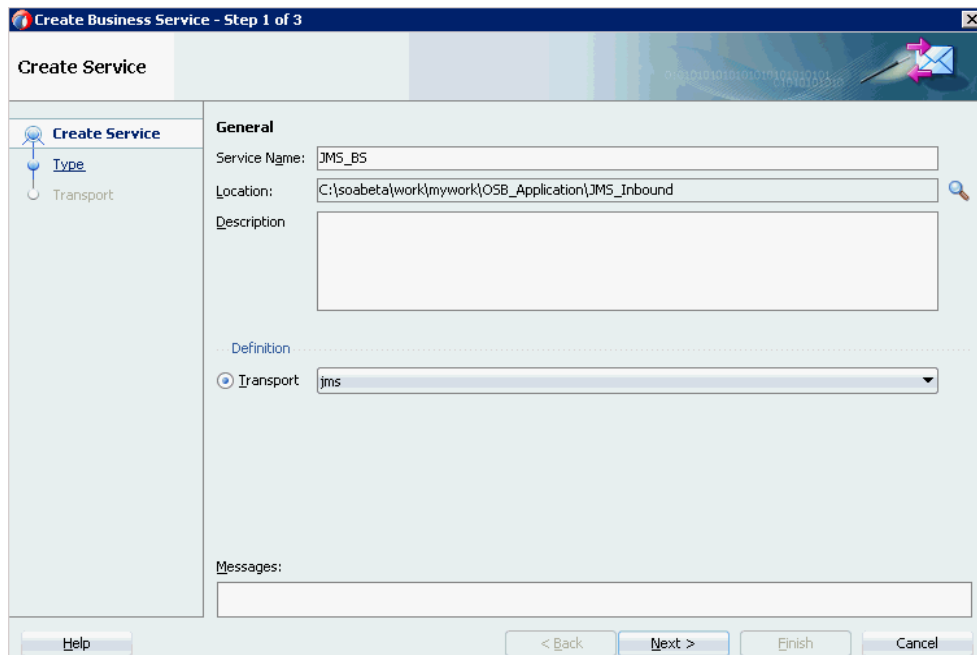
図 10-86 JMS トランスポート・コンポーネント



「ビジネス・サービスの作成」ダイアログが表示されます。

- b. 図 10-87 に示すように、「サービス名」フィールドにビジネス・サービスに使用する任意の名前 (JMS\_BS など) を入力し、「次」をクリックします。

図 10-87 「サービスの作成」ペイン



- c. 表示される「タイプ」ウィンドウで「任意の XML」を選択し、「次」をクリックします。  
「トランスポート」ウィンドウが表示されます。

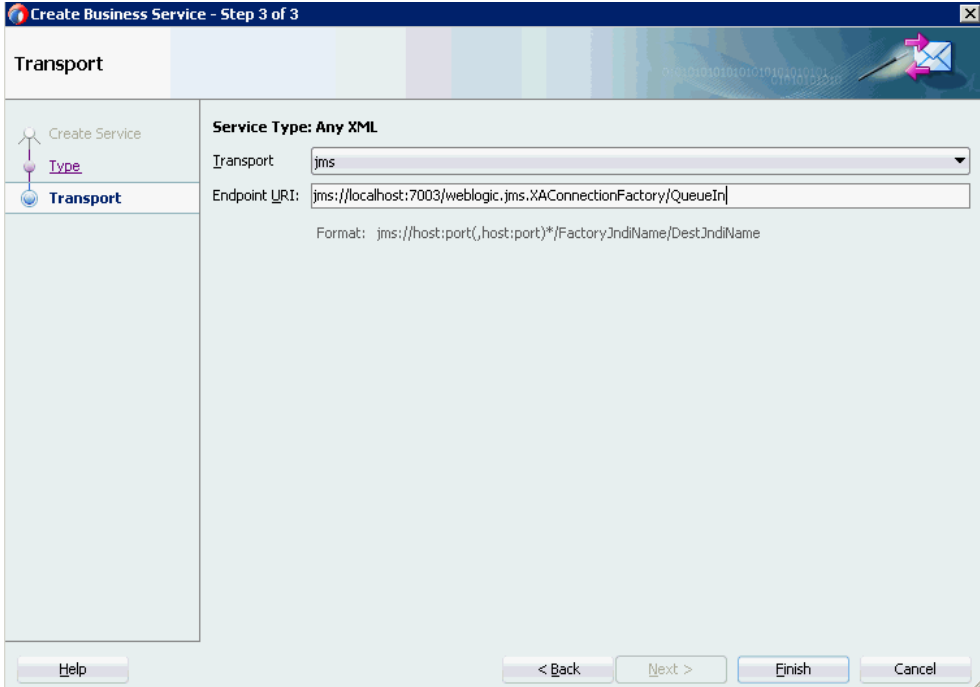
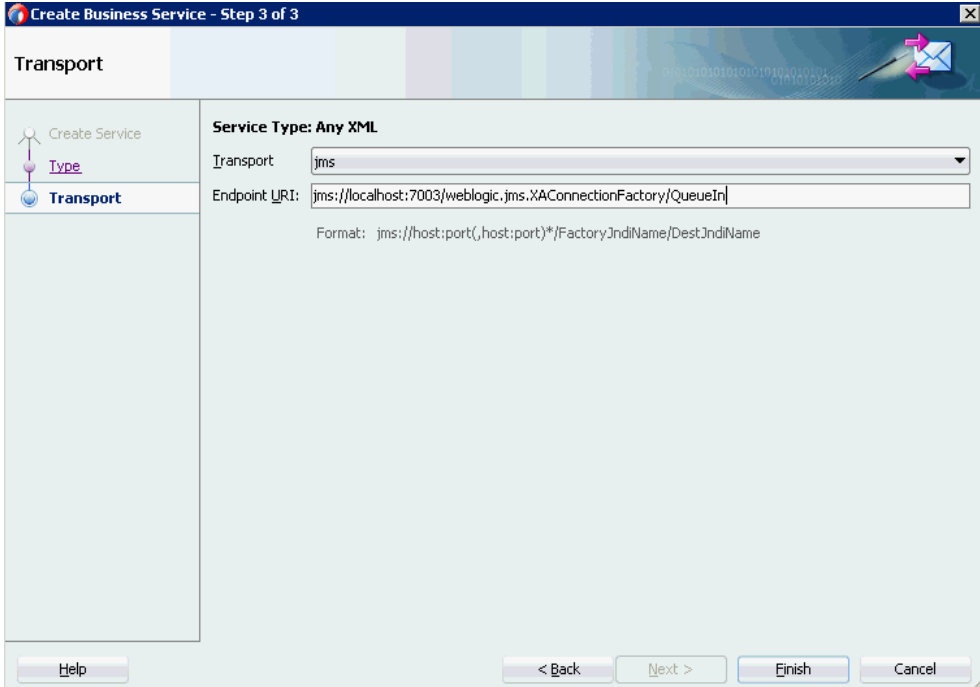
- d.  10-88 に示すように、「エンドポイント URI」フィールドの DestJndiName を QueueIn に置き換えて適切なホスト名とポート番号に変更し (jms://localhost:7003/weblogic.jms.XAConnectionFactory/QueueIn など)、「終了」をクリックします。

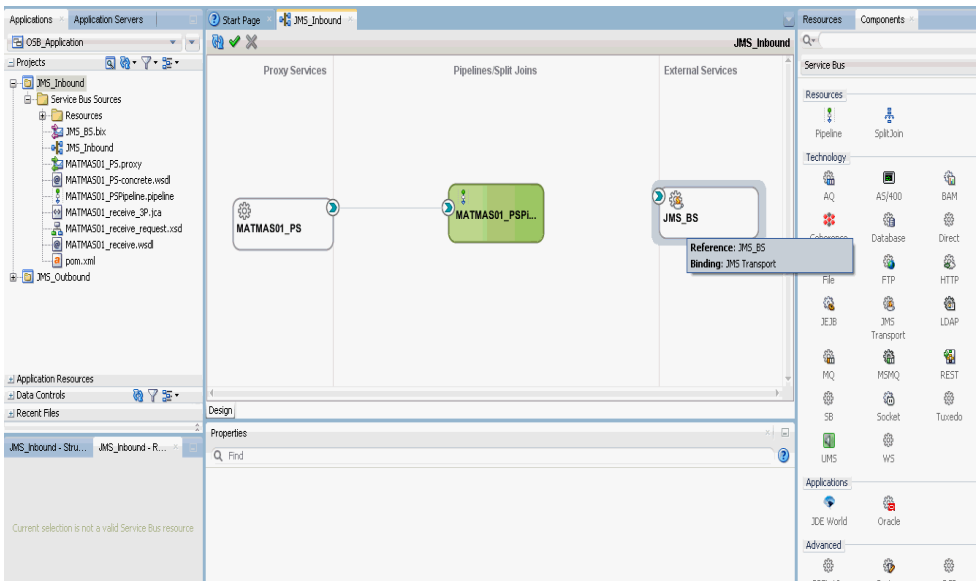
図 10-88 「トランスポート」 ウィンドウ



JMS ビジネス・サービスが作成されて表示されます。

- e.  10-89 に示すように、「JMS\_BS」をダブルクリックします。

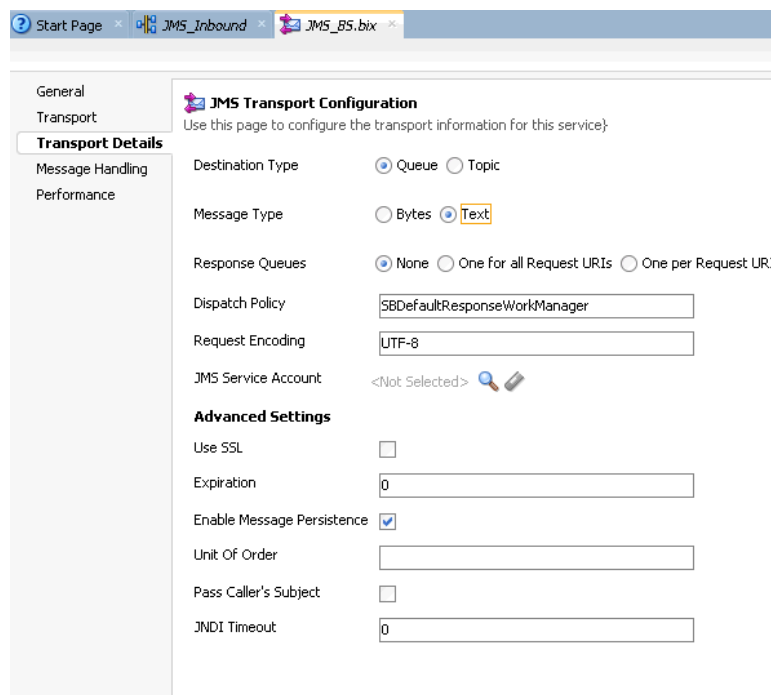
図 10-89 JMS ビジネス・サービス





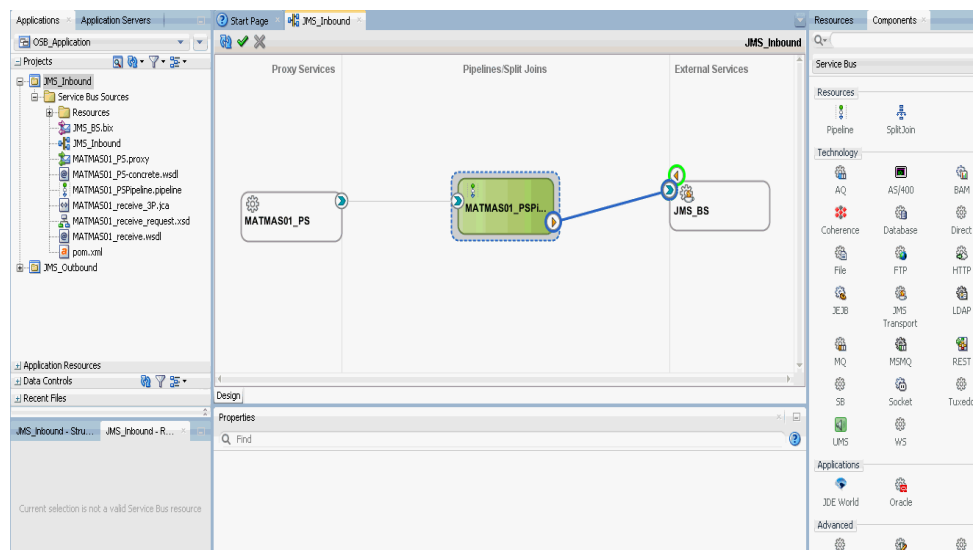
- f. 図 10-90 に示すように、表示されるビジネス・サービスの構成ページの「トランスポートの詳細」タブで、次のパラメータを指定します。

図 10-90 JMS トランスポート構成



- g. 「宛先タイプ」セクションで、「キュー」を選択します。
- h. 「メッセージ・タイプ」セクションで、「テキスト」を選択します。
6. ビジネス・サービスの構成ページを保存して閉じます。
7. 図 10-91 に示すように、パイプライン (xxxx\_PSPipeline など) と JMS ビジネス・サービス (JMS\_BS など) との間に接続を作成します。

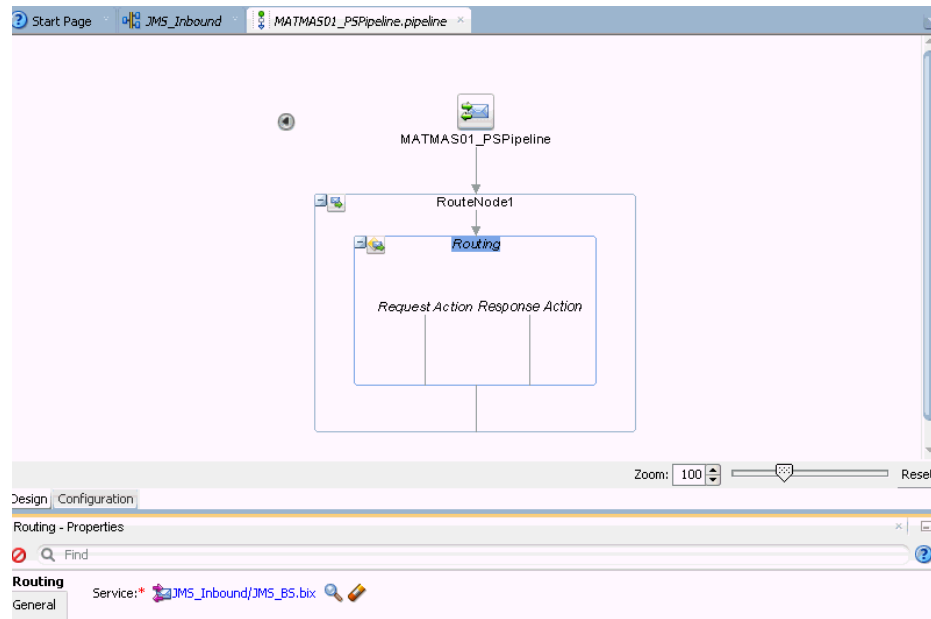
図 10-91 「構成」ページ



- パイプラインをダブルクリックします。

図 10-92 に示すように、パイプラインの構成ページが表示されます。

図 10-92 パイプラインの構成

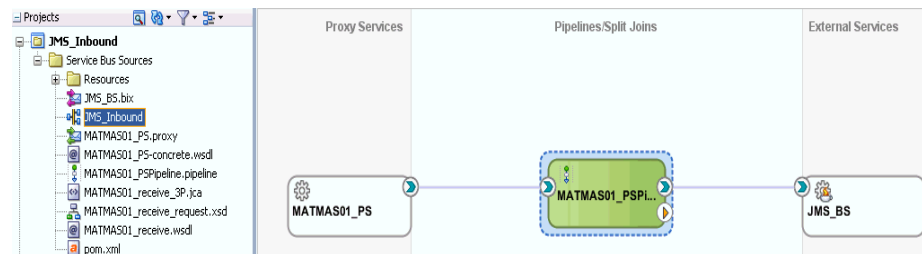


- 詳細が正しく構成されていることを確認し、パイプラインの構成ページを保存して閉じます。

コンポジット・エディタ・ウィンドウに戻ります。

- 図 10-93 に示すように、メニュー・バーの「すべて保存」をクリックして、OSB JMS プロセスを保存します。

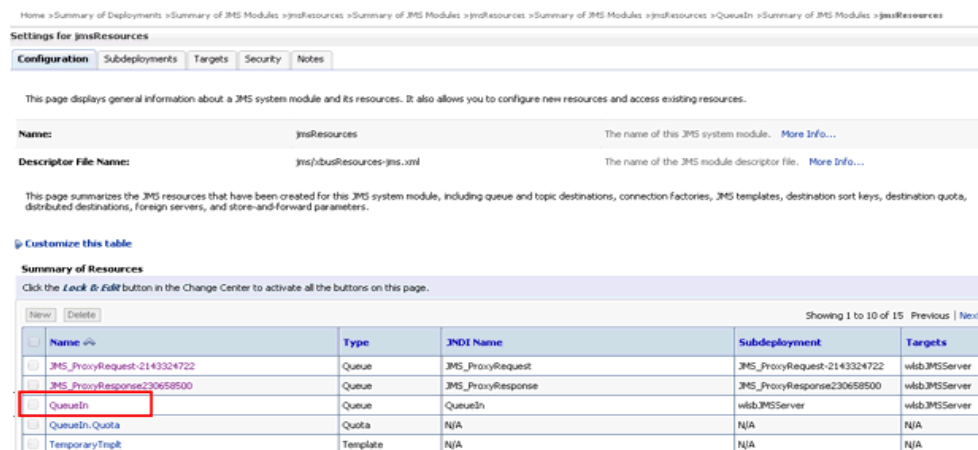
図 10-93 「すべて保存」アイコン



- OSB JMS インバウンド・プロセスをデプロイします。詳細は、10-32 ページの 10.2.3 項「OSB インバウンド・プロセスのデプロイ」を参照してください。
- プロセスが正常にデプロイされたら、イベント・メッセージをトリガーします。詳細は、6-60 ページの 6.5.5 項「SAP R/3 でのイベントのトリガー」を参照してください。
- Oracle WLS コンソールにログオンします。
- Oracle WLS コンソールで「サービス」を展開し、「メッセージング」をクリックして、「JMS モジュール」を選択してから「jmsResources」をクリックします。

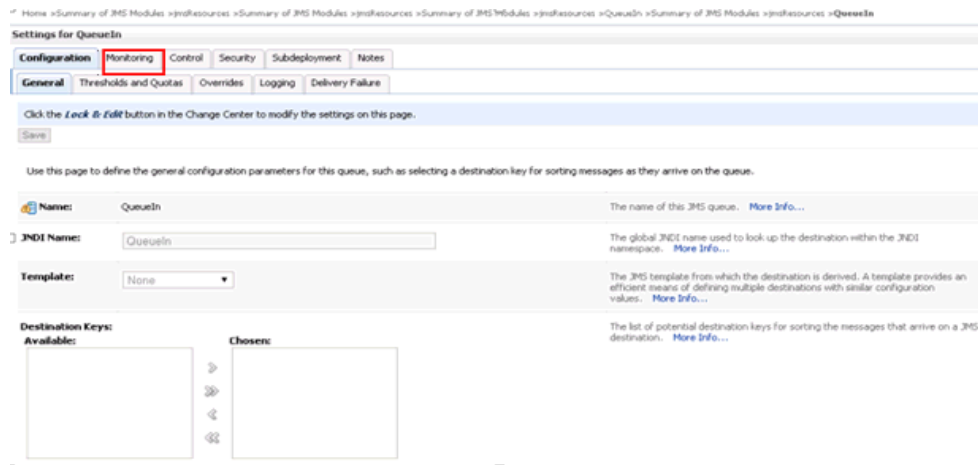
15. 図 10-94 に示すように、適切なレスポンス・リンク (QueueIn など) をクリックします。

図 10-94 QueueIn レスポンス・リンク



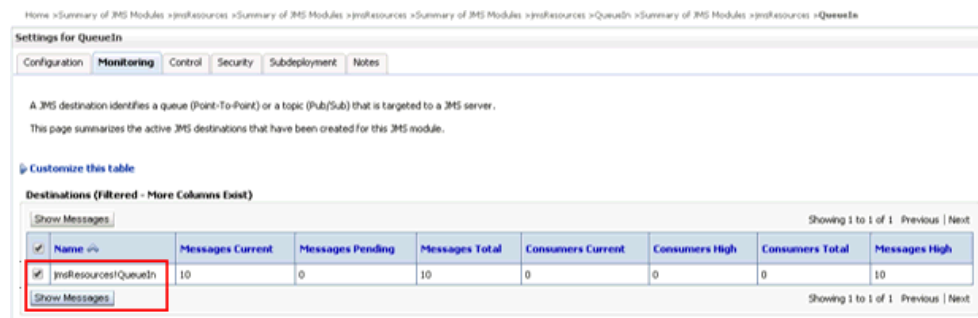
16. 図 10-95 に示すように、「モニタリング」タブをクリックします。

図 10-95 「モニタリング」タブ



17. 図 10-96 に示すように、チェック・ボックスを選択して「メッセージの表示」ボタンをクリックします。

図 10-96 「メッセージの表示」ボタン



18. 図 10-97 に示すように、適切な日時を含む ID リンクをクリックします。

図 10-97 「JMS メッセージのサマリー」 ウィンドウ

Summary of JMS Messages

This page summarizes the available messages for a stand-alone queue, a distributed queue, or a topic durable subscriber. Use this page to view message details, create new messages, delete selected messages, move messages to another destination, export message contents in XML format to another file, import XML formatted message contents from another file, or drain all the messages from a destination.

Click on a message to view its contents.

Message Selector:

[Customize this table](#)

JMS Messages (filtered - More Columns Exist)

New Delete Move Import Export Showing 1 to 10 of 10 Previous Next

ID	CorrId	Time Stamp	State String	JMS Delivery Mode	Message Size
<input type="checkbox"/> ID:<785570.1400633930707.0>		Tue May 20 17:58:50 PDT 2014	visible	Persistent	13320
<input type="checkbox"/> ID:<785570.1400633930774.0>		Tue May 20 17:58:50 PDT 2014	visible	Persistent	4473
<input type="checkbox"/> ID:<785570.1400633930794.0>		Tue May 20 17:58:50 PDT 2014	visible	Persistent	12961
<input type="checkbox"/> ID:<785570.1400633932186.0>		Tue May 20 17:58:52 PDT 2014	visible	Persistent	33583
<input type="checkbox"/> ID:<785570.1400633932958.0>		Tue May 20 17:58:52 PDT 2014	visible	Persistent	33548
<input type="checkbox"/> ID:<785570.1400633933037.0>		Tue May 20 17:58:53 PDT 2014	visible	Persistent	25156
<input type="checkbox"/> ID:<785570.1400633933871.0>		Tue May 20 17:58:53 PDT 2014	visible	Persistent	18775
<input type="checkbox"/> ID:<785570.1400633934379.0>		Tue May 20 17:58:54 PDT 2014	visible	Persistent	16671
<input type="checkbox"/> ID:<785570.1400633934420.0>		Tue May 20 17:58:54 PDT 2014	visible	Persistent	14488
<input type="checkbox"/> ID:<785570.1400633935250.0>		Tue May 20 17:58:55 PDT 2014	visible	Persistent	5684

New Delete Move Import Export Showing 1 to 10 of 10 Previous Next

図 10-98 に示すように、「テキスト」フィールドの下にレスポンス・ドキュメントが表示されます。

図 10-98 レスポンス・ドキュメント

Home > Summary of JMS Modules > jmsResources > Summary of JMS Modules > jmsResources > QueueIn > Summary of JMS Modules > jmsResources > QueueIn > Summary of JMS Messages > ID:<785570.1400633930707.0>;jmsResources!QueueIn:0

JMS Message Detail

Message ID: ID:<785570.1400633930707.0> Delivery Mode: Persistent

Type: (No value specified) Correlation ID: (No value specified)

Timestamp: Tue May 20 17:58:50 PDT 2014 Expiration: (No value specified)

Priority: 4 Redelivered: false

Delivery Time: (No value specified) Redelivery Limit: -1

Properties:

key	value	type
JMSXDeliveryCount	0	java.lang.Integer

Text:

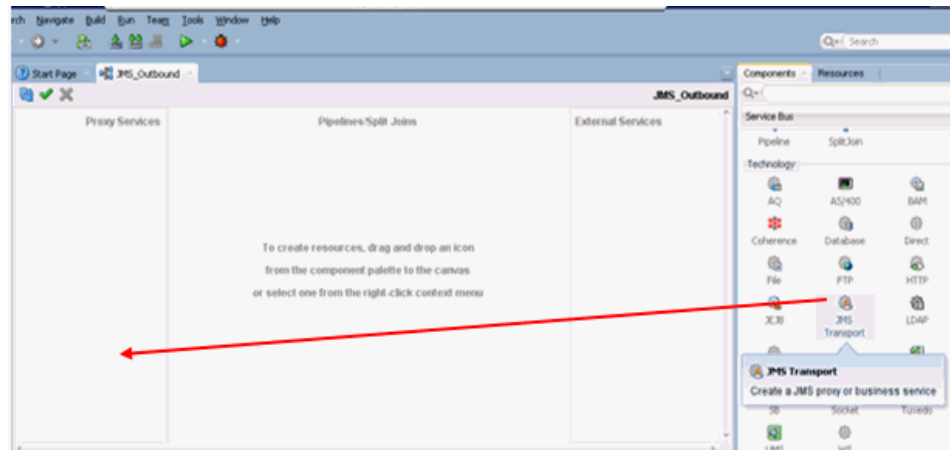
```
<?xml version="1.0" encoding="UTF-8"?>
<MATHAS01 tid="AC1301D72A27537B3D4600B9" xmlns="urn:sap-com:document:sap:idoc">
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT>800</MANDT>
      <DOCNUM>000000005013447</DOCNUM>
    </EDI_DC40 SEGMENT="1">
  </IDOC BEGIN="1">
</MATHAS01 tid="AC1301D72A27537B3D4600B9" xmlns="urn:sap-com:document:sap:idoc">
```

## 10.6 JDeveloper を使用した JMS アウトバウンド・プロセスの構成 (J2CA 構成)

この項では、Oracle JDeveloper を使用して、SAP R/3 システムへの JMS アウトバウンド・プロセスを構成する方法 (J2CA 構成) について説明します。

1. JMS プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-9 ページの 6.4.1 項「リクエスト/レスポンス・サービスの WSDL の生成」を参照してください。
2. Oracle JDeveloper を開始し、OSB に Service Bus アプリケーションを作成します。詳細は、10-2 ページの 10.1.1 項「OSB の Service Bus アプリケーションの作成」を参照してください。
3. サード・パーティ・アダプタ・サービス・コンポーネントを作成します。詳細は、7-13 ページの 7.3.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」を参照してください。
4. JCA バインディング・ファイルから、WSDL ベースのビジネス・サービスを作成します。詳細は、10-8 ページの 10.1.2.2 項「ファイル・トランスポート・タイプ・ビジネス・サービスの構成」を参照してください。
5. パイプラインを含む JMS プロキシ・サービスを作成し、次のようにします。
  - a. 図 10-99 に示すように、「JMS トランスポート」コンポーネントを「コンポーネント」の「テクノロジー」ペインから「プロキシ・サービス」ペインにドラッグ・アンド・ドロップします。

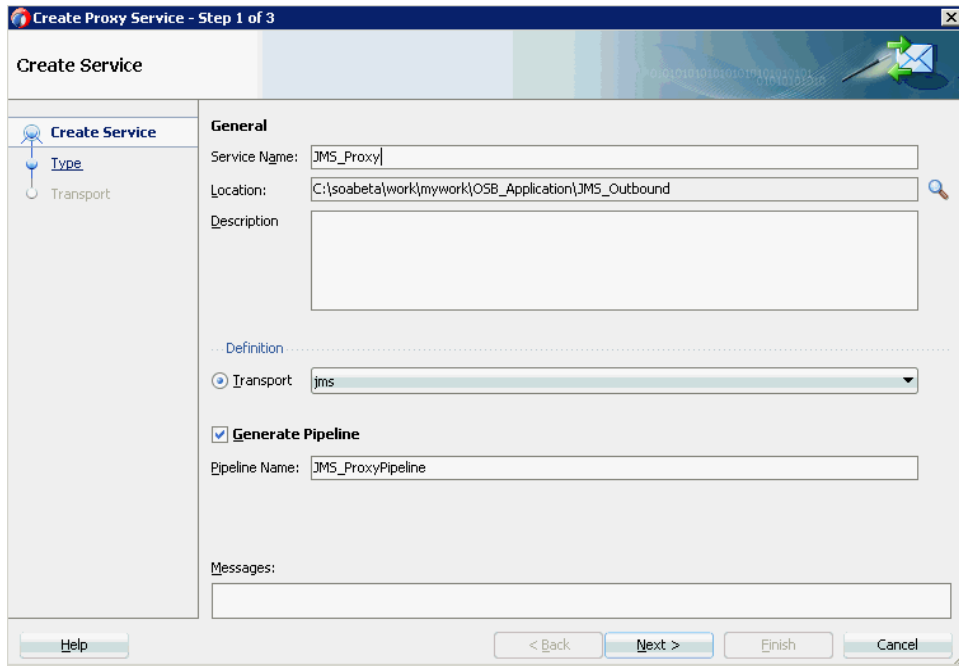
図 10-99 JMS トランスポート・コンポーネント



「ビジネス・サービスの作成」ダイアログが表示されます。

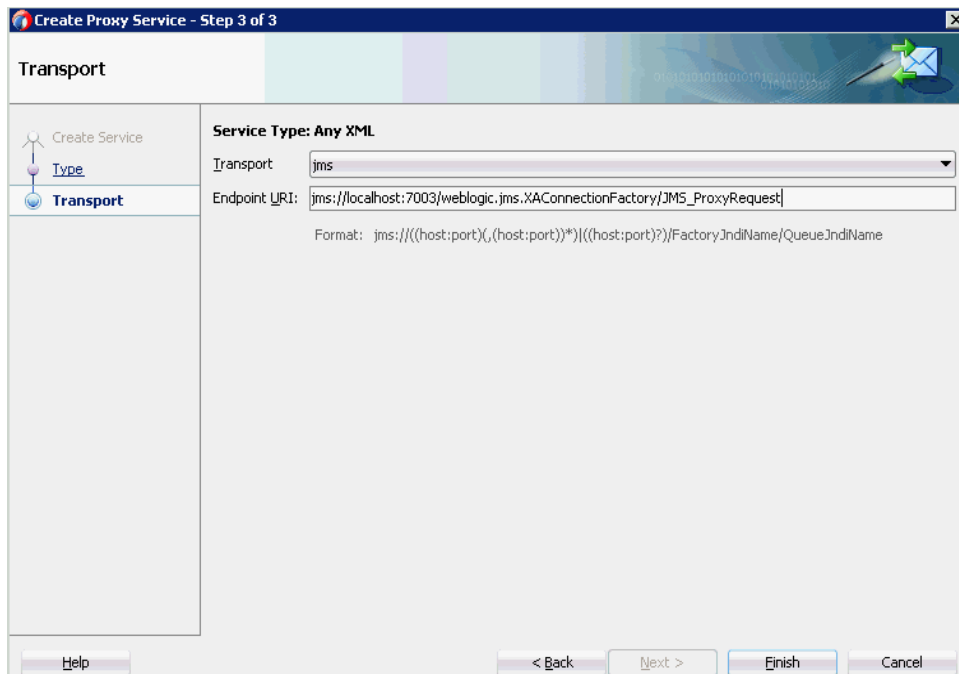
- b. 「サービス名」フィールドに、プロキシ・サービスに使用する任意の名前 (JMS\_Proxy など) を入力します。デフォルトでは、「パイプラインの生成」が選択されています。
- c. 図 10-100 に示すように、「次」をクリックします。

図 10-100 「プロキシ・サービスの作成」 ペイン



- d. 表示される「タイプ」ウィンドウで「任意の XML」を選択し、「次」をクリックします。  
「トランスポート」ウィンドウが表示されます。
- e. 図 10-101 に示すように、「エンドポイント URI」フィールドを置き換えて適切なホスト名とポート番号に変更し (jms://localhost:7003/weblogic.jms.XAConnectionFactory/JMS\_ProxyRequest など)、「終了」をクリックします。

図 10-101 「トランスポート」ウィンドウ



JMS プロキシ・サービスが、パイプラインとともに作成されて表示されます。


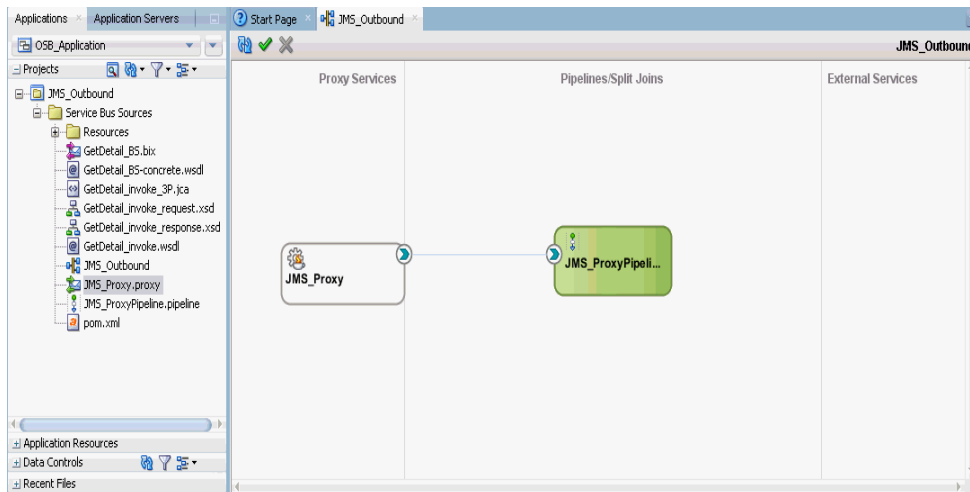
- f.  10-102 に示すように、作成されたプロキシ・サービス (JMS\_Proxy など) をダブルクリックします。

図 10-102 JMS プロキシ・サービス




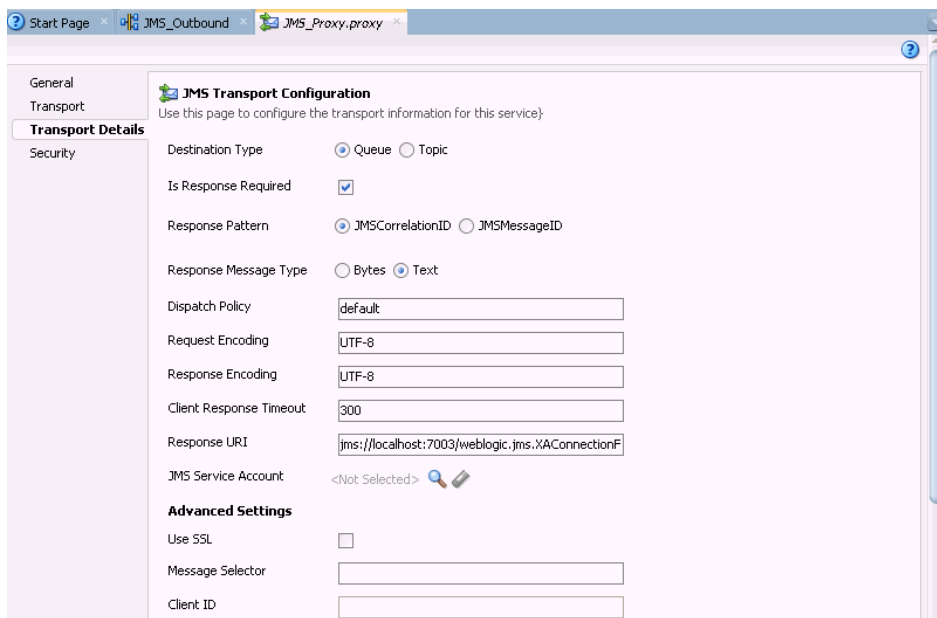
- g.  10-103 に示すように、表示されるプロキシ・サービスの構成ページの「トランスポートの詳細」タブで、次のパラメータを指定します。

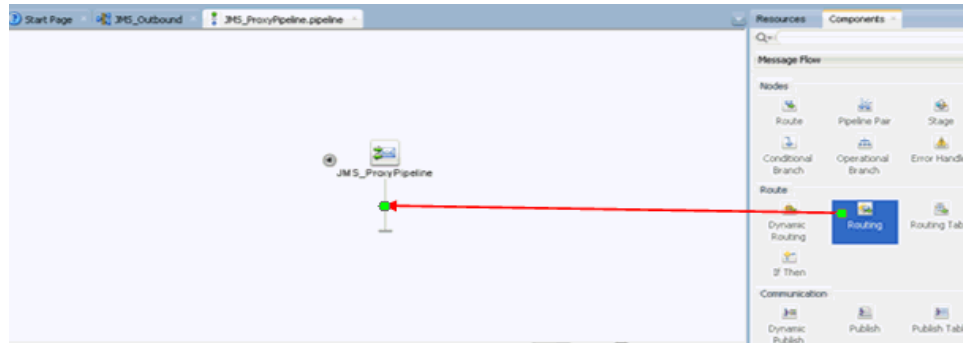
図 10-103 JMS トランスポート構成



- h. 「宛先タイプ」セクションで、「キュー」を選択します。
- i. 「レスポンスが必要」チェック・ボックスを選択します。
- j. 「レスポンス・メッセージ・タイプ」セクションで、「テキスト」を選択します。
- k. 「レスポンス URI」フィールドで、JMS トランスポート構成で使用するエンドポイント URI を指定し、Request を Response に変更します (jms://localhost:7003/weblogic.jms.XAConnectionFactory/JMS\_ProxyResponse など)。

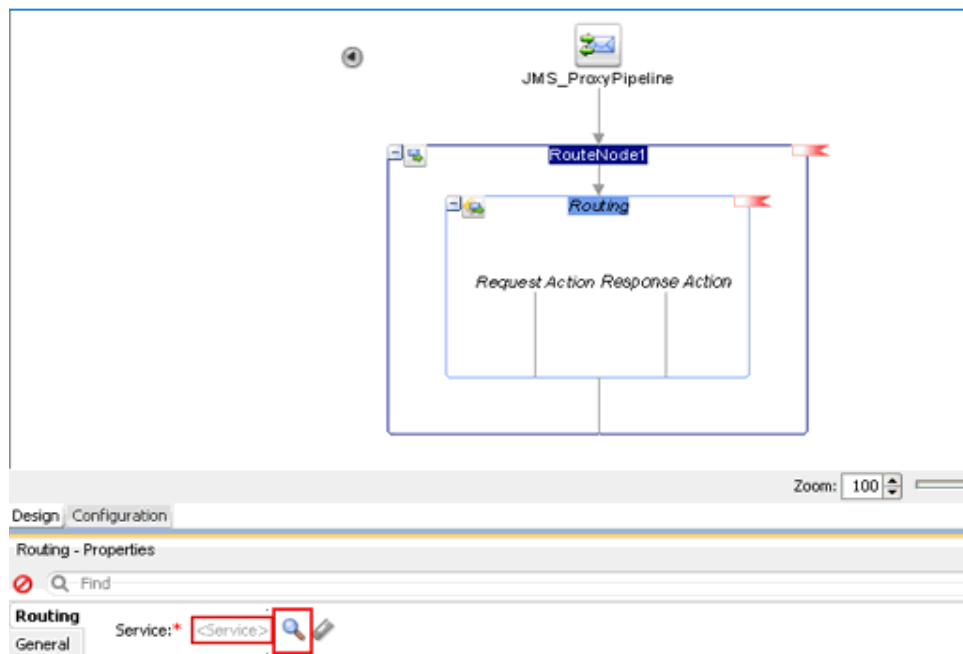
6. プロキシ・サービスの構成ページを保存して閉じます。
7. ルーティング・ルールを構成し、次の手順を続行します。
  - a. 「パイプライン/分割結合」 ペインで、パイプライン (JMS\_ProxyPipeline など) をダブルクリックします。  
パイプラインの構成ページが表示されます。
  - b. 図 10-104 に示すように、「ルーティング」 コンポーネントを「ルート」 セクションからパイプライン (JMS\_ProxyPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 10-104 ルーティング・コンポーネント



- c. 図 10-105 に示すように、パイプラインの構成ページで「ルーティング」を選択し、「ルーティング・プロパティ」ペインの「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 10-105 サービスの参照



- d. 表示される「リソース・チューザ」ウィンドウで、WSDL ベースのビジネス・サービス (xxxxx\_BS.bix など) を選択して、「OK」をクリックします。  
パイプラインの構成ページに戻ります。




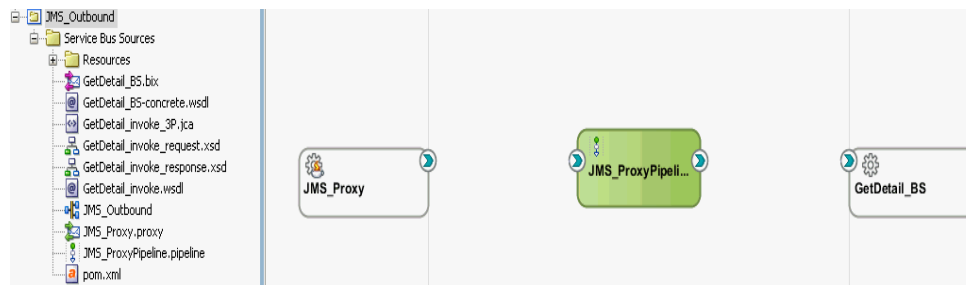
- e. パイプラインの構成ページを保存して閉じます。  
コンポジット・エディタ・ウィンドウに戻ります。
- f.  10-106 に示すように、メニュー・バーの「すべて保存」をクリックして、OSB JMS プロセスを保存します。

図 10-106 「トランスポート」 ウィンドウ




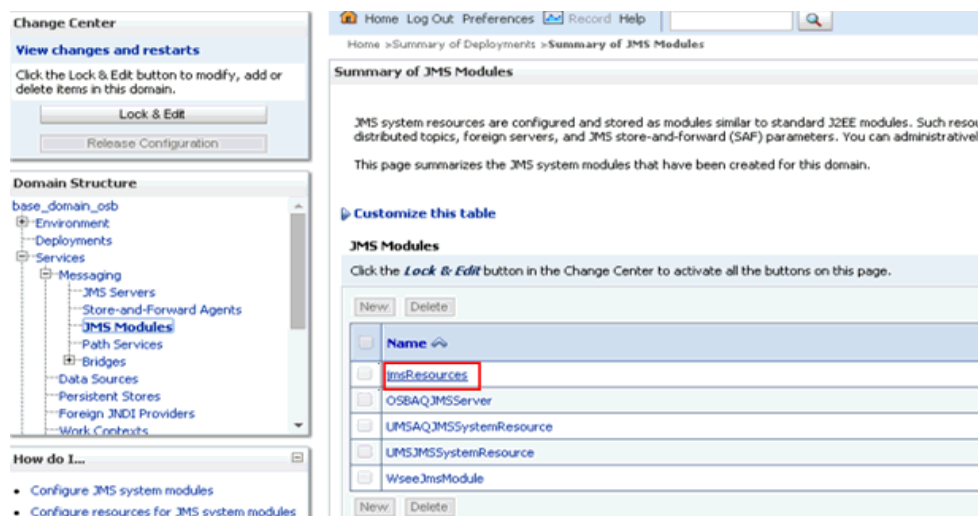
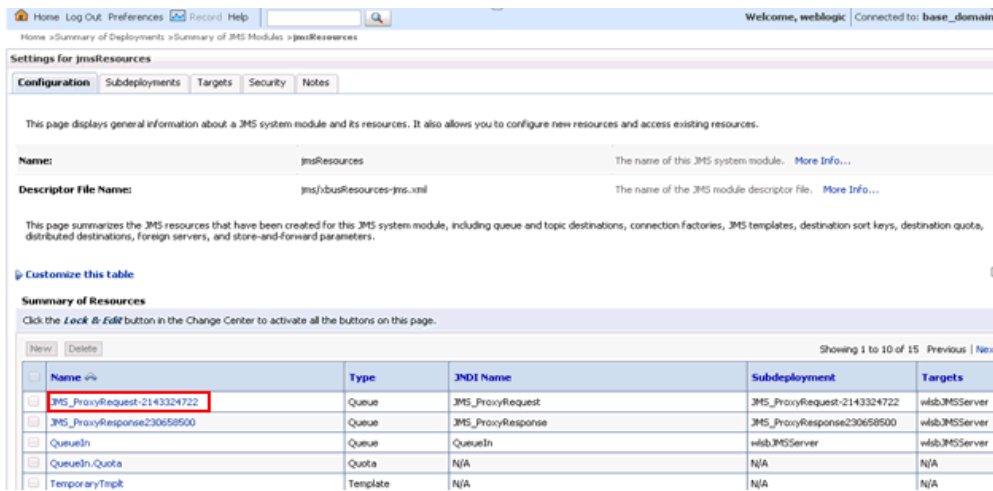
8. OSB JMS アウトバウンド・プロセスをデプロイします。詳細は、10-17 ページの 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」を参照してください。
9. プロセスが正常にデプロイされたら、Oracle WLS コンソールにログオンします。
10.  10-107 に示すように、Oracle WLS コンソールで「サービス」を展開し、「メッセージング」をクリックして、「JMS モジュール」を選択してから「jmsResources」をクリックします。

図 10-107 JMS リソース



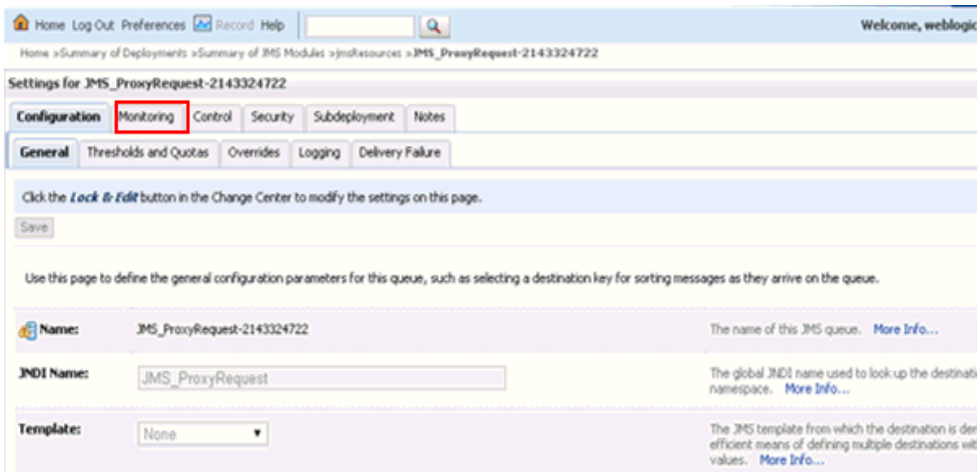
11. 図 10-108 に示すように、適切なリクエスト・リンク (JMS\_ProxyRequest など) をクリックします。

図 10-108 JMS\_ProxyRequest リンク



12. 図 10-109 に示すように、「モニタリング」タブをクリックします。

図 10-109 「モニタリング」タブ



13. 図 10-110 に示すように、チェック・ボックスを選択して「メッセージの表示」ボタンをクリックします。

図 10-110 「メッセージの表示」ボタン

Home > Summary of Deployments > Summary of JMS Modules > jmsResources > JMS\_ProxyRequest-2143324722

Settings for JMS\_ProxyRequest-2143324722

Configuration **Monitoring** Control Security Subdeployment Notes

A JMS destination identifies a queue (Point-To-Point) or a topic (Pub/Sub) that is targeted to a JMS server. This page summarizes the active JMS destinations that have been created for this JMS module.

Customize this table

Destinations (Filtered - More Columns Exist)

Show Messages Showing 1 to 1 of 1 Previous Next

<input checked="" type="checkbox"/>	Name	Messages Current	Messages Pending	Messages Total	Consumers Current	Consumers High	Consumers Total	Messages High
<input checked="" type="checkbox"/>	jmsResources!JMS_ProxyRequest-2143324722	0	0	0	16	16	16	0

Show Messages Showing 1 to 1 of 1 Previous Next

14. 図 10-111 に示すように、「新規」をクリックします。

図 10-111 JMS メッセージ

Home > Summary of Deployments > Summary of JMS Modules > jmsResources > JMS\_ProxyRequest-2143324722 > Summary of JMS Messages

Summary of JMS Messages

This page summarizes the available messages for a stand-alone queue, a distributed queue, or a topic durable subscriber. Use this page to view message details, create new messages, delete selected messages, move messages to another destination, export message contents in XML, format to another file, import XML formatted message contents from another file, or drain all the messages from a destination. Click on a message to view its contents.

Message Selector:  Apply

Customize this table

JMS Messages (Filtered - More Columns Exist)

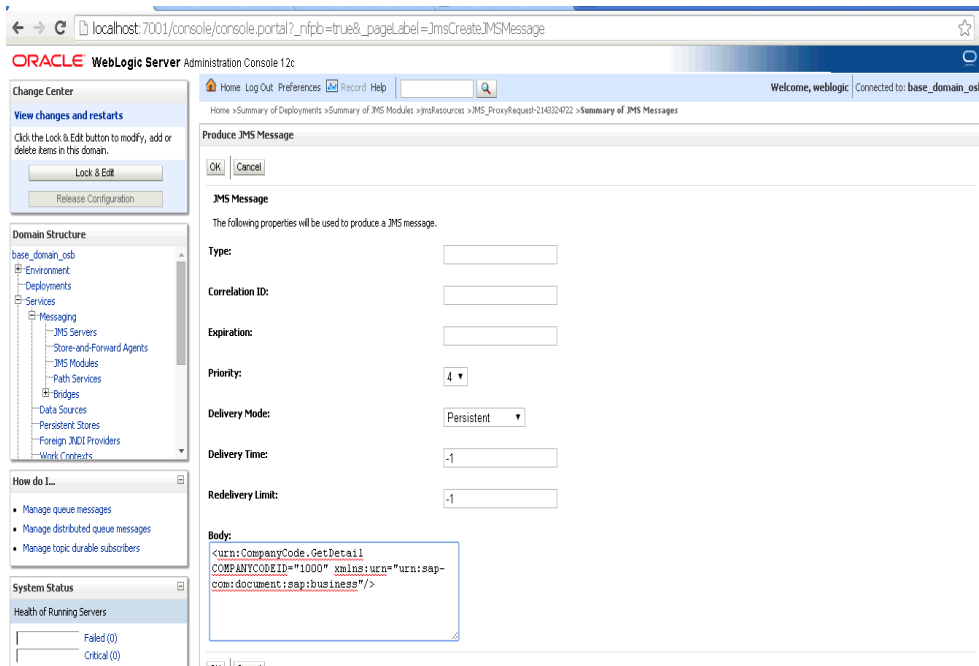
New Delete Move Import Export Showing 1 to 0 of 0 Previous Next

<input type="checkbox"/>	ID	Corrid	Time Stamp	State String	JMS Delivery Mode	Message Size
There are no items to display						

New Delete Move Import Export Showing 1 to 0 of 0 Previous Next

15. 図 10-112 に示すように、「本文」フィールドに入力ペイロードを指定し、「OK」をクリックします。

図 10-112 入力ペイロード



16. Oracle WLS コンソールで「サービス」を展開し、「メッセージング」をクリックして、「JMS モジュール」を選択してから「jmsResources」をクリックします。
17. 適切なレスポンス・リンク (JMS\_ProxyResponse など) をクリックします。
18. 「モニタリング」タブをクリックします。

19. 図 10-113 に示すように、チェック・ボックスを選択して「メッセージの表示」をクリックします。

図 10-113 宛先メッセージ

The screenshot shows the Oracle WebLogic Server Administration Console. The main content area displays the 'Settings for JMS\_ProxyResponse230658500' page. The 'Monitoring' tab is active. A 'Show Messages' button is visible. Below it, a table titled 'Destinations (Filtered - More Columns Exist)' shows the following data:

Name	Messages Current	Messages Pending	Messages Total	Consumers Current	Consumers High	Consumers Total	Messages High
jmsResources/JMS_ProxyResponse230658500	1	0	1	16	16	16	1

20. 図 10-114 に示すように、適切な日時を含む ID リンクをクリックします。

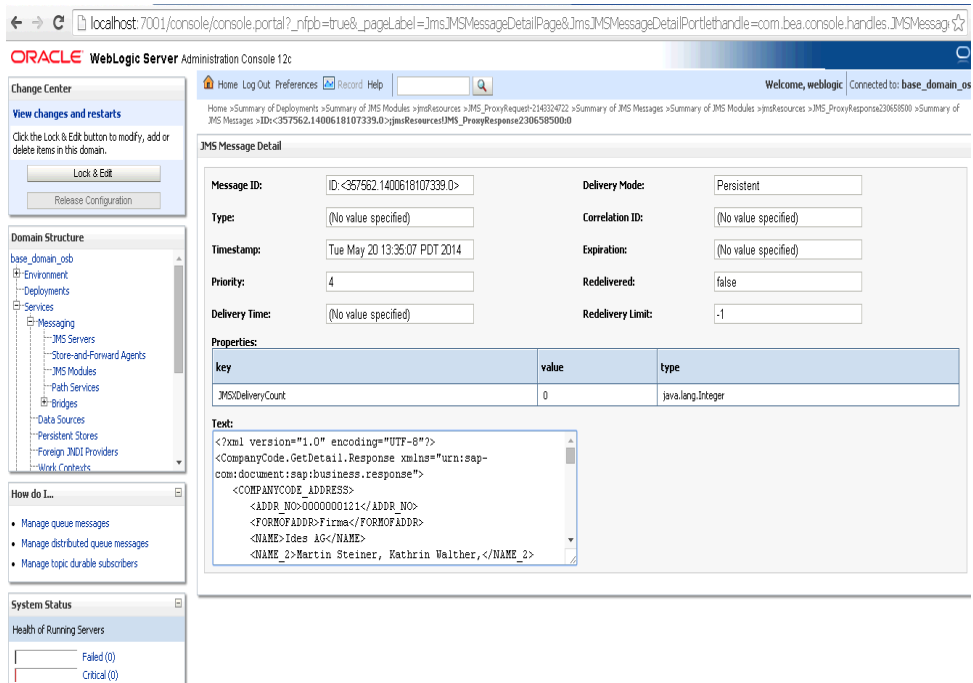
図 10-114 「JMS メッセージのサマリー」 ウィンドウ

The screenshot shows the Oracle WebLogic Server Administration Console. The main content area displays the 'Summary of JMS Messages' page. The 'Message Selector' field is empty. Below it, a table titled 'JMS Messages (Filtered - More Columns Exist)' shows the following data:

ID	Corrid	Time Stamp	State String	JMS Delivery Mode	Message Size
ID:<357562.1400618107339.0>		Tue May 20 13:35:07 PDT 2014	visible	Persistent	1914

図 10-115 に示すように、「テキスト」フィールドの下にレスポンス・ドキュメントが表示されます。

図 10-115 レスポンス・ドキュメント



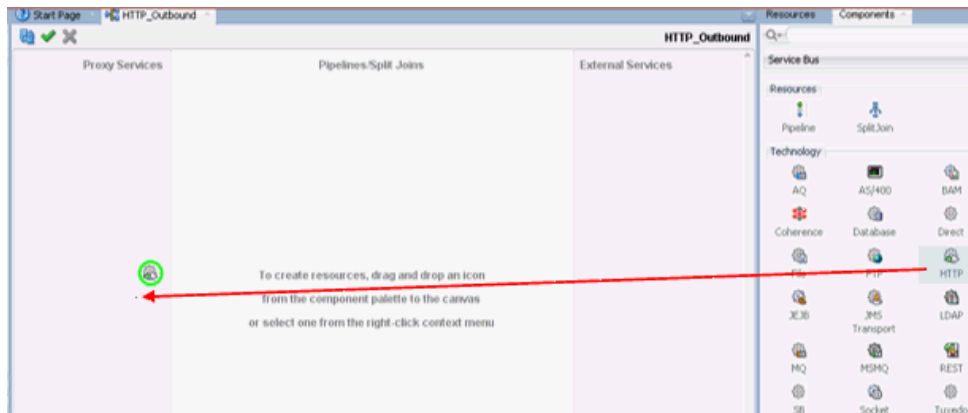
## 10.7 JDeveloper を使用した HTTP アウトバウンド・プロセスの構成 (J2CA 構成)

この項では、Oracle JDeveloper を使用して、SAP R/3 システムへの HTTP アウトバウンド・プロセスを構成する方法 (J2CA 構成) について説明します。

1. HTTP アウトバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細は、6-9 ページの 6.4.1 項「リクエスト/レスポンス・サービスの WSDL の生成」を参照してください。
2. Oracle JDeveloper を開始し、OSB に Service Bus アプリケーションを作成します。詳細は、10-2 ページの 10.1.1 項「OSB の Service Bus アプリケーションの作成」を参照してください。
3. サード・パーティ・アダプタ・サービス・コンポーネントを作成します。詳細は、7-13 ページの 7.3.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」を参照してください。

4. パイプラインを含む HTTP プロキシ・サービスを作成し、次のようにします。
  - a. 図 10-116 に示すように、「HTTP」コンポーネントを「コンポーネント」の「テクノロジー」ペインから「プロキシ・サービス」ペインにドラッグ・アンド・ドロップします。

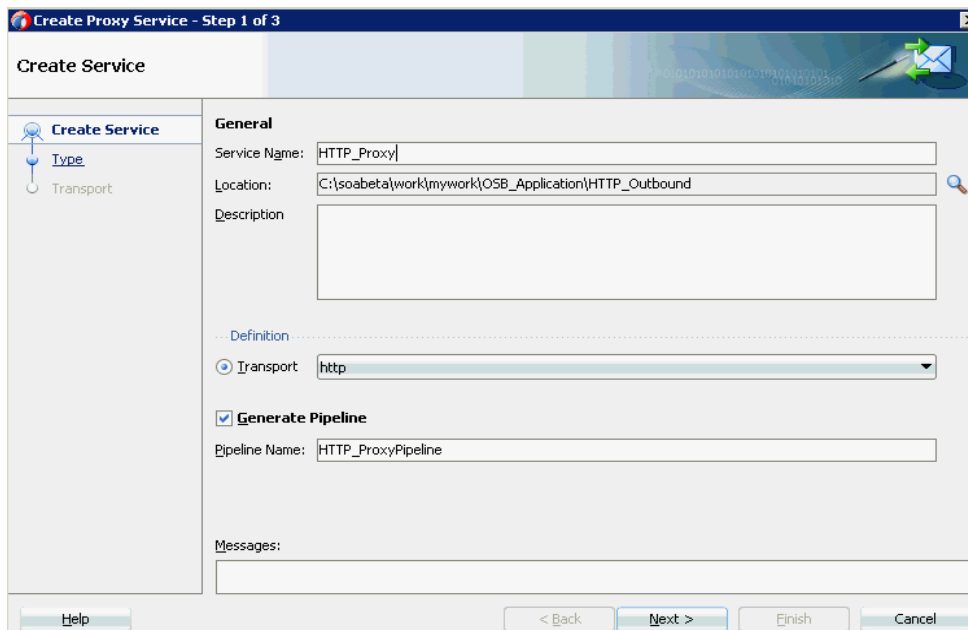
図 10-116 HTTP コンポーネント



「プロキシ・サービスの作成」ダイアログが表示されます。

- b. 「サービス名」フィールドに、プロキシ・サービスに使用する任意の名前 (HTTP\_Proxy など) を入力します。デフォルトでは、「パイプラインの生成」が選択されています。
- c. 図 10-117 に示すように、「次」をクリックします。

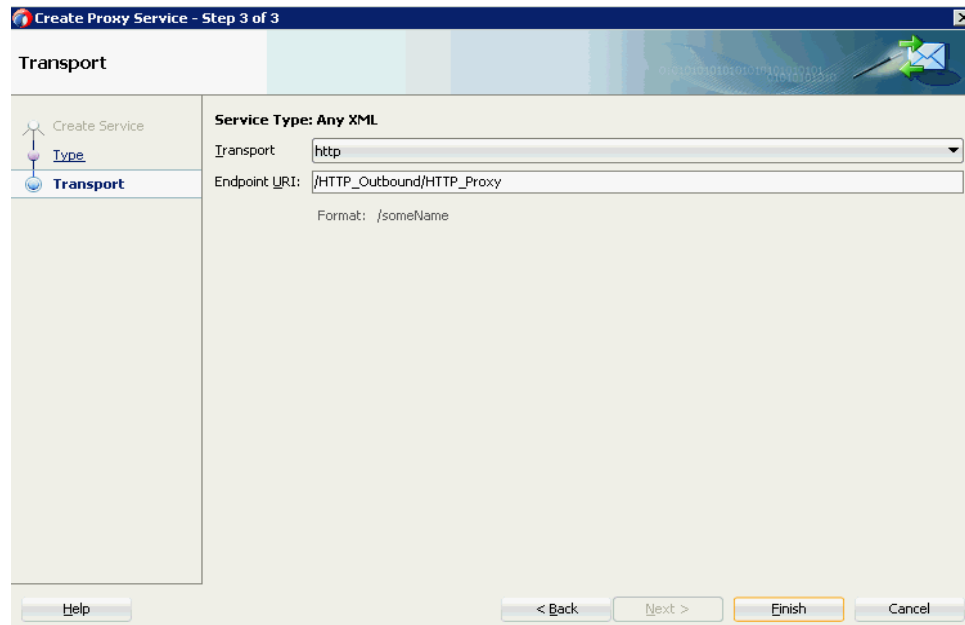
図 10-117 「プロキシ・サービスの作成」ペイン



- d. 表示される「タイプ」ウィンドウで「任意の XML」を選択し、「次」をクリックします。  
「トランスポート」ウィンドウが表示されます。

- e. 図 10-118 に示すように、デフォルト値のままにして、「終了」をクリックします。

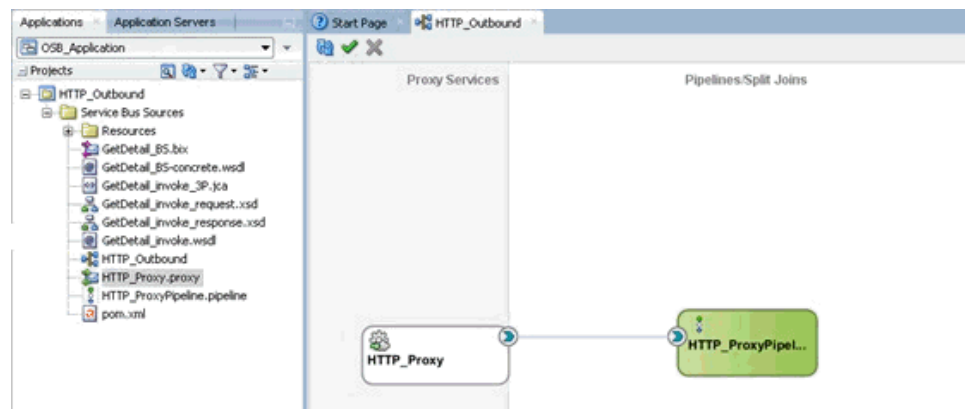
図 10-118 「トランスポート」 ウィンドウ



HTTP プロキシ・サービスが、パイプラインとともに作成されて表示されます。

- f. 図 10-119 に示すように、「パイプライン/分割結合」ペインで、作成されたパイプライン (HTTP\_ProxyPipeline など) をダブルクリックします。

図 10-119 プロキシ・サービス



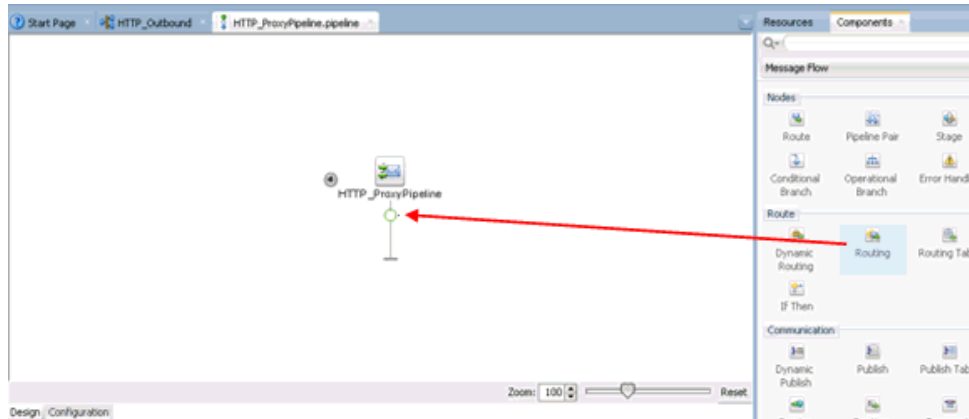
パイプラインの構成ページが表示されます。



5. ルーティング・ルールを構成し、次の手順を続行します。

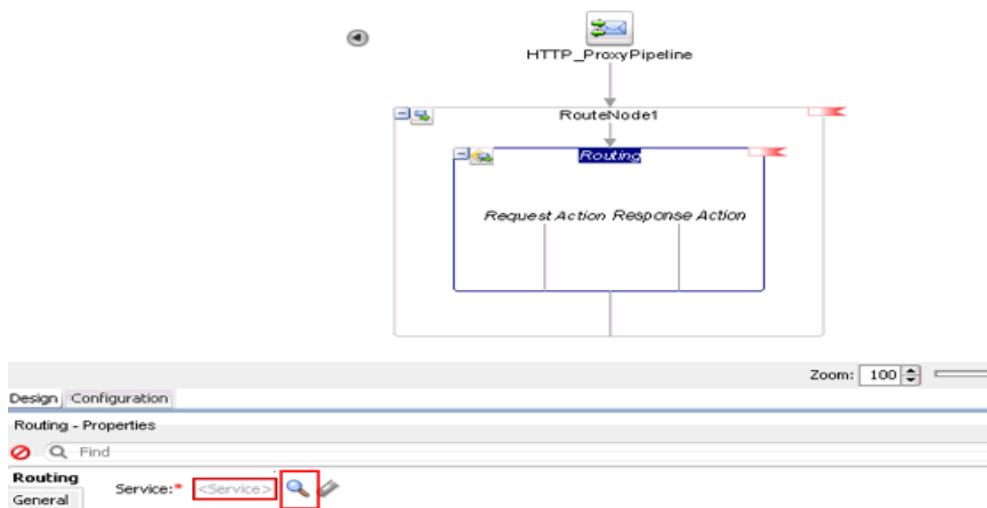
- a. 図 10-120 に示すように、「ルーティング」コンポーネントを「ルート」セクションからパイプライン (HTTP\_ProxyPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 10-120 ルーティング・コンポーネント



- b. 図 10-121 に示すように、パイプラインの構成ページで「ルーティング」を選択し、「ルーティング - プロパティ」ペインの「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 10-121 サービスの参照



- c. 表示される「リソース・チューザ」ウィンドウで、WSDL ベースのビジネス・サービス (xxxxx\_BS.bix など) を選択して、「OK」をクリックします。  
パイプラインの構成ページに戻ります。
- d. パイプラインの構成ページを保存して閉じます。  
コンポジット・エディタ・ウィンドウに戻ります。


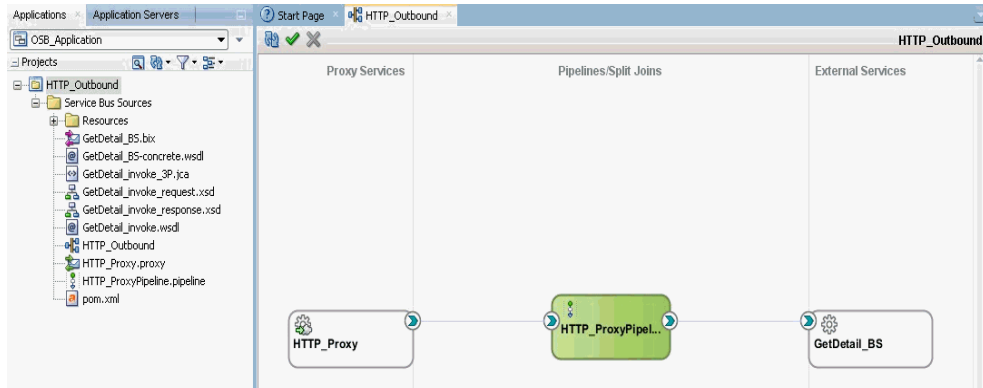
- e.  10-122 に示すように、メニュー・バーの「すべて保存」をクリックして、OSB HTTP プロセスを保存します。

図 10-122 「トランスポート」 ウィンドウ




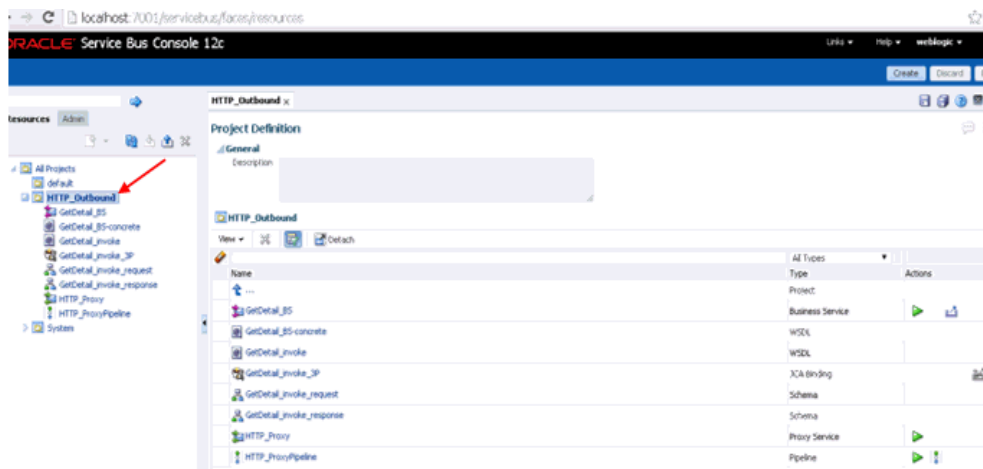
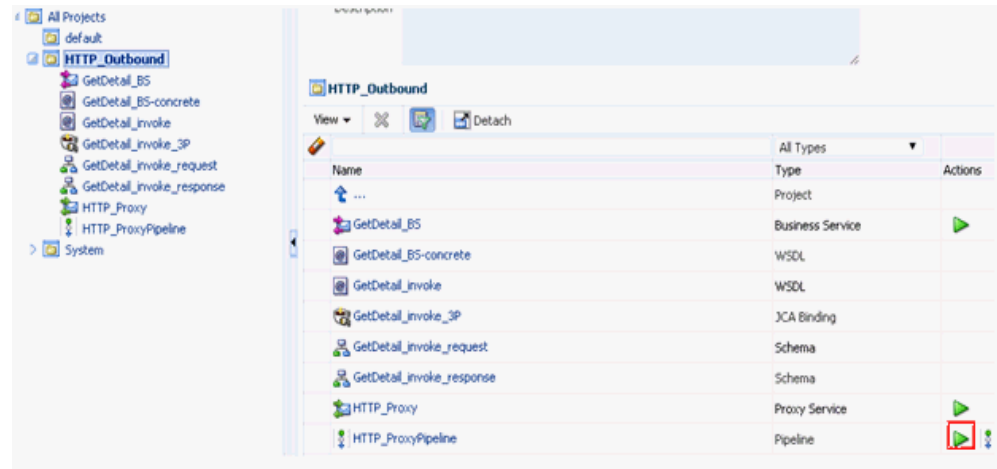
6. OSB HTTP アウトバウンド・プロセスをデプロイします。詳細は、10-17 ページの 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」を参照してください。
7. プロセスが正常にデプロイされたら、Service Bus コンソールにログオンします。
8.  10-123 に示すように、Service Bus コンソールで、デプロイされた HTTP アウトバウンド・プロジェクト (HTTP\_Outbound など) をクリックします。

図 10-123 Service Bus コンソール



9. 図 10-124 に示すように、OSB コンソールのテスト・アイコンをクリックします。

図 10-124 OSB コンソールのテスト・アイコン



10. 図 10-125 に示すように、表示される OSB コンソールのテスト・ページで、入力 XML を指定して「実行」ボタンをクリックします。

図 10-125 パイプライン・テスト

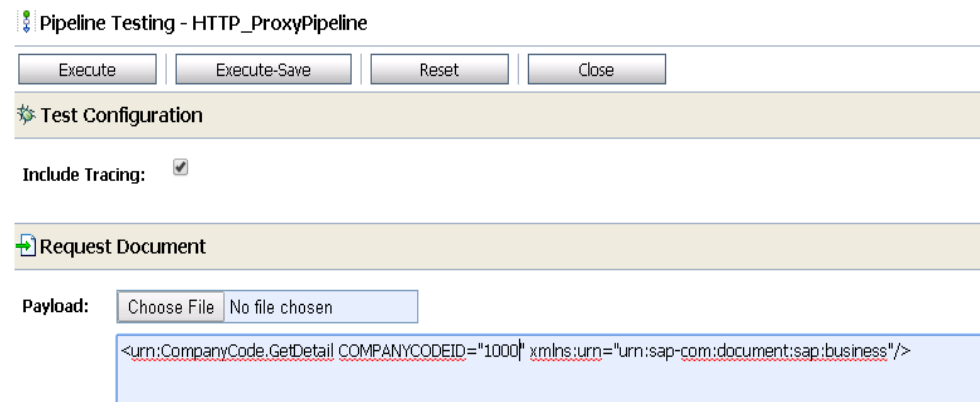
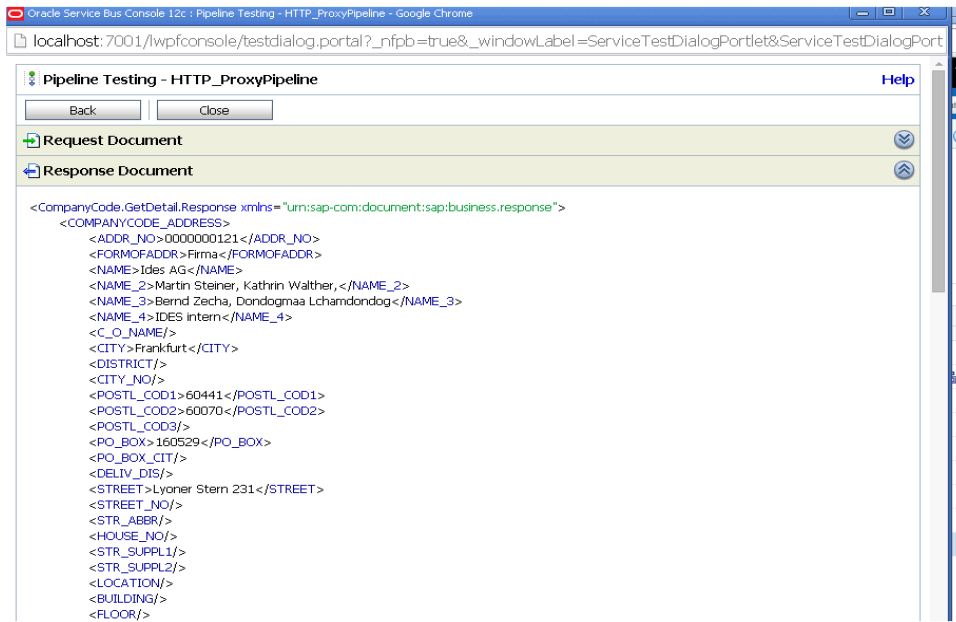


図 10-126 に示すように、表示される OSB コンソールのテスト・ページでレスポンスが受信されます。

図 10-126 レスポンス・ドキュメント



この章では、Oracle Application Adapter for SAP R/3 の主要な機能について説明します。この章は次の項で構成されています。

- 11.1 項「例外フィルタ」
- 11.2 項「Oracle SOA Suite (BPEL、メディエータまたは BPM) の資格証明マッピング」
- 11.3 項「JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング」
- 11.4 項「テキストからの Unicode 文字のフィルタリング」

## 11.1 例外フィルタ

この項では、サンプルのテスト・シナリオを使用して、Oracle Application Adapter for SAP R/3 の例外フィルタ機能を構成する方法について説明します。この項では、次の項目について説明します。

- 11.1.1 項「例外フィルタの構成」

例外フィルタは、J2CA 構成を使用するアウトバウンド・プロセスについてのみサポートされます。この機能は、BSE 構成、および J2CA 構成を使用するインバウンド・プロセスについてはサポートされていません。

例外フィルタでは、生成された例外をフィルタするために、`com.ibi.afjca.oracle.SAPExceptionFilter` クラスを使用します。このクラスにより、例外はフィルタされて次のカテゴリに分類されます。

- `PCRetriableResourceException`
- `PCResourceException`

次の例外は、フォルト・ポリシー・ファイルに表示されます。

- `PCRetriableResourceException` - リモート・フォルト。
- `PCResourceException` - バインディング・フォルト。

---

**注意：** `com.ibi.afjca.oracle.AdapterExceptionFilter` クラスを使用して、生成された例外をフィルタすることもできます。

---

## 11.1.1 例外フィルタの構成

ここでは、次の項目について説明します。

- 11.1.1.1 項「WSDL ファイルの生成」
- 11.1.1.2 項「例外フィルタ機能を備えた BPEL アウトバウンド・プロセスの作成」
- 11.1.1.3 項「フォルト・ポリシー・ファイルおよびフォルト・バインディング・ファイルの作成」
- 11.1.1.4 項「12c でのデプロイメントに関する既知の問題の調整」
- 11.1.1.5 項「例外フィルタ機能を備えた BPEL アウトバウンド・プロセスのデプロイおよびテスト」

例外フィルタを構成する手順は次のとおりです。

1. WSDL ファイルを生成します。  
詳細は、11-2 ページの「[WSDL ファイルの生成](#)」を参照してください。
2. 例外フィルタ機能を備えた BPEL アウトバウンド・プロセスを作成します。  
詳細は、11-2 ページの「[例外フィルタ機能を備えた BPEL アウトバウンド・プロセスの作成](#)」を参照してください。
3. フォルト・ポリシー・ファイルおよびフォルト・バインディング・ファイルを作成します。  
詳細は、11-4 ページの「[フォルト・ポリシー・ファイルおよびフォルト・バインディング・ファイルの作成](#)」を参照してください。
4. 例外フィルタ機能を備えた BPEL アウトバウンド・プロセスをデプロイおよびテストします。  
詳細は、11-14 ページの「[例外フィルタ機能を備えた BPEL アウトバウンド・プロセスのデプロイおよびテスト](#)」を参照してください。

### 11.1.1.1 WSDL ファイルの生成

WSDL ファイルを生成するには、次のようにします。

1. アプリケーション・エクスペローラを開き、J2CA 構成を作成します。  
詳細は、4-3 ページの「[J2CA の構成の作成](#)」を参照してください。
2. SAP R/3 アダプタのターゲットを作成してから、そのターゲットに接続します。  
詳細は、4-5 ページの「[SAP R/3 の接続 \(ターゲット\) の確立](#)」を参照してください。
3. 適切なオブジェクト (CompanyCode ビジネス・オブジェクトの GetDetail BAPI など) の WSDL を生成します。  
詳細は、4-15 ページの「[WSDL の生成 \(J2CA 構成のみ\)](#)」を参照してください。

### 11.1.1.2 例外フィルタ機能を備えた BPEL アウトバウンド・プロセスの作成

例外フィルタ機能を備えた BPEL アウトバウンド・プロセスを作成するには、次のようにします。

1. JDeveloper を開き、新しい SOA アプリケーションを作成します。  
詳細は、6-11 ページの「[SOA 用の空のコンポジットの作成](#)」を参照してください。
2. 新しい SOA プロジェクト (Exception\_Filter など) を作成します。
3. サード・パーティ・アダプタ・サービス・コンポーネントを作成します。  
詳細は、6-14 ページの「[サード・パーティ・アダプタ・サービス・コンポーネントの構成](#)」を参照してください。

サード・パーティ・アダプタ・サービス・コンポーネントが作成されると、WSDL ファイル (対応するスキーマおよび JCA ファイルを含む) が JDeveloper プロジェクトにインポートされます。

詳細は、6-13 ページの「BPEL アウトバウンド・プロセスの定義」を参照してください。

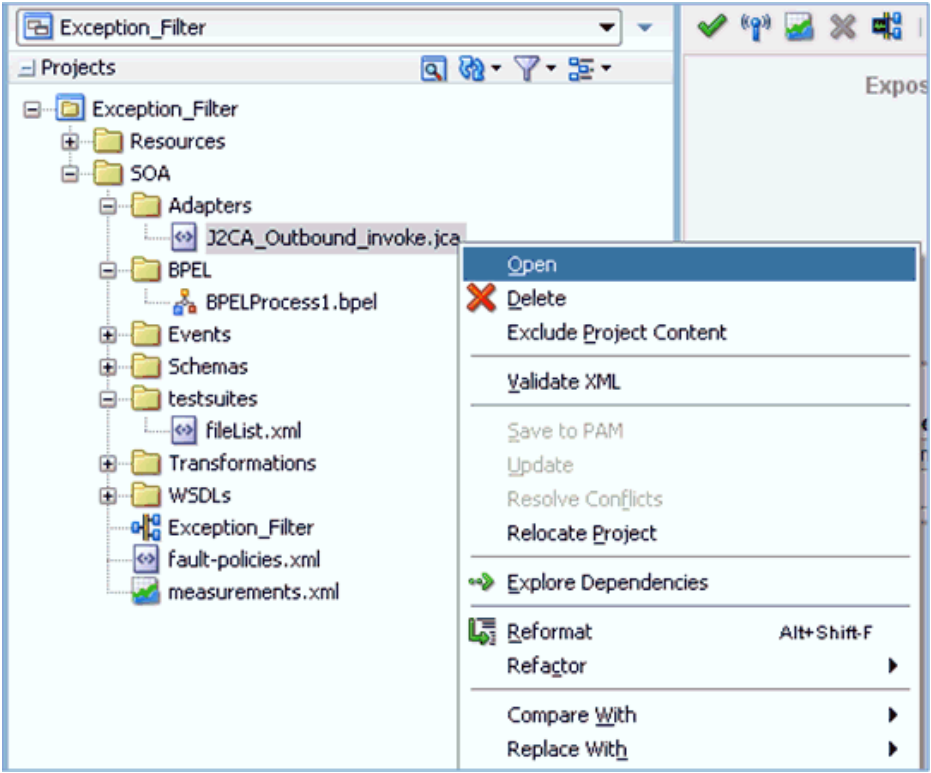
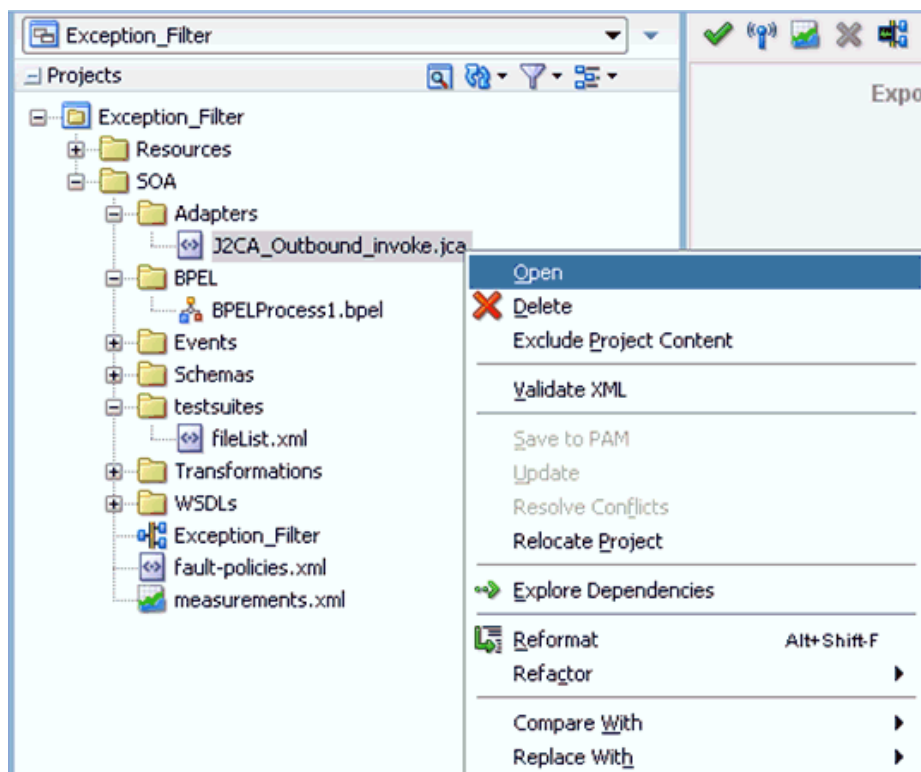
4. インポートされた JCA ファイルを変更します。
  - a.  11-1 に示すように、インポートされた JCA ファイルを右クリックし、メニューの「開く」を選択します。

図 11-1 「アプリケーション・ナビゲータ」タブ



- b. <interaction-spec> 要素に、ExceptionFilter プロパティを追加します。次に例を示します。
 

```
<interaction-spec className="com.ibi.afjca.cci.IWAFInteractionSpec">
  <property name="FunctionName" value="PROCESS"/><property name="ExceptionFilter"
  value="com.ibi.afjca.oracle.SAPEExceptionFilter"/></interaction-spec>
```
  - c. 変更された JCA ファイルを保存します。
5. サード・パーティ・アダプタ・サービス・コンポーネントが作成され、JCA ファイルが変更されたら、BPEL プロセス作成の残りの部分を続行します。
 

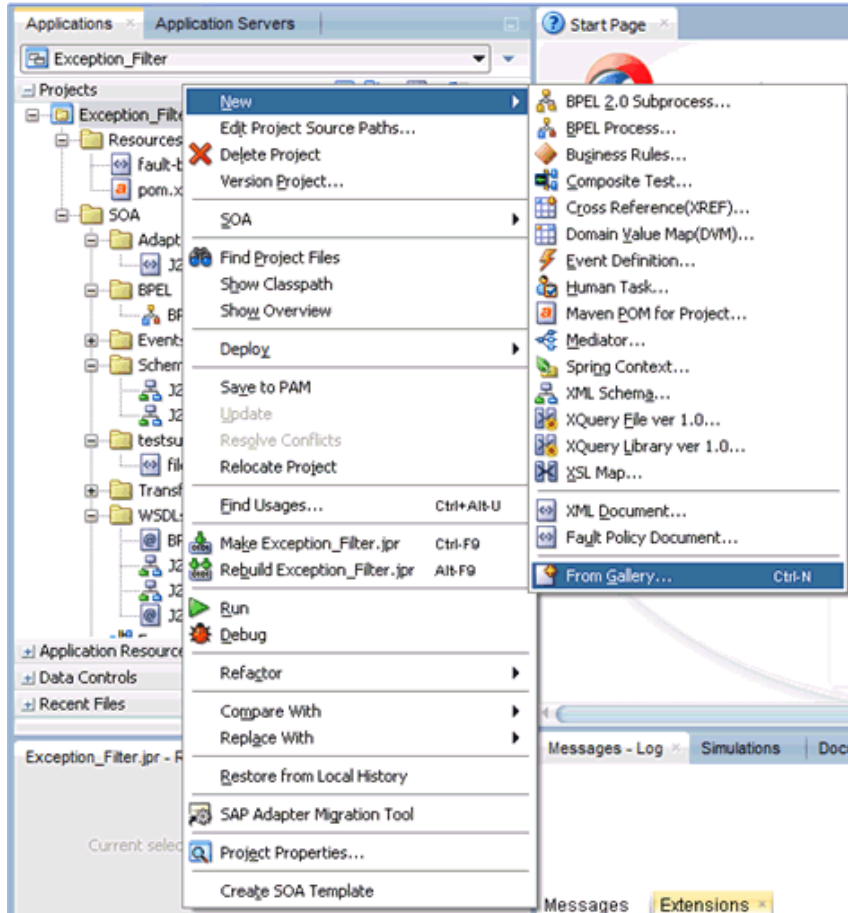
詳細は、6-13 ページの「BPEL アウトバウンド・プロセスの定義」を参照してください。

### 11.1.1.3 フォルト・ポリシー・ファイルおよびフォルト・バインディング・ファイルの作成

フォルト・バインディング・ファイルを作成するには、次のようにします。

1. 図 11-2 に示すように、作成された SOA プロジェクト (Exception\_Filter など) を右クリックして「新」を選択し、「ギャラリーから」をクリックします。

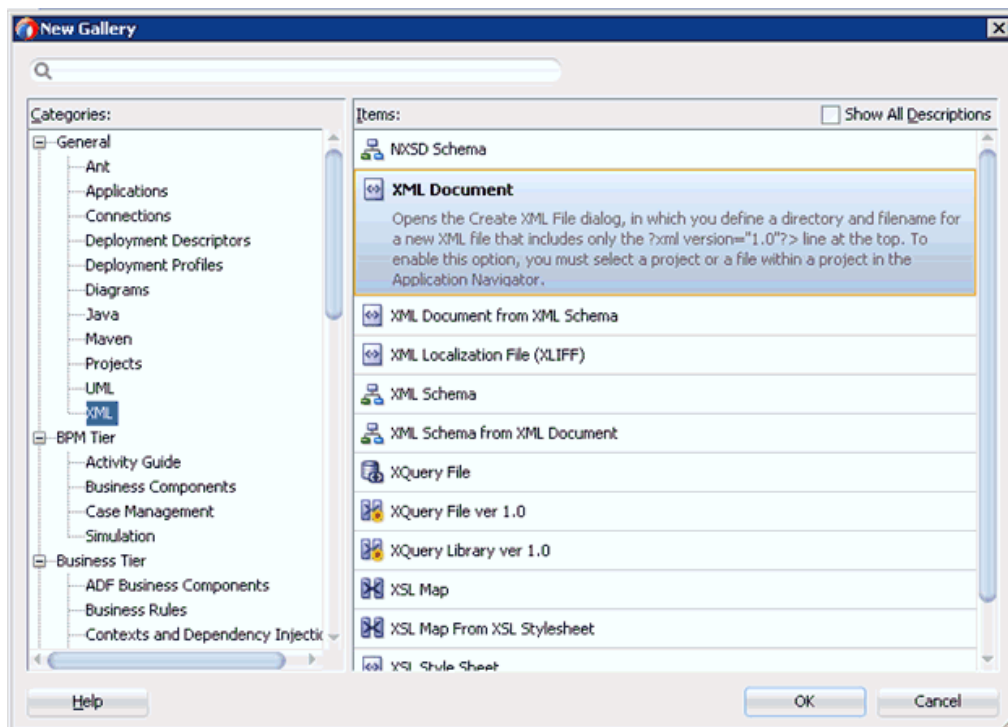
図 11-2 「アプリケーション」タブ





「新規ギャラリー」ダイアログが表示されます。図 11-3 に示すように、「一般」カテゴリの下の「XML」をクリックします。

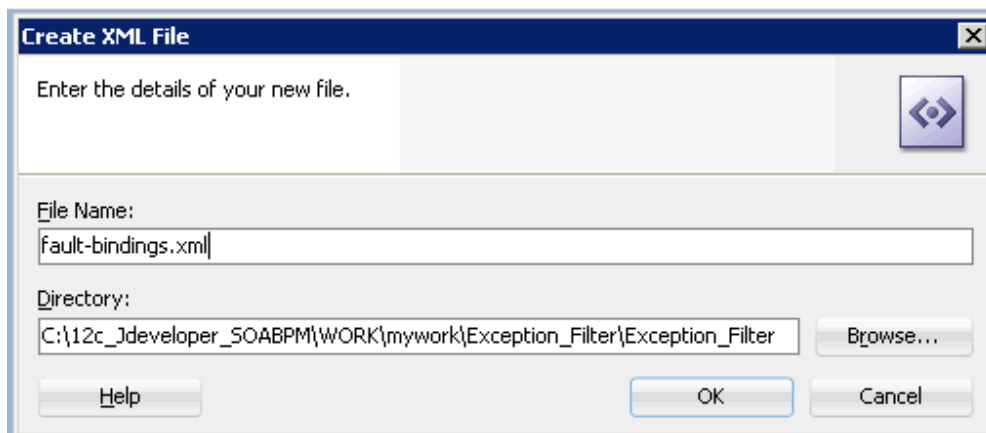
図 11-3 「新規ギャラリー」ダイアログ



2. 「アイテム」で「XML ドキュメント」を選択し、「OK」をクリックします。

図 11-4 に示すように、「XML ファイルの作成」ダイアログが表示されます。

図 11-4 「XML ファイルの作成」ダイアログ



3. 「ファイル名」フィールドに **fault-bindings.xml** と入力し、「OK」をクリックします。

4. **fault-bindings.xml** ファイルに、適切なフォルト・バインディング機能を追加します。サンプルの **fault-bindings.xml** ファイルを表示するには、11-6 ページの「[サンプルの fault-bindings.xml ファイル](#)」を参照してください。

---

**注意：** <name> 要素内のパラメータは、作成された BPEL プロセスの名前です。

---

5. **fault-bindings.xml** ファイルを保存します。

#### サンプルの **fault-bindings.xml** ファイル

```
<?xml version="1.0" encoding="UTF-8"?>
<faultPolicyBindings version="2.0.1"
xmlns="http://schemas.oracle.com/bpel/faultpolicy"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

<component faultPolicy="bpelFaultHandling">
<name>BPELProcess1</name>
</component>

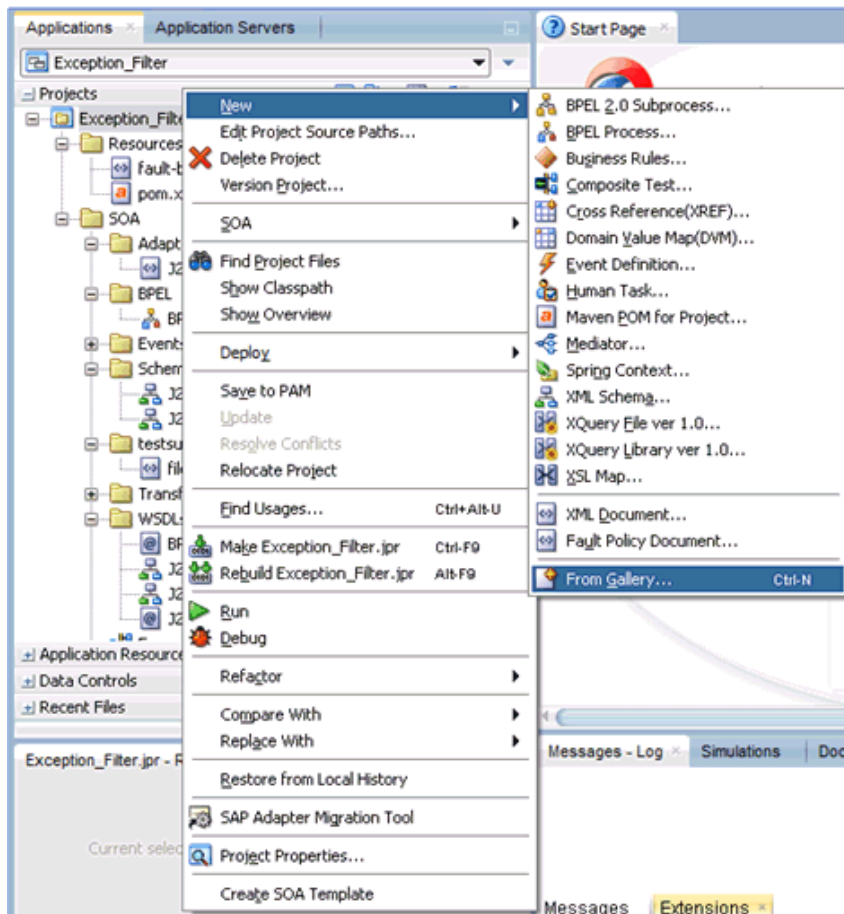
</faultPolicyBindings>
```

#### フォルト・ポリシー・ファイルの作成

フォルト・ポリシー・ファイルを作成するには、次のようにします。

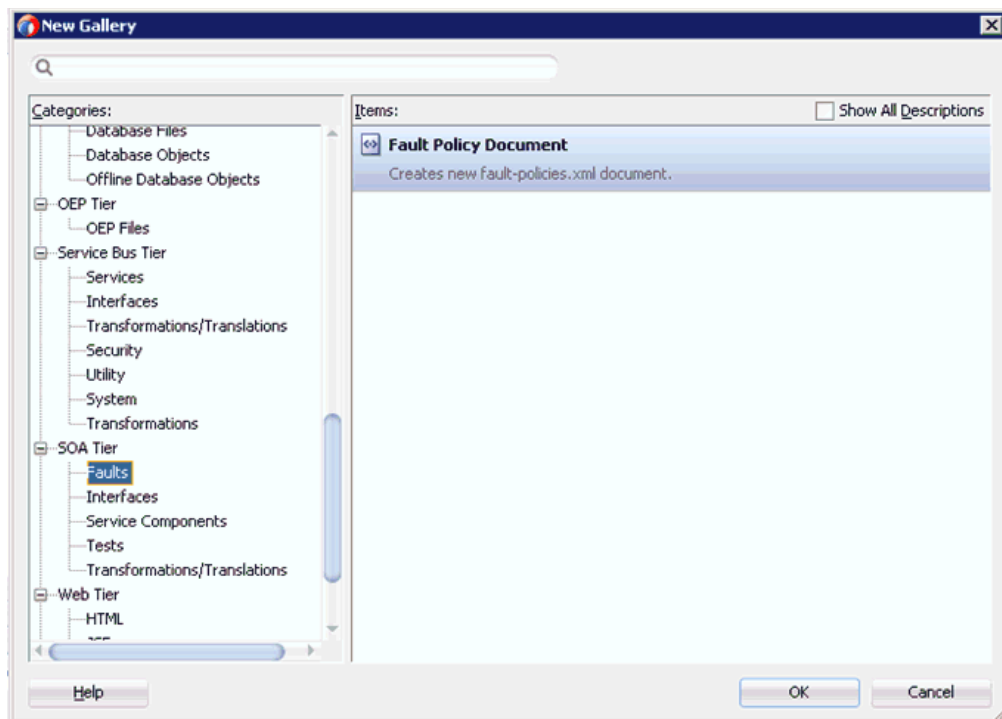
1. 図 11-5 に示すように、作成された SOA プロジェクト (Exception\_Filter など) を右クリックして「新」を選択し、「ギャラリーから」をクリックします。

図 11-5 「アプリケーション」タブ



「新規ギャラリー」ダイアログが表示されます。図 11-6 に示すように、「SOA 層」カテゴリの下の「フォルト」を選択します。

図 11-6 「新規ギャラリー」ダイアログ



2. 「アイテム」で「フォルト・ポリシー・ドキュメント」を選択し、「OK」をクリックします。

3.

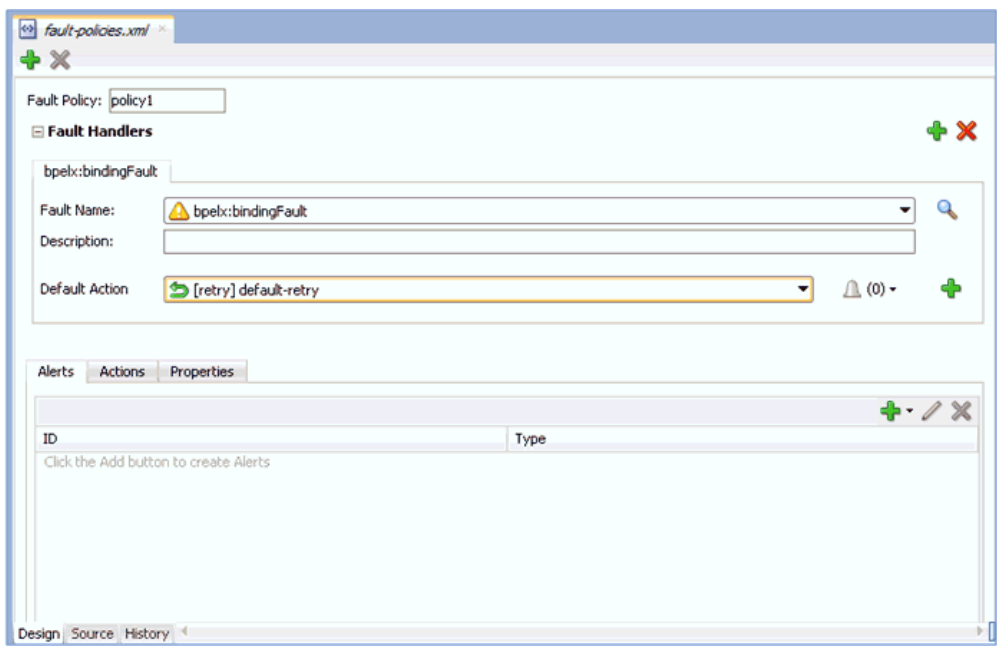


図 11-7 「fault-policies.xml」 タブ

4.

 「アクション」タブをクリックして、「default-retry」をダブルクリックします。  

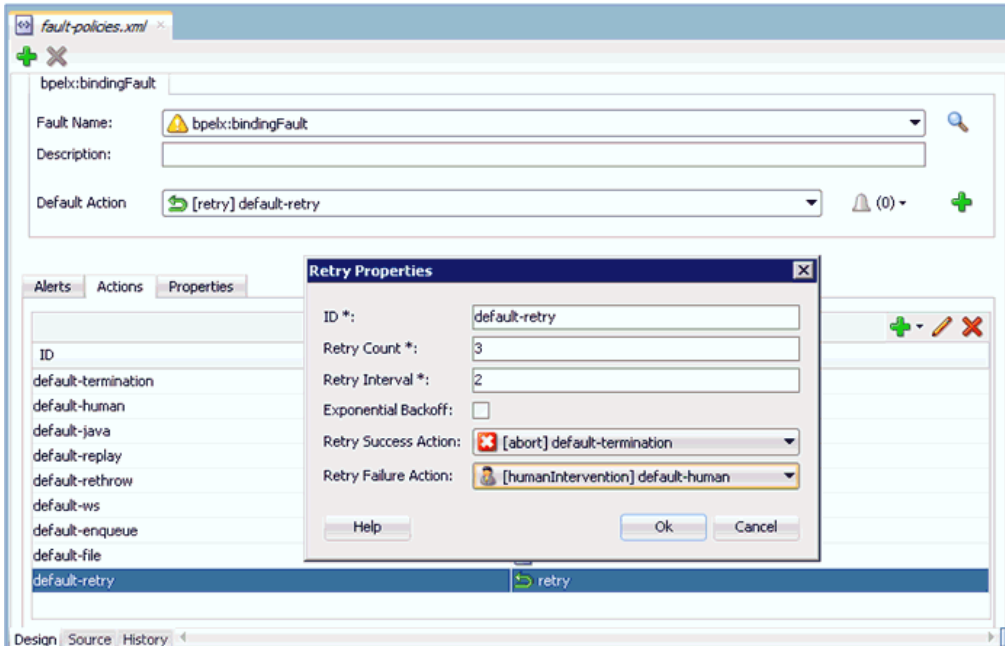


図 11-8 「再試行プロパティ」ダイアログ・ボックス

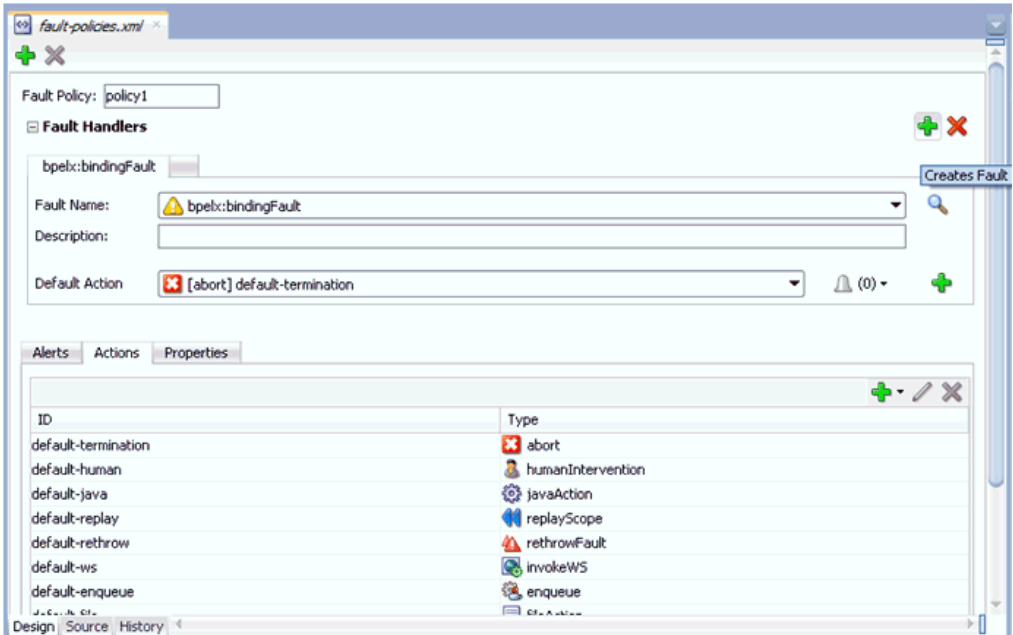
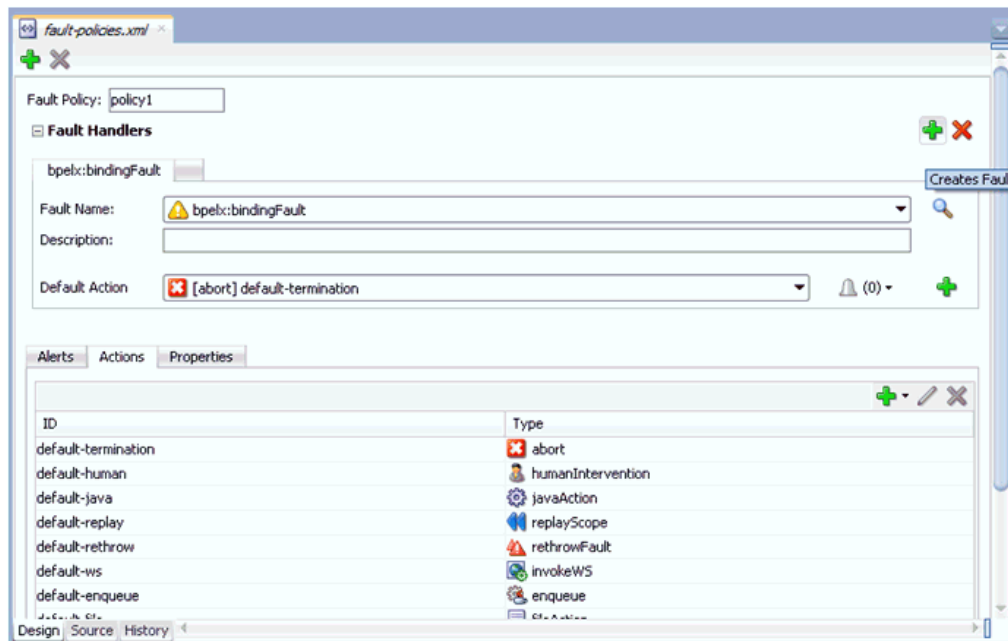
5. 「再試行成功アクション」ドロップダウン・リストで「**[abort] default-termination**」を選択し、「再試行失敗アクション」ドロップダウン・リストで「**[humanIntervention] default-human**」を選択します。
6. 「OK」をクリックします。
7.  11-9 に示すように、別のフォルト・ハンドラを作成するには、「追加」をクリックします。

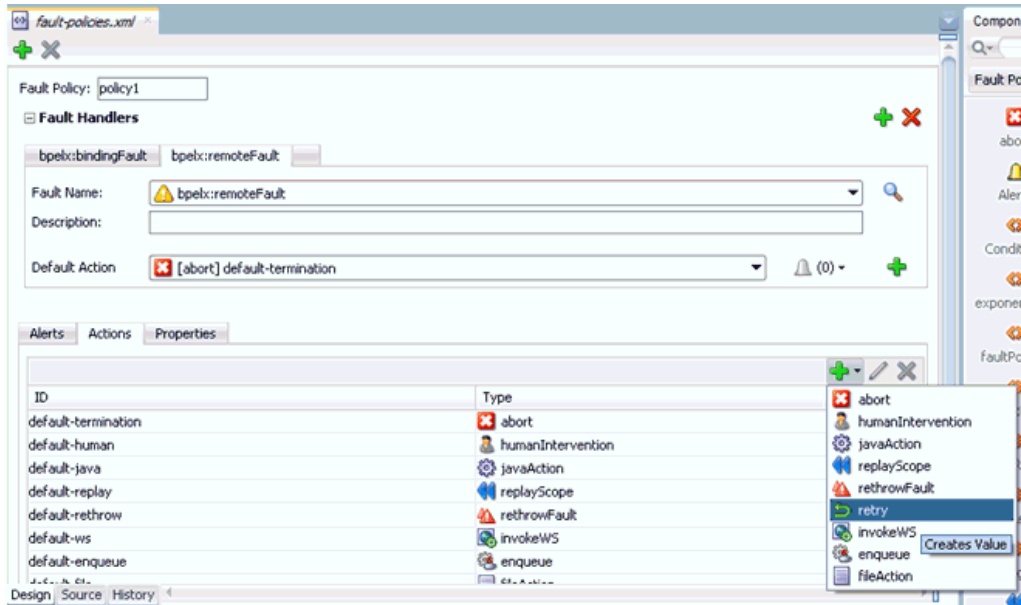
図 11-9 「fault-policies.xml」タブ



8. 「fault-policies.xml」タブの「フォルト名」ドロップダウン・リストで「**bpelx:remoteFault**」を選択し、「デフォルト・アクション」ドロップダウン・リストで「**[abort] default-termination**」を選択します。

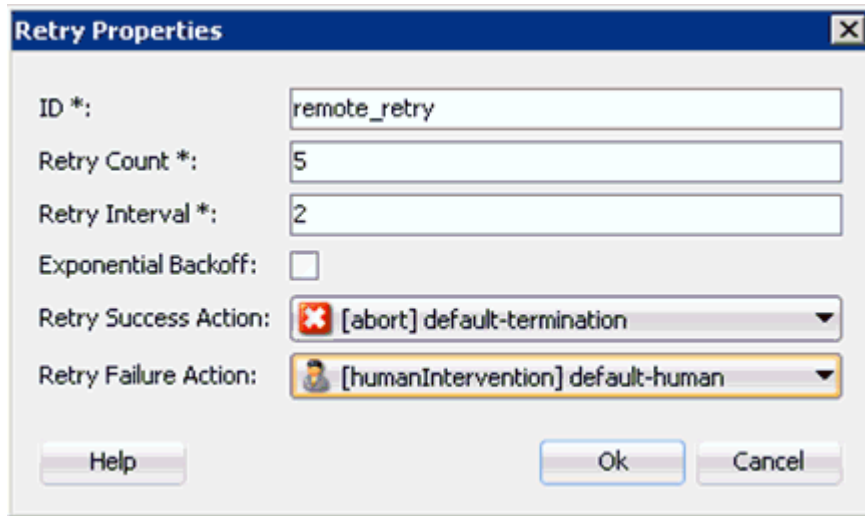
9. 図 11-10 に示すように、「アクション」タブで「追加」をクリックし、「retry」を選択します。

図 11-10 「アクション」タブ



10. 図 11-11 に示すように、「再試行プロパティ」ダイアログが表示されます。

図 11-11 「再試行プロパティ」ダイアログ・ボックス



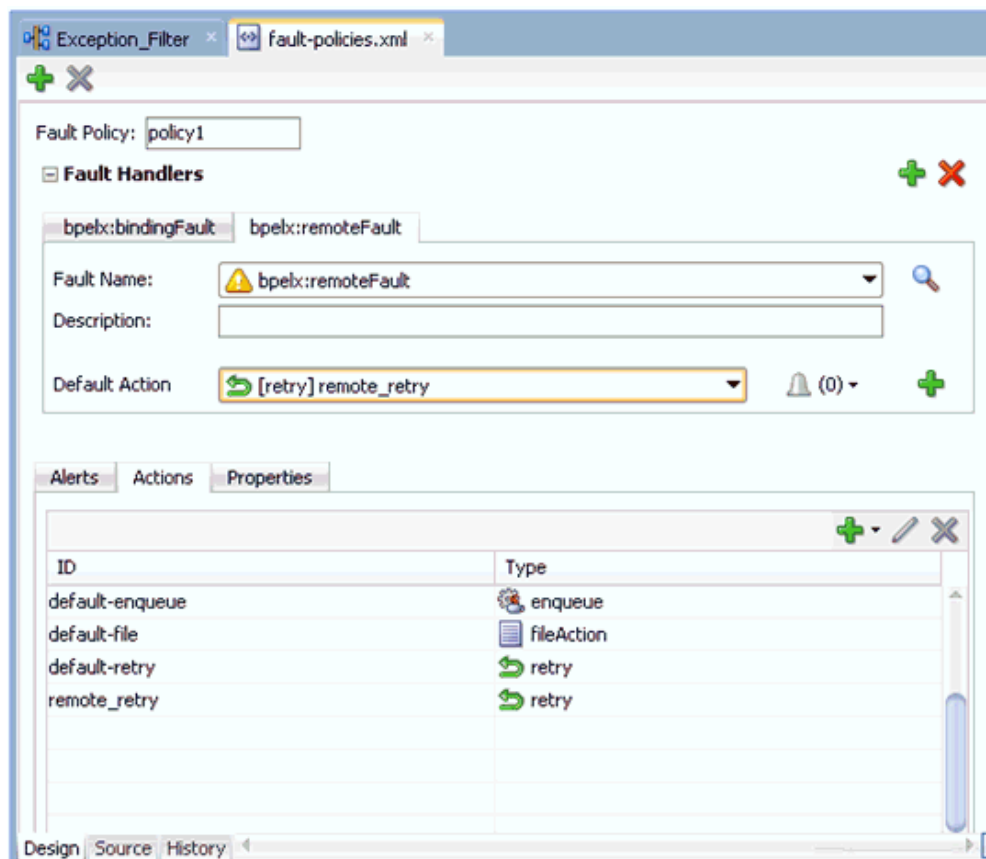
11. 「ID」、「再試行数」および「再試行間隔」の各フィールドに値を指定します。
12. 「再試行成功アクション」ドロップダウン・リストで「[abort] default-termination」を選択し、「再試行失敗アクション」ドロップダウン・リストで「[humanIntervention] default-human」を選択します。

12. 「OK」をクリックします。

作成された再試行 ID が「アクション」タブにリストされます。

図 11-12 に示すように、「デフォルト・アクション」ドロップダウン・リストで新しく作成された再試行 ID (remote\_retry など) を選択します。

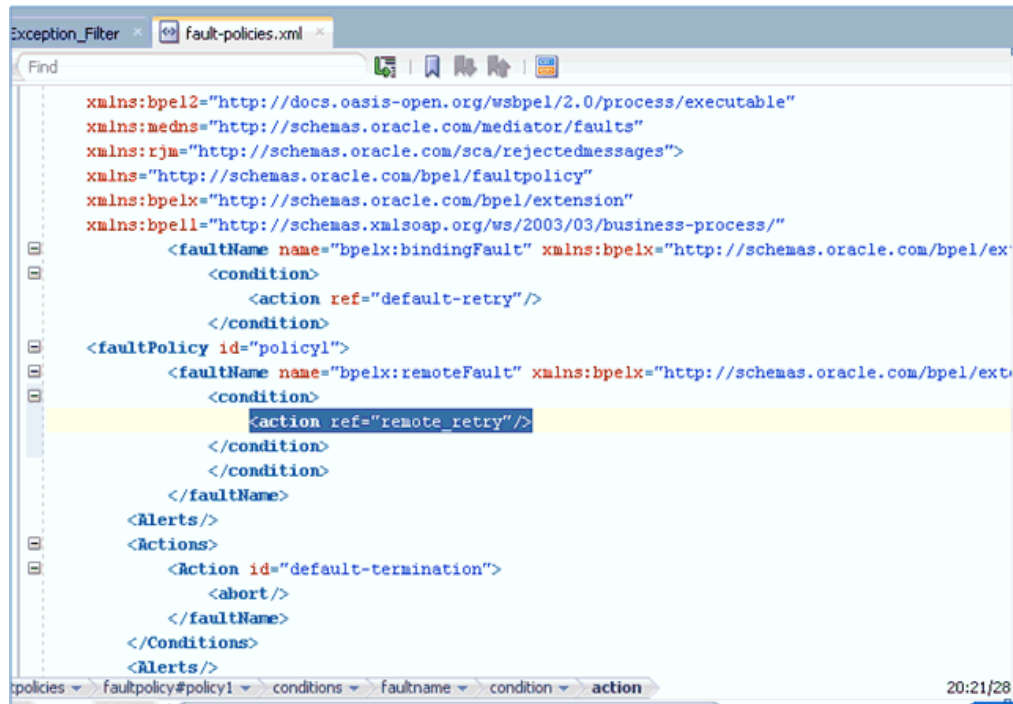
図 11-12 「fault-policies.xml」タブ



13. 「すべて保存」をクリックします。

14. 図 11-13 に示すように、「ソース」タブをクリックして、フォルト・ポリシーが正しく追加されていることを確認します。

図 11-13 「ソース」タブ



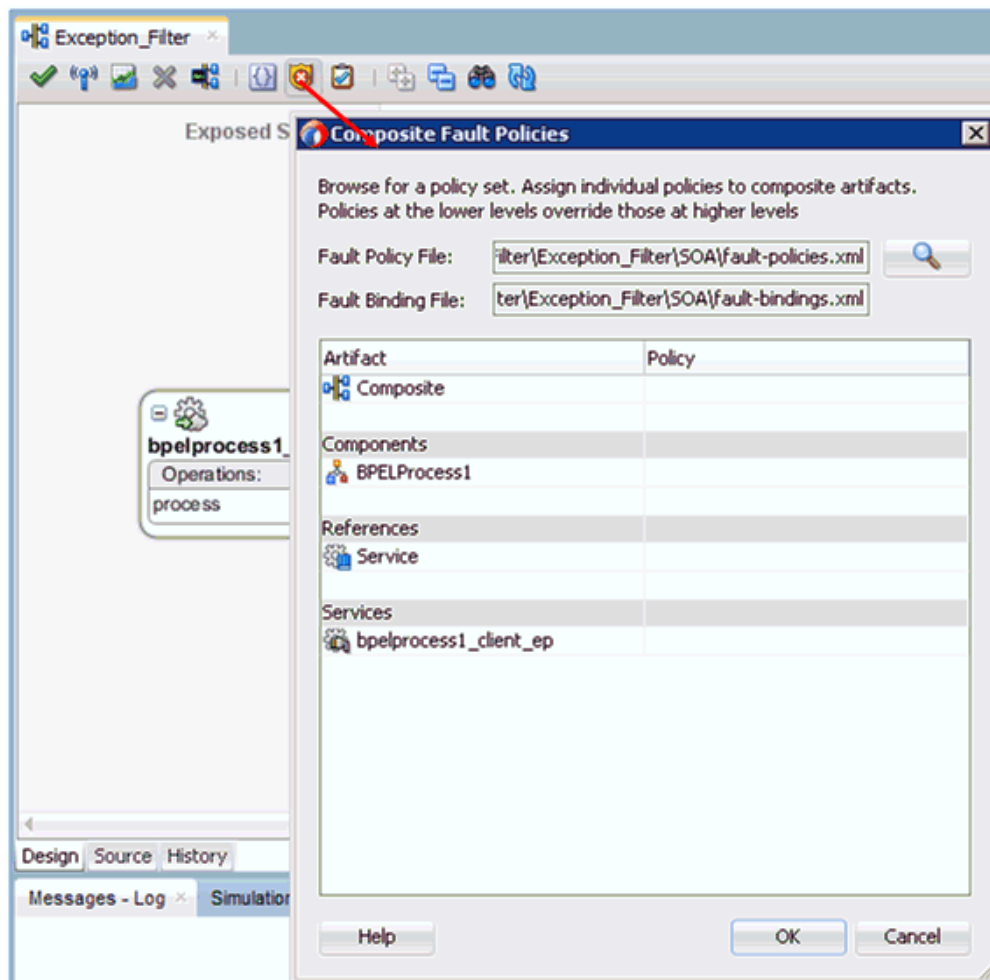
```
xmlns:bpel2="http://docs.oasis-open.org/wsbpel/2.0/process/executable"
xmlns:medns="http://schemas.oracle.com/mediator/faults"
xmlns:rjm="http://schemas.oracle.com/sca/rejectedmessages">
xmlns="http://schemas.oracle.com/bpel/faultpolicy"
xmlns:bpelx="http://schemas.oracle.com/bpel/extension"
xmlns:bpel1="http://schemas.xmlsoap.org/ws/2003/03/business-process/"
  <faultName name="bpelx:bindingFault" xmlns:bpelx="http://schemas.oracle.com/bpel/ex
    <condition>
      <action ref="default-retry"/>
    </condition>
  <faultPolicy id="policy1">
    <faultName name="bpelx:remoteFault" xmlns:bpelx="http://schemas.oracle.com/bpel/ext
      <condition>
        <action ref="remote_retry"/>
      </condition>
    </condition>
  </faultName>
  <Alerts/>
  <Actions>
    <Action id="default-termination">
      <abort/>
    </faultName>
  </Conditions>
  <Alerts/>
```



15. 「Exception\_Filter」プロジェクトをダブルクリックし、「コンポジット・フォルト・ポリシーの編集」をクリックします。

「コンポジット・フォルト・ポリシー」ウィンドウが表示されます。図 11-14 に示すように、フォルト・ポリシーとフォルト・バインディングが正しく選択されていることを確認します。

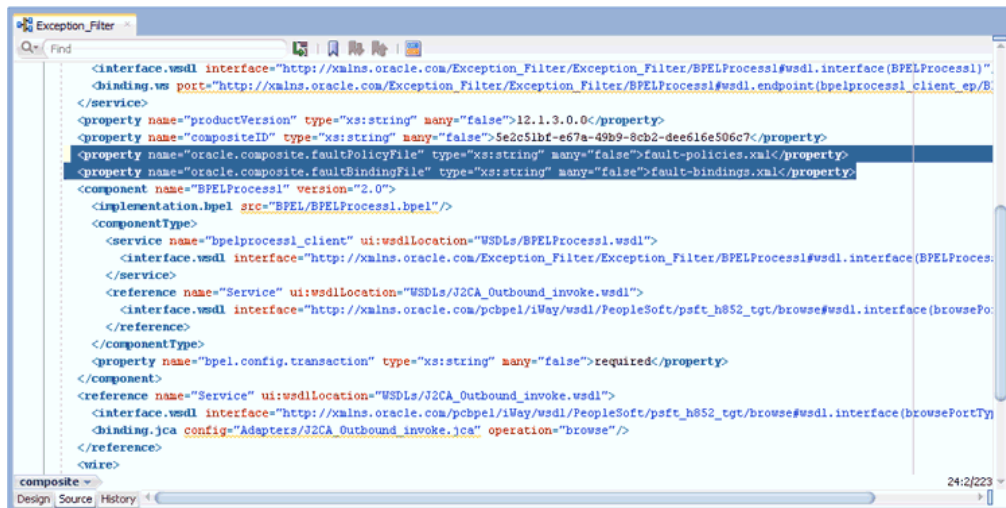
図 11-14 「コンポジット・フォルト・ポリシー」ウィンドウ



16. 「すべて保存」をクリックします。

17. 図 11-15 に示すように、「ソース」タブをクリックして、*fault-bindings.xml* ファイルと *fault-policies.xml* ファイルが正しく追加されていることを確認します。

図 11-15 「ソース」タブ



#### 11.1.1.4 12c でのデプロイメントに関する既知の問題の調整

12c でのデプロイメントに関する既知の問題を調整する方法については、6-32 ページの 6.4.3.3 項「12c でのデプロイメントに関する既知の問題の調整」を参照してください。

#### 11.1.1.5 例外フィルタ機能を備えた BPEL アウトバウンド・プロセスのデプロイおよびテスト

例外フィルタ機能を備えた BPEL アウトバウンド・プロセスをデプロイおよびテストするには、次のようにします。

1. 作成済 BPEL アウトバウンド・プロセスをデプロイします。  
詳細は、6-34 ページの「BPEL アウトバウンド・プロセスのデプロイ」を参照してください。
2. (サーバーが実行中の) システムをネットワークから切断して、通信エラーをシミュレートします。
3. 有効な入力を含む、デプロイされた BPEL アウトバウンド・プロセスを呼び出します。  
詳細は、6-39 ページの「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」を参照してください。
4. プロセス ID を選択します。

**fault-policies.xml** ファイルの構成に基づいて、再試行または中断されている BPEL アウトバウンド・プロセスを監視できます。

## 11.2 Oracle SOA Suite (BPEL、メディエータまたは BPM) の資格証明マッピング

この項では、Oracle SOA Suite (BPEL、メディエータまたは BPM) を使用する構成において、Oracle Application Adapter for SAP R/3 用に資格証明マッピング機能を構成する方法について説明します。サンプルのテスト・シナリオも含まれます。この項では、次の項目について説明します。

### ■ 11.2.1 項「資格証明マッピングの構成」

資格証明マッピングは、J2CA 構成を使用するアウトバウンド・プロセスについてのみサポートされます。この機能は、BSE 構成、および J2CA 構成を使用するインバウンド・プロセスについてはサポートされていません。

---

**注意：**J2CA コネクタは、SAP R/3、PeopleSoft、Siebel および J.D. Edwards OneWorld の 4 つアプリケーション・アダプタすべてに共通です。資格証明マッピングが必要な場合は、特定のインスタンスでアプリケーション・アダプタを 1 つのみ使用するようにしてください。たとえば、あるアダプタ・インスタンスでは、SAP R/3 アプリケーション・アダプタのみを使用できます。資格証明マッピングを個々のアダプタ・レベルで構成することはできません。2 つのアダプタについて資格証明マッピングを使用する必要がある場合は、両方のアダプタが、独立した 2 つのアダプタ・インスタンスで実行されている必要があります。

---

ユーザー資格証明を J2CA リソース・アダプタに渡すには、Oracle WebLogic Server ユーザー資格証明から EIS ユーザー資格証明 (SAP R/3 アダプタ) への資格証明マップを作成します。これらの資格証明は、EIS ユーザー資格証明にマップされた後に J2CA コンテナに渡され、J2CA コンテナによって EIS アダプタ (SAP R/3) への接続に使用されます。

### 11.2.1 資格証明マッピングの構成

この項では、資格証明マッピングの構成について説明し、次の手順と項目で構成されています。

#### 1. アダプタをデプロイします。

詳細は、[第 5 章「Oracle WebLogic Server のデプロイメントおよび統合」](#)を参照してください。

#### 2. Oracle WebLogic Server 資格証明と EIS 資格証明を関連付けます。

詳細は、11-16 ページの [11.2.1.1 項「Oracle WebLogic Server 資格証明と EIS 資格証明の関連付け」](#)を参照してください。

#### 3. WSDL ファイルを生成します。

詳細は、11-18 ページの [11.2.1.2 項「WSDL ファイルの生成」](#)を参照してください。

#### 4. アウトバウンド・プロセスを作成およびデプロイします。

詳細は、11-18 ページの [11.2.1.3 項「アウトバウンド・プロセスの作成およびデプロイ」](#)を参照してください。

#### 5. 呼び出して、EIS 資格証明が渡されたことを確認します。

詳細は、11-19 ページの [11.2.1.4 項「EIS 資格証明の検証」](#)を参照してください。

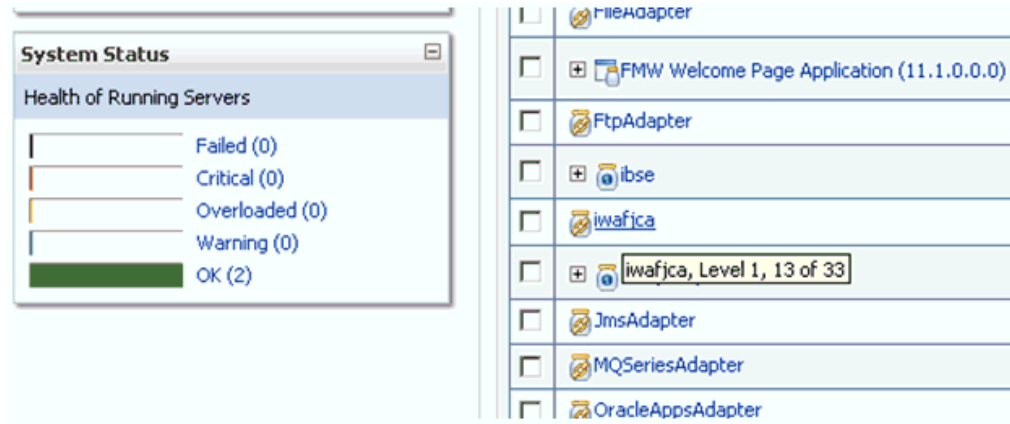
### 11.2.1.1 Oracle WebLogic Server 資格証明と EIS 資格証明の関連付け

Oracle WebLogic Server 資格証明と EIS 資格証明を関連付けるには、次のようにします。

1. Oracle WebLogic Server 管理コンソールにログインします。
2. 左ペインの「ドメイン構造」セクションで、「デプロイメント」をクリックします。

☒ 11-16 に示すように、「デプロイメント」ページが表示されます。

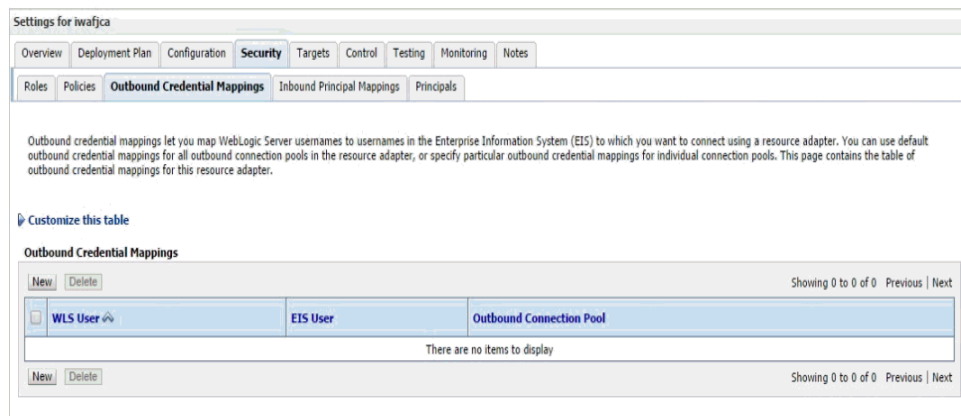
図 11-16 「デプロイメント」ページ



3. 「iwafjca」リソース・アダプタをクリックします。

☒ 11-17 に示すように、「iwafjca の設定」ページが表示されます。

図 11-17 「iwafjca の設定」ページ



- 「セキュリティ」タブで「アウトバウンド資格証明マッピング」タブをクリックし、「新規」をクリックします。

図 11-18 に示すように、「新しいセキュリティ資格証明マッピングの作成」ページが表示されます。

図 11-18 「新しいセキュリティ資格証明マッピングの作成」ページ

**Create a New Security Credential Mapping**

Back Next Finish Cancel

**Outbound Connection Pool**

Which Outbound Connection Pool would you like the credential map to be associated with this resource adapter. Each Outbound Connection Pool can then configure themselves

Customize this table

**Create a New Security Credential Map Entry for:**

<input type="checkbox"/>	Outbound Connection Pool
<input checked="" type="checkbox"/>	eis/OracleJCAAdapter/DefaultConnection
<input type="checkbox"/>	Resource Adapter Default

Back Next Finish Cancel

- アウトバウンド接続プールを選択します。

次に例を示します。

eis/OracleJCAAdapter/DefaultConnection

- 「次」をクリックします。

図 11-19 に示すように、「WebLogic Server ユーザー」ページが表示されます。

図 11-19 「WebLogic Server ユーザー」ページ

**Create a New Security Credential Mapping**

Back Next Finish Cancel

**WebLogic Server User**

Select the WebLogic Server User that you would like to map an EIS user to. Selecting 'User for creating initial connections' will configure the user that will be used for creating the initial connections when the resource adapter is first started. Selecting 'Default User' will configure the user that will be used as the default for any authenticated WebLogic Server user that does not have a credential mapping specifically for them. Selecting 'User for unauthenticated user' will configure the user that will be used for an unauthenticated WebLogic Server user. If you select 'Configured User' you must type in the WebLogic Server user that you are configuring. This user must be a configured WebLogic Server user.

User for creating initial connections  
 Default User  
 Unauthenticated WLS User  
 Configured User Name

WebLogic Server User Name:

Back Next Finish Cancel

7. 「デフォルト・ユーザー」を選択して、有効な Oracle WebLogic Server ユーザー名を入力し、「次」をクリックします。

図 11-20 に示すように、「EIS ユーザー名およびパスワード」ページが表示されます。

図 11-20 「EIS ユーザー名およびパスワード」ページ

8. EIS のユーザー名およびパスワードを入力し、「終了」をクリックします。

これで、Oracle WebLogic Server ユーザーの資格証明は、EIS ユーザー (SAP R/3) にマップされました。マッピングは、J2CA サービスの呼出し前に自動的に呼び出されます。

### 11.2.1.2 WSDL ファイルの生成

WSDL ファイルを生成するには、次のようにします。

1. アプリケーション・エクスプローラを開き、J2CA 構成を作成します。  
詳細は、4-3 ページの 4.3.2 項「J2CA の構成の作成」を参照してください。
2. SAP R/3 アダプタのターゲットを作成してから、そのターゲットに接続します。  
詳細は、4-5 ページの 4.4 項「SAP R/3 の接続 (ターゲット) の確立」を参照してください。
3. 適切なオブジェクトのアウトバウンド WSDL を生成します。  
詳細は、4-15 ページの 4.7 項「WSDL の生成 (J2CA 構成のみ)」を参照してください。

### 11.2.1.3 アウトバウンド・プロセスの作成およびデプロイ

この項では、アウトバウンド・プロセスを構成する方法について説明します。デモンストレーションの目的で、BPEL アウトバウンド・プロセスへの特定の参照を作成します。ただし、メディアータおよび BPM アウトバウンド・プロセスにも同じ手順が適用されます。

メディアータ・アウトバウンド・プロセスの作成の詳細は、第 7 章「Oracle SOA Suite でのメディアータ・サービス・コンポーネントとの統合」を参照してください。

BPM アウトバウンド・プロセスの作成の詳細は、第 8 章「Oracle SOA Suite での BPM サービス・コンポーネントとの統合」を参照してください。

BPEL アウトバウンド・プロセスを作成およびデプロイするには、次の項を参照してください。

- 6.4.2 項「SOA 用の空のコンポジットの作成」
- 6.4.3 項「BPEL アウトバウンド・プロセスの定義」
- 6.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」

#### 11.2.1.4 EIS 資格証明の検証

入力 XML を呼び出し、EIS ターゲットの資格証明が、この項の説明に従って WebLogic 管理コンソールでデフォルト・ユーザー用に構成された資格証明でオーバーライドされることを確認してください。

1. 有効な入力を含む、デプロイされた BPEL アウトバウンド・プロセスを呼び出します。  
詳細は、6-39 ページの「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの呼出し」を参照してください。
2. J2CA ログ・ファイルを確認し、ユーザー資格証明が Oracle WebLogic Server を介して EIS に渡されたことを示す暗号化パスワードを特定します。

次に例を示します。

```
FINEST IWFManagedConnectionFactory com.ibi.afjca.Util getPasswordCredential (78)
InLoop:
User-iwayqa:Password-ENCR(3109311731831131382333215315332323192322731773172)
FINEST IWFManagedConnectionFactory com.ibi.afjca.Util getPasswordCredential (90)
Use the system PasswordCredential:
User-iwayqa:Password-ENCR(3109311731831131382333215315332323192322731773172)
```

## 11.3 JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング

この項では、Oracle Service Bus (OSB) を使用する構成において、Oracle Application Adapter for SAP R/3 用に資格証明マッピング機能を構成する方法について説明します。サンプルのテスト・シナリオも含まれます。この項では、次の項目について説明します。

- 11.3.1 項「資格証明マッピングの構成」

資格証明マッピングは、J2CA 構成を使用するアウトバウンド・プロセスについてのみサポートされます。この機能は、BSE 構成、および J2CA 構成を使用するインバウンド・プロセスについてはサポートされていません。

---

**注意：**J2CA コネクタは、SAP R/3、PeopleSoft、Siebel および J.D. Edwards OneWorld の 4 つアプリケーション・アダプタすべてに共通です。資格証明マッピングが必要な場合は、特定のインスタンスでアプリケーション・アダプタを 1 つのみ使用するようにしてください。たとえば、あるアダプタ・インスタンスでは、SAP R/3 アプリケーション・アダプタのみを使用できます。資格証明マッピングを個々のアダプタ・レベルで構成することはできません。2 つのアダプタについて資格証明マッピングを使用する必要がある場合は、両方のアダプタが、独立した 2 つのアダプタ・インスタンスで実行されている必要があります。

---

ユーザー資格証明を J2CA リソース・アダプタに渡すには、Oracle WebLogic Server ユーザー資格証明から EIS ユーザー資格証明 (SAP R/3 アダプタ) への資格証明マップを作成します。次に、資格証明ポリシーを Web サービスに関連付けて、Oracle WebLogic Server ユーザー資格証明を使用して Web サービスを呼び出します。これらの資格証明は、EIS ユーザー資格証明にマップされた後に J2CA コンテナに渡され、J2CA コンテナによって EIS アダプタ (SAP R/3) への接続に使用されます。

## 11.3.1 資格証明マッピングの構成

この項では、資格証明マッピングの構成について説明し、次の手順と項目で構成されています。

1. アダプタをデプロイします。  
詳細は、第 5 章「Oracle WebLogic Server のデプロイメントおよび統合」を参照してください。
2. Oracle WebLogic Server 資格証明と EIS 資格証明を関連付けます。  
詳細は、11.3.1.1 項「Oracle WebLogic Server 資格証明と EIS 資格証明の関連付け」を参照してください。
3. WSDL ファイルを生成します。  
詳細は、11.3.1.2 項「WSDL ファイルの生成」を参照してください。
4. Oracle Service Bus (OSB) アウトバウンド・プロセスを作成します。  
詳細は、11.3.1.3 項「Oracle Service Bus (OSB) アウトバウンド・プロセスの作成」を参照してください。

### 11.3.1.1 Oracle WebLogic Server 資格証明と EIS 資格証明の関連付け

Oracle WebLogic Server 資格証明と EIS 資格証明を関連付けるには、次のようにします。

1. Oracle WebLogic Server 管理コンソールにログインします。
2. 図 11-21 に示すように、左ペインの「ドメイン構造」セクションで、「デプロイメント」をクリックします。

図 11-21 「ドメイン構造」セクション

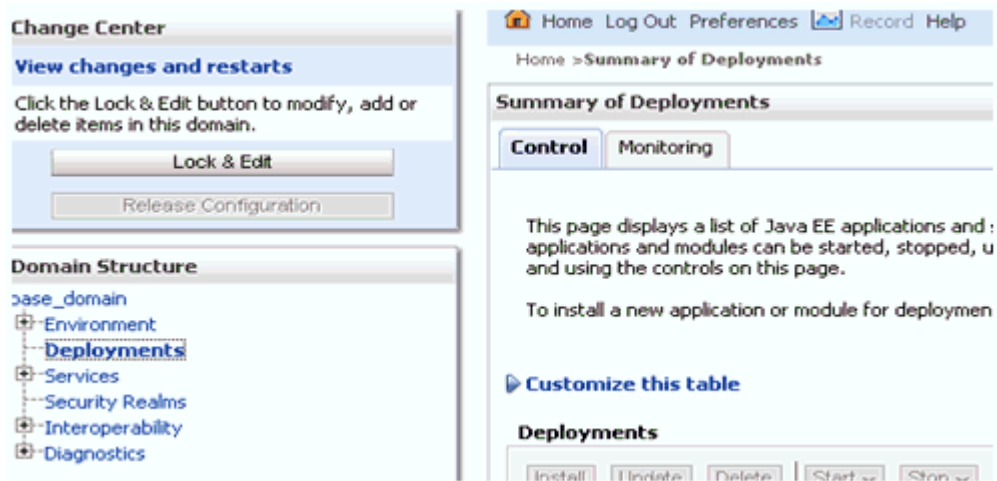
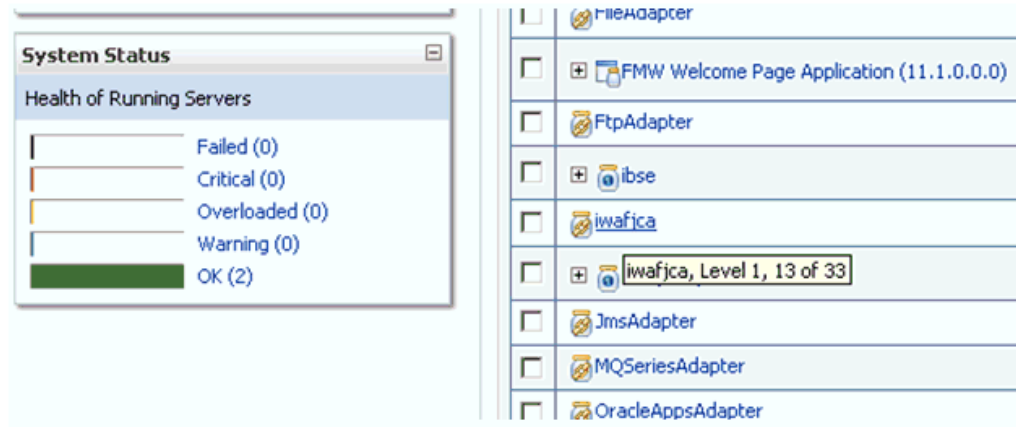




図 11-22 に示すように、「デプロイメント」ページが表示されます。

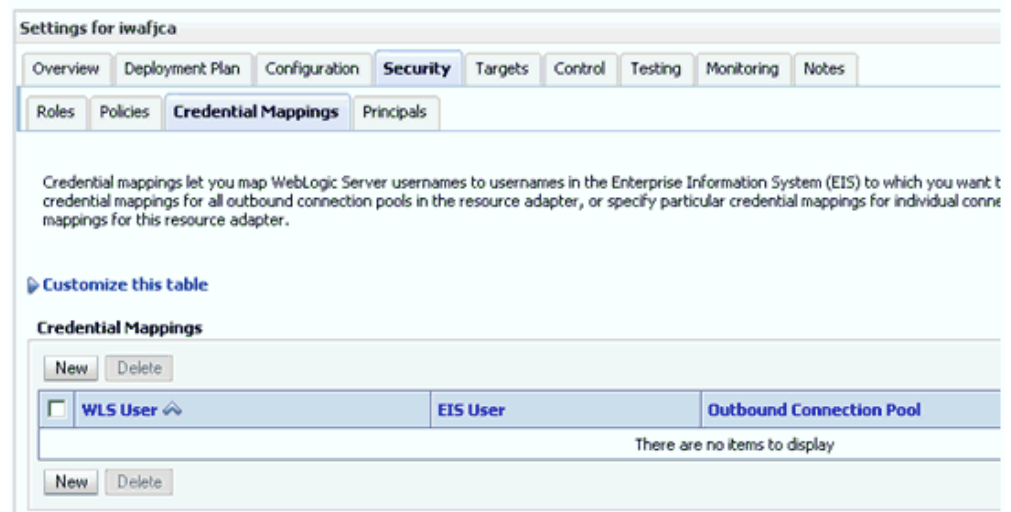
図 11-22 「デプロイメント」ページ



3. 「iwaafjca」リソース・アダプタをクリックします。

図 11-23 に示すように、「iwaafjca の設定」ページが表示されます。

図 11-23 「iwaafjca の設定」ページ



- 「セキュリティ」タブで「資格証明マッピング」タブをクリックして、「新規」をクリックします。

図 11-24 に示すように、「新しいセキュリティ資格証明マッピングの作成」ページが表示されます。

図 11-24 「新しいセキュリティ資格証明マッピングの作成」ページ

- アウトバウンド接続プールを選択します。

次に例を示します。

eis/OracleJCAAdapter/DefaultConnection

- 「次」をクリックします。

図 11-25 に示すように、「WebLogic Server ユーザー」ページが表示されます。

図 11-25 「WebLogic Server ユーザー」ページ

- 「構成済みユーザー名」を選択して、有効な Oracle WebLogic Server ユーザー名を入力し、「次」をクリックします。

図 11-26 に示すように、「EIS ユーザー名およびパスワード」ページが表示されます。

図 11-26 「EIS ユーザー名およびパスワード」ページ

- EIS のユーザー名およびパスワードを入力し、「終了」をクリックします。

これで、Oracle WebLogic Server ユーザーの資格証明は、EIS ユーザー (SAP R/3) にマップされました。マッピングは、J2CA サービスの呼出し前に自動的に呼び出されます。

### 11.3.1.2 WSDL ファイルの生成

WSDL ファイルを生成するには、次のようにします。

- アプリケーション・エクスプローラが Oracle Service Bus (OSB) と統合するためのクラス・パスを設定します。  
詳細は、9-6 ページの「[アプリケーション・エクスプローラが Oracle Service Bus と統合するためのクラス・パスの設定](#)」を参照してください。
- アプリケーション・エクスプローラを開き、J2CA 構成を作成します。  
詳細は、4-3 ページの「[J2CA の構成の作成](#)」を参照してください。
- SAP R/3 アダプタのターゲットを作成してから、そのターゲットに接続します。  
詳細は、4-5 ページの「[SAP R/3 の接続 \(ターゲット\) の確立](#)」を参照してください。
- 適切なオブジェクト (CompanyCode ビジネス・オブジェクトの GetDetail BAPI など) の WSDL を OSB にパブリッシュします。  
詳細は、9-6 ページの「[アプリケーション・エクスプローラから Oracle Service Bus への WSDL のパブリッシュ](#)」を参照してください。

### 11.3.1.3 Oracle Service Bus (OSB) アウトバウンド・プロセスの作成

この項では、Oracle JDeveloper を使用して、資格証明マッピング機能を備えた Oracle Service Bus (OSB) アウトバウンド・プロセスを作成する方法について説明します。

1. Oracle JDeveloper を開始し、OSB に Service Bus アプリケーションを作成します。  
詳細は、10-20 ページの 10.2.1 項「OSB の Service Bus アプリケーションの作成」を参照してください。
2. サード・パーティ・アダプタ・サービス・コンポーネントを作成します。  
詳細は、10-21 ページの 10.2.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」を参照してください。
3. JCA バインディング・ファイルから、WSDL ベースのビジネス・サービスを作成します。  
詳細は、10-33 ページの 10.3.2.1 項「WSDL ベースのビジネス・サービスの構成」を参照してください。
4. 出力先のファイル・トランスポート・タイプ・ビジネス・サービスを作成します。  
詳細は、10-28 ページの 10.2.2.3 項「ファイル・トランスポート・タイプ・ビジネス・サービスの構成」を参照してください。
5. パイプラインを含むプロキシ・サービスを作成します。  
詳細は、10-37 ページの 10.3.2.2 項「パイプラインを含むプロキシ・サービスの作成」を参照してください。
6. 作成されたパイプラインにルーティング・ルールを構成します。  
詳細は、10-30 ページの 10.2.2.4 項「ルーティング・ルールの構成」を参照してください。
7. サービス・アカウントを構成するには、[図 11-27](#) に示すように、OSB プロジェクトを右クリックして「新規」を選択し、「サービス・アカウント」をクリックします。

図 11-27 「サービス・アカウント」オプションの選択

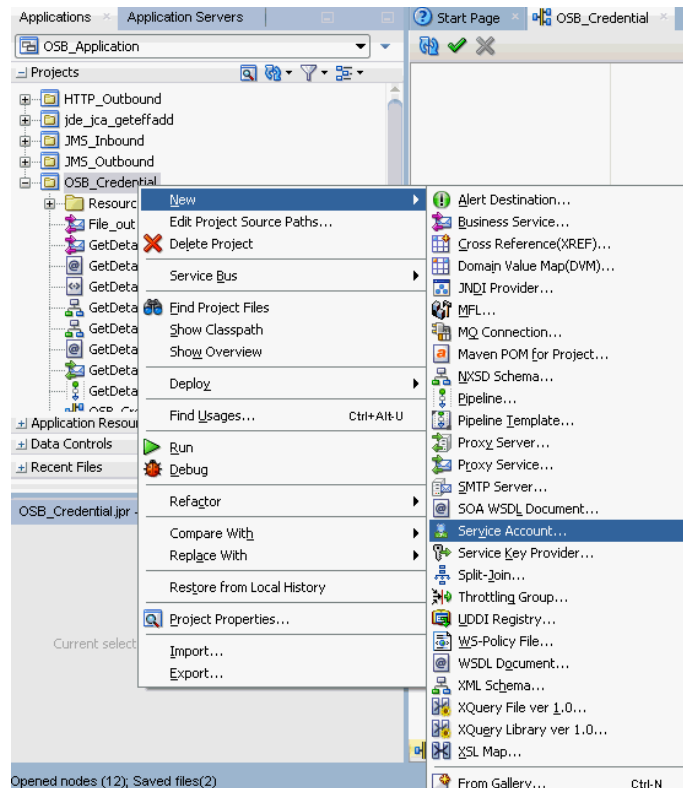
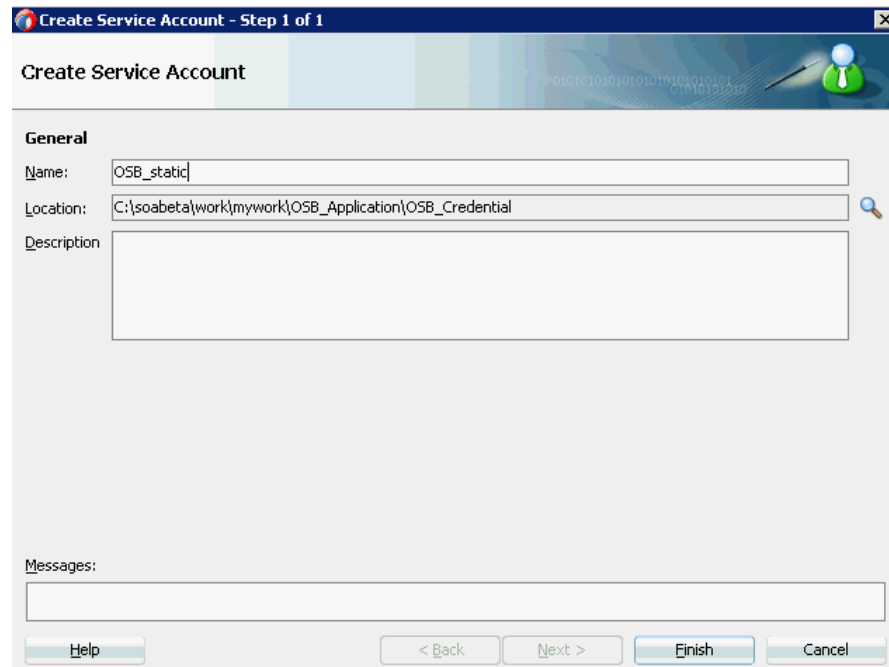


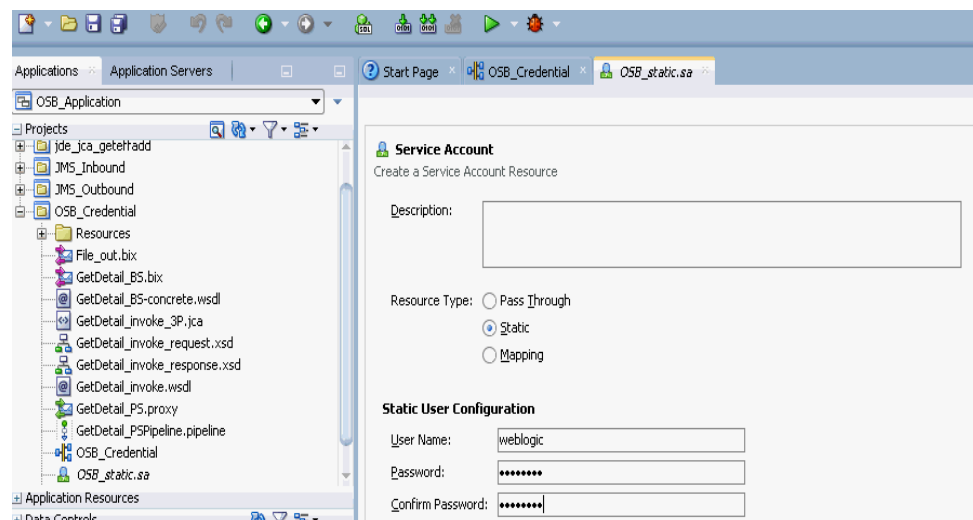
図 11-28 に示すように、「サービス・アカウントの作成」ペインが表示されます。

図 11-28 「サービス・アカウントの作成」ペイン



8. サービス・アカウントの名前を入力し、「終了」をクリックします。  
サービス・アカウントの構成ページが表示されます。
9. 「リソース・タイプ」セクションで、「静的」を選択します。
10. 図 11-29 に示すように、Oracle WebLogic Server 用の有効なユーザー名およびパスワードを指定します。

図 11-29 サービス・アカウントの構成ページ



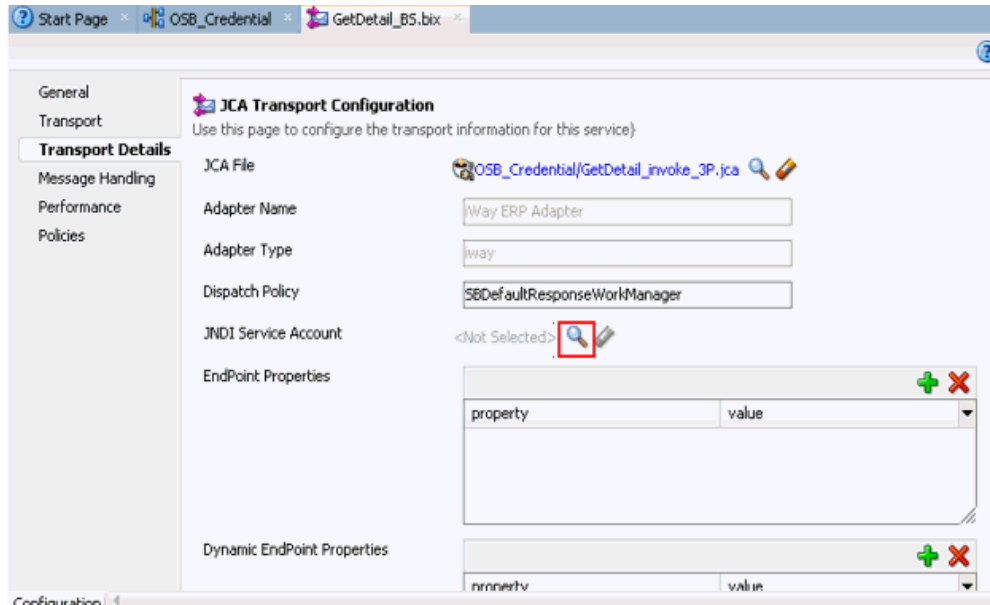
11. 構成ページを保存して閉じます。

- コンポジット・エディタ・ウィンドウで、手順 3 で作成した WSDL ベースのビジネス・サービスをダブルクリックします。

WSDL ベースのビジネス・サービスの構成ページが表示されます。

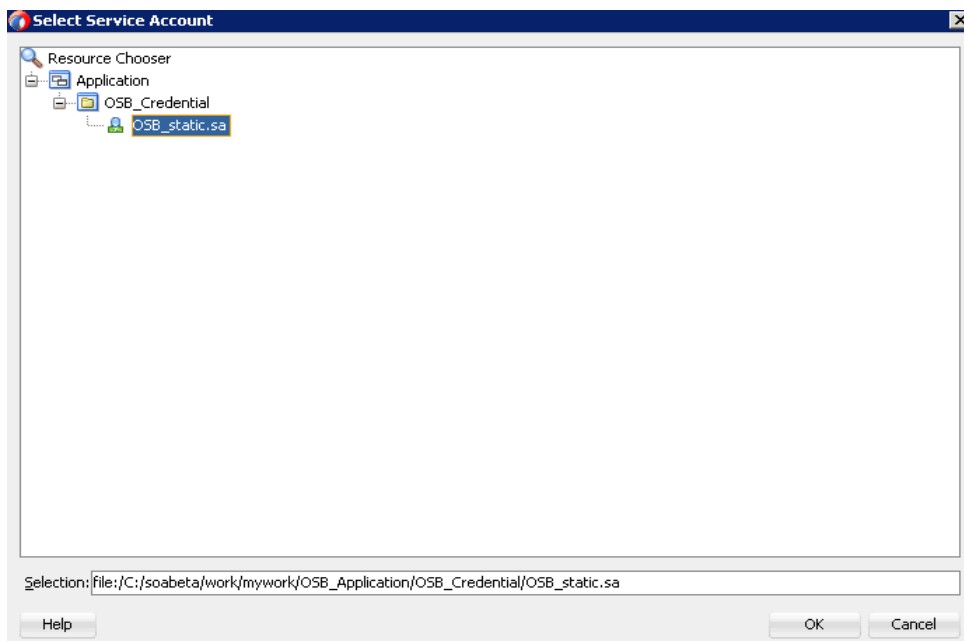
- 図 11-30 に示すように、「トランスポートの詳細」タブを選択します。

図 11-30 「トランスポートの詳細」タブ



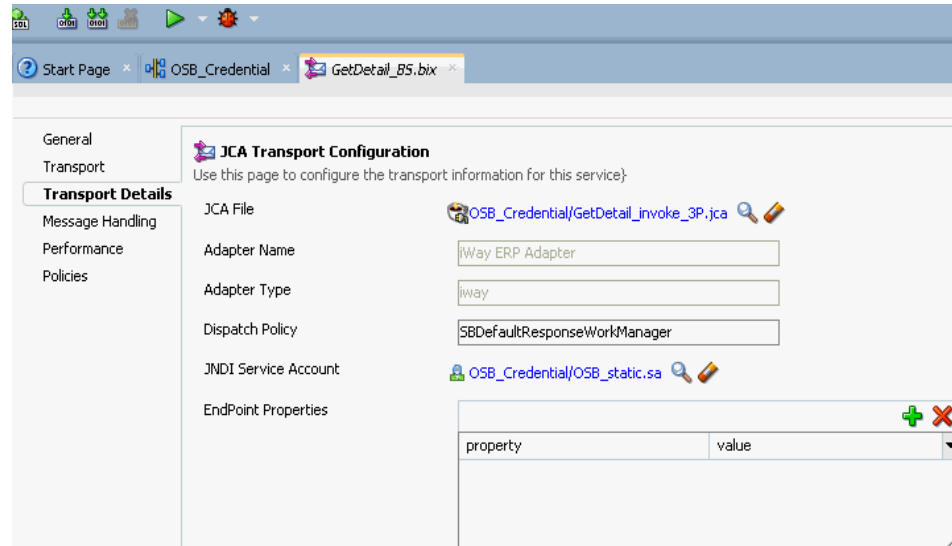
- 「JNDI サービス・アカウント」セクションで、「参照」アイコンをクリックします。  
「サービス・アカウントの選択」ウィンドウが表示されます。
- 図 11-31 に示すように、作成したサービス・アカウントを選択し、「OK」をクリックします。

図 11-31 サービス・アカウントの選択



16. 図 11-32 に示すように、構成ページを保存して閉じます。

図 11-32 ビジネス・サービスの構成ページ



17. OSB プロセスをデプロイします。

詳細は、10-17 ページの 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」を参照してください。

18. プロセスが正常にデプロイされたら、構成した入力フォルダに有効な入力 XML ファイルをコピーして貼り付け、出力が構成済の出力場所で受信されることを確認します。
19. J2CA ログ・ファイルを確認し、ユーザー資格証明が Oracle WebLogic Server を介して EIS に渡されたことを示す暗号化パスワードを特定します。

次に例を示します。

```
FINEST IWAFManagedConnectionFactory com.ibi.afjca.Util getPasswordCredential (78)
InLoop:
User-iwayqa:Password-ENCR (3189319731831132182333215323332323192322731773252)
FINEST IWAFManagedConnectionFactory com.ibi.afjca.Util getPasswordCredential (90)
Use the system PasswordCredential:
User-iwayqa:Password-ENCR (3109313331831131702333215320132323192322731773236)
```

## 11.4 テキストからの Unicode 文字のフィルタリング

この項では、XML 1.0 で無効な Unicode 文字をテキストからフィルタする方法について説明します。ここでは、次の項目について説明します。

- 11.4.1 項「概要」
- 11.4.2 項「フィルタの有効化」
- 11.4.3 項「置換文字の指定」
- 11.4.4 項「設定ファイルの保存およびサーバーの再起動」
- 11.4.5 項「フィルタの検証」
- 11.4.6 項「クイック・スタート・ガイド」
- 11.4.7 項「既知の制限」

### 11.4.1 概要

SAP IDoc データに出現する、XML 1.0 パーサー仕様で許容されない文字を対象とした Unicode 特殊文字フィルタを使用できます。このフィルタを使用して、XML パーサー内で例外が発生する可能性のある、SAP からアダプタへの特定の Unicode 文字が検出されます。次に、このフィルタにより、解析中にエラーが発生しないよう、これらの文字がシリアライズ前にデータ・ストリームから削除されます。

この機能により、次の重大で無効な XML 文字を対象としたフィルタが実装されます。

- VT: 垂直タブ、U+000B
- FF: フォーム・フィード、U+000C
- CR: 改行、U+000D
- NEL: 次の行、U+0085
- LS: 行セパレータ、U+2028
- PS: 段落セパレータ、U+2029

次の文字は、明示的にはフィルタされません。

- LF: 行送り、U+000A
- CR+LF: CR (U+000D) とそれに続く LF (U+000A)

この機能により、SAP アウトバウンド / Oracle インバウンド IDoc 内の指定の Unicode 文字がすべて、特定の置換文字で置き換えられます。アダプタ・インストールごとにシステム全体の置換文字が 1 つ存在します。この機能は、設定ファイルおよび指定の置換文字によって制御されます。

### 11.4.2 フィルタの有効化

XML エディタまたはテキスト・エディタを使用して、`mysap.settings.xml` ファイルを開きます。`mysap.settings.xml` ファイルは次のディレクトリにあります。

```
<ADAPTER_HOME>\lib\settings
```

`filtering` タグにより、フィルタ動作が制御されます。`enabled` を `on` に設定すると、受信 SAP データがフィルタされます。`enabled` を `off` に設定すると、フィルタ動作が無効になります。`enabled` フィールドにその他の値を入力しないでください。



### 11.4.3 置換文字の指定

`substitutionChar` フィールドにより、IDoc 内の無効な文字のかわりに表示される文字が制御されます。

# (シャープまたはハッシュ) 文字が SAP の置換文字であることを念頭に置き、データ内に本来出現しないような、十分に一意性のある置換文字を設定してください。# (シャープまたはハッシュ) 文字を置換文字として使用できますが、場合によっては SAP 自体で同じ文字が使用される可能性があることを覚えておいてください。SAP 置換文字は、SAP サーバーでの RFC Destination maintenance トランザクション (SM59) の「MDMP & Unicode」タブで設定されています。

次に、サンプル・ファイルを示します。

```
<mysap>
  <filtering>
    <enabled>on</enabled>
    <substitutionChar>'#</substitutionChar>
  </filtering>
</mysap>
```

---

**注意:** `mysap.settings.xml` ファイルの  
`<substitutionChar></substitutionChar>` 要素タグで置換文字が指定  
 されていない場合、デフォルトの置換文字として空白が使用されます。

---

### 11.4.4 設定ファイルの保存およびサーバーの再起動

選択したアプリケーション・エディタを使用して `mysap.settings.xml` ファイルに加えた変更を保存します。Oracle Application Adapter for SAP R/3 アダプタでは、アクティブ化時に設定ファイルが自動的に配置されます。

`mysap.settings.xml` ファイルに加えた変更をアクティブ化するには、サーバーを再起動します。

### 11.4.5 フィルタの検証

アダプタのアクティブ化時に、`DEBUG` レベルがログで有効になっている場合、設定ファイルが配置されているかどうか、およびフィルタ・オプションが有効になっているかどうかを示すメッセージが、ログに表示されます。このメッセージは、標準のトレース・レベルでは表示されません。

### 11.4.6 クイック・スタート・ガイド

この項では、フィルタ機能を使用するために必要となる重要な手順について説明します。

1. 次のフォルダにある `mysap.settings.xml` ファイルを参照します。

```
<ADAPTER_HOME>\lib\settings
```

2. `mysap.settings.xml` ファイルを編集し、`filtering` タグの `enabled` オプションを `on` に設定します。
3. 11-29 ページの 11.4.3 項「置換文字の指定」の説明に従って、置換文字を指定します。
4. `mysap.settings.xml` ファイルを保存します。
5. サーバーを再起動します。
6. ログで、`DEBUG` レベルが有効になっているかどうかを確認します。有効になっている場合、設定ファイルが配置されているかどうか、およびフィルタ・オプションが有効になっているかどうかを示すメッセージが、ログに表示されます。

## 11.4.7 既知の制限

この項では、Unicode 特殊文字フィルタの既知の制限について説明します。ここでは、次の項目について説明します。

- 11.4.7.1 項「アダプタ全体でのインストール」
- 11.4.7.2 項「全置換の制限」
- 11.4.7.3 項「SAP システム・タイプ」
- 11.4.7.4 項「文字制限」
- 11.4.7.5 項「ドキュメント・タイプおよび方向」
- 11.4.7.6 項「空白ルールの表示」
- 11.4.7.7 項「SAX パーサー」
- 11.4.7.8 項「置換文字タイプ」
- 11.4.7.9 項「置換文字数」
- 11.4.7.10 項「フォントの使用状況」
- 11.4.7.11 項「空白文字の表示」

### 11.4.7.1 アダプタ全体でのインストール

この機能はアダプタのインストールごとに動作し、アダプタ・チャンネル・ターゲットごとには動作しません。機能をデプロイする前に、アダプタの構成済チャンネルすべてが要件を満たしていることを確認してください。

### 11.4.7.2 全置換の制限

置換文字は、IDoc ドキュメント内の指定の置換文字をすべて置換します。詳細は、11-29 ページの 11.4.3 項「置換文字の指定」を参照してください。

### 11.4.7.3 SAP システム・タイプ

この機能は、Unicode SAP システム専用設計されています。SAP MDMP システムでは言語のコード・ページが使用されており、通信の両側で Unicode が使用されていない場合、Unicode 文字は、RFC データ・ストリームを正しく通過できないことがあります。

### 11.4.7.4 文字制限

この機能は、このマニュアルで明示的にリストされている文字についてのみ有効であり、その他の文字については無効です。IDoc ドキュメントには様々な制御文字が含まれる可能性があります。これらの制御文字の処理に関するルールが存在しないため制御文字はフィルタを通過します。

### 11.4.7.5 ドキュメント・タイプおよび方向

この機能は、Oracle インバウンド IDoc への SAP アウトバウンドに制限されています。これらのドキュメントはシステム間通信のために設計されており、XML 通信には無効な SAP 以外のシステムまたは他のシステムからの文字が含まれる可能性があります。SAP リモート関数呼出しを介して配信される、SAP アウトバウンド・イベント・データはフィルタされません。Oracle から SAP に送信されるデータ (リクエスト/レスポンスまたはクライアント通信) が、この機能によって影響を受けることはありません。データの所有者は、SAP に送信されるデータの正確性に責任があります。

### 11.4.7.6 空白ルールの表示

Java Unicode 定義で *is whitespace* 属性が *true* に設定されている文字は、RAW テキスト (XML ではない) 形式で文字の UTF-8 の 16 進コードを使用した場合にのみ表示可能です。Unicode 文字のガイドについては、次の URL を参照してください。

<http://www.fileformat.info/unicode/char/search.htm>

詳細は、11-31 ページの 11.4.7.11 項「空白文字の表示」を参照してください。

### 11.4.7.7 SAX パーサー

文字シーケンス *CR+LF* は、それが IDoc テキスト内に含まれる場合であってもフィルタされることはありません。XML IDoc をテキスト・エディタで表示すると、*LF* 文字のみが見つかりますが、これは通常は SAX パーサーによって処理されます。詳細は、次のルールを参照してください。

<http://www.w3.org/TR/REC-xml/#sec-line-ends>

このルールにより、*U+000D+U+000A* は単一の *U+000A* 文字に変換されます。文字シーケンス *U+000D* は、アダプタによってフィルタされます。

### 11.4.7.8 置換文字タイプ

置換文字は、ISO/IEC 646 で定義されている標準文字セットのものである必要があり、C0 や C1 の制御文字自体にすることはできません。通常、制御文字は ASCII の句読点および記号のセットから選択され、1 バイト文字で表現される必要があります。置換文字は一重引用符で囲む必要があります。

### 11.4.7.9 置換文字数

アダプタのインストールごとにフィルタ・ファイル内で使用できる置換文字は 1 つのみです。

### 11.4.7.10 フォントの使用状況

データを受信するシステムに、受信データに適切な言語パックおよび文字セットが含まれていることを確認してください。受信システム上のフォントにない文字データを表示する試行が送信されると、ボックス文字が表示されて、ラスター・フォントまたはプリンタ・フォントに表示対象文字をレンダリングするための文字ポイントがないことが示されることがあります。

### 11.4.7.11 空白文字の表示


文字 *U+0085* (次の行) は、XML を特別なテキスト・エディタで調べ、その文字が存在することが想定される空白の領域を強調表示して 16 進値を取得することによってのみ表示できますが、この場合、 11-33 に示すように、これは UTF-8 の 16 進数である *c285* として表示されます。

図 11-33 空白文字の表示

```

<Z1BP0002_1 SEGMENT="1">CRLF
... <COMP_CODE>0001</COMP_CODE>CRLF
... <COMP_NAME> □ SAP A.G.</COMP_NAME>CRLF
</Z1BP0002_1>CRLF
<Z1BP0002_1 SEGMENT="1">CRLF
... <COMP_CODE>0001</COMP_CODE>CRLF
... <COMP_NAME> C285 SAP A.G.</COMP_NAME>
</Z1BP0002_1>CRLF

```

それ以外の場合、この文字は、XML パーサー・ディレクティブにより空白として無視されるため、ファイルに表示されません。

---

---

## トラブルシューティングおよび エラー・メッセージ

この章では、SAP R/3 に接続する場合の制限および回避策について説明します。次の項で構成されています。

- 12.1 項「トラブルシューティング」
- 12.2 項「BSE エラー・メッセージ」

この章にリストされているアダプタ固有のエラーは、Oracle Adapter J2CA 構成または Oracle Adapter Business Services Engine (BSE) 構成のアダプタを使用した場合に発生する可能性があります。

### 12.1 トラブルシューティング

この項では、次の 3 つの項目で SAP R/3 に関するトラブルシューティング情報を提供します。

- 12.1.1 項「アプリケーション・エクスプローラ」
- 12.1.2 項「SAP R/3」
- 12.1.3 項「Oracle Adapter J2CA」

トラブルシューティングに関連する可能性のあるログ・ファイル情報は、アダプタのインストールに応じた次の場所で見つけることができます。

- Oracle Adapter J2CA トレース情報は、次のディレクトリにあります。

```
<ADAPTER_HOME>\config\configuration_name\log
```

- BSE トレース情報は、次のディレクトリにあります。

```
<ORACLE_HOME>\user_projects\domains\base_domain\servers\soa_
server1\stage\ibse\ibse.war\ibselogs
```

- アプリケーション・エクスプローラのログ・ファイルは、次のディレクトリにあります。

```
<ADAPTER_HOME>\tools\iwae\bin
```

## 12.1.1 アプリケーション・エクスプローラ

ここでは、アプリケーション・エクスプローラを使用しているときに発生する可能性がある様々なタイプのエラーについて説明します。

エラー	解決策
<p>SAP R/3 がアプリケーション・エクスプローラの「アダプタ」ノード・リストに表示されません。</p> <p>アプリケーション・エクスプローラから Oracle Application Adapter for SAP R/3 に接続できません。</p> <p>アダプタのアクティブ化中に問題が発生しました</p> <p>アプリケーション・エクスプローラを介して SAP R/3 ターゲットに接続できません。次のエラー・メッセージが表示されます。</p> <pre>Error getting target [SAP] - java.lang.Exception: Error Logon to SAP System</pre> <p>アプリケーション・エクスプローラを介して SAP R/3 システムに接続できません。次のエラー・メッセージが表示されます。</p> <pre>Problem activating adapter. (com.ibi.sapr3.SapAdapterExc eption: com.sap.mw.jco.JCO\$Exception: (102) RFC_ERROR_COMMUNICATION: Connect to SAP gateway failed Connect_PM GWHOST=isdsvr8, GWSERV=sapgw00, ASHOST=isdsvr8, SYSNR=00 LOCATION CPIC (TCP/IP) on local host ERROR partner not reached (host isdsvr8, service 3300) TIME Fri Aug 27 11:49:14 2004 RELEASE 620 COMPONENT NI (network interface) VERSION 36 RC -10 MODULE ninti.c LINE 979 DETAIL NiPConnect2 SYSTEM CALL SO_ERROR ERRNO 10061 ERRNO TEXT WSAECONNREFUSED: Connection refused COUNTER 1). Check logs for more information</pre> <p>SAP R/3 は実行中であるのにアプリケーション・エクスプローラを介して SAP R/3 システムに接続できません。次のエラー・メッセージが表示されます。</p> <pre>Problem activating adapter. (com.ibi.sapr3.SapAdapterExc eption:java.lang.ExceptionInInitiali zerError:JCO.classInitialize(): Could not load middleware layer 'com.sap.mw.jco.rfc.MiddlewareRFC' JCO.nativeInit(): Could not initialize dynamic link library sapjco3 [no sapjco3 in java.library.path]. java.library.path</pre>	<p>sapjco3.jar ファイルと sapjco3.dll ファイルが lib ディレクトリに追加されていることを確認します。</p> <p>次のことを確認します。</p> <ul style="list-style-type: none"> <li>■ SAP R/3 が実行中であること。</li> <li>■ アプリケーション・サーバー名、システム番号およびクライアント番号が正しいこと。</li> <li>■ SAP R/3 のユーザー ID およびパスワードが正しいこと。</li> </ul> <p>SAP R/3 ターゲットへの接続時に正しい接続パラメータを入力していることを確認します。</p> <p>SAP R/3 が実行中であること、およびアプリケーション・サーバーへの接続に正しいパラメータ値を使用していることを確認します。</p> <p>sapjco3.dll ファイルが lib ディレクトリおよび Windows の system32 フォルダに追加されていることを確認します。</p>

エラー	解決策
<p>dll が別のクラス・ローダーにロードされていません (BSE および J2CA は同じサーバー上にインストールされています)。次のエラー・メッセージが表示されます。</p> <pre>com.ibi.sapr3.SapAdapterException:java.lang.ExceptionInInitializerError:JCO.classInitialize(): Could not load middleware layer'com.sap.mw.jco.rfc.MiddlewareRFC'  JCO.nativeInit(): Could not initialize dynamic link library sapjco3 [Native Library F:\iWay55.008.0628\lib\sapjco3.dll loaded in another classloader].java.library.path</pre>	<p>sapjco3.jar をサーバーのクラス・パスに追加します。</p>
<p>Solaris 環境でアプリケーション・エクスプローラを起動できません。コンソールで次の例外がスローされます。</p> <pre>javax.resource.ResourceException: IWAManagedConnectionFactory: license violation.at com.ibi.afjca.spi.IWAManagedConnectionFactory.createConnectionFactory(IWAManagedConnectionFactory.java:98) at com.iwaysoftware.iwae.common.JCATransport.getConnectionFactory(JCATransport.java:133) at com.iwaysoftware.iwae.common.JCATransport.initJCA(JCATransport.java:69) at com.iwaysoftware.iwae.common.JCATransport.&lt;init&gt;(JCATransport.java:62) at com.iwaysoftware.iwae.common.AdapterClient.&lt;init&gt;(AdapterClient.java:85) at com.ibi.bse.ConfigWorker.run(ConfigWorker.java:41) at java.lang.Thread.run(Thread.java:534)</pre> <p>Could not create the connection factory.</p>	<p>JAVACMD がユーザー・システムで設定されていません。アプリケーション・エクスプローラを起動する前に、次のように JAVACMD をエクスポートします。</p> <p>JAVACMD=&lt;jdk_home&gt;/bin/java (ここで、&lt;jdk_home&gt; は、JDK がインストールされているシステム上のディレクトリです)</p>
<p>ランタイムにログオン失敗エラーが発生します。</p>	<p>SAP R/3 システムへの接続用のパスワードが、ターゲットの作成時またはアプリケーション・エクスプローラの「編集」オプションで指定されていない場合、SAP R/3 には接続できません。接続パスワードは repository.xml に保存されません。アプリケーション・エクスプローラの「編集」オプションを使用してパスワードを更新してから、アプリケーション・サーバーを再起動します。</p>
<p>ae.bat (iaexplorer.bat ではありません) をアクティブ化してアプリケーション・エクスプローラを起動すると、次の例外が発生します。</p> <pre>java.lang.ClassNotFoundException: org.bouncycastle.jce.provider.BouncyCastleProvider</pre>	<p>これは問題のない例外です。アダプタ機能に影響を及ぼすことはありません。次の URL から BouncyCastle ファイルをダウンロードします。</p> <p>ftp://ftp.bouncycastle.org/pub</p>

## 12.1.2 SAP R/3

エラー	解決策
<p>リクエストの実行時に、次のエラー・メッセージが表示されます。</p> <pre>AdapterException:java.lang.Exception : Function module CUSTOMER_ GETDETAIL2 does NOT exist.</pre>	<p>入力 XML ドキュメントの構文を確認し、リモート関数モジュールの名前が正しく、SAP R/3 で使用可能であり、アクティブ化されていることを確認します。</p>
<p>リクエストの実行時に、次のエラー・メッセージが表示されます。</p> <pre>AdapterException:java.lang.Exception : Object type unknown for business object: CUST</pre>	<p>入力 XML ドキュメントの構文を確認して、ビジネス・オブジェクト・タイプが正しく、SAP R/3 に存在し、アクティブ化されていることを確認します。また、ドキュメントの名前が正しいことも確認してください。</p>
<p>リクエストの実行時に、次のエラー・メッセージが表示されます。</p> <pre>AdapterException:java.lang.Exception : Unable to retrieve BAPI name for: CUSTOMER.DETAIL2</pre>	<p>入力 XML ドキュメントの構文を確認し、BAPI の名前が正しく、SAP R/3 で使用可能であり、アクティブ化されていることを確認します。</p>
<p>リクエストの実行時に、次のエラー・メッセージが表示されます。</p> <pre>java.lang.RuntimeException:com.sap.m w.jco.JCO\$AbapException: (126) OBJECT_UNKNOWN: Basic type or extension does not exist.</pre>	<p>入力 XML ドキュメントの構文を確認し、IDoc タイプまたは拡張タイプが正しく、SAP R/3 で使用可能であり、アクティブ化されていることを確認します。</p>
<p>リクエストの実行時に、次のエラー・メッセージが表示されます。</p> <pre>AdapterException:java.lang.Exception : BapiError/BapiAbort: You are not authorized to display customers.</pre>	<p>ユーザー ID に SAP R/3 で構成された正しい権限があることを確認します。詳細は、SAP R/3 管理者に問い合せてください。</p>

**注意：** アクティブ化は、SAP の重要な概念です。アクティブ化状態で存在しないオブジェクトは、システム上に存在するものとして表示されていても使用できません。これは特に、IDOC および SAP ビジネス・オブジェクトにとって重要です。詳細は、SAP ドキュメントを参照してください。

## 12.1.3 Oracle Adapter J2CA

エラー	解決策
<p>Oracle Adapter J2CA 構成への接続を試行すると、アプリケーション・エクスペローラで次のエラー・メッセージが表示されます。</p> <p>JCA を初期化できませんでした</p>	<p>右ペインの「詳細」タブで、「ホーム」フィールドで指定したディレクトリが次の例のような正しいディレクトリを指していることを確認します。</p> <pre>&lt;ADAPTER_HOME&gt;\tools\iwae\bin\..\..\..\</pre>



## 12.2 BSE エラー・メッセージ

ここでは、BSE を使用して Web サービスを処理しているときに発生する可能性がある様々なタイプのエラーについて説明します。

### 12.2.1 BSE での一般的なエラー処理

BSE は、アダプタ・フレームワークへの SOAP ゲートウェイとアダプタのエンジンの両方の役割を果たします。デザインタイムとランタイムの両方で、アダプタを使用する Web サービスが実行されるときに様々な条件により BSE でエラーが発生することがあります。これらの条件とその結果発生するエラーは、特定のアダプタにかかわらず同様に公開されますが、使用されるアダプタによっては、異なった形態で公開されるものもあります。ここでは、アダプタに固有であることを基準として、より一般的なエラー条件を満たす場合に発生する可能性があるものについて説明します。通常、BSE 内の SOAP ゲートウェイ ([エージェント](#)) により、SOAP リクエスト・メッセージが、Web サービスに必要なアダプタに渡されます。エラーが発生した場合、その公開方法は、アダプタおよび API、またはアダプタで使用されるインタフェースによって異なります。いくつかのシナリオにおいて、SOAP ゲートウェイで SOAP フォルトが生成されます。一般に、BSE 内の SOAP エージェントで無効な SOAP リクエストが受信されるたびに、SOAP フォルト要素が SOAP レスポンスで生成されます。SOAP フォルト要素には、フォルト文字列およびフォルト・コードの要素が含まれます。フォルト・コードには、SOAP エージェント・エラーの記述が含まれます。

次の SOAP レスポンス・ドキュメントは、BSE で無効な SOAP リクエストが受信されたときに生成されます。

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Client</faultcode>
      <faultstring>Parameter node is missing</faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

この例では、BSE により、この Web サービスの WSDL に必須の要素が SOAP リクエスト・メッセージで受信されませんでした。

### 12.2.2 アダプタ固有のエラー処理

ここでは、次の項目について説明します。

- [12.2.2.1 項「Oracle Application Adapter for SAP R/3 の無効な SOAP リクエスト」](#)
- [12.2.2.2 項「SOAP リクエストからの空の結果」](#)
- [12.2.2.3 項「SAP R/3 への接続の失敗」](#)
- [12.2.2.4 項「無効な SOAP リクエスト」](#)
- [12.2.2.5 項「Oracle Application Adapter for SAP R/3 リクエストからの空の結果」](#)

アダプタでランタイム中に例外が発生すると、BSE 内の SOAP エージェントにより、SOAP フォルト要素を含む SOAP レスポンスが生成されます。SOAP フォルト要素には、フォルト・コードおよびフォルト文字列の要素が含まれます。フォルト文字列には、アダプタ・ターゲット・システムからのネイティブ・エラーの説明が含まれます。アダプタではターゲット・システムのインタフェースと API が使用されるため、例外が発生するかどうかは、ターゲット・システムのインタフェースまたは API でのエラー条件の処理方法によって異なります。SOAP リクエスト・メッセージが BSE 内の SOAP エージェントによってアダプタに渡され、このリクエストがこのサービスの WSDL に基づいて無効である場合、SOAP フォルトを生成する例外がアダプタで発生することがあります。アダプタで発生するエラー条件をすべて予期することはほぼ不可能ですが、次に、アダプタにより一般的なエラー条件がどのように処理され、Web サービス・コンシューマ・アプリケーションにどのように公開されるかについて説明します。

### 12.2.2.1 Oracle Application Adapter for SAP R/3 の無効な SOAP リクエスト

Oracle Application Adapter for SAP R/3 で、実行中の Web サービスの WSDL に準拠していない SOAP リクエスト・メッセージが受信された場合、次の SOAP レスポンスが生成されます。

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
<SOAP-ENV:Fault>
  <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>Error processing agent [XDSapIfrAgent] - XD[FAIL]
SapIFRException:java.sql.SQLException:
com.ibi.sapjco3.SapCallableStatement: execute() j
java.util.NoSuchElementException</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

### 12.2.2.2 SOAP リクエストからの空の結果

Oracle Application Adapter for SAP R/3 で、SOAP リクエスト・メッセージで渡された入力パラメータを使用して SAP R/3 オブジェクトを実行する場合、そのパラメータが SAP R/3 内のレコードと一致していないと、次の SOAP レスポンスが生成されます。

```
<?xml version="
1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
    <faultcode>SOAP-ENV:Server</faultcode>
    <faultstring>Error processing agent [XDSapIfrAgent] - XD[FAIL] SapIFRException:
java.sql.SQLException:com.ibi.sapjco3.SapCallableStatement: execute()
java.sql.SQLException: JCO Error Key: NO_RECORD_FOUND Short Description:
com.sap.mw.jco.JCO$AbapException: (126) NO_RECORD_FOUND: NO_RECORD_FOUND</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

### 12.2.2.3 SAP R/3 への接続の失敗

Web サービスの実行時に、Oracle Application Adapter for SAP R/3 から SAP R/3 に接続できない場合、次の SOAP レスポンスが生成されます。

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
    <faultcode>SOAP-ENV:Server</faultcode>
    <faultstring>Error processing agent [XDSapIfrAgent] - XD[RETRY]
Connect to SAP gateway failed Connect_PM GWHOST=ESDSUN, GWSERV=sapgw00, ASHOST=ESDSUN,
SYSNR=00 LOCATION CPIC (TCP/IP) on local host ERROR partner not reached (host ESDSUN,
service 3300)
TIME Mon Jun 30 16:01:02 2003 RELEASE 620 COMPONENT NI (network interface) VERSION 36
RC -10 MODULE ninti.c LINE 976 DETAIL NiPConnect2
SYSTEM CALL SO_ERROR ERRNO 10061 ERRNO TEXT WSAECONNREFUSED: Connection refused COUNTER
1</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

### 12.2.2.4 無効な SOAP リクエスト

Oracle Application Adapter for SAP R/3 で、実行中の Web サービスの WSDL に準拠していない SOAP リクエスト・メッセージが受信された場合、次の SOAP レスポンスが生成されます。

```
<?xml version="1.0" encoding="ISO-8859-1"
  ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
    <faultcode>SOAP-ENV:Server</faultcode>
    <faultstring>RPC server connection failed: Connection refused: connect</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

### 12.2.2.5 Oracle Application Adapter for SAP R/3 リクエストからの空の結果

Oracle Application Adapter for SAP R/3 で、渡された入力パラメータを使用して SOAP リクエストを実行する場合、そのパラメータがターゲット・システム内のレコードと一致していないと、次の SOAP レスポンスが生成されます。

---

**注意：** このアダプタの条件では、SOAP フォルトは発生しません。

---

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/1999/XMLSchema">
  <SOAP-ENV:Body>
    <m:RunDBQueryResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
      xmlns="urn:schemas-iwaysoftware-com:iwse"
      cid="2A3CB42703EB20203F91951B89F3C5AF">
      <RunDBQueryResult run="1" />
    </m:RunDBQueryResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



---

---

## リモート処理での SAP のシステム・ロール について

Oracle Application Adapter for SAP R/3 は、3つの異なるメッセージ・タイプ・シナリオおよび2つの通信ロールを使用して SAP とメッセージを交換します。各ロールには、通信を有効にしてメッセージを送信するために、SAP サーバーおよびアダプタで有効にする必要がある構成オプションがあります。この付録では、ロール・タイプ、メッセージ・タイプおよびそれらの関連用語について説明します。SAP 本番システム環境では、特定の構成ロールに対して適切な管理者のみが、通信エントリを変更または作成する権限を持ちます。この付録では、必要なタスクに関する管理者用のチェックリストを提供し、インバウンドおよびアウトバウンド・メッセージ処理の構成のプロセス全体について説明します。この情報は、本番以外のシステムを構成する目的で、または参照目的で使用できます。

この付録は次の項で構成されています。

- [A.1 項「アダプタの機能」](#)
- [A.2 項「SAP インバウンド・メッセージ処理」](#)
- [A.3 項「SAP アウトバウンド・メッセージ処理」](#)

### A.1 アダプタの機能

ここでは、次の項目について説明します。

- [A.1.1 項「ロール」](#)
- [A.1.2 項「クライアント」](#)
- [A.1.3 項「サーバー」](#)

Oracle Application Adapter for SAP R/3 の実装では、Java Native Interface (JNI) を介してネイティブ C 通信レイヤーと通信する Java インタフェースである、SAP Java Connector (JCo) が使用されます。RFC (リモート関数呼出し) API では、SAP システムの対応するコンポーネントと通信するために、CPIC (Common Programming Interface - Communications) および TCP/IP プロトコルが実装されます。各メッセージは、XML に、または XML から RFC 形式にシリアライズされ、SAP への送信のために現在の SAP メッセージ・タイプにマーシャルされます。SAP サーバーによって、メッセージは現在のランタイム・オブジェクトにアンマーシャルされ、オブジェクトが実行されます。レスポンス・メッセージは RFC にマーシャルされ、XML へのアンマーシャルおよびシリアライズのためにアダプタに返されます。このプロセスはシームレスで非常に迅速です。不適切な資格証明、ドキュメント・エンコーディングまたは実際のアプリケーション・エラー (たとえば、指定されたフィールドの値の誤りなど) によってエラーが発生する場合があります。

## A.1.1 ロール

RFC プロトコルは、リクエストのモードに応じた動的プロトコルです。リクエストが RFC 接続ノードへのインバウンドである場合、そのノードの機能はリクエストを満たすサーバーになります。リクエストがノードからのアウトバウンドである場合、そのノードの機能はクライアントになり、サーバーがリクエストに応答するのを待ちます。レスポンスを必要としないアウトバウンド・イベントも、サーバーによって処理されます。

## A.1.2 クライアント

別のシステムへの接続をインスタンス化するシステムはクライアントであり、クライアント・ロールによってリクエスト/レスポンス・メッセージは同期的に処理されます。クライアント・システムは、メッセージ処理を実行しませんが、有効にします。メッセージはクライアントで開始されて終了します。アダプタが Oracle から SAP へメッセージを送信する場合、アダプタはクライアントとして機能します。

クライアント接続のタイプは次のとおりです。

- 単一の SAP アプリケーション・サーバーに接続するアプリケーション・サーバー。
- 分散ロード処理を目的とした SAP アプリケーション・サーバーのグループへの接続のために、SAP ログオン・ロード・バランサ・メッセージ・サーバーに接続する、メッセージ・サーバー。

どちらの接続タイプも接続プールと呼ばれる特別なタイプの接続を使用しますが、接続プールについてはこの付録で後述します。

クライアント・メッセージ・タイプのタイプは次のとおりです。

- XML 属性としてのパラメータを持つオブジェクトスタイルのメッセージである、BAPI メッセージ。
- XML 要素としてのパラメータを持つ RFC スタイルのメッセージである、RFC メッセージ。
- ALE メッセージ・システムを使用して定位置区切りフラット・テキストまたは XML 要素パラメータを送信する Intermediate Documents (IDoc)。

## A.1.3 サーバー

別のシステムからの接続リクエストを受け取るシステムは、サーバーです。サーバーは、メッセージ処理を実行して結果をクライアントに返します。サーバーは、特定のメッセージの処理を実装していない場合、NOT IMPLEMENTED の例外によってメッセージ・リクエストを拒否できます。サーバーは、次の 2 つのタイプのメッセージを処理できます。

- メッセージ・リクエストとそのリクエストのパラメータを含む、インバウンド・メッセージ。
- インバウンド・メッセージ・リクエストを取得し、追加作業を実行するか、別のシステムに渡すか、または追加処理のための処理を実行してから SAP に返す、同期メッセージ。

SAP アプリケーション・サーバーは、同期メッセージ・プロセスが終了するのを待ちしてから、続行します。

サーバー接続のタイプは次のとおりです。

- メッセージ・タイプとパラメータの単純な受信である、処理モード (リクエスト)。
- 同期プロセスを実行するレスポンス。

サーバー・メッセージ・タイプは次のとおりです。

- 要素としてのパラメータを持つ RPC スタイルのメッセージである、RFC メッセージ。
- ALE メッセージ・システムを使用して定位置区切りフラット・テキストまたは XML 要素パラメータを送信する Intermediate Documents (IDoc)。

## A.2 SAP インバウンド・メッセージ処理

ここでは、次の項目について説明します。

- [A.2.1 項「SAP R/3 インバウンド処理の構成」](#)
- [A.2.2 項「論理システムの構成」](#)
- [A.2.3 項「論理システム用の配信モデルの構成」](#)
- [A.2.4 項「パートナ・プロファイルの定義」](#)

リモート関数呼出し (RFC) または Business Application Programming Interface (BAPI) オブジェクト用のリクエスト/レスポンス (クライアント) メッセージを SAP システムに送信するには、一連の有効なログオン資格証明が必要です。必要なパラメータのリストを含む、アダプタ・ターゲットの構成方法の詳細は、2-3 ページの「[SAP R/3 ログオン・パラメータの識別](#)」を参照してください。アダプタ・ターゲットの有効な値については、SAP システム管理者に問い合せてください。

最初のログオンの前に、次の値を入手します。

- 有効な SAP 接続パラメータ。
- SAP 通信ユーザー (セキュリティ上の理由から、SAP GUI DIALOG ユーザーはお薦めしません)。
- アダプタで実行するタスクに対する適切な SAP 認可。

### SAP 認可

サーバーに接続する前に、適切な SAP 認可を取得する必要があります。同じユーザー ID への RFC および DIAG のアクセスを許可しないことが、SAP の推奨されるセキュリティ上の措置です。

RFC アクセスを保護するために使用される基本的な認可オブジェクトは、次のとおりです。

**S\_RFC:** 関数グループを保護します (認可オブジェクト・クラス AAAB 内にあります)。

- **認可オブジェクト+S\_RFC:**
  - 認可フィールド: RFC\_TYPE
  - 認可値: FUGR のみが有効な値です。
  - 認可フィールド: RFC\_NAME
  - 認可値: アクティブ化された関数グループの名前。例: ("ABCD") や [\*]。
  - 認可フィールド: ACTVT
  - 認可値: 16 (実行) のみが有効な値です。

個々の関数モジュールの実行可能コード内に、追加のセキュリティが組み込まれている場合があります。関数またはアプリケーションのユーザーにその関数またはアプリケーションを使用するための認可要件を通知することは、関数開発者またはアプリケーション作成者の要件です。必要な追加の認可があるかどうかを判別するには、管理者に問い合せてください。

次の認可オブジェクトを使用して、個々の表をアクセスから保護できます。

**S\_TABU\_DIS:** 表を保護します (認可オブジェクト・クラス BC\_A 内にあります)。

- **認可オブジェクト+S\_TABU\_DIS:**
  - 認可フィールド: DICBERCLS: または (「認可グループ」)。
  - 認可値: たとえば、テーブル MARA はグループ MA 内にあります。
  - 認可フィールド: ACTVT
  - 認可値: 03 (表示)

この認可オブジェクトの使用の詳細は、SAP システムのドキュメントを参照してください。

アダプタ・ターゲットに対して適切な資格証明が指定され、SAP アプリケーション・システムで適切な認可が構築されると、アダプタを初期化する準備は完了です。

### リクエスト/レスポンス(クライアント)メッセージ

ビジネス・オブジェクト・リポジトリ

アプリケーション・エクスプローラを使用して適切なアプリケーション・ツリー・ブランチ (*Financial Accounting* など) を展開し、ビジネス・オブジェクト (*Company Code* など) にナビゲートして展開します。オブジェクトをクリックして、そのオブジェクトのメソッドを表示します。BAPI には、リリースとバージョンがあります。ランタイムに使用されるすべての BAPI はリリース・ステータスである必要があり、そうでない場合は実行時例外が発生します。XML スキーマを生成するメソッドを選択します。XML スキーマから XML インスタンス・ドキュメントを作成し、これにデータを移入します。適切な入力と出力 (ファイル /html/ その他) の伝送方法を使用して、Oracle プロセスを作成します。処理のためにドキュメントをアダプタに送信し、結果を受信します。

リモート関数呼出し

アプリケーション・エクスプローラを使用して適切なアプリケーション・ツリー・ブランチ (*Financial Accounting* など) を展開し、関数グループ (*0002 - Company Code* など) にナビゲートして展開します。関数ごとに異なる処理メソッドが公開されています。すべての BAPI は、RFC を介して実装されるため、RFC オブジェクトとして存在します。これらは、たとえば BAPI 処理が不可能なときに、イベント処理が可能となるように呼び出すことができます。すべての RFC が BAPI として存在するわけではありません。関数を選択して XML スキーマを生成します。XML スキーマから XML インスタンス・ドキュメントを作成し、これにデータを移入します。適切な入力と出力 (ファイル /html/ その他) の伝送方法を使用して、Oracle プロセスを作成します。処理のためにドキュメントをアダプタに送信し、結果を受信します。

ALE IDOC メッセージ

SAP ERP ALE\* インバウンド・メッセージ処理では、IDoc が SAP R/3 システムに送信されて格納されますが、使用される SAP サーバーで一定の構成が必要です。このロールでは、メッセージはトランザクション・データとして SAP に送信され、SAP サーバー上に格納されます。ドキュメント・データは、トリガーされたワークフロー中に、SAP 内の第 2 手順で処理されます。

## A.2.1 SAP R/3 インバウンド処理の構成

この項では、次の項目について説明します。

### ■ A.2.1.1 項「アダプタのチェックリスト」

IDoc を SAP ERP に送信するには、メッセージ・パーティに関する次の情報が必要です。

- メッセージの送信者
- メッセージの受信者
- 送信されるメッセージのタイプ

送信者および受信者は ALE システムで定義する必要があり、各パーティがどのようなメッセージを受け入れ、それがどのように処理されるかをパーティごとに定義する必要があります。

接続情報には物理的な接続情報は含まれていないため、送信システムと受信システムの情報は論理システムと呼ばれるユニット内で定義されます。論理システムは、特定の送信者または受信者に関するすべての情報のコンテナとして機能します。

論理システムが定義されると、このシステムで送受信される様々なメッセージに関する情報が配信モデルに格納されます。このモデルは処理が発生する前に定義する必要があり、送信者または受信者のエラーを防ぐために正確に定義する必要があります。配信モデルでは、IDoc メッセージ・アプリケーションでフィルタを有効にすることもでき、IDoc の特定の部分のみを送信または配信することが可能になります。論理システム、配信モデルおよびメッセージ・タイプの間には、1 対 1 の対応があります。フィルタ・グループ、フィルタ・オブジェクト、ルールおよび依存関係によって、複雑なフィルタ・ルールを構成できます。配信モデルでのフィルタの構成の詳細は、SAP システムのドキュメントを参照してください。



SAP システムによる IDoc の処理は、パートナー・プロファイルで定義されます。パートナーおよびメッセージごとに、メッセージを様々な処理できるプロセス・コードを定義することが可能です。

### A.2.1.1 アダプタのチェックリスト

SAP にメッセージを送信するには、ALE 管理者は (ログオン資格証明の後に) 次の割当てを行う必要があります。

1. 論理システム名。
2. SAP アプリケーション・サーバーに送信するメッセージ・タイプおよび IDoc 基本タイプ。  
SAP ERP システム内で、ALE 管理者は次のことを行う必要があります。
  1. 論理システムを作成します。
  2. 配信モデルを作成します。
    - a. モデルの送信者を割り当てます。
    - b. モデルの受信者を割り当てます。
    - c. メッセージ・タイプを割り当てます。
  3. 論理システムのパートナー・プロファイルを作成します。
    - a. メッセージ・タイプごとに、メッセージを処理するための SAP のプロセス・コードを割り当てます。

#### 例

- 論理システム ORACLIENT が作成されます (SAP トランザクション SALE を使用)。
- 配信モデル ZORACLNT が作成されます (SAP トランザクション BD64 を使用)。
- 送信パートナー ORACLIENT および受信パートナー T90ALECNT (中央ホストの ALE コンポーネント) 用にメッセージ・タイプ MATMAS が追加されます。
- パートナー・プロファイル ORACLIENT が作成され (SAP トランザクション WE20 を使用)、インバウンド・メッセージ MATMAS にプロセス・コード MATM が割り当てられます。

---

**注意:** ここでは、インバウンド・メッセージ用に SAP ERP を構成するための手順を参考用に説明しています。より包括的な情報は、SAP ERP のドキュメントを参照してください。

---

インバウンド IDoc 処理用に SAP R/3 を構成するには、次のようにする必要があります。

1. 論理システムを構成します。
2. 配信モデルを構成します。
3. インバウンド・パートナー・プロファイルを定義します。

## A.2.2 論理システムの構成

ここでは、次の項目について説明します。

- [A.2.2.1 項「SAP トランザクションのショートカット」](#)
- [A.2.2.2 項「論理システムの定義」](#)

論理システムは、メッセージの送信者または受信者のパーティに関する構成情報のためのコンテナ・オブジェクトです。これが論理システムである理由は、接続情報が格納されず、エンド・ポイントで処理可能なメッセージおよびメッセージの処理に使用されるメソッドに関する情報のみが格納されているためです。

### A.2.2.1 SAP トランザクションのショートカット

- */n* - 現在のトランザクションを終了し、新しいトランザクションをただちに開きます。
- */o* - 現在のトランザクションのコンテキストを維持し、新しいウィンドウで新しいトランザクションを開きます。

### A.2.2.2 論理システムの定義

論理システムの定義を作成するには、SAP Implementation Guide for Customizing (IMG) でエントリを作成します。

ALE は、SAP ERP Application Link Enablement システムであるため、*/nSALE* トランザクションを使用して ALE 用の IMG にナビゲートします。

1. SALE トランザクションにナビゲートします。  
「Display IMG」ウィンドウに ALE の構成ツリーが表示されます。
2. 次のようにします。
  - a. 「Basic Settings」ノードを展開します。
  - b. 「Logical Systems」を展開します。
  - c. 「Define Logical System」をクリックします。
  - d. SAP R/3 ユーザーの場合、画面が若干異なる可能性があり、特別なノードである「Sending and Receiving Systems」を展開します。  
「Caution: The table is cross-client」というメッセージが表示されます。  
このメッセージは、論理システム内のカスタマイズ・データへの変更が SAP ERP サーバーのすべてのユーザーに適用されることを通知するために表示されます。これは、明確にログオン・クライアントにバインドされるほとんどのアプリケーション・データとは異なります。論理システムのエントリは一切変更しないでください。新しい情報の追加のみを行います。これを行うには、正しい権限を持つ SAP ログオン ID を使用します。
3. 緑色のチェック・マークをクリックして続行します。  
「Change View "Logical Systems"」ウィンドウが表示されます。
4. 「New Entries」ボタンをクリックします。  
「New Entries: Overview of Added Entries」ウィンドウが表示されます。
5. 「Log.System」列に論理システム (ORACLETDS など) を入力し、「Name」列に説明を指定します。

---

**注意：**これらのフィールドは、必須フィールドであることがわかるよう黄色で強調表示されています。

---

6. 上部のメニュー・バーの「Save」アイコンをクリックします。  
「Prompt for Workbench request」ダイアログが表示されます。

---

**注意:** システムの使用状況によっては、最終ワークベンチのリクエスト番号がダイアログに表示されます。これを使用せず、次の手順に進みます。

---

7. 「**New Request**」アイコンをクリックして、新しいリクエストを作成します。
8. リクエスト・システムで、エントリの説明を入力します。残りのフィールドは自動的に入力され、変更する必要はありません。  
「**Prompt for Workbench Request**」ダイアログが再び表示され、新しいリクエスト番号にはワークベンチで作成された最終リクエスト番号が入力されていますが、この番号は使用しません。
9. チェック・マークをクリックして、新しいリクエスト番号を作成します。  
「**Logical System Entries**」ウィンドウが新しい論理システム情報でリフレッシュされ、新しい論理システムが青色で強調表示され、この手順が正常に完了したことが示されます。

## A.2.3 論理システム用の配信モデルの構成

この項では、次の項目について説明します。

### ■ A.2.3.1 項「配信モデルの定義」

配信モデルには、送信者論理システム、受信者論理システムおよび送受信されるメッセージのタイプが含まれます。指定された送信者または受信者に有効な配信モデルがない場合、メッセージは処理されず、エラー位置にルーティングされます。SAP BAPI オブジェクトやフィルタリングを含め、配信モデルの詳細なトピックについては、SAP のドキュメントを参照してください。

#### A.2.3.1 配信モデルの定義

配信モデルを定義するには、次のようにします。

1. **bd64** トランザクションを実行します。  
「**Display Distribution Model**」ウィンドウがデフォルトの表示モードで表示され、現在のモデル・ビューのエントリが表示されます。
2. ツリーを展開して、各モデル・ビューのエントリを表示します。  
アイテムを追加するには、トランザクションを変更モードに切り替える必要があります。
3. 「**Distribution model**」をクリックし、「**Switch processing mode**」を選択します。  
画面がリフレッシュされ、モデル・ビュー・ウィンドウが「**Change Distribution Model**」に切り替わります。
4. 使用可能なメニュー・ボタンから、「**Create Model View**」を選択します。  
「**Create Model View**」ダイアログが表示されます。
5. 「**Short text**」フィールドにモデル・ビュー名を入力し、「**Technical name**」フィールドに説明としての役割も果たす名前を入力します。  
文字 **Z** で始まる名前を使用することをお勧めしますが、これは、SAP がシステム更新時にこれらの名前を上書きしないためです。これは、モデル・ビューの名前についてのみです。
6. 緑色のチェック・マーク・アイコンをクリックして、情報を入力します。  
画面がリフレッシュされ、「**Change Distribution Model**」メイン・ウィンドウに戻ります。構成された配信モデルがリストに追加されました。  
モデル・ビューは、「**Save**」アイコンをクリックするまでトランザクションの一時記憶域に置かれます(エントリは明るい色で表示されます)。システム接続が切断またはタイムアウトすると、エントリは消失します。ベスト・プラクティスとして、配信モデルの構成プロセス中は、変更内容が失われないように各ダイアログの後に変更を保存します。

7. 以前に定義したエントリにカーソルを置きます。  
エントリが強調表示されます。
8. 中央のメニュー・バーで、「Add Message Type」ボタンをクリックします。  
「Add Message Type」ダイアログが表示されます。
9. 次のようにします。
  - a. 「Sender」フィールドで、Oracle システムの送信論理システムの名前を入力します (ORACLETDS など)。詳細は、A-6 ページの「論理システムの構成」を参照してください。
  - b. 「Receiver」フィールドで、SAP システムのクライアント ALE 論理システムの名前を入力するか (ALE 受信者)、ドロップダウンから選択します (T90CLNT090 など)。
  - c. 「Message type」フィールドで、使用するメッセージ・タイプを入力します (MATMAS など)。  
各フィールドの右側にあるドロップダウン・アイコンをクリックして、使用可能な値をリストから参照できます。
  - d. 緑色のチェック・マーク・アイコンをクリックして、情報を入力します。  
「Change Distribution Model」メイン・ウィンドウがリフレッシュされます。
10. 「Save」をクリックします。
11. ALE 階層ツリーで、作成した配信モデルにナビゲートします。送信者および受信者は、送信者、受信者およびメッセージの関係が視覚的にわかるように示されます。  
同じシステムのその後のメッセージ・タイプについて、モデル・ビューを展開して受信者名を表示し、名前にカーソルを置きます。次に、「Add Message Type」ボタンをクリックします。すべてのフィールドに自動的に情報が入力されます。これでメッセージ・タイプを追加できます。

## A.2.4 パートナ・プロファイルの定義

パートナ・プロファイルは、論理システムごとに、処理するメッセージとメッセージの処理方法を定義します。これにより、パートナの処理の詳細を柔軟に構成することが可能となります。パートナ・プロファイルには、メッセージ・タイプごとにエントリを1つのみ含めることができます。パートナ・プロファイルを定義するには、次のようにします。

1. 画面の左上にあるコマンド・バーに /nwe20 と入力し、**we20** トランザクションを実行します。  
「Partner profiles」ウィンドウが表示されます。
2. 「Partner Type LS」を選択して、中央のメニュー・バーの白色のアイコンをクリックします。上部のメニューの「Create」を選択することもできます。  
画面の右側の表示が変わり、パートナ・プロファイルの未記入フォームが表示されます。
3. パートナ番号フィールドに論理システムの名前 (ORACLETDS など) を入力し、「Save」アイコンをクリックします。  
画面が新しいプロファイルによって変更されます。
4. 埋込みテキスト「Inbound parmtrs」が表示されるまで、下にスクロールします。
5. 緑色のプラス記号のアイコンをクリックします。  
「Partner profiles: Inbound parameters」ダイアログが表示されます。
6. 「Message Type」として **MATMAS** を入力します。
7. 「Process code」ドロップダウンを使用して、**MATMAS** を表す **MATM** を見つけて選択します。
8. 「Process Mode」として「Immediate」または「Background」を選択します。

---

**注意:** 適切な設定については、SAP 管理者に問い合せてください。

---

9. メニュー・バーの「**Save**」アイコンをクリックして、MATMAS のインバウンド・パートナ・プロファイルのエントリを保存します。
10. メニュー・バーの「**Back**」(左矢印)ボタンをクリックして、前のウィンドウに戻ります。MATMAS がエントリ・リストに表示されます。

これで、論理システム/パートナ・プロファイル (OracleTDS/MATMAS) のインバウンド構成は完了です。インバウンド IDoc メッセージ・タイプのエントリを追加するには、A-7 ページの「[論理システム用の配信モデルの構成](#)」から始まる手順を繰り返します。別の論理システムおよびメッセージ・タイプを追加するには、A-6 ページの「[論理システムの構成](#)」からの手順を繰り返します。

これで、適切な Oracle プロセスの構成後に、IDoc の送信を開始できます。

SAP IDoc 監視コンソールから、SAP トランザクション **WE02** で IDoc トランザクションを監視できます。IDoc の処理状態およびステータスが、送信者、受信者およびデータとともに使用可能です。詳細は、SAP のオンライン・ヘルプ・ページにある「[IDoc Display](#)」のトピックを参照してください。

## A.3 SAP アウトバウンド・メッセージ処理

Oracle Application Adapter for SAP R/3 では、リモート関数モジュール、BAPI ALE のインタフェース・スタイルを使用した SAP メッセージからのアウトバウンド、またはトランザクション ID を持つトランザクション RFC メッセージを処理できます。メッセージ・タイプは、リモート関数呼出しまたは IDoc です。SAP JCo を介したすべての処理は同期 (リクエスト/レスポンス) ですが、メッセージ処理は非同期 (配信のみ) または同期 (イベント/返信) であり、この場合は編成されたプロセスが作業を実行して、待機中のアプリケーション・サーバーに情報を返します。RFC メッセージは、SAP ABAP プログラムまたは SAP ワークフロー・システムから送信できます。IDoc は、システム・トランザクションまたは標準プログラムで送信されます。SAP での BAPI イベントは、ALE インタフェース用に特別にプログラムされ、有効になっている場合を除いて内部用です。

SAP でのイベントは、イベント処理の作業を行うイベント・ハンドラへのイベント・リンクによって有効になっています。イベントを送信するメカニズムはすべて SAP システムで有効になっており、アダプタは SAP イベント・ハンドラの終了プロセスを受け取ります。

アウトバウンド・メッセージは、アダプタではイベントとして定義されます。イベントは、非同期 (メッセージを配信するのみ) または同期 (返信を返す) です。アダプタのインバウンド・イベントが双方向 (イベント - 返信) になるには、受信プロセスでプロセス・フローにリクエスト・ハンドラが実装される必要があります。リクエスト・ハンドラは、受信 SAP データを受け取り、それを操作して、SAP に返すために有効なイベント返信フォーマットでデータをアダプタに返します。

RFC アウトバウンド・メッセージの場合、リモート関数呼出しを受信するために必要になるものは、SAP RFC 宛先とプログラム ID のみです。IDoc 処理の場合、RFC でメッセージ・イベントを受信するには、論理システム、配信モデル、パートナ・プロファイルおよびポートを作成する必要があります。

### SAP アウトバウンド用語

- イベント作成者 - JCO/RFC クライアントであり、これは、イベント (リクエスト) を作成し、オプションでイベント返信 (レスポンス) を待機する SAP システムです。
- イベント・リスナー - JCO/RFC サーバーであり、これは、イベントを受信し、リクエストを処理し、オプションでレスポンスを返すアダプタです。

ここでは、次の項目について説明します。

- **A.3.1 項「SAP ゲートウェイ」**
- **A.3.2 項「プログラム ID とロード・バランシング」**
- **A.3.3 項「SAP GUI での RFC 宛先とプログラム ID の登録」**
- **A.3.4 項「イベント・アダプタ用の Application Link Embedding 構成」**
- **A.3.5 項「ポートの定義」**
- **A.3.6 項「論理システムの構成」**
- **A.3.7 項「論理システム用の配信モデルの構成」**
- **A.3.8 項「パートナ・プロファイルの定義」**
- **A.3.9 項「SAP R/3 ALE 構成のテスト」**

多くの場合、アウトバウンド処理はイベントとして実装されます。SAP アプリケーション・サーバーには、定義済イベントの多数のステージがあります。イベント処理はリンクによって結合されます。本番 SAP サーバーでは、すべてのリンクがデフォルトで有効になっているわけではありません。イベントをアクティブにしてメッセージを送信できるようにするには、オブジェクトのステータス変更を送信できるオブジェクト、有効なイベントのリストへのリンクおよびイベント・ハンドラが必要です。イベント・ハンドラによってメッセージの処理が実装されます。

アクティブな SAP イベントは、イベント・ハンドラが実装された定義済のコンポーネントでのステータス変更です。SAP アプリケーション・サーバーには多数のイベント・タイプがあり、最もよく知られているイベント・ハンドラは ALE Message Control で、SAP によって Material Master や Customer Master などのオブジェクトに関して実装されます。これらのオブジェクトにおける変更によってアウトバウンド・イベント IDoc を生成できます。

イベントの作成は通常は SAP によって実装されますが、多数のオブジェクトをイベント処理に対して有効にできるものの、多くの場合、これには高度な構成やプログラミングが必要になります。SAP イベントのタイプと SAP イベントの構成方法の詳細は、SAP のドキュメントを参照してください。

RFC アウトバウンド・メッセージの場合、リモート関数呼出しを受信するために必要になるものは、SAP RFC 宛先とプログラム ID のみです。

IDoc 処理の場合、tRFC でメッセージ制御イベントを受信するには、論理システム、配信モデル、パートナ・プロファイルおよびポートを作成する必要があります。

#### SAP アウトバウンド用語

- JCO/RFC クライアントは、RFC リクエストを作成し、オプションでレスポンスを待機するシステムです。
- JCO/RFC サーバーは、RFC リクエストを受信し、リクエストを処理し、オプションでレスポンスを返すシステムです。

### A.3.1 SAP ゲートウェイ

SAP ゲートウェイは、SAP アプリケーション・サーバーまたは専用のゲートウェイ・サーバーで実行されるプロセスです。ゲートウェイ・プロセスによって、SAP へのすべての接続が管理されます。SAP へのサーバー接続は、それらが SAP アプリケーションで登録済プログラムとして事前に構成され、必要なプログラム ID を提示する場合を除いて受け入れられません。

サーバー接続は、それ自体をゲートウェイに提示し、プログラム ID を公開します。プログラム ID が登録済プログラム ID のリストで見つかった場合、ゲートウェイ・サーバーは接続をサーバーに提供し、サーバーは接続を受け入れます。このプログラム ID は SAP 内の RFC 宛先にリンクされ、これによって SAP 関数モジュールと ALE ドキュメント (IDoc または BAPI IDoc) を宛先にルーティングできるようになります。RFC 宛先は、SAP プログラム ID を識別するためのキーとして機能します。また、プログラム ID は、SAP への接続を確立するアプリケーション・サーバーによって公開され、接続サーバーの IP アドレスへの直接リンクになります。メッセージは、RFC 宛先を介してアプリケーション・サーバーにルーティングされます。適切なセキュリティ上の措置は、プログラム ID を実装するアダプタ・インスタンスにアクセスする Oracle Application Adapter for SAP R/3 ユーザーと SAP 内でプログラム ID を変更または表示する SAP トランザクション権限を持つユーザーを分離することであり、すべてのリクエストは RFC 宛先を介してルーティングされます。

---

**注意:** RFC サーバー・プログラムは、SAP ゲートウェイに登録され、受信 RFC コール・リクエストを待機します。RFC サーバー・プログラムは、特定の SAP システムではなく、SAP ゲートウェイでプログラム ID によってそれ自体に登録します。SAPGUI では、接続タイプ T と登録モードを使用して、トランザクション SM59 で宛先を定義する必要があります。さらに、このエントリには、RFC サーバー・プログラムが登録されている SAP ゲートウェイに関する情報が含まれている必要があります。

---

### A.3.2 プログラム ID とロード・バランシング

ここでは、次の項目について説明します。

- A.3.2.1 項「SAP RFC アウトバウンドの概要」
- A.3.2.2 項「SAP IDoc イベント」

ゲートウェイ・サーバーに特定のサーバー・インスタンスへの接続があり、別のサーバー・インスタンスがそれ自体をゲートウェイに提示した場合、ゲートウェイは接続を提供し、ロード・バランシング・モードで機能し始めます。ゲートウェイは、固有のアルゴリズムを使用して、要求と合計処理時間に応じて様々なメッセージを各サーバーに送信します。これにより、スキーマおよびアプリケーションによってメッセージが検証されたときに、予期しない結果が発生する場合があります。

Oracle で単一の SAP プログラム ID を使用して複数のイベントを構成する場合、SAP によってイベント・データはロード・バランシングされます。たとえば、複数のリモート関数呼出しまたは BAPI によって同じプログラム ID (ORACLETDS など) が使用され、複数の SAP リスナーがこのプログラム ID で構成されている場合、SAP によって 1 つのリクエストが 1 つのリスナーに、次のリクエストは別のリスナーにというように送信されます。SAP ゲートウェイ・サーバーには、ロード・バランシング・アルゴリズムがあります。このメカニズムは SAP アプリケーション開発に固有であり、接続の合計スループット、待機状態になった回数などを比較することで機能します。1 つの接続が 9 個のメッセージを受信し、2 番目の接続が 1 個のメッセージを受信する場合があります。9 個のメッセージのうちの 5 個がスキーマの検証に対して拒否され、もう一方の接続の 1 個のメッセージがスキーマの検証に対して拒否された場合、SAP イベント処理メッセージが欠落している疑いがあります。

サーバー (SAP からアダプタへのインバウンド) 状況でのロード・バランシングは、アダプタの複数のインスタンスを SAP システムに接続することによって処理されます。SAP システムによって接続はロード・バランシングされます。このパフォーマンスは調整できません。



### A.3.2.1 SAP RFC アウトバウンドの概要

1. RFC 宛先を作成します (SM59 を使用)。
  - a. 宛先名を入力します。
  - b. 登録されているプログラム ID を入力します。
  - c. 「MDMP & Unicode」タブで、「Unicode」オプションが選択されていることを確認します。
  - d. Oracle BPEL またはメディエータを起動します。
  - e. BPEL またはメディエータのチャンネルの手順 b から、同じ登録済プログラム ID 値を入力します。
  - f. チャンネルを起動します。
  - g. SAP で「Test」ボタンを使用して接続をテストします。
2. SAP RFC メッセージをシミュレートします (SE37 を使用)。

ABAP ワークベンチ認可が必要です。

  - a. 手順 1a で定義した RFC 宛先名を「Destination」フィールドに入力します。
  - b. 有効な SAP RFC 関数名を「Function」フィールドに入力します。
  - c. 関数のパラメータにデータを入力します。
  - d. 「Execute」(時計) ボタンをクリックします。

データが BPEL チャンネルに到達します。

### A.3.2.2 SAP IDoc イベント

IDoc を SAP から受信できるようにするには、アプリケーション・エクスペローラを使用して、Oracle Application Adapter for SAP R/3 のチャンネルを構成する必要があります。また、SAP アプリケーション・サーバー上での構成も必要です。

1. RFC 宛先を作成するか (SM59 を使用)、前の項 (「SAP RFC アウトバウンドの概要」) で定義したものと同一宛先を使用します。
  - a. 宛先名を入力します。
  - b. 登録されているプログラム ID を入力します。
  - c. 「MDMP & Unicode」タブで、「Unicode」オプションが選択されていることを確認します。
  - d. Oracle BPEL またはメディエータを起動します。
  - e. BPEL またはメディエータのチャンネルの手順 b から、同じ登録済プログラム ID 値を入力します。
  - f. チャンネルを起動します。
  - g. SAP で「Test」ボタンを使用して接続をテストします。
2. A-6 ページの「論理システムの構成」の説明に従って、論理システムを作成または使用します。
3. A-7 ページの「論理システム用の配信モデルの構成」の説明に従って、配信モデルを定義または使用します。
4. トランザクション WE21 を使用して、SAP トランザクション RFC ポートを作成します。
5. トランザクション WE20 を使用して、SAP アウトバウンド・パートナ・プロファイルを作成します。



### A.3.3 SAP GUI での RFC 宛先とプログラム ID の登録

ここでは、次の項目について説明します。

- A.3.3.1 項「プログラム ID の登録」
- A.3.3.2 項「構成のテスト」
- A.3.3.3 項「同期 RFC コール」

SAP R/3 システムが SAP R/3 イベント・アダプタに次の呼出またはインタフェースを発行できるようにするには、RFC 宛先にプログラム ID を登録する必要があります。

- リモート関数呼出し (RFC)
- Business Application Programming Interfaces (BAPI) - BAPI に外部イベントはなく、RFC フォームを使用します。
- Intermediate Documents (IDocs)

RFC 宛先は、プログラム ID をマスキングして、イベントをターゲット・システムに送るために使用されるシンボリック名 (ORACLETDS など) です。プログラム ID は、SAP GUI および イベント・アダプタで構成されます。

#### A.3.3.1 プログラム ID の登録

プログラム ID を登録するには、次のようにします。

1. SAP GUI を起動し、SAP R/3 システムにログインします。
2. 「Tools」 → 「Administration」 → 「Network」 → 「RFC destination」を選択します。
3. SM59 トランザクションを実行します。  
「Configuration of RFC Connections」ウィンドウが表示されます。
4. 「TCP/IP connections」を選択し、「Create」(白紙のアイコン)をクリックします。  
「RFC Destination」ウィンドウが表示されます。
5. 次の情報を指定します。
  - a. 「RFC destination」フィールドに、名前 (ORACLETDS など) を入力します。  
このフィールドの値は大文字と小文字が区別されます。送信プログラムでの大文字小文字と一致させます。
  - b. 「Connection type」フィールドに、宛先タイプ TCP/IP を表す T を入力します。
  - c. 「Description」フィールドに、簡単な説明を入力します。
6. ツール・バーの「Save」アイコン (ディスク) をクリックするか、上部のメニュー・バーの「Connection」 → 「Save」を選択します。  
ウィンドウがリフレッシュされ、2 番目のパネルで「Technical Settings」タブが強調表示されます。
7. 「Registered Server Program」ボタンをただちにクリックします。  
画面が再度リフレッシュされます。
8. 大文字と小文字を区別したプログラム ID を入力し、これはアダプタ・チャンネルでの同じ名前と一致している必要があります。
9. ツール・バーの「Save」アイコン (ディスク) をクリックするか、上部のメニュー・バーの「Connection」 → 「Save」を選択します。
10. 「MDMP & Unicode」タブを選択し、「Unicode」ボタンをクリックします。
11. ツール・バーの「Save」アイコン (ディスク) をクリックするか、上部のメニュー・バーの「Connection」 → 「Save」を選択します。

これで、RFC 宛先とプログラム ID が定義されました。関数の呼出し API の宛先パラメータを使用して、リモート関数呼出し (RFC) をサーバーに送信できます。これは逆呼出しであるため、関数のデータはマーシャルしてアダプタに送信する必要があります。

### A.3.3.2 構成のテスト

構成をテストするには、次のようにします。

1. Oracle Application Adapter for SAP R/3 の「BPEL process / channel」にプログラム ID を入力し、チャンネルを起動します。
2. SAP GUI を開始して SE37 にナビゲートします。  
ABAP ワークベンチ認可が必要です。
3. 有効な関数名を入力し、「Test Run」(ツール) アイコンをクリックします。  
「Test Function Module: Initial Screen」ウィンドウが表示されます。
4. RFC 宛先の名前および任意のデータを入力し、「Execute」(時計) アイコンをクリックします。

テストが成功すると、実行時間が表示されます。アダプタは最初に関数のメタデータを取得してメタデータをキャッシュする必要があるため、最初の実行は常に最も時間がかかり、その後の実行は速くなります。

### A.3.3.3 同期 RFC コール

同期 RFC コールは、SAP 側では非同期と同じように定義されます。アダプタ側では、コールのモードとして「request\_response」が選択されていることを確認します。SAP が提供するデータを取得し、それを処理し、関数書式で SAP に返すプロセスを編成します。書式エラーがあるとリクエストは即時に終了します。例 (疑似コード) を示します。

関数: MY\_COMPANY\_GETLIST DESTINATION 'ORACLETDS'

```
tables mycompanies
    field1: name(20)
    field2: company(20)
```

同期 Get List 呼出しによって空の表 *mycompanies* が BPEL サーバーに渡され、SAP は待機します。BPEL 編成には、次の SQL 文を持つオブジェクトが含まれています。

```
select name(20), company(20) into mycompanies where country eq us
```

SQL 文は Oracle データベースで実行され、結果セットが返されて、編成内の代入文によって *mycompanies* 表に渡されます。プロセス全体を通じて SAP アプリケーションが待機しており、SAP サーバーには SAP によってプロセスが自動的に終了される最大プロセス・オンライン時間がある場合があることに注意してください。

関数は同期呼出しを介して SAP に戻り、表はデータとともに呼出し側プロセスに返されます。

実行時間が接続プール実行時間よりも長い場合にタイムアウトが発生する可能性があります。型、長さまたは小数位に関して書式例外が発生する場合があります。

### A.3.4 イベント・アダプタ用の Application Link Embedding 構成

SAP R/3 イベント・アダプタは、SAP R/3 から IDoc (Intermediate Documents) を受信します。IDoc を SAP R/3 イベント・アダプタに送信するように SAP R/3 システムを構成するには、ALE (Application Link Embedding) 構成を使用して次のようにします。

1. SAP GUI でプログラム ID を登録します。
2. ポートを定義します。
3. 論理システムを作成します。
4. パートナ・プロファイルを作成します。
5. パートナおよびメッセージ・タイプの配信モデルを作成します。
6. SAP R/3 イベント・アダプタをテストします。

### A.3.5 ポートの定義

ポートによってメッセージの送信先が識別されます。Oracle Application Adapter for SAP R/3 で使用されるポート・タイプは、トランザクション RFC ポートです。ALE ポートを定義して RFC 宛先にリンクすることによって、メッセージ実行プロセスの経路が作成されます。

ポートを定義するには、次のようにします。

1. ALE 構成で、「Tools」 → 「Business Communications」 → 「IDocs Basis」 → 「IDoc」 → 「Port Definition」を選択します。  
WE21 トランザクションを実行することもできます。  
「Ports in IDoc processing」ウィンドウが表示されます。
2. 左ペインの「Ports」の下で、「Transactional RFC」を選択し、「Create」（白紙のアイコン）をクリックします。
3. 「Generate port name」をクリックします。  
ポート名が生成されます。
4. このポートで送信する IDoc のバージョンを入力します（通常は 4）。
5. 作成した RFC 宛先の名前を入力または選択するか（ORACLETDS など）、最初の数文字とアスタリスクを入力して選択リストを表示します。
6. セッションを保存し、生成された RFC ポートを書き留めます。

### A.3.6 論理システムの構成

---

**注意：**以前に定義された論理システムを再利用する場合は、この項をスキップできます。

---

論理システムは、メッセージの送信者または受信者のパーティに関する構成情報のためのコンテナ・オブジェクトです。これが論理システムである理由は、接続情報が格納されず、エンド・ポイントで処理可能なメッセージおよびメッセージの処理に使用されるメソッドに関する情報のみが格納されているためです。

ここでは、次の項目について説明します。

- [A.3.6.1 項「SAP トランザクションのショートカット」](#)
- [A.3.6.2 項「論理システムの定義」](#)

### A.3.6.1 SAP トランザクションのショートカット

- /n - 現在のトランザクションを終了し、新しいトランザクションをただちに開きます。
- /o - 現在のトランザクションのコンテキストを維持し、新しいウィンドウで新しいトランザクションを開きます。

### A.3.6.2 論理システムの定義

論理システムの定義を作成するには、SAP Implementation Guide for Customizing (IMG) でエントリを作成します。

ALE は、SAP ERP Application Link Enablement システムであるため、/nSALE トランザクションを使用して ALE 用の IMG にナビゲートします。

1. SALE トランザクションにナビゲートします。  
「Display IMG」ウィンドウに ALE の構成ツリーが表示されます。
2. 次のようにします。
  - a. 「Basic Settings」ノードを展開します。
  - b. 「Logical Systems」を展開します。
  - c. 「Define Logical System」をクリックします。
  - d. SAP R/3 ユーザーの場合、画面が若干異なる可能性があり、特別なノードである「Sending and Receiving Systems」を展開します。  
「Caution: The table is cross-client」というメッセージが表示されます。  
このメッセージは、論理システム内のカスタマイズ・データへの変更が SAP ERP サーバーのすべてのユーザーに適用されることを通知するために表示されます。これは、明確にログオン・クライアントにバインドされるほとんどのアプリケーション・データとは異なります。論理システムのエントリは一切変更しないでください。新しい情報の追加のみを行います。これを行うには、正しい権限を持つ SAP ログオン ID を使用します。
3. 緑色のチェック・マークをクリックして続行します。  
「Change View "Logical Systems"」ウィンドウが強調表示なしで表示されます。
4. 「New Entries」ボタンをクリックします。  
「New Entries: Overview of Added Entries」ウィンドウが表示されます。
5. 「Log.System」列に論理システム (ORACLETD5 など) を入力し、「Name」列に説明を指定します。

---

**注意:** これらのフィールドは、必須フィールドであることがわかるよう黄色で強調表示されています。

---

6. 上部のメニュー・バーの「Save」アイコン (フロッピー・ディスクのアイコン) をクリックします。  
「Prompt for Workbench request」ダイアログが表示されます。

---

**注意:** システムの使用状況によっては、最終ワークベンチのリクエスト番号がダイアログに表示されます。これを使用せず、次の手順に進みます。

---

7. 「New Request」アイコンをクリックして、新しいリクエストを作成します。

8. リクエスト・システムで、エントリの説明を入力します。残りのフィールドは自動的に入力され、変更する必要はありません。  
「Prompt for Workbench Request」ダイアログが再び表示され、新しいリクエスト番号にはワークベンチで作成された最終リクエスト番号が入力されていますが、この番号は使用しません。
9. チェック・マークをクリックして、新しいリクエスト番号を作成します。  
新しい論理システムが青色で強調表示され、この手順が正常に完了したことが示されます。

### A.3.7 論理システム用の配信モデルの構成

配信モデルには、送信者論理システム、受信者論理システムおよび送受信されるメッセージのタイプが含まれます。指定された送信者または受信者に有効な配信モデルがない場合、メッセージは処理されず、エラー位置にルーティングされます。SAP BAPI オブジェクトやフィルタリングを含め、配信モデルの詳細なトピックについては、SAP のドキュメントを参照してください。

この項では、次の項目について説明します。

#### ■ A.3.7.1 項「配信モデルの定義」

##### A.3.7.1 配信モデルの定義

配信モデルを定義するには、次のようにします。

1. 画面の左上にあるコマンド・バーに **/nbd64** と入力し、**bd64** トランザクションを実行します。  
「Display Distribution Model」ウィンドウがデフォルトの表示モードで表示され、現在のモデル・ビューのエントリが示されます。
2. ツリーを展開して、各モデル・ビューのエントリを表示します。  
アイテムを追加するには、トランザクションを変更モードに切り替える必要があります。
3. 上部のメニューから、「**Distribution model**」をクリックし、リストから「**Switch processing mode**」を選択します。  
画面がリフレッシュされ、「Change Distribution Model」メイン・ウィンドウに戻ります。
4. 使用可能なメニュー・ボタンから、「**Create Model View**」を選択します。  
「Create Model View」ダイアログが表示されます。
5. 「Short text」フィールドにモデル・ビュー名を入力し、「Technical name」フィールドに説明としての役割も果たす名前を入力します。  
文字 **Z** で始まる名前を使用することをお勧めしますが、これは、SAP がシステム更新時にこれらの名前を上書きしないためです。これは、モデル・ビューの名前についてのみです。
6. 緑色のチェック・マーク・アイコンをクリックして、情報を入力します。  
画面がリフレッシュされ、「Change Distribution Model」メイン・ウィンドウに戻ります。構成された配信モデルがリストに追加されました。  
モデル・ビューは、「Save」アイコンをクリックするまでトランザクションの一時記憶域に置かれます(エントリは明るい色で表示されます)。システム接続が切断またはタイムアウトすると、エントリは消失します。ベスト・プラクティスとして、配信モデルの構成プロセス中は、変更内容が失われないように各ダイアログの後に変更を保存します。
7. 以前に定義したエントリにカーソルを置きます。  
エントリが強調表示されます。
8. 中央のメニュー・バーで、「**Add Message Type**」ボタンをクリックします。  
「Add Message Type」ダイアログが表示されます。

9. ALE セントラル・インスタンスの送信者を入力します。  
この情報については、SAP 管理者に問い合せてください。
10. 「Receiver」フィールドに受信者 (ORACLETDS など) を入力します。
11. 有効なメッセージ・タイプを入力します。
12. このパートナ・シナリオのアウトバウンド・メッセージ・タイプごとに、この手順を繰り返します。

### A.3.8 パートナ・プロファイルの定義

---

**注意：** 以前に定義されたパートナ・プロファイルを再利用する場合は、この項をスキップできます。

---

パートナ・プロファイルは、論理システムごとに、処理するメッセージとメッセージの処理方法を定義します。これにより、パートナの処理の詳細を柔軟に構成することが可能となります。パートナ・プロファイルには、メッセージ・タイプごとにエントリを1つのみ含めることができます。パートナ・プロファイルを定義するには、次のようにします。

1. 画面の左上にあるコマンド・バーに **/nwe20** と入力し、**we20** トランザクションを実行します。  
「Partner profiles」ウィンドウが表示されます。
2. 「**Partner Type LS**」を選択して、中央のメニュー・バーの白色のアイコンをクリックします。上部のメニューの「**Create**」を選択することもできます。  
画面の右側の表示が変わり、パートナ・プロファイルの未記入フォームが表示されます。
3. パートナ番号フィールドに論理システムの名前 (ORACLETDS など) を入力し、「**Save**」アイコンをクリックします。  
画面が新しいプロファイルによって変更されます。
4. 「**Message Type**」フィールドにメッセージ・タイプを入力します。
5. 「**Receiver Port**」フィールドにポート番号を入力します。
6. 「**Transfer IDoc immed.**」ボタンを選択するか「**Collect IDoc**」を選択し、バッチ・サイズを選択して、**bd87** を使用して処理のために IDoc をリリースします。
7. 基本タイプを入力します (MATMAS の場合の MATMAS05 など)。
8. 「**Save**」アイコンをクリックします。

## A.3.9 SAP R/3 ALE 構成のテスト

SAP サーバーでは、BD12 トランザクションによって、任意の論理システム ( イベント・アダプタなど ) へのマスター・データ IDoc の送信が可能になります。

ここでは、次の項目について説明します。

- [A.3.9.1 項「SAP R/3 ALE 構成のテスト」](#)
- [A.3.9.2 項「SAP IDoc イベント」](#)
- [A.3.9.3 項「IDoc のバージョンの操作」](#)

### A.3.9.1 SAP R/3 ALE 構成のテスト

SAP R/3 Application Link Embedding (ALE) 構成をテストするには、次のようにします。

1. 「Send Customers」ウィンドウで、「Output type」フィールドに IDoc メッセージ・タイプ (DEBMAS など) を入力します。
2. 「Logical system」フィールドに、論理システム (ORACLETDS など) を入力します。
3. 「Run」をクリックします。

SAP R/3 イベント・アダプタが、XML 形式の IDoc を受信します。イベント・アダプタからのレスポンスは想定されていません。

確認メッセージが表示されます。

同じ数の通信 IDoc も生成される必要があります。通信 IDoc の数とマスター IDoc の数が異なる場合は、一部の IDoc が構文チェックに失敗した可能性があります。IDoc アプリケーション開発者に確認してください。通常、IDoc はほぼ即時にアダプタ・サーバーに表示されます。IDoc が表示されない場合は、次のことを確認してください。

1. パートナ・プロファイルで「Collect IDocs」オプションが選択されています (bd87 に移動して、それらを処理のためにリリースします)。
2. 他のサーバーが同じプログラム ID を公開していません。

### A.3.9.2 SAP IDoc イベント

IDoc を SAP から受信できるようにするには、アプリケーション・エクスペローラを使用して、Oracle Application Adapter for SAP R/3 のチャンネルを構成する必要があります。また、SAP アプリケーション・サーバー上での構成も必要です。

1. RFC 宛先を作成するか (SM59 を使用)、前の項 (「SAP RFC アウトバウンドの概要」) で定義したものと同一宛先を使用します。
  - a. 宛先名を入力します。
  - b. 登録されているプログラム ID を入力します。
  - c. 「MDMP & Unicode」タブで、「Unicode」オプションが選択されていることを確認します。
  - d. Oracle BPEL またはメディエータを起動します。
  - e. BPEL またはメディエータのチャンネルの手順 b から、同じ登録済プログラム ID 値を入力します。
  - f. チャンネルを起動します。
  - g. SAP で「Test」ボタンを使用して接続をテストします。
2. A-6 ページの「[論理システムの構成](#)」の説明に従って、論理システムを作成または使用します。
3. A-7 ページの「[論理システム用の配信モデルの構成](#)」の説明に従って、配信モデルを定義または使用します。
4. トランザクション WE21 を使用して、SAP トランザクション RFC ポートを作成します。

5. トランザクション WE20 を使用して、SAP アウトバウンド・パートナ・プロファイルを作成します。

### A.3.9.3 IDoc のバージョンの操作

SAP アウトバウンド IDoc 内の IDoc アウトバウンド・セグメントは、リリースに適したメタデータと一致している必要があります。

アダプタには、IDoc リリース・バージョン用の次の 3 つの設定があります。

- SAP ログオン・リリース - 現在のバージョン。
- IDoc リリース - 送信アプリケーションは、IDoc 制御レコードを変更して正しいバージョンを出力する必要があり、変更しない場合は、デフォルトで SAP ログオン・リリースになります。
- ユーザー入力

ユーザー入力フィールドは、2 つのタイプの入力を受け入れます。入力が SAP リリース (700 など) である場合、そのリリースのメタデータが使用されます。

このフィールドが空白値の場合は、リリースされているかどうかにかかわらず、SAP はその最上位バージョンのメタデータを返します。不明または最近の IDoc を操作する場合、これが最適のオプションであることが多くあります。リリース済セグメント・リストでセグメント名が見つからない場合、SAP は (アダプタを介して) メッセージ「Segment unknown」を返します。この場合に IDoc を処理する唯一のオプションは、セグメントをリリースするか、またはアダプタ設定を空白値のユーザー入力に変更することです。



---

---

# 用語集

## アダプタ

1つの電子のインタフェースを別の電子のインタフェースに適応させることにより、(機能を損なうことなく)汎用的な接続を提供します。

## エージェント

リスナーおよびドキュメントにおけるサービス・プロトコルをサポートします。

## チャンネル

バックエンド・システムの特特定インスタンスへの構成された接続を表します。チャンネルにより、アダプタで管理される特定のリスナーへの1つ以上のイベント・ポートがバインドされます。

## ビジネス・サービス

Web サービスとも呼ばれます。Web サービスは自己完結型のモジュール化された機能であり、オープン・スタンダードを使用してネットワーク全体でパブリッシュしたりアクセスできます。コンポーネント別のインタフェースの実装であり、実行可能なエンティティです。

## ポート

アダプタによって公開されている特定のビジネス・オブジェクトと特定の配置とを関連付けます。配置とは、イベント・データのプロトコルおよび場所を定義する URL です。ポートにより、イベント消費のエンド・ポイントが定義されます。

## リスナー

クライアント・アプリケーションからのリクエストを受け入れるコンポーネント。

リスナー

# 索引

## B

BAPI (Business Application Programming Interfaces), 12-4  
BAPI リクエスト, 1-4  
BAPI 例外パラメータ, 4-9, 4-22, 6-45  
BPEL Designer, 6-1, 8-1  
BPEL Process Manager  
    OracleAS Adapter for MySAP ERP, 6-1, 8-1  
    アダプタとの統合, 6-1, 8-1  
BSE (OracleAS Adapter Business Services Engine), 1-2, 1-5  
    トラブルシューティング, 12-5  
Business Application Programming Interfaces (BAPI), 12-4

## C

CCI コール, 5-1

## D

DEBMAS05  
    検索, 6-46  
DEBMAS05 の検索, 6-46

## E

Eclipse。「JDeveloper」を参照  
「Edi バージョン」パラメータ, 4-8  
EJB (Enterprise Java Beans), 5-1  
Enterprise Java Beans (EJB), 5-1

## I

「iBSE URL」フィールド, 4-3  
IDoc (Intermediate Documents), 12-4  
IDoc リクエスト, 1-4  
Intermediate Documents (IDoc), 12-4  
IWayConfig パラメータ, 5-4  
IWayHome パラメータ, 5-4  
IWayRepoPassword パラメータ, 5-4  
IWayRepoURL パラメータ, 5-4  
IWayRepoUser パラメータ, 5-4

## J

J2CA (J2EE Connector Architecture 用エンタープライズ・コネクタ), 1-5  
    OracleAS Adapter, 5-1

## J2CA 構成

    アプリケーション・エクスプローラ, 5-3  
J2CA サービス, 6-42  
J2CA リソース・アダプタ, 1-2, 5-1  
J2EE Connector Architecture (J2CA) 用エンタープライズ・コネクタ, 1-5  
Java プログラム・クライアント, 5-1  
JDeveloper, 6-1, 8-1

## L

Loglevel パラメータ, 5-4

## M

ManagedConnectionFactory パラメータ, 5-4  
「MySAP Easy Access」ウィンドウ, 6-60  
MySAP ERP  
    接続先, 1-4, 4-5 ~ 4-12, 6-9, 6-46 ~ 6-47, 12-2, 12-6  
    切断, 4-12  
    統合, 1-4  
MySAP ERP から切断, 4-12  
MySAP ERP でのイベントのトリガー, 6-60  
MySAP ERP のトレース・パラメータ, 4-7, 4-21, 6-45  
MySAP ERP ビジネス関数, 4-5, 4-14 ~ 4-17  
    スキーマの作成, 4-14  
MySAP ERP ビジネス・プロセス, 1-4  
MySAP ERP プラットフォーム, 1-4  
MySAP ERP への接続, 4-5, 4-12, 6-9, 6-46, 6-47, 12-2, 12-6  
MySAP Workbench, 6-60  
「MySAP」ノード, 6-46 ~ 6-47  
MySAP プログラム ID (識別子), 6-61

## O

OC4J-ra.xml ファイル, 5-3 ~ 5-5  
Oracle Application Server  
    コネクタのデプロイメント, 5-1  
Oracle JDeveloper, 6-1, 8-1  
Oracle JDeveloper BPEL Designer。「BPEL Designer」、  
    「JDeveloper」または「Oracle JDeveloper」を参照  
Oracle Unified Method (OUM), ix  
OracleAS Adapter  
    J2CA, 5-1  
    インストール・ディレクトリ, 5-4

OracleAS Adapter Application Explorer。「アプリケーション・エクスプローラ」を参照  
OracleAS Adapter Business Services Engine (BSE), 1-2, 1-5  
トラブルシューティング, 12-5  
OracleAS Adapter for MySAP ERP  
BPEL Process Manager, 6-1, 8-1  
構成, 4-1, 4-25  
デプロイ, 1-2, 6-2, 8-2  
トラブルシューティング, 12-1, 12-7  
OracleAS Adapter J2CA  
アプリケーション・エクスプローラ, 12-4  
接続先, 12-4  
OracleAS Adapter J2CA への接続, 12-4  
Oracle データベース, 5-4

## R

RFC リクエスト, 1-4

## S

sapjco3.dll ファイル, 12-2  
sapjco3.jar ファイル, 12-2 ~ 12-3  
「Send Customers」ウィンドウ, 6-61  
「SNC パートナ」パラメータ, 4-11  
「SNC 名」パラメータ, 4-11  
「SNC モード」パラメータ, 4-11  
「SNC ライブラリ・パス」パラメータ, 4-11  
「SNC レベル」パラメータ, 4-11  
SOAP エージェント, 12-5  
SOAP フォルト, 12-5  
SOAP リクエスト, 12-5 ~ 12-7  
エラー, 12-6 ~ 12-7  
SOA (サービス指向アーキテクチャ), 6-1, 8-1

## W

Web Service Definition Language (WSDL), 4-15 ~ 4-16  
Web サービス, 1-2  
作成, 4-16 ~ 4-17  
テスト, 4-18  
デプロイ, 12-5 ~ 12-7  
統合, 6-1, 8-1  
トラブルシューティング, 12-5 ~ 12-7  
リポジトリ・プロジェクト, 4-2  
「Web サービスの作成」ダイアログ・ボックス, 4-17  
Web サービスのテスト, 4-18  
Web サービス・プロジェクト  
作成, 4-2  
Web サービス名, 4-17  
WSDL (Web Service Definition Language), 4-15, 4-16  
WSDL ドキュメント, 6-1, 8-1  
「WSDL のエクスポート」ダイアログ・ボックス, 6-48  
WSDL ファイル, 6-1, 8-1  
アプリケーション・エクスプローラ, 6-42

## X

XML スキーマ  
作成, 4-14  
XML ドキュメント  
入力, 12-4  
XML メッセージ, 1-2

## あ

アウトバウンド JCA サービス (リクエスト/レスポンス) の作成, 6-10  
アウトバウンド MySAP ERP 処理, A-1 ~ ??  
アダプタ  
BPEL Process Manager との統合, 6-1, 8-1  
構成, 4-1 ~ 4-25, 5-3  
デプロイ, 1-5, 5-1, 5-3, 6-2, 8-2  
トラブルシューティング, 12-1 ~ 12-7  
アダプタ構成  
更新, 5-3  
アダプタ構成の更新, 5-3  
アダプタ・タイプ  
リソース, 1-2, 5-1 ~ 5-3  
「アダプタ」ノード, 6-46  
アダプタの構成, 4-1, 4-25, 5-3  
アダプタのデプロイ, 5-1 ~ 5-3, 6-2, 8-2  
アダプタ例外, 12-4 ~ 12-5  
アプリケーション・アダプタ  
BPEL Process Manager との統合, 6-1, 8-1  
構成, 4-1, 4-25, 5-3  
デプロイ, 5-1, 5-3, 6-2, 8-2  
トラブルシューティング, 12-1, 12-7  
アプリケーション・エクスプローラ, 1-2, 6-8, 6-42  
J2CA 構成, 5-3  
OracleAS Adapter J2CA, 12-4  
WSDL ファイル, 6-42  
アプリケーション・システム, 4-5  
スキーマ, 4-14  
チャンネル, 4-19  
テスト, 12-3  
デバッグ, 12-3  
トラブルシューティング, 12-2, 12-3  
「アプリケーション・サーバー」ダイアログ・ボックス, 4-6  
「アプリケーション・サーバー」パラメータ, 4-8  
アプリケーション・システム  
アプリケーション・エクスプローラ, 4-5  
サポートされている, 4-5

## い

イベント, 1-2, 4-19, A-1  
構成, 4-19  
作成, 6-46, 6-49  
トリガー, 6-60  
イベント・アダプタ  
構成, 4-19  
イベント処理, A-1  
イベント統合  
検証, 6-60  
イベントの作成, 6-46, 6-49  
イベント・ポート  
作成, 6-46 ~ 6-49  
イベント・メッセージ, 6-42  
インストール・ディレクトリ, 5-4  
インバウンド J2CA サービス, 6-42

## え

---

エラー・メッセージ, 12-2 ~ 12-7  
ターゲット・システム, 12-5

## お

---

オブジェクト・タイプ, 12-4

## か

---

管理対象コネクタ・ファクトリ, 5-3  
管理対象コネクタ・ファクトリ・オブジェクト, 5-3  
複数, 5-5

## き

---

「既存のサービス名」リスト, 4-17

## く

---

「クライアント」パラメータ, 4-7, 4-21, 6-45

## け

---

権限  
構成, 12-4  
「言語」パラメータ, 4-7, 4-21, 6-45

## こ

---

構成  
上書き, 5-4  
接続先, 4-4  
定義, 4-2 ~ 4-4  
「構成」ノード, 4-2 ~ 4-3  
構成パラメータ, 5-4  
IWayConfig, 5-4  
IWayHome, 5-4  
IWayRepoPassword, 5-4  
IWayRepoURL, 5-4  
IWayRepoUser, 5-4  
Loglevel, 5-4  
「コードページ」パラメータ, 4-7, 4-21, 6-45  
顧客マスター・データ, 6-61  
コネクタ  
Oracle Application Server へのデプロイ, 5-1  
コネクタ・ファクトリ, 5-3  
コネクタ・ファクトリ・オブジェクト, 5-3  
複数, 5-5

## さ

---

サービス, 1-2  
作成, 4-16, 4-17  
テスト, 4-18  
サービス指向アーキテクチャ (SOA), 6-1, 8-1  
「サービス・プロバイダ」リスト, 4-3 ~ 4-4  
サービス名, 4-17  
サブレット, 5-1

## し

---

「システム」タブ, 4-7, 4-21, 6-44  
「システム番号」パラメータ, 4-8  
「詳細」タブ, 4-9, 4-22, 6-45  
「新規構成」ダイアログ・ボックス, 4-2 ~ 4-4

## す

---

スキーマ  
アプリケーション・エクスプローラ, 4-14  
作成, 4-14

## せ

---

「制限」ページ, 6-61  
「セキュリティ」タブ, 4-11  
接続  
確立, 4-5 ~ 4-12, 6-47  
削除, 4-12  
閉じる, 4-12  
「接続」ダイアログ・ボックス, 4-12, 6-47  
接続パラメータ, 4-6 ~ 4-12, 4-21, 4-22, 12-2  
BAPI 例外処理, 4-9  
Edi バージョン, 4-8  
MySAP ERP のトレース, 4-7  
SNC パートナ, 4-11  
SNC 名, 4-11  
SNC モード, 4-11  
SNC ライブラリ・パス, 4-11, 4-22, 6-45  
SNC レベル, 4-11  
アプリケーション・サーバー, 4-8  
クライアント, 4-7, 4-21, 6-45  
言語, 4-7  
コードページ, 4-7  
システム, 4-7, 4-21, 6-44  
システム番号, 4-8  
詳細, 4-9, 4-22, 6-45  
セキュリティ, 4-11  
接続プール・サイズ, 4-9  
接続プール名, 4-9  
待機中コミット, 4-9  
パスワード, 4-7, 4-21, 6-45  
ユーザー, 4-7, 4-21, 6-45  
ログオン・チケット (SSO2), 4-11  
ログオン・チケット (X509), 4-11  
「接続プール・サイズ」パラメータ, 4-9, 4-22, 6-45  
「接続プール名」パラメータ, 4-9, 4-22, 6-45  
「説明」フィールド, 4-6, 4-17

## た

---

ターゲット  
削除, 4-12  
接続先, 4-12, 6-47, 12-6  
切断, 4-12  
定義, 4-5  
編集, 4-13  
ターゲット・システム  
エラー, 12-5  
ターゲット・タイプ・リスト, 4-6  
「ターゲットの追加」ダイアログ・ボックス, 4-6  
ターゲットの編集, 4-13

「待機中コミット」パラメータ, 4-9, 4-22, 6-45

## ち

---

チャンネル, 4-19  
アプリケーション・エクスプローラ, 4-19  
開始, 4-24, 6-46  
削除, 4-25  
作成, 4-19, 6-42 ~ 6-46  
停止, 4-24, 6-46  
テストとデバッグ, 6-46  
編集, 4-25  
「チャンネル」ノード, 4-20, 6-43  
チャンネルの作成, 6-42, 6-46  
「チャンネルの追加」ダイアログ・ボックス, 4-19  
チャンネルの編集, 4-25  
チャンネル。「リスナー」も参照

## て

---

データベース  
Oracle, 5-4  
接続先, 5-4  
データベース接続  
開く, 5-4  
デザインタイム, 6-42  
構成, 6-8, 6-42

## と

---

トラブルシューティング, 12-1 ~ 12-7  
BSE, 12-5  
Web サービス, 12-5, 12-7  
アプリケーション・エクスプローラ, 12-2 ~ 12-3  
トランザクション処理, 1-2  
トレース情報, 12-1

## な

---

「名前」フィールド, 4-6

## に

---

入力XMLドキュメント, 12-4

## の

---

ノード  
MySAP, 6-46, 6-47  
アダプタ, 6-46  
構成, 4-2, 4-3  
接続済, 4-12, 6-47  
切断, 4-13  
チャンネル, 4-20, 6-43  
ポート, 6-43  
ノードのリスト, 12-2  
ノード・リスト, 12-2

## は

---

パスワード, 4-7, 4-21, 5-4, 6-45  
「パスワード」パラメータ, 4-7, 4-21, 6-45

パラメータ・タイプ  
構成, 5-4  
接続, 4-21 ~ 4-22, 12-2  
チャンネル構成, 4-25

## ひ

---

ビジネス・イベント, 1-2  
ビジネス関数, 4-5, 4-14, 4-17  
スキーマの作成, 4-14  
ビジネス・サービス  
作成, 4-16, 4-17  
テスト, 4-18  
ビジネス・プロセス, 1-4

## ふ

---

フォルト・コード要素, 12-5  
フォルト文字列要素, 12-5  
プログラムID (識別子), 6-61  
プロセス・マネージャ。「BPEL Process Manager」を参照  
プロパティ, 5-4

## ほ

---

ポート  
作成, 6-46, 6-49  
「ポート」ノード, 6-43  
ポート番号パラメータ, 4-3  
ホスト名パラメータ, 4-3

## め

---

「メソッド名」フィールド, 4-17  
メッセージ, 1-2  
「メッセージ・サーバー」ダイアログ・ボックス, 4-19  
メッセージ・タイプ  
イベント, 6-42  
メディアータ・アウトバウンド・プロセス, 7-2  
メディアータ・インバウンド・プロセス, 7-13

## ゆ

---

「ユーザー」タブ, 4-7, 4-21, 6-45  
「ユーザー」パラメータ, 4-7, 4-21, 6-45, 12-4

## ら

---

ライセンス, 4-17  
ライセンスとメソッド・ダイアログ・ボックス, 4-17  
「ライセンス」フィールド, 4-17

## り

---

リクエスト  
実行, 12-4  
リクエスト処理  
BAPI, 1-4  
IDoc, 1-4  
RFC, 1-4  
リクエスト・スキーマ, 4-14  
リスナー, 1-5, 4-4  
リスナー。「チャンネル」も参照

リソース・アダプタ, 5-1 ~ 5-3  
デプロイ, 1-2  
リポジトリ・プロジェクト  
  Web サービス, 4-2  
  作成, 4-2 ~ 4-4  
リポジトリ・プロジェクトの作成, 4-2, 4-4  
リモート関数モジュール, 12-4

## れ

---

レスポンス・スキーマ, 4-14

## ろ

---

「ログオン・チケット (SSO2)」パラメータ, 4-11  
「ログオン・チケット (X509)」パラメータ, 4-11  
ログ・ファイル, 12-1  
ログ・レベル  
  上書き, 5-4  
論理システム, 6-61

