

Oracle® Fusion Middleware Application Adapters

Application Adapter for SAP R/3 ユーザーズ・ガイド for
12c (12.2.1.2.0)

E85898-01

2016 年 12 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 Oracle WebLogic Server 12c (12.2.1.2.0) Application Adapter 12c (12.2.1.2.0) for SAP R/3 ユーザーズ・ガイド

E85898-01

Copyright © 2001, 2016, Oracle and/or its affiliates. All rights reserved.

原著者 : Stefan Kostial

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

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクルまでご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

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, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このソフトウェアまたはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアまたはハードウェアは、危険が伴うアプリケーション (人的傷害を発生させる可能性があるアプリケーションを含む) への用途を目的として開発されていません。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用する場合、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。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、Oracle Corporation およびその関連会社は一切の責任を負いかねます。

Oracle および Java はオラクルおよびその関連会社の登録商標です。その他の社名、商品名等は各社の商標または登録商標である場合があります。

Intel、Intel Xeon は、Intel Corporation の商標または登録商標です。すべての SPARC の商標はライセンスをもとに使用し、SPARC International, Inc. の商標または登録商標です。AMD、Opteron、AMD ロゴ、AMD Opteron ロゴは、Advanced Micro Devices, Inc. の商標または登録商標です。UNIX は、The Open Group の登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。Oracle Corporation およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。Oracle Corporation およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

はじめに	xi
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
1 概要	
アダプタの機能	1-1
サポートされるバージョンおよびプラットフォーム	1-3
ABAP 用の古典的な SAP テクノロジ	1-3
SAP R/3 との統合	1-4
アダプタのアーキテクチャ	1-5
Oracle Adapter Business Services Engine (BSE) のアーキテクチャ	1-5
Oracle Adapter J2CA のアーキテクチャ	1-6
BSE と Oracle Adapter J2CA デプロイメントの差異	1-7
サンプル・プロジェクト	1-8
クイック・スタート・ガイド	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・エクスプローラでの構成、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
2 スタート・ガイド	
SAP Java Connector (SAP JCo) の検証	2-1
Windows プラットフォーム上での SAP JCo の検証	2-1
UNIX プラットフォーム上での SAP JCo の検証	2-2

SAP R/3 ログオン・パラメータの特定	2-3
ユーザー・パラメータ.....	2-4
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.....	2-5
接続プール・パラメータ.....	2-5
SAP ゲートウェイ・パラメータ.....	2-6
ALE パラメータ.....	2-6
グローバル処理パラメータ.....	2-7
SNC パラメータ.....	2-7

3 SAP Java Connector 3.0xx について

サポート対象プラットフォーム.....	3-1
接続管理.....	3-2
接続プール - イベント.....	3-2
接続プール - サービス.....	3-3
単一サーバーのサポート.....	3-6
サーバー・スレッド.....	3-6
一意性.....	3-6
ロード・バランシングのためのアダプタ構成.....	3-7
ロード・バランシングのための SAP 構成.....	3-7
変更パラメータとテスト・パラメータ.....	3-7
トレース・レベル・パラメータ.....	3-8

4 Oracle Application Adapter for SAP R/3 の構成

SAP R/3 のネスト構造およびネストされた表について.....	4-1
アプリケーション・エクスプローラの起動.....	4-4
リポジトリ設定の構成.....	4-5
リポジトリ構成の作成.....	4-5
BSE 用の構成の作成.....	4-5
J2CA 用の構成の作成.....	4-6
BSE または J2CA 構成への接続.....	4-7
SAP R/3 用の接続 (ターゲット) の確立.....	4-8
SAP R/3 へのターゲットの定義.....	4-8
定義済みの SAP R/3 ターゲットへの接続.....	4-15
SAP R/3 への接続の管理.....	4-15
アプリケーション・システム・オブジェクトの表示.....	4-17
XML スキーマの作成.....	4-18
WSDL の生成 (J2CA 構成のみ).....	4-18
Web サービスの作成およびテスト (BSE 構成のみ).....	4-20
Web サービスの作成.....	4-20
Web サービスのテスト.....	4-21
伝播の識別.....	4-22

イベント・アダプタの構成.....	4-22
チャンネルの作成および編集.....	4-23
スキーマの検証.....	4-29
SAProuter サービスの構成.....	4-31
SAProuter の構成.....	4-31
Oracle コンポーネントを使用した SAProuter のテスト.....	4-32
SAProuter 構成の検証.....	4-34

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

Oracle WebLogic Server とのアダプタの統合.....	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.....	5-2
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.....	5-4
追加の接続ファクトリ値用の WSDL ファイルの変更.....	5-6

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

概要.....	6-1
アダプタのデプロイメント.....	6-2
新規のアプリケーション・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).....	6-7
リクエスト / レスポンス型のサービス用の WSDL の生成.....	6-8
SOA 用の空のコンポジットの作成.....	6-9
BPEL アウトバウンド・プロセスの定義.....	6-11
BPEL アウトバウンド・プロセスのデプロイ.....	6-30
Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動.....	6-33
アウトバウンド BPEL およびアウトバウンド・メディアータ・プロセスのテスト.....	6-36
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).....	6-37
イベント統合のための WSDL の生成.....	6-37
SOA 用の空のコンポジットの作成.....	6-45
BPEL インバウンド・プロセスの定義.....	6-45
BPEL インバウンド・プロセスのデプロイ.....	6-53
SAP R/3 でのイベントのトリガー.....	6-54
BPEL を使用した同期 SAP R/3 イベントの作業 (J2CA 構成).....	6-55
概要.....	6-56
アダプタ・ターゲットの構成.....	6-57
チャンネルの構成.....	6-57
メッセージ用の WSDL の生成.....	6-58
JDeveloper での BPEL プロセスの作成.....	6-58
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).....	6-63
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.....	6-64
SOA 用の空のコンポジットの作成.....	6-65
BPEL アウトバウンド・プロセスの定義.....	6-65

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

新規のアプリケーション・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.....	7-13
Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動.....	7-13
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.....	7-14
メディエータ・インバウンド・プロセスの定義.....	7-14
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.....	7-21
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・コンポーネントの統合

概要.....	8-1
アダプタのデプロイメント.....	8-2
新規のアプリケーション・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 との既知のデプロイメント問題の調整.....	8-25
BPM アウトバウンド・プロセスのデプロイ.....	8-25
Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動.....	8-25
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).....	8-28
BPM 用の空のコンポジットの作成.....	8-28
BPM インバウンド・プロセスの定義.....	8-28
BPM を使用した同期 SAP R/3 イベントの作業 (J2CA 構成).....	8-44
アダプタのターゲットおよびチャネルの構成.....	8-44
メッセージ用の WSDL ファイルの生成.....	8-44
JDeveloper での BPM プロセスの作成.....	8-45
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).....	8-51
BPM 用の空のコンポジットの作成.....	8-51
BPM アウトバウンド・プロセスの定義.....	8-51

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処理の構成

Application Adapter と Oracle Service Bus の統合の概要.....	9-1
sbconsole を使用したアウトバウンド処理の構成 (J2CA 構成).....	9-2
Oracle Service Bus の起動およびプロジェクト・フォルダの作成.....	9-2

Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・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 の公開	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.....	9-10
プロキシ・サービスを使用したパイプラインの構成.....	9-14
sbconsole を使用したインバウンド処理の構成 (J2CA 構成).....	9-22
Oracle Service Bus の起動およびプロジェクト・フォルダの作成.....	9-23
Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定.....	9-23
アプリケーション・エクスプローラから Oracle Service Bus への WSDL の公開	9-23
WSDL ベースのプロキシ・サービスの構成.....	9-25
ファイル・タイプ・ビジネス・サービスの構成.....	9-27
パイプラインの構成.....	9-30
sbconsole を使用したアウトバウンド処理の構成 (BSE 構成).....	9-37
Oracle Service Bus の起動およびプロジェクト・フォルダの作成.....	9-37
Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定.....	9-37
アプリケーション・エクスプローラから Oracle Service Bus への WSDL の公開	9-38
ファイル・タイプ・ビジネス・サービスの構成.....	9-39
WSDL ベースのビジネス・サービスの構成.....	9-39
プロキシ・サービスを使用したパイプラインの構成.....	9-41
Oracle Service Bus を使用した同期 SAP R/3 イベントの作業 (J2CA 構成)	9-50
Oracle Service Bus の起動およびプロジェクト・フォルダの作成.....	9-51
アダプタ・ターゲットおよびチャンネルの構成.....	9-51
メッセージ用の WSDL ファイルの生成.....	9-51
Oracle Service Bus コンソールでのビジネス・サービスの作成.....	9-52
Oracle Service Bus コンソールでのプロキシ・サービスの作成.....	9-53
プロキシ・サービスでのパイプラインの構成.....	9-57
Oracle Service Bus を使用した JMS プロキシ・サービスの構成 (J2CA 構成)	9-58
Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成)	9-76

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

JDeveloper を使用した OSB アウトバウンド・プロセスの構成 (J2CA 構成).....	10-1
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
JDeveloper を使用した OSB インバウンド・プロセスの構成 (J2CA 構成).....	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	10-30
JDeveloper を使用した OSB アウトバウンド・プロセスの構成 (BSE 構成).....	10-30
OSB の Service Bus アプリケーションの作成.....	10-31

OSB アウトバウンド・プロセスの定義.....	10-31
OSB アウトバウンド・プロセスのデプロイ.....	10-44
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)...	10-44
OSB の Service Bus アプリケーションの作成.....	10-45
OSB プロセスの定義.....	10-45
OSB プロセスのデプロイ.....	10-50
JDeveloper を使用した JMS インバウンド・プロセスの構成 (J2CA 構成).....	10-50
JDeveloper を使用した JMS アウトバウンド・プロセスの構成 (J2CA 構成).....	10-57
JDeveloper を使用した HTTP アウトバウンド・プロセスの構成 (J2CA 構成).....	10-66

11 主要な機能

ロギング機能の構成.....	11-1
J2CA コネクタ・アプリケーションのログ・ファイル管理の構成.....	11-1
Business Services Engine (BSE) のログ・ファイル管理の構成.....	11-8
診断機能の構成.....	11-11
サポートされるプロトコル.....	11-12
SOA デバッグ機能の構成.....	11-12
SOA デバッガの使用のガイドライン.....	11-13
前提条件.....	11-13
Oracle JDeveloper での BPEL プロセスのデバッグ.....	11-13
Oracle JDeveloper での OSB プロセスのデバッグ.....	11-26
例外フィルタの構成.....	11-44
例外フィルタの構成.....	11-44
Oracle SOA Suite (BPEL、メディエータまたは BPM) の資格証明マッピング.....	11-57
資格証明マッピングの構成.....	11-58
JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング.....	11-62
資格証明マッピングの構成.....	11-63
テキストからの Unicode 文字のフィルタリング.....	11-70
概要.....	11-71
フィルタの有効化.....	11-71
置換文字の指定.....	11-71
設定ファイルの保存およびサーバーの再起動.....	11-72
フィルタの検証.....	11-72
クイック・スタート・ガイド.....	11-72
既知の制限.....	11-72

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.....	12-5
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
アダプタ特定のエラー処理.....	12-6

A リモート・プロセスにおける 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・メッセージの処理.....	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-9
SAP アウトバウンド・メッセージの処理.....	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-12
SAP GUI で 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 の構成.....	A-15
ポートの定義.....	A-15
論理システムの構成.....	A-16
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 Oracle WebLogic Server Application Adapter for SAP R/3 ユーザーズ・ガイド*へようこそ。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 (12.2.1.2.0) 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 Oracle WebLogic Server Application Adapter アップグレード・ガイド』
- 『Oracle Fusion Middleware Oracle WebLogic Server Application Adapter ベスト・プラクティス・ガイド』
- Oracle's 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

表記規則

このドキュメントでは次の表記規則を使用します。

規則	意味
太字	太字は、操作に関連する Graphical User Interface 要素、または本文中で定義されている用語および用語集に記載されている用語を示します。
イタリック	イタリックは、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。
固定幅フォント	固定幅フォントは、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 を通じて SAP R/3 システムに接続されます。Oracle Application Adapter for SAP R/3 接続を確立し、SAP R/3 システム上で相互作用を実行します。

注意： このドキュメントでは、<ORACLE_HOME> は、12c (12.2.1.0.0) SOA/OSB のインストール先であるホーム・ディレクトリを指します。

<ADAPTER_HOME> は、以下を意味します。

- SOA の場合：
`<ORACLE_HOME>\soa\soa\thirdparty\ApplicationAdapters`
 - OSB の場合：
`<ORACLE_HOME>\osb\3rdparty\ApplicationAdapters`
-

この章の内容は以下のとおりです。

- [1.1 項「アダプタの機能」](#)
- [1.2 項「ABAP 用の古典的な 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 デプロイメントの差異」](#)
- [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

Oracle Application Adapter for SAP R/3 (SAP Java Connector (SAP JCo) 3.0xx を使用) 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、データベースおよび外部ビジネス・パートナーのシステムとの間でリアルタイムのビジネス・データを交換するための手段を提供する Remote Function Call 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 ポリシーは、SAP JCo の現在リリースされているバージョンだけがサポートされているバージョンであることを示します。

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。

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、次のトピックについて説明します。

■ 1.1.1 項「サポートされる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る): 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コールできるようになります。
- イベント (インバウンド処理とも呼ばれる): イベントが発生したときのみ、アプリケーションが SAP R/3 のデータにアクセスできるようになります。

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。

- 双方向のメッセージ対話のサポート
- Oracle Adapter アプリケーション・エクスプローラ (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 XML スキーマと Web サービスを構築し、アダプタ・リクエストやイベント・データを処理する GUI ツールです。
- SAP R/3 への、Remote Function Call (RFC)、Business Application Programming Interfaces (BAPI)、および Intermediate Document (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 XML スキーマおよび WSDL ファイル
- BSE 用 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 概要*

1.1.1 サポートされる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 47x100
- 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 の最新のリリース・ステータスについては、Service Marketplace の SAP Note #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。また SAP の機能は、SAP 製品バージョンおよびサポート・パッケージにより異なる場合があります。

1.2 ABAP 用の古典的な 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)、および IDoc (Intermediate Documents) などの SAP R/3 インタフェースへの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、SAP R/3 のリクエストとしてアダプタで利用可能であり、また SAP がそのリモート・リクエストを呼び出すときにイベント・アダプタで利用可能です。これらは、次のように動作します。

- **Business Application Programming Interface (BAPI)** は、SAP コンポーネントをお互いにリンクする、あるいはサードパーティ製のコンポーネントにリンクするために使用するビジネス・フレームワーク内のインタフェースです。BAPI は同期してコールされ、情報を返します。
- **Remote Function Call (RFC) モジュール**は、クライアントが SAP テクノロジを呼び出してレスポンスを受信できるようにするための 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、このドキュメントで説明する例が、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 構造を特定する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。

サービス・モードで、Oracle Application Adapter for SAP R/3 は 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。

アダプタは、使用している SAP R/3 IDoc、RFC、および BAPI を基幹 SAP R/3 システム・アプリケーションやその他のエンタープライズ・アプリケーションに簡単かつ即座に統合します。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

- 一貫したデータ表記
SAP R/3 用のイベント・データとリクエスト / レスポンスのドキュメントについての標準 XML 表記を提供します。
開発者は、SAP R/3 インタフェース (BAPI、RFC、IDoc) に固有の細部の記述やターゲット SAP R/3 システムに固有の細部の構成から開放されます。
- SAP AG が発行している SAP R/3 ABAP のシリアライズ・ルールと SAP R/3 インタフェース・リポジトリ規格に適合
イベント処理の間、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。次のトピック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 のアーキテクチャ」](#)

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 は、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ています。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。

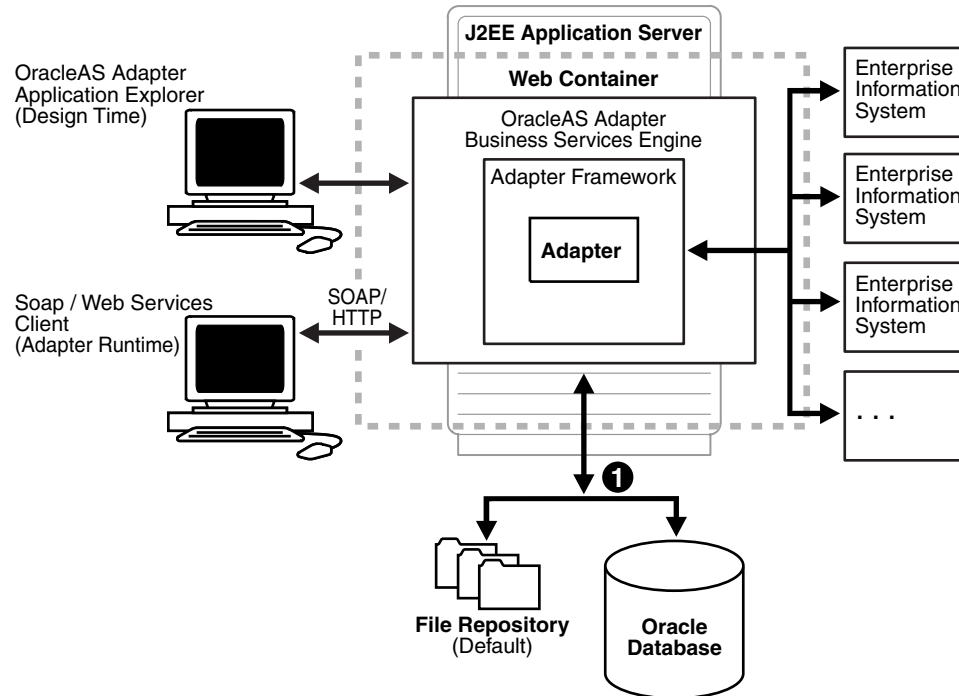
アプリケーション・エクスプローラは、BSE とともにデプロイされるデザインタイム・ツールであり、アダプタ接続の構成、EIS オブジェクトの参照、サービスの構成および EIS イベントをリスニングするためのリスナーの構成に使用されます。これらの操作の実行中に作成されたメタデータは、BSE によってリポジトリに格納されます。

BSE は、クライアントからリクエストの受信、EIS との対話、EIS からのレスポンスのクライアントへの返信に SOAP をプロトコルとして使用します。

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・リポジトリは本番環境では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



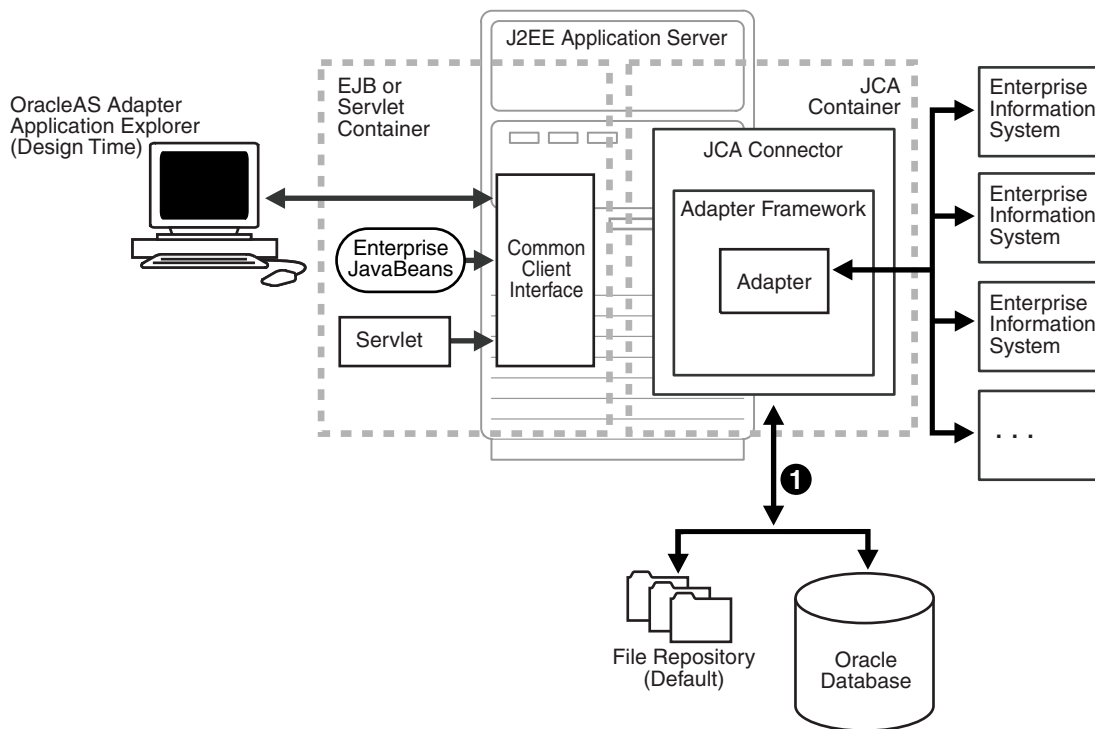
① Use either the default file repository or an Oracle database as your repository.

1.4.2 Oracle Adapter J2CA のアーキテクチャ

図 1-2 は、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ています。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。詳細については、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 のアーキテクチャ



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

関連項目：

- *Oracle Application Server Adapter 概要*
- *Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストレーション・ガイド*

1.5 BSE と Oracle Adapter J2CA デプロイメントの差異

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 または OSB) とともに使用する場合、次の点に注意してください。

- Oracle Adapter J2CA デプロイメントのみ、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) がサポートされます。
- Oracle Adapter J2CA and BSE デプロイメントでは、Oracle SOA Suite とのアウトバウンド統合 (リクエスト/レスポンス型サービス) がサポートされます。

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、Service Oriented Architecture (SOA) モデルにより厳密に準拠します。

- Oracle Adapter J2CAの方がBSEよりもわずかに高いパフォーマンスが得られます。

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

サンプル・プロジェクト 場所

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
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

Oracle Application Adapter for SAP R/3 をインストールするには、Oracle Fusion Middleware Application Adapter インストーラをダウンロードし、SOA/OSB のインストールを完了します。

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 インストール・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。

1. *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"> ■ SOA の場合: C:\12c_soa\soa\soa\thirdparty\ApplicationAdapters ■ OSB の場合: C:\12c_OSB\osb\3rdparty\ApplicationAdapters
<i>IWayConfig</i>	構成の名前。次に例を示します。 <i>jca_sample</i>

2. *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
<i>ibseroot</i>	<p><ADAPTER_HOME>\ibse.war</p> <p>次に例を示します。</p> <ul style="list-style-type: none"> ■ SOA の場合: <p>C:\12c_ soa\soa\soa\thirdparty\ApplicationAdapters\ibse.war</p> <ul style="list-style-type: none"> ■ OSB の場合: <p>C:\12c_ OSB\osb\3rdparty\ApplicationAdapters\ibse.war</p>
<i>IWay.home</i>	<p><ADAPTER_HOME></p> <p>次に例を示します。</p> <ul style="list-style-type: none"> ■ SOA の場合: <p>C:\12c_soa\soa\soa\thirdparty\ApplicationAdapters</p> <ul style="list-style-type: none"> ■ OSB の場合: <p>C:\12c_OSB\osb\3rdparty\ApplicationAdapters</p>
<i>Iway.config</i>	<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*(データベース・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*(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、チャネルの作成の詳細は、次に示すこのユーザー・ガイドの各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: [4.2 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: [4.4.1 項「BSE 用の構成の作成」](#)

- J2CA 構成の作成: [4.4.2 項「J2CA 用の構成の作成」](#)
- 作成された構成の接続: [4.4.3 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: [4.5 項「SAP R/3 用の接続 \(ターゲット\) の確立」](#)
- Web サービスの作成およびテスト: [4.9 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: [4.8 項「WSDL の生成 \(J2CA 構成のみ\)」](#)
- チャンネルの作成および使用: [4.10 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、SOASuite 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・コンポーネントの統合」](#)

1.7.7 Oracle Service Bus の使用

Oracle Application Adapter for SAP R/3 は、Oracle Service Bus (OSB) に統合されており、Web サービスの統合を可能にします。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処理の構成」](#)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 イベントの作業 \(J2CA 構成\)」](#)
- BPM: [8.6 項「BPM を使用した同期 SAP R/3 イベントの作業 \(J2CA 構成\)」](#)
- OSB: [9.5 項「Oracle Service Bus を使用した同期 SAP R/3 イベントの作業 \(J2CA 構成\)」](#)

1.7.9 追加の機能

以下に、追加の機能、およびそれらの機能に関連するこのユーザー・ガイドの項を示します。

- [接続プーリングの構成: 3.4 項「接続プール - サービス」](#)
- [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

以下に、その他の機能、およびそれらの機能に関連するこのユーザー・ガイドの項を示します。

- 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 : 11.4 項「例外フィルタの構成」
- 資格証明マッピングの構成 :
 - 11.5 項「Oracle SOA Suite (BPEL、メディアータまたは BPM) の資格証明マッピング」
 - 11.6 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

この項は、Oracle Application Adapter for SAP R/3 を使用するためのクイック・スタート・ガイドです。この章の内容は以下のとおりです。

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

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、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・ファイルが利用可能であることを検証することをお勧めします。この節では、次のトピックについて説明します。

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

2.1.1 Windows プラットフォーム上での SAP JCo の検証

以下の手順を実施して、Windows 上での SAP JCo を検証します。

1. sapjco3.jar ファイルが置かれたディレクトリに移動します。
2. sapjco3.jar ファイルを右クリックし、コンテキスト・メニューから「Open With」を選択し、Java 2 Platform Standard Edition バイナリをクリックします。

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

図 2-1 「SAP Java Connector (JCo)」 ダイアログ



Windows プラットフォーム上で、SAP Java Connector に関して必要な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ら、「Close」をクリックします。

2.1.2 UNIX プラットフォーム上での SAP JCo の検証

以下の手順を実施して、UNIX 上での SAP JCo を検証します。

1. UNIX のコマンド・プロンプトに移動します。
2. 以下のコマンドを実行します。

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

次の例で示すように、UNIX プラットフォーム上で、SAP Java Connector に関して必要な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/oraoem/Oracle/Middleware/Oracle_
Home/soa/soa/thirdparty/ApplicationAdapters/lib/sapjco3.jar
Path to JCo library: /rdbms/oraoem/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 プラットフォーム上で、SAP Java Connector についての情報をレビューします。

2.2 SAP R/3 ログオン・パラメータの特定

この項では、SAP R/3 ログオン・パラメータを特定します。このパラメータは、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します。この情報は、リファレンスとして使用することができます。この項では、次のトピック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 ユーザー・パラメータを一覧で表示して説明しています。

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 クライアントの特定	800	商業的、組織的、および技術的な用語。個別のマスター・レコードと専用のテーブル・セットを備えた、SAP システム内の自立ユニット。クライアントとしては、たとえば、企業グループがあります。
ユーザー	SAP ログイン ID.	“abc123”	対話の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 システムにアクセスするユーザーを特定または認証するための、保護された単語または文字列
言語	言語キー。EN(英語)がデフォルトです。	EN	
コードページ	キャラクタ・コード・ページの値	Cp1252	
認証モード	接続を検証する方法	選択 - 次の列を参照してください。	パスワード - 供給フィールドの値を使用 ここで述べるパスワード・パラメータとは、アプリケーション・エクスペローラのパスワード・パラメータのことです。

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 システム設定 (アプリケーション・サーバー) パラメータを一覧で表示して説明しています。

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・サーバーへの接続	iwjpsap	R/3 システムでのアプリケーション・プログラムはアプリケーション・サーバー上で動作します。メタデータ情報を取得するには、単一アプリケーション・サーバーへの接続が必要となります。
システム番号	アプリケーション・サーバー上で一意のインスタンスを特定します。	00	アプリケーション・サーバーにはいろいろなシステム番号が備わっている場合があります。管理者が提供する番号を使用してください。

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 システム設定 (メッセージ・サーバー) パラメータを一覧で表示して説明しています。

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・サーバーへの接続	iwjpsap	ロード・バランシングの目的のため、ある SAP システムからのアプリケーション・サーバーは通常、ログオン・グループで構成され、各グループが特殊なユーザーに対応します。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する責任があります。メッセージ・サーバーは、あるアプリケーション・サーバーからのリクエストをシステム内の別のアプリケーション・サーバーに渡します。また、メッセージ・サーバーは、アプリケーション・サーバー・グループについての情報とそのグループ間の現在のロード・バランシングについての情報を把握しています。この情報を使用して、ユーザーがシステムにログオンするときに、適切なサーバーを選択します。
R/3 名	アプリケーション・サーバー上で一意のインスタンスを特定します。	P47	システムを特定するために使用される SAP システムの記号名
サーバー・グループ	ログオン・グループの特定		ユーザー ID が属しているログオン・グループ

2.2.4 接続プール・パラメータ

表 2-4 接続プール・パラメータを一覧で表示して説明しています。

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	プールから割り当てることのできる接続の最大数を設定します。
接続タイムアウト	空き接続のオープンを維持する最大時間 (分)。	10	少なくとも接続タイムアウト間隔の間に使われなかった接続はクローズされます。
接続待機時間	空き接続を待機する最大時間	30(秒)	空き接続を求める接続リクエストにおいて待機する最大時間を設定します。プールが使い尽くされている場合で、指定した時間の後も使用可能な接続がない場合、キー JCO_ERROR_RESOURCE とともに 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 SAP ゲートウェイ・パラメータを一覧で表示して説明しています。

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、SAP ワールドの範囲内で CPI-C サービスを実行します。これは TCP/IP に基づいたサービスです。このサービスにより、SAP システムと外部プログラムが互いに通信を行うことができます。
SAP ゲートウェイ・サービス	サービス名を入力します (通常、サービス名とシステム番号を合成したもの)。	sapgw00	ゲートウェイ・ホストでのサービス名
プログラム ID	SAP ゲートウェイ・サービス上で指定されているプログラム識別子 (大文字/小文字を区別)	"S1PROG"	システム管理者が指定した通信のセッションのための一意の識別子。このフィールドに入力した値は、ゲートウェイ上で公開された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 ALE パラメータを一覧で表示して説明しています。

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 以降のすべての R/3 パートナ・システムでは、ポートの説明にバージョン「3」(リリース 4.0 以降)を選択する必要があります。 リリース 4.0 よりも前のすべての R/3 パートナ・システムでは、ポートの説明にバージョン「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)。	以前のリリースのセグメント定義を現在のリリースの IDoc タイプに割り当てることができます。これは、たとえば、パートナが、現在のセグメント定義ではなく現在の IDoc タイプをサポートしている古いリリースを使用している場合に必要となります。
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 - 次の列を参照してください。	IDOCDOREL は、IDOC ヘッダーの情報を使用します。 SAP リリースは、ユーザー・アカウント・ログオンから情報を取得します。 USERINPUT は、上記の 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 グローバル処理パラメータを一覧で表示して説明しています。

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メソッドを指定します。	選択 - 次の列を参照してください。	「エラー文書が作成されます」により、例外文書がフル・エラー・テキストとともに出力先に対して記述されます。 「例外がスローされます」により、java 例外が生成されます。これは、基本コンポーネント・エラーに応じて、フル・エラー・テキストが表示される場合と表示されない場合があります。
待機中コミット	コミット動作を指定します。	選択 - 次の列を参照してください。	「オフ」(デフォルト) ドキュメントでアプリケーション・サーバーにコミット・リクエストを送信します。コミット・エラーがあっても反映されません(最適パフォーマンス)。 「オン」(選択) - ドキュメントでフル・データベース・サーバー・コミットを待ってから返します。コミット・エラーは元のアダプタ・レベルに反映されます(最小パフォーマンス)。 推奨設定値については、SAP 管理者に問い合わせてください。
SAP トレース	SAP Java Connector のトレース動作を有効にします。	選択 - 次の列を参照してください。	「オフ」(デフォルト) - ハード・エラーのみが追加モードでトレース・ファイルに記録されます。 「オン」 - 各リクエストごとに個別の rfc*.trc と JCO*.trc が書き込まれます。エラーの検出に役立ちますが生産システムでは推奨されません。
トレース・レベル	SAP トレースでの詳細レベルを示します。	5	0 ~ 10 の範囲の値をリストから選択します。

2.2.8 SNC パラメータ

表 2-8 SNC パラメータを一覧で表示して説明しています。

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 を有効にするためのフラグ	1(オン)	必須
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	プロファイル・パラメータ snc/identity/as にアプリケーション・サーバーの SNC 名が見つかります。
SNC レベル	接続で使用する保護のレベルを指定します。	選択 - 次の列を参照してください。	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: snc/ データ保護の値を使用 / アプリケーション・サーバー上で使用 9: snc/ データ保護の値を使用 / アプリケーション・サーバー上で最大 (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、正しい SNC 名を確実に接続で使用できるように、これ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とファイル名を指定します。	C:\SAP J2EE_ Engine\SAPCryp tolib\sa pcrypto.dll	デフォルトは、環境変数 SNC LIB で定義したシステム定義ライブラリです。

SAP Java Connector 3.0xx について

この章では、SAP Java Connector (SAP JCo) 3.0xx に関連付けられている Oracle Application Adapter for SAP R/3 の新しい使用方法について説明します。内容は次のとおりです。

- 3.1 項「サポート対象プラットフォーム」
- 3.2 項「接続管理」
- 3.3 項「接続プール - イベント」
- 3.4 項「接続プール - サービス」
- 3.5 項「単一サーバーのサポート」
- 3.6 項「サーバー・スレッド」
- 3.7 項「変更パラメータとテスト・パラメータ」
- 3.8 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 ポリシーは、SAP JCo の現在リリースされているバージョンだけがサポートされているバージョンであることを示します。

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。

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

3.1 サポート対象プラットフォーム

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となります。

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 3.0.14 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 接続プール - イベント

チャネル構成で「プール・サイズの受け入れ」設定が **TRUE** に設定されている場合、各接続プール・パラメータをターゲットごとに手動で調整することができます。デフォルトモードでは、これまでと同様に自動モードが使用されるため、特定のチャネルを調整していない限り、ターゲットを再調整する必要はありません。

各サーバー・スレッドには最初にメタデータ接続があり、メタデータ・リポジトリは、自動モードで使用される場合、自動的に別の接続を使用します。必要なオブジェクトがメタデータ・リポジトリに追加されると、このプールは、通常、1つまたは2つの接続に迅速に縮小されます。

概要

「リスナー」または「サーバー」モードでは、iWay アダプタは、SAP からリクエストを受信し、別のエンドポイントに配信するかエンドポイント・プロセスを実行することでリクエストを処理します。リクエストの処理には、関数または受信した IDoc、パラメータ、表、フィールドに関するメタデータが必要です。各テーブルまたはフィールドには、タイプ、長さ、オフセットなどの情報があります。

このメタデータ情報は、SAP サーバーから問合せされた後、リスナー・セッションの期間中、ローカル・リポジトリ・ストアにキャッシュされます。この取得とキャッシュのメカニズムでは、通常、最初に最大量のメタデータが必要となるため接続プールの初期サイズは大きくなり、その後、接続プールはリポジトリ・キャッシュのニーズに適したサイズに縮小されるため小さくなります。

SAP JCo は接続管理を制御します。リスナー・チャネルは、接続パラメータを収集して JCo に渡し、JCo は、接続を作成し、接続を iWay に渡して作業を実行します。JCo は、接続プーリングの多くの実装とは異なる方法で接続プーリングを実行します。JCo 接続プールは、常に、1つの接続から始まります。2つ目のリクエストが到着したときにその1つの接続がビジー状態である場合、2つ目の接続が開かれます。タイムアウト期間が経過すると、未使用の空き接続がすべて閉じられます。接続プールがサイズ制限に達すると、直ちにすべての未使用の接続が閉じられます。サーバーのキープ・アライブ機能を動作させるため、接続プール・サイズの下限值は1になっています。

各「リスナー」または「サーバー」は、デフォルトでは、3つのサーバー接続で構成されます。

- リスナー・チャネル構成でパラメータが公開される接続の数。
- メタデータのリクエストを生成する各接続。
- 特定のメタデータ・リクエスト自体を自動的に生成する JCo リポジトリ。

最適なリポジトリ・パフォーマンスを実現するために、3:1 というサイズの JCo サーバー・スレッドと SAP エミッタ・スレッドの比率が使用されます。

サーバー接続の数を減らすことは可能ですが、推奨されません。これはチャンネルのパフォーマンスに重大な影響を及ぼしますが、スループットの低いサーバーではこの間接的な方法でメタデータ接続の数を減らすことができます。接続数を減らすと、サーバーのスループットや、特定の時間内に処理できるリクエストの数が低下します。

関数または IDoc を送出する SAP 送信スレッドの数は SAP QOUT スケジューラで増やすことができます。詳細は、SAP ヘルプの QOUT スケジューラのシステム・リソースの構成に関するトピックを参照してください。その後、3:1 の比率に従ってリスナー・スレッド数を調整してください。これにより、複数の iWay インスタンスを使用する場合の並列処理が可能になります。詳細なトピックについては、『Oracle SAP ベスト・プラクティス・ガイド』を参照してください。

表 3-1 接続チューニング・パラメータの説明 (サーバー・モードではメタデータのみに影響)

チューニング・パラメータ	説明
プール・パラメータの受け入れ (True False)	False の場合は、デフォルトのアダプタ接続チューニングを使用します。 True の場合は、ユーザーが指定した値を使用します。
接続プール・サイズ	プールの最大接続数。この値が「最大接続数」よりも小さい場合、プール・サイズは JCo によって「最大接続数」のサイズに設定されます。
最大接続数	一度に作成される最大接続数。
接続タイムアウト (分)	未使用の接続が解放されるまでの時間 (分)。
接続待機時間 (秒)	すべての接続がビジー状態の場合に空き接続ハンドルを待機する時間 (秒)。この時間を経過すると、リソース例外がスローされます。

注意

接続プール・サイズの割り当てが小さすぎると、各サーバー接続は前のリクエストが解放されるまでメタデータ接続を待機する必要があるため、スループットが低下します。

サイズが小さすぎる場合、接続タイムアウトにより、接続がプールから早期に削減されたときに、新しい接続を作成するためにさらなる SAP、Oracle および iWay リソースの割り当てが必要となります。

接続の割り当てが多すぎると、SAP、Oracle および iWay システムにおいて他の場所で利用することもできた貴重なリソースが枯渇することになります。「フェッチとキャッシュ」というルールは、リスナー用のリポジトリにメタデータがキャッシュされた後、使用可能な最小数に接続が縮小されることを示します。

SAP サーバー上の様々なリソース割り当ておよびセキュリティ上の目的のため、アダプタおよびリスナーのすべてのユーザーを SAP の「ダイアログ「A」」タイプのユーザーではなく「通信「C」」タイプのユーザーにすることをお勧めします。

3.4 接続プール - サービス

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・エクスペローラでターゲットが作成されます。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削減し、アプリケーションの応答時間を高速化するために、この機能を使用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、SAPJava 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・チェッカ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せん。

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ることができます。

共有 JCo 機能の有効化

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。

デフォルトでは、共有 JCo 機能は、次のディレクトリにある `mysap.settings.xml` ファイルで有効化されています。

```
<ADAPTER_HOME>\lib\settings\mysap.settings.xml
```

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

共有 JCo 機能の無効化

共有 JCo 機能を無効化するには、次のステップを実行します。

1. アプリケーション・サーバーが現在実行中の場合には停止します。
2. 次のディレクトリにある `mysap.settings.xml` ファイルを編集します。

```
<ADAPTER_HOME>\lib\settings\mysap.settings.xml
```
3. `ShareJCO` オプションを `false` に設定します。
4. `mysap.settings.xml` ファイルを保存します。

3.5 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・スレッド数と SAP ゲートウェイのアクティブなサーバー・スレッド数が一致するよう、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。

3.6 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。内容は次のとおりです。

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

3.6.1 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。

3.6.2 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、スレッド数のパラメータを 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。

3.6.3 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やし、スレッド数を増やして、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 システム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・バランシングを有効にしないで、同じプログラム ID に複数のサーバー・チャンネルを登録し、Oracle のスキーマ検証を実施しない場合、チャンネルまたは宛先上にルーティングがないため、誤ったチャンネルで IDoc を受信する可能性があります。

3.7 変更パラメータとテスト・パラメータ

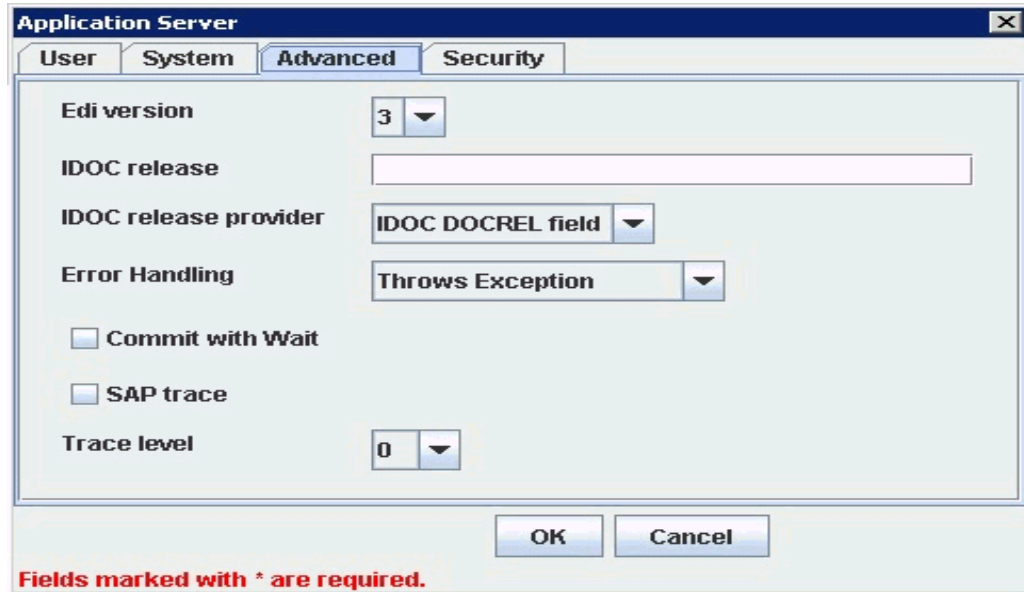
SAP 変更パラメータでは、ネスト構造 (列自体が構造である表) や固定長のないデータ (文字列データ型など) を含む、ほとんどすべての SAP R/3 タイプを使用できます。変更パラメータは、関数モジュールへの入力や関数モジュールからエクスポートのどちらにも使用できます。これは、特定の SAP 関数の設計者の要件によって異なります。表パラメータは、定義された長さの固定列と必須ヘッダー行を含むフラット・データ構造 (リレーショナル) に制限されています。変更パラメータでは、ネスト構造 (列自体が構造である表) や固定長のないデータ (文字列データ型など) を含む、ほとんどすべての SAP R/3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

3.8 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。0～10 の範囲の値をリストから選択します。使用される共通の値は、次のとおりです。

- 0- なし
- 1- エラーおよび警告
- 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 &EDjco.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・システムのレベルで設定することができます。

```
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。

特にホストされる環境では、環境設定をチェックすることをお勧めします。

Oracle Application Adapter for SAP R/3 の構成

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。

この章の内容は次のとおりです。

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

4.1 SAP R/3 のネスト構造およびネストされた表について

SAP R/3 リモート関数モジュールには、複数のタイプのパラメータ・リストを含めることができます。パラメータという用語は、変数に名前を付けるため、および関数への入力として提供されるデータの 1 つを指すために使用されます。すべての SAP R/3 リモート関数モジュールは厳密にタイプ指定されています。つまり、パラメータに渡すことができるデータのタイプは関数の定義で事前に定義されています。パラメータ・リストは、パラメータのタイプまたは方向によってグループ化されます。関数には、次のタイプのパラメータ・リストがあります。

- インポート。関数に渡される値。
- エクスポート。関数から渡される値。
- 変更。関数に渡して関数で変更できる値。
- 表。関数との間でやり取りできるリレーショナル・タイプのデータ。

例外:一部の関数には厳密にタイプ指定された例外があります。これらはスキーマには含まれませんが、エラー時に、エラー・キーおよび値とともに、関数によって (ABAP 例外の形式で) 正式に生成されます。

Oracle Application Adapter for SAP R/3 には、次の操作モードが用意されています。

- **同期処理。**SAP R/3 への RFC および BAPI の送信。
- **非同期処理。**(トランザクション RFC を介した)SAP R/3 への同期送信: IDoc 対応の BAPI および IDoc。
- **SAP からの同期イベントの受信と SAP へのリプライ。**RFC、および RFC の形式の BAPI。
- **非同期処理。**(トランザクション RFC を介した)SAP R/3 からの同期受信: IDoc 対応の BAPI および IDoc。
- **同期受信のみ。**RFC、および RFC の形式の BAPI。

リモート関数呼び出し (RFC) は、オプションまたは必須として定義されたパラメータによって定義できます。パラメータがオプションであり、値が渡されない場合、パラメータは非アクティブとしてマークされます。パラメータが必須である場合、パラメータに値が割り当てられるまで関数を呼び出すことはできません。

すべてのデータは、リモート関数モジュールとの間で値によってやり取りされます。つまり、実際の値はコールとともに送信されます。

すべてのパラメータは、SAP R/3 データ・ディクショナリで定義されているデータ・タイプによって定義する必要があります。関数のコール元は、このディクショナリを呼び出して、パラメータのタイプを確認し、適切な値を渡すことができます。

これらは基本および複合として定義されている ABAP データ・タイプであり、その使用法はパラメータ・タイプおよびデータ・タイプの定義に役立ちます。使用可能な基本タイプは、次のとおりです。

数値:

- 整数 - I
- パック - P
- 浮動 - F

文字:

- 文字 - C
- 日付 - D
- 時刻 - T
- 数字 - N

バイト:

- 16 進 - X

複合データ・タイプは、基本タイプから構築され、SAP R/3 データ・ディクショナリで次のように定義されます。

- **構造。**名前によって定義され、1つの単位として扱われる基本タイプ (列) のグループ。
- **表。**構造の場合と同様の基本タイプのグループで、複数の行 (行) にまたがります。次のデータ・タイプでは、可変長レコードが処理されます。

- **STRING**。文字のグループで、長さは文字数 * 文字幅です。
- **XSTRING**。バイトのグループで、長さはバイト数です。

関数パラメータで **STRING** または **XSTRING** を使用する場合、データを XML に変換し、トランスポートを介して送信した後、データ・タイプに変換し直す必要があるため、RFC よりも関数の実行が遅くなります。構造内で **STRING** か **XSTRING** または表を使用すると、構造が **ディープ構造** になります。

SAP R/3 では、**IMPORT** パラメータおよび **EXPORT** パラメータの使用は基本データ・タイプまたは構造に制限されていました。SAP ERP ECC の最新リリースでは、任意のパラメータ・リストで任意のデータ・タイプを使用できるようにする機能が追加されました。現在、**IMPORT** パラメータは、基本タイプを含むことができるのとまったく同じように表を含むことができます。

フラット構造には、固定長 (内部表または文字列なし) の基本データ・タイプのみが含まれます。

ネスト構造

構造には、それ自身が構造または表である列を含めることができます。これにより、構造は **ネスト構造** になります。いずれかの要素に可変長要素または表が含まれている場合は、**ディープ・ネスト構造** になります。

ネスト構造のサンプル XML 構造を次に示します。

```
<REQHDR>
  <MESSGID/>
  <TMSTP/>
  <CORRID/>
  <CONSID/>
  <PROVID/>
  <TRINFO/>
  <TRACK>
    <ZBUSID/>
    <ZCONVID/>
    <ZUSERID/>
  </TRACK>
</REQHDR>
```

構造 **REQHDR** には、それ自身が構造である **TRACK** という列があります。

ディープ・ネスト構造 のサンプル XML 構造を次に示します。

```
<SYSTEMINFO>
  <FDPOS/>
  <COLNO/>
  <DTSTRING/>
  <SYST2>
    <STINFO/>
    <TABIX/>
    <TFILER/>
  </SYST2>
</SYSTEMINFO>
```

構造 **SYSTEMINFO** には、**DTSTRING** という文字列と **SYST2** という構造があります。

表は、基本タイプを含むことも、基本タイプで構成することも、ネスト構造およびネストされた表を含むこともできます。

次のリストに、使用可能なタイプの組み合わせを示します。

- 固定長の一連の基本データ・タイプで構成される構造 (非ネスト、フラット構造)。

- 行タイプが基本タイプである内部表。
- 行タイプが非ネスト構造である内部表。これは実表とみなすことができます。
- 構成要素として構造を含む構造 (ネスト構造、フラットまたはディープ)。
- 構成要素として内部表を含む構造 (ディープ構造)。
- 行タイプにさらに内部表が含まれる内部表。

ネストされた表のサンプル XML 構造を次に示します。

```
<INBOUNDINFO>
<_ -SAPCND_ -SRMA_ LIN>
  <item>
    <CLIENT/>
    <SCALE_ID/>
    <SCALE_LINE_ID/>
    <NUM_FLIGHTS/>
    <DBACTION/>
  </item>
</_ -SAPCND_ -SRMA_ LIN>
<_ -SAPCND_ -SRMDEF>
  <item>
    <CLIENT/>
    <SCALE_ID/>
    <SCALE_BASE_TYPE/>
    <SCALE_TYPE/>
    <DBACTION/>
  </item>
</_ -SAPCND_ -SRMDEF>
```

Oracle Application Adapter for SAP R/3 は、これらのタイプをすべてサポートしています。ただし、タイプ (定義) 自体が行タイプであるネストされた表を除きます。これは、構造が複雑であるため、要求があった場合にのみサポートされます。

4.2 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、次の手順を実行します。

1. 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 Oracle WebLogic Server が起動し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移動します。

```
<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 にアクセスして、WSDL ファイルを Oracle Service Bus (OSB) コンソールに公開できます。

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移動します。

```
<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 システムへの新規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
```

4.3 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 インストール・ガイド』を参照してください。

4.4 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 とともに使用する前に、リポジトリ構成を作成する必要があります。アダプタのデプロイ先のコンテナに基づき、Web サービスと J2CA の 2 種類のリポジトリ構成を作成できます。

この節では、次のトピックについて説明します。

- 4.4.1 項「BSE 用の構成の作成」
- 4.4.2 項「J2CA 用の構成の作成」
- 4.4.3 項「BSE または J2CA 構成への接続」

デザインタイム時に、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。詳細については、1-1 ページ「[アダプタの機能](#)」を参照してください。

4.4.1 BSE 用の構成の作成

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。

この節では、次のトピックについて説明します。

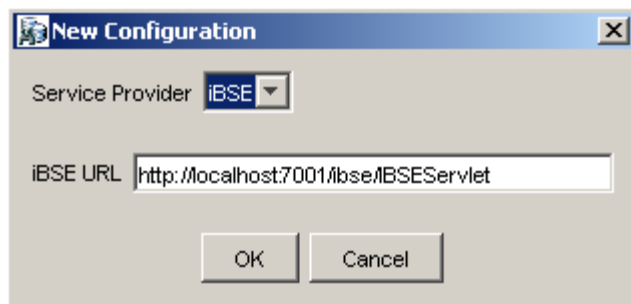
- 4.4.1.1 項「BSE 用の新規構成の定義」

4.4.1.1 BSE 用の新規構成の定義

BSE の新規構成を定義するには、次の手順を実行します。

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、「新規」を選択します。
「新規構成」ダイアログが表示されます。
3. 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



4. 図 4-1 に示すように、「サービス・プロバイダ」リストから「iBSE」を選択します。
5. 「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 に置き換えます。

`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、*port* は管理対象 Oracle WebLogic Server (soa_server1 など) の HTTP ポート番号です。

6. 「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



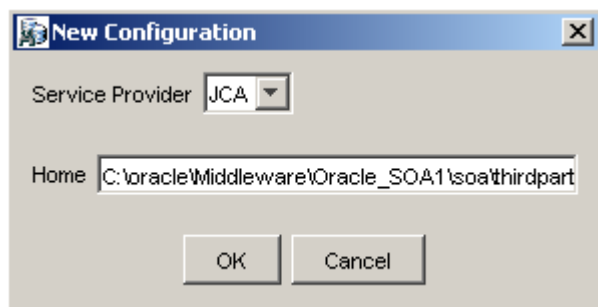
4.4.2 J2CA 用の構成の作成

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

J2CA の新規構成を定義するには、次の手順を実行します。

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、「新規」を選択します。
「新規構成」ダイアログが表示されます。
3. 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」リストから「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)。

4.4.3 BSE または J2CA 構成への接続

新規構成に接続するには、次の手順を実行します。

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。イベントを構成できるのは、J2CA 構成のみを使用している場合です。

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」ノードを使用して、SAP R/3 とのインバウンド相互作用を作成します。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」ノードを使用して、SAP R/3 のイベントをリスニングするリスナーを構成します。
- 「ビジネス・サービス」ノード (BSE 構成でのみ使用可能) を使用して、Web サービスをテストします。「ビジネス・サービス」ノードのセキュリティ機能を使用して、Web サービスのセキュリティ設定を制御することもできます。

これで、SAP R/3 に対して新規ターゲットを定義できます。

4.5 SAP R/3 用の接続 (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が含まれます。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、そのターゲットに固有の情報が必要です。

この節では、次のトピックについて説明します。

- 4.5.1 項 「SAP R/3 へのターゲットの定義」
- 4.5.2 項 「定義済みの SAP R/3 ターゲットへの接続」
- 4.5.3 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。ターゲットを定義すると、自動的に保存されます。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、SAP R/3 システムに接続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。

4.5.1 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および SNC の追加プロトコルをサポートします。SAP R/3 アプリケーション・サーバーに接続すると、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って管理されます。SAP アプリケーション・セキュリティの詳細は、該当する SAP ドキュメントを参照してください。

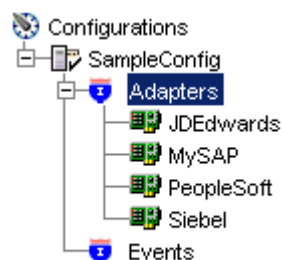
J2CA 構成で作業している場合、ターゲットを作成、更新および削除するには、アプリケーション・サーバーを再起動する必要があります。また、ターゲットの作成後、ターゲットへの接続後、および SAP R/3 ビジネス・オブジェクト用の WSDL の生成後にも、アプリケーション・サーバーを再起動する必要があります。さらに、アプリケーション・サーバーを再起動する前に、必ずアプリケーション・エクスプローラを閉じてください。

ターゲットを定義するには、次の手順を実行します。

1. 図 4-6 に示すように、左ペインで「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、「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、記述名 (「SAPTarget」など) 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」リストから、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) があります。

注意： ロード・バランシングの目的のため、ある 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、ログオン・グループで構成され、各グループは特定の種類のユーザーを処理します。各グループ内のアプリケーション・サーバーは、最小負荷の方式によ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。

図 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

次のタブが使用可能です。

- 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) では、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定義されたクライアント数
ユーザー	SAP R/3 アプリケーション用の有効なユーザー ID
パスワード	SAP R/3 アプリケーション用の有効なパスワード
言語	言語キー。EN(英語) がデフォルトです。
コードページ	キャラクタ・コード・ページの値
認証モード	SAP R/3 システムに接続するとき使用する認証モード。デフォルトでは、リストから「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。


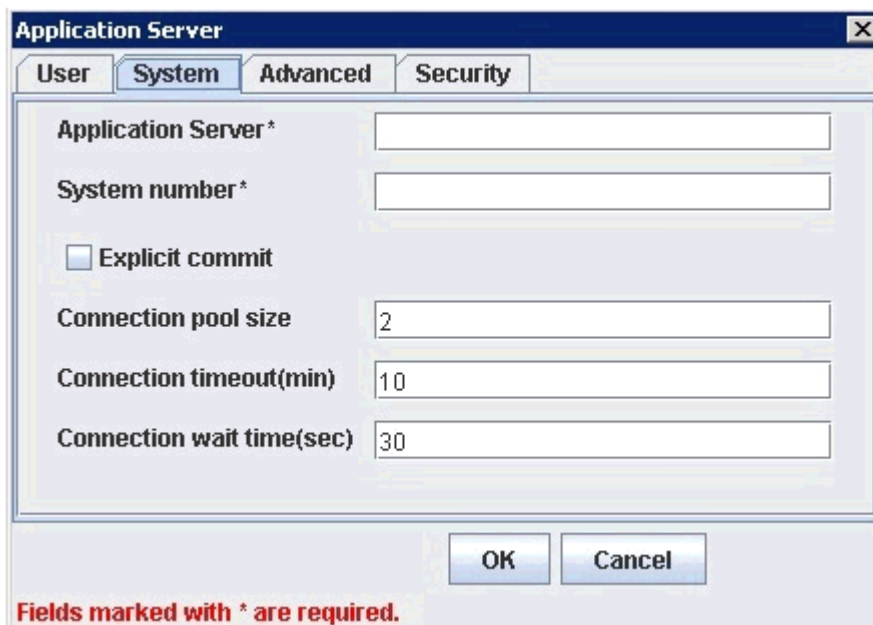
5.  4-9 に示すように、「システム」タブ (必須) では、この項の情報に基づいて SAP R/3 ターゲットに適切な情報を入力します。

図 4-9 「システム」タブ



「システム」タブを使用すると、接続している SAP R/3 システムのアプリケーション・サーバー名、システム番号、および接続プーリング情報を入力することができます。

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。

表 4-2 「システム」タブのパラメータ (続き)

ターゲット・パラメータ	説明
システム番号	クライアント通信用に SAP R/3 に対して定義されているシステム番号
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。このオプションを有効にすると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、呼び出されて即時に戻ります。引き続きデータベース・コミットが失敗する可能性はありますが、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	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。 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。 アプリケーション・エクンプローラを使用して Web サービスを作成する場合、ランタイム時に、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。
接続タイムアウト (分)	接続プールのタイムアウト値 (分)。デフォルト値は 10 分です。
接続待機時間 (秒)	接続プールの待機時間 (秒)。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することができます。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 システムで必要なリソースについて詳細に説明されています。

6. 「詳細」タブ (任意) では、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。デフォルト値は、バージョン 3 です。
IDOC リリース	接続に使用したい IDOC のバージョンニング
IDOC リリース・プロバイダ	接続用の IDOC リリース・プロバイダ。リストから、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>例外のイベントのリストから、「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>次のルールに従います。</p> <ul style="list-style-type: none"> アプリケーションが Java 中心 の場合は、コード・コンポーネントが例外を捕捉し、それに応じ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。 アプリケーションが ドキュメント・ベース の場合は、「エラー・ドキュメントが作成されます」を選択して、Java 例外を含む XML ドキュメントを作成します。 <p>XML 文書の読み取りおよびエラーの把握は、アプリケーション側の役割です。</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>アプリケーションで高度な正確さが要求される場合は、「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、選択する前に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・キューに格納されます。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、ポスト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・エラーが発生した場合、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」タブで有効にされ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、同期コミットが有効になります。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、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です。</p>
SAP トレース	<p>このオプションを選択して SAP トレースを有効にします。</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><ADAPTER_HOME>\tools\iwae\bin</pre> <p>ランタイム :</p> <pre><ORACLE_HOME>\user_projects\domains\base_domain</pre> <p>SAP トレースに使用するファイル名のフォーマットは、次のとおりです (デザインタイムおよびランタイム)。</p> <ul style="list-style-type: none"> ■ rfc02664_04332.trc ■ JCO100226_043846994.trc
トレース・レベル	<p>SAP トレースでの詳細レベルを示します。0 ~ 10 の範囲の値をリストから選択します。</p>

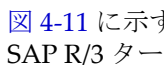
7.  4-11 に示すように、「セキュリティ」タブ (任意) では、この項の情報に基づいて SAP R/3 ターゲットに適切な情報を入力します。

図 4-11 「セキュリティ」タブ

The screenshot shows the 'Application Server' dialog box with the 'Security' tab selected. The fields are as follows:

- SNC mode: 0 (dropdown)
- SNC partner: (empty text box)
- SNC level: 1 (dropdown)
- SNC name: (empty text box)
- SNC library path: (empty text box)

Buttons: OK, Cancel

Note: Fields marked with * are required.

セキュリティタブにより、接続している SAP R/3 システムの SNC (Secure Network Communication) 情報を指定することができます。

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 は無効です。SNC を有効にするには、リストから「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	リストから 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。

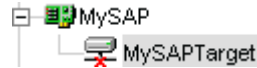
SNC により、SAP R/3 システムの分散コンポーネント間の通信リンクが保護されます。SNC を使用して、SAP R/3 は GSS-API バージョン 2 標準に準拠している製品をサポートできます。SNC は、アプリケーション・レベル (エンドツーエンド・セキュリティ)、スマートカード認証およびシングル・サインオンをサポートしています。

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

- ターゲットについて必要なすべての情報を指定したら、「OK」をクリックします。抽出が終了すると、[図 4-12](#) に示すように、新規ターゲット「MySAP Target」が「MySAP」アダプタ・ノードの下に表示されます。

図 4-12 MySAP Target



これで、SAP R/3 ターゲットに接続できます。

アダプタのスキーマの作成方法の詳細は、4-18 ページ「[XML スキーマの作成](#)」を参照してください。

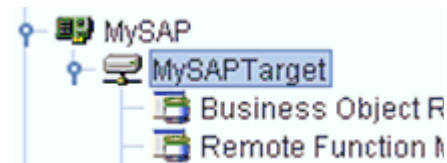
4.5.2 定義済みの SAP R/3 ターゲットへの接続

既存のターゲットに接続するには、次の手順を実行します。

- 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（「SAPTarget」など）をクリックします。「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。
- 接続パラメータを確認します。
- ターゲット名を右クリックして、「接続」を選択します。

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

図 4-13 接続された MySAP Target



4.5.3 SAP R/3 への接続の管理

この節では、次のトピックについて説明します。

- [4.5.3.1 項「SAP R/3 への接続の切断」](#)
- [4.5.3.2 項「接続パラメータの変更」](#)
- [4.5.3.3 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に対して開かれた複数の接続を維持管理できますが、使用していない接続を切断することをお勧めします。

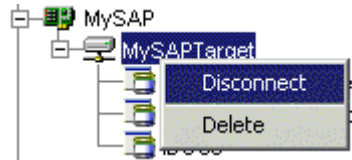
- ターゲットを編集します。
システム・プロパティを変更する際に接続パラメータを変更できます。切断後に既存のターゲットを変更できます。
- 不要になった接続を削除します。

4.5.3.1 SAP R/3 への接続の切断

ターゲットを切断するには、次の手順を実行します。

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、接続先のターゲット (「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、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



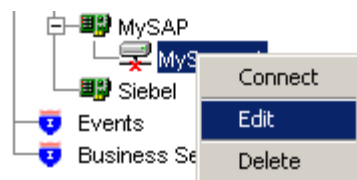
4.5.3.2 接続パラメータの変更

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、以前に入力したいずれの情報も編集できます。ターゲットを編集した後、Oracle WebLogic Server を再起動して、ランタイムの目的で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、次の手順を実行します。

1. 編集するターゲットの接続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、「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せん。

- 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。

4.5.3.3 SAP R/3 への接続の削除

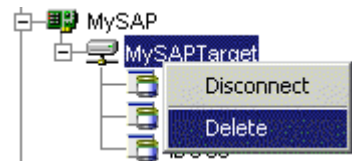
切断は、単に切断したり閉じるだけではなく、削除できます。接続を削除すると、アプリケーション・エクスプローラの左ペインにある SAP R/3 接続のリストにノードが表示されなくなります。

接続を削除する場合、ランタイムの目的で Oracle WebLogic Server を再起動して、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、次の手順を実行します。

- 削除するターゲットを見つけます。
- 図 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からそのノードが消えます。

4.6 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、アプリケーション・エクスプローラを使用して、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 SAP R/3 ビジネス・オブジェクトを検索して参照することができます。

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、このドキュメントで説明する例が、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。

4.7 XMLスキーマの作成

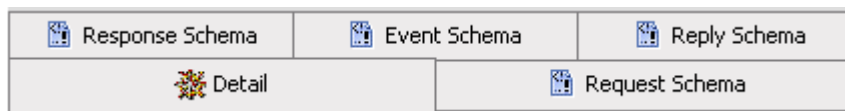
SAP R/3 ビジネス関数ライブラリを検索してオブジェクトを選択した後、アプリケーション・エクスプローラを使用してその関数の XML リクエスト・スキーマと XML レスポンス・スキーマを作成できます。

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、次の手順を実行します。

1. 4-15 ページ「定義済みの 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」ノードの左にあるアイコンをクリックします。
4. スクロール・ダウンして、「CompanyCode」ビジネス・オブジェクトの左にあるアイコンをクリックします。
5. スクロール・ダウンして、「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。

4.8 WSDL の生成 (J2CA 構成のみ)

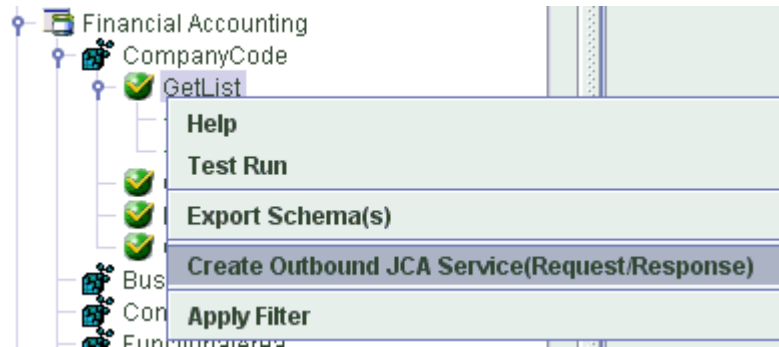
Web Service Definition Language (WSDL) を使用してサービスを記述すると、そのサービスをホスト・サーバー内の他のサービスで使用できるようになります。アプリケーション・エクスプローラを使用して、アダプタのリクエスト/レスポンス (アウトバウンド) とイベント通知 (インバウンド) の両方の J2CA サービスを作成します。

注意： インバウンド JCA サービスの作成 (イベント) 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、次の手順を実行します。

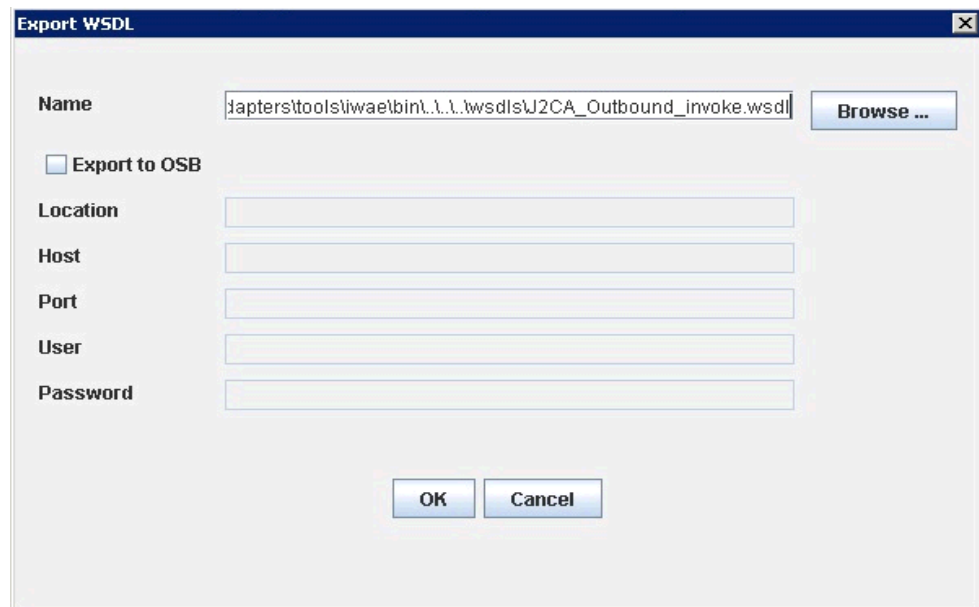
1. スキーマの作成後、各オブジェクトを右クリックします。
図 4-19 に示すように、「GetList」メニューが表示されます。

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



2. アウトバウンド JCA サービスの作成 (リクエスト/レスポンス) 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 ファイルの名前の末尾には `_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が、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。詳細については、6-37 ページ「[イベント統合のための WSDL の生成](#)」を参照してください。

4.9 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 (**ビジネス・サービス**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 からマテリアル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 を使用するように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。

この節では、次のトピックについて説明します。

- 4.9.1 項「Web サービスの作成」
- 4.9.2 項「Web サービスのテスト」
- 4.9.3 項「伝播の識別」

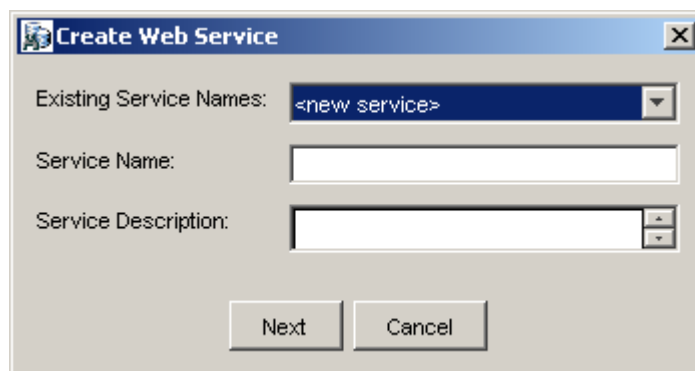
4.9.1 Web サービスの作成

SAP R/3 ビジネス関数の Web サービスを作成するには、次の手順を実行します。

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



次のステップを実行します。

- a. 「既存のサービス名」リストから、<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、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のリストで 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。
「ライセンスおよびメソッド」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、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・エクスプローラで、「ビジネス・サービス」ノードに表示が切り替わり、新規の Web サービスが左ペインに表示されます。
 6. 新規の Web サービスを右クリックし、メニューから「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。

4.9.2 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、次の手順を実行します。

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。
ビジネス・サービスの名前は、右ペインにリンクとして表示されます。
4. 右ペインで、指定するビジネス・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。

4.9.3 伝播の識別

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は、有効である必要があります。SAP R/3 への接続に使用されます。アプリケーション・エクス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 ヘッダーのサンプルを示します。

```
<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。

4.10 イベント・アダプタの構成

イベントは、データベース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。イベントを使用して、アプリケーションでのアクションをトリガーできます。たとえば、データベースへの更新は、顧客情報への更新を反映できます。これ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、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てチャンネルの作成が可能になる前に 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。詳細については、A-9 ページ [A.3 項「SAP アウトバウンド・メッセージの処理」](#) を参照してください。

この節では、次のトピックについて説明します。

- [4.10.1 項「チャンネルの作成および編集」](#)
- [4.10.2 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、各チャンネルは一意の 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 つの一意の 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接続を表します。詳細については、4-23 ページ「[チャンネルの作成および編集](#)」を参照してください。

4.10.1 チャンネルの作成および編集

次の手順では、イベント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。

この項では、次のトピックについて説明します。

- [4.10.1.1 項「チャンネルの作成」](#)
- [4.10.1.2 項「チャンネルの編集」](#)
- [4.10.1.3 項「チャンネルの削除」](#)

チャンネルを作成、変更または削除する場合、ランタイムの目的で Oracle WebLogic Server を再起動して、変更を適用し、リポジトリを更新する必要があります。

注意： チャンネルは、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、構成および開始できます。リモート・ホストのチャンネルの構成および開始は、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。イベントを構成できるのは、J2CA 構成のみを使用している場合です。

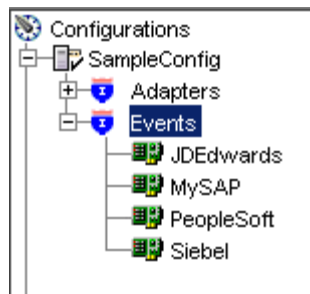
チャンネルの作成および更新には、アプリケーション・サーバーの再起動が必要です。また、チャンネルの作成後、およびインバウンド WSDL の生成後にも、アプリケーション・サーバーを再起動する必要があります。さらに、アプリケーション・サーバーを再起動する前に、必ずアプリケーション・エクスプローラを閉じてください。

4.10.1.1 チャンネルの作成

チャンネルを作成するには、次の手順を実行します。

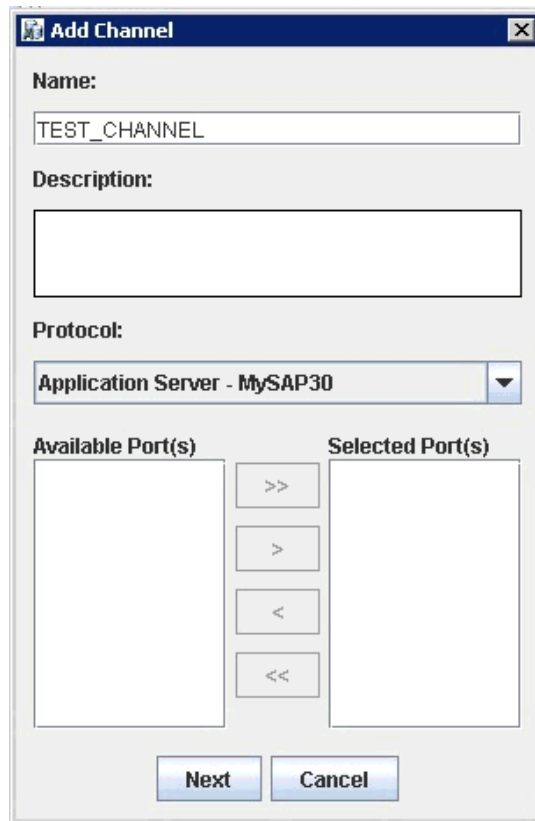
1. [図 4-22](#) に示すように、「イベント」ノードをクリックします。

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・ノードが表示されます。
3. 「チャンネル」を右クリックし、「チャンネルの追加」を選択します。
図 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



- 次のステップを実行します。
- a. 「TEST_CHANNEL」など、チャンネルの名前を入力します。
 - b. 簡単な説明を入力します。
 - c. 「プロトコル」リストから、「Application Server - MySAP30」または「Message Server - 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。次のタブが使用可能です。

- 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」タブでは、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定義されたクライアント数
ユーザー	SAP アプリケーション用の有効なユーザー ID
パスワード	SAP アプリケーション用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とき使用する認証モード。デフォルトでは、リストから「パスワード」が選択されています。

6. 「システム」タブでは、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	このチャンネルで使用したい SAP R/3 のプログラム 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 に対して定義されているシステム番号
プール・パラメータの受け入れ (True False)	False の場合は、デフォルトのアダプタ接続チューニングを使用します。 True の場合は、ユーザーが指定した値を使用します。
接続プール・サイズ	プールの最大接続数。この値が「最大接続数」よりも小さい場合、プール・サイズは JCo によって「最大接続数」のサイズに設定されます。
最大接続数	一度に作成される最大接続数。
接続タイムアウト (分)	未使用の接続が解放されるまでの時間 (分)。
接続待機時間 (秒)	すべての接続がビジー状態の場合に空き接続ハンドルを待機する時間 (秒)。この時間を経過すると、リソース例外がスローされます。

7. 「セキュリティ」タブ (任意) では、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 は無効です。SNC を有効にするには、リストから「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	リストから 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 (任意) では、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	<p>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。</p> <p>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」に変更されます。</p> <p>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。</p>
IDOC フォーマット	<p>リストから IDOC タイプを選択します。</p> <ul style="list-style-type: none"> ■ XML (デフォルト) ■ XML-CDATA-ENVELOPED
IDOC リリース	接続に使用したい IDOC のバージョン
IDOC リリース・プロバイダ	<p>接続用の IDOC リリース・プロバイダ。リストから、IDOC DOCREL フィールド (デフォルト)、SAP リリース、ユーザー入力、または自動チェックを選択します。</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>

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
SAP トレース	<p>このオプションを選択して SAP トレースを有効にします。</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： <ADAPTER_HOME>\tools\iwaie\bin</p> <p>ランタイム： <ORACLE_HOME>\user_projects\domains\base_domain</p> <p>SAP トレースに使用するファイル名のフォーマットは、次のとおりです (デザインタイムおよびランタイム)。</p> <ul style="list-style-type: none"> ■ rfc02664_04332.trc ■ JCO100226_043846994.trc
トレース・レベル	SAP トレースでの詳細レベルを示します。0 ~ 10 の範囲の値をリストから選択します。
スレッド数	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>リストから同期処理のタイプを選択します。可能な値として、「REQUEST」と「REQUEST_RESPONSE」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에만、WSDL 作成プロセス時に表示されます。それ以外の場合、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



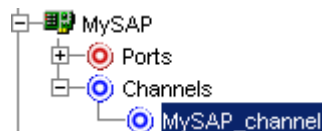
アイコン上の「X」は、そのチャンネルが現在切断されていることを示します。イベント構成をアクティブ化するには、チャンネルを開始する必要があります。

注意: Oracle Application Adapter for SAP R/3 を BPM、BPEL、メディエータまたは OSB インバウンド・プロセス・コンポーネントと統合する予定の場合は、チャンネルを起動しないでください。チャンネルは、BPM、BPEL、メディエータまたは OSB プロセス・コンポーネントがデプロイされた後、ランタイム・サーバーによって管理されます。テストまたはデバッグ目的でチャンネルをアプリケーション・エクスプローラから起動する場合は、イベントのランタイム・デプロイメントの前にすべてのチャンネルを停止し、アプリケーション・エクスプローラを閉じてください (BPM、BPEL、メディエータまたは OSB プロセス・コンポーネントと連携する場合)。

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。

チャンネルが非アクティブになり、アイコン上に「X」が表示されます。

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使用できます。

- 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、次のことを考慮してください。

- **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、指示固有ではなく、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、特定の順序で構成された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ー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 内では、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継続して追加され、セグメントの数は、各 IDOC バージョンの増加に伴って増加します。

4.10.1.2 チャネルの編集

チャネルを編集するには、次の手順を実行します。

1. 左ペインで、編集するチャネルを見つけます。
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、ランタイムの目的で Oracle WebLogic Server を再起動して、変更を適用し、リポジトリを更新する必要があります。

4.10.1.3 チャネルの削除

チャネルを削除するには、次の手順を実行します。

1. 左ペインで、削除するチャネルを見つけます。
2. チャネルを右クリックし、「**削除**」を選択します。

左ペインにある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、ランタイムの目的で Oracle WebLogic Server を再起動して、変更を適用し、リポジトリを更新する必要があります。

4.10.2 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。

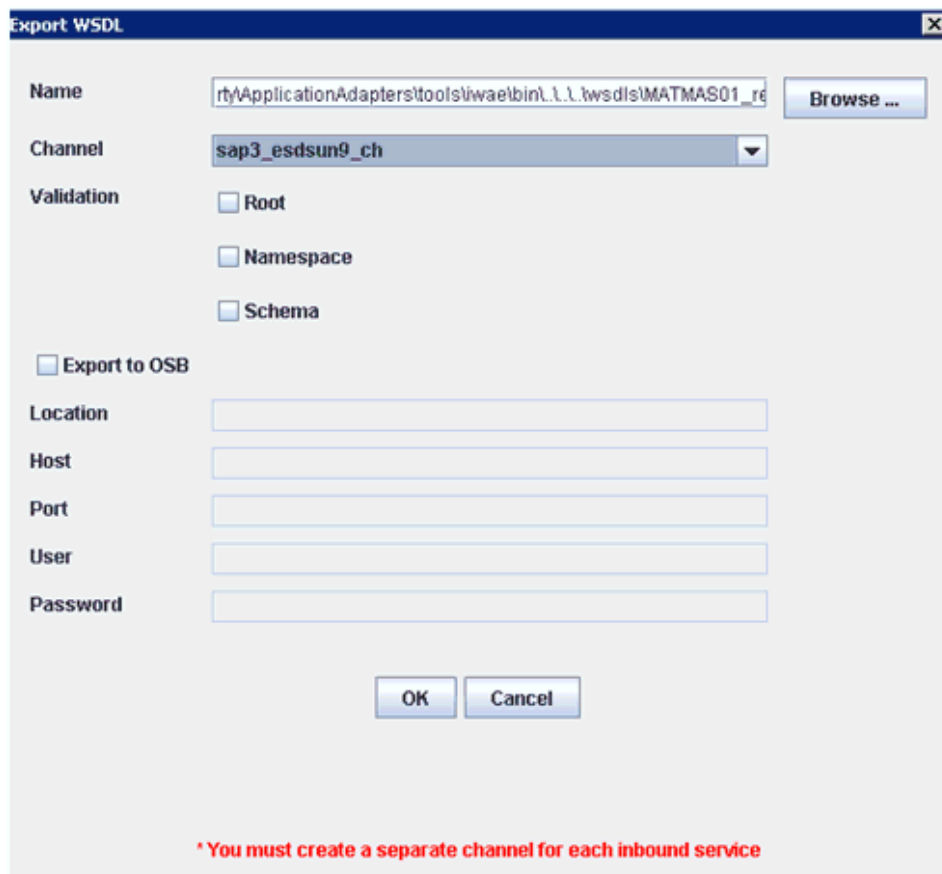
Oracle Application Adapter for SAP R/3 を使用してインバウンド処理を検証するには、次の手順を実行します。この手順では、MATMAS (Material Master) をインバウンド処理の例として使用します。

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 アダプタ用にチャン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 サービスの作成 (イベント) 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」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。
- 実行時には、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、実行時に最初の検証が失敗すると、残りの検証オプションは処理されません。
- ルート検証およびネームスペース検証は、適度な検証レベルであるとみなされます。スキーマ検証はより厳しい検証レベルです。

- ルート要素およびネームスペースが SAP 環境の IDOC 間で異なるかぎり、ルート検証とネームスペース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、BPEL プロセスを作成します。
 8. SAP の GUI 環境からトランザクション (IDOC) をトリガーします。
検証が失敗したインバウンド・トランザクションは、SAP トランザクション監視 (SM58) に示されます。
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。

4.11 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。

- [4.11.1 項「SAProuter の構成」](#)
- [4.11.2 項「Oracle コンポーネントを使用した SAProuter のテスト」](#)
- [4.11.3 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。

機能および構成に関連する 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 サイトにアクセスすると、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 のバージョンによって異なる場合があります。

4.11.1 SAProuter の構成

SAProuter を構成するには、次の手順を実行します。

1. SAProuter のルート表を編集または作成します。

ルート文字列構文のサンプルを次に示します。

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

注意： スラッシュ間の文字は、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>
```

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。ホスト名、IP アドレスまたは IP サブネットワークを指定できます。
- 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。ホスト名、IP アドレスまたは IP サブネットワークを指定できます。
- 宛先ポート `<dest serve>` 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、ピリオド (.) 文字を使用して範囲を指定する 2 つのポートを区切ることによって、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 まで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 をテストできます。

4.11.2 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。

ターゲットの作成方法の詳細は、[4.5 項「SAP R/3 用の接続 \(ターゲット\) の確立」](#)を参照してください。チャネルの作成方法の詳細は、[4.10.1 項「チャネルの作成および編集」](#)を参照してください。

ルート文字列構文のサンプルを次に示します。

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

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
```

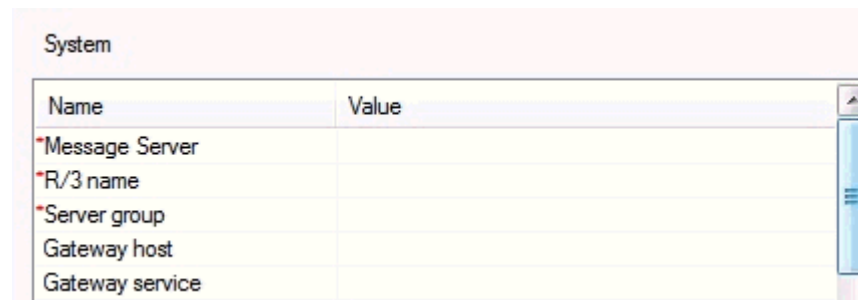
[図 4-27](#) は、アプリケーション・エクスプローラでターゲット用に構成されたルート文字列を示しています。

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、サービス名および (アプリケーション・ホストではなく) メッセージ・サーバーのポートを指定して、ルーターに提供する必要があります。

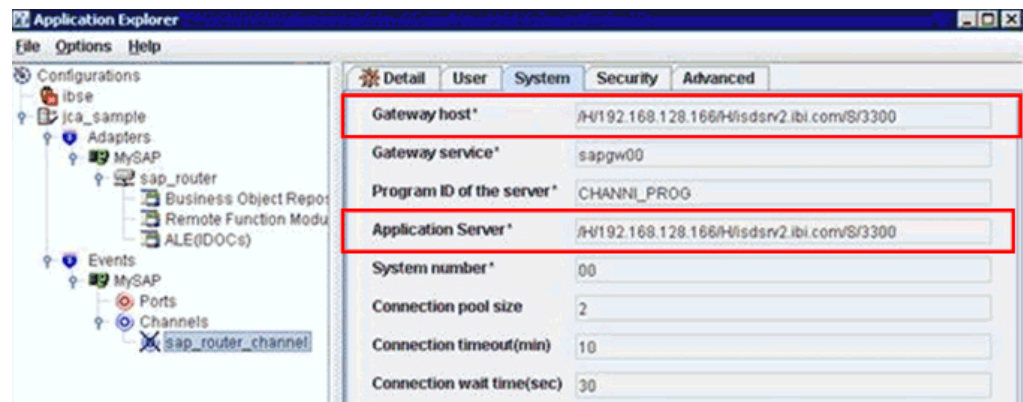
図 4-28 メッセージ・サーバーの接続パラメータ



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 は、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ています。

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、メディエータ、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。

4.11.3 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・コマンドを発行します。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
```

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 が起動す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を調べます。

図 4-30 は、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した SAProuter を示しています。

図 4-30 SAProuter コマンド・プロンプト



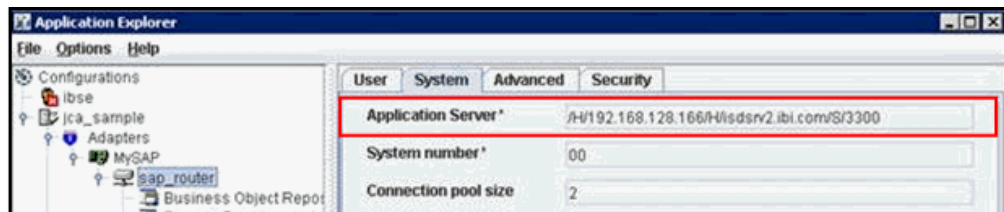
```
C:\sap_router>saprouter -r -T saproutertrace.trc -U3 -G saprouterlog.trc
trcfile saproutertrace.trc
logfile saprouterlog.trc
```

2. 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、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 は、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ています。

図 4-31 構成されたターゲット・ルート文字列



3. 図 4-32 に示すように、接続が適切に確立されたことを検証するには、定義済ログ・ファイル (saprouterlog.trc など) を参照してください。

図 4-32 SAProuter ログ・ファイル

```
saprouterlog.txt
1 Mon Apr 16 19:59:42 2012 INIT LOGFILE
2 Mon Apr 16 19:59:42 2012 READ ROUTTAB ./saprountab 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 (isdsrv2.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 (isdsrv2.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
```

4. 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 設定をランタイム時に利用できます。

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。内容は次のとおりです。

- 5.1 項「Oracle WebLogic Server とのアダプタの統合」
- 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 概要*

5.1 Oracle WebLogic Server とのアダプタの統合

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。

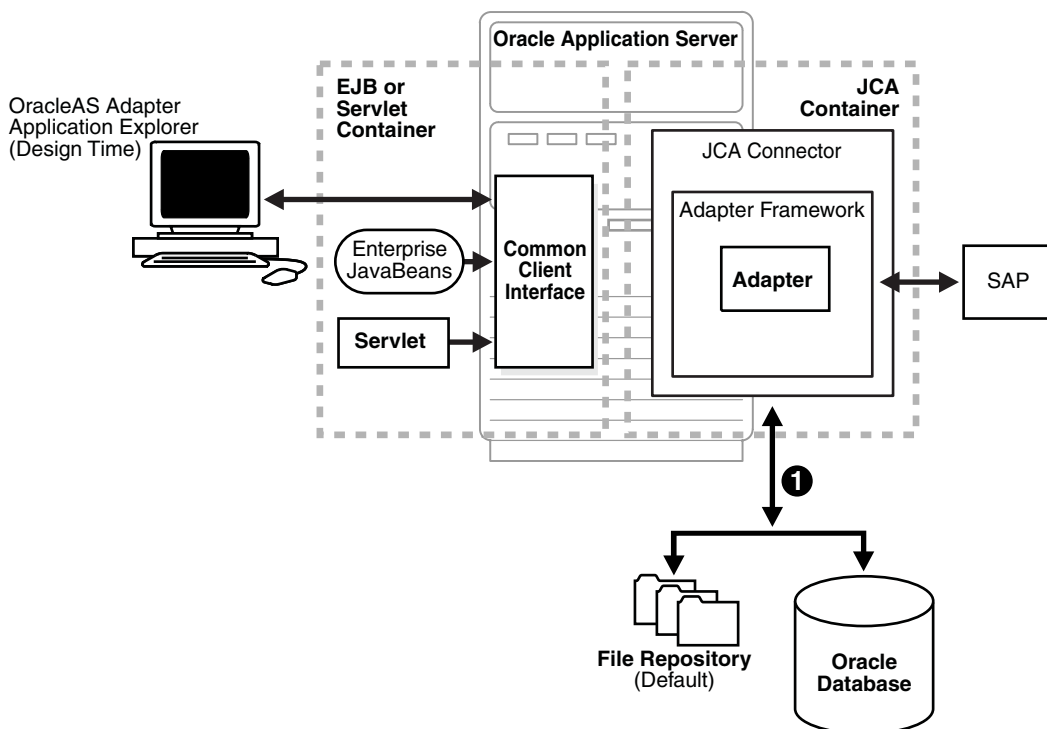
関連項目：

- *Oracle WebLogic Server Adapter 概要*

5.2 アダプタのデプロイメント

図 5-1 は、Oracle Application Server への J2CA コネクタのデプロイメントを示しています。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、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 概要

5.3 アダプタ構成の更新

この節では、次のトピック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用の 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 のデプロイメント記述子は、上記のディレクトリにあります。

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) の場合のみサポートされています。

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>IWAYJCA10</display-name>
  <vendor-name>IWAY Software</vendor-name>
  <spec-version>1.0</spec-version>
  <eis-type>IWAY</eis-type>
  <version>1.0</version>
  <license>
    <license-required>>false</license-required>
  </license>
  <resourceadapter>

  <managedconnectionfactory-class>com.ibi.afjca.spi.IWAYManagedConnectionFactory</ma
nagedconnectionfactory-class>

  <connectionfactory-interface>javax.resource.cci.ConnectionFactory</connectionfacto
ry-interface>

  <connectionfactory-impl-class>com.ibi.afjca.cci.IWAYConnectionFactory</connectionf
actory-impl-class>
    <connection-interface>javax.resource.cci.Connection</connection-interface>

  <connection-impl-class>com.ibi.afjca.cci.IWAYConnection</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-i
nterface>
  </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 がパッケージされているアプリケーション・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・エクスプローラに事前構成された 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する URL。これは、Oracle データベースをリポジトリとして使用する場合にのみ必要です。
IWayRepoUser	データベースに接続するときに使用するユーザー名。これは、Oracle データベースをリポジトリとして使用する場合にのみ必要です。
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 データベースをリポジトリとして使用する場合にのみ必要です。
loglevel	ManagedConnectionFactory プロパティによって設定されるレベルが上書きされます。

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_SOA1\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できます。また、J2CA 構成 (jca_sample2 など) は、アプリケーション・エクスペ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用の WSDL ファイルの変更

アプリケーション・エクスペローラでは、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。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 の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"/>

```

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・コンポーネントの統合

Oracle Application Adapter for SAP R/3 は、Business Process Execution Language (BPEL) Process Manager にシームレスに統合されており、Web サービスの統合を可能にします。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。Web Service Definition Language (WSDL) ドキュメントとして公開されるアダプタ・サービスを使用します。

この章のセクションは以下のとおりです。

- [6.1 項「概要」](#)
- [6.2 項「アダプタのデプロイメント」](#)
- [6.3 項「新規のアプリケーション・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 イベントの作業 \(J2CA 構成\)」](#)
- [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。また、基になるアダプタ・サービスは、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) の両方のサービス用に Oracle Adapter アプリケーション・エクスプローラ (アプリケーション・エクスプローラ) でデザインタイム中に生成される WSDL ファイルとして公開される必要があります。詳細については、4-18 ページ「[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・サービスにそれぞれ適切な BPEL プロセスの設計に使用されます。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後すぐに、新規構築されたプロセスはすべて自動的に Oracle Enterprise Manager Console にデプロイされ、BPEL プロセスの実行、監視、管理、およびアダプタ・イベントのリスニングを行うことができます。

6.2 アダプタのデプロイメント

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

6.3 新規のアプリケーション・サーバー接続の構成

Oracle JDeveloper で新規のアプリケーション・サーバー接続を構成する手順は、次のとおりです。

1. システムで **Oracle JDeveloper** を開きます。
2. 図 6-1 に示すように、メニュー・バーで、「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

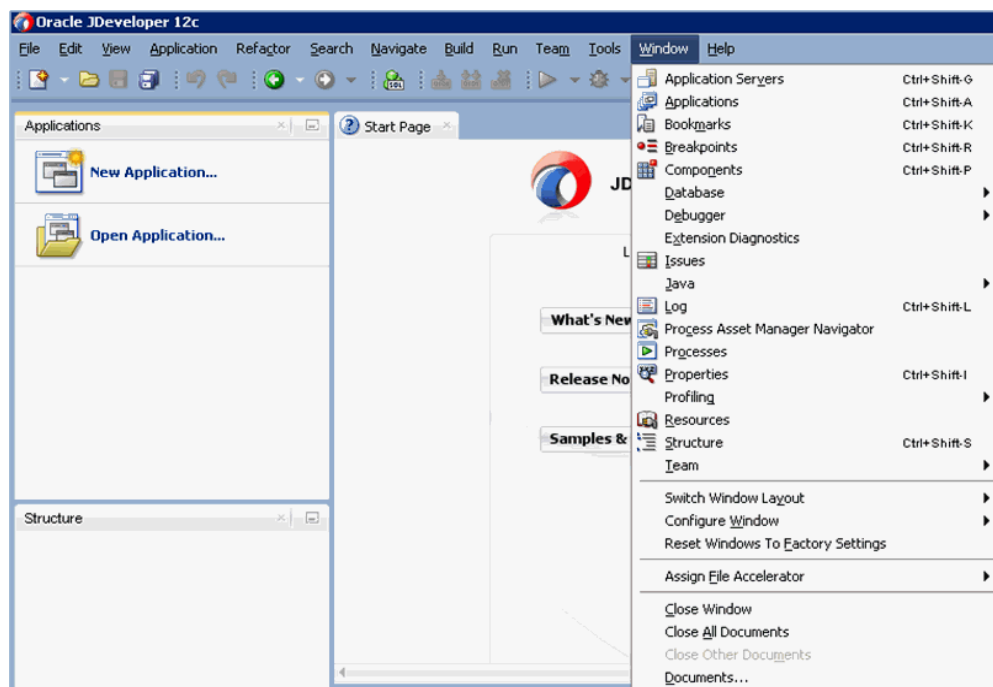
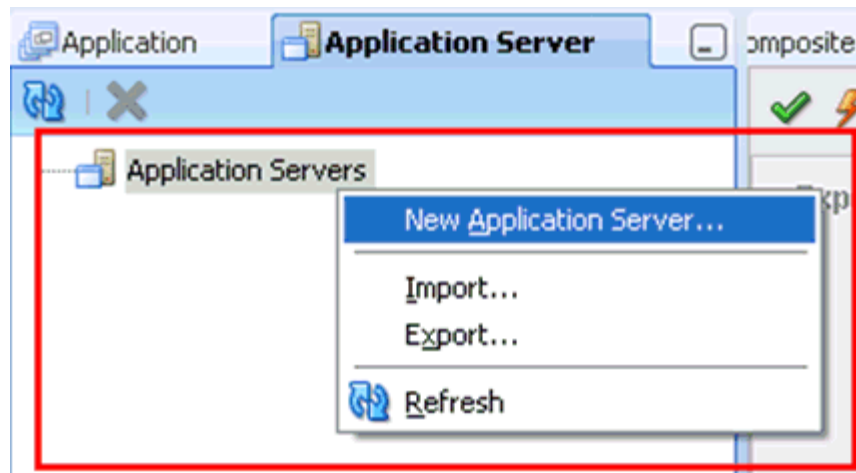


図 6-2 に示すように、「アプリケーション・サーバー」タブが表示されます。

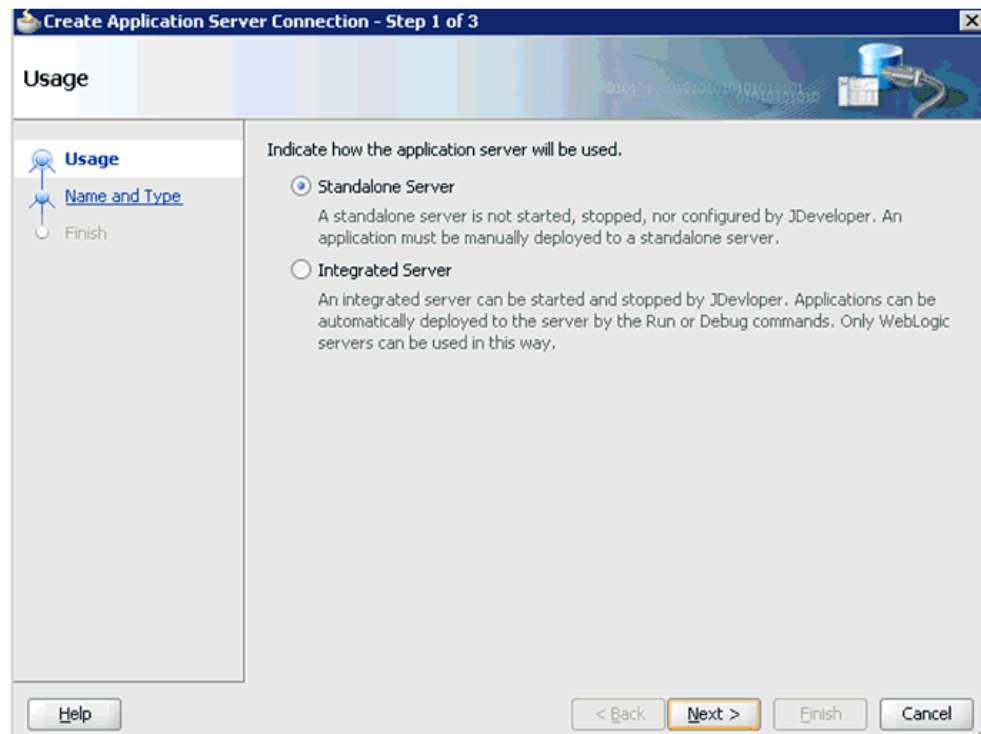
図 6-2 「アプリケーション・サーバー」タブ



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。

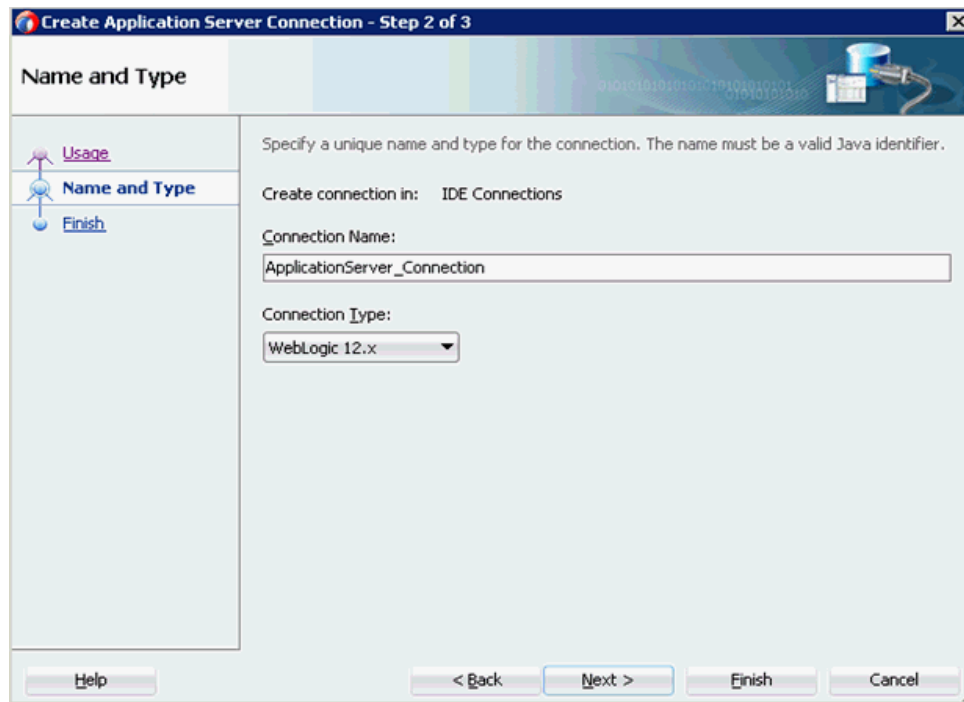
図 6-3 アプリケーション・サーバー接続の作成ウィザード



4. デフォルトの選択（「スタンドアロン・サーバー」）を受け入れ、「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。

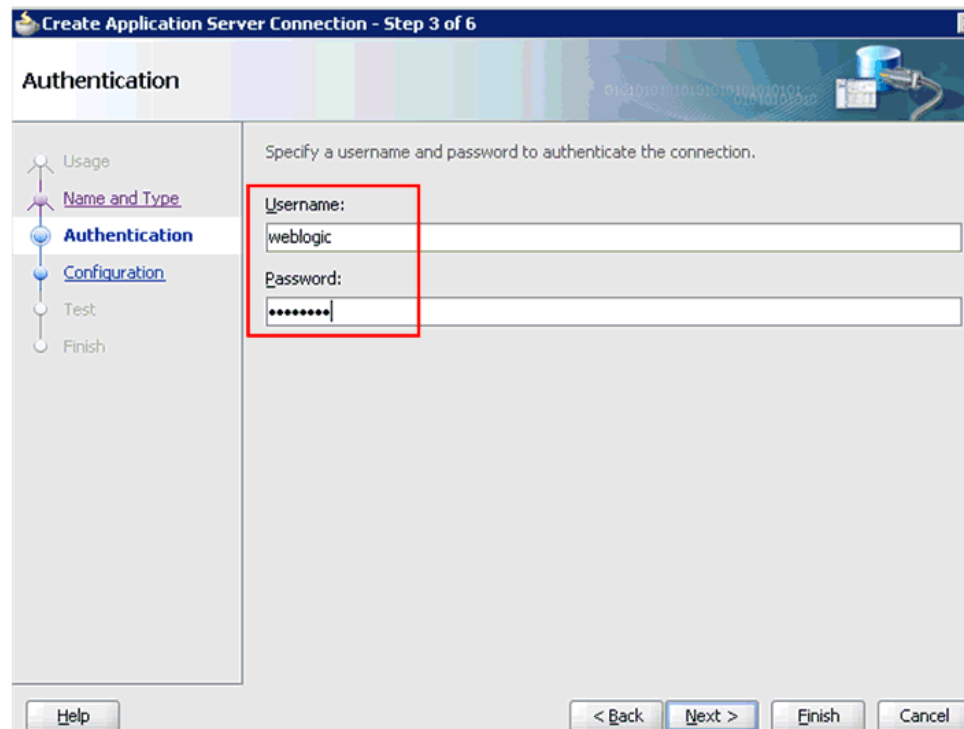
図 6-4 「名前とタイプ」 ページ



5. アプリケーション・サーバー接続の新規の名前を指定し、「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。

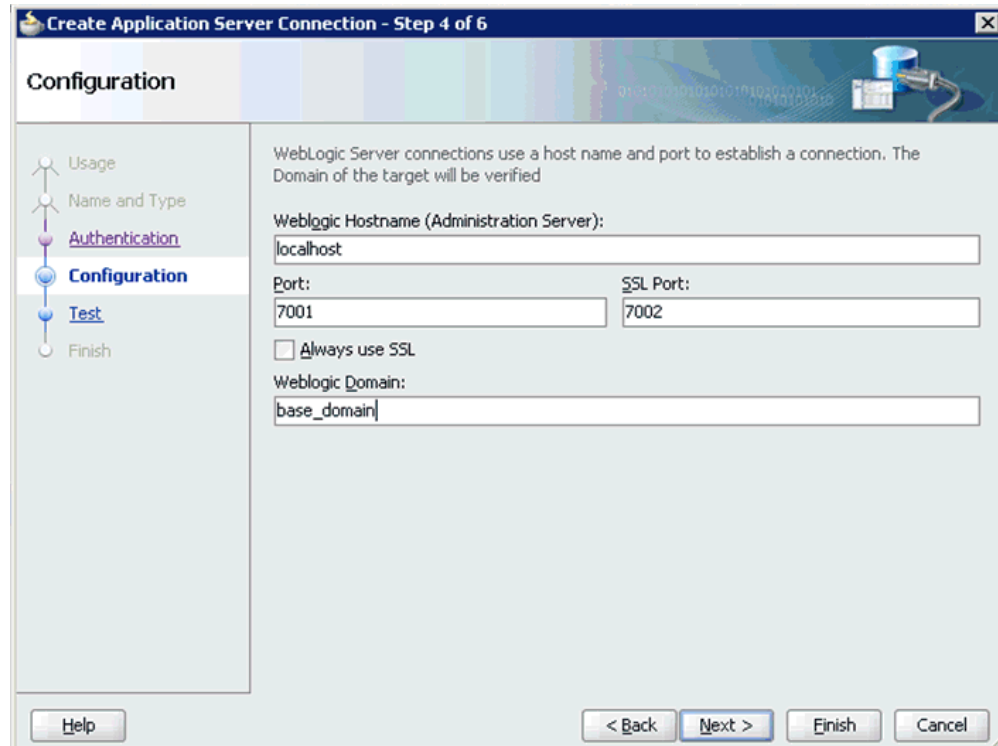
図 6-5 「認証」 ページ



- 新規の接続用のユーザー名 (「weblogic」など) およびパスワード (「welcome1」など) を指定します。
- 「次へ」をクリックします。

図 6-6 に示すように、「構成」ページが表示されます。

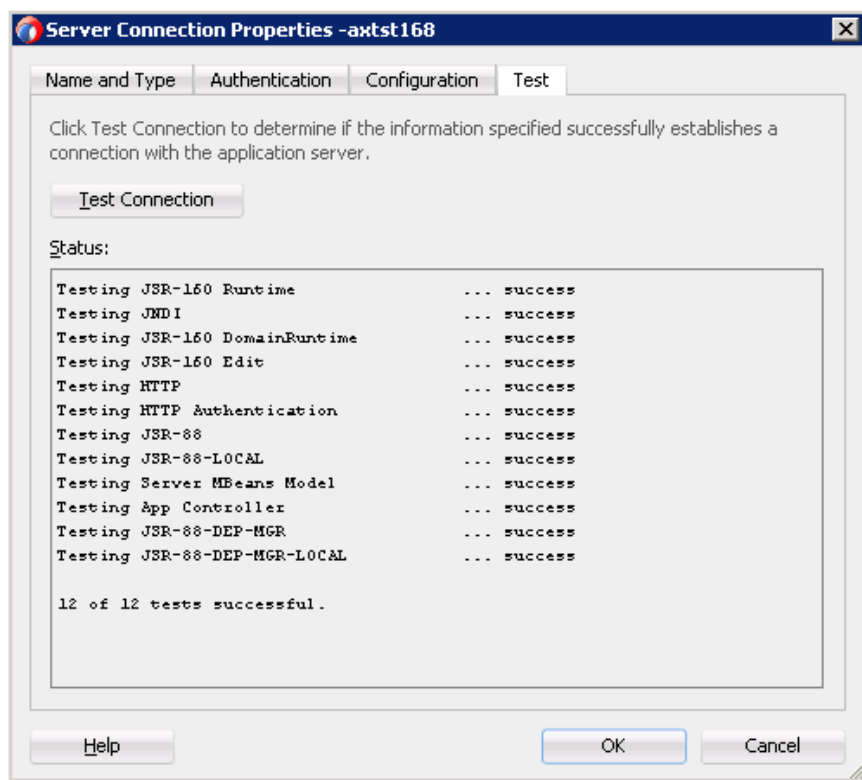
図 6-6 「構成」ページ



- Oracle WebLogic のホストの名前を指定します (「localhost」など)。これは、プロセスのデプロイが必要なシステムの IP および Oracle WebLogic のドメインです (「base_domain」など)。
- 「次へ」をクリックします。

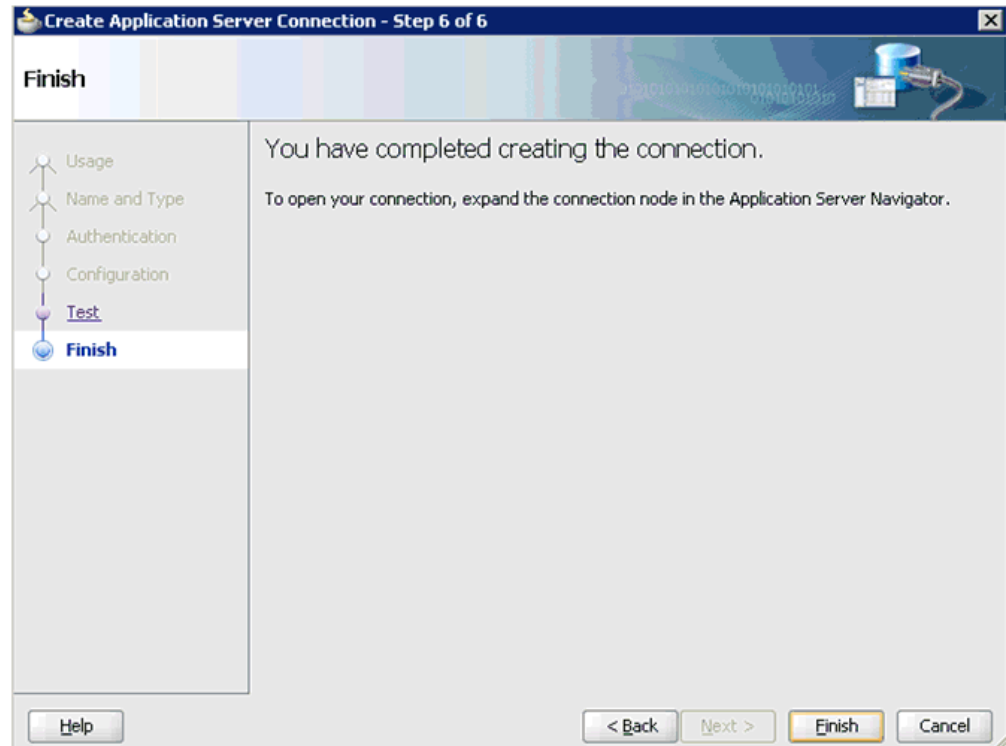
図 6-7 に示すように、「テスト」ページが表示されます。

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. テストのステータスが成功であ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。

新規のアプリケーション・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。

Application Adapters インストールの次のフォルダに、このアウトバウンドの使用ケース・シナリオのサンプル・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
```

アウトバウンドの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 アプリケーション・エクスプローラ (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPEL デザイナ (JDeveloper)

注意： この章に記載されている例では JDeveloper が使用されています。

この節では、次のトピックについて説明します。

- [6.4.1 項「リクエスト/レスポンス型のサービス用の 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 ドキュメントの起動」
- 6.4.6 項「アウトバウンド BPEL およびアウトバウンド・メディエータ・プロセスのテスト」

6.4.1 リクエスト/レスポンス型のサービス用の 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。

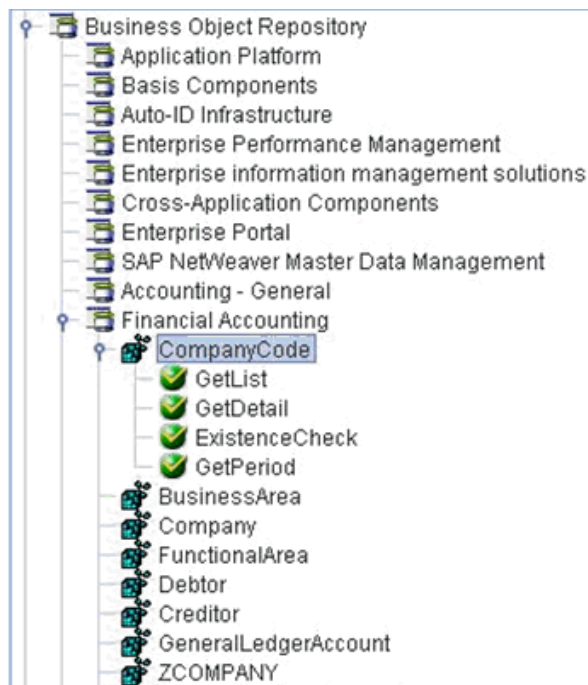
次の手順を実行して、リクエスト/レスポンス型のサービス用の WSDL を生成します。

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とターゲットの定義、および SAP R/3 への接続の詳細は、4-4 ページ 4.2 項「アプリケーション・エクスプローラの起動」および 4-8 ページ 4.5.1 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、「ビジネス・オブジェクト・リポジトリ」、**Financial Accounting**、**Company Code Business Object** の順に展開し、「GetDetail」を選択します。

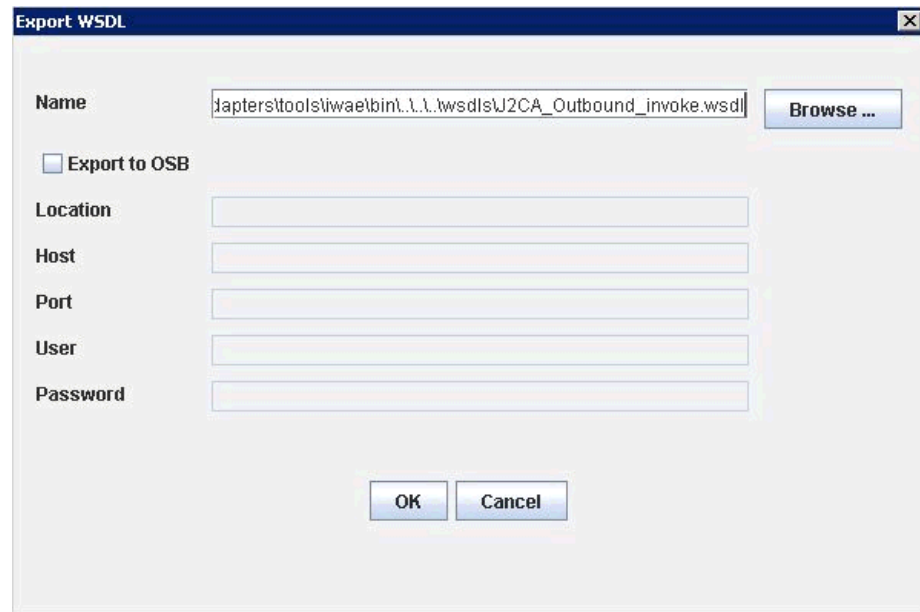
図 6-9 接続された MySAP_Target



4. 「GetDetail」ノードを右クリックします。
メニューが表示されます。

5. アウトバウンド JCA サービスの作成 (リクエスト/レスポンス) をクリックします。
 図 6-10 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 6-10 「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 での BPEL アウトバウンド・プロセスの定義に必要な最初のステップである、SOA 用の空のコンポジットを作成する準備ができました。

6.4.2 SOA 用の空のコンポジットの作成

SOA 用に空のコンポジットを作成するには、以下の手順を実行します。

1. 新規の SOA アプリケーションを作成します。
2. 図 6-11 に示すように、新規の SOA アプリケーションに名前を入力し、「次へ」をクリックします。

図 6-11 「アプリケーションの名前付け」ページ

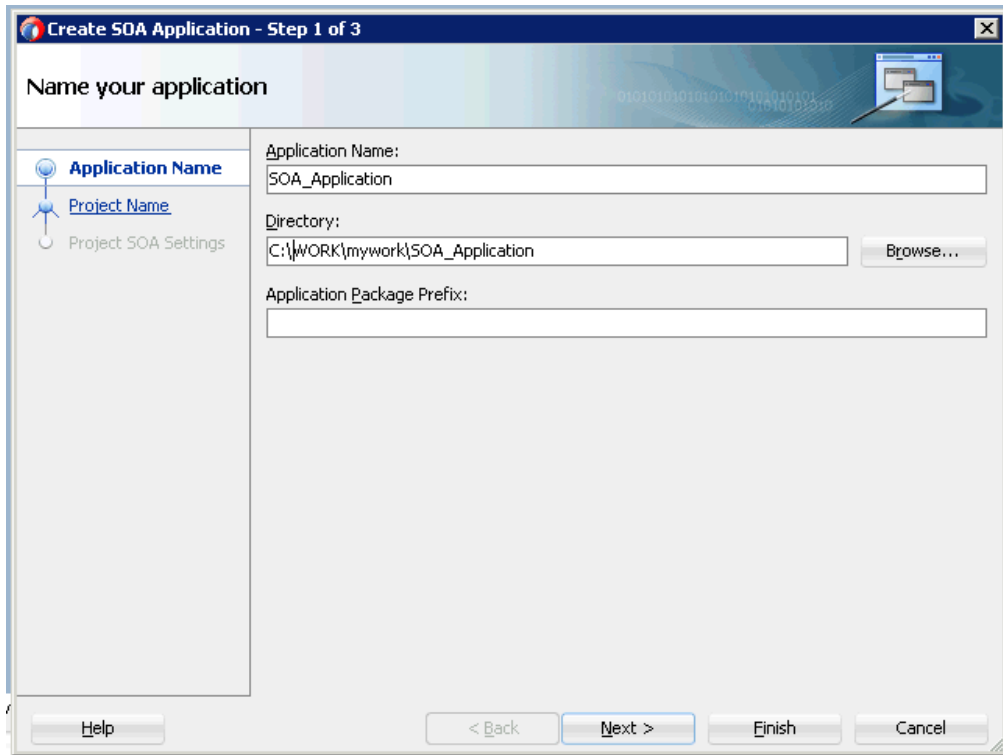
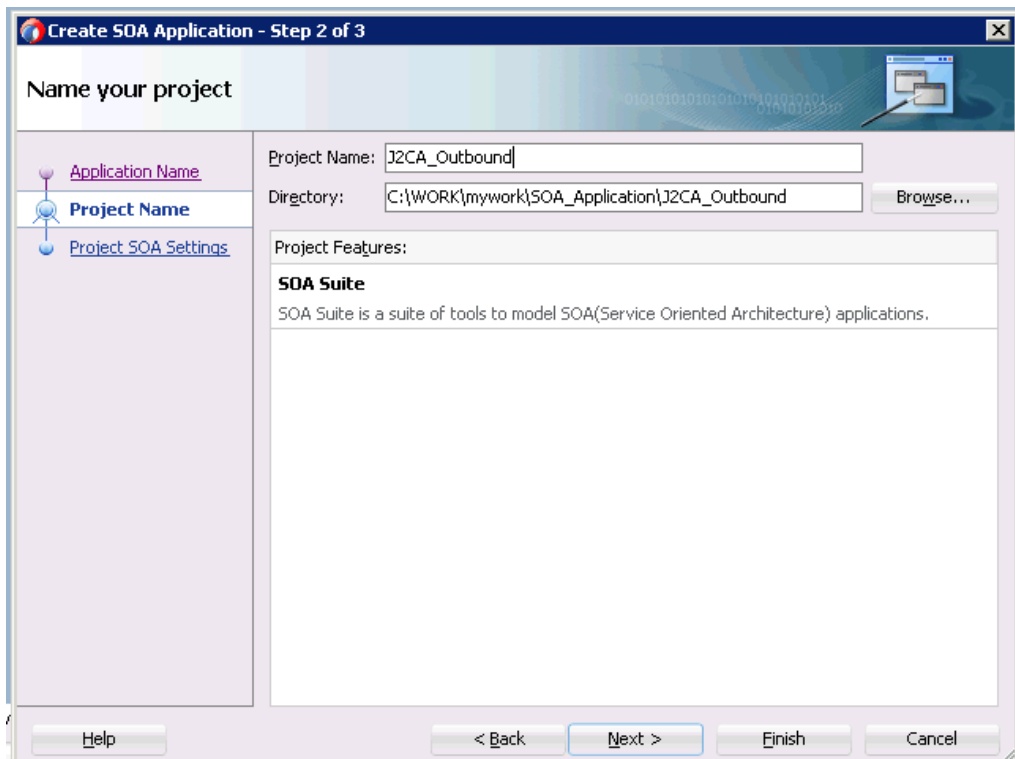


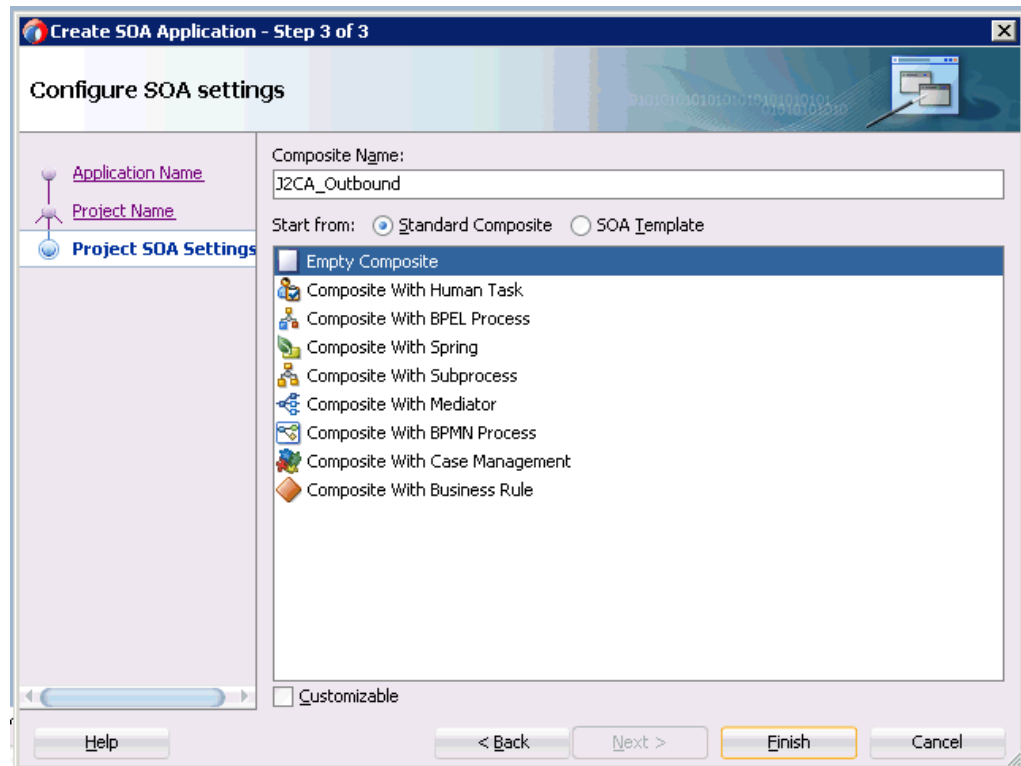
図 6-12 に示すように、「プロジェクトの名前付け」ページが表示されます。

図 6-12 「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て、「次へ」をクリックします。
図 6-13 に示すように、SOA 設定の構成ページが表示されます。

図 6-13 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、次のトピックで構成される BPEL アウトバウンド・プロセスの定義方法について説明します。

- 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 との既知のデプロイメント問題の調整」

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、次のステップを実行します。

- 図 6-14 に示すように、「サービス・アダプタ」ペインから「外部参照」ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・アンド・ドロップします。

図 6-14 「サード・パーティ・アダプタ」コンポーネント

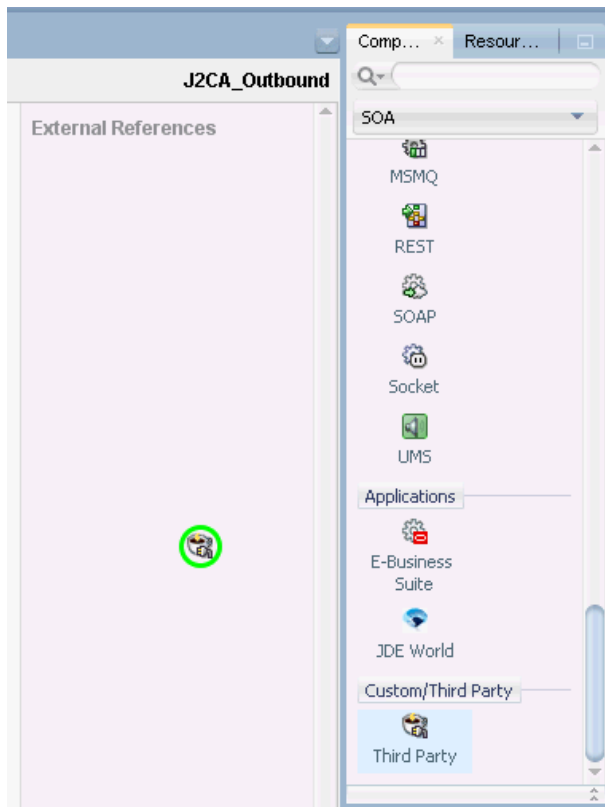
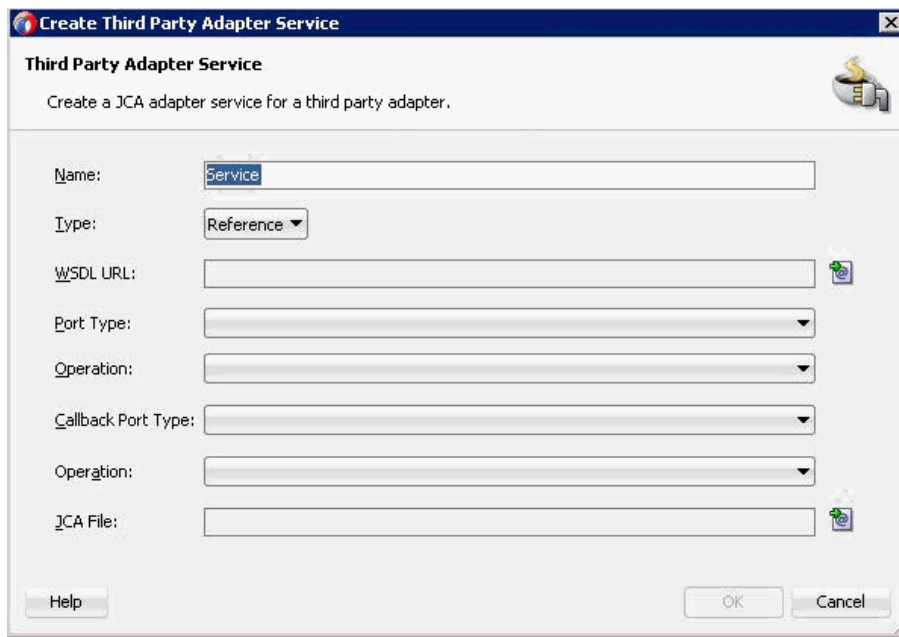


図 6-15 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

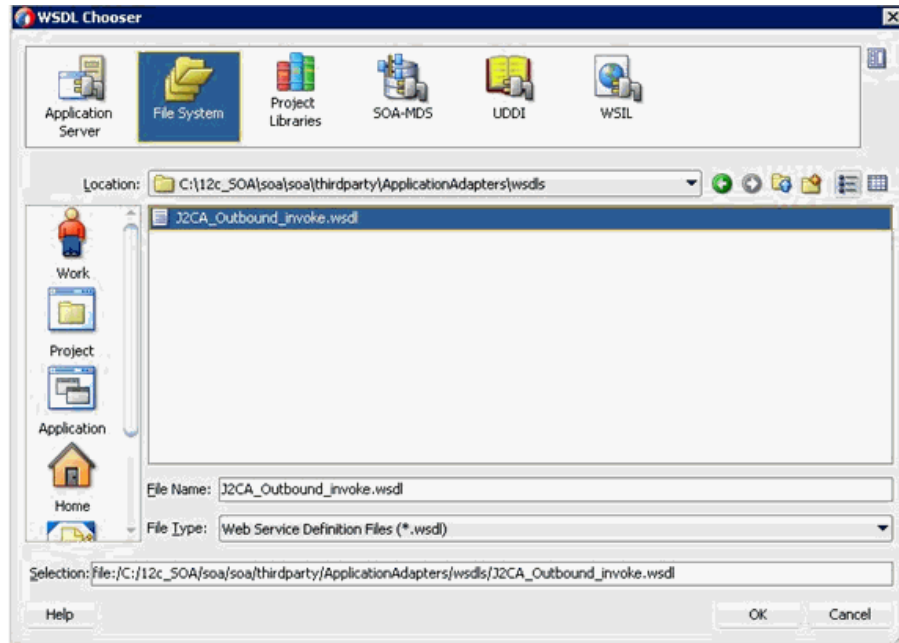
図 6-15 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



2. サード・パーティ・アダプタ・サービスの名前を入力するか、デフォルト名をそのまま使用します。

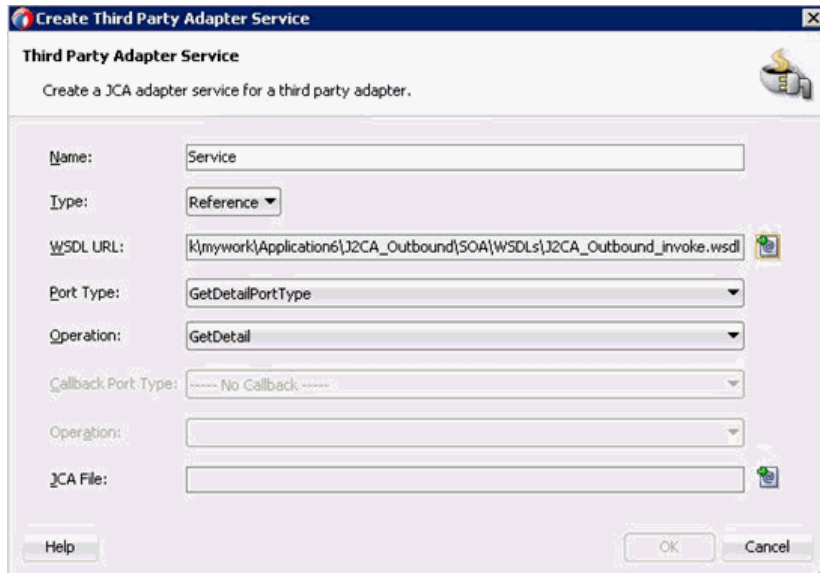
3. 「タイプ」リストで「参照」が選択されていることを確認します (デフォルト)。
4. 「WSDL URL」フィールドの右の**既存の WSDL の検索**アイコンをクリックします。
 図 6-16 に示すように、「WSDL チューザ」ダイアログが表示されます。

図 6-16 「WSDL チューザ」ダイアログ



5. 次のディレクトリまでブラウズし、アウトバウンド WSDL ファイルを選択します。
 <ADAPTER_HOME>\wsdl
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。
 アウトバウンド WSDL ファイルおよび関連リクエスト/レスポンス XML スキーマ・ファイル (.xsd) が、作成済のプロジェクト・フォルダにインポートされます。
 図 6-17 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。

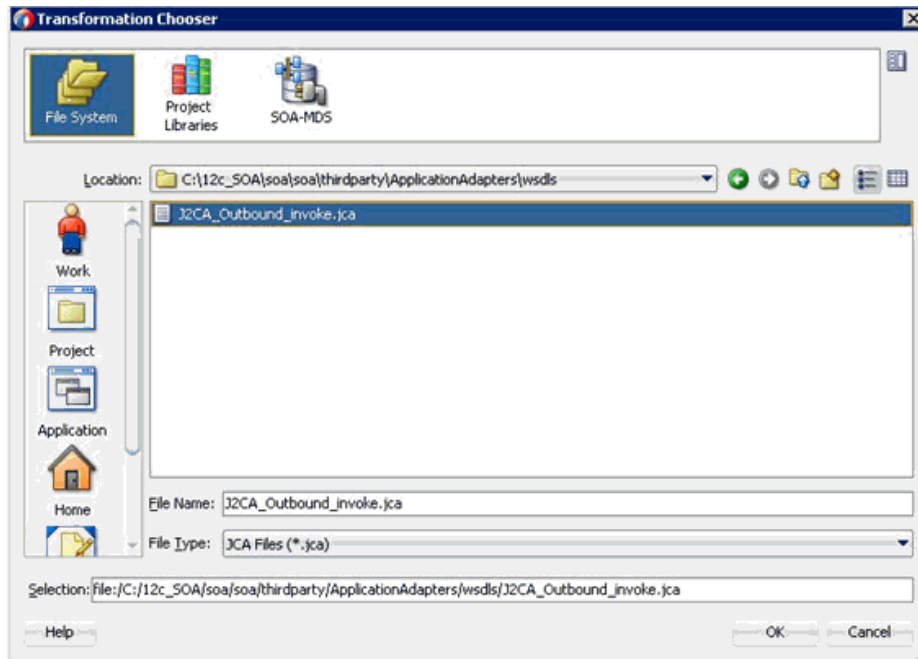
図 6-17 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



8. 「JCA ファイル」フィールドの右の JCA ファイルの検索アイコンをクリックします。

図 6-18 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 6-18 トランスフォーメーション・チューザ・ダイアログ



9. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。
<ADAPTER_HOME>\wsdl\

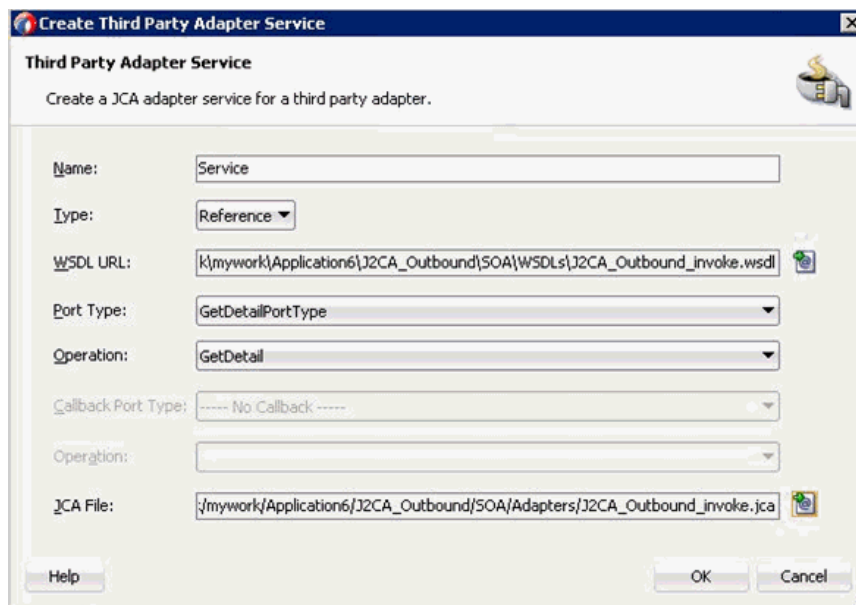
10. 「OK」をクリックします。

「ファイルのコピー」メッセージが表示されます。

11. 「はい」 をクリックします。

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

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**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。

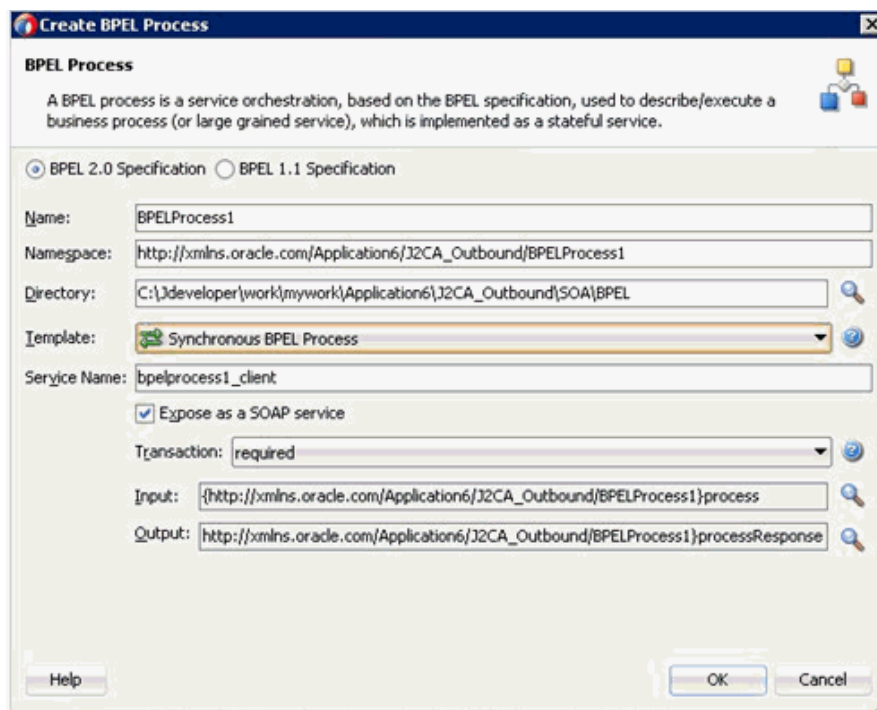
6.4.3.2 アウトバウンド BPEL プロセス・コンポーネントの構成

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、次のステップを実行します。

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

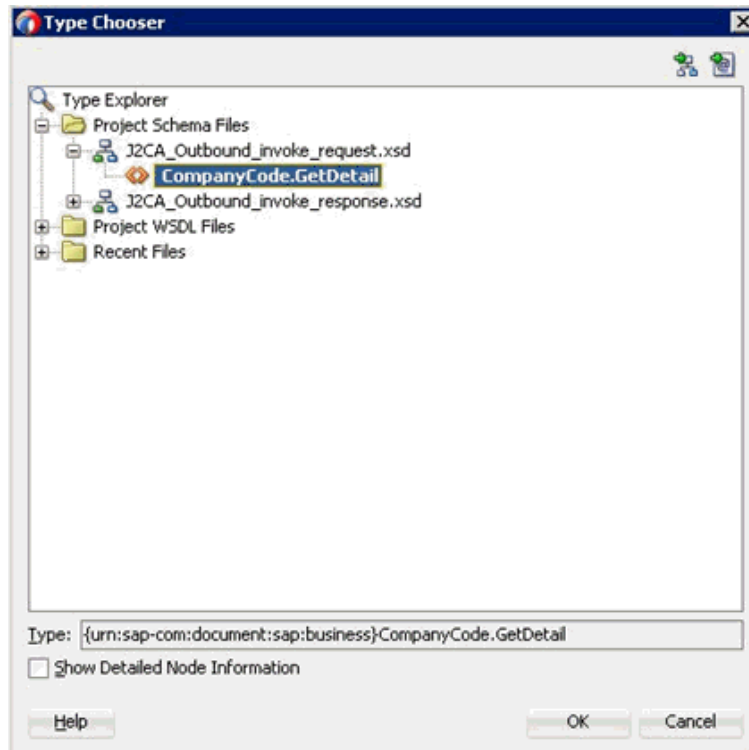
図 6-20 に示すように、「BPEL プロセスの作成」ダイアログが表示されます。

図 6-20 「BPEL プロセスの作成」 ダイアログ



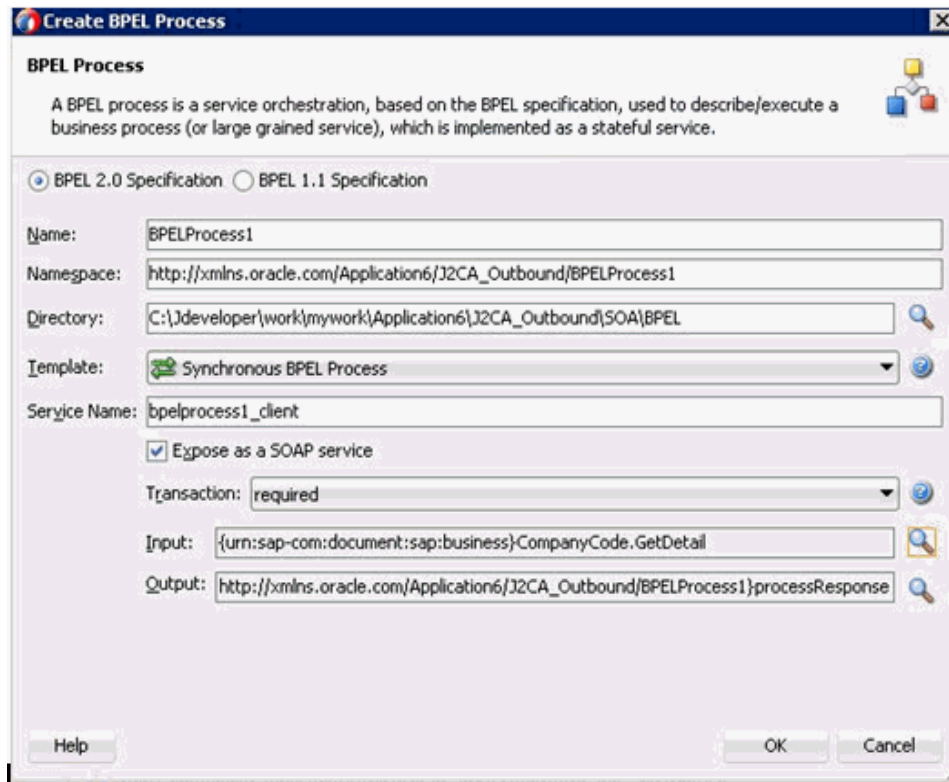
2. 「名前」フィールドで、新規のアウトバウンド BPEL プロセス・コンポーネントの名前を入力するか、デフォルト名をそのまま使用します。
デフォルトで、「BPEL 2.0 仕様」オプションが選択されています。
3. 「テンプレート」リストから、「同期 BPEL プロセス」を選択します。
4. 関連する XML リクエスト・スキーマ・ファイルの選択用の「入力」フィールドの右の「参照」アイコンをクリックします。
図 6-21 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 6-21 「タイプ・チューザ」ダイアログ



5. 「プロジェクトのスキーマ・ファイル」、「J2CA_Outbound_invoke_request.xsd」の順に展開し、「CompanyCode.GetDetail」を選択します。
6. 「OK」をクリックします。
図 6-22 に示すように、「BPEL プロセスの作成」ダイアログに戻ります。

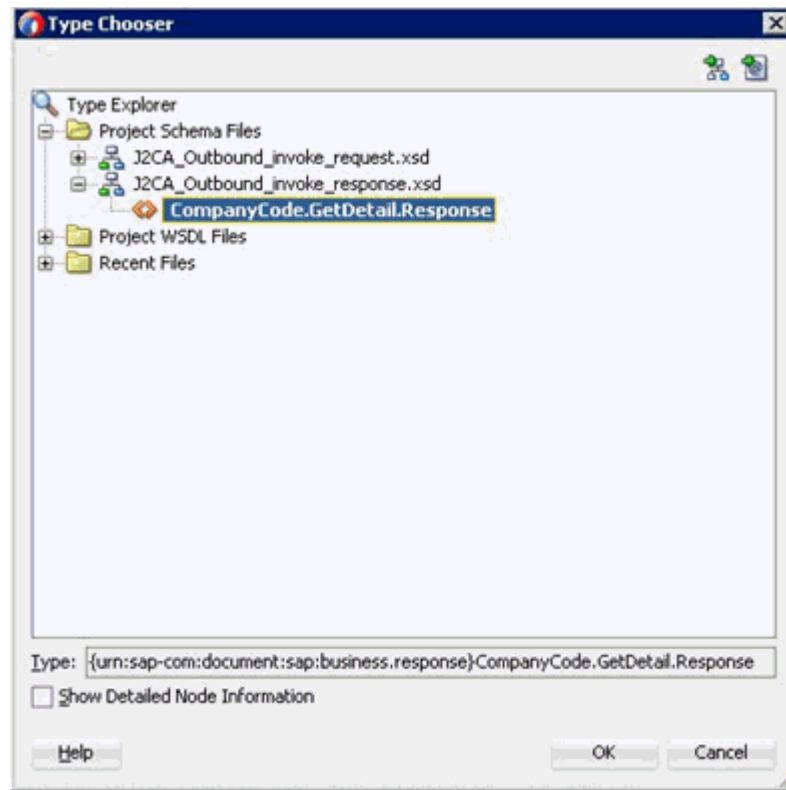
図 6-22 「BPEL プロセスの作成」 ダイアログ



7. 関連する XML レスポンス・スキーマ・ファイルの選択用の「出力」フィールドの右の「参照」アイコンをクリックします。

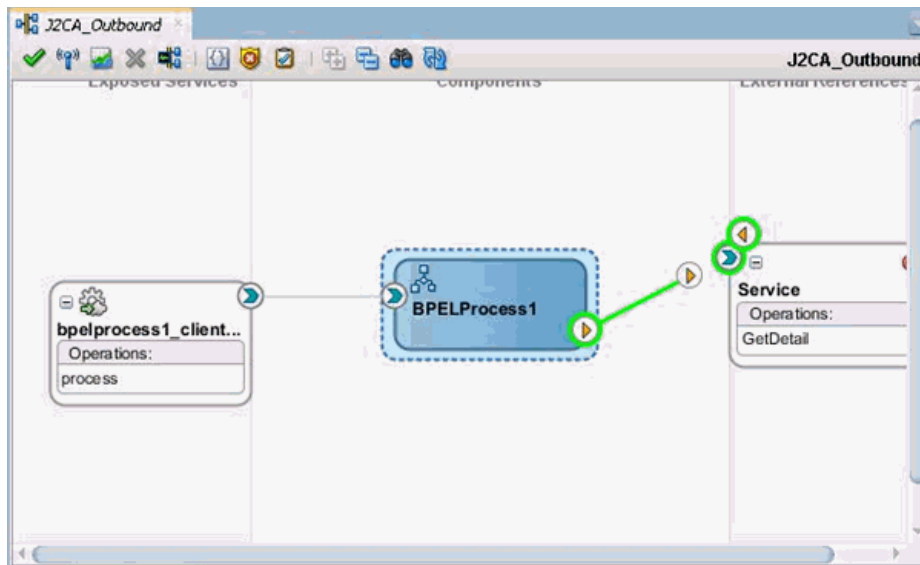
図 6-23 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 6-23 「タイプ・チューザ」ダイアログ



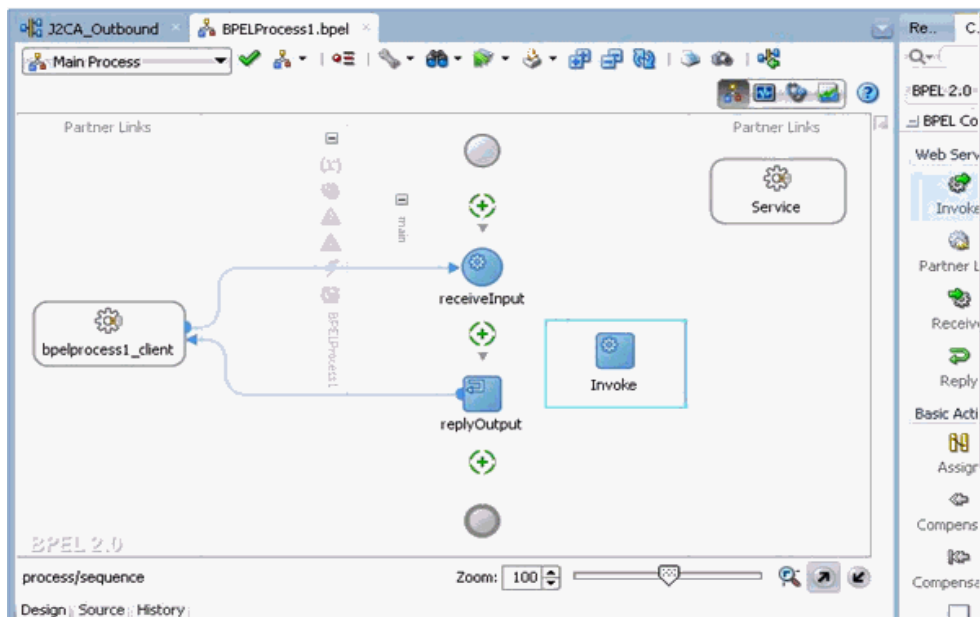
8. 「プロジェクトのスキーマ・ファイル」、「J2CA_Outbound_invoke_response.xsd」の順に展開し、「CompanyCode.GetDetailResponse」を選択します。
9. 「OK」をクリックします。
「BPEL プロセスの作成」ダイアログに戻ります。
10. 「OK」をクリックします。
11. 図 6-24 に示すように、アウトバウンド BPEL プロセス・コンポーネントとサード・パーティ・アダプタ・サービス・コンポーネント間の接続を作成します。

図 6-24 作成された接続



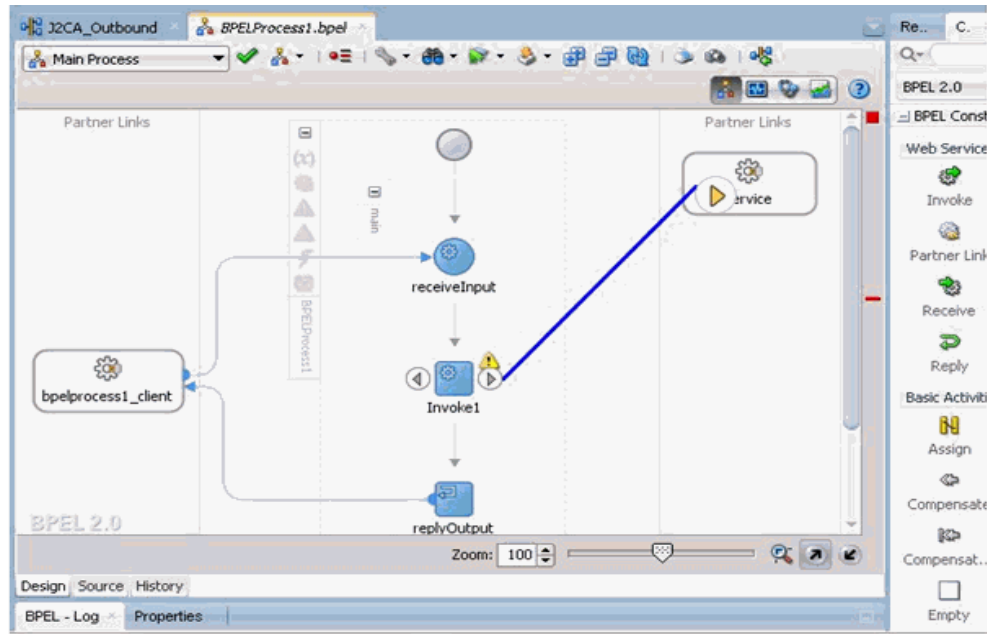
12. 「コンポーネント」 ペインで、アウトバウンド BPEL プロセス・コンポーネントをダブルクリックします。
13. 図 6-25 に示すように、「BPEL コンストラクト」- 「Web サービス」 の下の「起動」アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップし、「receiveInput」アクティビティ・コンポーネントと「replyOutput」アクティビティ・コンポーネントの間に置きます。

図 6-25 「起動」アクティビティ・コンポーネント



14. 図 6-26 に示すように、新規の起動アクティビティ・コンポーネント (Invoke1) とサード・パーティ・アダプタ・サービス・コンポーネントの間に接続を作成します。

図 6-26 作成された接続



起動の編集ダイアログが表示されます。

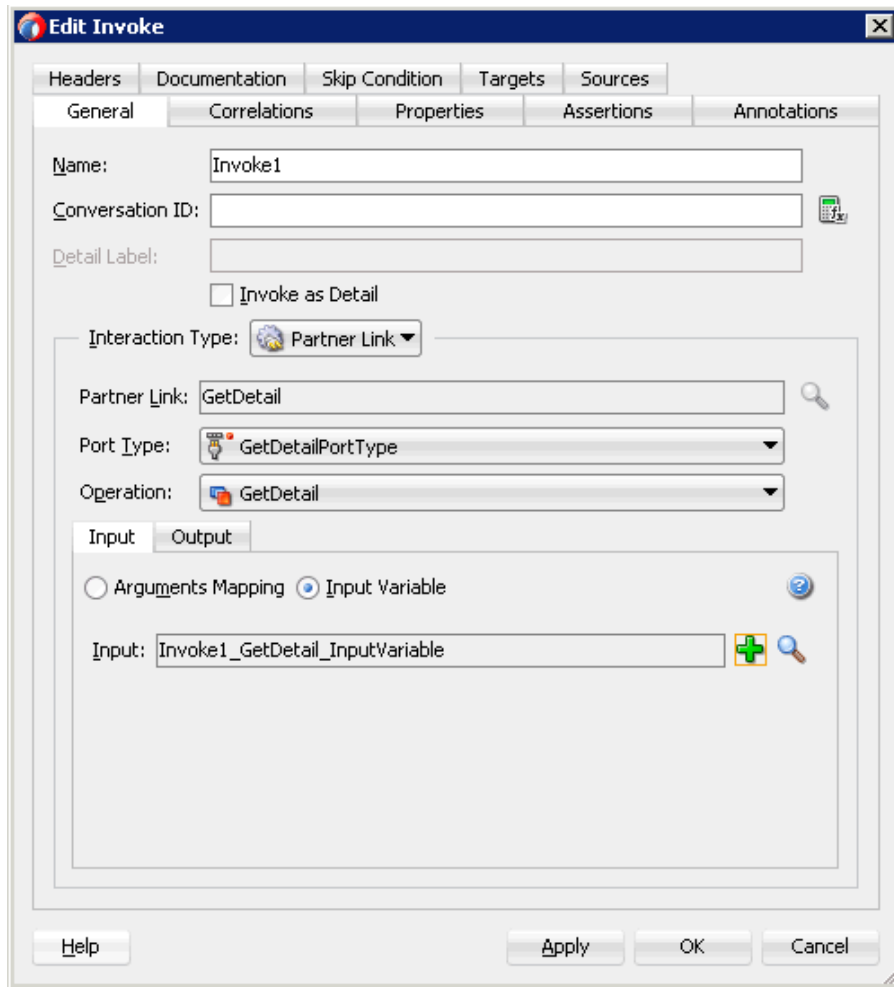
15. 新規入力変数の構成用の「入力」フィールドの右の**プラス記号**アイコンをクリックします。

「変数の作成」ダイアログが表示されます。

16. 新規の入力変数に示されたデフォルト値を受け入れ、「OK」をクリックします。

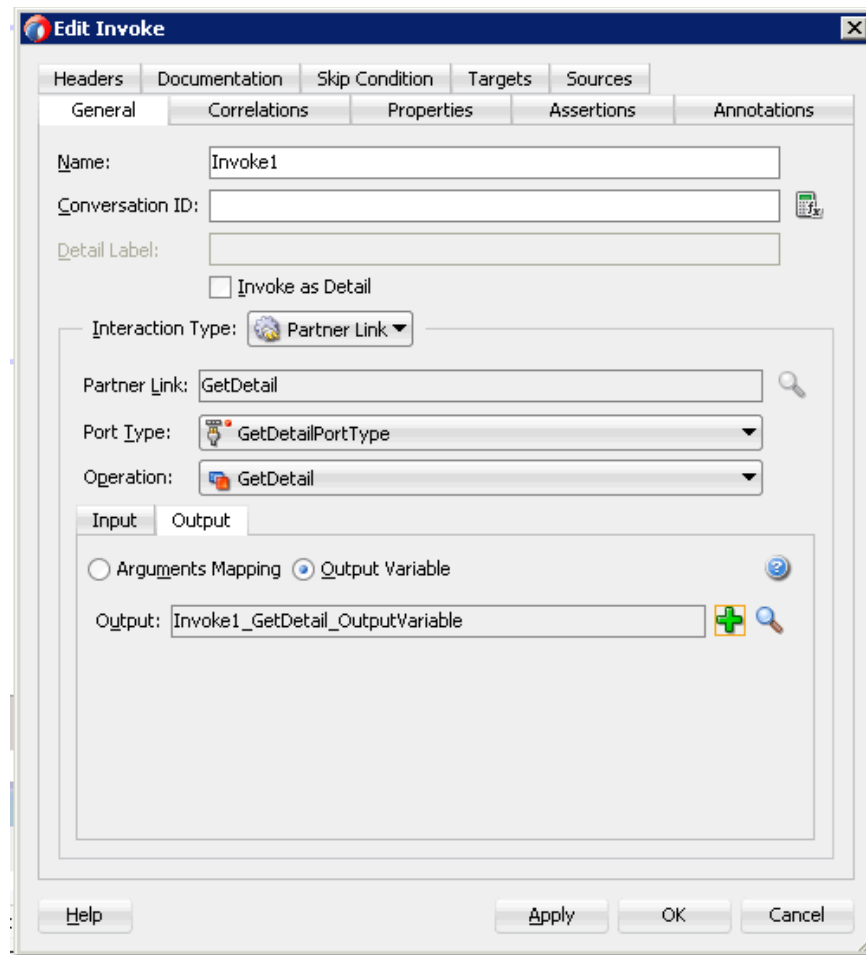
図 6-27 に示すように、起動の編集ダイアログに戻ります。

図 6-27 起動の編集ダイアログ



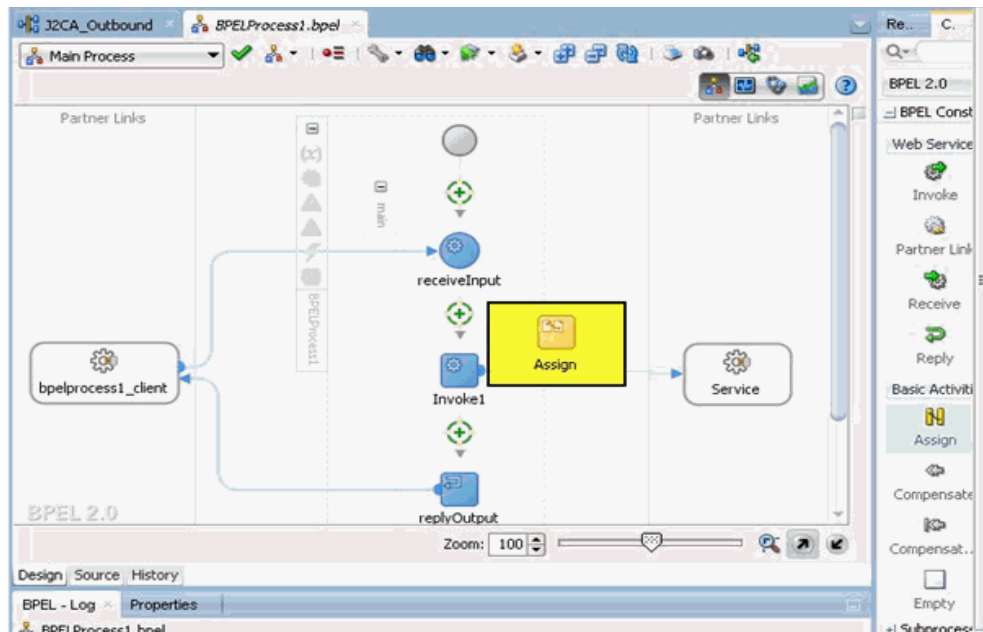
17. 「出力」タブを選択し、新規出力変数の構成用の「出力」フィールドの右のプラス記号アイコンをクリックします。
「変数の作成」ダイアログが表示されます。
18. 新規の出力変数に示されたデフォルト値を受け入れ、「OK」をクリックします。
図 6-28 に示すように、起動の編集ダイアログに戻ります。

図 6-28 起動の編集ダイアログ



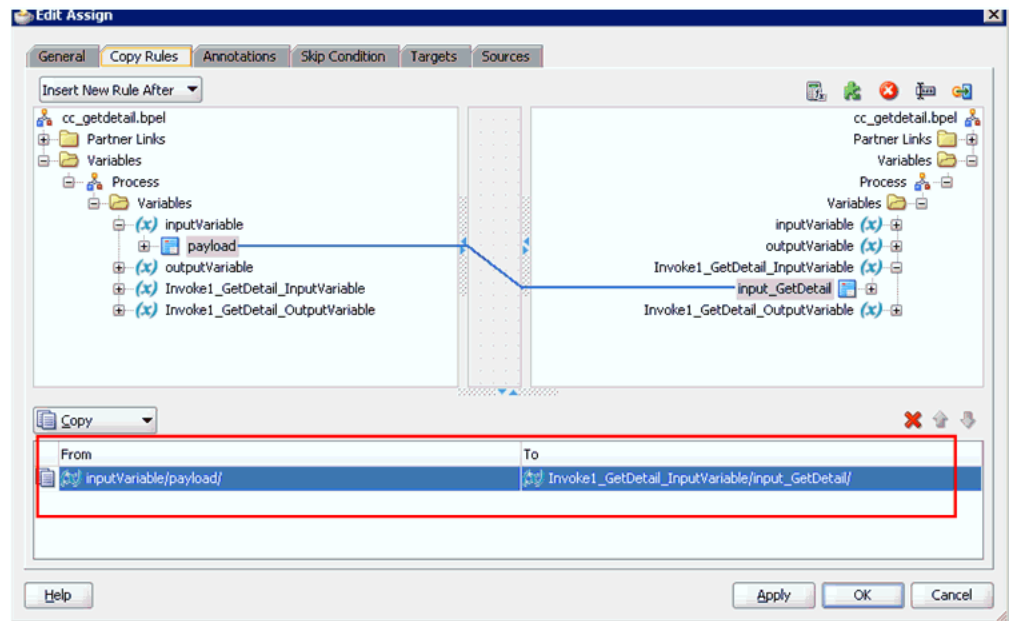
19. 「適用」、「OK」の順にクリックします。
20. 図 6-29 に示すように、「BPEL コンストラクト」- 「基本アクティビティ」コンポーネントの下の「割当て」アクティビティを「コンポーネント」ペインにドラッグ・アンド・ドロップし、「受信」アクティビティ・コンポーネント (receiveInput) と「起動」アクティビティ・コンポーネント (Invoke1) の間に置きます。

図 6-29 「割当て」 アクティビティ・コンポーネント



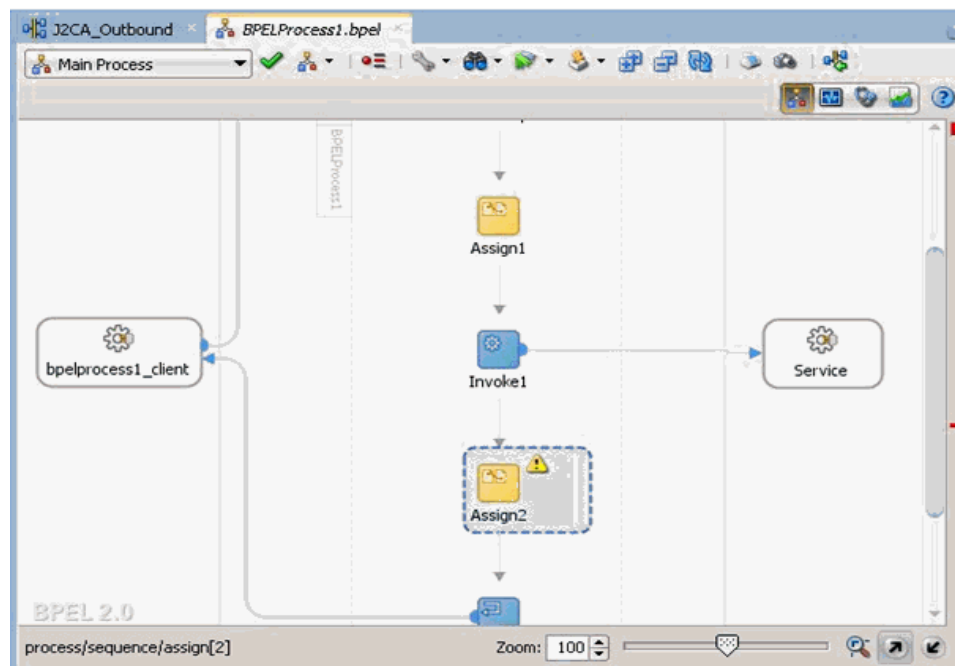
21. 新規の「割当て」アクティビティ・コンポーネント (**Assign1**) をダブルクリックします。
割当ての編集ダイアログが表示されます。
22. 左ペインで、「変数」の下の「**InputVariable**」を展開し、「**payload**」を選択します。
23. 右ペインで、「変数」の下の「**Invoke1_GetDetail_InputVariable**」を展開し、「**input_GetDetail**」を選択します。
24. 「**payload**」変数を「**input_GetDetail**」変数にドラッグしてマップします。
図 6-30 に示すように、マップされた変数が強調表示された領域に移入されます。

図 6-30 割当ての編集ダイアログ



25. 「適用」、「OK」の順にクリックします。
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) の間に置きます。
27. 図 6-31 に示すように、新規の「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。

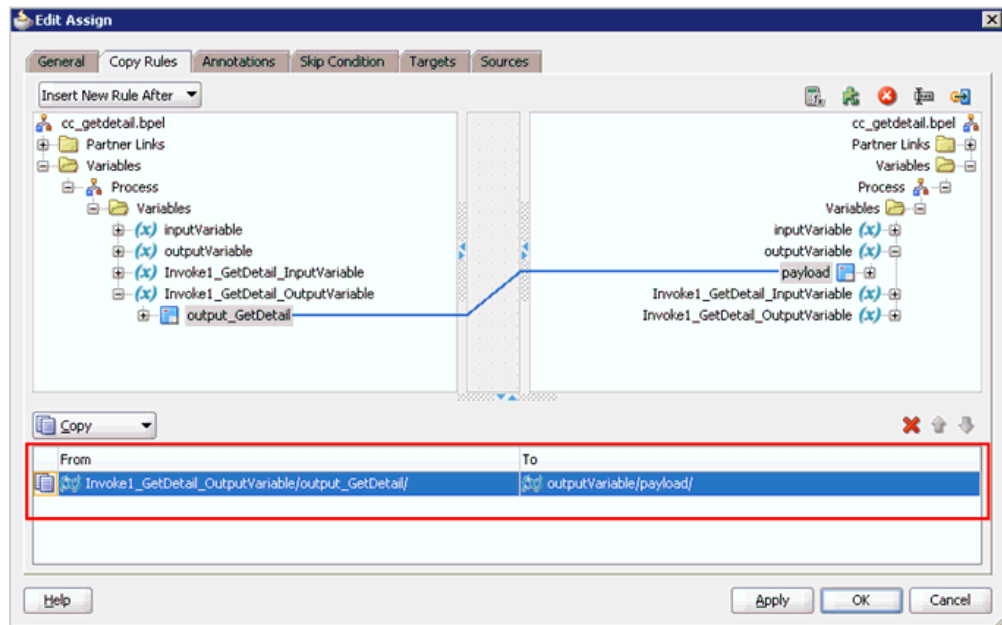
図 6-31 新規の「割当て」アクティビティ・コンポーネント



割当ての編集ダイアログが表示されます。

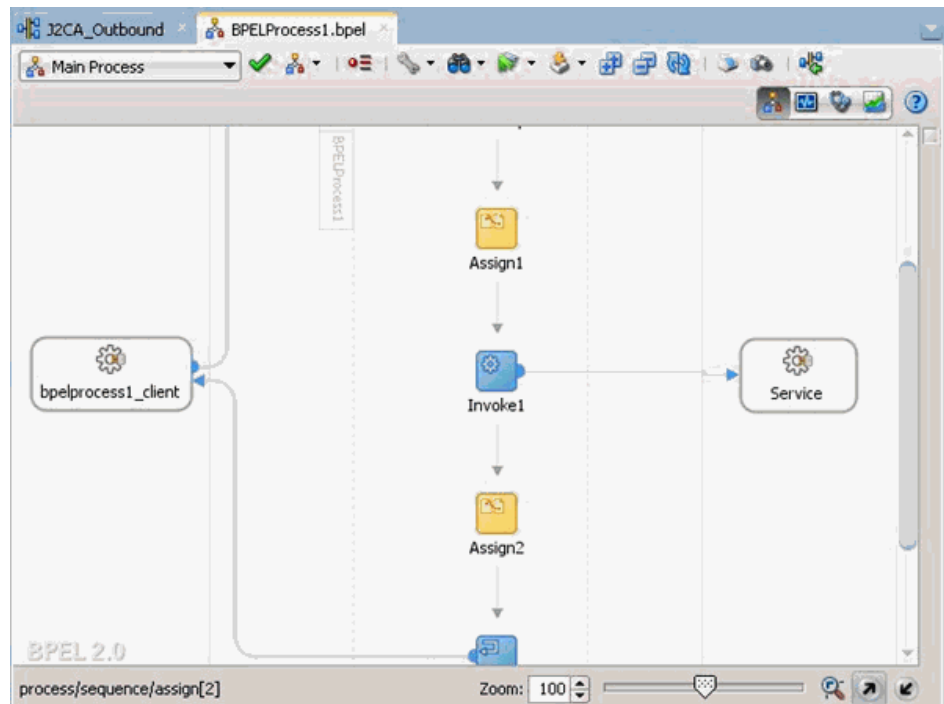
28. 左ペインで、「変数」の下の「Invoke1_GetDetail_OutputVariable」を展開し、「output_GetDetail」を選択します。
29. 右ペインで、「変数」の下の「outputVariable」を展開し、「payload」を選択します。
30. 「output_GetDetail」変数を「payload」変数にドラッグしてマップします。
☒ 6-32 に示すように、マップされた変数が強調表示された領域に移入されます。

図 6-32 割当ての編集ダイアログ



31. 「適用」、「OK」の順にクリックします。
☒ 6-33 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 6-33 アクティビティ・コンポーネント・ペイン



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

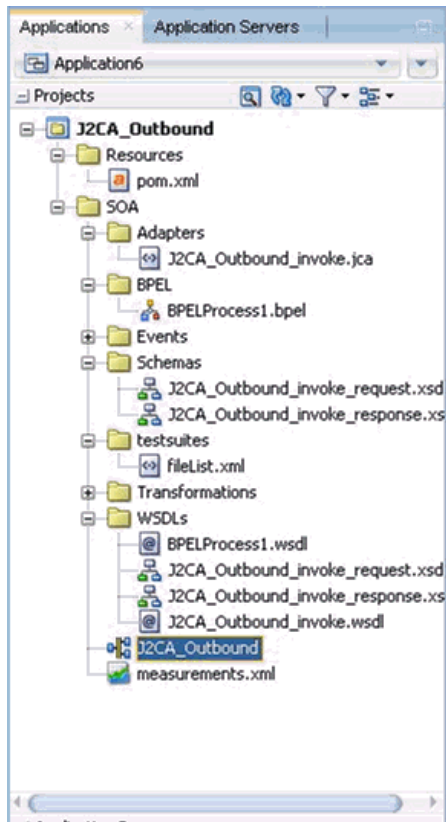
これで、BPEL アウトバウンド・プロセスをデプロイする準備ができました。

6.4.3.3 12c との既知のデプロイメント問題の調整

12c での既知のデプロイメント問題を調整するには、次のステップを実行します。

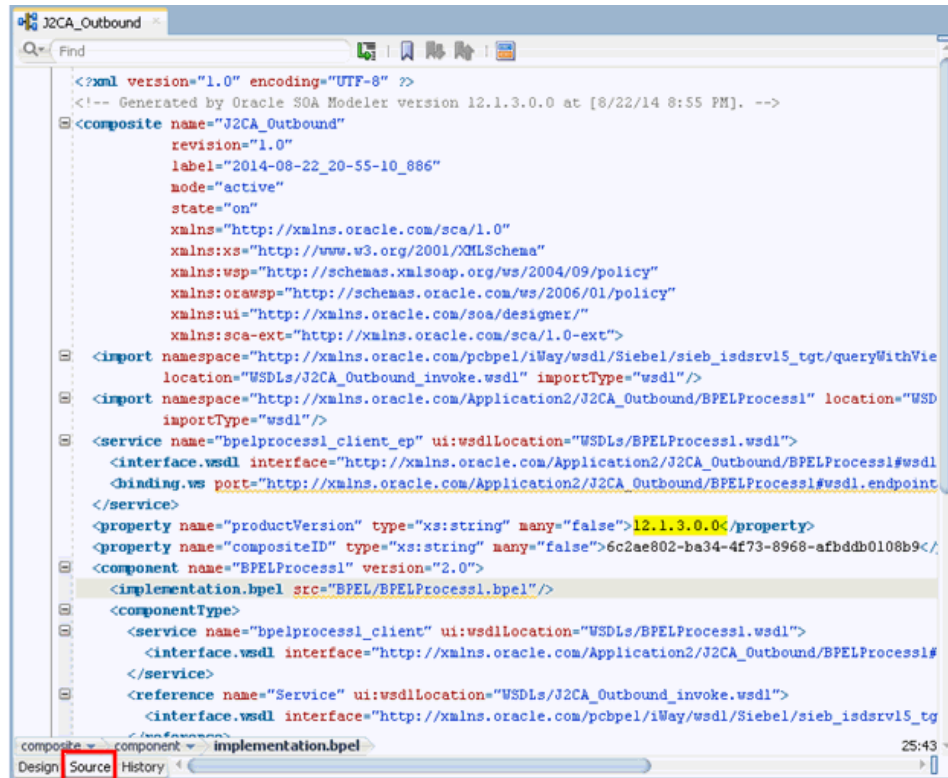
1. 図 6-34 に示すように、作成したプロセスの **J2CA_Outbound** (作成した BPEL プロセス) をダブルクリックします。

図 6-34 「J2CA_Outbound」ノード



2. 図 6-35 に示すように、開いているプロセスの下にある「ソース」タブをクリックします。

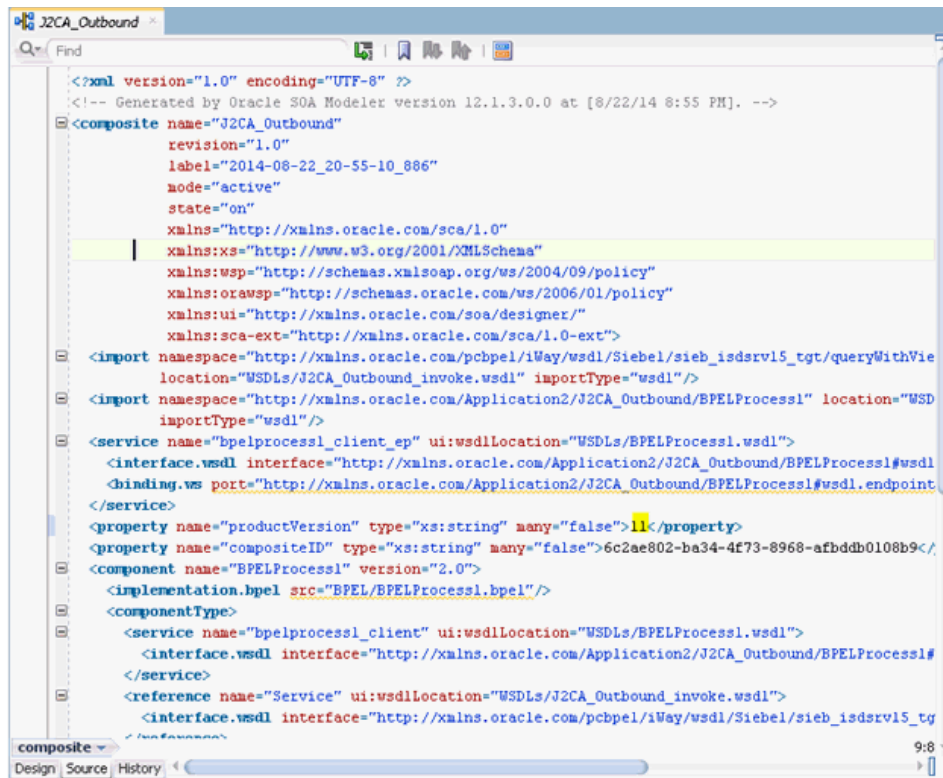
図 6-35 「ソース」タブ



```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Generated by Oracle SOA Modeler version 12.1.3.0.0 at [8/22/14 8:55 PM]. -->
<composite name="J2CA_Outbound"
  revision="1.0"
  label="2014-08-22_20-55-10_886"
  mode="active"
  state="on"
  xmlns="http://xmlns.oracle.com/sca/1.0"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
  xmlns:orawsp="http://schemas.oracle.com/ws/2006/01/policy"
  xmlns:ui="http://xmlns.oracle.com/soa/designer/"
  xmlns:sca-ext="http://xmlns.oracle.com/sca/1.0-ext">
  <import namespace="http://xmlns.oracle.com/pcbpel/iWay/wsd1/Siebel/sieb_isdsrv15_tgt/queryWithVie
    location="WSDLs/J2CA_Outbound_invoke.wsd1" importType="wsdl"/>
  <import namespace="http://xmlns.oracle.com/Application2/J2CA_Outbound/BPELProcess1" location="WSD
    importType="wsdl"/>
  <service name="bpelprocess1_client_ep" ui:wsdLocation="WSDLs/BPELProcess1.wsd1">
    <interface.wsd1 interface="http://xmlns.oracle.com/Application2/J2CA_Outbound/BPELProcess1#wsd1"
    <binding.ws port="http://xmlns.oracle.com/Application2/J2CA_Outbound/BPELProcess1#wsd1.endpoint
  </service>
  <property name="productVersion" type="xs:string" many="false">12.1.3.0.0</property>
  <property name="compositeID" type="xs:string" many="false">6c2ae802-ba34-4f73-8968-afbdb0108b9</
  <component name="BPELProcess1" version="2.0">
    <implementation.bpel src="BPEL/BPELProcess1.bpel"/>
    <componentType>
      <service name="bpelprocess1_client" ui:wsdLocation="WSDLs/BPELProcess1.wsd1">
        <interface.wsd1 interface="http://xmlns.oracle.com/Application2/J2CA_Outbound/BPELProcess1#
        </service>
        <reference name="Service" ui:wsdLocation="WSDLs/J2CA_Outbound_invoke.wsd1">
          <interface.wsd1 interface="http://xmlns.oracle.com/pcbpel/iWay/wsd1/Siebel/sieb_isdsrv15_tg
        </reference>
      </componentType>
    </component>
  </composite>
```

3. 図 6-36 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 6-36 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、次のステップを実行します

1. 図 6-37 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、「J2CA_Outbound」をクリックします。

図 6-37 「J2CA_Outbound」オプション

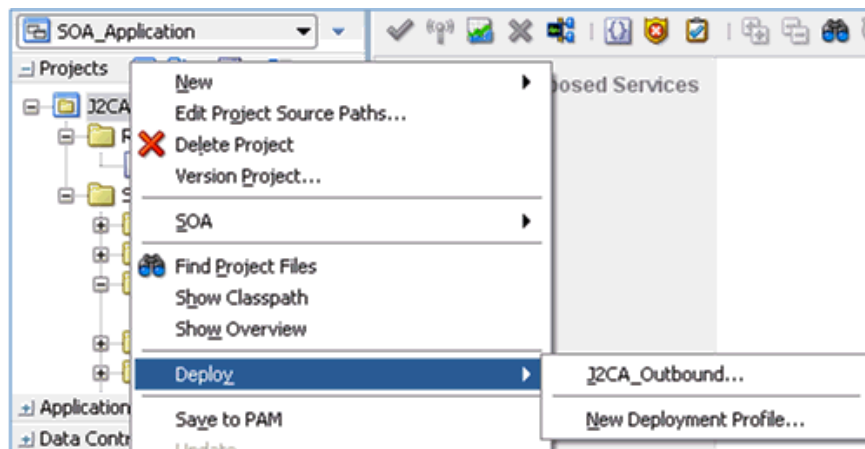
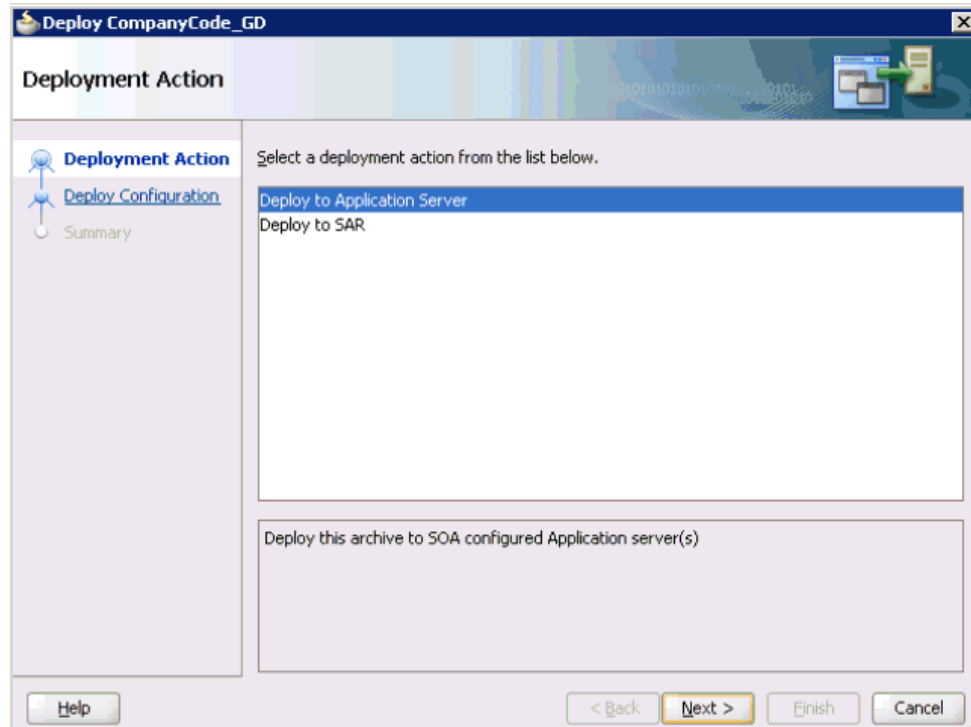


図 6-38 に示すように、「デプロイメント・アクション」ページが表示されます。

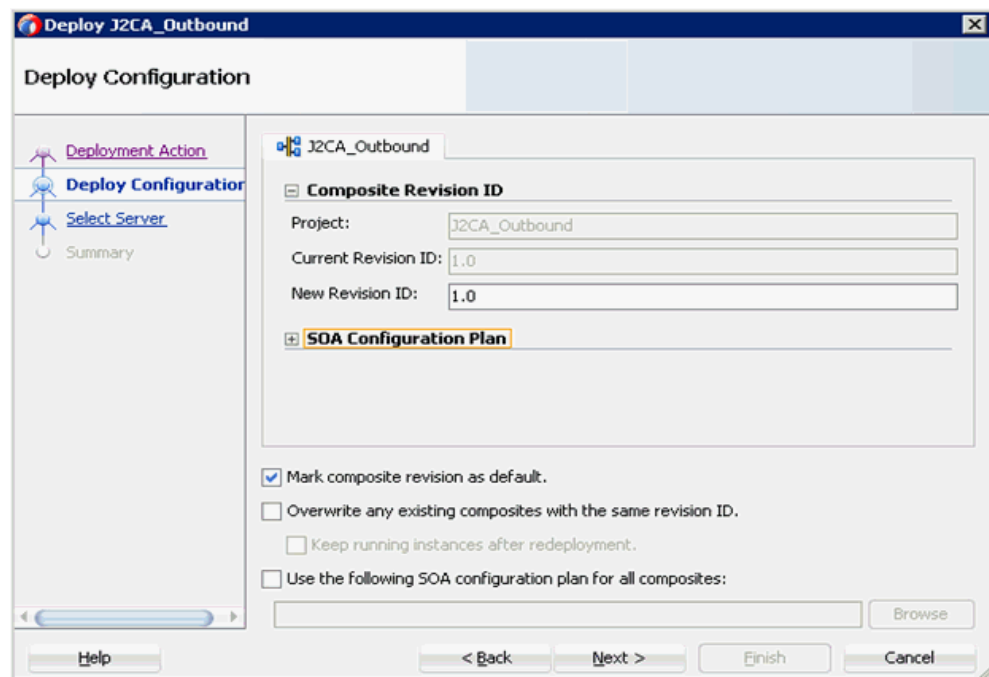
図 6-38 「デプロイメント・アクション」 ページ



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。

図 6-39 に示すように、「構成のデプロイ」ページが表示されます。

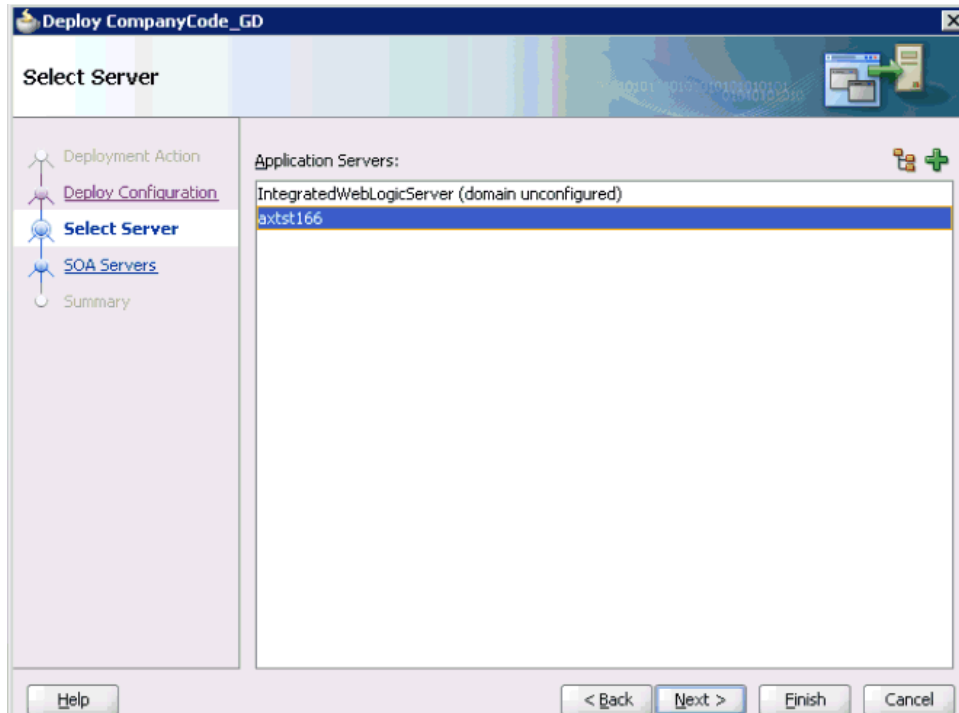
図 6-39 「構成のデプロイ」 ページ



4. デフォルト値を選択したままにし、「次へ」をクリックします。

図 6-40 に示すように、「サーバーの選択」ページが表示されます。

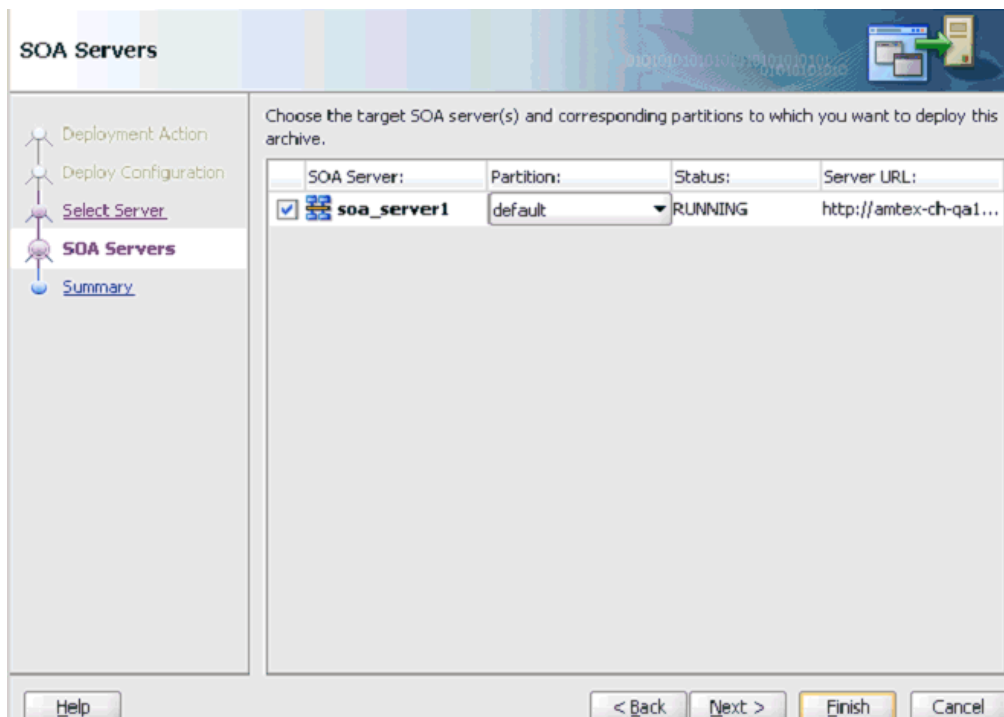
図 6-40 「サーバーの選択」ページ



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 6-41 に示すように、「SOA サーバー」ページが表示されます。

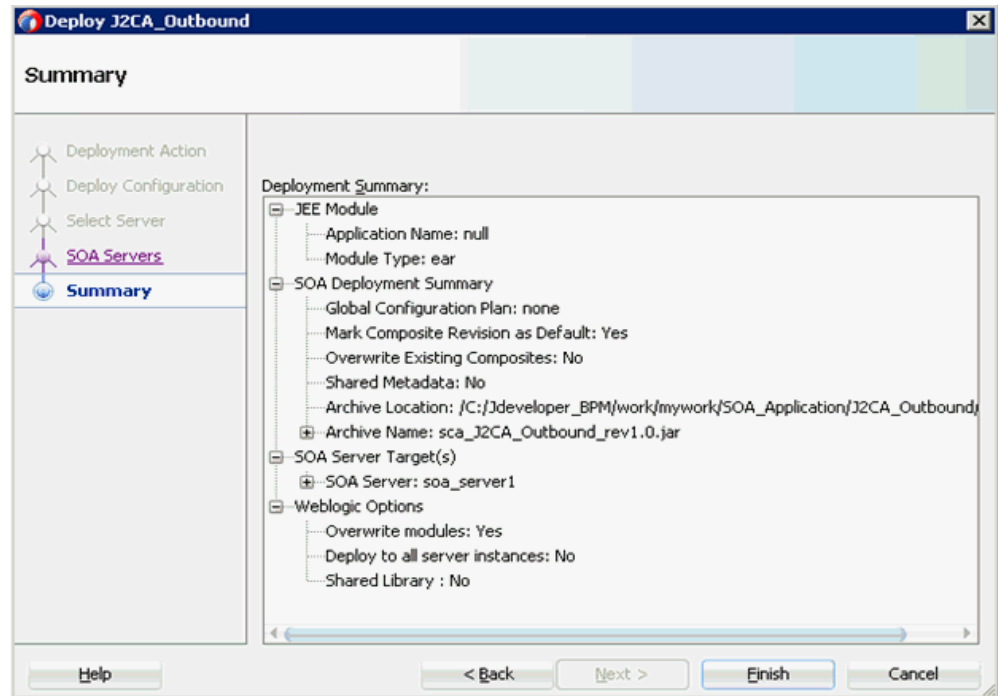
図 6-41 「SOA サーバー」ページ



6. ターゲットの SOA サーバーを選択し、「次へ」をクリックします。

図 6-42 に示すように、「サマリー」ページが表示されます。

図 6-42 「サマリー」ページ



7. プロジェクトに使用可能なすべてのデプロイメント情報をレビューおよび確認し、「終了」をクリックします。

図 6-43 に示すように、プロセスが正常にデプロイされます。

図 6-43 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 ドキュメントの起動

Oracle Enterprise Manager コンソールで入力 XML ドキュメントを起動するには、次のステップを実行します

1. Oracle Enterprise Manager コンソールにログインします。

注意: 12c (12.2.1.1.0) を使用しているお客様は、次のステップを実行します。

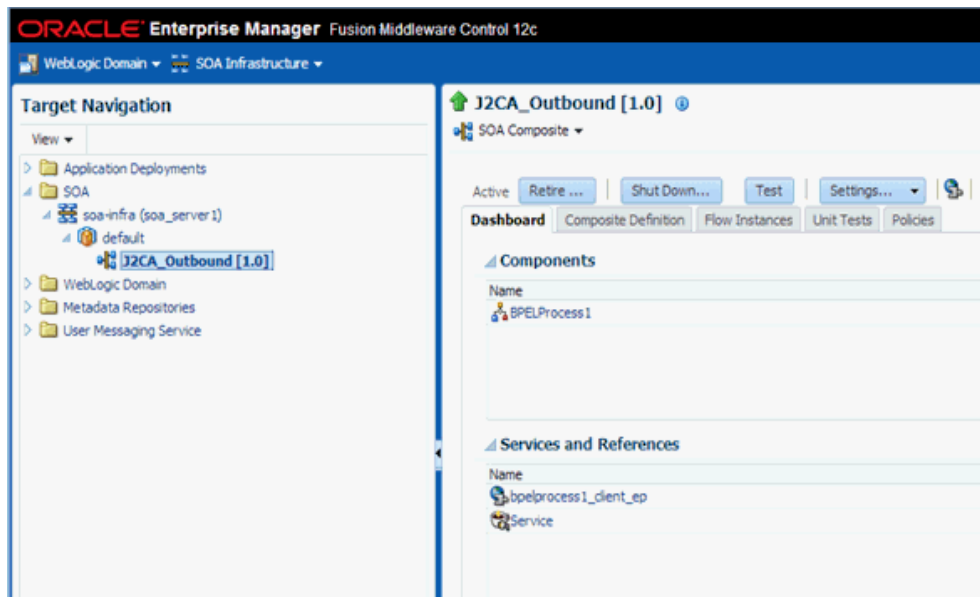
2. 左ペインの「ターゲット・ナビゲーション」をクリックし、「SOA」を展開し、「soa-infra (soa_server1)」を選択します。

3. 「デプロイ済コンポジット」タブをクリックすると、デプロイされたすべてのコンポジットが表示されます。使用可能なプロジェクト(「J2CA_Outbound」など)をクリックします。

この手順のステップ 4 にスキップします。

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(「J2CA_Outbound」など)を選択します。

図 6-44 「テスト」 ボタン



4. 図 6-44 に示すように、「テスト」をクリックします。

図 6-45 「リクエスト」タブ

Request Response

Security

OWSM Security Policies HTTP Basic Auth Advanced None

Quality of Service

HTTP Header

Additional Test Options

Input Arguments

Tree View Enable Validation Load Payload No file chosen

SOAP Body

View

Name	Type	Value
* payload	payload	
* COMPANYCODEID	string	1000

5. 図 6-45 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入力し、「Web サービスのテスト」をクリックします。
- 図 6-46 に示すように、Oracle Enterprise Manager コンソールに出力レスポンスが表示されます。

図 6-46 出力レスポンス

Service cc_getdetail_client_ep
Port cc_getdetail_pt
Operation process

Endpoint URL http://antex-ch-qal66.antexpdc:8001/soa-infra/services/default/CompanyCode_GD/ Edit Endpoint URL

Request Response

Test Status Passed
Response Time (ms) 23438

Tree View

Name	Type	Value
payload	payload	
COMPANYCODE_ADD...	BAPI0002_3	
ADDR_NO	string	000000121
FORMOFADDR	string	Firma
NAME	string	Ides AG
NAME_2	string	
NAME_3	string	
NAME_4	string	
C_D_NAME	string	
CITY	string	Frankfurt
DISTRICT	string	
CITY_NO	string	
POSTL_C001	string	60441

XML ビューを使用して入力 XML ドキュメントを起動するには、次の手順を実行します。

- a. 図 6-47 に示すように、リストから **XML 表示** を選択します。

図 6-47 「入力引数」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. 「入力引数」領域に適切な入力 XML ドキュメントを入力し、「Web サービスのテスト」をクリックします。

図 6-48 に示すように、Oracle Enterprise Manager コンソールに出力レスポンスが表示されます。

図 6-48 表示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">
<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>0000000121</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 およびアウトバウンド・メディエータ・プロセスのテスト

Oracle Enterprise Manager コンソールでアウトバウンド 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とき、これらのコンソールで生成された XML エンベロープは使用しないでください。かわりに、これらを削除し、ネームスペース修飾の WSDL に準拠する、スキーマから生成された XML ペイロードを使用します。

Enterprise Manager コンソールを使用してメディエータ・データ・フローをテストできます。メディエータ・データ・フローおよび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・プロセスの作成方法の詳細は、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、SAP R/3 イベントは顧客レコードが SAP R/3 システムに追加されたときに発生しています。

Application Adapters インストレーションの次のフォルダに、このインバウンドの使用ケース・シナリオのサンプル・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
```

アダプタの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 アプリケーション・エクスプローラ (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPEL デザイナ (JDeveloper)

注意： この章に記載されている例では JDeveloper が使用されていません。

この節では、次のトピック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 を生成するとき、各インバウンド J2CA サービスに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、正しい 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。

この節では、次のトピックについて説明します。

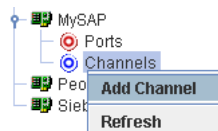
- 6.5.1.1 項「チャンネルの作成」
- 6.5.1.2 項「イベント通知のための WSDL の生成」

6.5.1.1 チャンネルの作成

チャンネルを作成する手順は、次のとおりです。

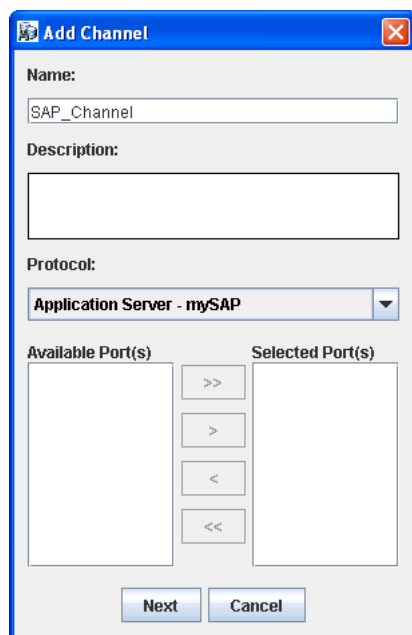
1. アプリケーション・エクスプローラを起動し、J2CA 構成に接続します。
2. 左ペインで、「イベント」ノードを展開します。
3. 「MySAP」ノードを展開します。
 - ☒ 6-49 に示すように、左ペインにポートおよびチャンネル・ノードが表示されます。

図 6-49 「MySAP」の「チャンネル」ノード



4. 「チャンネル」を右クリックし、「チャンネルの追加」を選択します。
 - ☒ 6-50 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 6-50 「チャンネルの追加」ダイアログ



次のステップを実行します。

- a. 「SAP_Channel」など、チャンネルの名前を入力します。
 - b. 簡単な説明を入力します (オプション)。
 - c. 「プロトコル」リストから、「Application Server - mySAP」を選択します。
5. 「次へ」をクリックします。
 - ☒ 6-51 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。次のタブが使用可能です。
 - ユーザー (必須)
 - システム (必須)

- セキュリティ
- 詳細

図 6-51 「アプリケーション・サーバー」ダイアログ

The screenshot shows a dialog box titled 'Application Server' with a close button (X) in the top right corner. It has four tabs: 'User', 'System', 'Security', and 'Advanced'. The 'Security' tab is selected. Inside the dialog, there are four input fields: 'Client*', 'User*', 'Password*', and 'Authentication mode'. The 'Authentication mode' is a dropdown menu currently showing 'Password'. At the bottom of the dialog are 'OK' and 'Cancel' buttons. Below the dialog, a red text note reads 'Fields marked with * are required.'

6. 「ユーザー」タブでは、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定義されたクライアント数
ユーザー	SAP R/3 アプリケーション用の有効なユーザー ID
パスワード	SAP R/3 アプリケーション用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ときに使用する認証モード。デフォルトでは、リストから「パスワード」が選択されています。

7. 図 6-52 に示すように、「システム」タブをクリックします。

図 6-52 「システム」タブ

8. 「システム」タブでは、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	このチャンネルで使用したい SAP R/3 のプログラム ID
アプリケーション・サーバー	使用している 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 のシステム番号
プール・パラメータの受け入れ (True False)	False の場合は、デフォルトのアダプタ接続チューニングを使用します。 True の場合は、ユーザーが指定した値を使用します。
接続プール・サイズ	プールの最大接続数。この値が「最大接続数」よりも小さい場合、プール・サイズは JCo によって「最大接続数」のサイズに設定されます。
最大接続数	一度に作成される最大接続数。
接続タイムアウト (分)	未使用の接続が解放されるまでの時間 (分)。
接続待機時間 (秒)	すべての接続がビジー状態の場合に空き接続ハンドルを待機する時間 (秒)。この時間を経過すると、リソース例外がスローされます。

9. 「セキュリティ」タブ (任意) では、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 は無効です。SNC を有効にするには、リストから「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	リストから 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 (任意) では、次の表の情報に基づいて SAP R/3 チャ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	<p>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。</p> <p>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」に変更されます。</p> <p>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。</p>
IDOC フォーマット	<p>リストから IDOC タイプを選択します。</p> <ul style="list-style-type: none"> ■ XML (デフォルト) ■ XML-CDATA-ENVELOPED
IDOC リリース	接続に使用する IDOC のバージョン。
IDOC リリース・プロバイダ	接続用の IDOC リリース・プロバイダ。リストから、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。

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
SAP トレース	<p>このオプションを選択して SAP トレースを有効にします。</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><ADAPTER_HOME>\tools\iwae\bin</pre> <p>ランタイム :</p> <pre><ORACLE_HOME>\user_projects\domains\base_domain</pre> <p>SAP トレースに使用するファイル名のフォーマットは、次のとおりです (デザインタイムおよびランタイム)。</p> <ul style="list-style-type: none"> ■ rfc02664_04332.trc ■ JCO100226_043846994.trc
処理モード	<p>リストから同期処理のタイプを選択します。可能な値として、「REQUEST」と「REQUEST_RESPONSE」があります。</p>

11. 「OK」をクリックします。

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。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ます。

注意： チャンネルは BPEL PM Server で管理されているので開始しないでください。テストまたはデバッグ目的でチャンネルを起動する場合は、イベントのランタイム・デプロイメントの前にすべてのチャンネルを停止し、アプリケーション・エクスプローラを閉じてください。

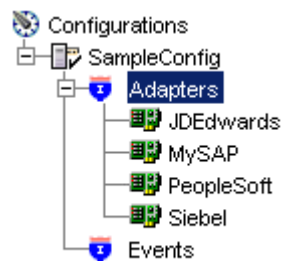
6.5.1.2 イベント通知のための WSDL の生成

チャンネルを作成し、開始していないことを確認した後、アプリケーション・エクスプローラを使用してイベント用に 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 6-53 に示すように、すべてのアダプタのリストが表示されます。

図 6-53 「アダプタ」ノード

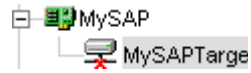


次のステップを実行します。

- a. 「MySAP」ノードを展開します。

図 6-54 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 6-54 切断された「MySAP Target」ノード



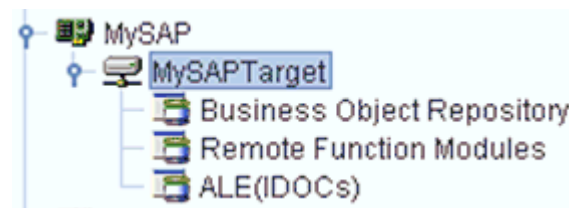
- b. 「MySAP」ノードの下のターゲット名 (「SAPTarget」など) をクリックします。

「接続」ダイアログに保存したパラメータが表示されます。

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

図 6-55 に示すように、「x」アイコンが消え、ノードが接続されたことが示されます。

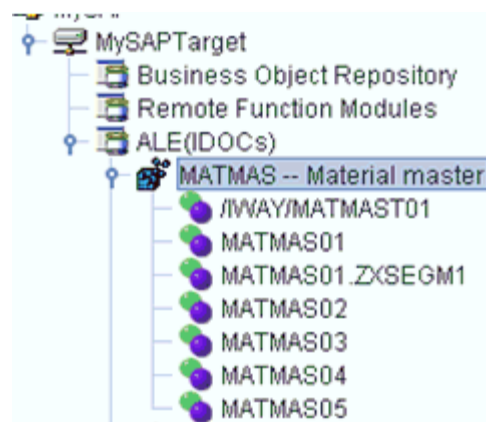
図 6-55 接続された MySAP Target



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

図 6-56 に示すように、MATMAS のリストが表示されます。

図 6-56 「MATMAS - Material Master」リスト



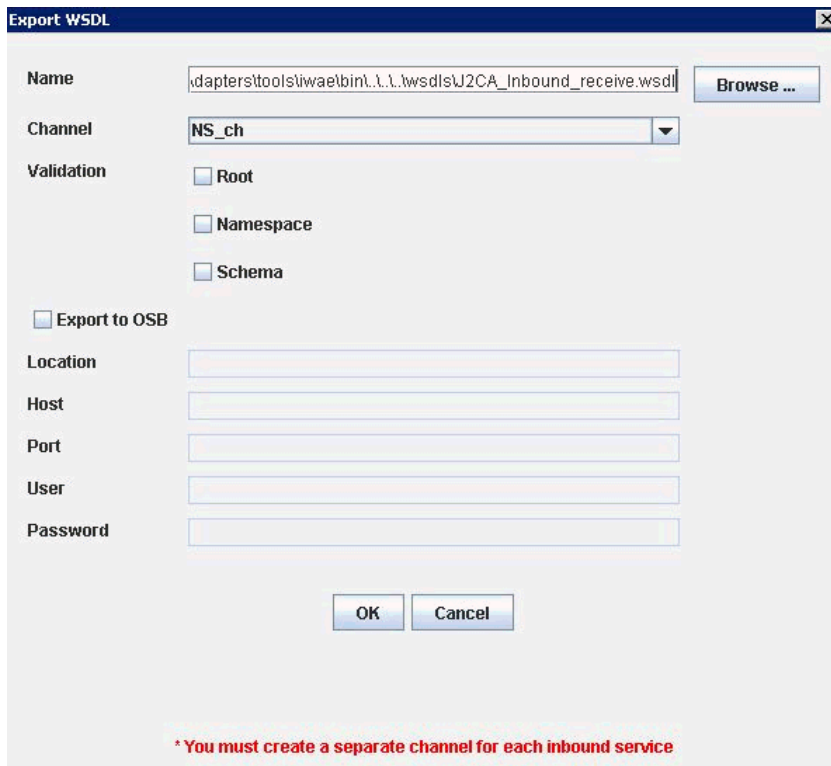
6. MATMAS のリストから「MATMAS01」を右クリックします。

図 6-57 インバウンド JCA サービスの作成 (イベント) オプション



7. 図 6-57 に示すように、インバウンド JCA サービスの作成 (イベント) を選択します。
 図 6-58 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 6-58 「WSDL のエクスポート」ダイアログ



次のステップを実行します。

- a. 「名前」フィールドで、WSDL ファイルの名前を指定します。
- b. 「チャンネル」リストから、このインバウンド・サービス用に作成したチャンネルを選択します。

重要: 各イベントに個別のチャンネルを作成する必要があります。ランタイム前にチャンネルが停止していることを確認してください。

- c. 「ルート」、「ネームスペース」および「スキーマ」の検証の3つのチェック・ボックスも使用できます。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。

実行時には、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、実行時に最初の検証が失敗すると、残りの検証オプションは処理されません。ルート検証およびネームスペース検証は、適度な検証レベルであるとみなされます。スキーマ検証はより厳しい検証レベルです。ルート要素およびネームスペースが SAP 環境の IDOC 間で異ならないかぎり、ルート検証とネームスペース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 での BPEL インバウンド・プロセスの定義に必要な最初のステップである、SOA 用の空のコンポジットを作成する準備ができました。

6.5.2 SOA 用の空のコンポジットの作成

SOA 用に空のコンポジットを作成するには、以下の手順を実行します。

1. 新規の SOA アプリケーションを作成します。
2. 新規の SOA アプリケーションに名前を入力し、「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。
詳細については、6-9 ページ [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、次のトピックで構成される BPEL インバウンド・プロセスの定義方法について説明します。

- [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、次のステップを実行します。

1. [図 6-59](#) に示すように、「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「サード・パーティ・アダプタ」コンポーネントをドラッグ・アンド・ドロップします。

図 6-59 「サード・パーティ・アダプタ」コンポーネント

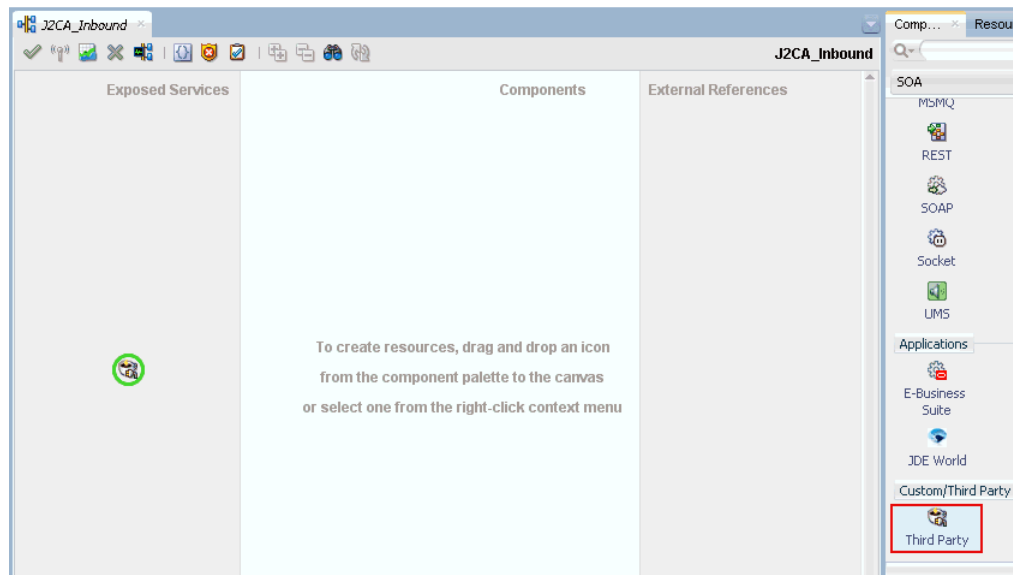


図 6-60 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

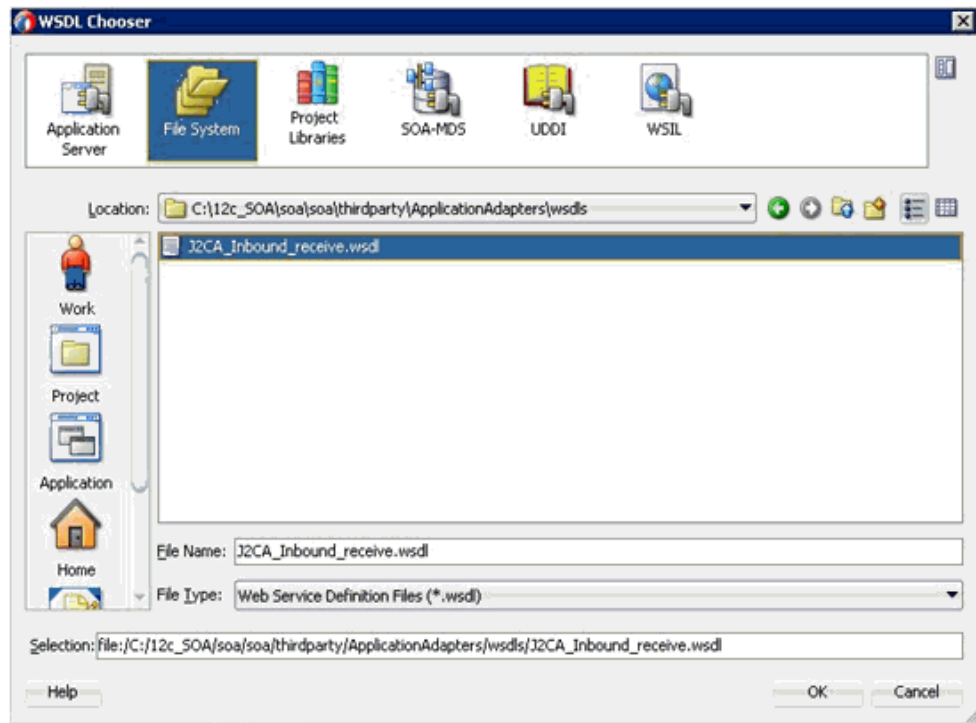
図 6-60 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



2. サード・パーティ・アダプタ・サービスの名前を入力するか、デフォルト名をそのまま使用します。
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」フィールドの右の既存の WSDL の検索アイコンをクリックします。

図 6-61 に示すように、「WSDL チューザ」ダイアログが表示されます。

図 6-61 「WSDL チューザ」 ダイアログ



5. 次のディレクトリまでブラウズし、インバウンド WSDL ファイルを選択します。

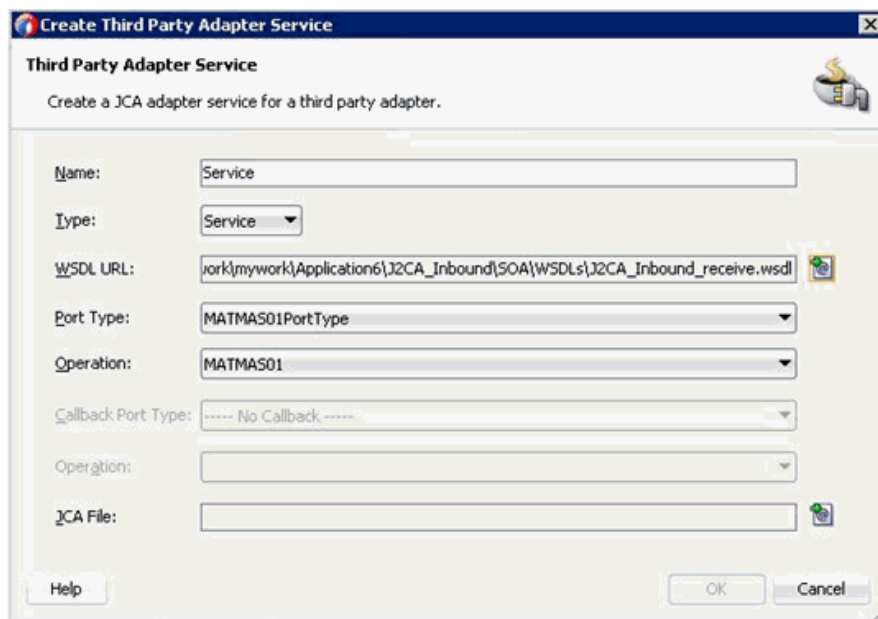
<ADAPTER_HOME>\wsdl

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 ファイルおよび関連リクエスト/レスポンス XML スキーマ・ファイル (.xsd) が、作成済のプロジェクト・フォルダにインポートされます。

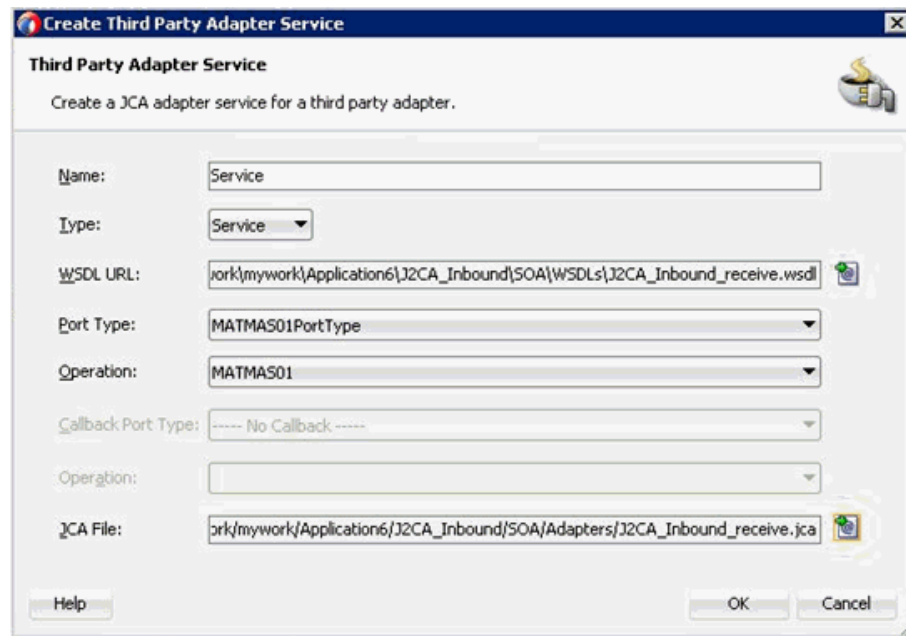
図 6-62 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

図 6-62 「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」フィールドの右の **JCA ファイルの検索** 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. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。
<ADAPTER_HOME>\wsdl1
10. 「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。
11. 「はい」をクリックします。
JCA プロパティ・ファイルのコピーがプロジェクト・フォルダ内に作成されます。
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」ダイアログに戻ります。

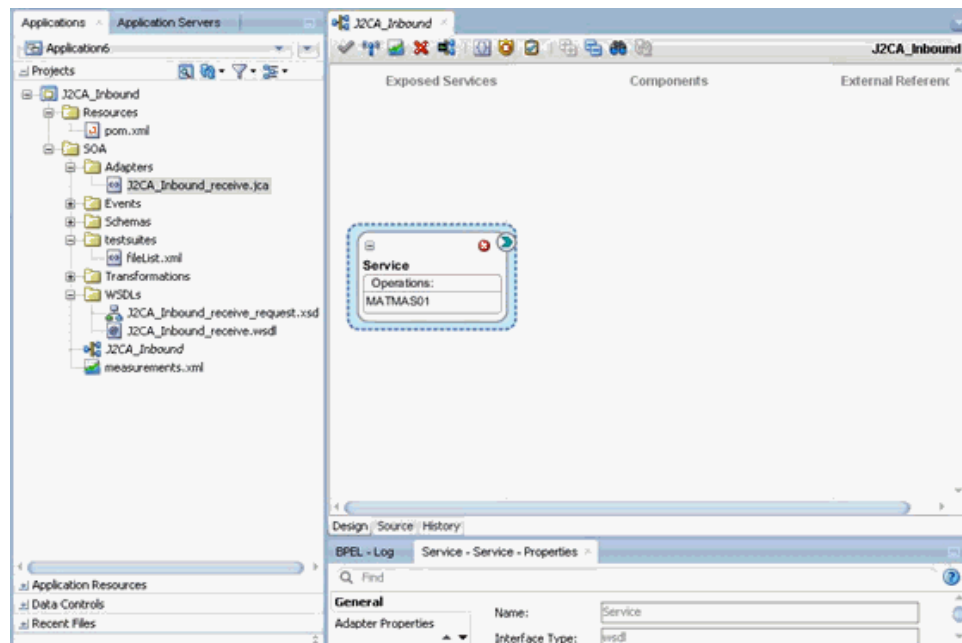
図 6-63 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



12. 「OK」をクリックします。

図 6-64 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 6-64 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。

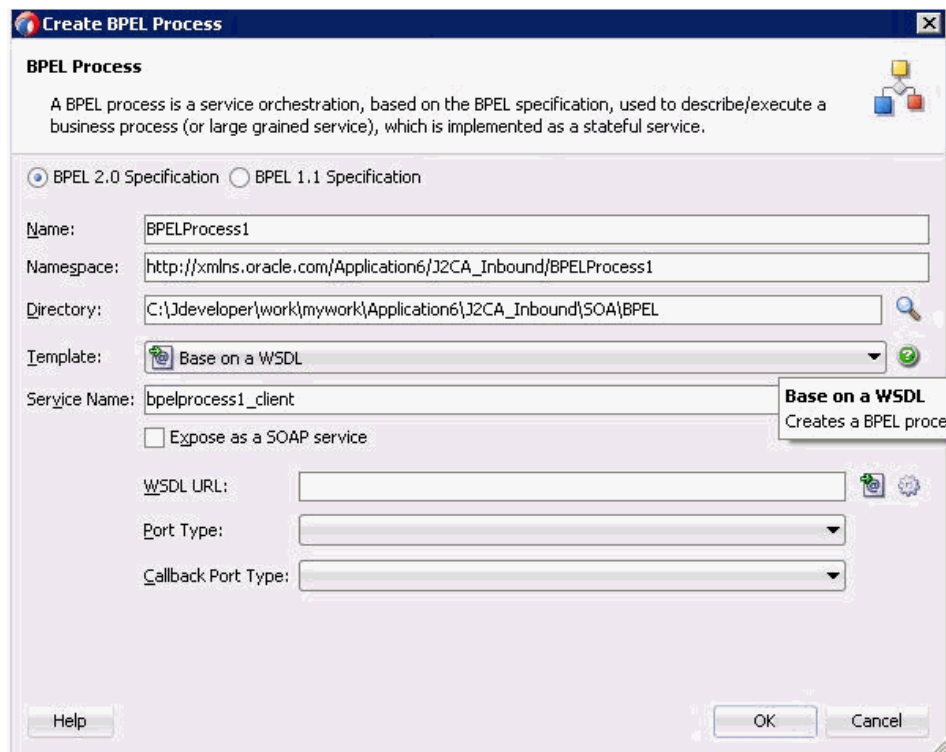
6.5.3.2 インバウンド BPEL プロセス・コンポーネントの作成

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、次のステップを実行します

1. 「サービス・コンポーネント」 ペインから「BPEL プロセス」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・ドロップします。

図 6-65 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。

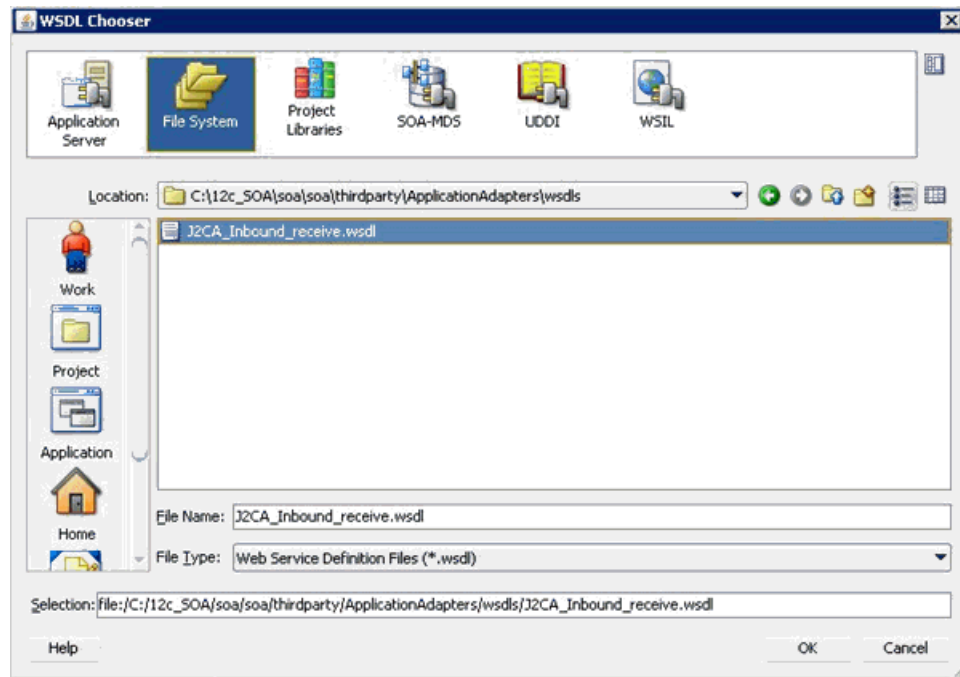
図 6-65 「BPEL プロセスの作成」ダイアログ



2. 「名前」フィールドで、新規のインバウンド BPEL プロセス・コンポーネントの名前を入力するか、デフォルト名をそのまま使用します。
デフォルトで、「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」リストから「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」フィールドの右の既存の WSDL の検索アイコンをクリックします。

図 6-66 に示すように、「WSDL チューザ」ダイアログが表示されます。

図 6-66 「WSDL チューザ」 ダイアログ



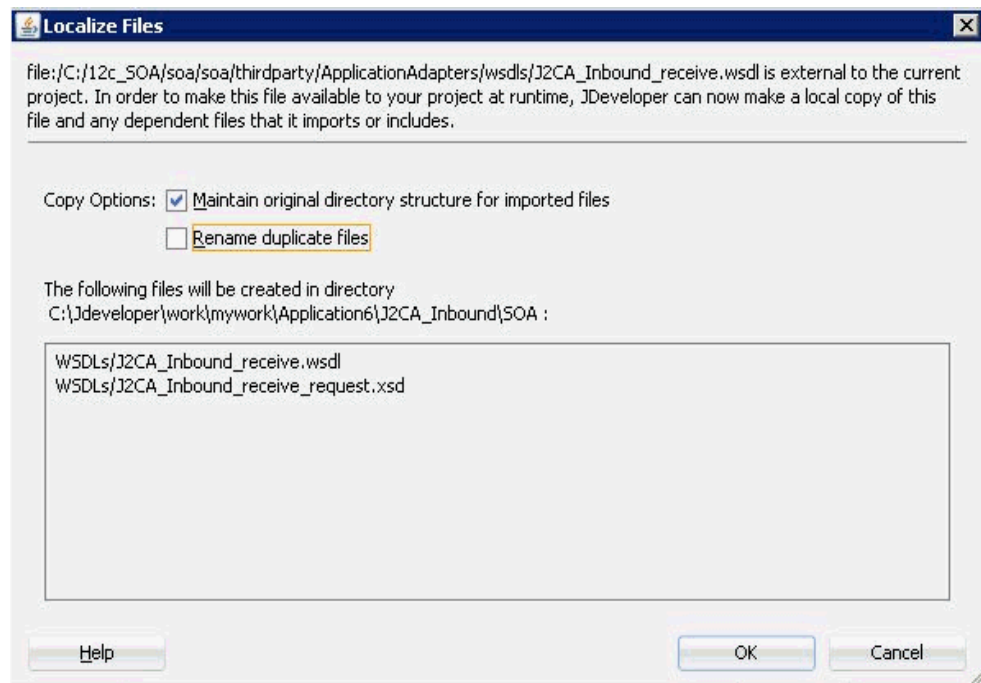
6. 次のディレクトリからインバウンド WSDL ファイルを選択します。

```
<ADAPTER_HOME>\wsdls
```

7. 「OK」をクリックします。

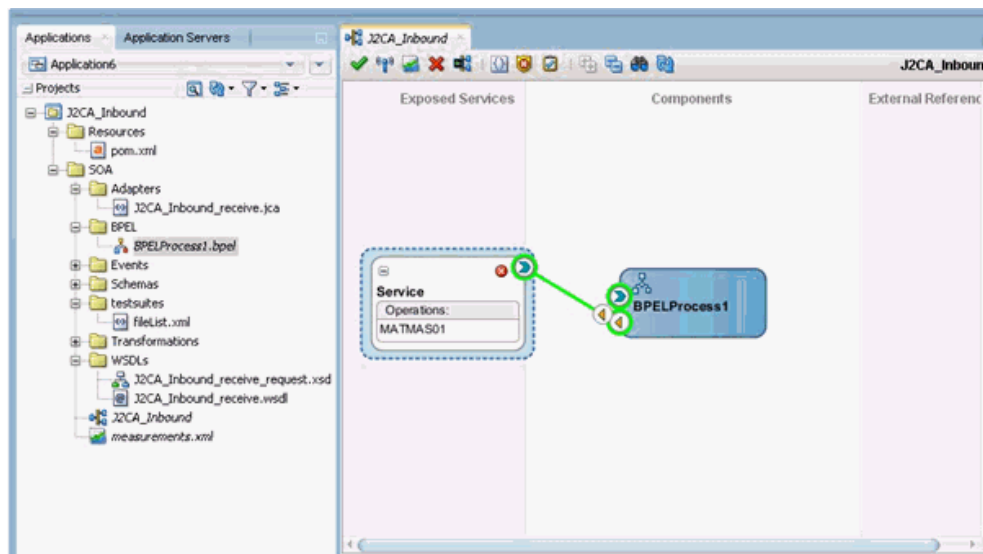
図 6-67 に示すように、「ファイルのローカライズ」ダイアログが表示されます。

図 6-67 「ファイルのローカライズ」ダイアログ



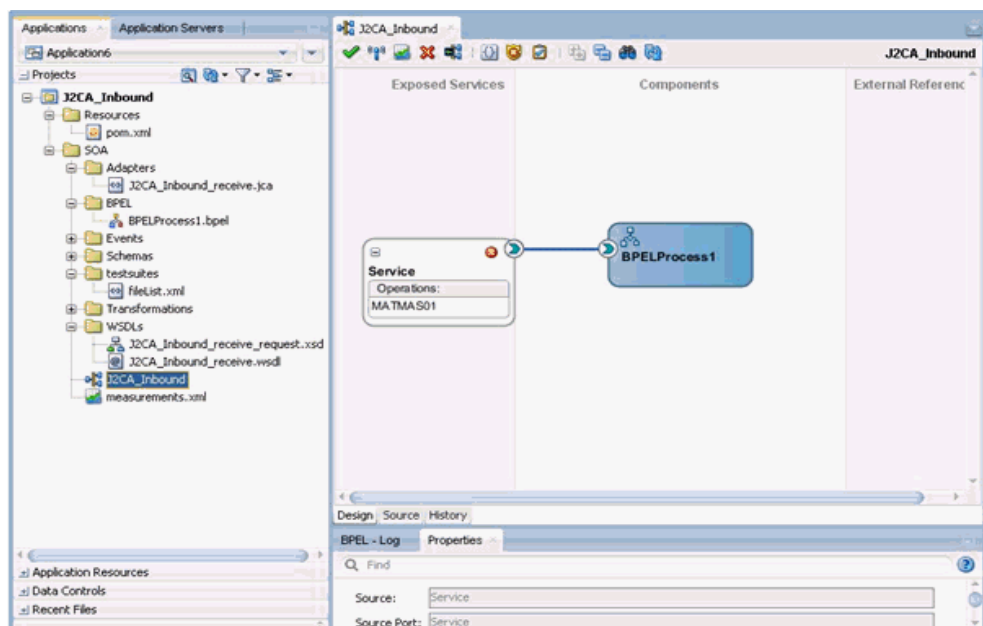
8. 「重複ファイルの名前変更」オプションの選択を解除します。
9. 「OK」をクリックします。
「BPEL プロセスの作成」ダイアログに戻ります。
10. 「OK」をクリックします。


図 6-68 作成された接続



11. 図 6-68 に示すように、サード・パーティ・アダプタ・サービス・コンポーネントとインバウンド BPEL プロセス・コンポーネント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 6-69 「すべて保存」アイコン



13.  [図 6-69](#) に示すように、メニュー・バーで「すべて保存」アイコンをクリックし、構成された新規のインバウンド 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 との既知のデプロイメント問題の調整

12c での既知のデプロイメント問題の調整方法については、次を参照してください。
6-27 ページ [6.4.3.3 項「12c との既知のデプロイメント問題の調整」](#)

6.5.4 BPEL インバウンド・プロセスのデプロイ

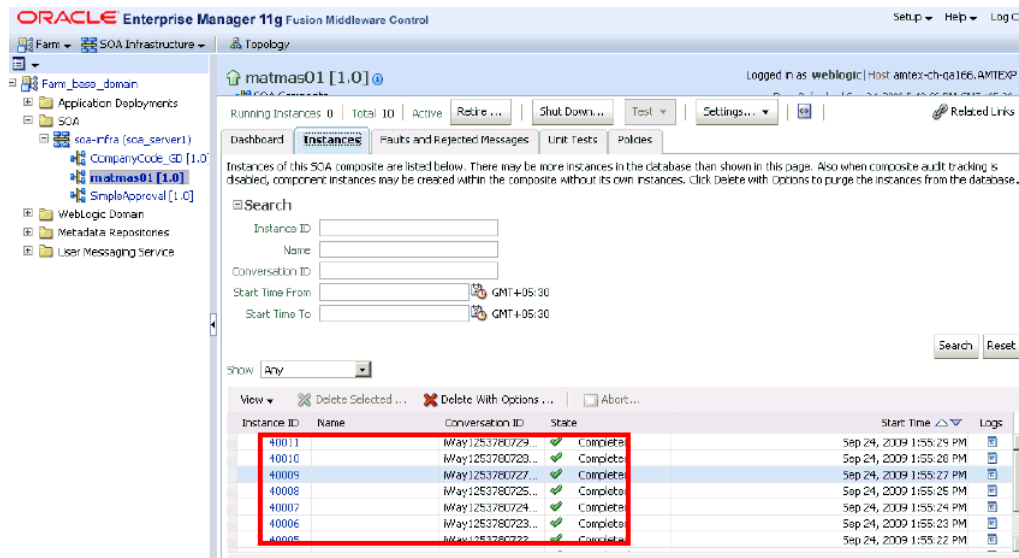
BPEL インバウンド・プロセスをデプロイするには、次のステップを実行します

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、「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したままにし、「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. ターゲットの 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. プロジェクトに使用可能なすべてのデプロイメント情報をレビューおよび確認し、「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。

詳細については、6-30 ページ [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。

イベント・メッセージが SAP GUI を介してトリガーされると、[図 6-70](#) に示すように、Oracle Enterprise Manager コンソール側でインスタンスが正常に受信されます。

図 6-70 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・システム内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 は、システム内で顧客情報が更新されると、イベントを生成する可能性があります。イベントの詳細は、4-22 ページ「[イベント・アダプタの構成](#)」を参照してください。

この節では、次のトピックについて説明します。

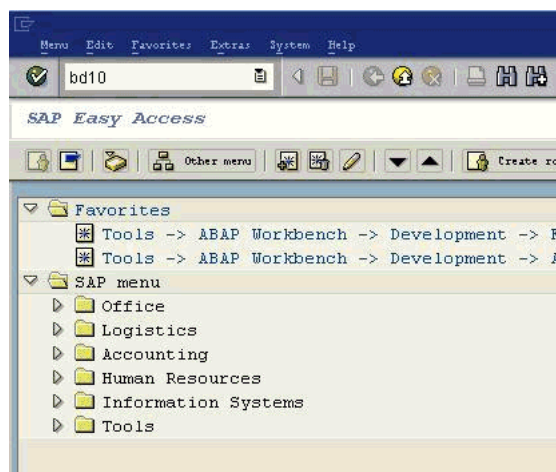
■ 6.5.5.1 項「結果の確認」

次のトピックでは、Oracle Application Adapter for SAP R/3 を使用して、SAP R/3 のイベントをトリガーし、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手順は、次のとおりです。

1. [図 6-71](#) に示すように、SAP Workbench を起動して SAP R/3 システムにログインします。

図 6-71 SAP Workbench



2. **bd10** トランザクションを実行します。

図 6-72 「Send Material」 ウィンドウ

「Send Material」 ウィンドウで次の情報を入力します。

- a. 図 6-72 に示すように、「Material」フィールドに 1 から 100 の範囲でマテリアル番号を入力します。
 - b. 論理システム・フィールドに SAP R/3 とともに使用する論理システムを指定します。
3. 「実行」 ボタンをクリックします。

マテリアル・マスター・データが指定し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が同じ値のプログラム ID を定義している場合、そのチャンネルは SAP R/3 からこのマテリアル・マスター・データを受信します。

6.5.5.1 結果の確認

結果を確認する手順は、次のとおりです。

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、「SOA」フォルダの順に展開します。
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」をクリックすると、受信したイベント・メッセージが表示されます。

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

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

Application Adapters インストレーションの次のフォルダに、この使用ケース・シナリオのサンプル・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

この節では、次のトピック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「チャネルの構成」
- 6.6.4 項「メッセージ用の 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、SAP R/3 イベントを受信する 2 つの方法 (リモート関数呼出し(RFC) または Intermediate Document (IDoc)) をサポートします。イベント処理の間、アダプタは、SAP R/3 から RFC と IDoc を直接受信します。SAP メッセージ・オブジェクトは、SAP R/3 イベントを定義し、SAP R/3 から公開されます。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ド・パーティ・アプリケーションに公開する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、非同期または同期的に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公開し、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・スコアがチェックされます。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。これらのすべてによって、単一のトランザクションが構成されます。

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 との同期イベントの場合、SAP R/3 はイベントを公開する際、クライアントになります。Oracle Application Adapter for SAP R/3 は、リクエストを受信してサーバーとして機能し、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、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。

この使用ケース・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。この使用ケースでは、2 つの SAP R/3 システムが構成され、これら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。1 つの 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 1 つの 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。この使用ケース・シナリオは他の EIS でも使用できます。ただし、ソース・システムは常に 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。

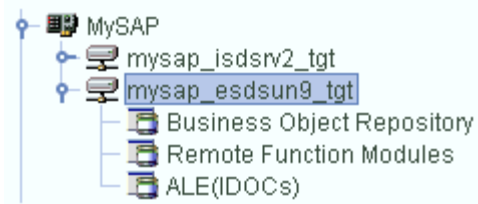
注意： 同期イベントはリプライ・スキーマのみを所有しているオブジェクトでサポートされ、IDoc ではサポートされていません。

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 6-73 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 に対して 2 つのターゲットを作成します。

図 6-73 SAP ターゲット



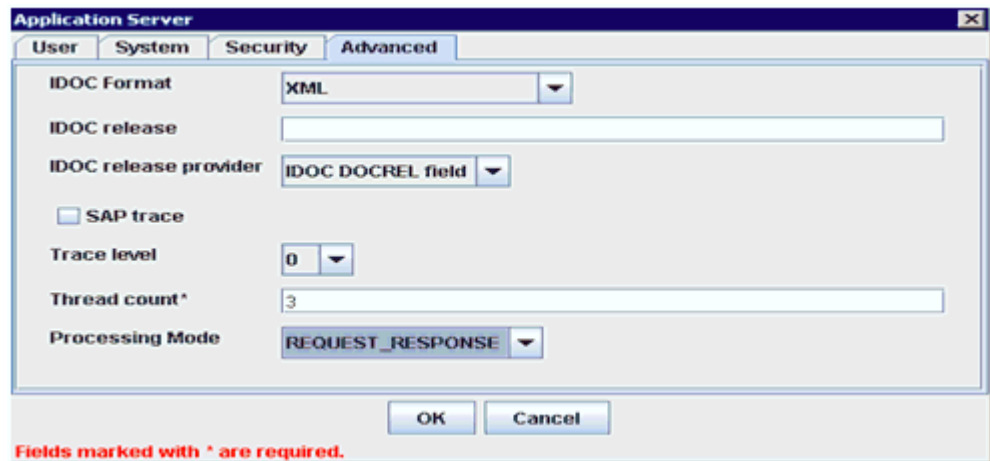
アプリケーション・エクスプローラの起動、J2CA 構成の作成、および SAP R/3 へのターゲットの定義の詳細は、第 4 章「Oracle Application Adapter for SAP R/3 の構成」を参照してください。

6.6.3 チャネルの構成

この項では、チャネルを構成する方法について説明します。

1. 図 6-74 に示すように、チャネルを作成し、「詳細」タブの「処理モード」リストから「REQUEST_RESPONSE」が選択されていることを確認します。

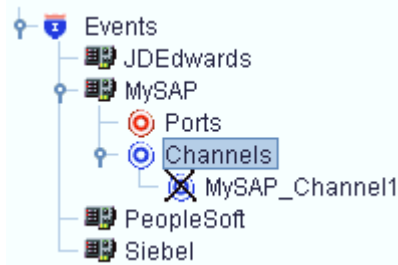
図 6-74 「アプリケーション・サーバー」ダイアログ



チャネルの作成方法の詳細は、4.10 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。左ペインのチャネル・ノードの下にチャネルが表示されます。図 6-75 に示すように、アイコン上の「X」は、そのチャネルが現在停止していることを示します。

図 6-75 新規チャネル



注意： チャネルは BPEL PM Server で管理されているので開始しないでください。テストまたはデバッグ目的でチャネルを起動する場合は、イベントのランタイム・デプロイメントの前にすべてのチャネルを停止し、アプリケーション・エクスペローラを閉じてください。

6.6.4 メッセージ用の WSDL の生成

次のステップを実行して、メッセージ用の WSDL を生成します。

1. WSDL を生成する前に、チャネルが開始し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 (「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 (「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)。

詳細については、6.4.1 項「リクエスト / レスポンス型のサービス用の 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手順は、次のとおりです。

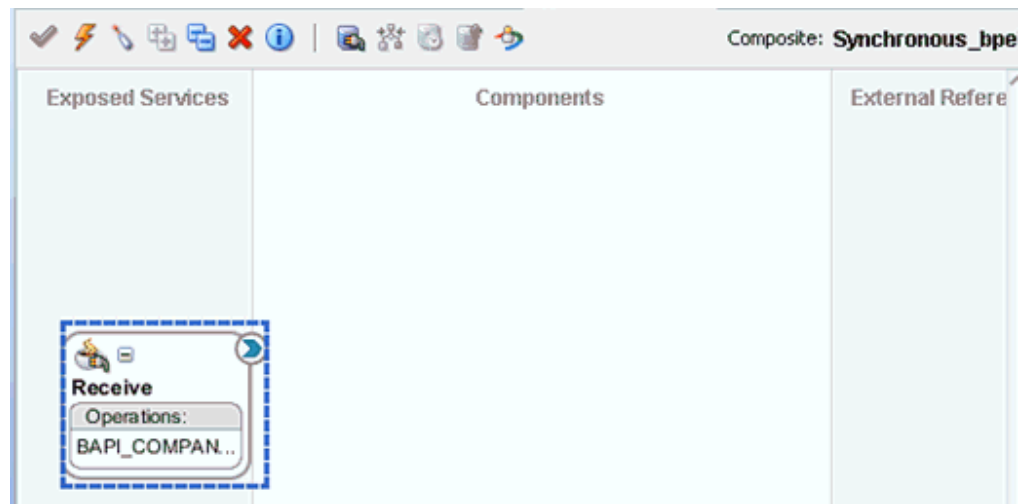
1. JDeveloper で、SOA アプリケーションを選択し、新規プロジェクトを作成します。
2. 「新規ギャラリー」ウィンドウで、SOA プロジェクトを選択して「OK」をクリックします。
3. プロジェクト名を入力して、「次へ」をクリックします。
4. テンプレートとして「空のコンポジット」を選択し、「終了」をクリックします。

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

インバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスを作成する手順は、次のとおりです。

1. 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「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドで名前を入力し、「WSDL URL」フィールドの横のアイコンをクリックします。
「WSDL チューザ」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 の場所から作成されたインバウンド WSDL をブラウザして選択し、「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) が作成済のプロジェクト・フォルダにインポートされます。
4. JCA ファイルの検索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・ファイルを選択して、「OK」をクリックします。
6. 「はい」をクリックします。
JCA プロパティ・ファイルのコピーがプロジェクト・フォルダ内に作成されます。
7. 「OK」をクリックします。
図 6-76 に示すように、インバウンド WSDL 用のサード・パーティ・アダプタ・サービスが「公開されたサービス」ペインに作成されます。

図 6-76 「公開されたサービス」ペイン



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

アウトバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスを作成する手順は、次のとおりです。

1. カスタム/サードパーティ領域から「外部参照」ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドで名前を入力し、「WSDL URL」フィールドの横のアイコンをクリックします。

「WSDL チューザ」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 の場所から作成されたアウトバウンド WSDL をブラウザして選択し、「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) が作成済のプロジェクト・フォルダにインポートされます。

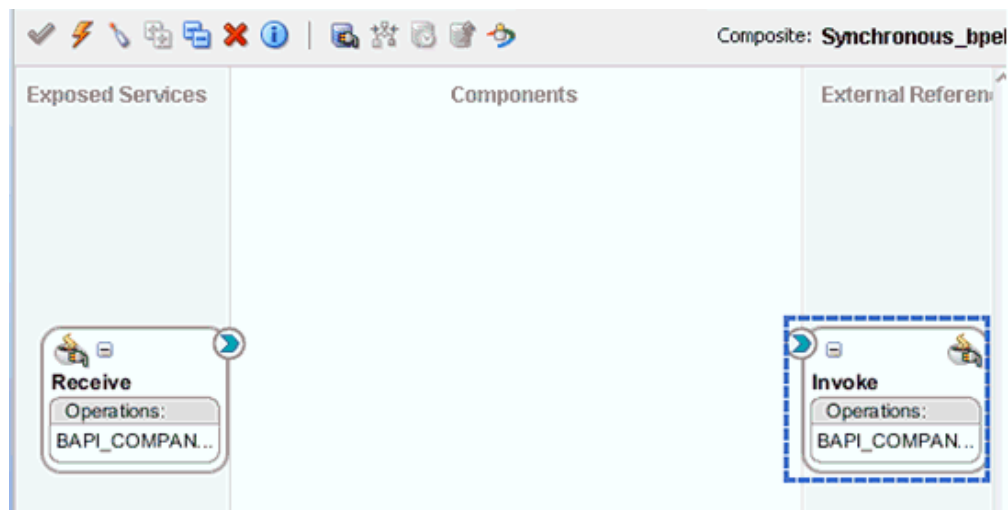
4. JCA ファイルの検索アイコンをクリックして、特定のアウトバウンド JCA プロパティ・ファイルを選択します。
5. WSDL の場所から JCA プロパティ・ファイルを選択して、「OK」をクリックします。
6. 「はい」をクリックします。

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

7. 「OK」をクリックします。

図 6-77 に示すように、アウトバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスが「外部参照」ペインに作成されます。

図 6-77 「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手順は、次のとおりです。

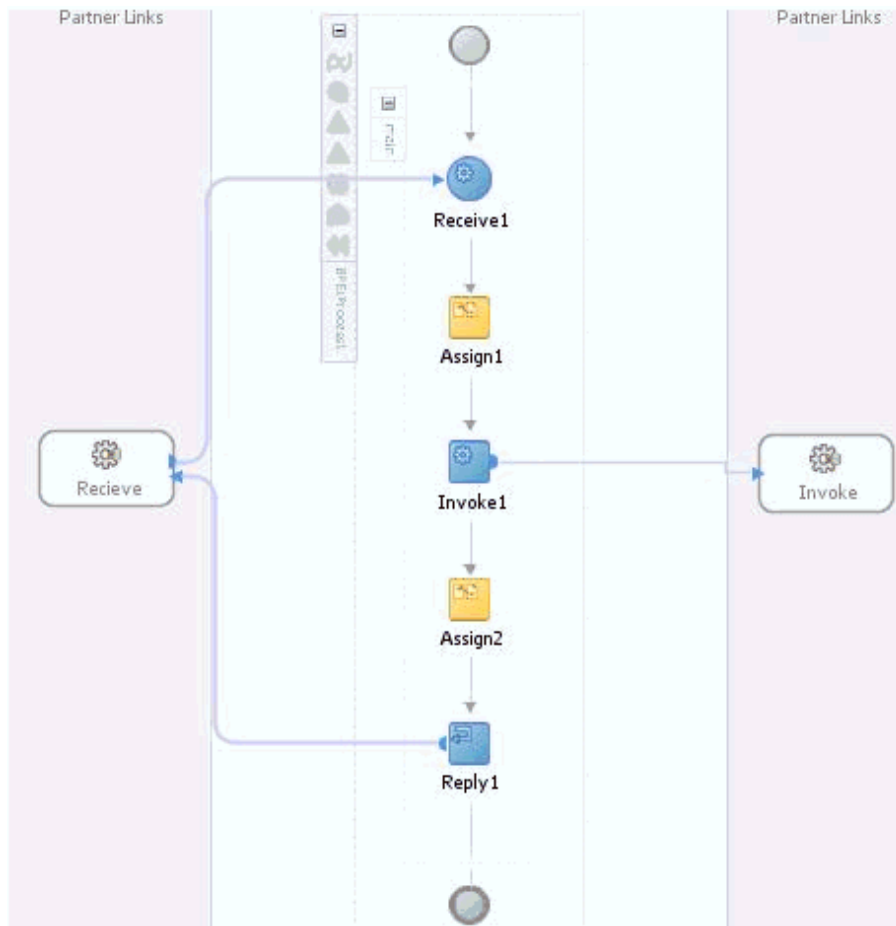
1. 「コンポーネント」領域から「コンポーネント」ペインに「BPEL プロセス」コンポーネントをドラッグ・アンド・ドロップします。
「BPEL プロセスの作成」ウィンドウが表示されます。
2. 「名前」フィールドに BPEL プロセスの名前を入力し、「テンプレート」リストから「サービスを後で定義」を選択します。
3. 「OK」をクリックします。
「BPEL プロセス」コンポーネントが「コンポーネント」ペインに作成されます。
4. サード・パーティ・アダプタ・サービス「受信」を BPEL プロセス・コンポーネント「Synchronous_Events」に接続します。
5. 「OK」をクリックします。

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「受信」サード・パーティ・アダプタ・サービスに接続します。
受信の編集ダイアログが表示されます。
10. 「変数」フィールドの右の**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」、「OK」の順にクリックします。
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「起動」サード・パーティ・アダプタ・サービスに接続します。
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」フィールドの右の**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」フィールドの右の**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」、「OK」の順にクリックします。
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「受信」サード・パーティ・アダプタ・サービスに接続します。
返信の編集ダイアログが表示されます。
25. 「変数」フィールドの**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。

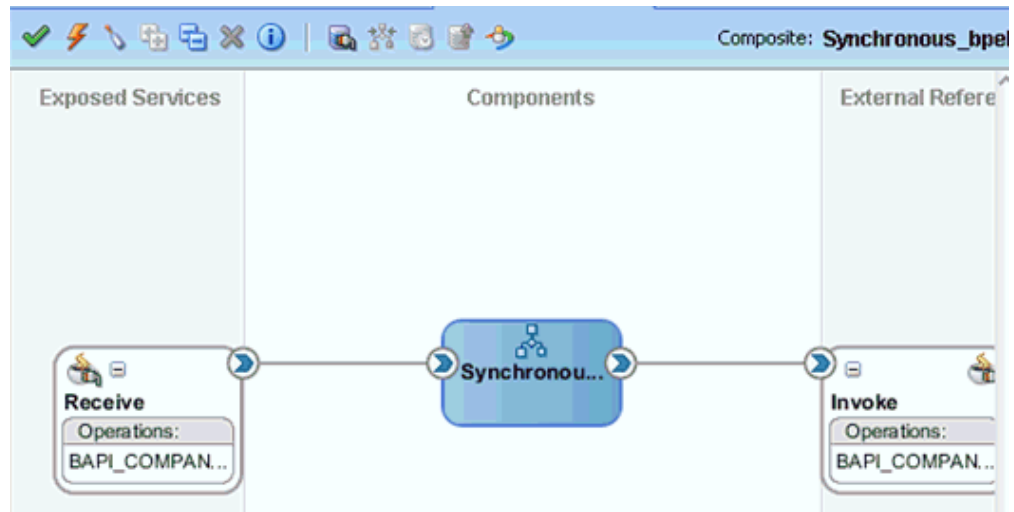
26. 「OK」をクリックします。
返信の編集ダイアログに戻ります。
27. 「適用」、「OK」の順にクリックします。
28. 「Assign1」アイコンをダブルクリックして編集モードで開きます。
29. この「割当て」アクティビティでは、「受信」アクティビティの入力変数を「起動」アクティビティの入力変数に割り当てる必要があります。
30. 「適用」、「OK」の順にクリックします。
ビジュアル・エディタに戻ります。
31. 「Assign2」アクティビティをダブルクリックして編集モードで開きます。
32. この「割当て」アクティビティでは、「起動」アクティビティの出力変数を「返信」アクティビティの出力変数に割り当てる必要があります。
33. 「適用」、「OK」の順にクリックします。
ビジュアル・エディタに戻ります。図 6-78 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 6-78 ビジュアル・エディタ



34. 図 6-79 に示すように、「ファイル」メニューからすべてのプロセスを保存します。

図 6-79 完成した BPEL プロセス



6.6.5.5 12c との既知のデプロイメント問題の調整

12c での既知のデプロイメント問題の調整方法については、6-27 ページの [6.4.3.3 項「12c との既知のデプロイメント問題の調整」](#) 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、6-30 ページ [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。

Application Adapters インストレーションの次のフォルダに、このアウトバウンドの使用ケース・シナリオのサンプル・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
```

アウトバウンドの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 アプリケーション・エクスプローラ (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPEL デザイナ (JDeveloper)

この節では、次のトピック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手順は、次のとおりです。

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。
ターゲットの定義および SAP R/3 への接続の詳細は、4.5.1 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. 「ビジネス・オブジェクト・リポジトリ」、**Financial Accounting**、**Company Code Business Object** の順に展開し、「GetDetail」を選択します。
4. 図 6-80 に示すように、「GetDetail」ノードを右クリックし、メニューから「Web サービスの作成」を選択します。

図 6-80 「GetDetail」ノード

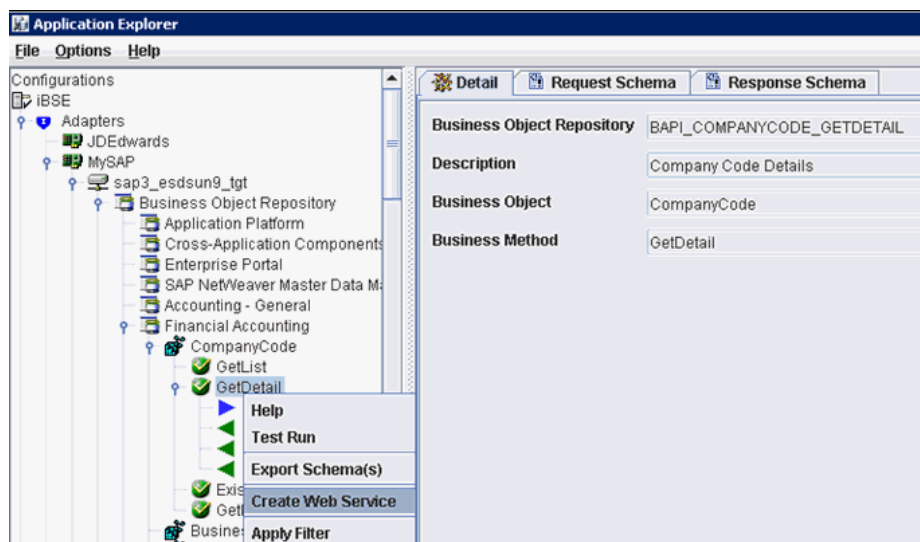
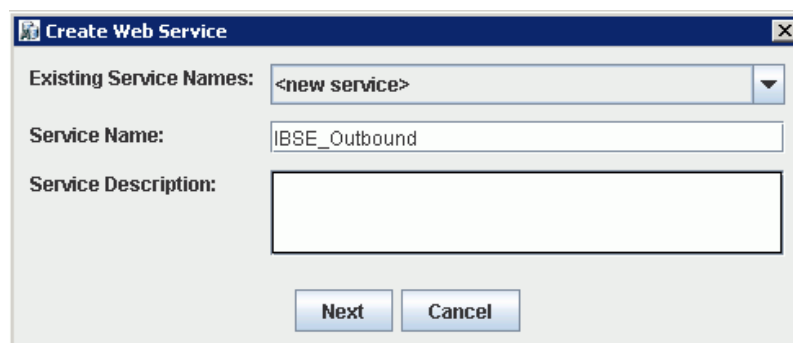


図 6-81 に示すように、「Web サービスの作成」ダイアログが表示されます。

図 6-81 「Web サービスの作成」ダイアログ



5. サービス名を入力して、「次へ」をクリックします。
6. 次に表示されるダイアログで、「OK」をクリックします。
アプリケーション・エクスプローラで、「ビジネス・サービス」ノードに表示が切り替わり、新規の Web サービスが左ペインに表示されます。
7. 新規の Web サービスを右クリックし、メニューから「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 での BPEL アウトバウンド・プロセスの定義に必要な最初のステップである、SOA 用の空のコンポジットを作成する準備ができました。

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手順は、次のとおりです。

1. 新規の SOA アプリケーションを作成します。
2. SOA アプリケーションに名前を入力し、「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。
詳細については、6-9 ページ [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。

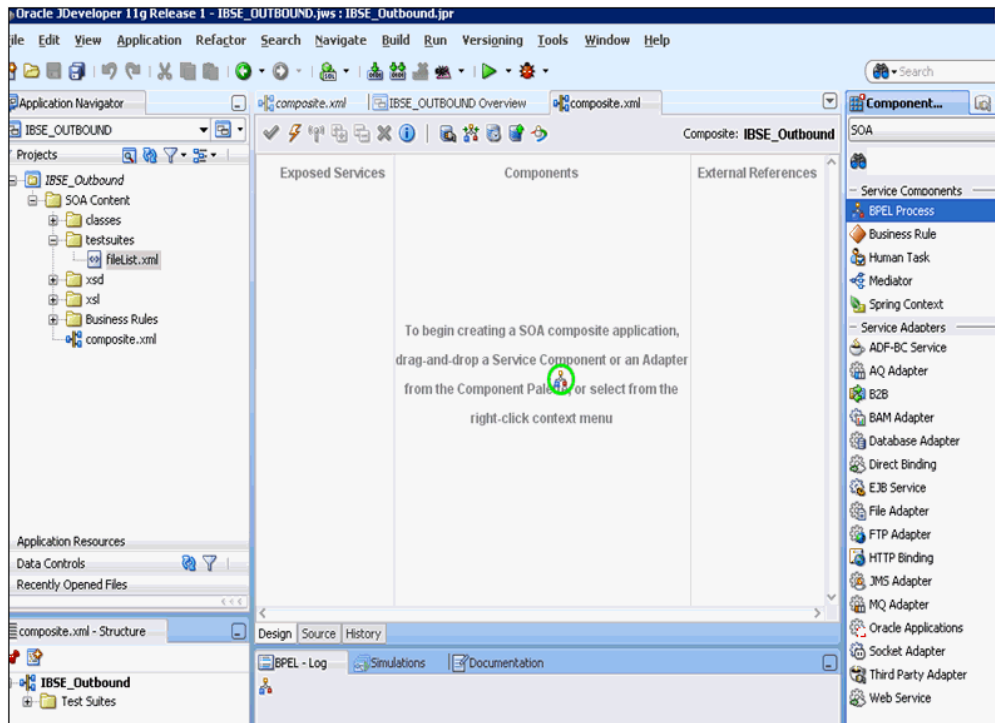
この節では、次のトピックについて説明します。

- [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手順は、次のとおりです。

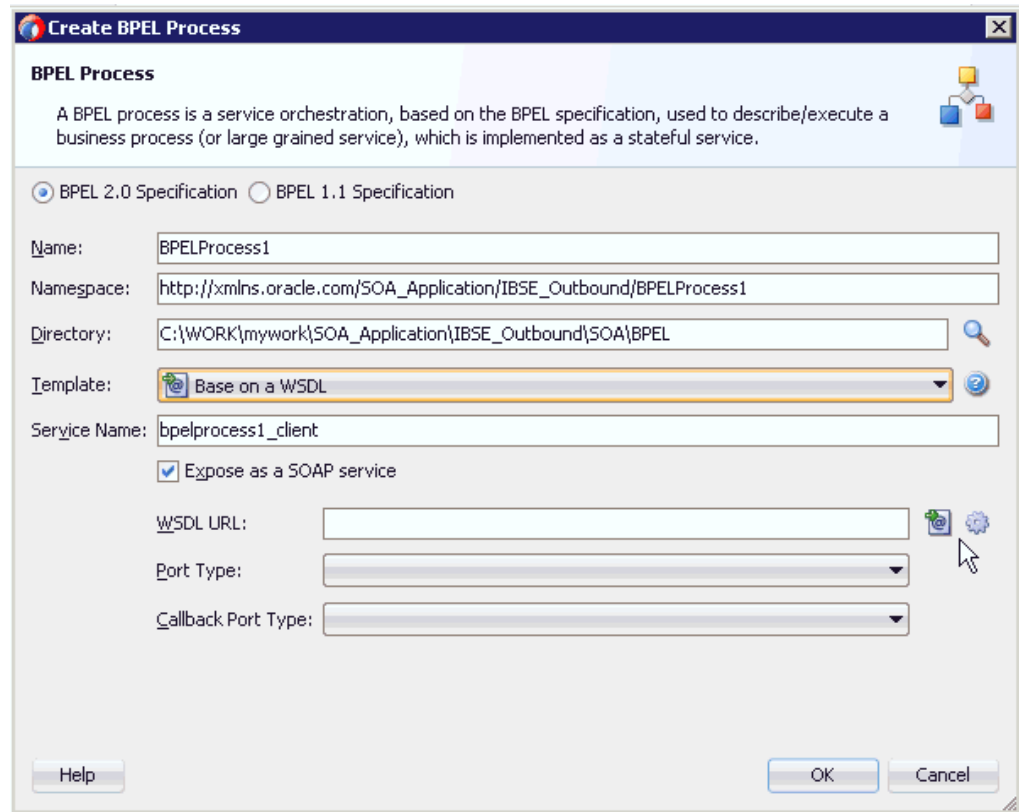
1. [図 6-82](#) に示すように、「サービス・コンポーネント」ペインから「コンポーネント」ペインに「BPEL プロセス」コンポーネントをドラッグ・アンド・ドロップします。

図 6-82 「BPEL プロセス」コンポーネント



2. 「名前」フィールドで、新規のアウトバウンド BPEL プロセス・コンポーネントの名前を入力します。
デフォルトで、「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・リストから「WSDL に基づく」を選択します。
4. 図 6-83 に示すように、「WSDL URL」フィールドの右の既存の WSDL の検索アイコンをクリックします。

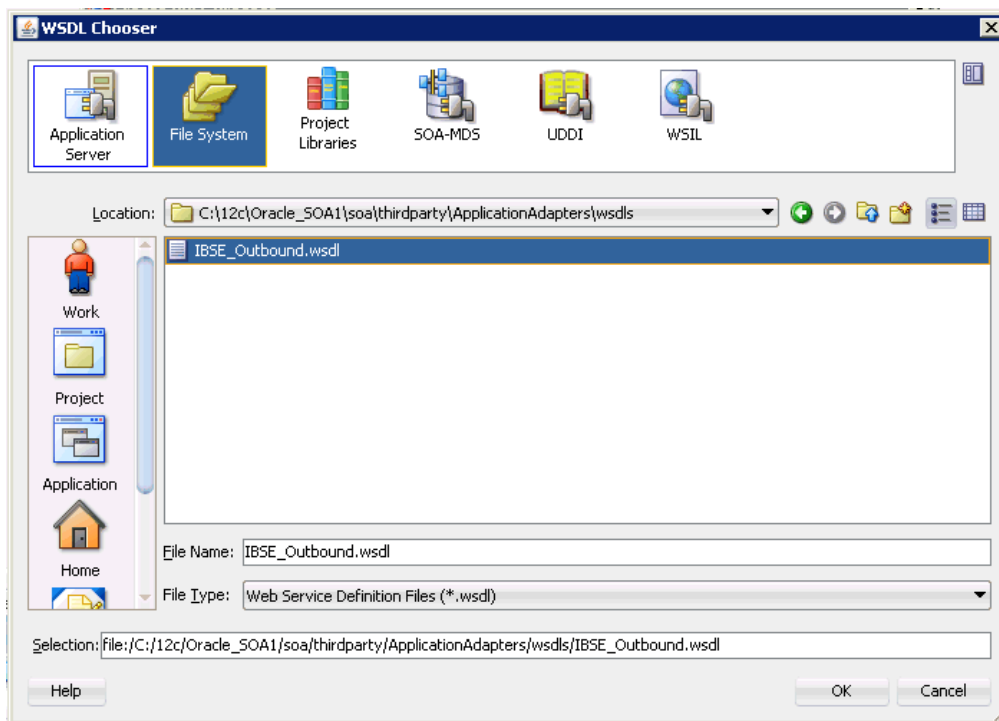
図 6-83 既存の WSDL の検索アイコン



「WSDL チューザ」ダイアログが表示されます。

5. 図 6-84 に示すように、WSDL がアプリケーション・エクスプローラからエクスポートされた場所にナビゲートして、WSDL を選択し、「OK」をクリックします。

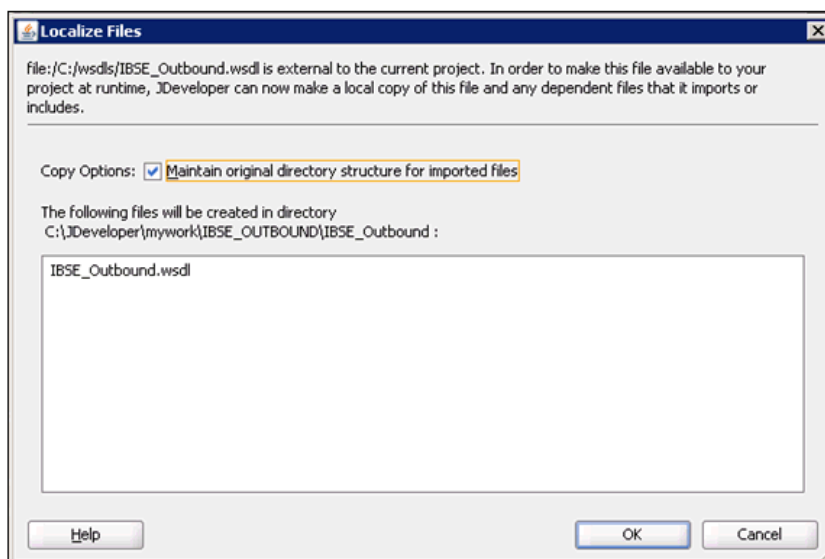
図 6-84 「WSDL チューザ」 ダイアログ



「ファイルのローカライズ」 ウィンドウが表示されます。

- 表示された「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。図 6-85 に示すように、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 6-85 「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。

- 図 6-86 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 6-86 「BPEL プロセス」 ペイン

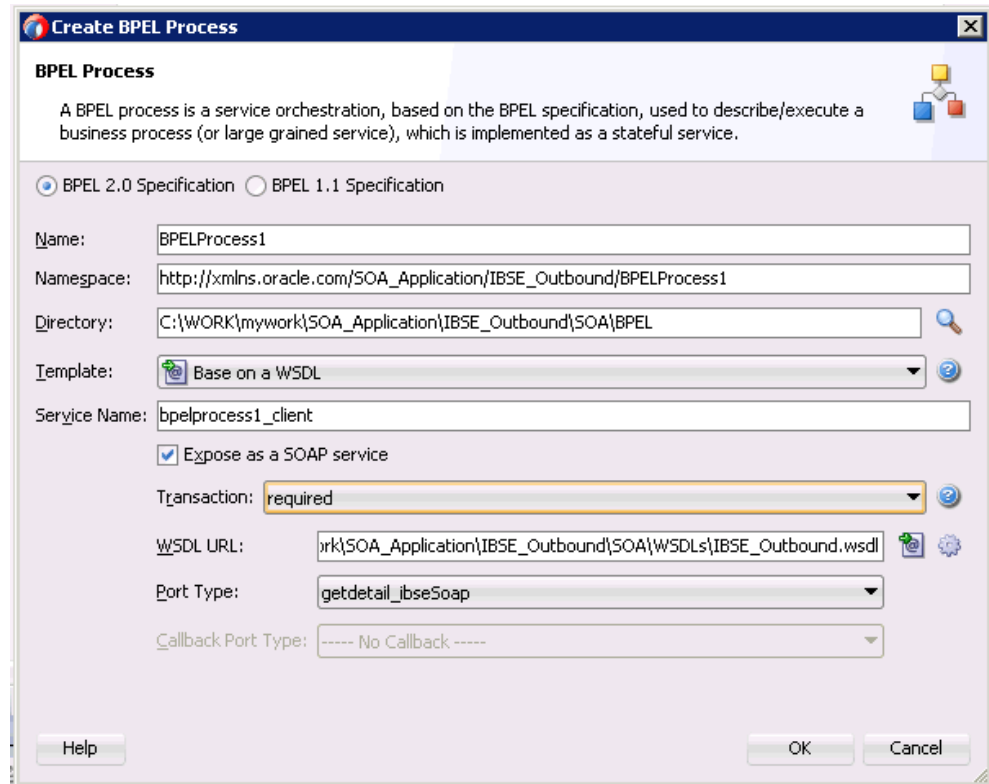
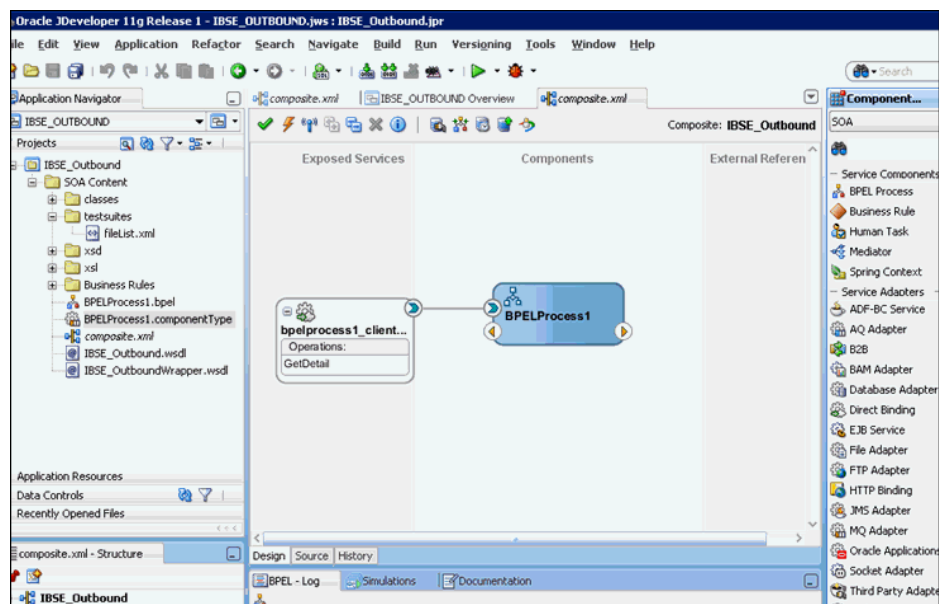


図 6-87 に示すように、「BPEL プロセス」コンポーネントが作成され、表示されます。

図 6-87 「BPEL プロセス」コンポーネント



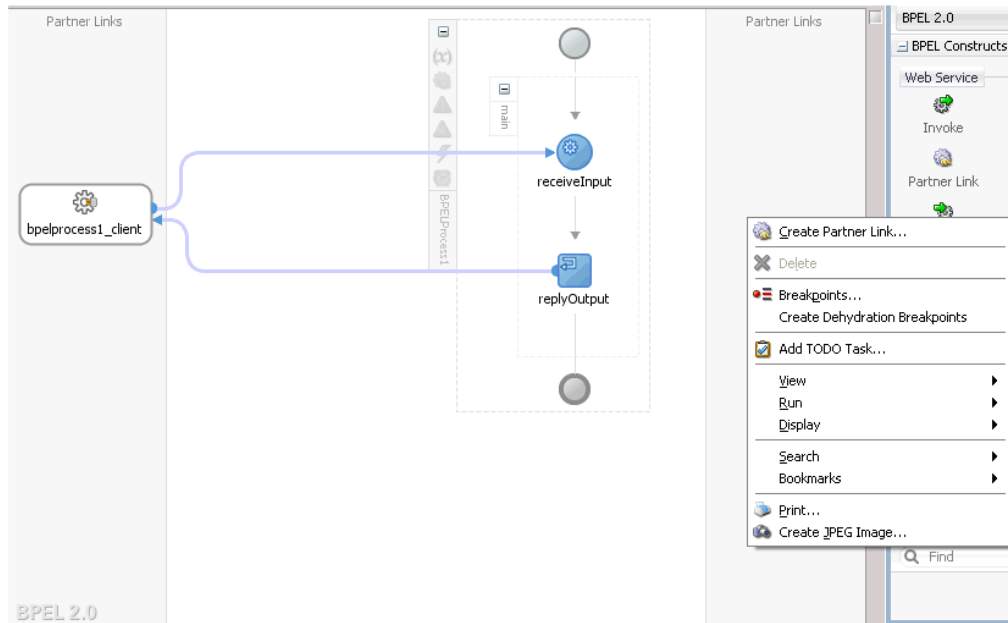
6.7.3.1 パートナ・リンクの作成

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・リンクを作成する手順は、次のとおりです。

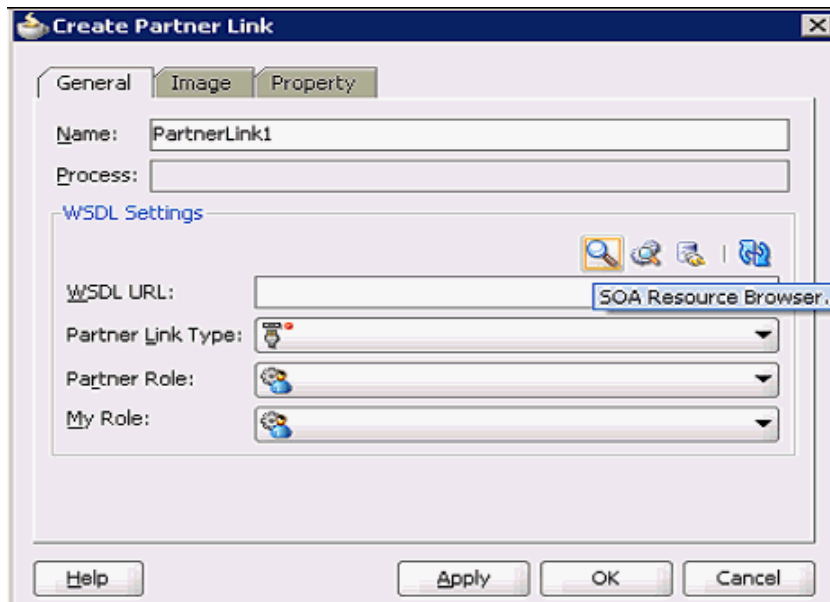
1. 「コンポーネント」ペインで、アウトバウンド BPEL プロセス・コンポーネントをダブルクリックします。
2. 図 6-88 に示すように、「パートナ・リンク」ペインを右クリックし、「パートナ・リンクの作成」を選択します。

図 6-88 パートナ・リンクの作成



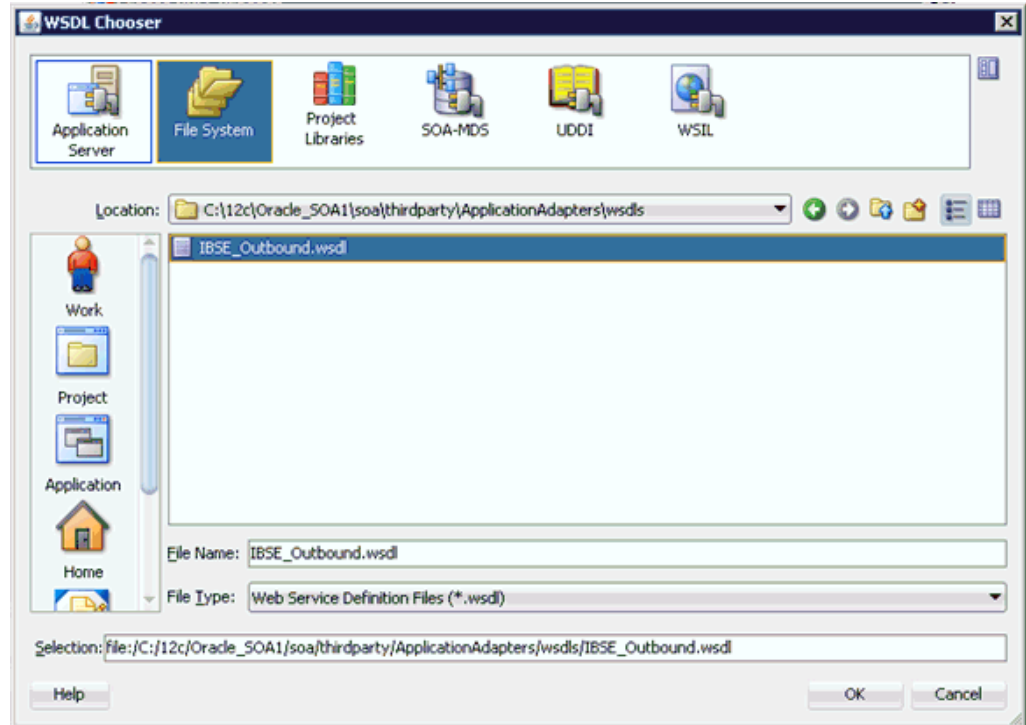
3. 図 6-89 に示すように、表示される「パートナ・リンクの作成」ウィンドウで、適切な名前を指定し、「SOA リソース・ブラウザ」ツールをクリックします。

図 6-89 「SOA リソース・ブラウザ」ツール



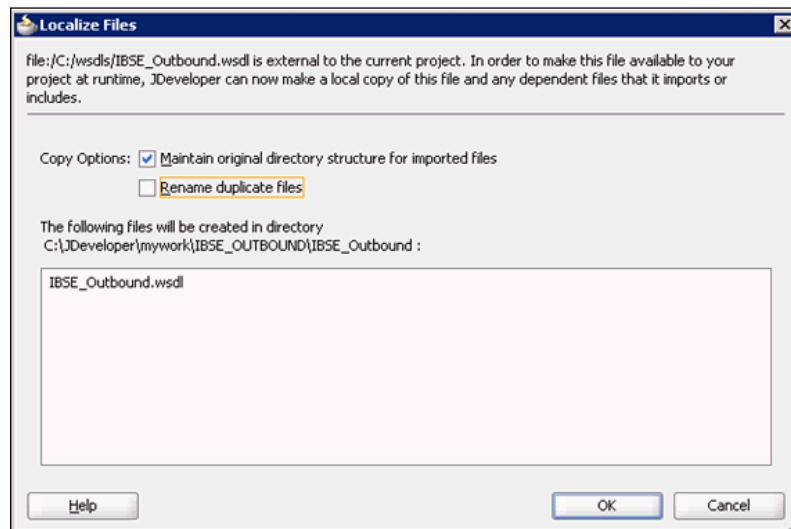
4. 図 6-90 に示すように、表示される「WSDL チューザ」ダイアログで、WSDL がアプリケーション・エクスプローラからエクスポートされた場所にナビゲートして、WSDL を選択し、「OK」をクリックします。

図 6-90 「WSDL チューザ」ダイアログ



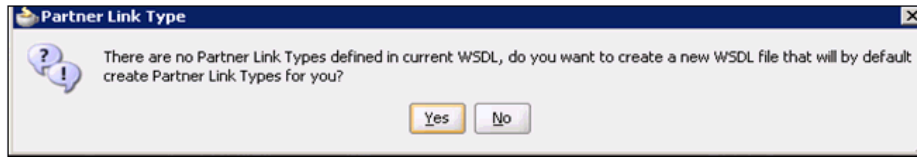
5. 図 6-91 に示すように、表示される「ファイルのローカライズ」ウィンドウで、「重複ファイルの名前変更」チェック・ボックスの選択を解除し、「OK」をクリックします。

図 6-91 「ファイルのローカライズ」ウィンドウ



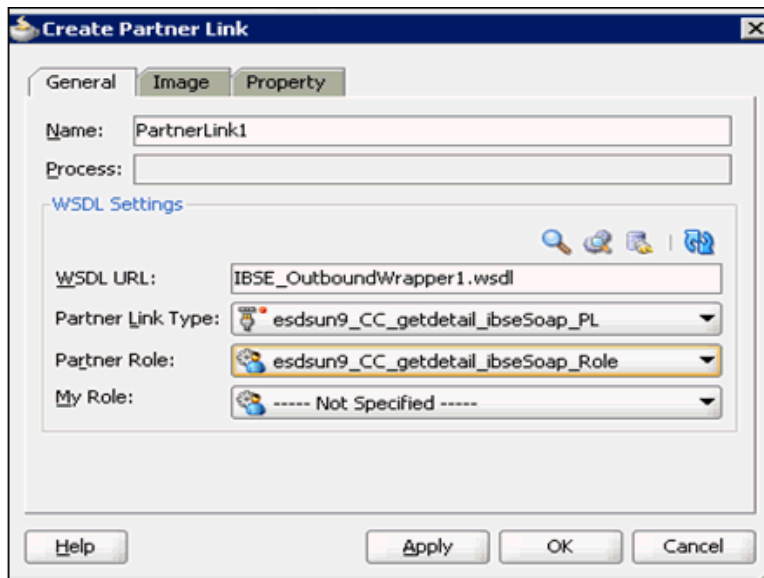
6. 図 6-92 に示すように、表示される「パートナ・リンク・タイプ」ウィンドウで「はい」をクリックします。

図 6-92 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、使用可能なパートナ・ロールを選択します。
8. 図 6-93 に示すように、「適用」、「OK」の順にクリックします。

図 6-93 パートナ・リンクの作成



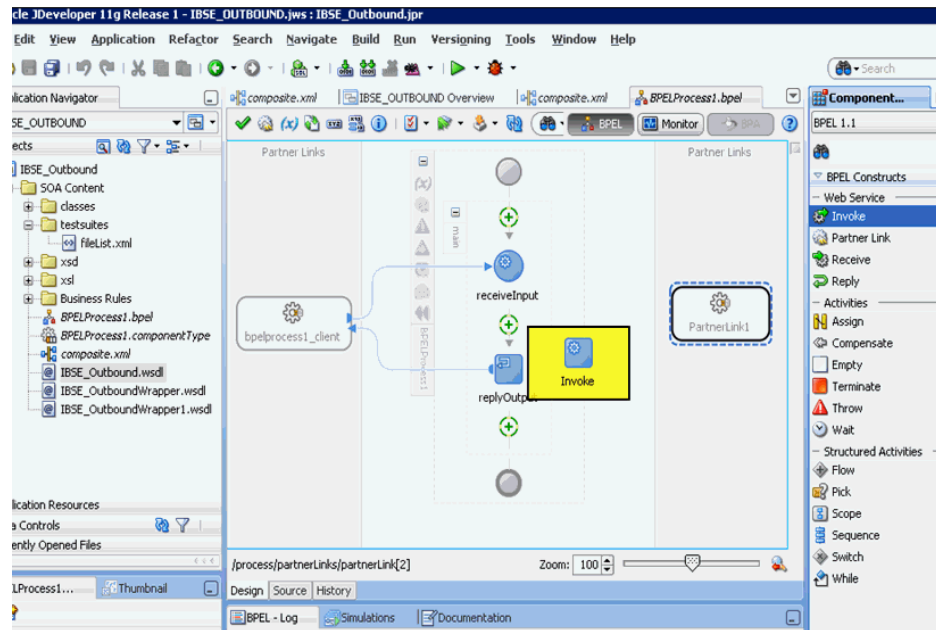
6.7.3.2 BPEL アクティビティの作成および作成済パートナ・リンクとのマッピング

この項では、BPEL アクティビティを作成し、作成済パートナ・リンクとマップ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、作成済パートナ・リンクとマップする手順は、次のとおりです。

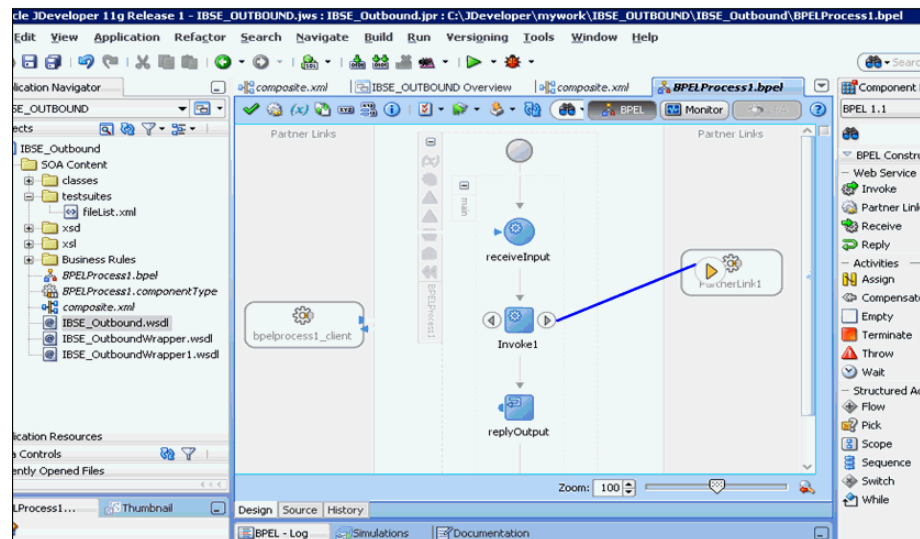
1. 「BPEL コンストラクト」から「コンポーネント」ペインに「起動」アクティビティ・コンポーネントをドラッグ・アンド・ドロップします。図 6-94 に示すように、これを「receiveInput」アクティビティ・コンポーネントと replyOutput アクティビティ・コンポーネントの間に置きます。

図 6-94 「起動」 アクティビティ・コンポーネント



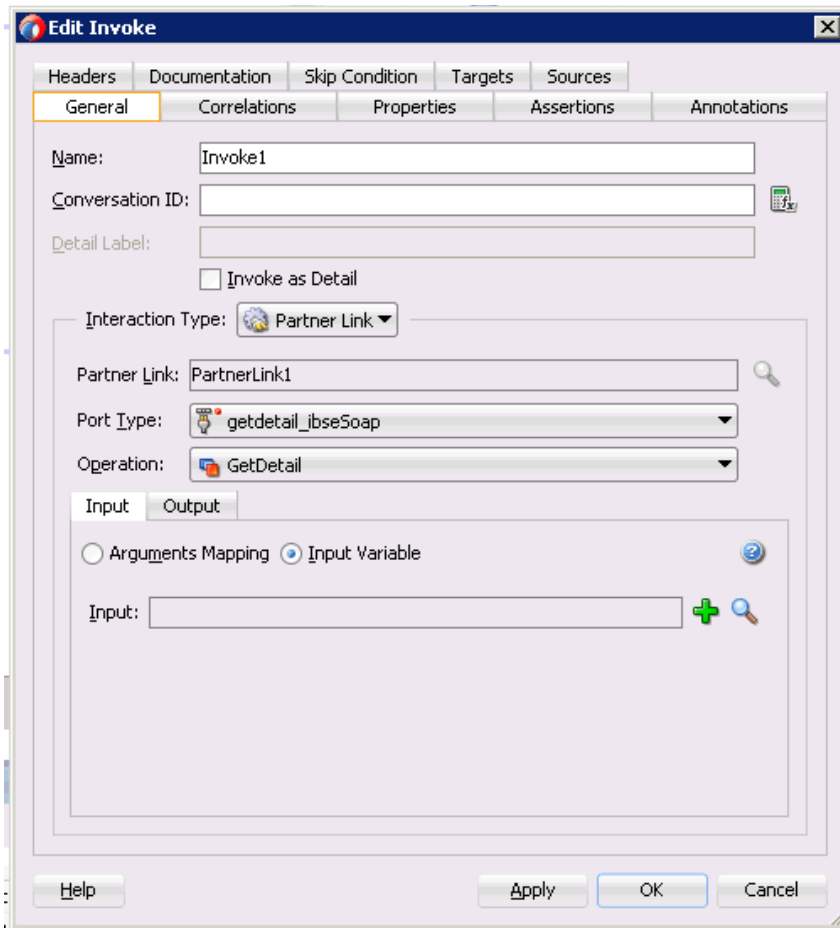
2. 図 6-95 に示すように、新規の「起動」アクティビティ・コンポーネント (Invoke1) と「パートナ・リンク」コンポーネント (Partner link1) の間に接続を作成します。

図 6-95 「パートナ・リンク」コンポーネント



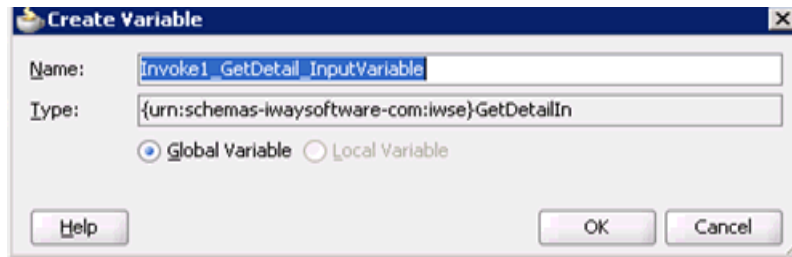
3. 図 6-96 に示すように、表示される起動の編集ウィンドウで、新規入力変数の構成用の「入力」フィールドの右のプラス (+) アイコンをクリックします。

図 6-96 起動の編集ウィンドウ



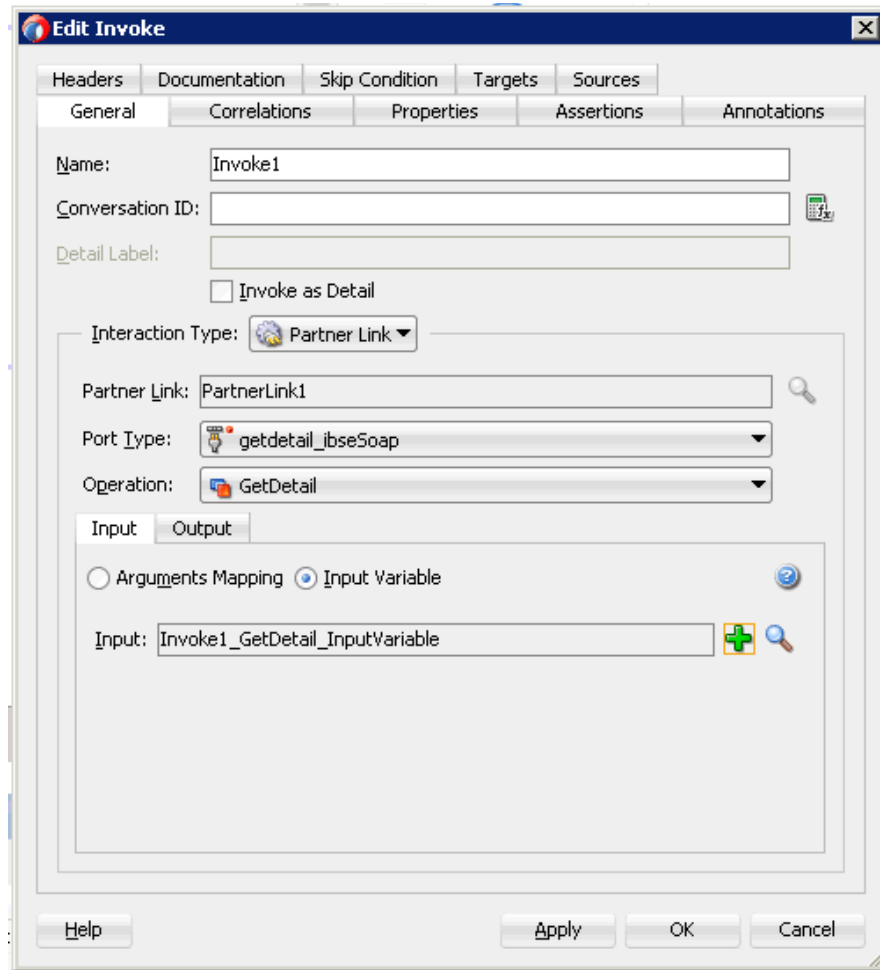
4. 図 6-97 に示すように、新規の入力変数に示されたデフォルト値を受け入れ、「OK」をクリックします。

図 6-97 新規の入力変数



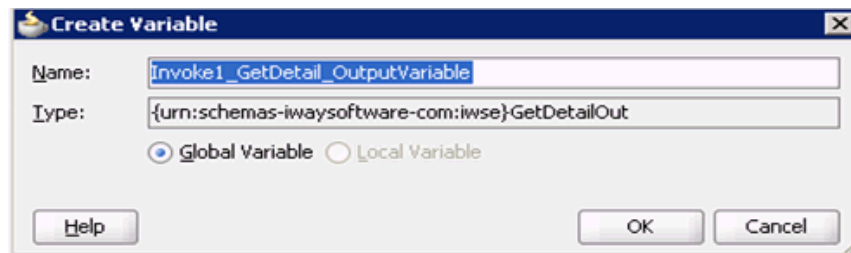
5. 図 6-98 に示すように、新規出力変数の構成用の「出力」フィールドの右のプラス (+) アイコンをクリックします。

図 6-98 起動の編集ウィンドウ



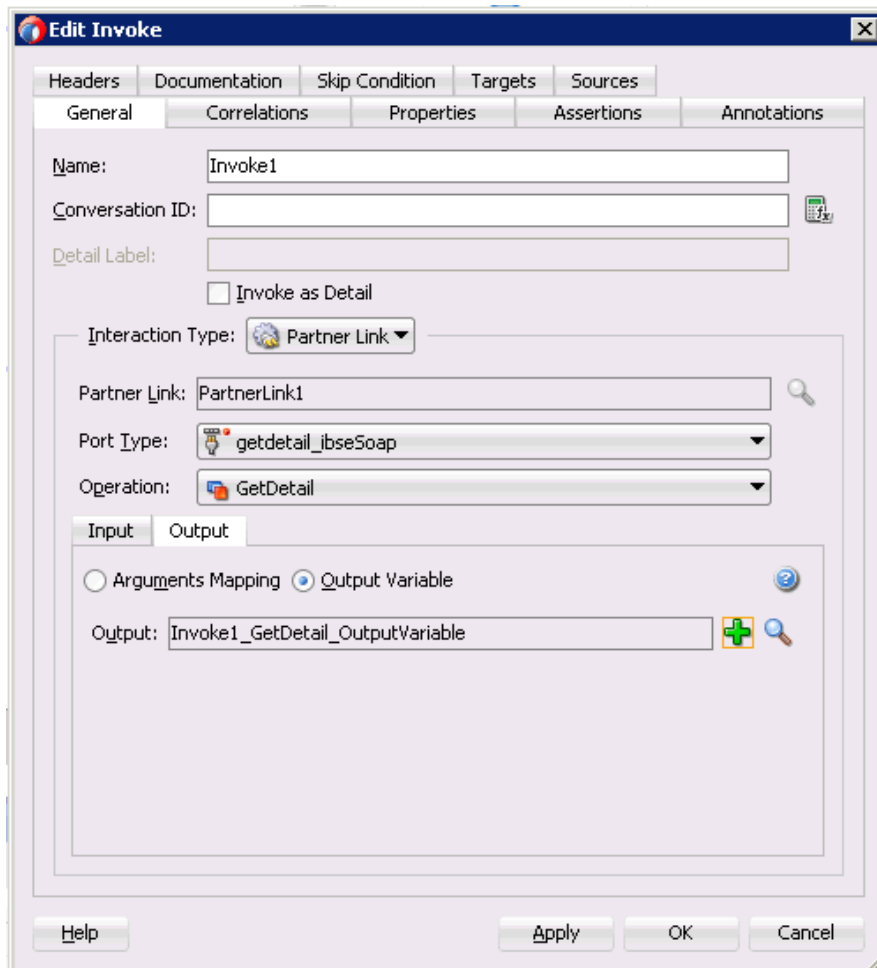
6. 図 6-99 に示すように、新規の出力変数に示されたデフォルト値を受け入れ、「OK」をクリックします。

図 6-99 新規の出力変数



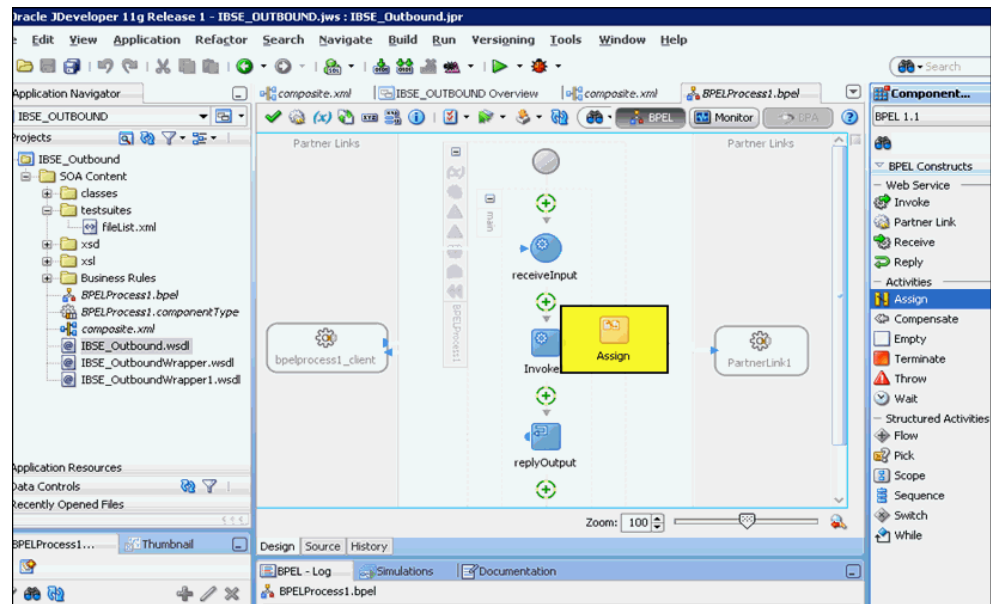
7. 図 6-100 に示すように、「適用」、「OK」の順にクリックします。

図 6-100 起動の編集ウィンドウ



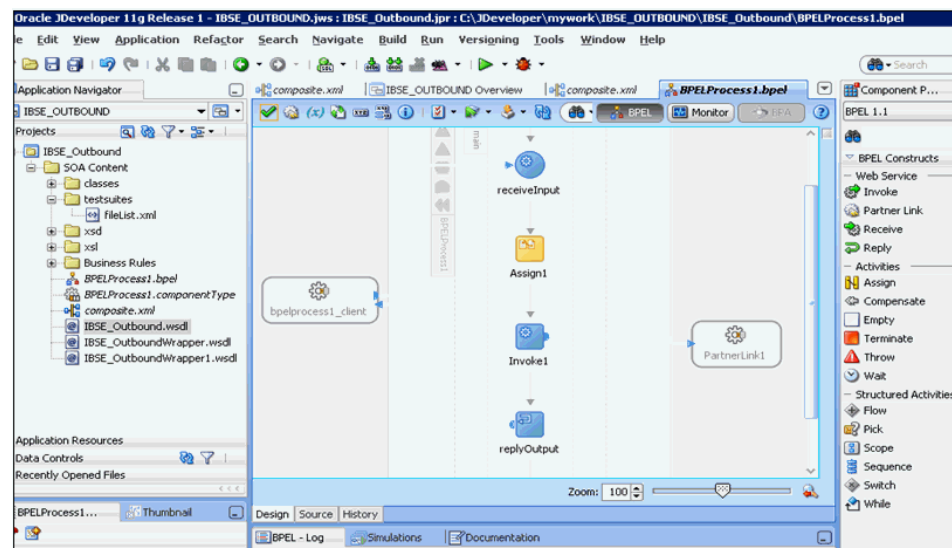
- 「BPEL コンストラクト」から「コンポーネント」ペインに「割当て」アクティビティ・コンポーネントをドラッグ・アンド・ドロップします。図 6-101 に示すように、これを「受信」アクティビティ・コンポーネント (receiveInput) と「起動」アクティビティ・コンポーネント (Invoke1) の間に置きます。

図 6-101 「割当て」 アクティビティ・コンポーネント



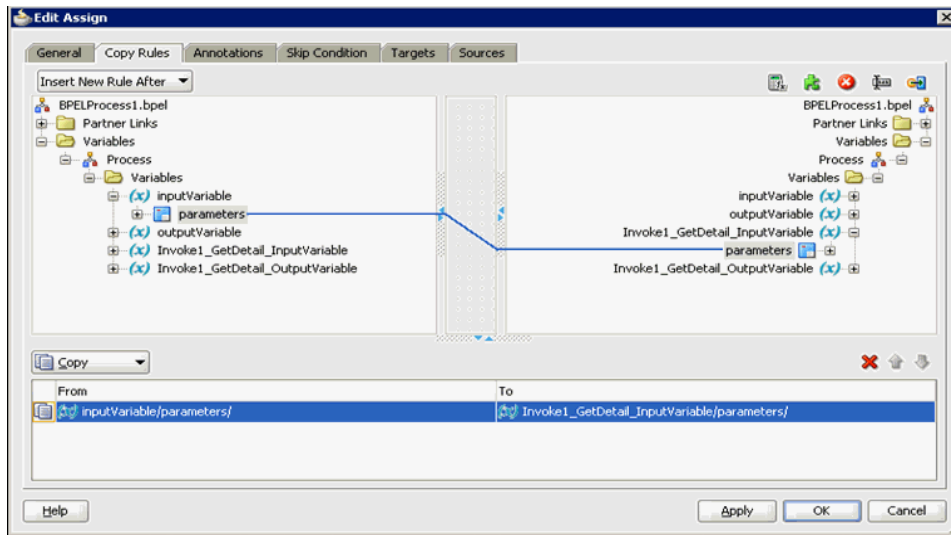
9. 図 6-102 に示すように、新規の「割当て」アクティビティ・コンポーネント (Assign1) をダブルクリックします。

図 6-102 「割当て」 アクティビティ・コンポーネント



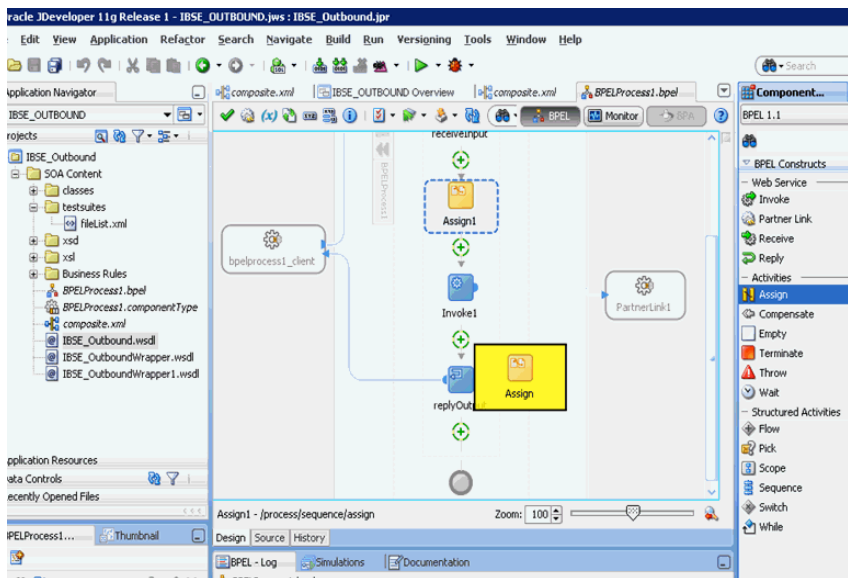
10. 左ペインで、「変数」の下の「InputVariable」を展開し、「parameters」を選択します。
11. 右ペインで、「変数」の下の「Invoke1_GetDetail_InputVariable」を展開し、「parameters」を選択します。
12. 図 6-103 に示すように、「InputVariable」パラメータを「Invoke1_GetDetail_InputVariable」パラメータにドラッグしてマップします。

図 6-103 「InputVariable」パラメータ



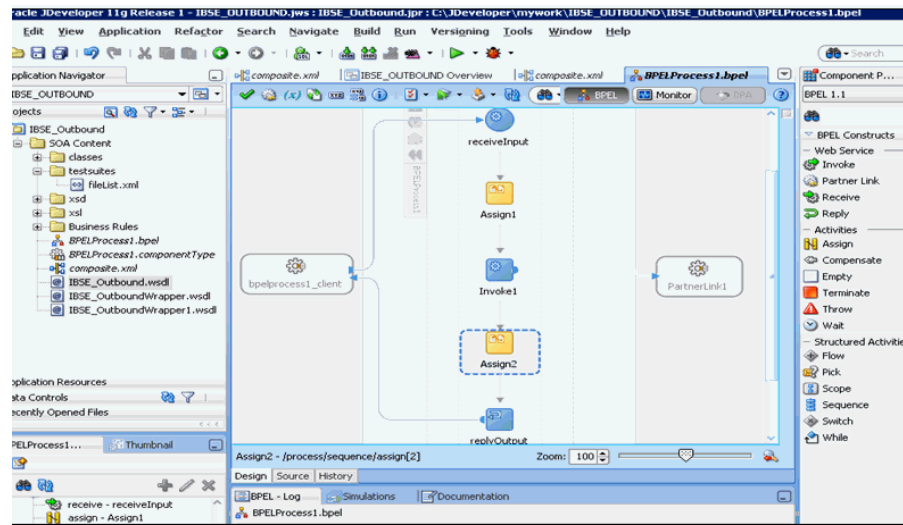
13. 「適用」、「OK」の順にクリックします。
14. 図 6-104 に示すように、「割当て」アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップし、「起動」アクティビティ (Invoke1) と「返信」アクティビティ (replyOutput) の間に置きます。

図 6-104 「割当て」アクティビティ・コンポーネント



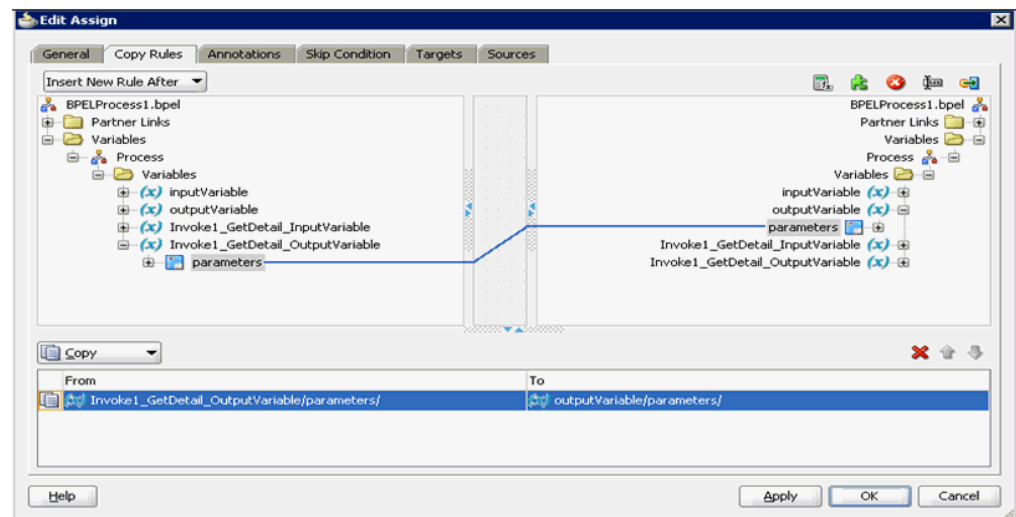
15. 図 6-105 に示すように、新規の「割当て」アクティビティ・コンポーネント (Assign2) をダブルクリックします。

図 6-105 新規の「割当て」アクティビティ・コンポーネント



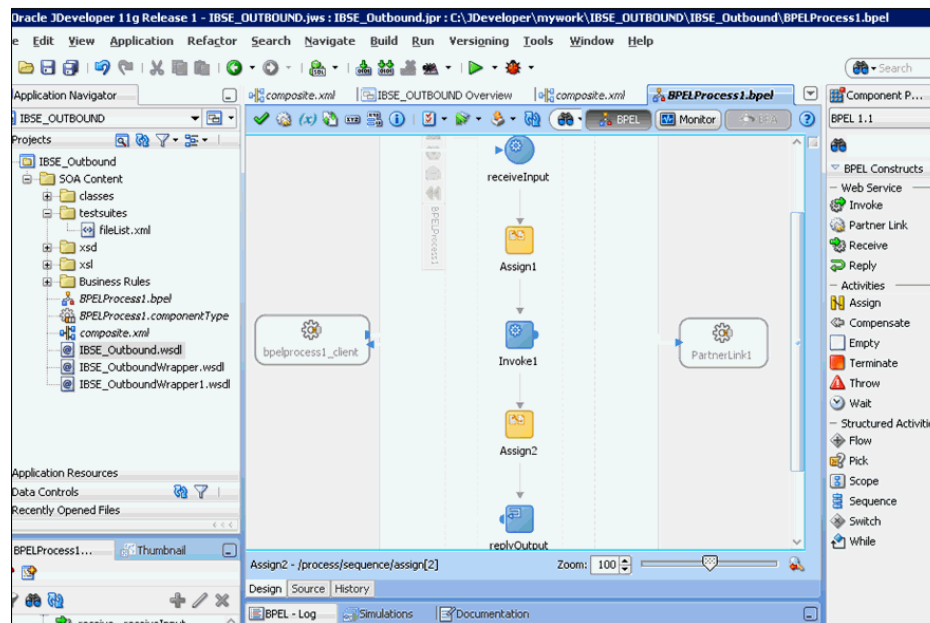
16. 左ペインで、「変数」の下の「Invoke1_GetDetail_OutputVariable」を展開し、「parameters」を選択します。
17. 右ペインで、「変数」の下の「outputVariable」を展開し、「parameters」を選択します。
18. 図 6-106 に示すように、「Invoke1_GetDetail_OutputVariable」パラメータを「outputVariable」パラメータにドラッグしてマップします。

図 6-106 「outputVariable」パラメータ



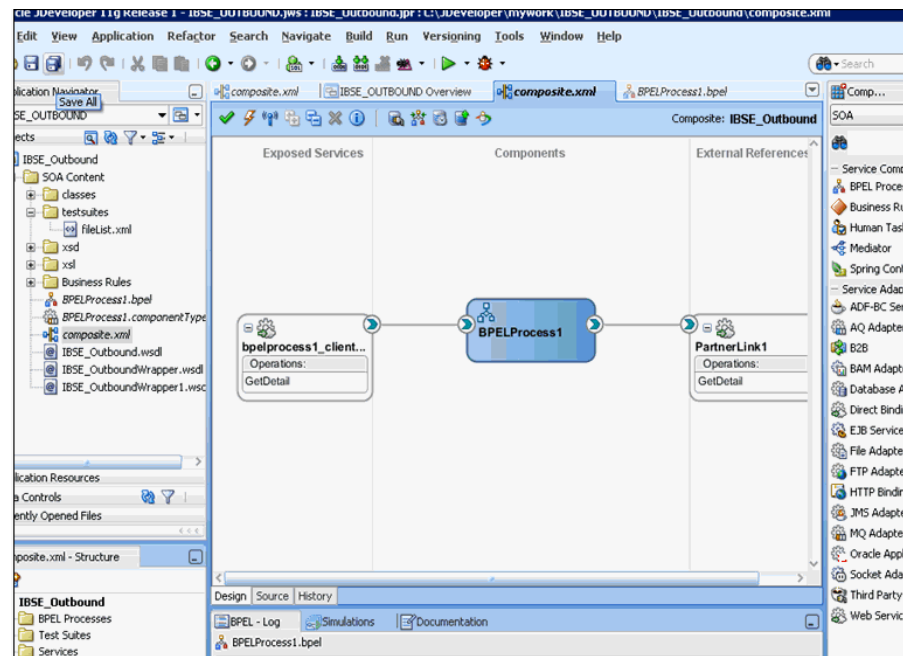
19. 「適用」、「OK」の順にクリックします。
 図 6-107 に示すように、コンポーネント・ペインに戻ります。

図 6-107 コンポーネント・ペイン



20. 図 6-108 に示すように、メニュー・バーで「すべて保存」アイコンをクリックし、構成された新規のアウトバウンド BPEL プロセス・コンポーネントを保存します。

図 6-108 「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」](#)と同じ手順に従うことができます。

デプロイされたら、[6.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動」](#)に定義されたように入力 XML を起動できます。

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。内容は次のとおりです。

- 7.1 項「新規のアプリケーション・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)」

この章で示すシナリオには、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 が 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 が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。詳細については、[付録 A 「リモート・プロセスにおける 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する必要があります。

- 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。詳細については、[第 4 章 「Oracle Application Adapter for SAP R/3 の構成」](#) を参照してください。
- Oracle JDeveloper を構成する方法。詳細については、[第 6 章 「Oracle SOA Suite での BPEL サービス・コンポーネントの統合」](#) 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・コードを使用せずに、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・ソリューションとは対照的に、機能的なモデリングにより、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、次の 2 ステップのプロセスです。

1. **デザインタイム**: 第4章「Oracle Application Adapter for SAP R/3 の構成」の説明に従って、サービスおよびイベント用にアプリケーション・エクスペローラで Oracle Application Adapter for SAP R/3 が構成されます。統合ロジックはメディエータでモデル化されます。
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 新規のアプリケーション・サーバー接続の構成

Oracle JDeveloper での新規のアプリケーション・サーバー接続の構成の詳細は、6.3 項「新規のアプリケーション・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 システムにメディエータ・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。

Application Adapters インストールの次のフォルダに、このアウトバウンドの使用ケース・シナリオのサンプル・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
```

この節では、次のトピック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 ドキュメントの起動」

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。詳細については、6-8 ページ「リクエスト/レスポンス型のサービス用の 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 用に空のコンポジットを作成するには、以下の手順を実行します。

1. 新規の SOA アプリケーションを作成します。
2. 新規の SOA アプリケーションに名前を入力し、「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。

詳細については、6-9 ページ 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 メディアータ・アウトバウンド・プロセスの定義

この項では、次のトピックで構成されるメディアータ・アウトバウンド・プロセスの定義方法について説明します。

- 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・メディアータ・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 との既知のデプロイメント問題の調整」

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、次のステップを実行します。

1. 「サービス・アダプタ」ペインから「外部参照」ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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するか、デフォルト名をそのまま使用します。
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」フィールドの右の**既存の WSDL の検索**アイコンをクリックします。
5. 次のディレクトリまでブラウズし、アウトバウンド WSDL ファイルを選択します。

```
<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リクエスト/レスポンス XML スキーマ・ファイル (.xsd) が、作成済のプロジェクト・フォルダにインポートされます。

8. 「JCA ファイル」フィールドの右の**JCA ファイルの検索**アイコンをクリックします。
9. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。

```
<ADAPTER_HOME>\wsdls
```

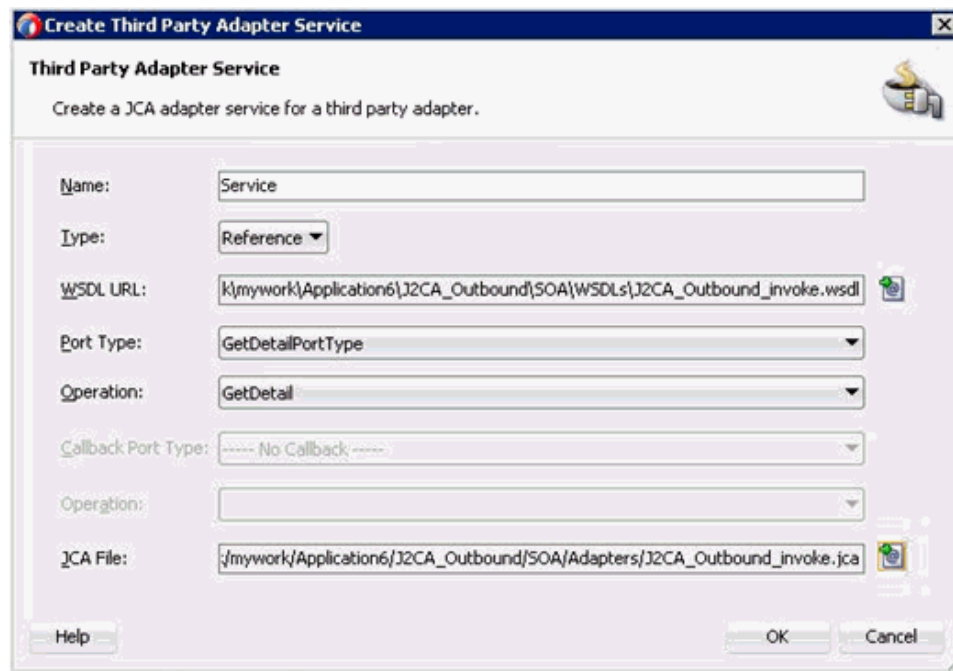
10. 「OK」をクリックします。

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。

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

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。

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・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。

詳細については、6-11 ページ [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。

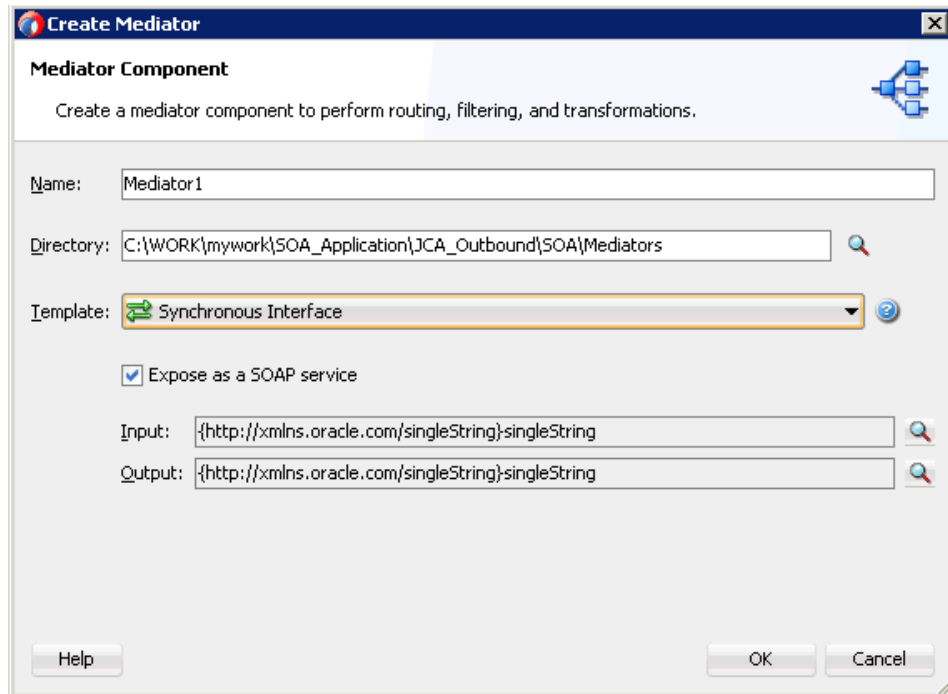
7.2.2.2 アウトバウンド・メディエータ・プロセス・コンポーネントの構成

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、次のステップを実行します。

1. 「コンポーネント」ペインから「コンポーネント」ペインにメディエータ・プロセス・コンポーネントをドラッグ・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。

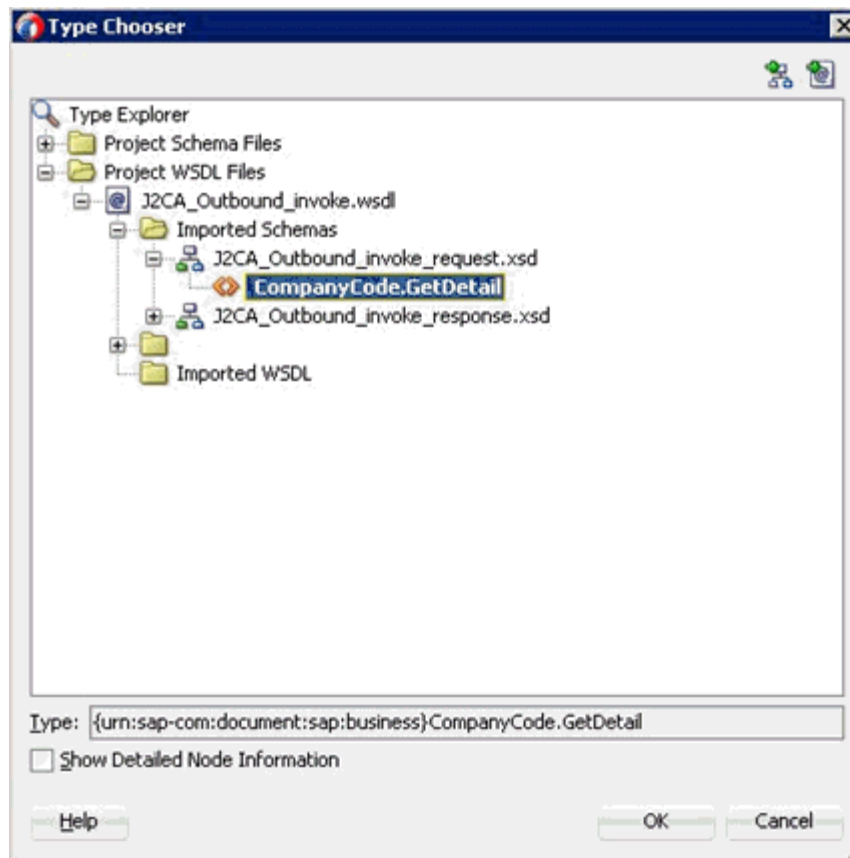
図 7-2 「メディエータの作成」ダイアログ



2. 「名前」フィールドで、新規のアウトバウンド・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・リストから「同期インタフェース」を選択します。
4. 関連する XML リクエスト・スキーマ・ファイルの選択用の「入力」フィールドの右の「参照」アイコンをクリックします。

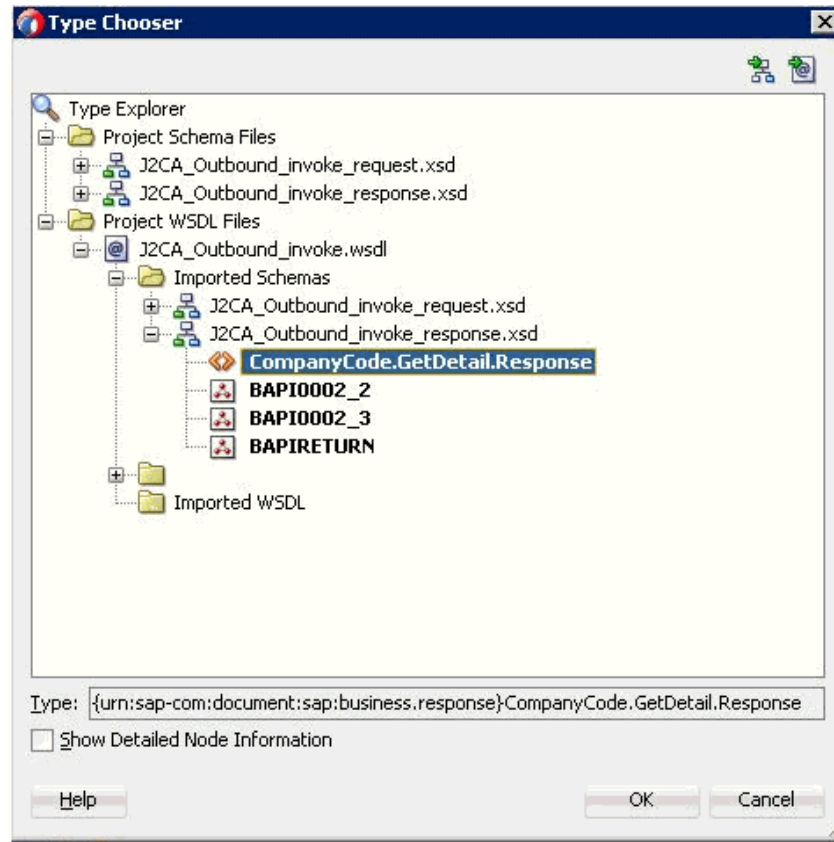
図 7-3 に示すように、「タイプ・チューザ」ダイアログが表示されます。

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」、「J2CA_Outbound_invoke.wSDL」、「インポートしたスキーマ」、「J2CA_Outbound_invoke_request.xsd」の順に展開し、「CompanyCode.GetDetail」を選択します。
6. 「OK」をクリックします。
「メディエータの作成」ダイアログに戻ります。
7. 関連する XML レスポンス・スキーマ・ファイルの選択用の「出力」フィールドの右の「参照」アイコンをクリックします。
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

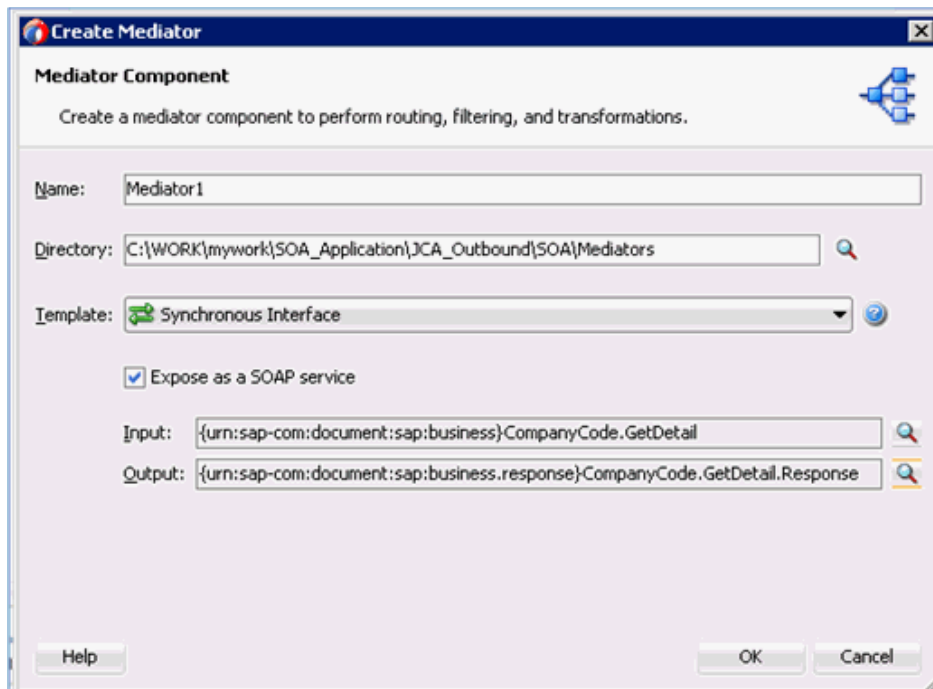


8. 「プロジェクトの WSDL ファイル」、「J2CA_Outbound_invoke.wsdl」、「インポートしたスキーマ」、「J2CA_Outbound_invoke_response.xsd」の順に展開し、「CompanyCode.GetDetailResponse」を選択します。

9. 「OK」をクリックします。

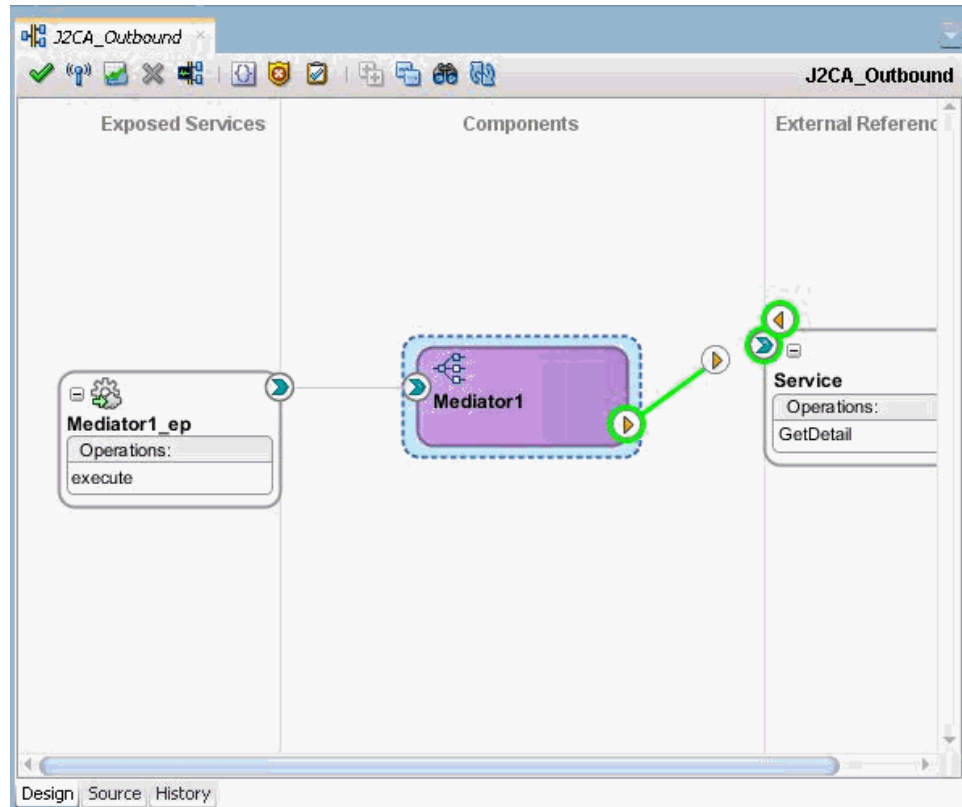
図 7-5 に示すように、「メディアータの作成」ダイアログに戻ります。

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。

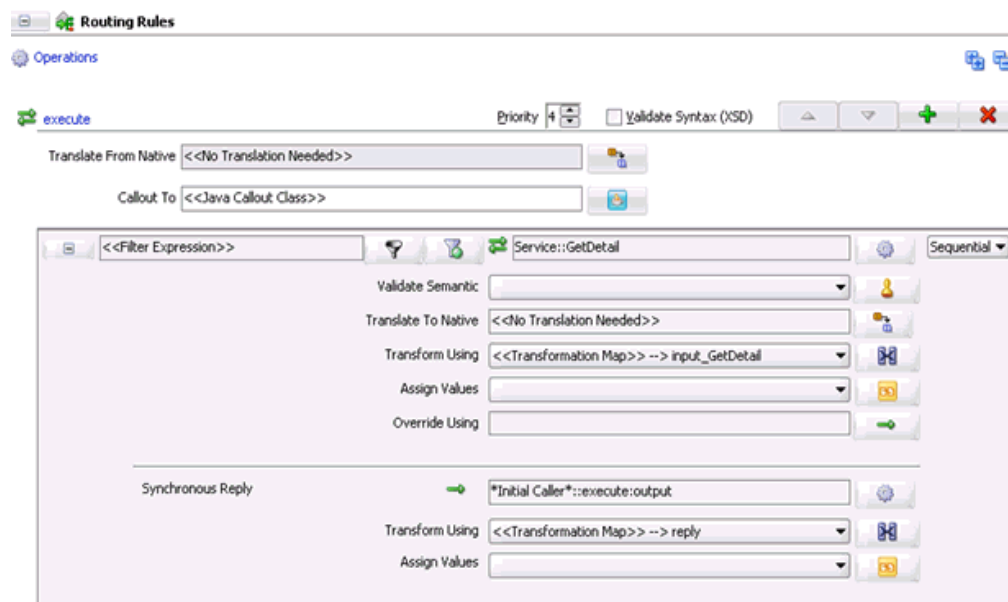
7.2.2.3 ルーティング・ルールの構成

メディエータ・アウトバウンド・プロセス・コンポーネントのルーティング・ルールを構成するには、次のステップを実行します。

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」フィールドの右のアイコンをクリックします。

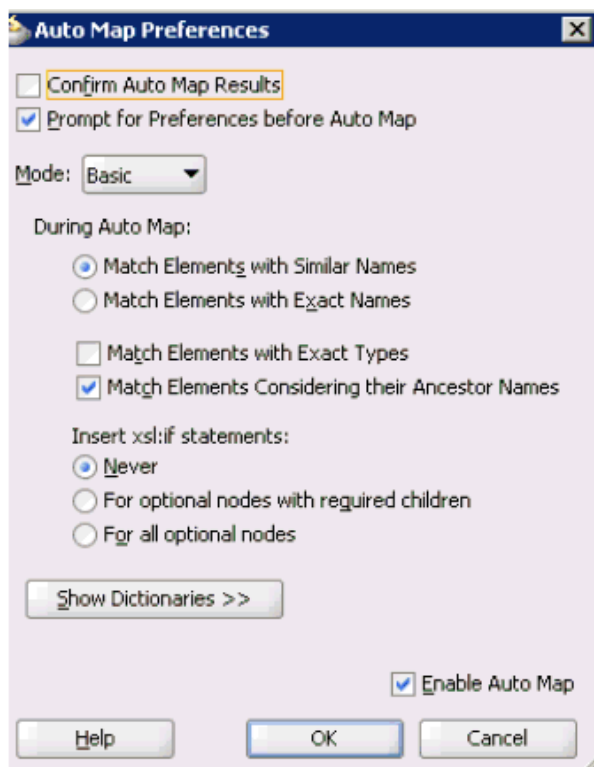
図 7-8 に示すように、「リクエスト・トランスフォーメーション・マップ」ダイアログが表示されます。

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



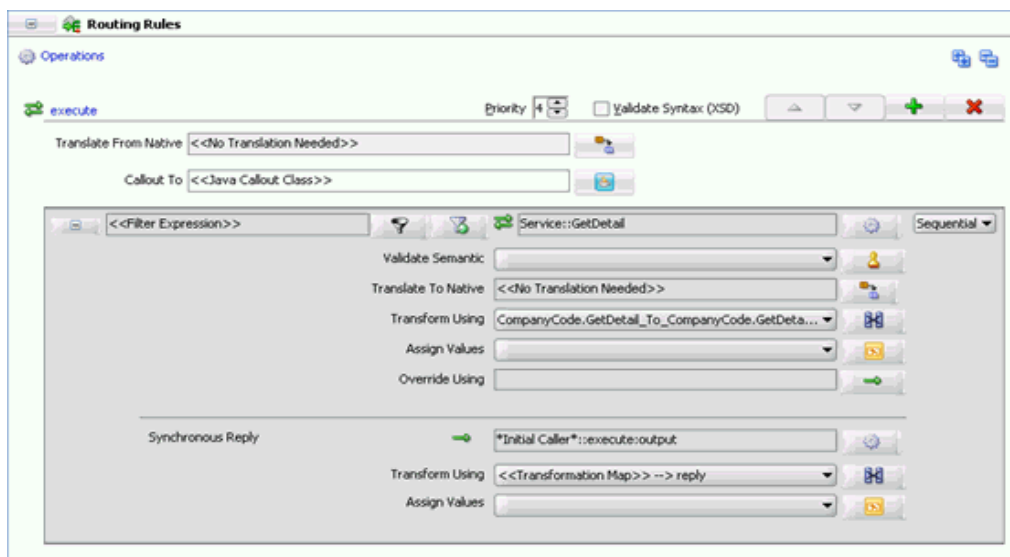
3. 「追加」 (+) アイコンをクリックします。
「トランスフォーメーション・マップの作成」ページが表示されます。
4. 「タイプ」が「XSLT」に選択されていることを確認し、「OK」をクリックします。
5. 「OK」をクリックします。
6. 「ns0:CompanyCode.GetDetail.Response」ソース要素を
「ns0:CompanyCode.GetDetail.Response」ターゲット要素にマッピングします。
図 7-9 に示すように、「自動マップ・プリファレンス」ダイアログが表示されます。

図 7-9 「自動マップ・プリファレンス」ダイアログ



7. デフォルト値を維持し、「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。

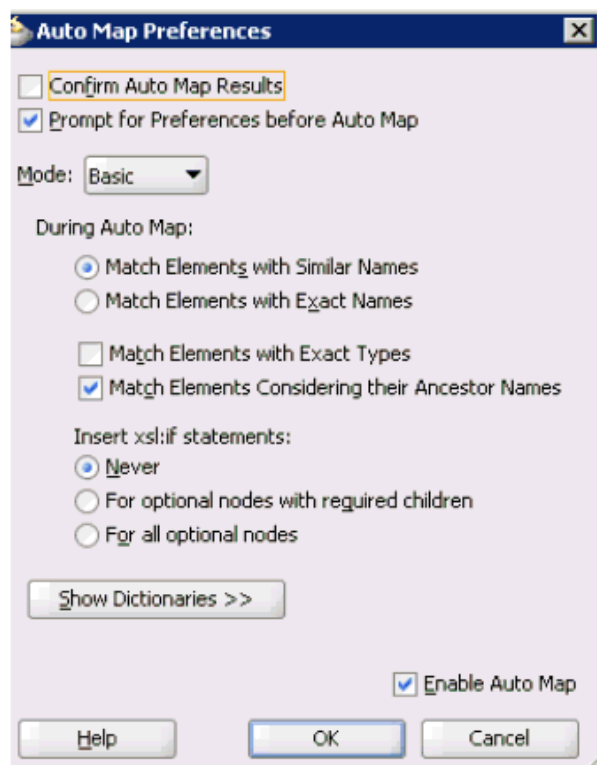
図 7-10 「ルーティング・ルール」ダイアログ



9. 「同期リプライ」領域で、「次を使用して変換」の右側の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。
10. 「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。

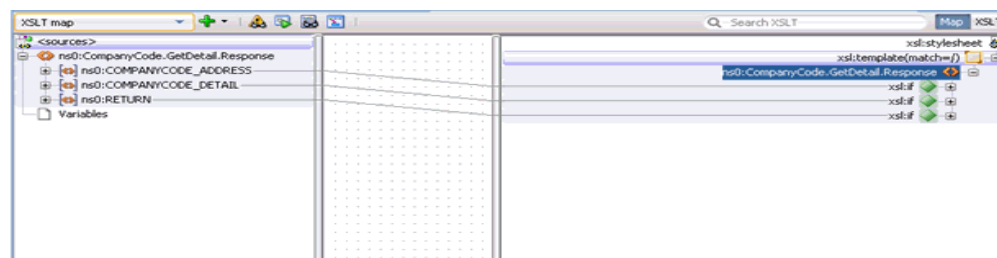
11. 「タイプ」が「XSLT」に選択されていることを確認し、「OK」をクリックします。
マッピング・ページが表示されます。
12. 「OK」をクリックします。
13. 「ns0:CompanyCode.GetDetail.Response」ソース要素を
「ns0:CompanyCode.GetDetail.Response」ターゲット要素にマッピングします。
図 7-11 に示すように、「自動マップ・プリファレンス」ダイアログが表示されま
す。

図 7-11 「自動マップ・プリファレンス」ダイアログ



14. デフォルト値を維持し、「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す。

図 7-12 完了したマッピング



15. メニュー・バーで「すべて保存」アイコンをクリックし、構成された新規のアウトバウンド・メディアータ・プロセス・コンポーネントを保存します。
これで、メディアータ・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 との既知のデプロイメント問題の調整

12c での既知のデプロイメント問題の調整方法については、6-27 ページ [6.4.3.3 項「12c との既知のデプロイメント問題の調整」](#) 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、次のステップを実行します

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、「**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したままにし、「**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. ターゲットの 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. プロジェクトに使用可能なすべてのデプロイメント情報をレビューおよび確認し、「**終了**」をクリックします。
詳細については、6-30 ページ [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 ドキュメントの起動

詳細については、6-33 ページ [6.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動」](#) 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。

Application Adapters インストールの次のフォルダに、このインバウンドの使用ケース・シナリオのサンプル・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
```

この節では、次のトピック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

メディアータ・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・エクステンション・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。詳細については、6-37 ページ「[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 用に空のコンポジットを作成するには、以下の手順を実行します。

1. 新規の SOA アプリケーションを作成します。
2. 新規の SOA アプリケーションに名前を入力し、「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。

詳細については、6-9 ページ [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 メディアータ・インバウンド・プロセスの定義

この項では、次のトピックで構成されるメディアータ・インバウンド・プロセスの定義方法について説明します。

- [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・メディアータ・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 との既知のデプロイメント問題の調整」](#)

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、次のステップを実行します。

1. 「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「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドの右の既存の WSDL の検索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. 次のディレクトリまでブラウズし、インバウンド WSDL ファイルを選択します。
<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 ファイルおよび関連リクエスト/レスポンス・スキーマ・ファイル (.xsd) が、作成済のプロジェクト・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」フィールドの右の **JCA ファイルの検索** 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. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。

<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。

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

「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。

サード・パーティ・アダプタ・サービス・コンポーネントが「公開されたサービス」ペインに作成されます。

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

詳細については、6-45 ページ [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、次のステップを実行します

1. 「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 **メディエータ・プロセス・コンポーネント** をドラッグ・アンド・ドロップします。


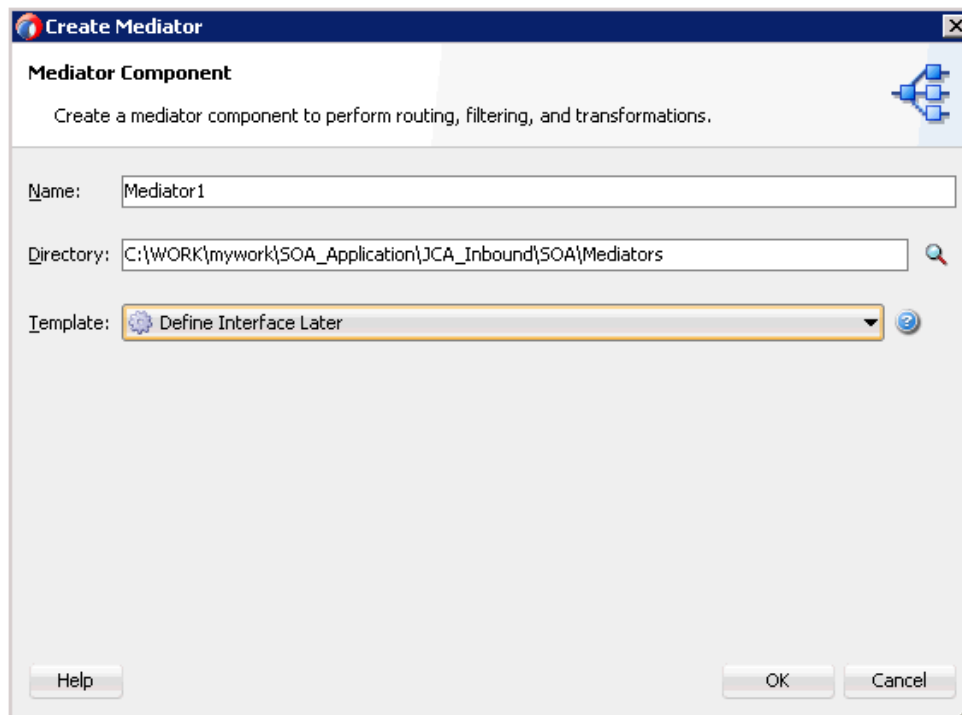
 [7-13](#) に示すように、「メディエータの作成」ダイアログが表示されます。

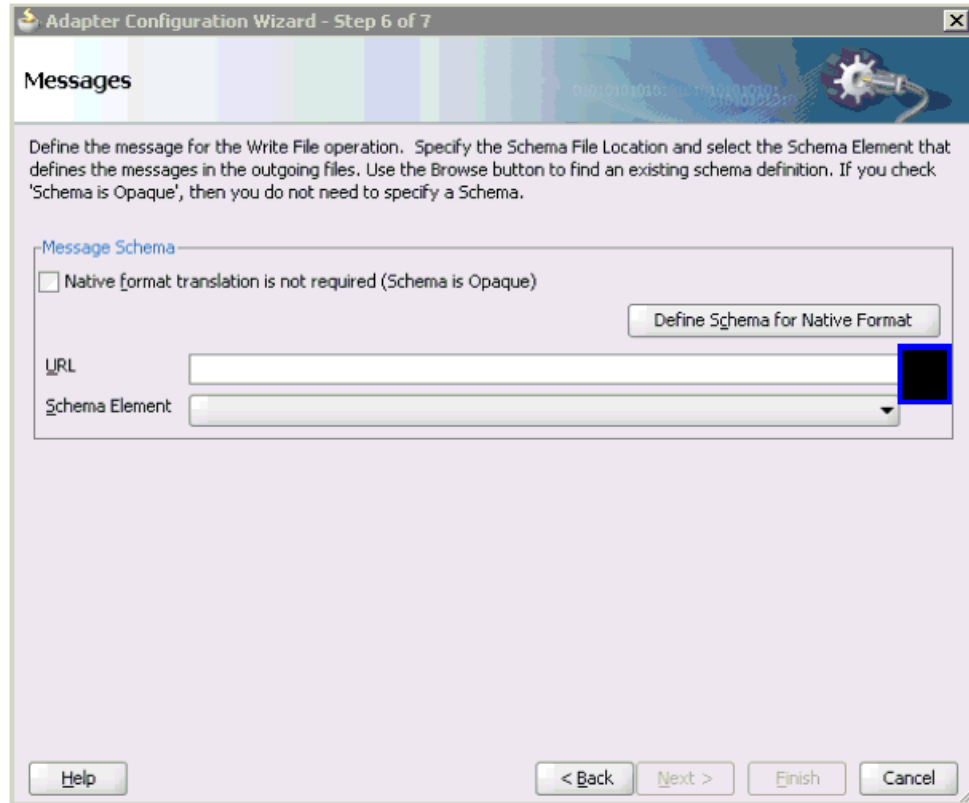
図 7-13 「メディエータの作成」ダイアログ



2. 「名前」フィールドで、新規のインバウンド・メディエータ・プロセス・コンポーネントの名前を入力します。
3. 「テンプレート」ドロップダウン・リストから「**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。
新規の「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. 「テクノロジー・アダプタ」ペインから「外部参照」ペインに「**ファイル**」コンポーネントをドラッグ・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. 新規の**ファイル・アダプタ**に名前を入力し、「**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ます。
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から「**Write File**」を選択し、「**操作名**」(「**Write**」など)を指定します。
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. 出力ファイルの書き込み先となる**ファイル・システム上の場所**を指定します。

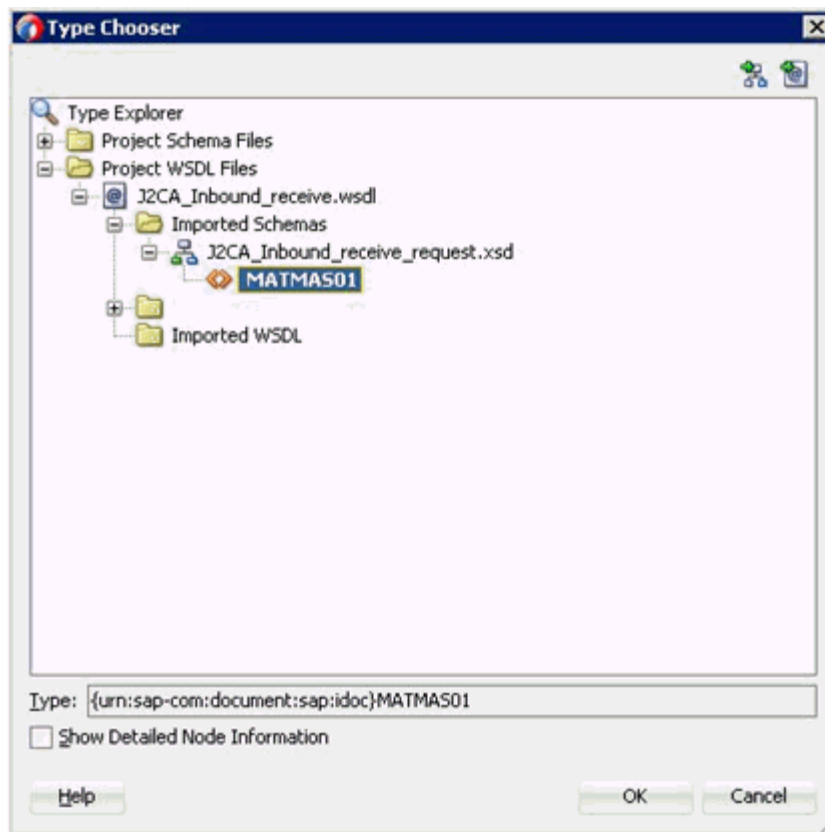
13. ファイル・ネーミング規則フィールドで、出力ファイルの名前を指定します。
14. 「次へ」をクリックします。
☒ 7-14 に示すように、「メッセージ」ページが表示されます。

図 7-14 「メッセージ」ページ



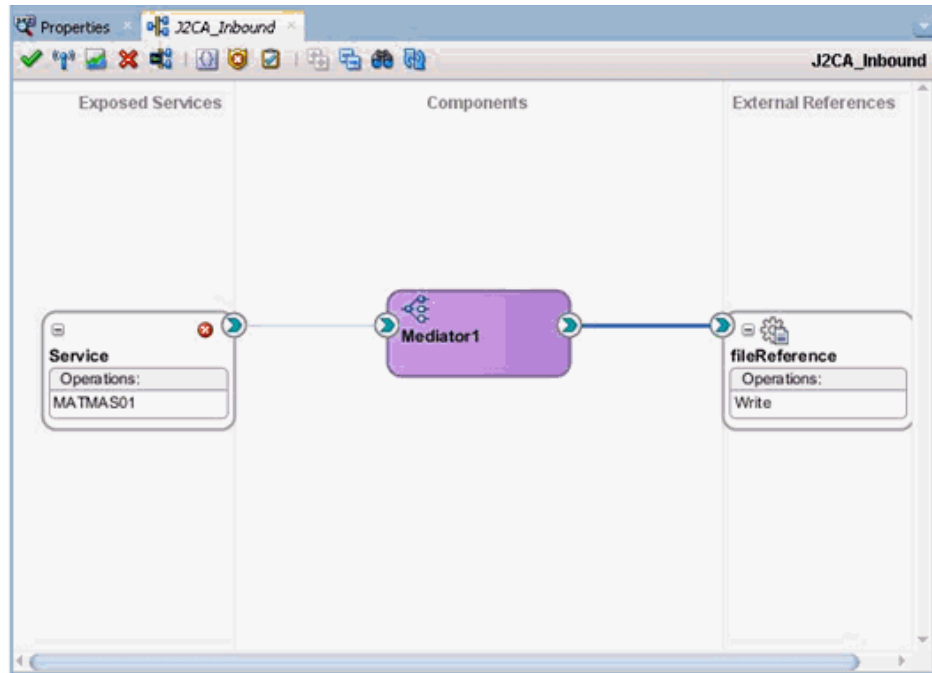
15. 「URL」フィールドの右の「参照」をクリックします。
☒ 7-15 に示すように、「タイプ・チューザ」ダイアログが表示されます。

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、「J2CA_Inbound_receive.wSDL」、インポート済スキーマ、「J2CA_Inbound_receive_request.xsd」の順に展開し、「MATMAS01」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・メディアータ・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・メディアータ・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。

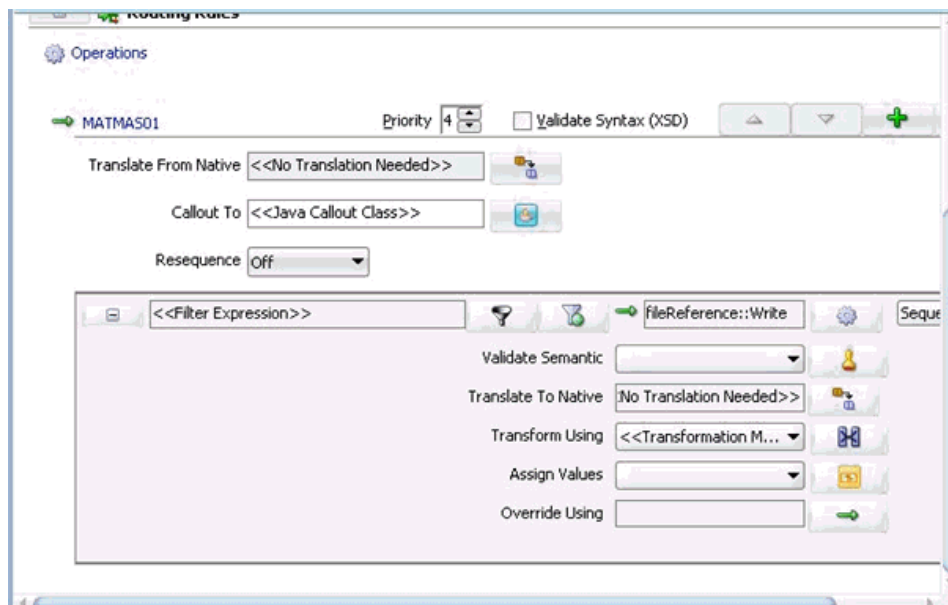
7.3.2.3 ルーティング・ルールの構成

メディエータ・インバウンド・プロセス・コンポーネントのルーティング・ルールを構成するには、次のステップを実行します。

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。

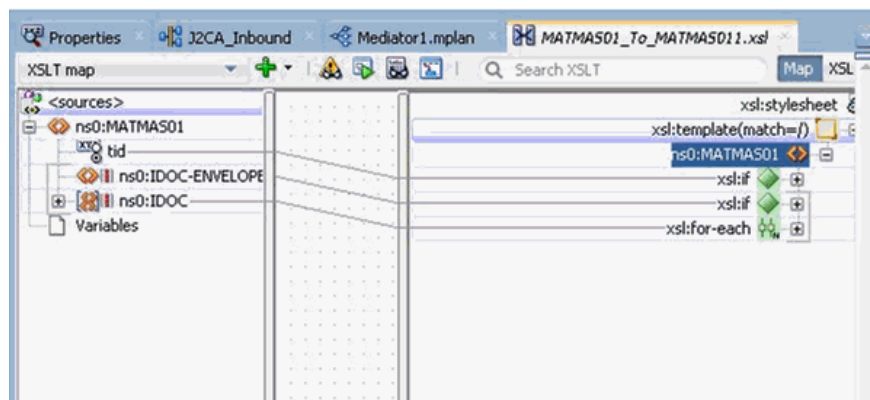
図 7-17 に示すように、「ルーティング・ルール」 ダイアログが表示されます。

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」フィールドの右の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、選択された「タイプ」が「XSLT」であることを確認してから、「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:MATMAS01」ソース要素を「ns0:MATMAS01」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. デフォルト値を維持し、「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・バーで「すべて保存」アイコンをクリックし、構成された新規のインバウンド・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 との既知のデプロイメント問題の調整

12c での既知のデプロイメント問題の調整方法については、6-27 ページ [6.4.3.3 項「12c との既知のデプロイメント問題の調整」](#) 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。6-53 ページ [「BPEL インバウンド・プロセスのデプロイ」](#) と同じ手順に従うことができます。

イベント・メッセージが SAP GUI を介してトリガーされると、ファイル・アダプタ・コンポーネント用に指定された場所に出力 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、6-54 ページ [「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。

Application Adapters インストールの次のフォルダに、このアウトバウンドの使用ケース・シナリオのサンプル・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
```

この節では、次のトピック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。詳細については、[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 用に空のコンポジットを作成するには、以下の手順を実行します。

1. 新規の SOA アプリケーションを作成します。
2. SOA アプリケーションに名前を入力し(「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て(「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。

詳細については、6-65 ページ [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。次のトピック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」](#)

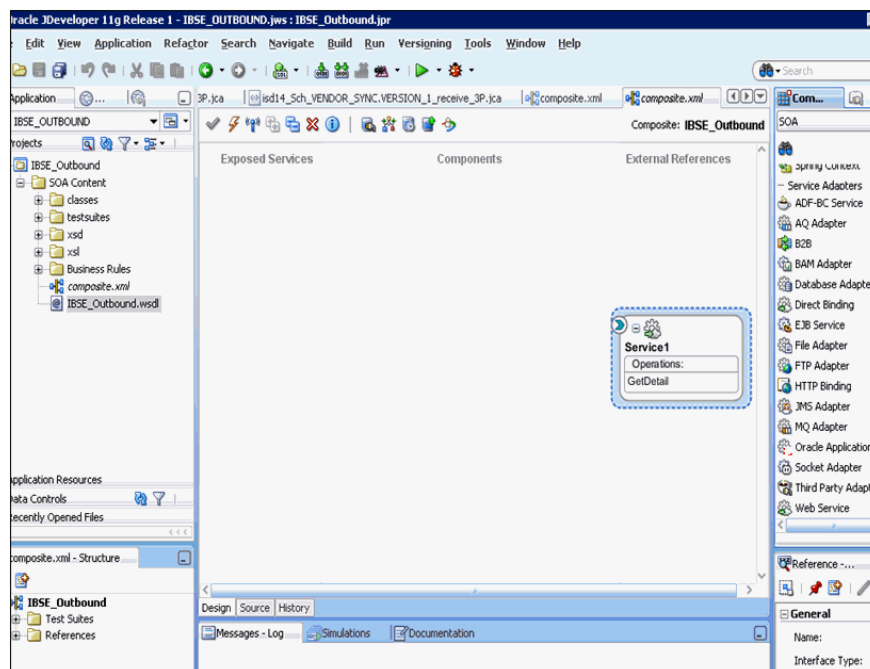
7.4.2.1 SOAP サービスの構成

SOAP サービスを構成するには、次のステップを実行します。

1. 「テクノロジー・アダプタ」ペインから「外部参照」ペインに「SOAP」ノードをドラッグ・アンド・ドロップします。
2. 「SOAP サービス」に適切な名前を入力し、「WSDL URL」フィールドの右の既存の WSDL の検索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、WSDL がアプリケーション・エクスペローラからエクスポートされた場所にナビゲートして、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、表示されます。

図 7-19 作成された Web サービス

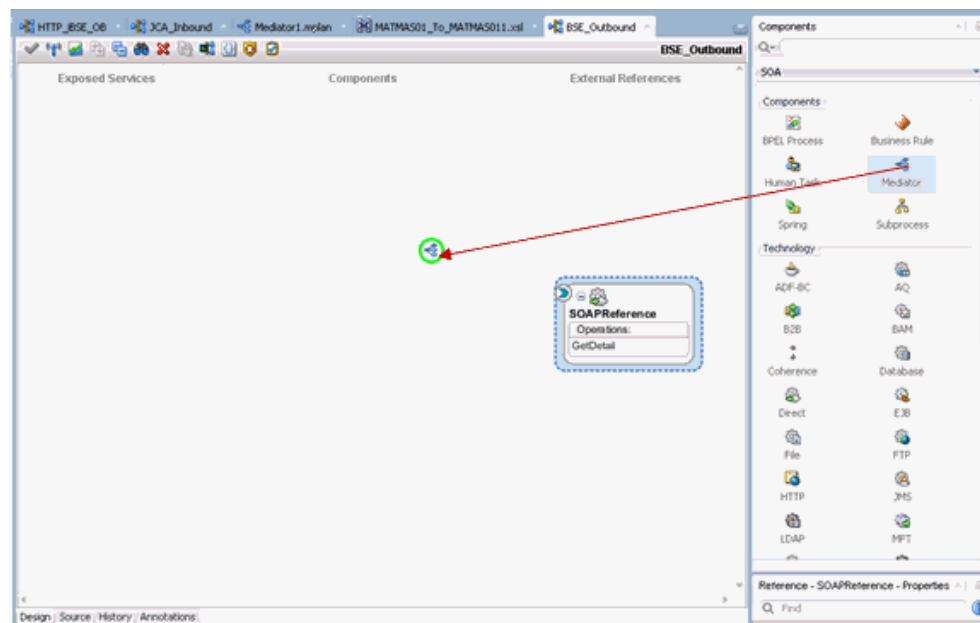


7.4.2.2 メディエータ・コンポーネントの作成

メディエータ・コンポーネントを作成するには、次のステップを実行します。

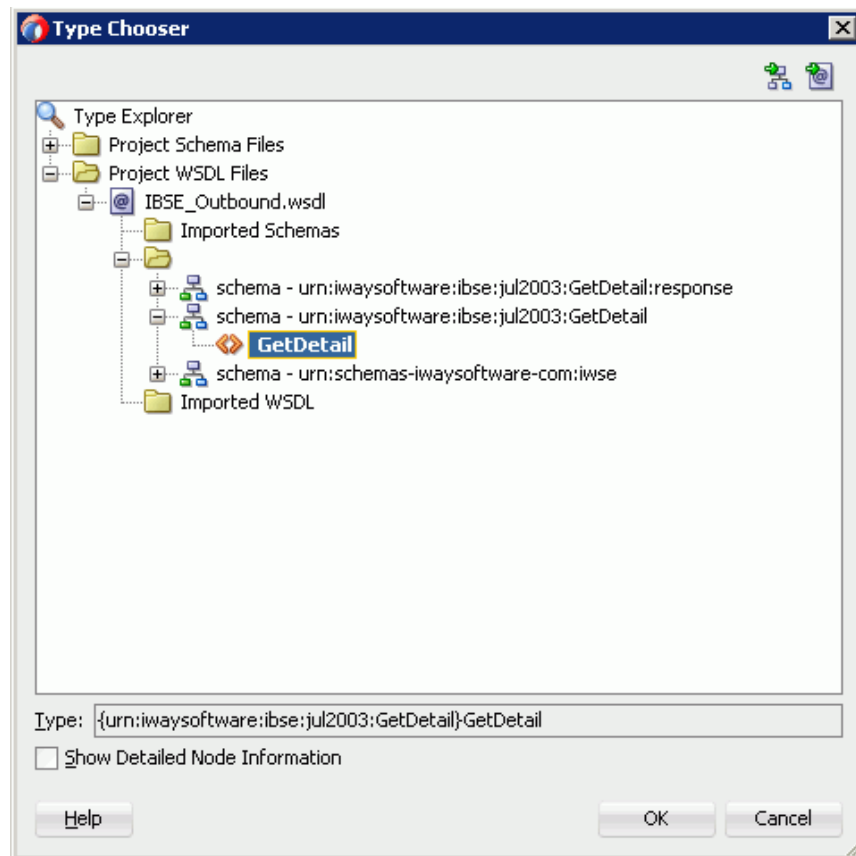
1. 図 7-20 に示すように、「コンポーネント」ペインから「コンポーネント」ペインに「メディエータ」コンポーネントをドラッグ・アンド・ドロップします。

図 7-20 「メディエータ」コンポーネント・ダイアログ



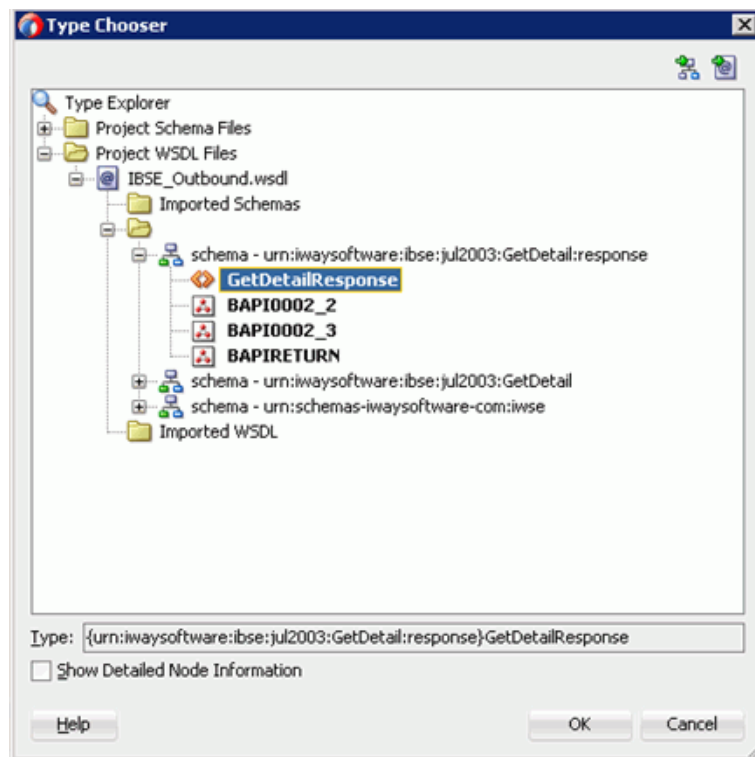
2. 「名前」フィールドで、新規のアウトバウンド・メディアータ・プロセス・コンポーネントの名前を入力します。
3. 「テンプレート」ドロップダウン・リストから「同期インタフェース」を選択します。
4. 関連する XML リクエスト・スキーマ・ファイルの選択用の「入力」フィールドの右の「参照」アイコンをクリックします。
5. 図 7-21 に示すように、「タイプ・チューザ」ダイアログで、「プロジェクトの WSDL ファイル」を展開し、「IBSE_Outbound.wsdl」を選択して、「GetDetail」をクリックします。

図 7-21 「タイプ・チューザ」ダイアログ



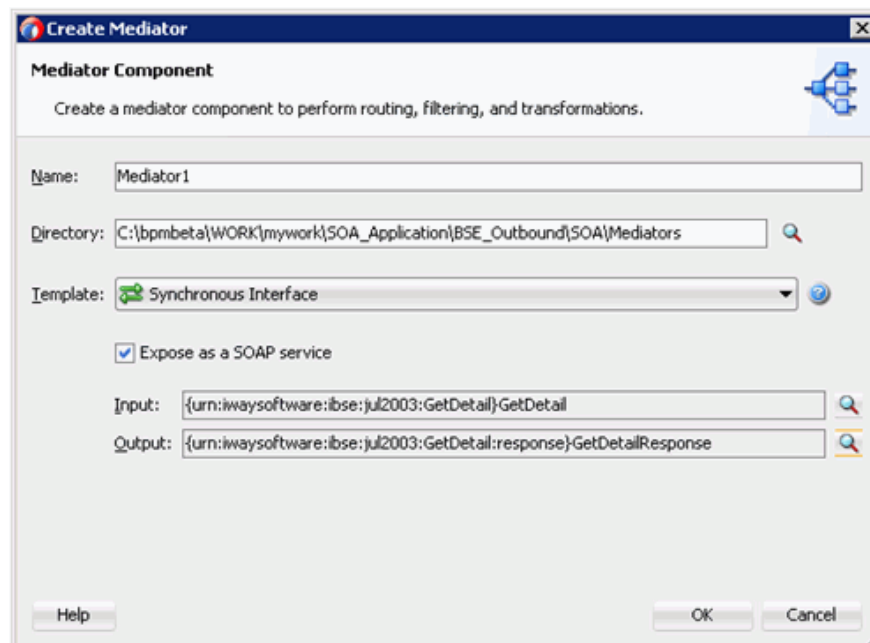
6. 「OK」をクリックします。
7. 関連する XML レスポンス・スキーマ・ファイルの選択用の「出力」フィールドの右の「参照」アイコンをクリックします。
8. 図 7-22 に示すように、「タイプ・チューザ」ダイアログで、「プロジェクトの WSDL ファイル」を展開し、「IBSE_Outbound.wsdl」を選択して、「GetDetailResponse」をクリックします。

図 7-22 「タイプ・チューザ」ダイアログ



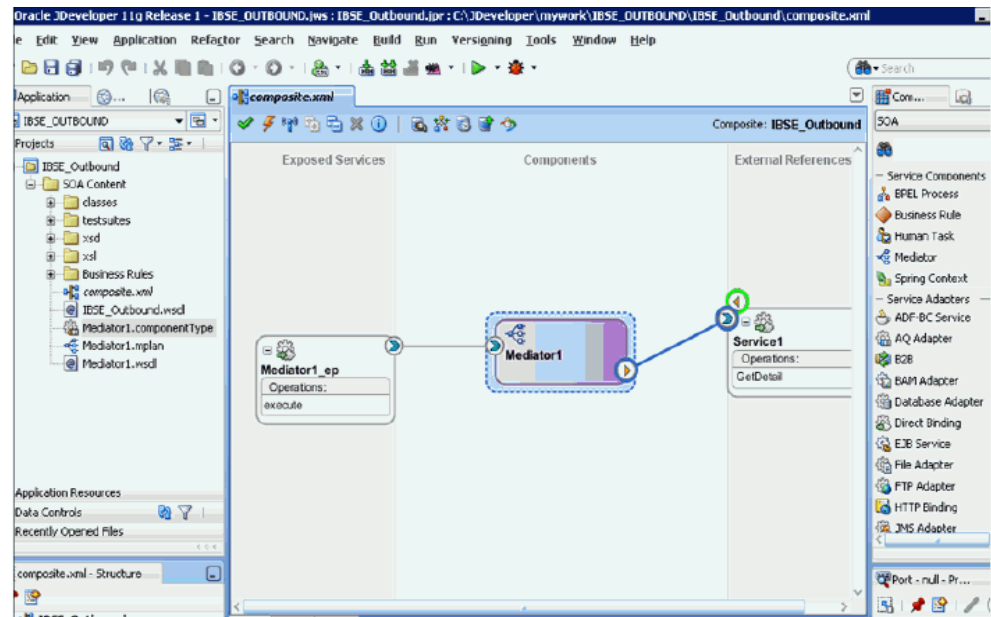
9. 図 7-23 に示すように、「OK」をクリックします。

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、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

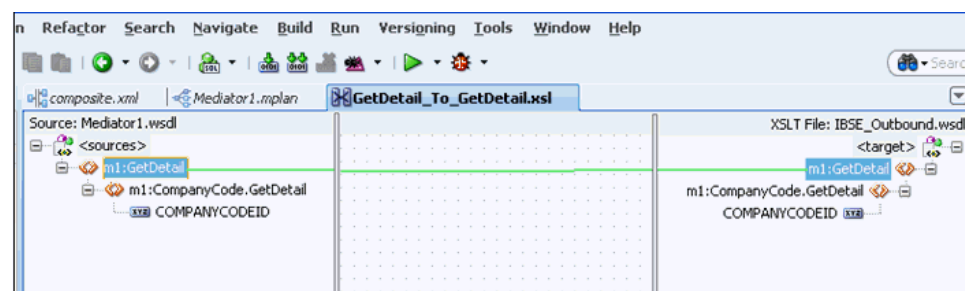


7.4.2.3 ルーティング・ルール構成

ルーティング・ルールを構成するには、次のステップを実行します。

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」フィールドの右の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た「タイプ」が「XSLT」であることを確認し、「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ピングします。

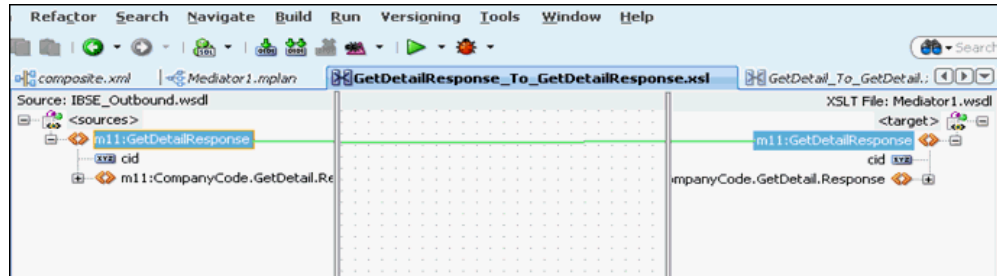
図 7-25 「GetDetail_To_GetDetail.xsl」タブ



6. 表示される「自動マップ・プリファレンス」ウィンドウで、デフォルト値を維持し、「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」の右側のアイコンをクリックします。

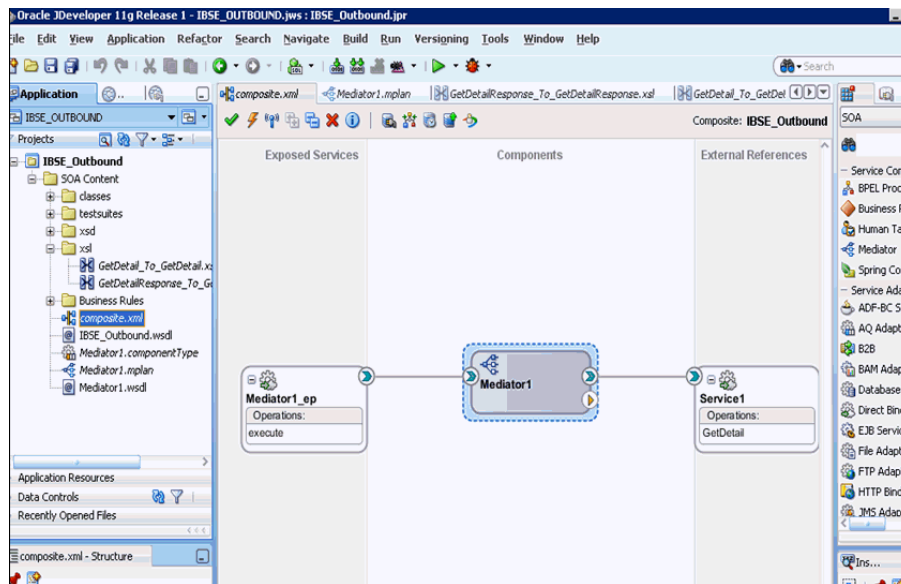
8. 表示される「リプライ・トランスフォーメーション・マップ」ウィンドウで、「追加」(+) アイコンをクリックし、「トランスフォーメーション・マップの作成」ダイアログ・ボックスで「タイプ」が「XSLT」に選択されていることを確認し、「OK」をクリックします。
9. 図 7-26 に示すように、「ns0:GetDetailResponse」ソース要素を「ns0:GetDetailResponse」ターゲット要素にマッピングします。

図 7-26 ソースおよびターゲット要素



10. 表示される「自動マップ・プリファレンス」ウィンドウで、デフォルト値を維持し、「OK」をクリックします。
11. 図 7-27 に示すように、メニュー・バーで「すべて保存」アイコンをクリックし、構成された新規のアウトバウンド・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」と同じ手順に従うことができます。

デプロイされたら、7.2.4 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動」に定義されたように入力 XML を起動できます。

Oracle SOA Suite での BPM サービス・コンポーネントの統合

Oracle Application Adapter for SAP R/3 は Business Process Management (BPM) にシームレスに統合されており、Web サービスの統合を可能にします。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。Web Service Definition Language (WSDL) ドキュメントとして公開されるアダプタ・サービスを使用します。

この章の内容は以下のとおりです。

- 8.1 項「概要」
- 8.2 項「アダプタのデプロイメント」
- 8.3 項「新規のアプリケーション・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 イベントの作業 (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 を統合するには、Oracle Application Adapter for SAP R/3 が 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。また、基になるアダプタ・サービスは、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) の両方のサービス用に Oracle Adapter アプリケーション・エクスプローラ (アプリケーション・エクスプローラ) でデザインタイム中に生成される WSDL ファイルとして公開される必要があります。詳細については、4-18 ページ「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・サービスにそれぞれ適切な BPM プロセスの設計に使用されます。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後すぐに、新規構築されたプロセスはすべて自動的に Oracle Enterprise Manager Console にデプロイされ、BPM プロセスの実行、監視、管理、およびアダプタ・イベントのリスニングを行うことができます。

8.2 アダプタのデプロイメント

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

8.3 新規のアプリケーション・サーバー接続の構成

Oracle JDeveloper での新規のアプリケーション・サーバー接続の構成の詳細は、6.3 項「新規のアプリケーション・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、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 プロセスを設計する方法について説明します。

Application Adapters インストレーションの次のフォルダに、このアウトバウンドの使用ケース・シナリオのサンプル・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
アウトバウンドの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 アプリケーション・エクスプローラ (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPM デザイナ (JDeveloper)

注意： この章に記載されている例では JDeveloper が使用されていません。

この節では、次のトピック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 との既知のデプロイメント問題の調整」
- 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 ドキュメントの起動」

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。詳細については、6-8 ページ「リクエスト/レスポンス型のサービス用の 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 用に空のコンポジットを作成するには、次のステップを実行します。

1. 新規の BPM アプリケーションを作成します。
2. 新規の BPM アプリケーションに名前を入力し、「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」を選択してから、「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、次のトピックで構成される BPM アウトバウンド・プロセスの定義方法について説明します。

- 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「書き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、次のステップを実行します。

1. 作成し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. カスタム/サードパーティ・ペインから「外部参照」ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドの右の**既存の WSDL の検索**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. 次のディレクトリまでブラウズし、アウトバウンド WSDL ファイルを選択します。
<ADAPTER_HOME>\wsdls
7. 「OK」をクリックします。
「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リクエスト/レスポンス XML スキーマ・ファイル(.xsd)が、作成済のプロジェクト・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」フィールドの右の**JCA ファイルの検索**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. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。
<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。
JCA プロパティ・ファイルのコピーがプロジェクト・フォルダ内に作成されます。
「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・ショットを含む詳細については、6-11 ページ [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

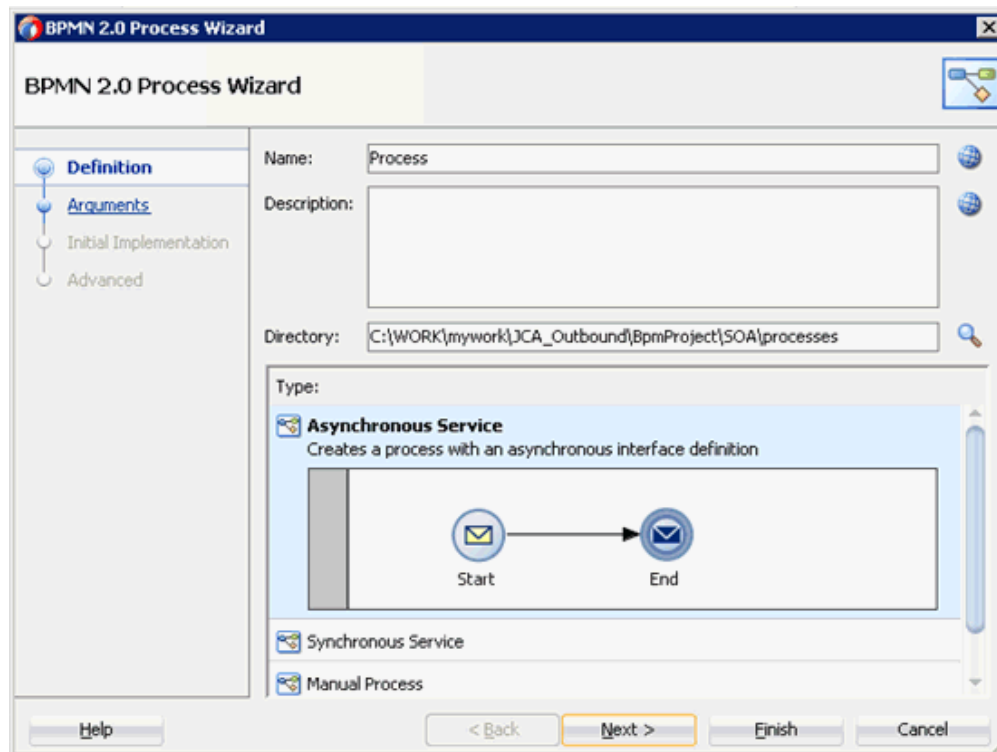
この項では、アウトバウンド BPM プロセス・コンポーネントを構成する方法について説明します。

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、次のステップを実行します。

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

図 8-1 に示すように、「BPMN プロセスの作成」ダイアログが表示されます。

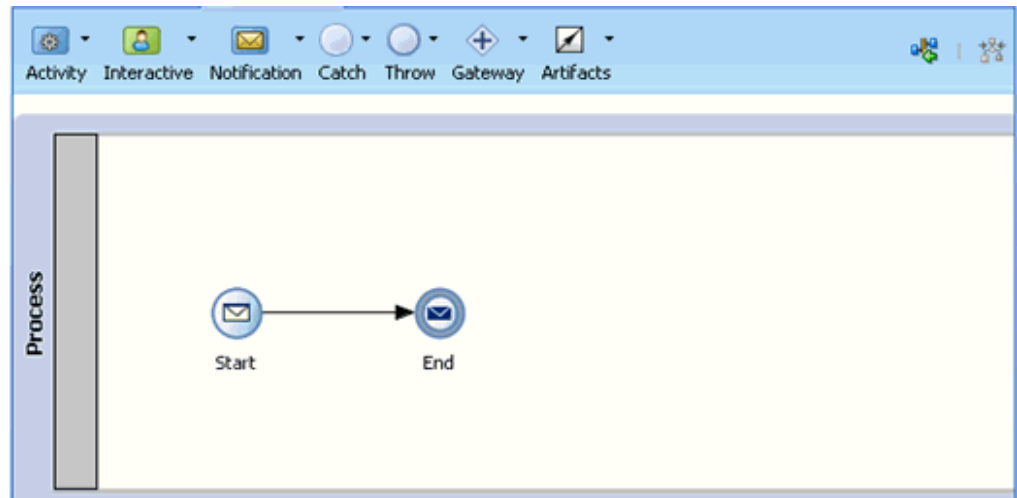
図 8-1 「BPMN プロセスの作成」ダイアログ



2. 「タイプ」領域（「非同期サービス」）で選択されたデフォルトのオプションを受け入れ、「終了」をクリックします。

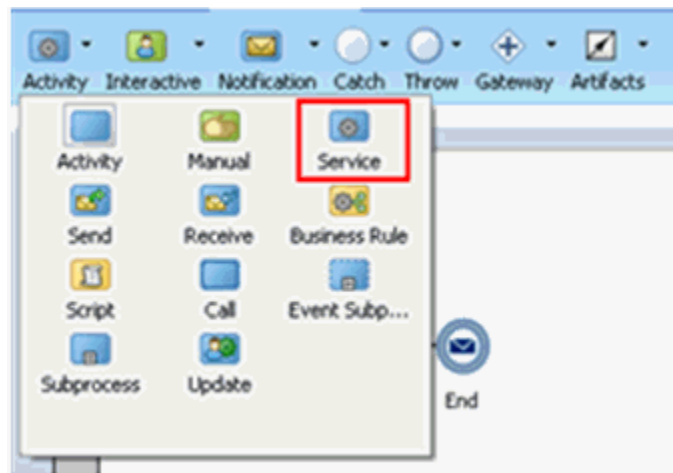
図 8-2 に示すように、「BPMN プロセス」が表示されます。

図 8-2 「BPMN プロセス」



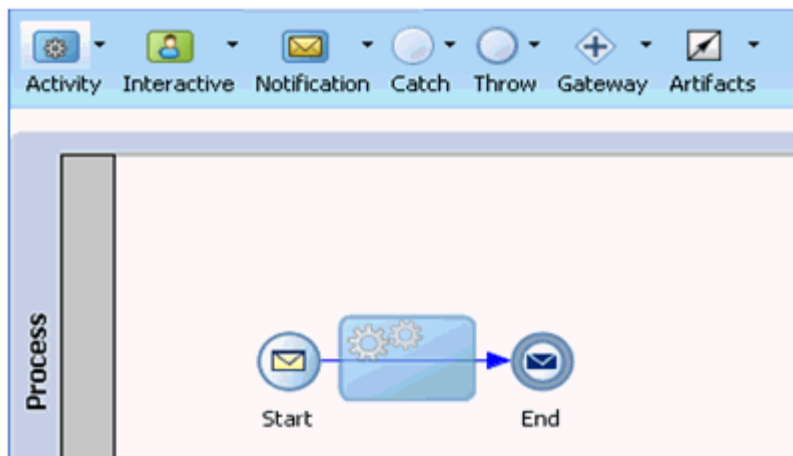
3. 図 8-3 に示すように、「アクティビティ」ドロップダウン・メニューをクリックして、「サービス」を選択します。

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、「開始」と「終了」イベント・コンポーネント間の伝送路に「サービス」アイコンをドロップします。

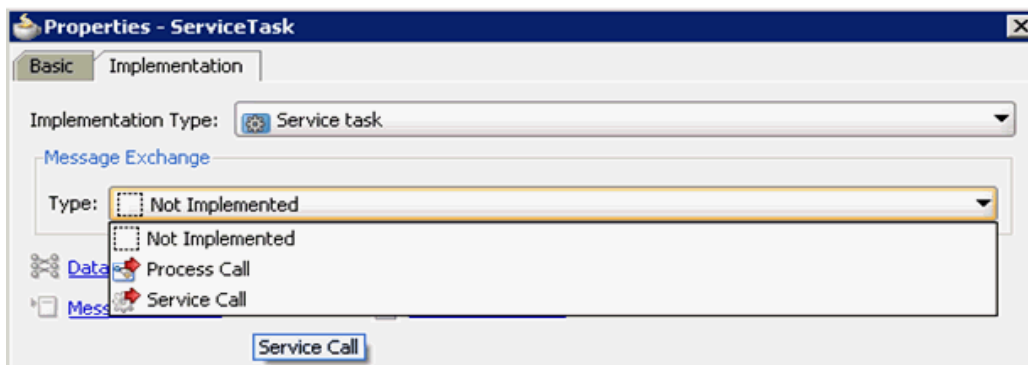
図 8-4 「アクティビティ」アイコン



「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、メッセージ交換タイプ・リストから「サービス・コール」を選択します。

図 8-5 「サービス・コール」



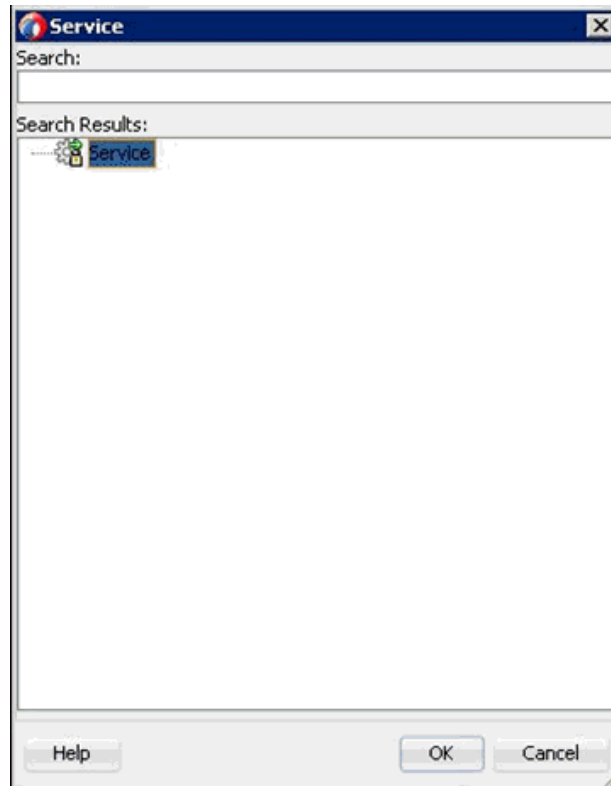
7. 図 8-6 に示すように、「サービス」フィールドの右の「参照」アイコンをクリックします。

図 8-6 「参照」アイコン



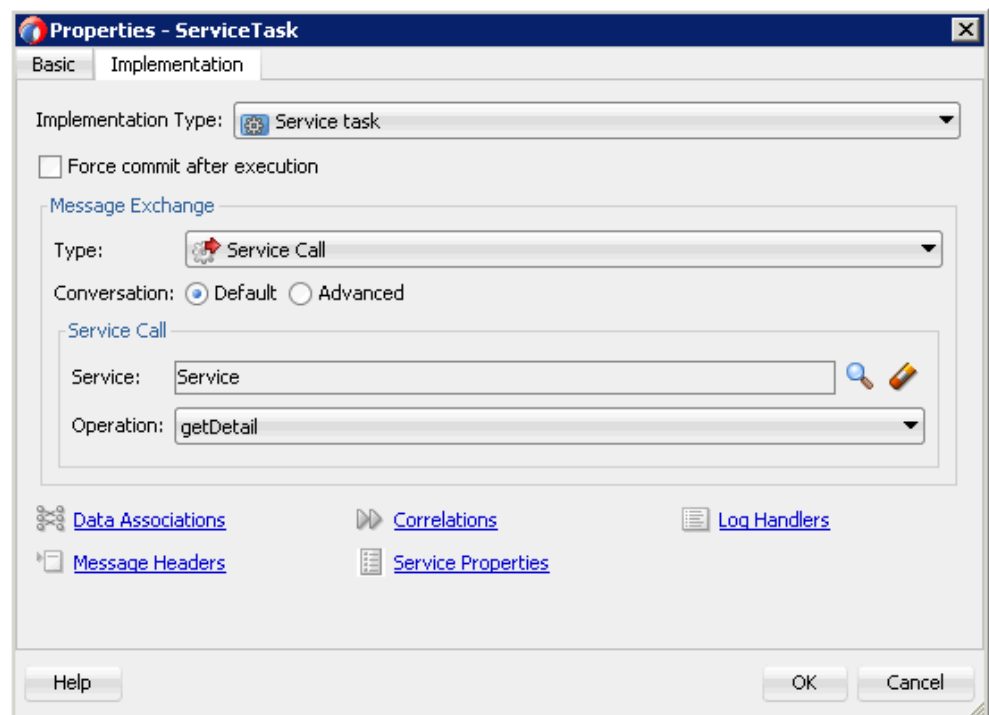
図 8-7 に示すように、「サービス」ダイアログが表示されます。

図 8-7 「サービス」 ダイアログ



- 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。
図 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. 図 8-9 に示すように、「プロセス」の下で左ペインの「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。

図 8-9 「新規」オプション

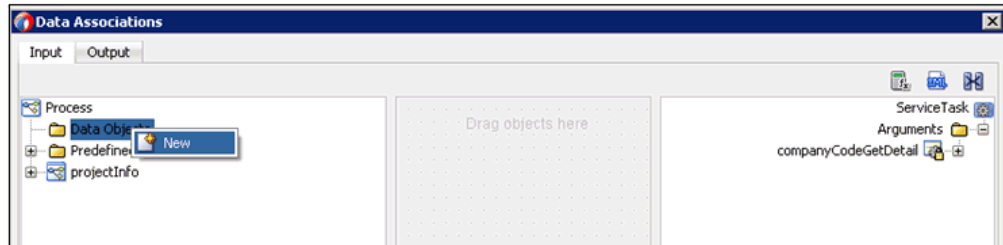
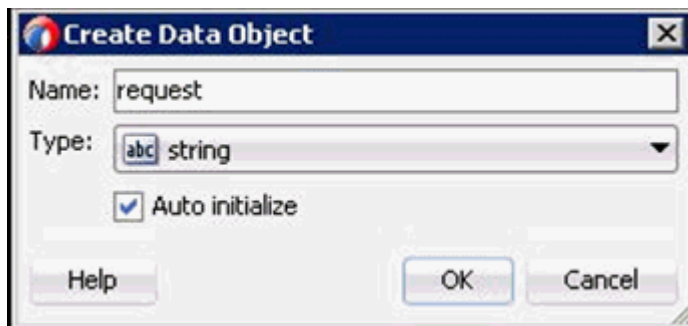


図 8-10 に示すように、データ・オブジェクトの作成ダイアログが表示されます。

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」フィールドで名前を入力し（「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、リストから「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

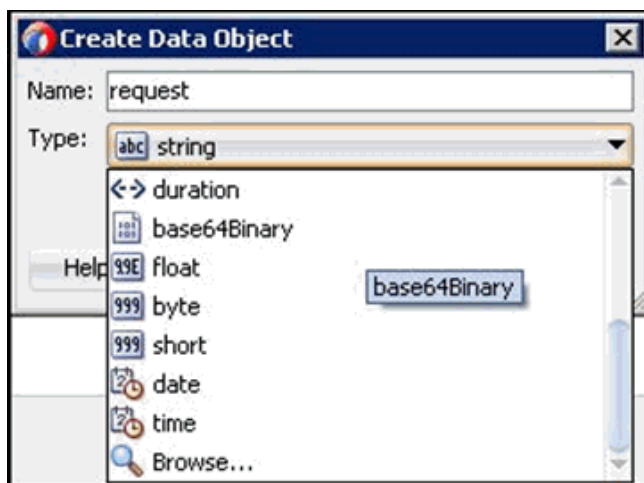
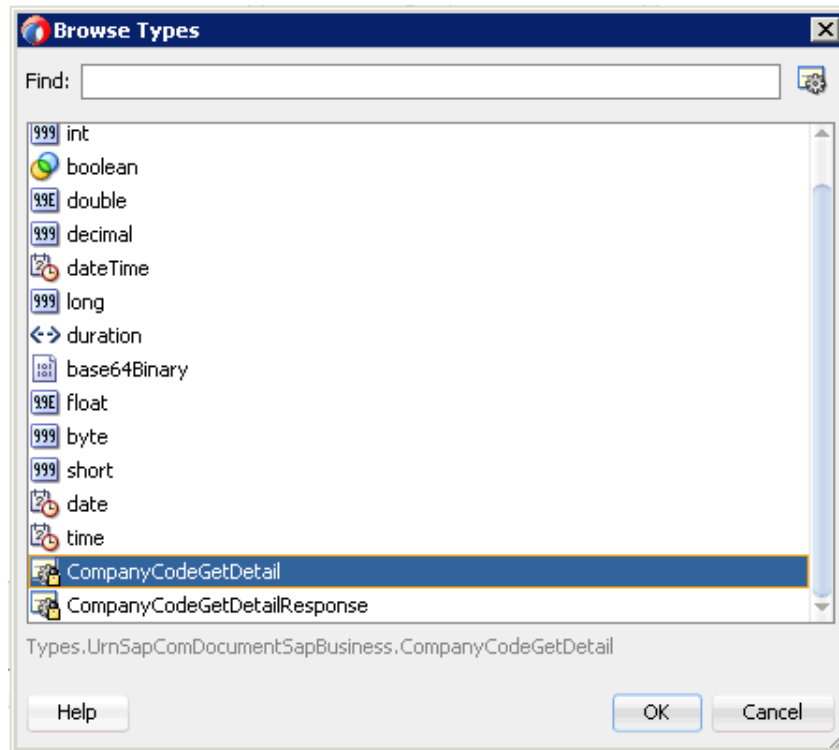


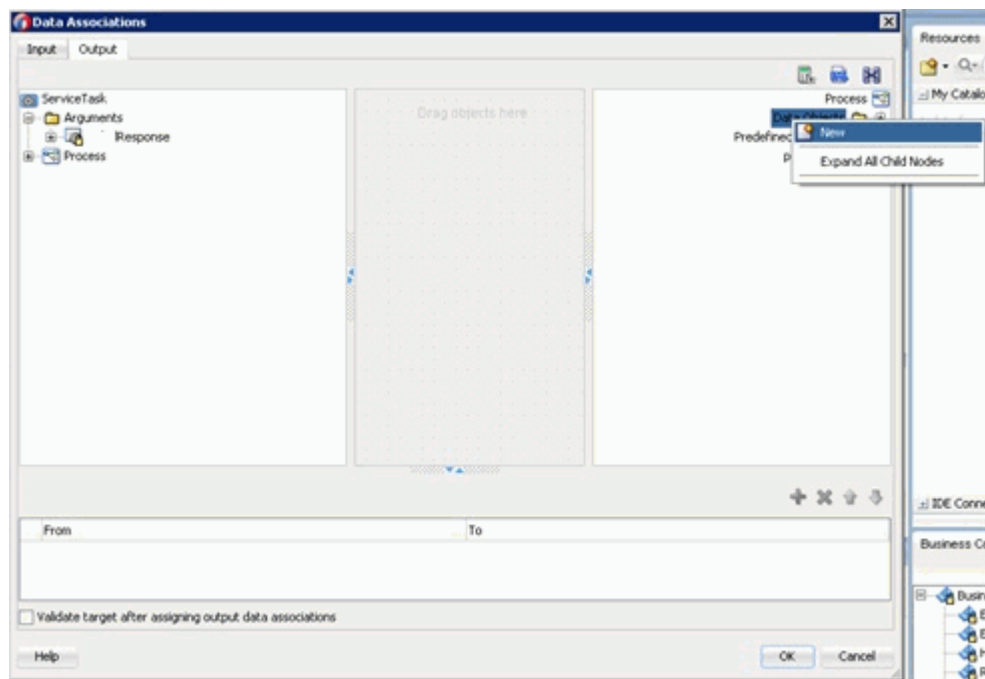
図 8-12 に示すように、「参照タイプ」ダイアログが表示されます。

図 8-12 「参照タイプ」 ダイアログ



12. 最初のコンポーネントを選択し (「CompanyCodeGetDetail」など)、「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。
13. 「OK」をクリックします。
作成された「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」を選択して、他のデータ・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

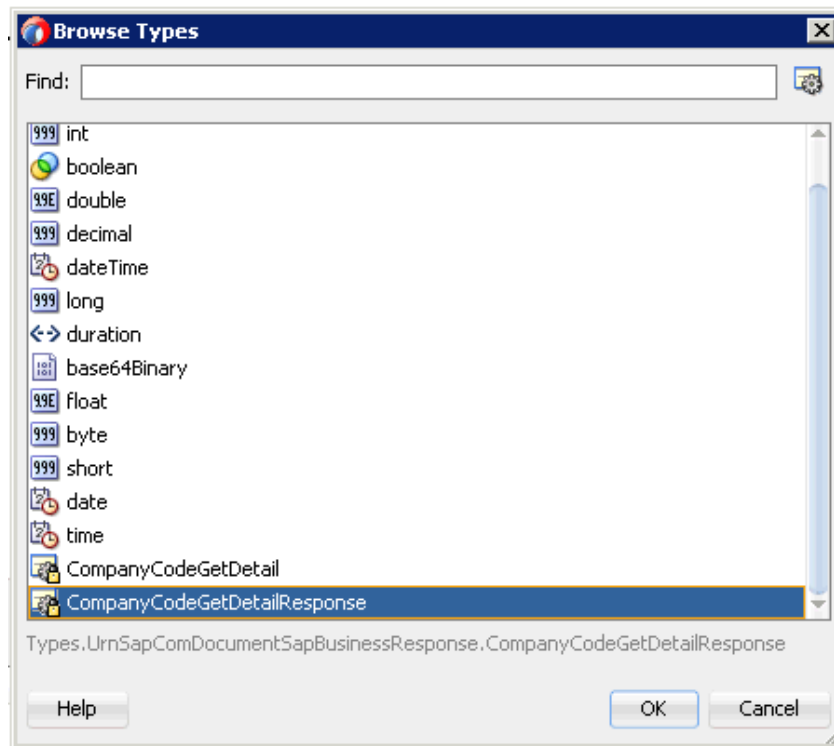


データ・オブジェクトの作成ダイアログが表示されます。

15. 「名前」 フィールドで名前を入力し (「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、リストから「参照」を選択します。

図 8-14 に示すように、「参照タイプ」ダイアログが表示されます。

図 8-14 「参照タイプ」 ダイアログ



16. 2 番目のコンポーネントを選択し (「CompanyCodeGetDetailResponse」など)、「OK」をクリックします。

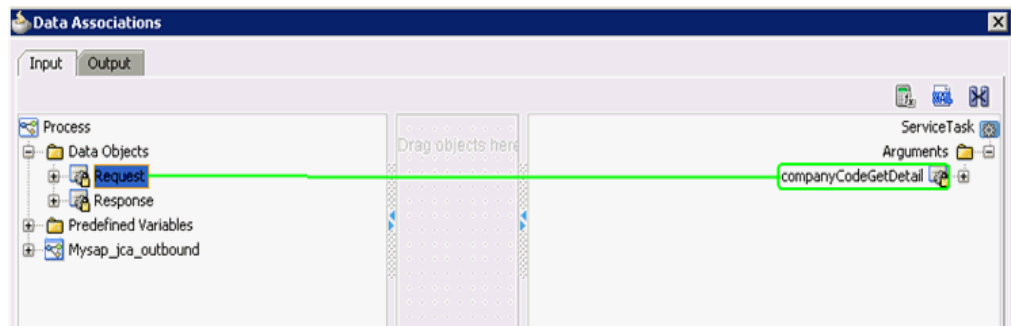
データ・オブジェクトの作成ダイアログに戻ります。

17. 「OK」をクリックします。

作成された「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。

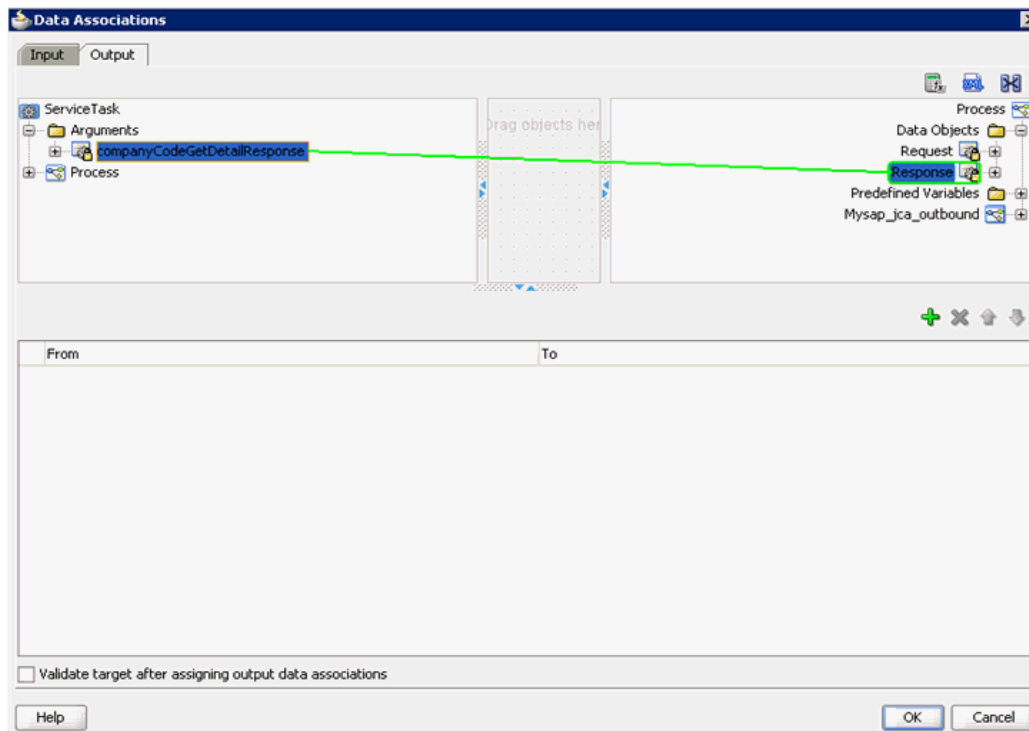
18. 図 8-15 に示すように、「入力」タブの左ペインの「データ・オブジェクト」ノードの下の「リクエスト」データ・オブジェクトを選択し、右ペインの「引数」ノードの下の「CompanyCodeGetDetail」にドラッグして接続します。

図 8-15 「リクエスト」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、左ペインの「引数」ノードの下の「CompanyCodeGetDetailResponse」を選択し、「データ・オブジェクト」ノードの下の「レスポンス」データ・オブジェクトにドラッグして接続します。

図 8-16 「レスポンス」データ・オブジェクト



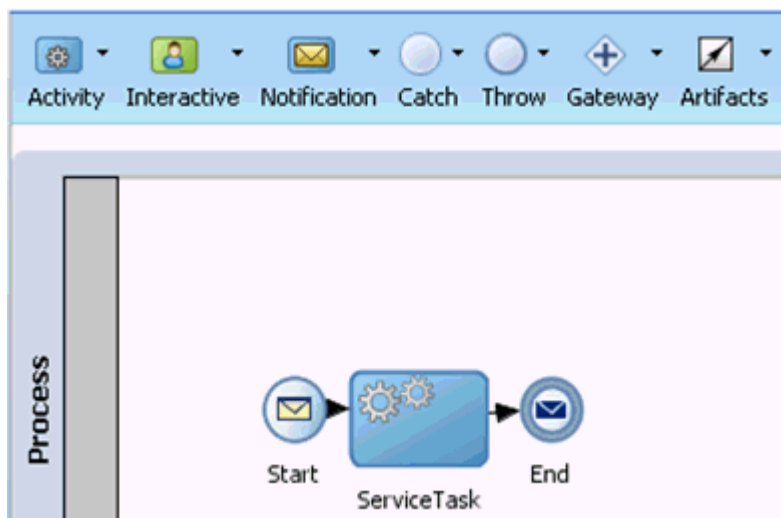
20. 「OK」をクリックします。

「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、「開始」と「終了」イベント・コンポーネント間に「サービス・タスク」が作成されます。

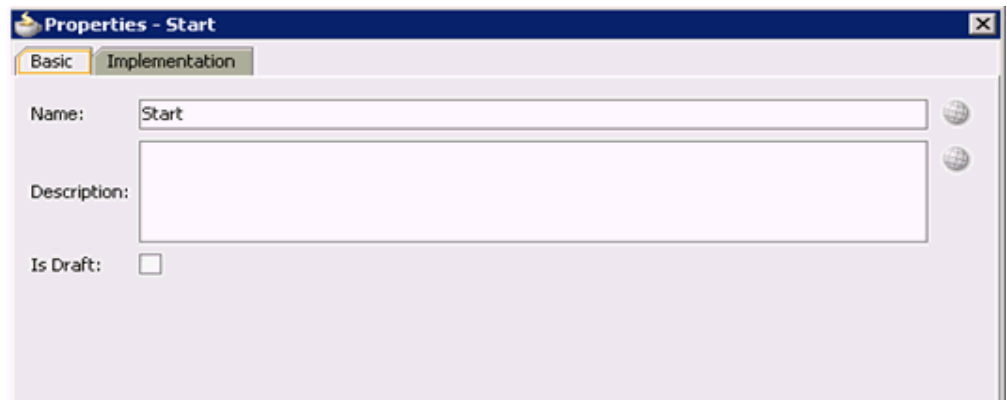
図 8-17 サービス・タスク



22. プロセスを保存し、「開始」イベント・コンポーネントをダブルクリックします。

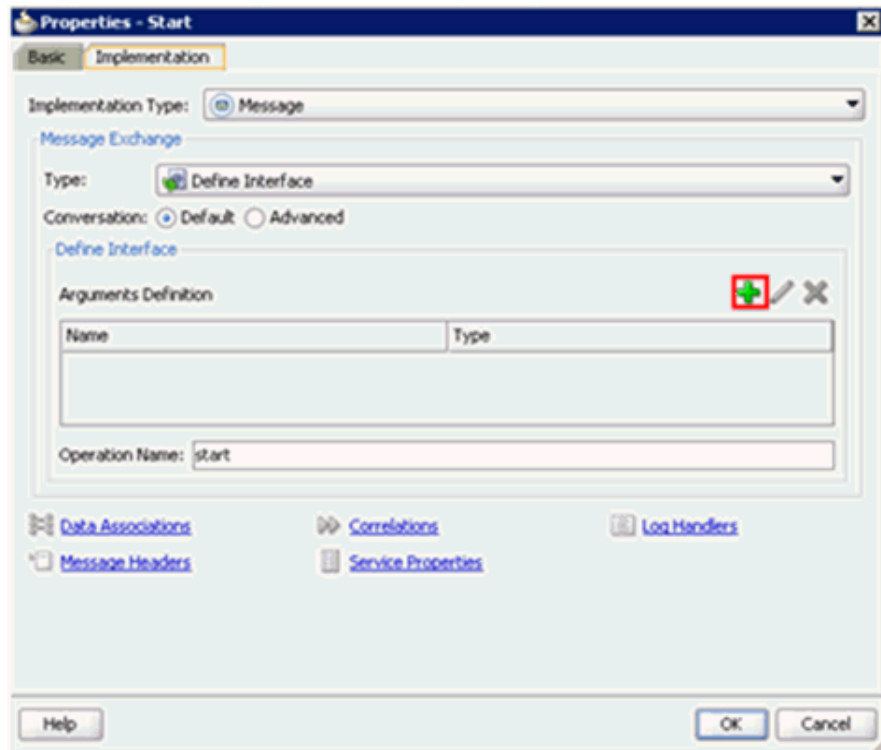
図 8-18 に示すように、「プロパティ - 開始」ダイアログが表示されます。

図 8-18 「プロパティ - 開始」 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. 「引数の定義」フィールドの右の「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」フィールドで名前を入力し (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、リストから「参照」を選択します。

図 8-20 「引数の作成」ダイアログ

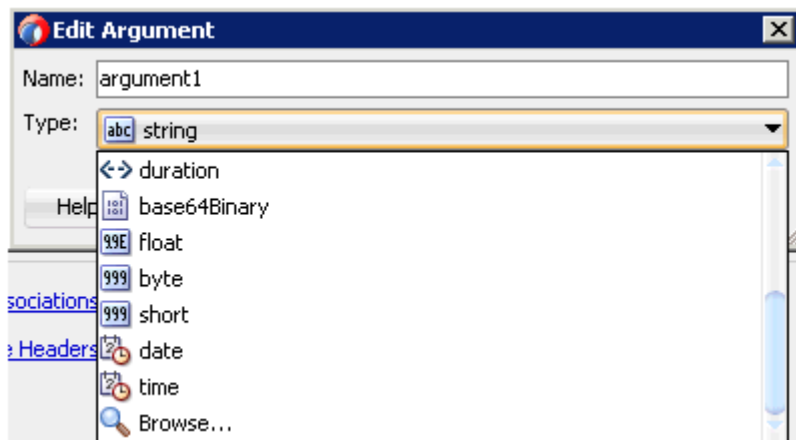
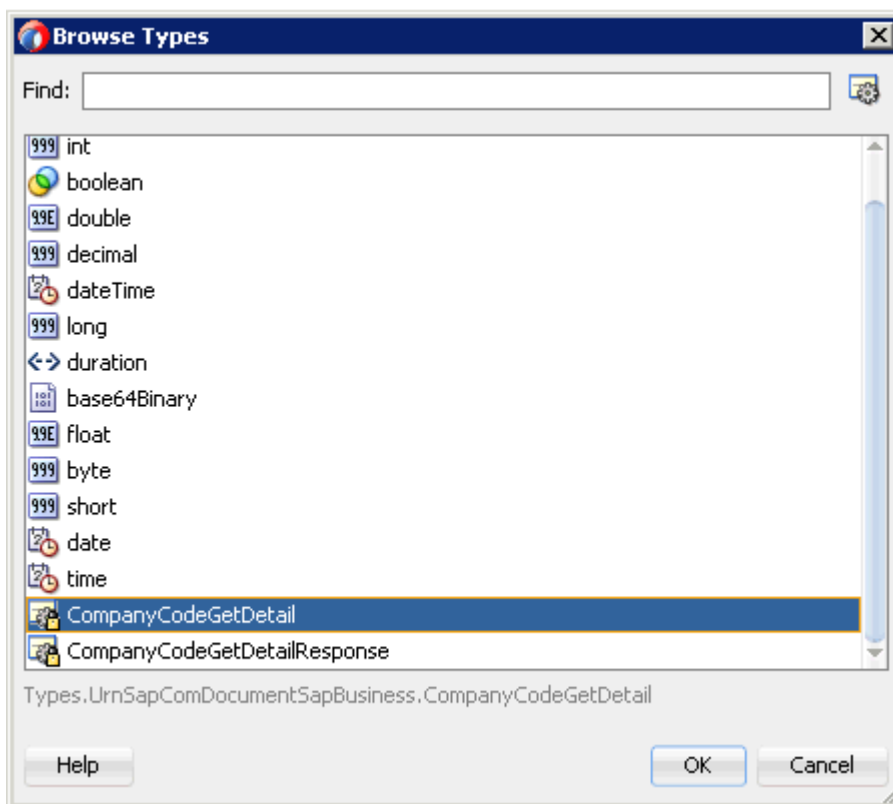


図 8-21 に示すように、「参照タイプ」ダイアログが表示されます。

図 8-21 「参照タイプ」ダイアログ

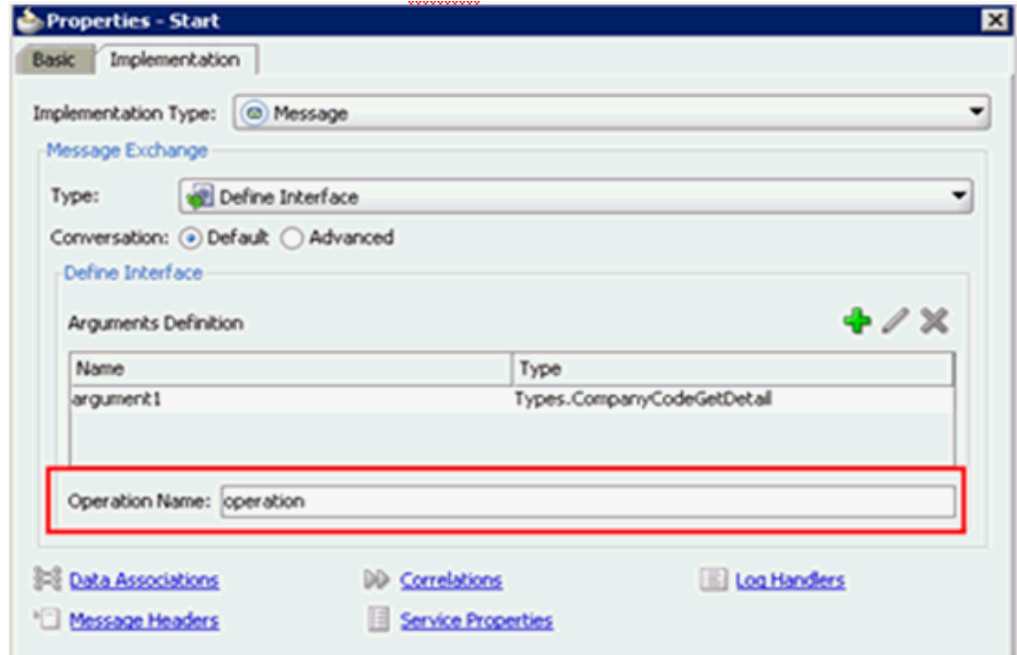


26. 最初のコンポーネントを選択し (「CompanyCodeGetDetail」など)、「OK」をクリックします。
「引数の作成」ダイアログに戻ります。
27. 「OK」をクリックします。
「プロパティ - 開始」ダイアログに戻ります。

28. 図 8-22 に示すように、「操作名」フィールドで、「開始」(デフォルト)を「操作」に変更します。

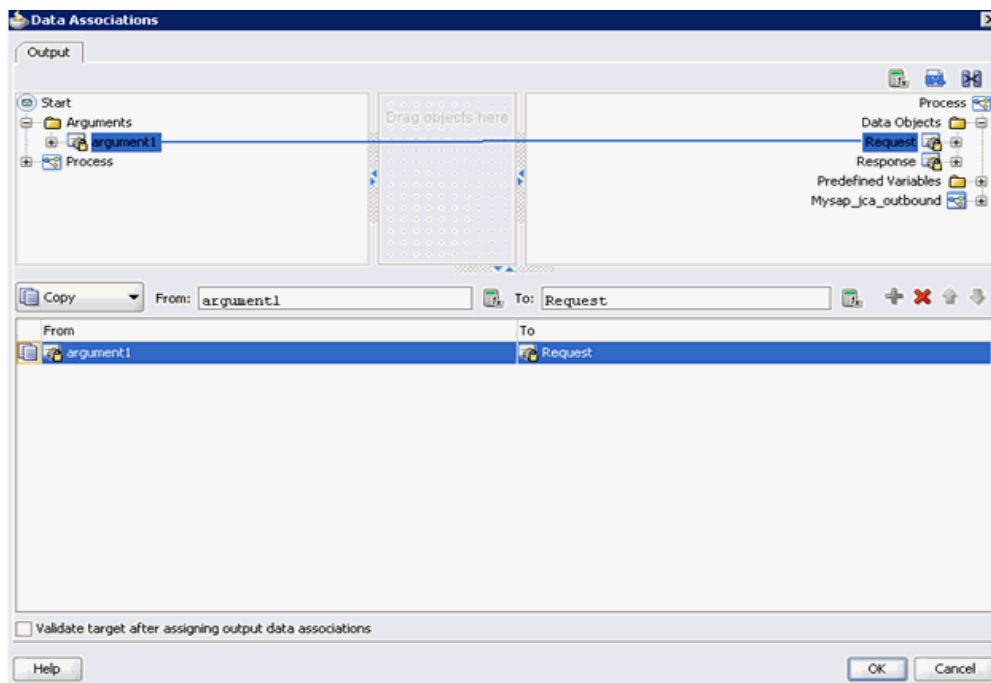
注意: この変更は、古い BPM ペイロードで作業する場合に必要です。

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. 左ペインの「引数」ノードの下の「arguments1」を選択し、右ペインの「データ・オブジェクト」の下の「リクエスト」データ・オブジェクトにドラッグして接続します。
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

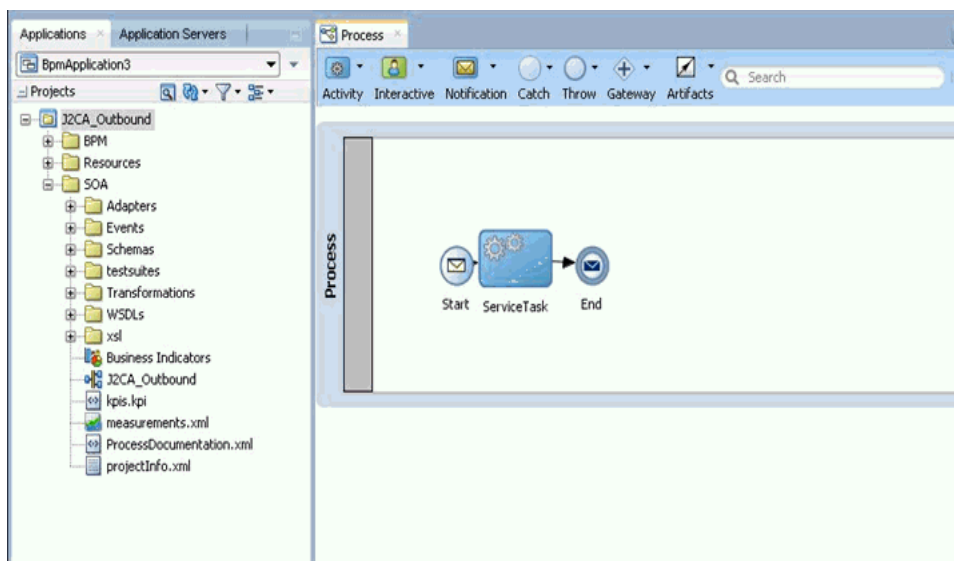


「プロパティ - 開始」 ダイアログに戻ります。

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・バーで「すべて保存」アイコンをクリックし、構成された新規のアウトバウンド 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。

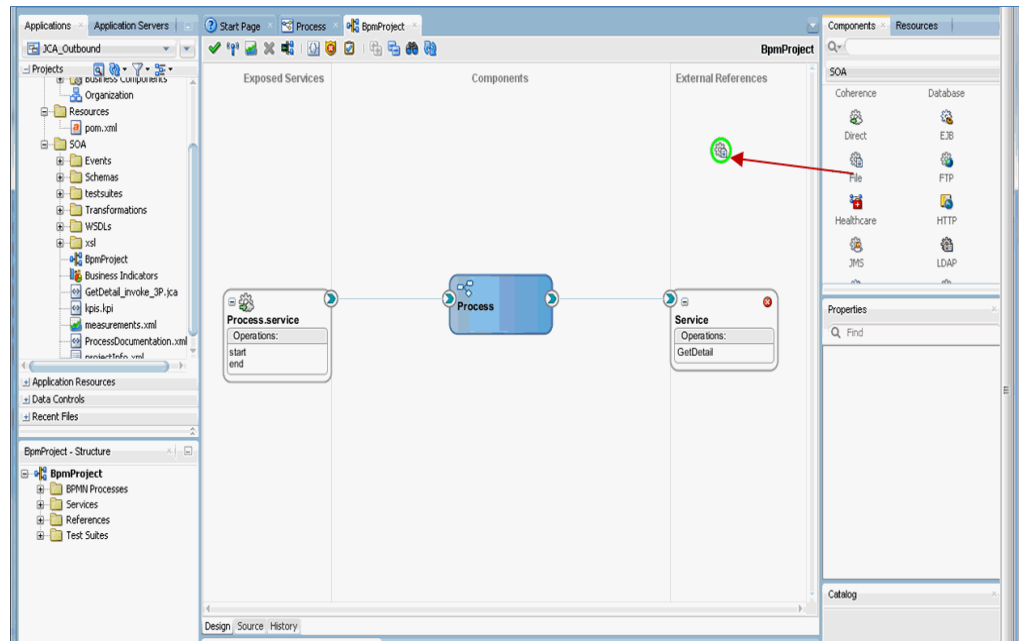
8.4.2.3 書込み操作のファイル・アダプタの作成

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、次のステップを実行します。

1. 図 8-25 に示すように、「テクノロジー・アダプタ」ペインから「外部参照」ペインに「ファイル・アダプタ」コンポーネントをドラッグ・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。

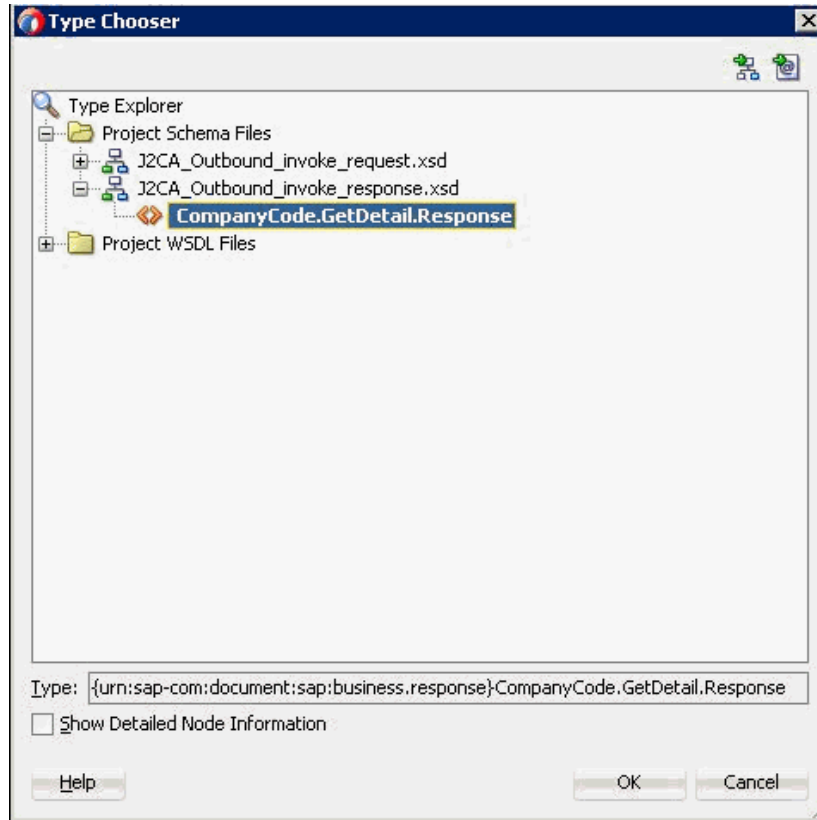
2. 参照名を入力します (「FileWrite」など)。
3. 「次へ」をクリックします。
「アダプタ・インタフェース」ページが表示されます。
4. 「操作およびスキーマから定義 (後で指定)」オプションが選択されていることを確認します。
5. 「次へ」をクリックします。
「ファイル・サーバーの接続」ページが表示されます。
6. 「次へ」をクリックします。
「操作」ページが表示されます。
7. 「操作タイプ」オプションのリストから「Write File」を選択し、「操作名」(「Write」など)を指定します。
8. 「次へ」をクリックします。
「ファイル構成」ページが表示されます。
9. 出力ファイルの書込み先となるファイル・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」フィールドの右の「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。

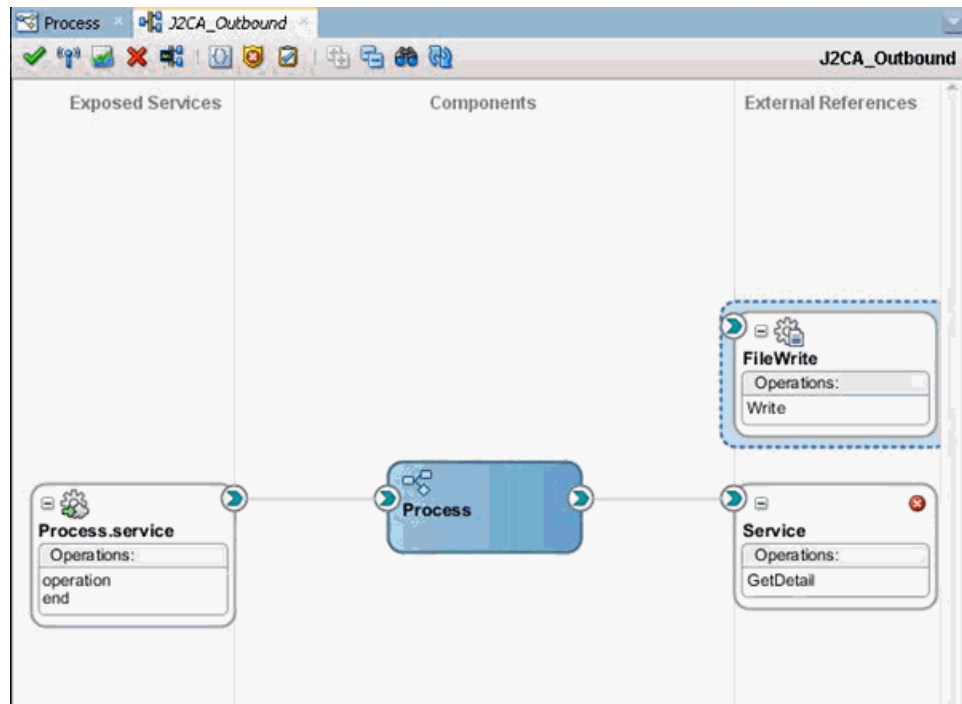
図 8-26 「タイプ・チューザ」ダイアログ



13. 「プロジェクトのスキーマ・ファイル」、「J2CA_Outbound_invoke_response.xsd」の順に展開します。
14. 使用可能なプロジェクト（「CompanyCode.GetDetail.Response」など）を選択します。
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。

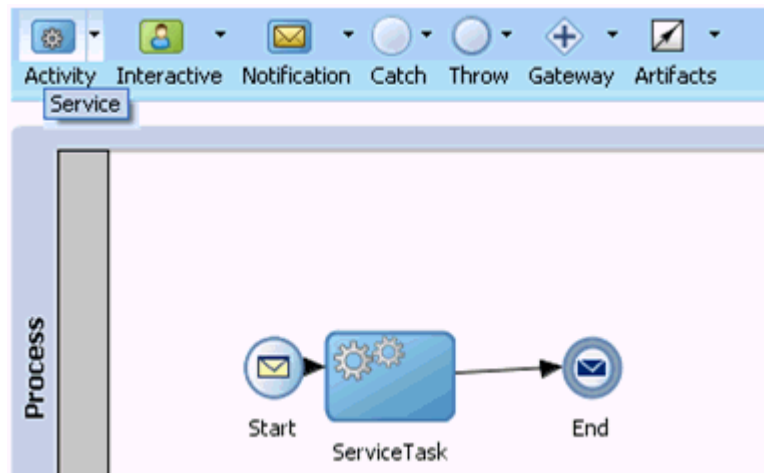
図 8-27 に示されるように、「外部参照」ペインでファイル・アダプタ・サービスが作成されます。

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。
図 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. 図 8-29 に示すように、「サービス・タスク」と「終了」イベント・コンポーネント間の伝送路に「サービス」アイコンをドロップします。

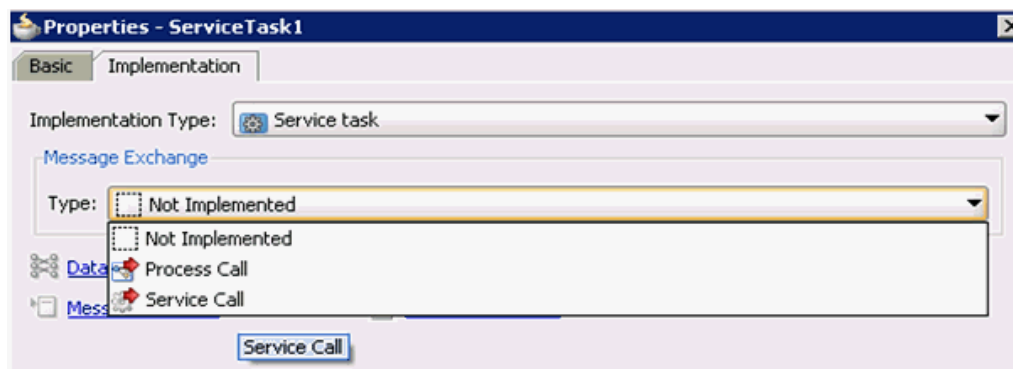
図 8-29 「アクティビティ」アイコン



「プロパティ - ServiceTask1」ダイアログが表示されます。

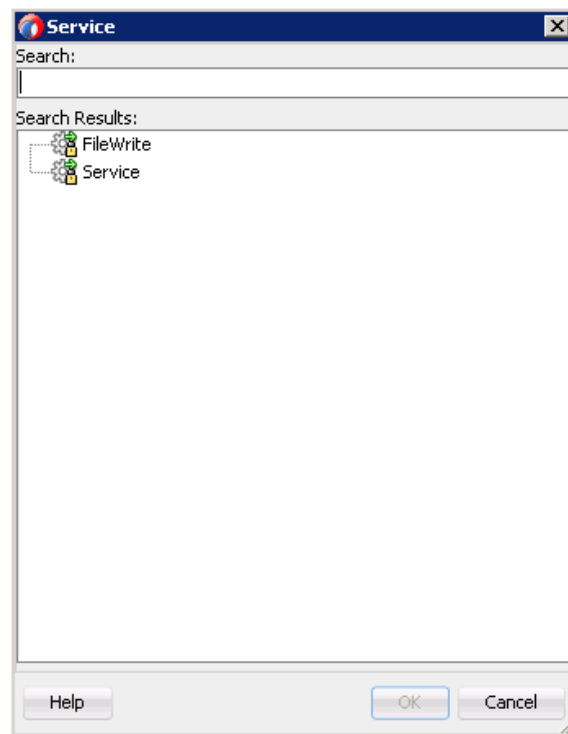
21. 「実装」タブをクリックします。
22. 図 8-30 に示すように、「メッセージ交換」セクションの「タイプ」ドロップダウン・リストから「サービス・コール」を選択します。

図 8-30 「サービス・コール」



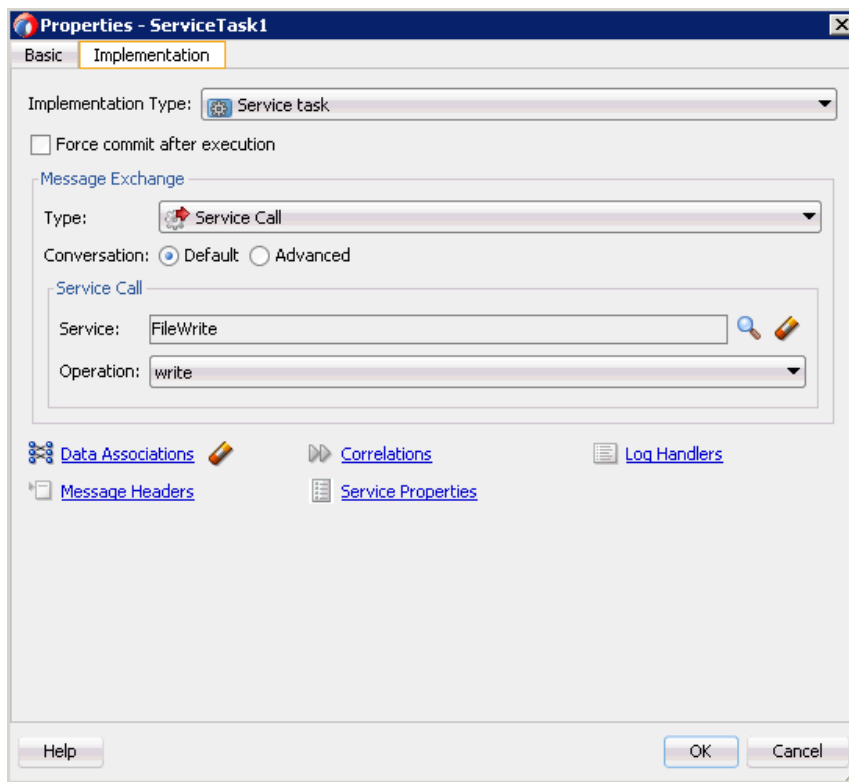
23. 「サービス」フィールドの右の「参照」アイコンをクリックします。
図 8-31 に示すように、「サービス」ダイアログが表示されます。

図 8-31 「サービス」 ダイアログ



24. 作成された書き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。

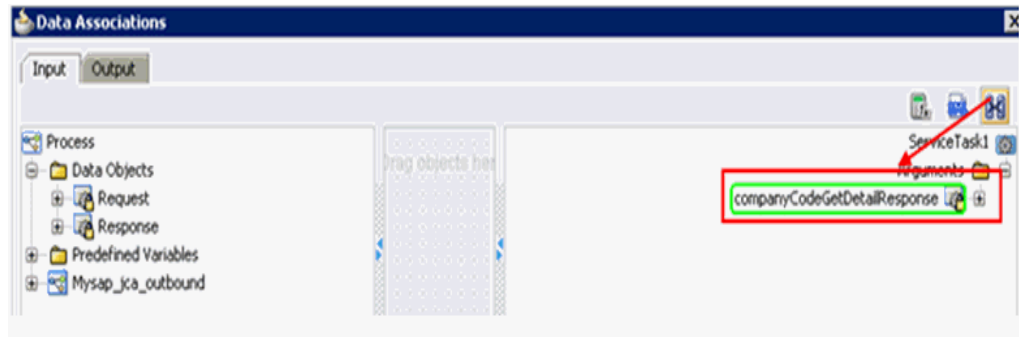
図 8-33 「データ・アソシエーション」ダイアログ



26. 「入力」タブで、右上隅の「XSL 変換」アイコンをクリックします。

27. 図 8-34 に示すように、「XSL 変換」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

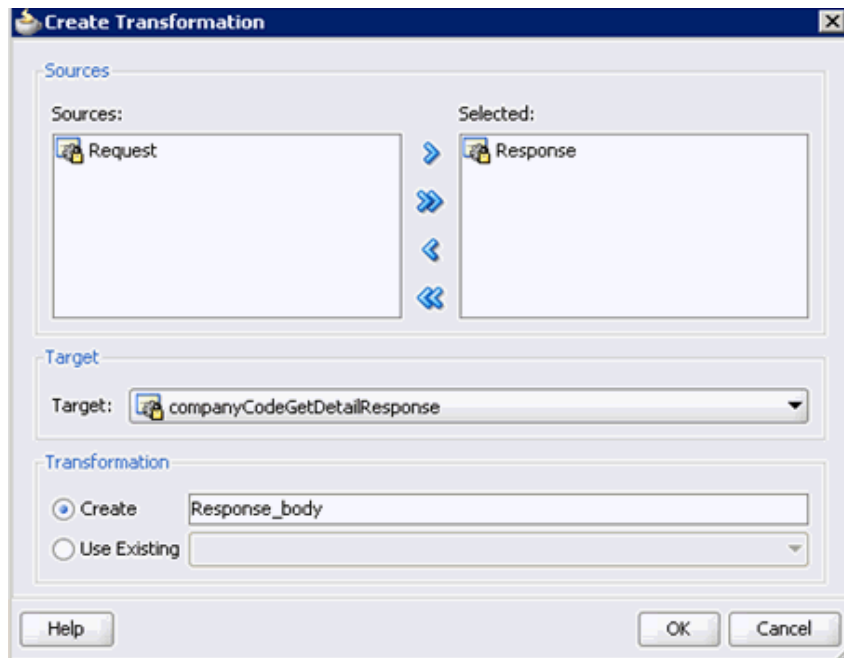


トランスフォーメーションの作成ダイアログが表示されます。

28. 「ソース」セクションで「レスポンス」を選択して、右矢印記号をクリックします。

図 8-35 に示すように、「レスポンス」オブジェクトが「選択済」要素領域に追加されます。

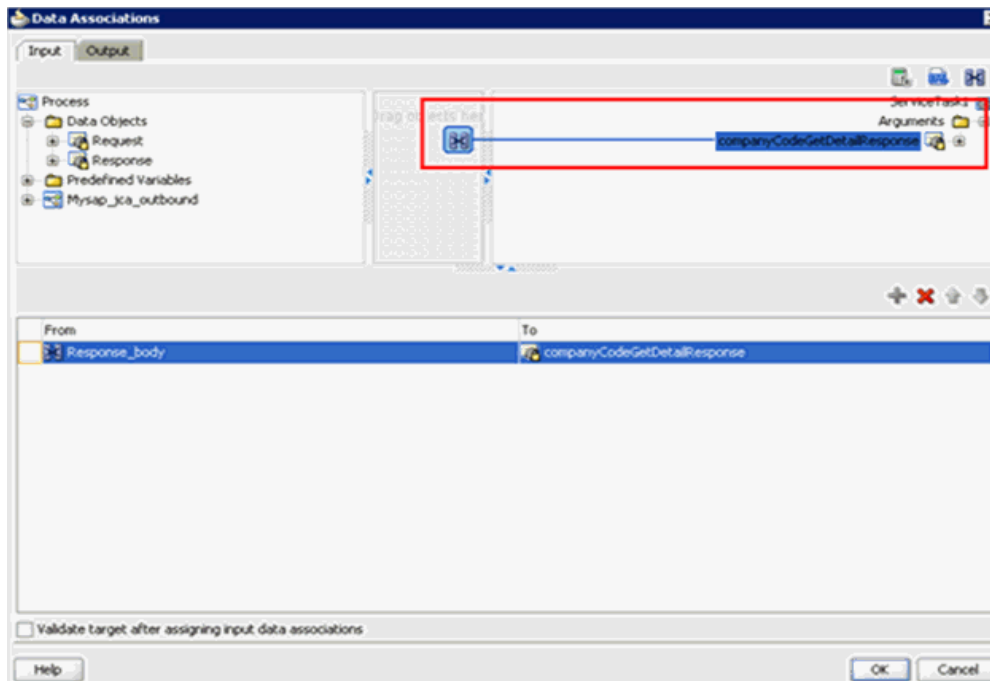
図 8-35 「レスポンス」オブジェクト



29. 「OK」をクリックすることで、「ターゲット」ドロップダウン・リストで選択されたデフォルト値および「作成」フィールドのデフォルト名を受け入れます。

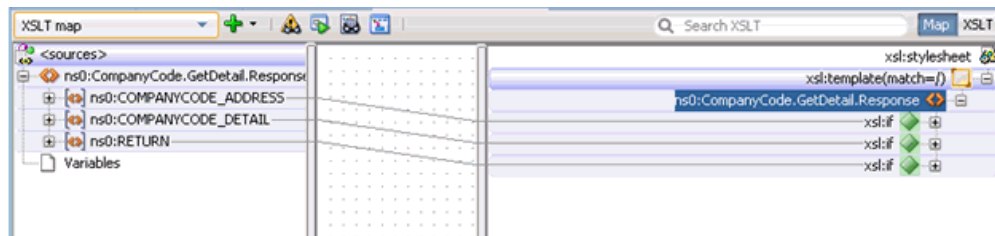
図 8-36 に示すように、XSL トランスフォーメーションが作成された状態で「データ・アソシエーション」ダイアログ・ウィンドウに戻ります。

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、「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成します。

図 8-37 完成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。
「ServiceTask」コンポーネントと「終了」イベント・コンポーネント間に「ServiceTask1」コンポーネントが作成されます。
36. メニュー・バーで「すべて保存」アイコンをクリックし、構成された新規のアウトバウンド 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 との既知のデプロイメント問題の調整

12c との既知のデプロイメント問題の調整方法については、6-27 ページ [6.4.3.3 項「12c との既知のデプロイメント問題の調整」](#) 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、次のステップを実行します

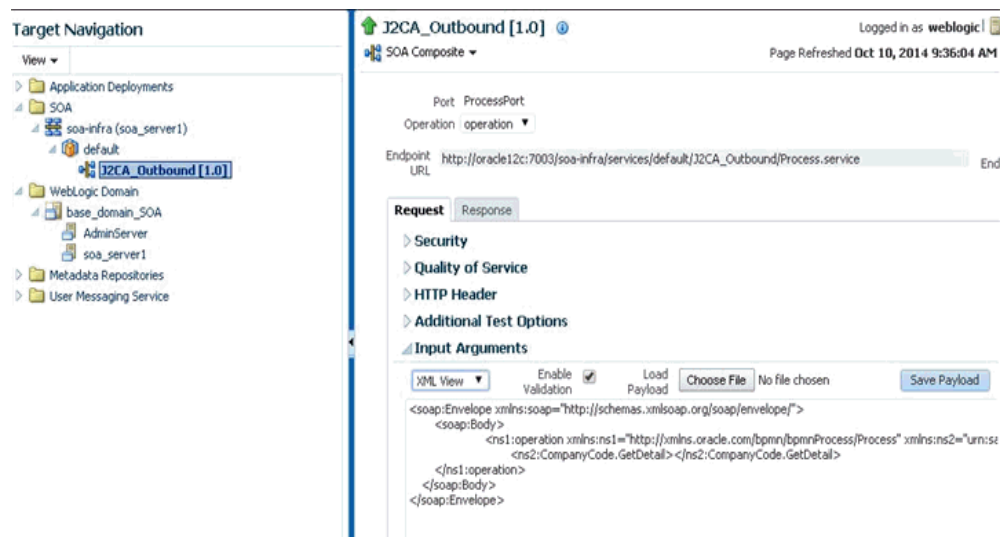
1. 左ペインのプロジェクト名を右クリックし、「**デプロイ**」を選択して、「**J2CA_Outbound**」をクリックします。
「デプロイメント・アクション」ページが表示されます。
2. 「**アプリケーション・サーバーにデプロイ**」が選択されていることを確認します。
3. 「**次へ**」をクリックします。
「構成のデプロイ」ページが表示されます。
4. デフォルト値を選択したままにし、「**次へ**」をクリックします。
「サーバーの選択」ページが表示されます。
5. 構成済の使用可能なアプリケーション・サーバーを選択し、「**次へ**」をクリックします。
「SOA サーバー」ページが表示されます。
6. ターゲットの 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. プロジェクトに使用可能なすべてのデプロイメント情報をレビューおよび確認し、「**終了**」をクリックします。
詳細については、[を参照してください。6-30 ページ 6.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」](#)

8.4.5 Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動

Oracle Enterprise Manager コンソールで入力 XML ドキュメントを起動するには、次のステップを実行します

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、「**SOA**」フォルダの順に展開します。
3. 使用可能なプロジェクト (「**Mysap_jca_outbound**」など) を選択します。
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。

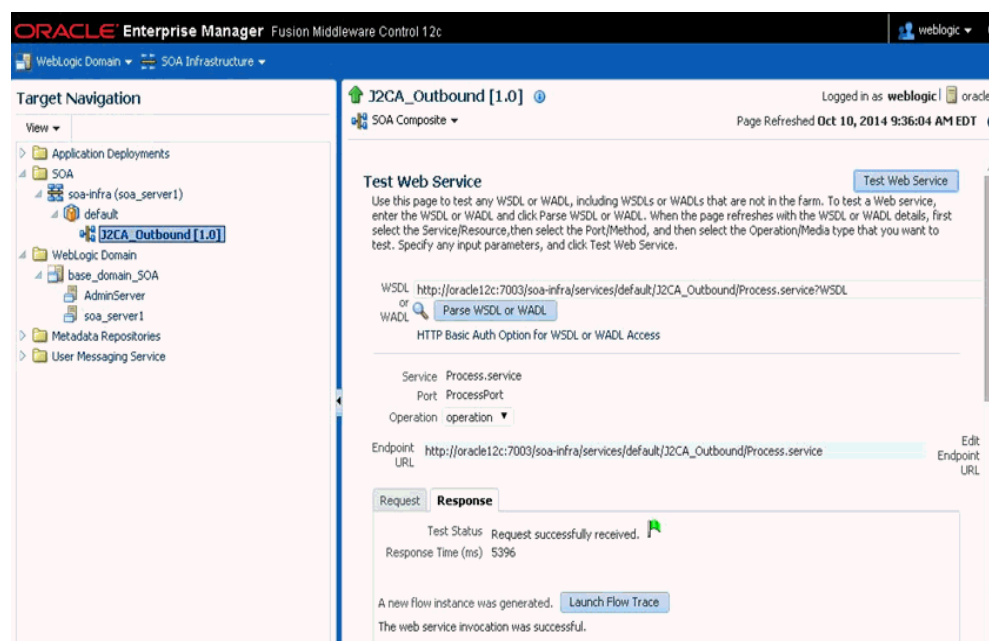
図 8-38 「リクエスト」タブ



6. 図 8-38 に示すように、「値」フィールドに適切な入力値を入力し、「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、Oracle Enterprise Manager コンソールで起動に成功したことを示す、レスポンスが「レスポンス」タブで受信されます。

図 8-39 受信されたレスポンス



7. ファイル・システムの定義出力カディレクトリにナビゲートし、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、値と生成された出力が含まれます。例を以下に示します。

```
<?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: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>000000121</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。

Application Adapters インストールの次のフォルダに、このインバウンドの使用ケース・シナリオのサンプル・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
アダプタの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 アプリケーション・エクスプローラ (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPM デザイナ (JDeveloper)

注意： この章に記載されている例では JDeveloper が使用されています。

この節では、次のトピック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。詳細については、6-37 ページ「[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 での新規のアプリケーション・サーバー接続の構成の詳細は、6-2 ページ [6.3 項「新規のアプリケーション・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、次のトピックで構成される BPM インバウンド・プロセスの定義方法について説明します。

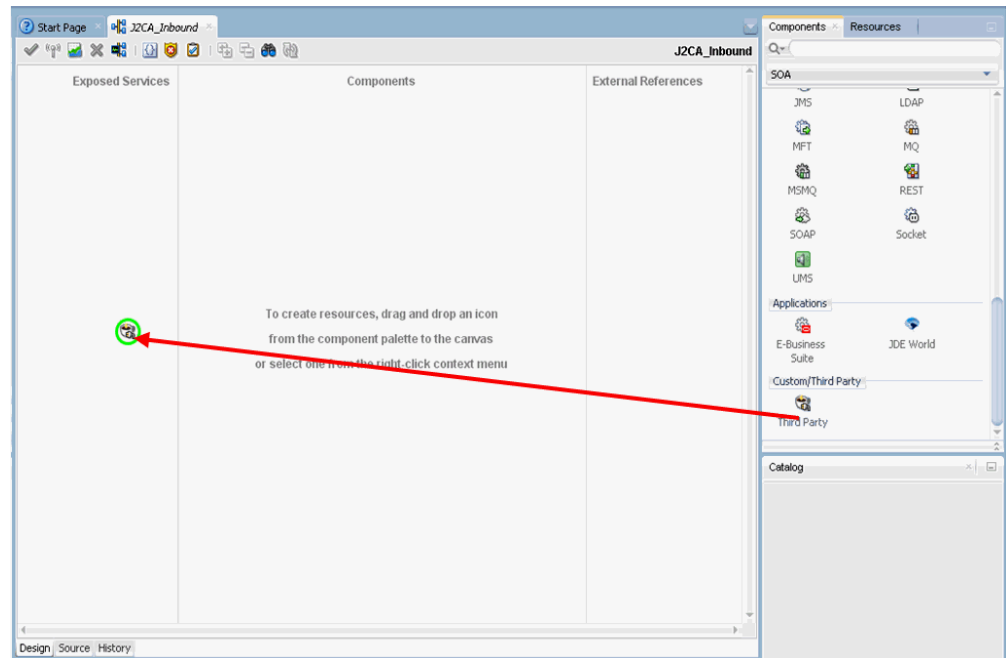
- [8.5.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」](#)
- [8.5.2.2 項「インバウンド BPM プロセス・コンポーネントの構成」](#)
- [8.5.2.3 項「書込み操作のファイル・アダプタの作成」](#)
- [8.5.2.4 項「12c との既知のデプロイメント問題の調整」](#)

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、次のステップを実行します。

1. 作成し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、カスタム / サードパーティ・ペインから公開された参照ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドの右の既存の WSDL の検索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」を選択してから、次のディレクトリまでブラウズし、インバウンド WSDL ファイルを選択します。

```
<ADAPTER_HOME>\wsdl\
```

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 ファイルおよび関連 receive_request XML スキーマ・ファイル (.xsd) が、作成済のプロジェクト・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」フィールドの右の JCA ファイルの検索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」を選択してから、次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。

<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。

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

「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 (matmas) が公開された参照ペインに作成されます。

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。

詳細については、を参照してください。6-45 ページ [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」](#)

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

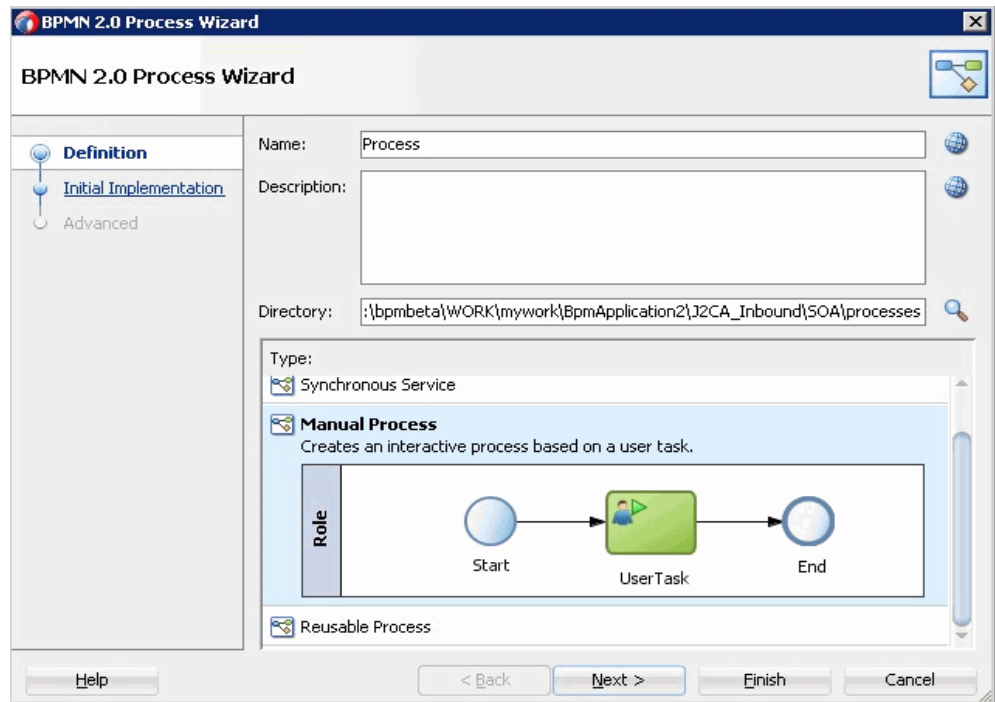
この項では、インバウンド BPM プロセス・コンポーネントを構成する方法について説明します。

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、次のステップを実行します。

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

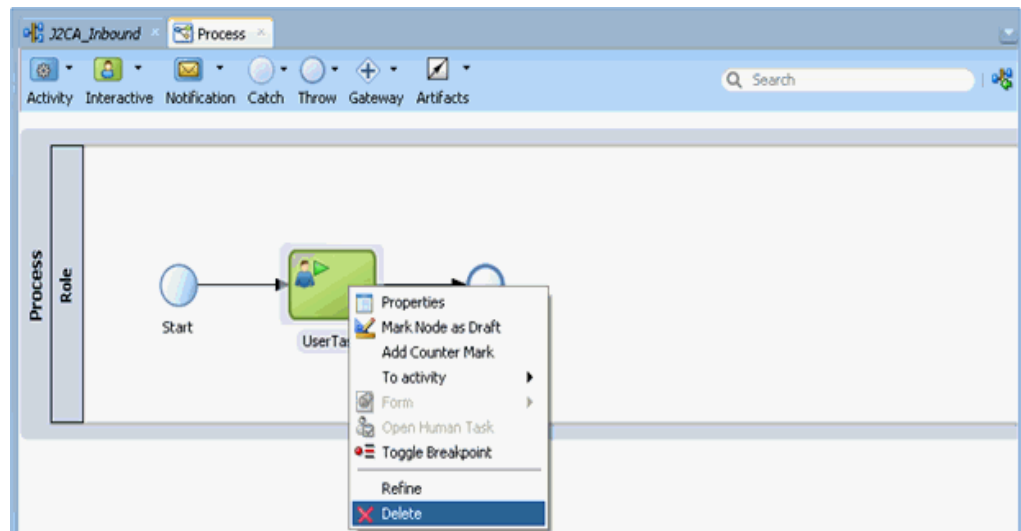
 [図 8-41](#) に示すように、「BPMN プロセスの作成」ダイアログが表示されます。

図 8-41 「BPMN プロセスの作成」ダイアログ



2. 「タイプ」セクションの「手動プロセス」を選択します。
3. 「終了」をクリックします。
 図 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. 「ユーザー・タスク」を右クリックし、メニューから「削除」を選択します。
5. 「開始」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 - 開始」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」リストから「メッセージ」を選択します。
8. メッセージ交換タイプ・ドロップダウン・リストから「インタフェースの使用」を選択します。
9. 図 8-43 に示すように、「参照」フィールドの右の「参照」アイコンをクリックします。

図 8-43 「参照」アイコン

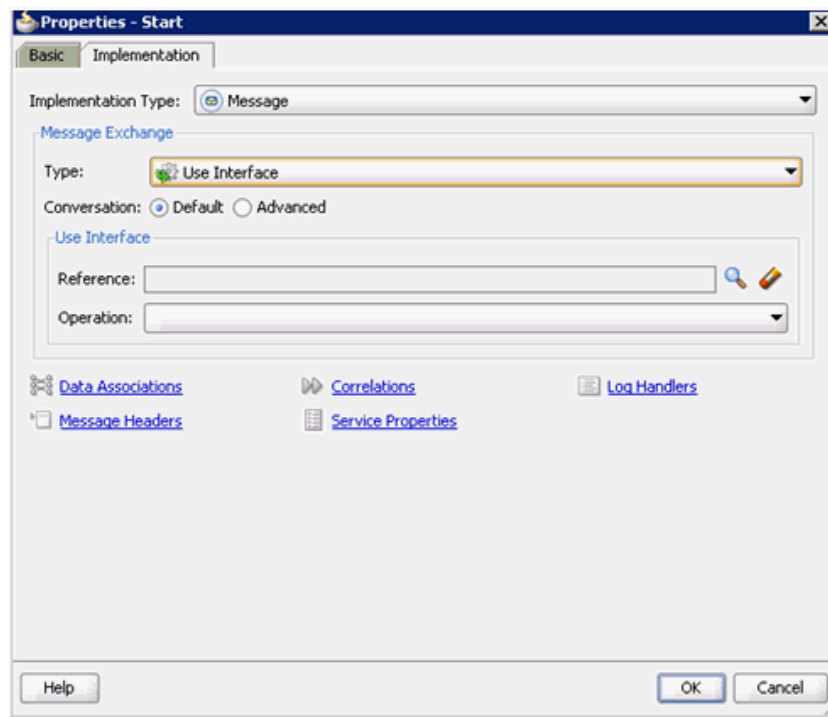
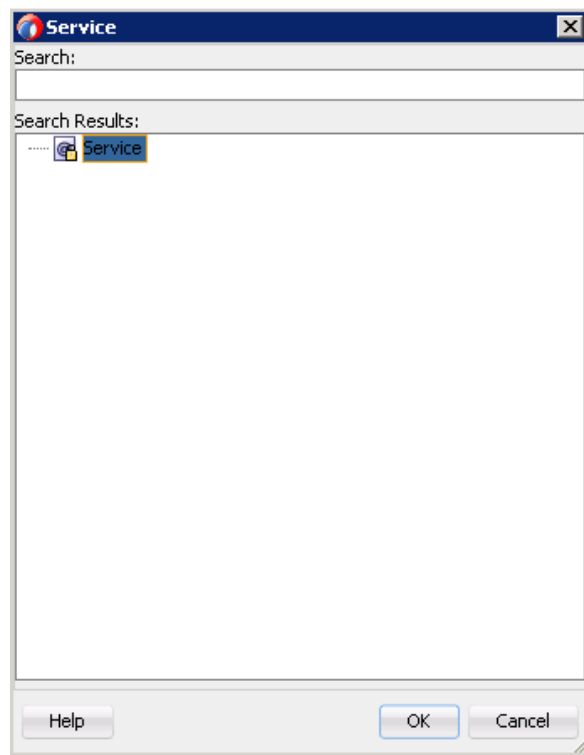


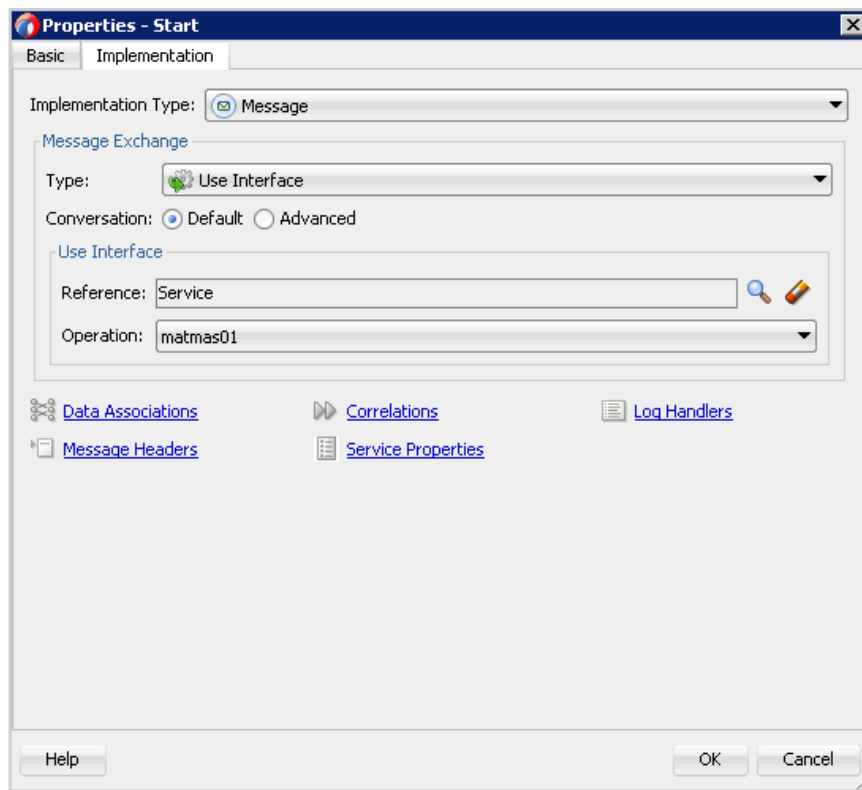
図 8-44 に示すように、「サービス」ダイアログが表示されます。

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。
図 8-45 に示すように、「プロパティ - 開始」ダイアログに戻ります。

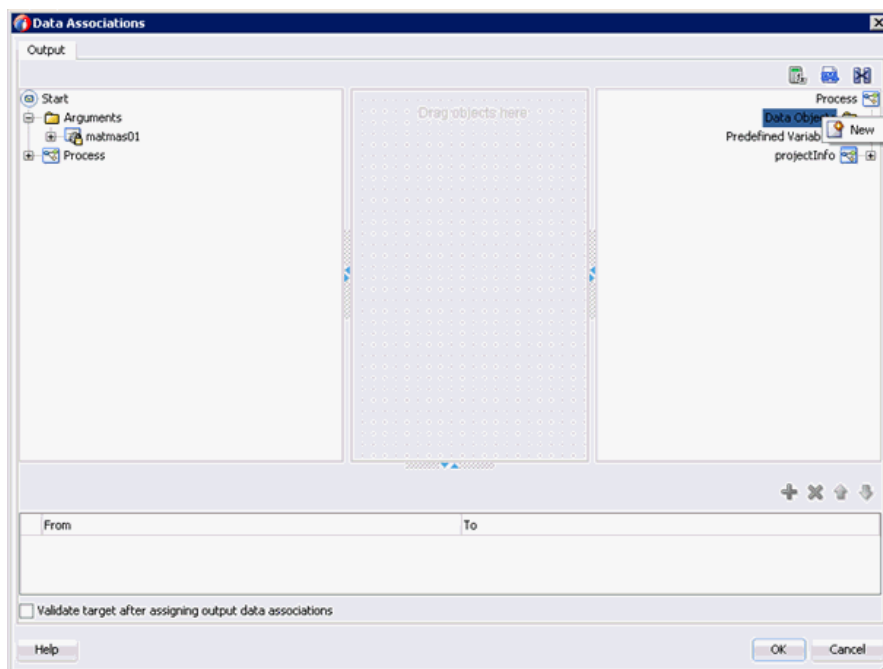
図 8-45 「プロパティ - 開始」 ダイアログ



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



- 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。

- 図 8-47 に示すように、「名前」フィールドで名前を入力し、「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、リストから「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

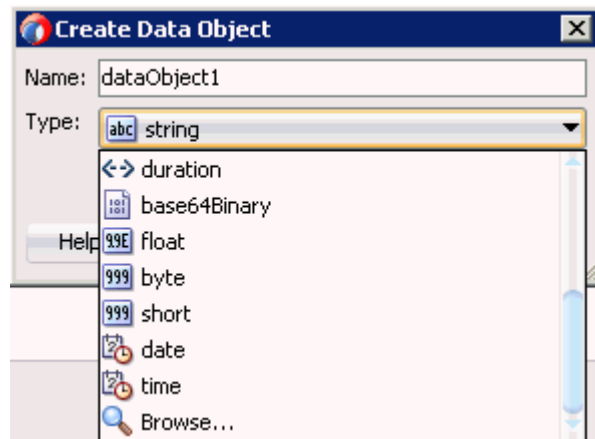
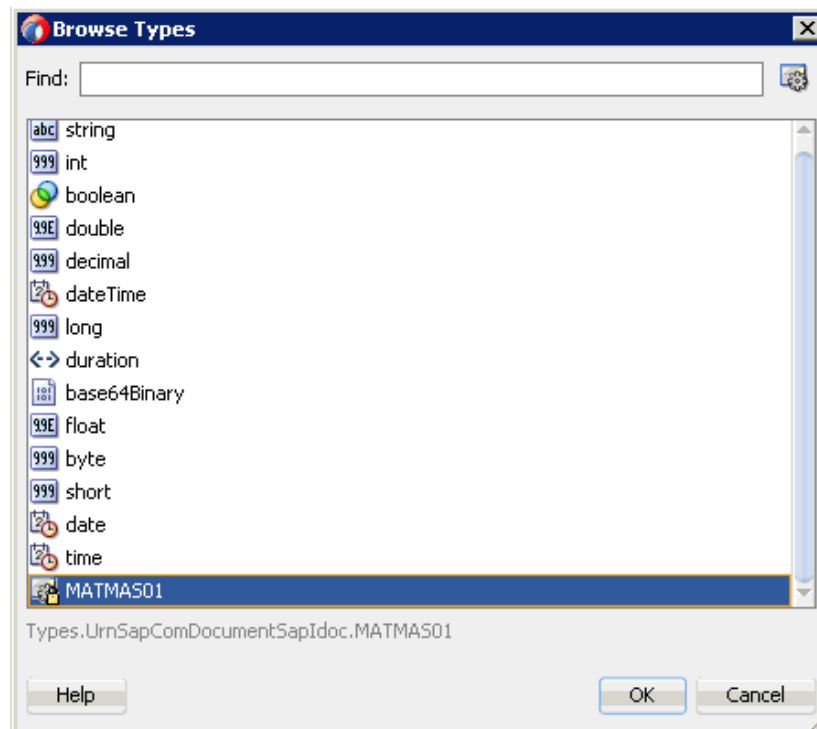


図 8-48 に示すように、「参照タイプ」ダイアログが表示されます。

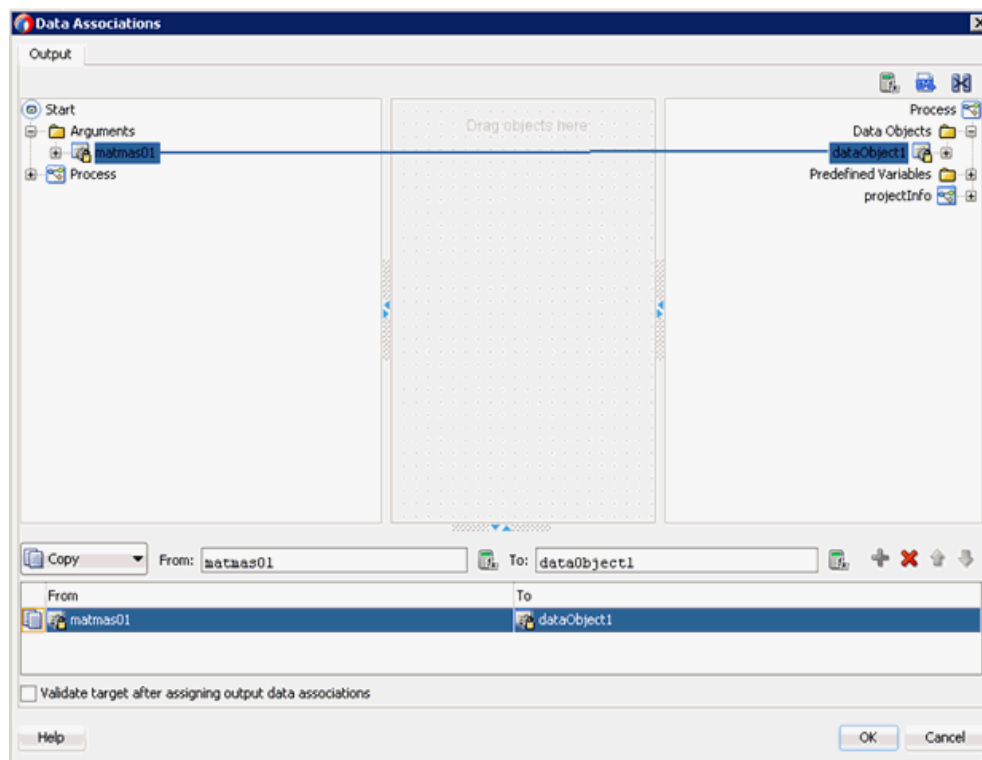
図 8-48 「参照タイプ」ダイアログ



- コンポーネントを選択して、「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。

図 8-49 に示すように、作成された「データ・オブジェクト」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。

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「開始」ノードの下の「matmas01」引数を選択してドラッグし、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 - 開始」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・バーで「すべて保存」アイコンをクリックし、構成された新規のインバウンド 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。

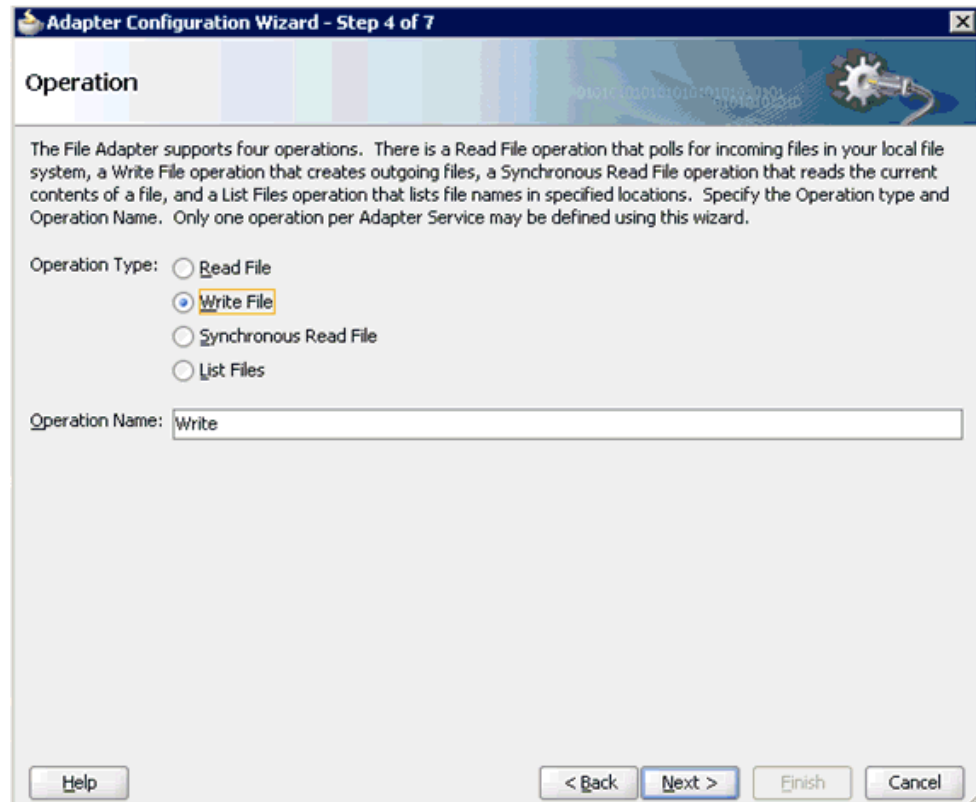
8.5.2.3 書込み操作作用のファイル・アダプタの作成

この項では、書込み操作作用のファイル・アダプタを作成する方法について説明します。書込み操作作用のファイル・アダプタを作成するには、次のステップを実行します。

1. 「テクノロジ・アダプタ」ペインから「外部参照」ペインに「ファイル・アダプタ」コンポーネントをドラッグ・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」フィールドに新規の「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。

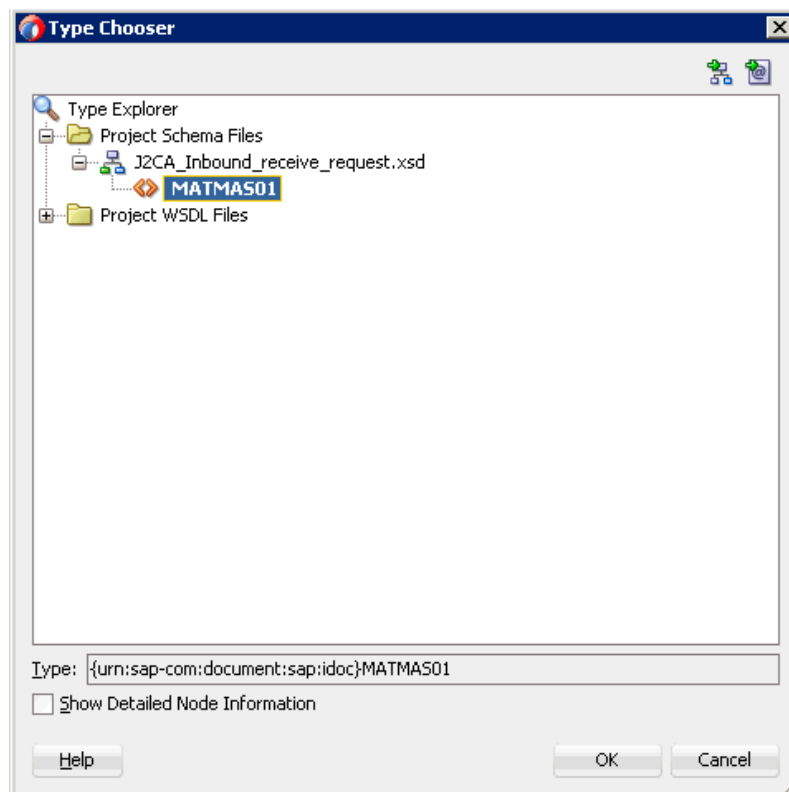
3. 「操作およびスキーマから定義 (後で指定)」オプションが選択されていることを確認します。
4. 「次へ」をクリックします。
「ファイル・サーバーの接続」ページが表示されます。
5. 「次へ」をクリックします。
図 8-50 に示すように、「操作」ページが表示されます。

図 8-50 「操作」ページ



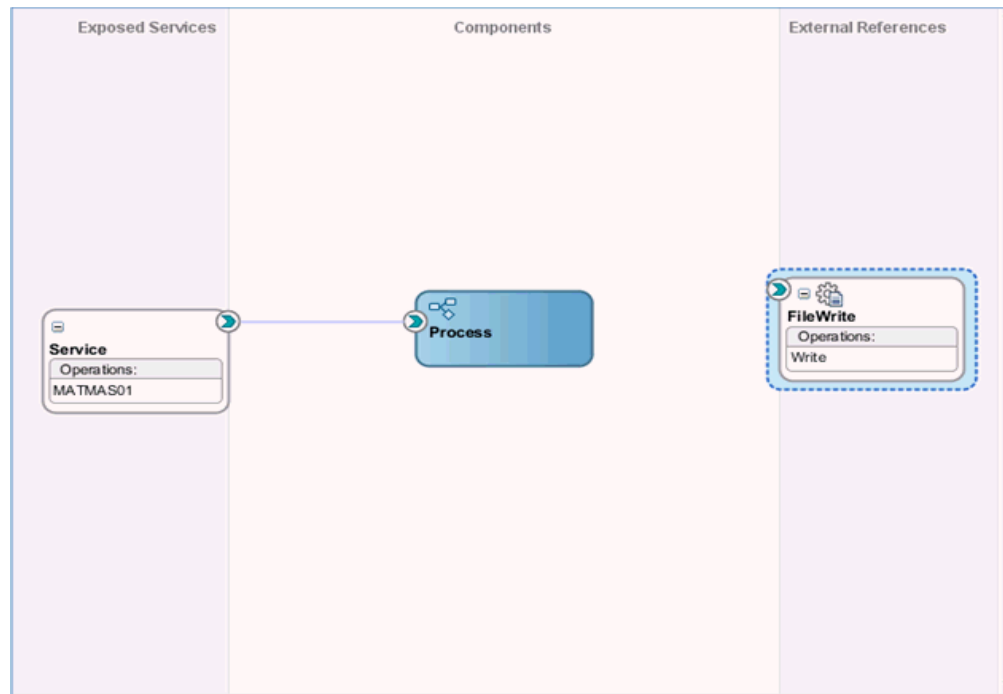
6. 「操作タイプ」オプションのリストから「Write File」を選択し、「操作名」(「Write」など)を指定します。
7. 「次へ」をクリックします。
「ファイル構成」ページが表示されます。
8. 出力ファイルの書込み先となるファイル・システム上の場所を指定します。
9. ファイル・ネーミング規則フィールドで、出力ファイルの名前を指定します。
10. 「次へ」をクリックします。
「メッセージ」ページが表示されます。
11. 「URL」フィールドの右の「参照」をクリックします。
図 8-51 に示すように、「タイプ・チューザ」ダイアログが表示されます。

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」の順に展開します。
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」ペインでファイル・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



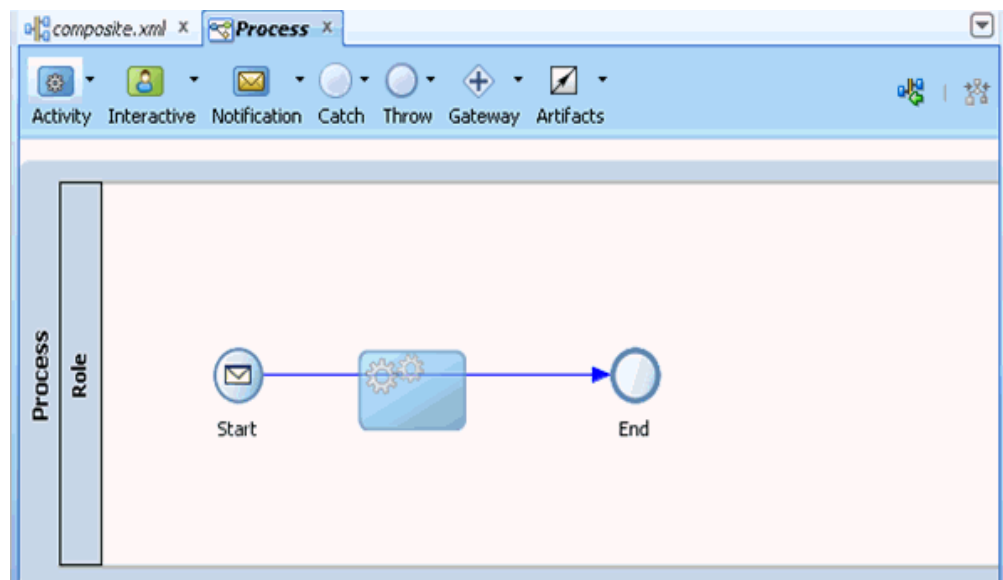
17. BPMN プロセス・コンポーネントをダブルクリックします。

「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、「開始」と「終了」イベント・コンポーネント間の伝送路に「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

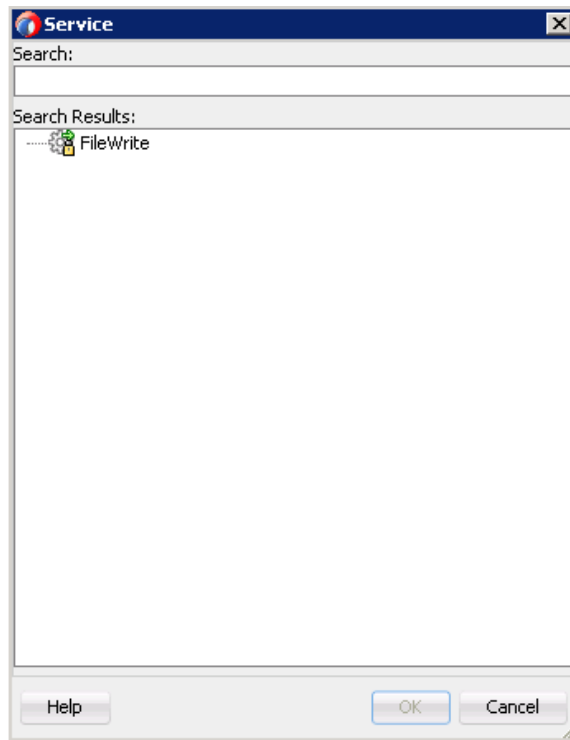


「プロパティ - ServiceTask」ダイアログが表示されます。

20. 「実装」タブをクリックします。

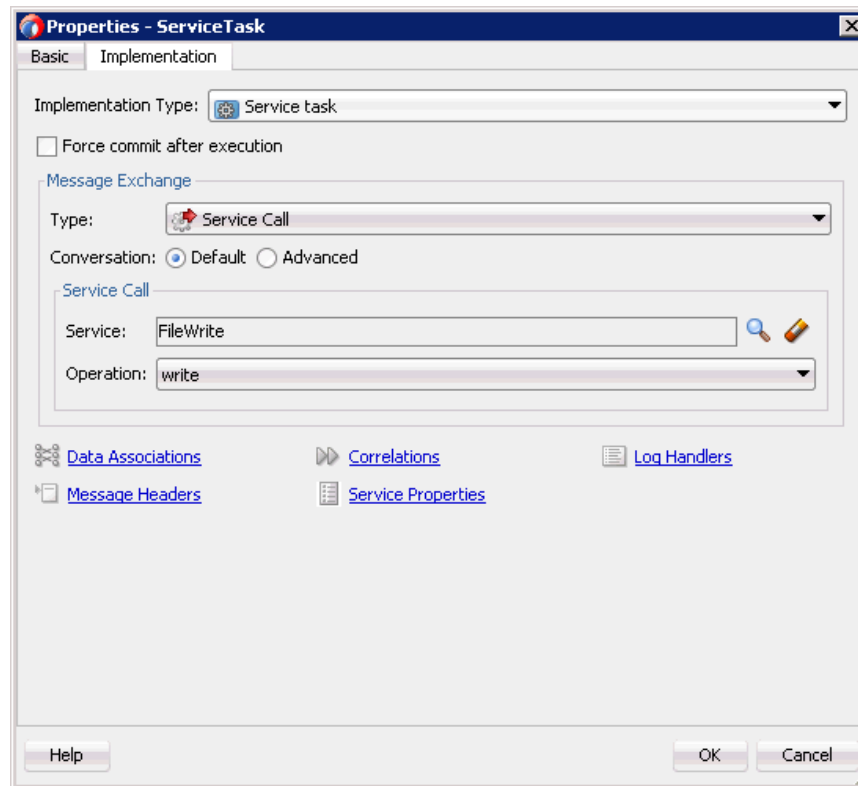
21. 「実装タイプ」リストから「サービス・タスク」を選択します。
22. メッセージ交換タイプ・リストから「サービス・コール」を選択します。
23. 「サービス」フィールドの右の「参照」アイコンをクリックします。
図 8-54 に示すように、「タイプ」ダイアログが表示されます。

図 8-54 「タイプ」ダイアログ



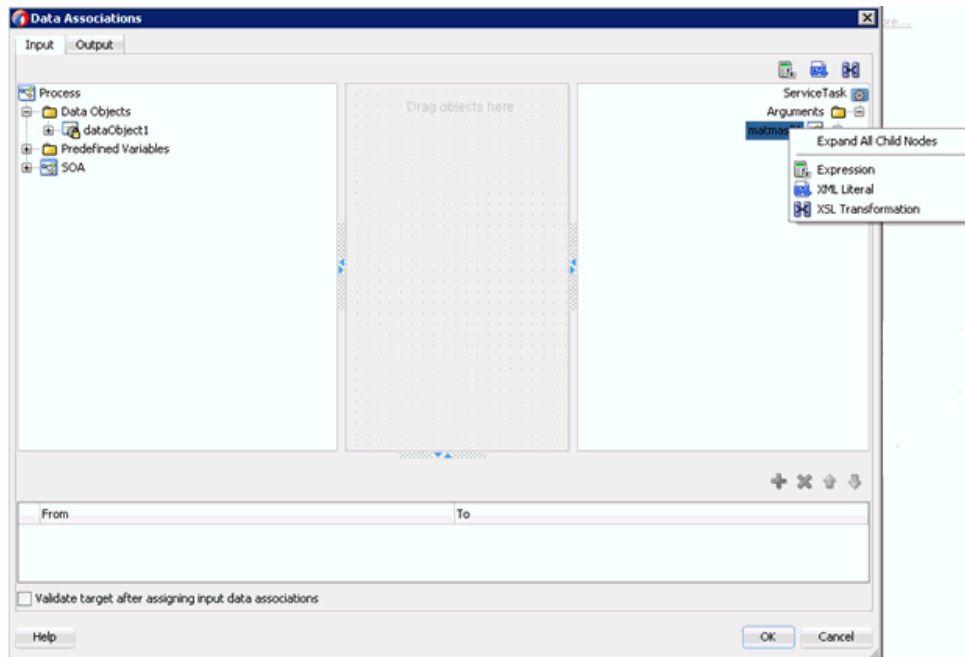
24. 作成された書込み操作のサービスを選択し、「OK」をクリックします。
図 8-55 に示すように、「プロパティ - ServiceTask」ダイアログに戻ります。

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、「XSL 変換」を選択します。

図 8-56 「XSL 変換」

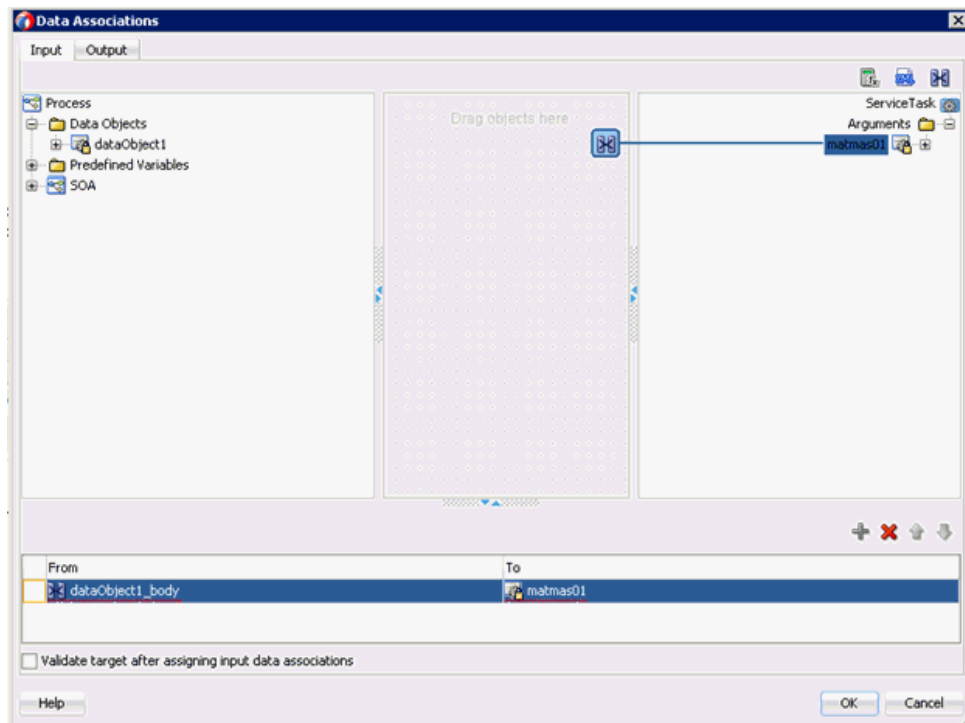


トランスフォーメーションの作成ダイアログが表示されます。

27. 作成済みデータ・オブジェクトが「選択済」要素領域に追加されるように、「ソース」領域で作成済みデータ・オブジェクトを選択し、右矢印アイコンをクリックします。
28. 「OK」をクリックします。

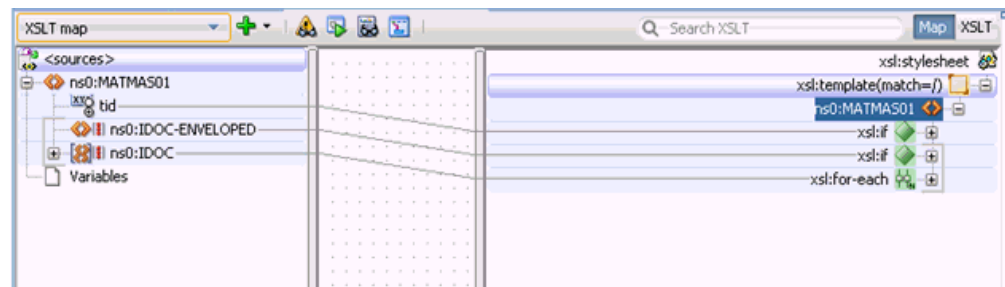
図 8-57 に示すように、「データ・アソシエーション」ダイアログに戻ります。

図 8-57 「データ・アソシエーション」ダイアログ



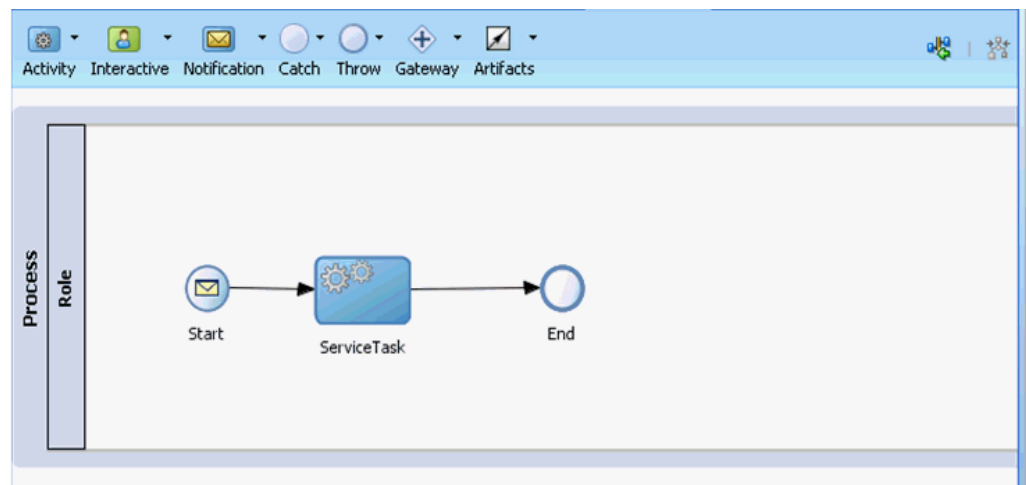
29. 「OK」をクリックします。
「プロパティ - ServiceTask」ダイアログに戻ります。
30. 「OK」をクリックします。
「dataobject1_body.xml」タブが表示されます。
31. ソース要素とターゲット要素を自動マップします。
「自動マップ・プリファレンス」ダイアログが表示されます。
32. デフォルト値を受け入れ、「OK」をクリックします。
図 8-58 に示すように、トランスフォーメーションが完成します。

図 8-58 完成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



「開始」イベント・コンポーネント間に「ServiceTask」が作成されます。

35. メニュー・バーで「すべて保存」アイコンをクリックし、構成された新規のインバウンド BPM プロセス・コンポーネントを保存します。

8.5.2.4 12c との既知のデプロイメント問題の調整

12c との既知のデプロイメント問題の調整方法については、6-27 ページ [6.4.3.3 項「12c との既知のデプロイメント問題の調整」](#) 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。6-53 ページ「[BPEL インバウンド・プロセスのデプロイ](#)」で説明したのと同じ手順に従うことができます。SAP R/3 でイベントをトリガーする方法の詳細については、6-54 ページ「[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 イベントの作業 (J2CA 構成)

この項では、BPM を使用して同期 SAP R/3 イベントを作業する方法について説明します。同期 SAP R/3 イベントの詳細については、[6.6.1 項「概要」](#)を参照してください。

注意： 同期イベントはリプライ・スキーマのみを所有しているオブジェクトでサポートされ、IDoc ではサポートされていません。

Application Adapters インストレーションの次のフォルダに、この使用ケース・シナリオのサンプル・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
```

この節では、次のトピックについて説明します。

- [8.6.1 項「アダプタのターゲットおよびチャンネルの構成」](#)
- [8.6.2 項「メッセージ用の 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 アダプタのターゲットおよびチャンネルの構成

アプリケーション・エクンプローラを開き、J2CA 構成を使用して SAP に対して 2 つのターゲットと 1 つのチャンネルを作成します。

詳細については、[第 4 章「Oracle Application Adapter for SAP R/3 の構成」](#)を参照してください。

8.6.2 メッセージ用の WSDL ファイルの生成

次のステップを実行して、メッセージ用の WSDL ファイルを生成します。

1. WSDL ファイルを生成する前に、チャンネルが開始し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 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 (「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)。

詳細については、[6.4.1 項「リクエスト / レスポンス型のサービス用の 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 用に空のコンポジットを作成するには、次のステップを実行します。

1. 新規の BPM アプリケーションを作成します。
2. 新規の BPM アプリケーションに名前を入力し、「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て、「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。

詳細については、6-9 ページ 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 ファイル用のサード・パーティ・アダプタ・サービスの作成

インバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスを作成するには：

1. 「カスタム Ä/ サード・パーティ」領域から「公開されたサービス」ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドで名前を入力し、「WSDL URL」フィールドの横の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」タブを選択してから、WSDL の場所からインバウンド WSDL ファイルをブラウズして選択し、「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) が作成済みのプロジェクト・フォルダにインポートされます。
5. **JCA ファイルの検索**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・ファイルを選択して、「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。
JCA プロパティ・ファイルのコピーがプロジェクト・フォルダ内に作成されます。
8. 「OK」をクリックします。

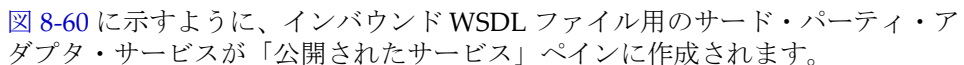
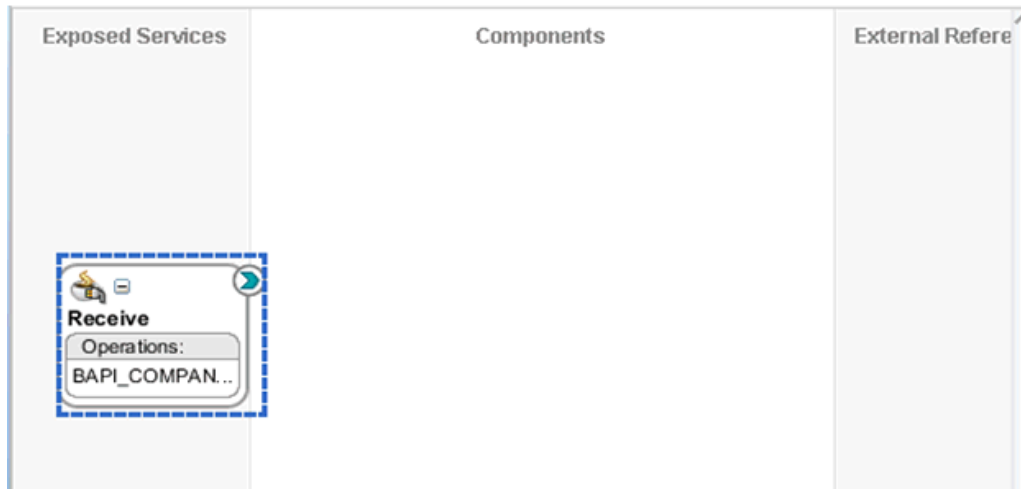
 8-60 に示すように、インバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスが「公開されたサービス」ペインに作成されます。

図 8-60 「公開されたサービス」 ペイン



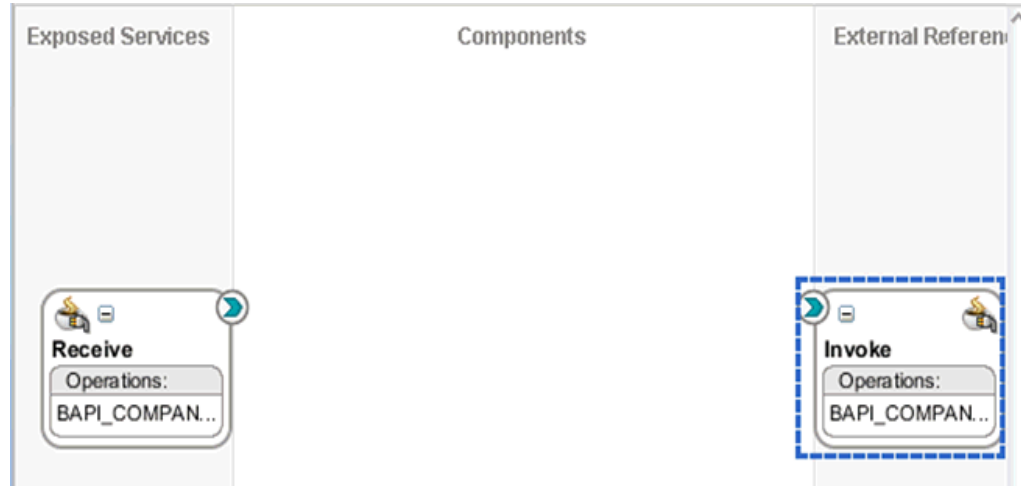
8.6.3.3 アウトバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスの作成

アウトバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスを作成するには:

1. 「カスタム/サード・パーティ」領域から「外部参照」ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドで名前を入力し、「WSDL URL」フィールドの横の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 の場所から作成されたアウトバウンド WSDL をブラウザして選択し、「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) が作成済みのプロジェクト・フォルダにインポートされます。
5. JCA ファイルの検索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・ファイルを選択して、「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。
JCA プロパティ・ファイルのコピーがプロジェクト・フォルダ内に作成されます。
9. 「OK」をクリックします。

図 8-61 に示すように、アウトバウンド WSDL ファイル用のサード・パーティ・アダプタ・サービスが「外部参照」ペインに作成されます。

図 8-61 「外部参照」ペイン

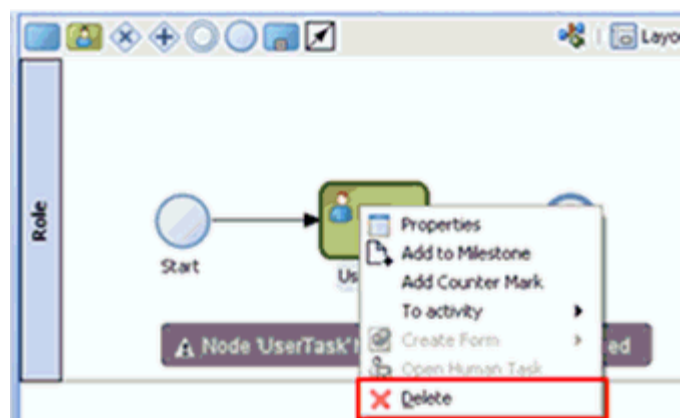


8.6.3.4 同期 BPM プロセスの作成

同期 BPMN プロセスを作成するには：

1. 「コンポーネント」領域から「コンポーネント」ペインに「BPMN プロセス」コンポーネントをドラッグ・アンド・ドロップします。
「BPMN 2.0 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」セクションの「手動プロセス」を選択してから、「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、「ユーザー・タスク」アイコンを右クリックし、メニューから「削除」を選択します。

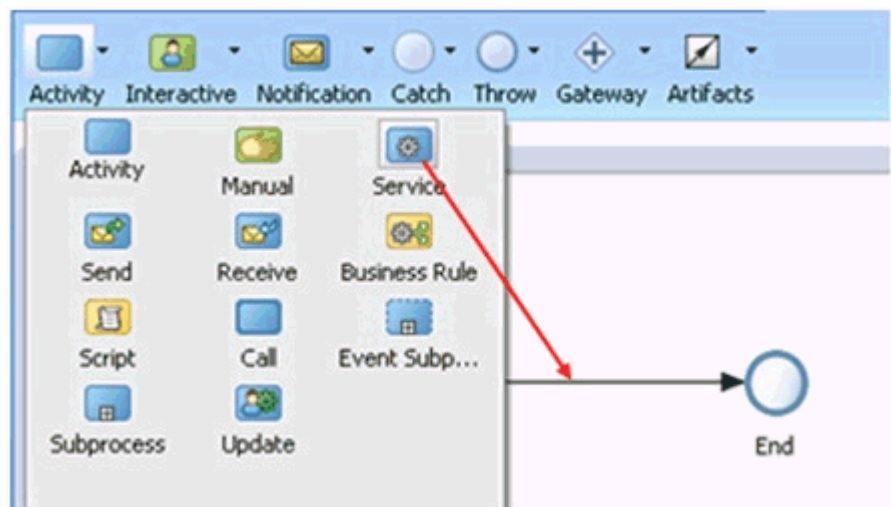
図 8-62 「削除」オプション



4. 「開始」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 - 開始」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」リストから「メッセージ」をクリックし、「メッセージ交換」タブの下の「タイプ」リストから「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」フィールドの右の「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 - 開始」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 (「receive_in」など) を入力します。
 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、リストから「参照」を選択します。
「参照タイプ」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て、「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。
作成された「データ・オブジェクト」(「receive_in」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」の下に表示されます。
 15. 左ペインの「引数」ノードの下に一覧表示されたコンポーネントを選択し、右ペインの「データ・オブジェクト」の下の作成済みデータ・オブジェクトにドラッグして接続します。
 16. 「OK」をクリックします。
「プロパティ - 開始」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. 図 8-63 に示すように、「開始」と「終了」イベント・コンポーネント間の伝送路にサービス・アクティビティ・アイコンを選択してドロップします。

図 8-63 サービス・アクティビティ・アイコン

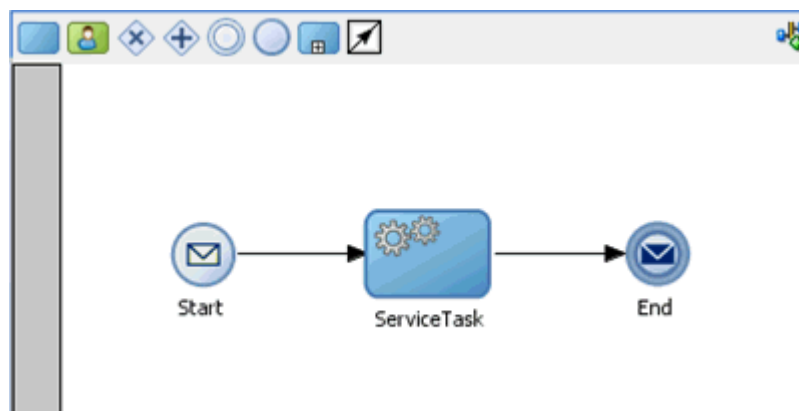


「プロパティ - ServiceTask」ダイアログが表示されます。

19. 「実装」タブをクリックしてから、「メッセージ交換」タブの下の「タイプ」リストから「サービス・コール」を選択します。
20. 「サービス」フィールドの右の「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 - 開始」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」領域の下で、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、リストから参照 ... を選択します。
「参照タイプ」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て、「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。
作成された「データ・オブジェクト」(「invoke_in」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」の下に表示されます。
27. 左ペインの「データ・オブジェクト」の下の作成済みデータ・オブジェクトを選択し、右ペインの「引数」ノードの下に一覧表示されたコンポーネントにドラッグして接続します。
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。
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、リストから参照 ... を選択します。
「参照タイプ」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て、「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。
作成された「データ・オブジェクト」(「invoke_out」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」の下に表示されます。
32. 左ペインの「引数」ノードの下に一覧表示されたコンポーネントを選択し、右ペインの「データ・オブジェクト」の下の作成済みデータ・オブジェクト(「Response」)にドラッグして接続します。
33. 「OK」をクリックします。
「プロパティ - 開始」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. 「終了」 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 - 終了」 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」 リストから「メッセージ」を選択し、「メッセージ交換」 タブの下の「タイプ」 リストから「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」 の下の「タイプ」 フィールドの「同期」 ボタンを選択します。
 39. 「インタフェースの使用」 タブの下の「参照」 フィールドの右の「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」の下に表示されたレスポンス (「invoke_out」 など) の作成済みデータ・オブジェクトを選択し、右ペインの「引数」の下に一覧表示されたコンポーネントにドラッグして接続します。
 43. 「OK」 をクリックします。
「プロパティ - 終了」 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



同期イベントに対して作成された BPM プロセスが完成しました。

8.6.3.5 12c との既知のデプロイメント問題の調整

12c での既知のデプロイメント問題の調整方法については、6-27 ページの [6.4.3.3 項「12c との既知のデプロイメント問題の調整」](#) 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については、8-25 ページ [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 システム用に 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。

Application Adapters インストールの次のフォルダに、このアウトバウンドの使用ケース・シナリオのサンプル・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
```

アウトバウンドの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 アプリケーション・エクスプローラ (アプリケーション・エクスプローラ)
- Oracle JDeveloper BPM デザイナ (JDeveloper)

この節では、次のトピック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。詳細については、[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 用に空のコンポジットを作成するには、以下の手順を実行します。

1. 新規の BPM アプリケーションを作成します。
2. BPM アプリケーションに名前を入力し、「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。

詳細については、6-9 ページ [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。次のトピックについて説明します。

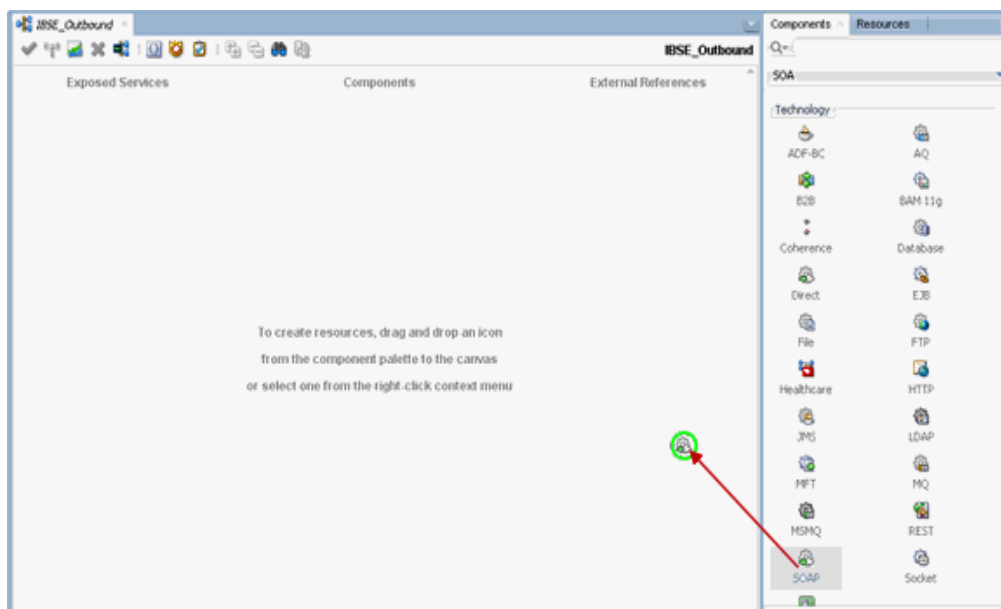
- [8.7.2.1 項「Web サービス・コンポーネントの構成」](#)
- [8.7.2.2 項「BPM プロセス・コンポーネントの構成」](#)
- [8.7.2.3 項「書込み操作のファイル・アダプタの作成」](#)

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、次のステップを実行します。

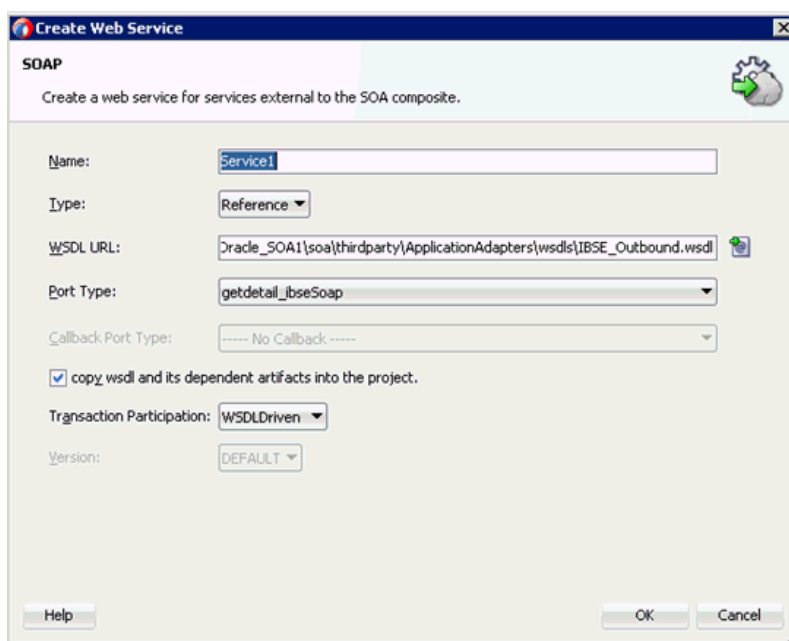
1. 作成したプロジェクトをダブルクリックして、コンポーネントをロードします。
2. [図 8-65](#) に示すように、「テクノロジ・アダプタ」ペインから「外部参照」ペインに「Web サービス」ノードをドラッグ・アンド・ドロップします。

図 8-65 「Web サービス」 ノード



3. 「Web サービス」に適切な名前を入力し、「WSDL URL」フィールドの右の**既存の WSDL の検索**アイコンをクリックします。
4. 表示された「WSDL の選択」ウィンドウで、WSDL がアプリケーション・エクスプローラからエクスポートされた場所にナビゲートして、WSDL を選択します。
5. 「OK」をクリックします。
6. 図 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. 表示された「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。

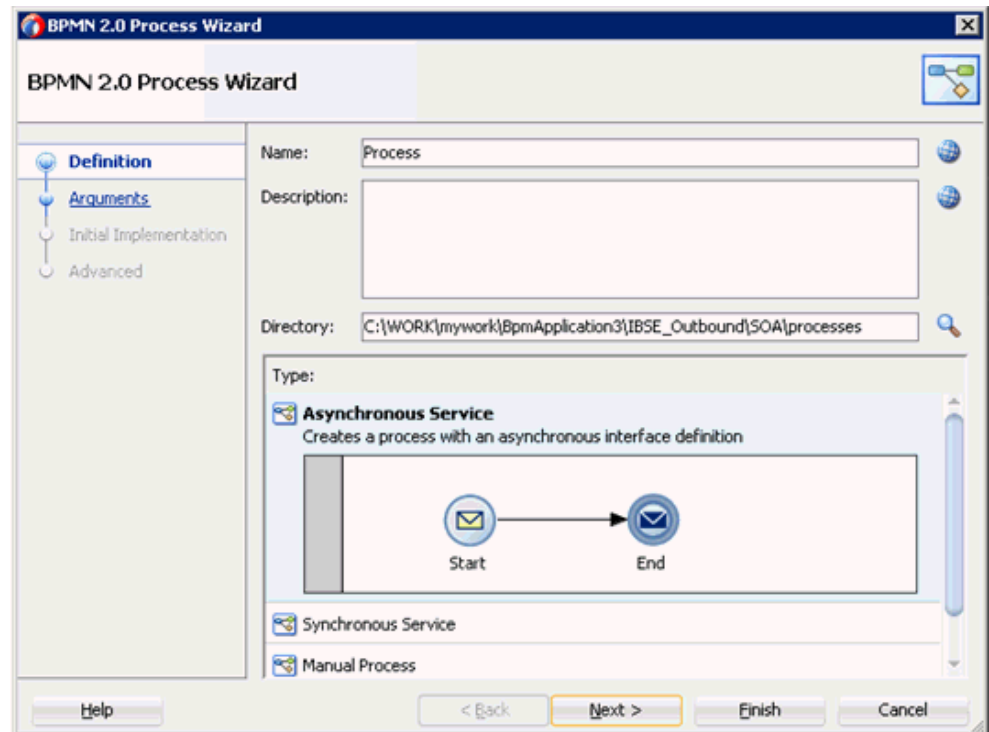
8.7.2.2 BPM プロセス・コンポーネントの構成

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、次のステップを実行します。

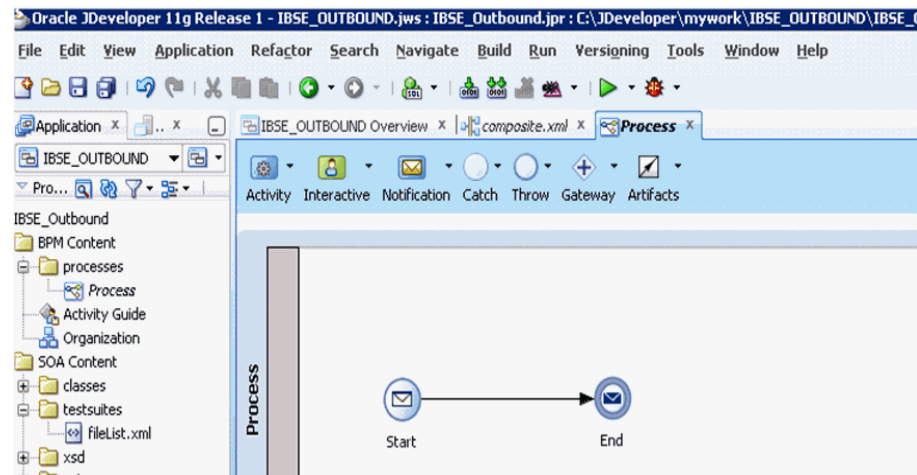
1. コンポーネント」ペインから「コンポーネント」ペインに「BPMN プロセス」コンポーネントをドラッグ・アンド・ドロップします。
2. 図 8-67 に示すように、「タイプ」領域(「非同期サービス」)で選択されたデフォルトのオプションを受け入れ、「終了」をクリックします。

図 8-67 「タイプ」領域



3. 図 8-68 に示すように、「開始イベント」コンポーネントをダブルクリックします。

図 8-68 「開始イベント」コンポーネント



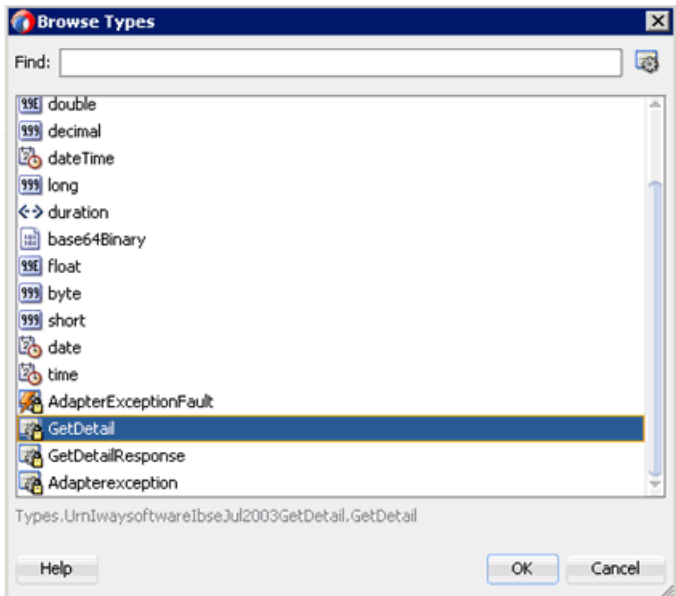
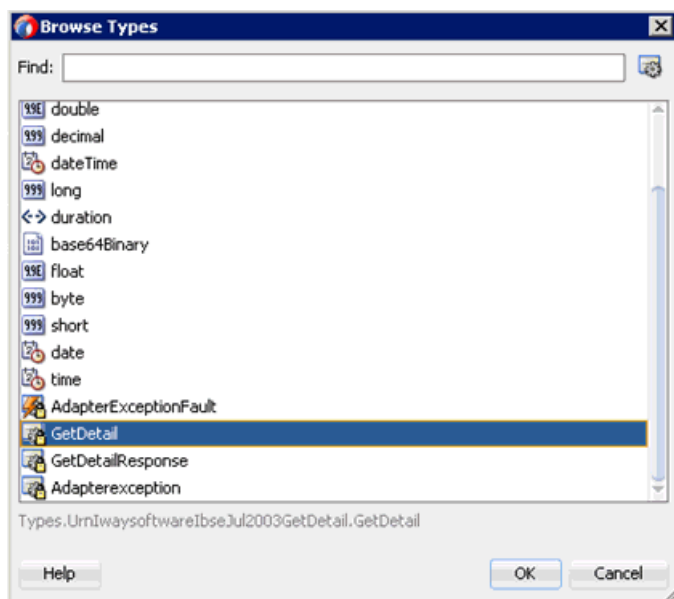
4. 「プロパティ - Start」 ウィンドウで、「**実装**」 タブをクリックします。
5. 「引数の定義」 フィールドの右のプラス (+) アイコンをクリックします。
「引数の編集」 ウィンドウが表示されます。
6. 「名前」 フィールドで名前を入力してから、「タイプ」 ドロップダウン・リストをクリックし、「**参照**」 を選択します。
7.  8-69 に示すように、「リクエスト」 コンポーネント (「GetDetail」 など) を選択し、「**OK**」 をクリックします。

図 8-69 「リクエスト」 コンポーネント




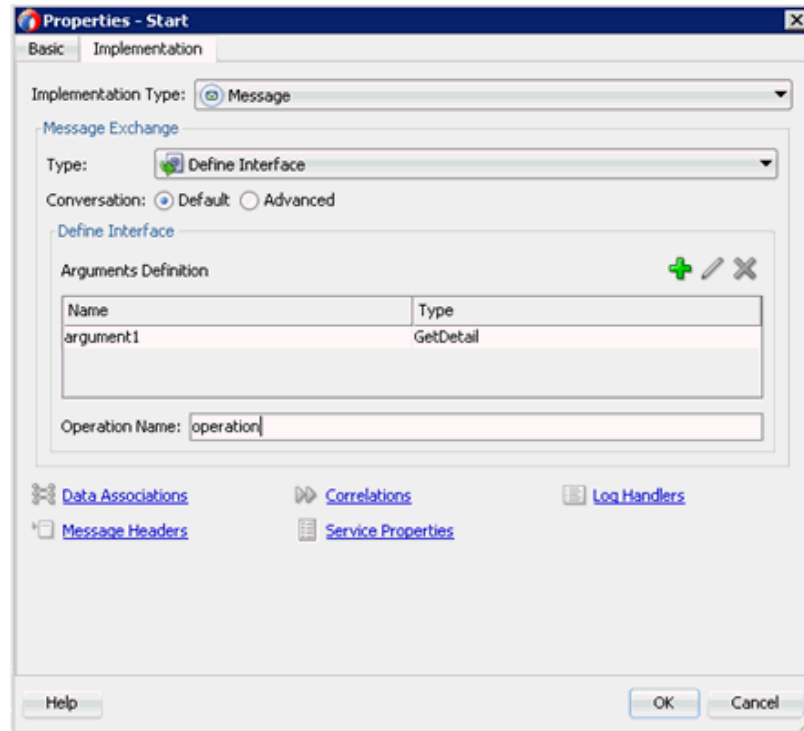
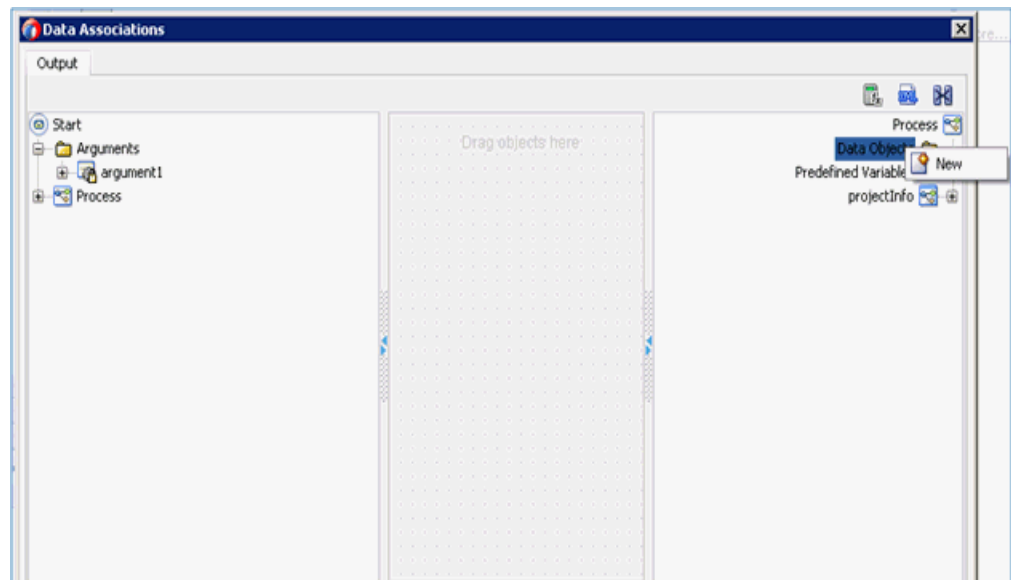
8. 表示された「引数の編集」 ウィンドウで、「**OK**」 をクリックします。
「プロパティ - 開始」 ウィンドウが表示されます。
9. 「操作名」 フィールドで、デフォルト・エントリを「**start**」 から「**operation**」 に変更します。
10.  8-70 に示すように、「**データ・アソシエーション**」 ハイパーリンクをクリックします。

図 8-70 「プロパティ - 開始」 ウィンドウ



11. 図 8-71 に示すように、「プロセス」の下の右ペインの「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。

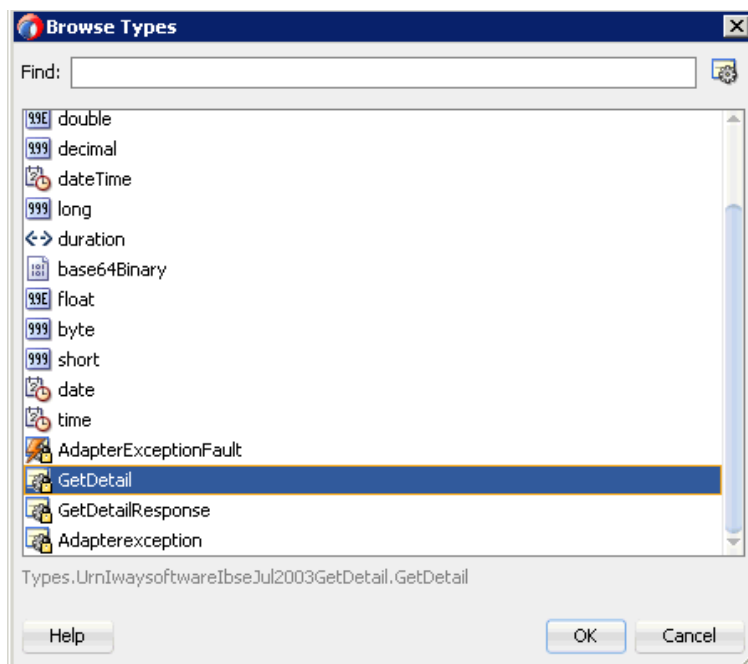
図 8-71 「データ・オブジェクト」ノード



データ・オブジェクトの作成ウィンドウが表示されます。

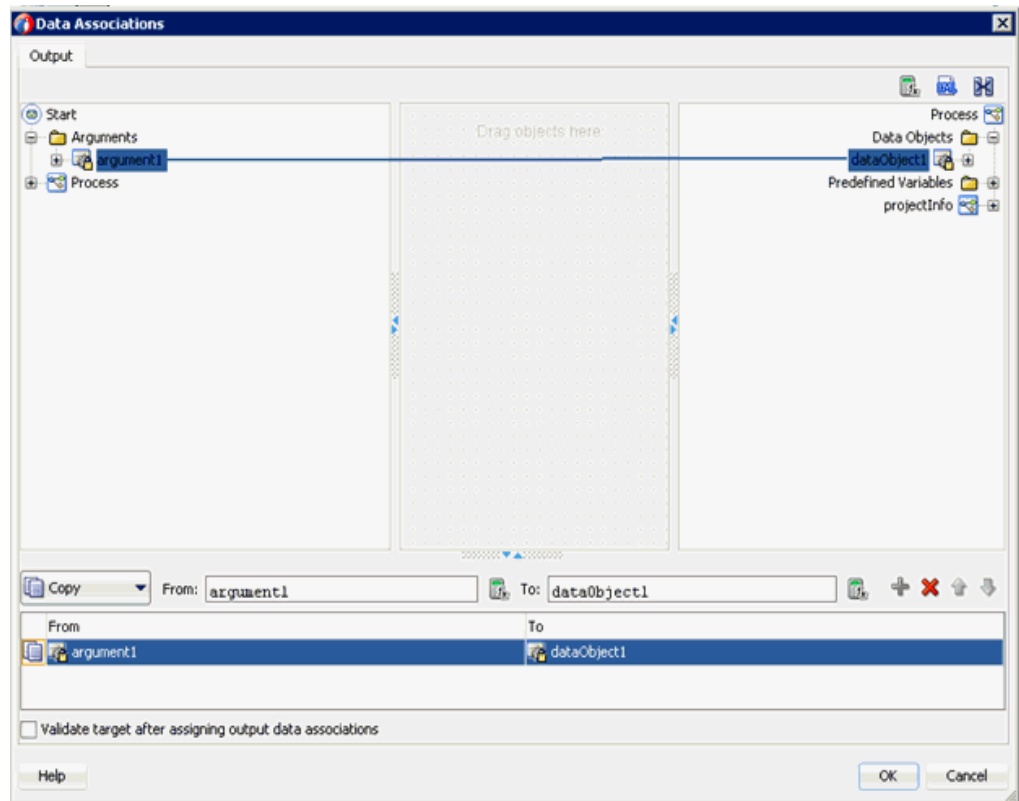
12. 「名前」フィールドで名前を入力してから、「タイプ」ドロップダウン・リストをクリックし、「参照」を選択します。
13. 図 8-72 に示すように、「リクエスト」コンポーネント（「GetDetail」など）を選択し、「OK」をクリックします。

図 8-72 「リクエスト」コンポーネント(「GetDetail」)



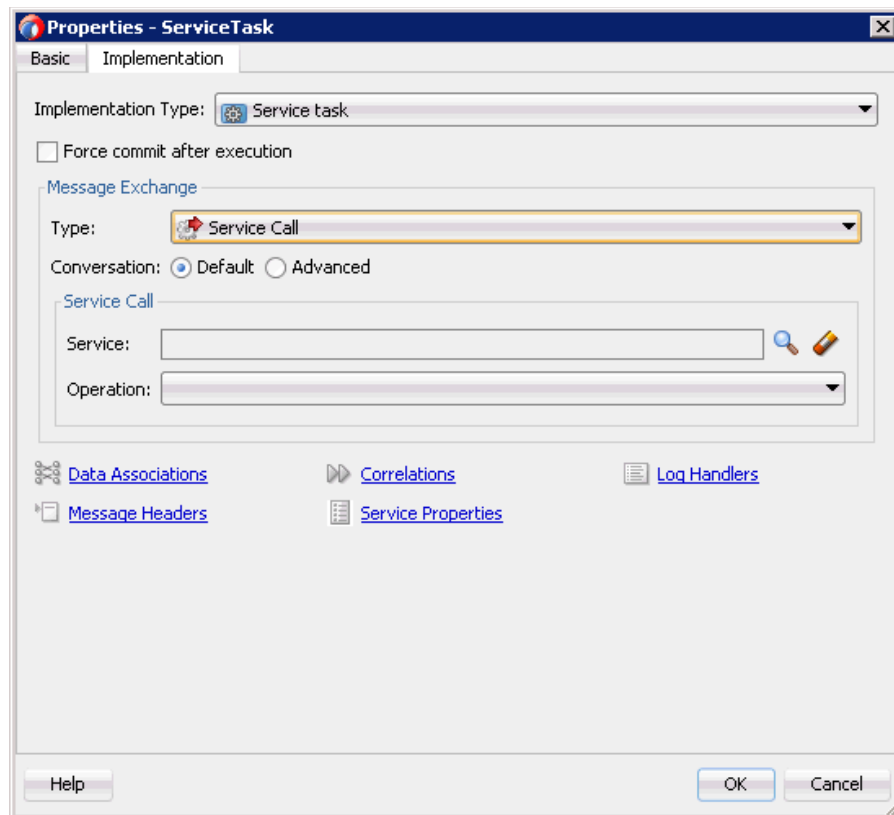
14. データ・オブジェクトの作成ウィンドウで、「OK」をクリックします。
「データ・アソシエーション」ウィンドウが表示されます。
15. 左ペインの「引数」ノードの下の「argument1」を選択し、右ペインの「データ・オブジェクト」の下の「dataObject1」にドラッグして接続します。
16. 図 8-73 に示すように、「OK」をクリックします。

図 8-73 「データ・アソシエーション」



17. 表示された「プロパティ - 開始」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て、「サービス」を選択します。
19. 「開始」と「終了」イベント・コンポーネント間の伝送路に「サービス」アイコンをドロップします。
20. 表示された「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. メッセージ交換タイプ・リストから「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」フィールドの右の「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.  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」

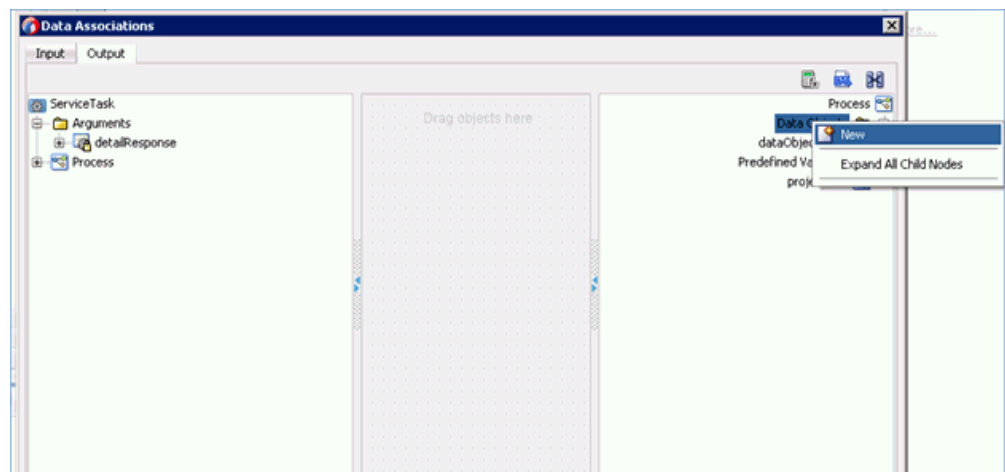


24. 表示された「プロパティ - ServiceTask」ウィンドウで、「データ・アソシエーション」ハイパーリンクをクリックします。

「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」を選択して、「レスポンス」データ・オブジェクトを作成します。

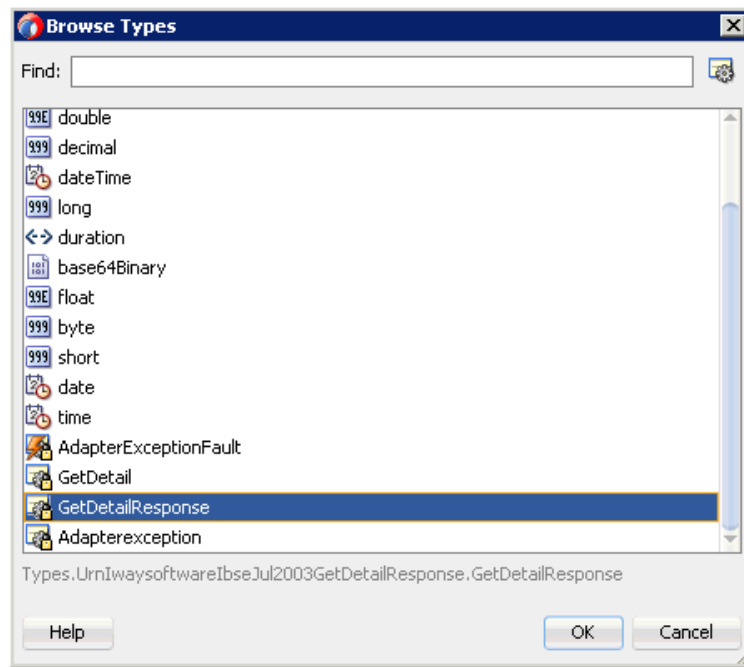
図 8-76 「データ・オブジェクト」ノード



データ・オブジェクトの作成ウィンドウが表示されます。

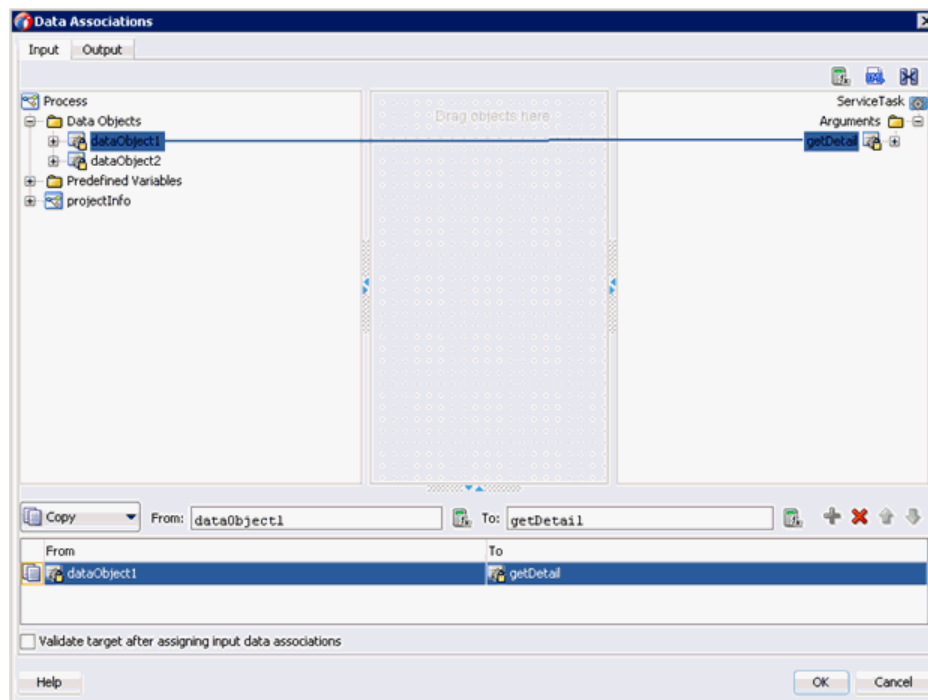
26. 「名前」フィールドで名前を入力してから、「タイプ」ドロップダウン・リストをクリックし、「参照」を選択します。
27. 図 8-77 に示すように、「レスポンス」コンポーネント（「GetDetailResponse」など）を選択し、「OK」をクリックします。

図 8-77 「レスポンス」コンポーネント (「GetDetailResponse」)



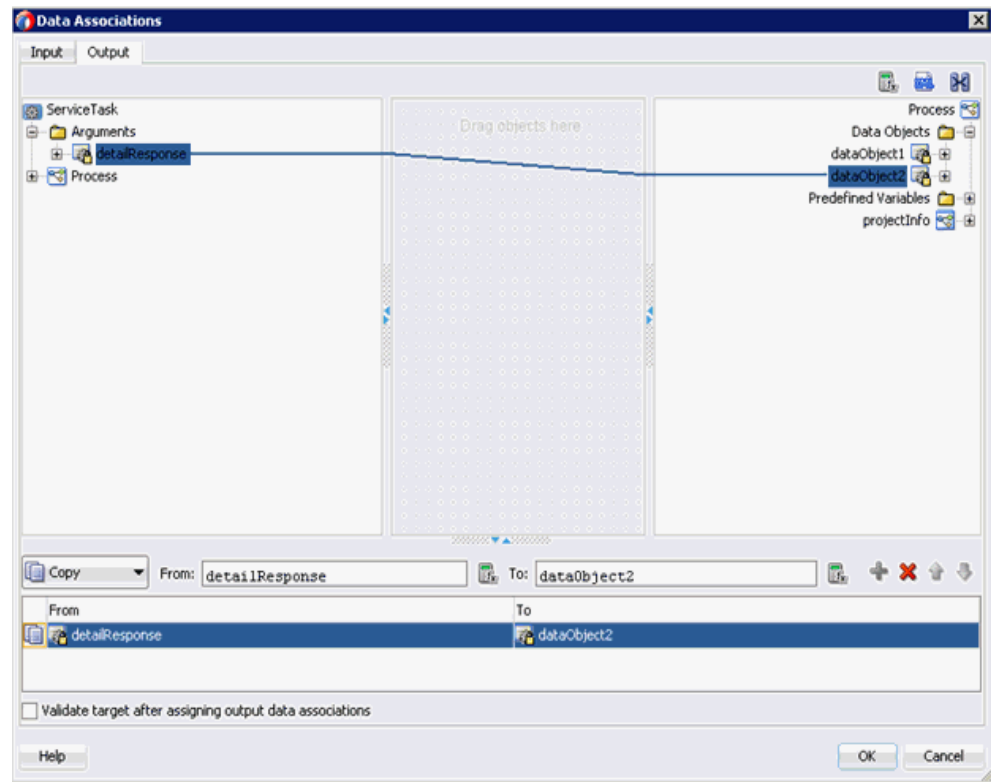
28. データ・オブジェクトの作成ウィンドウで、「OK」をクリックします。
「データ・アソシエーション」ウィンドウが表示されます。
29. 図 8-78 に示すように、「入力」タブの左ペインの「データ・オブジェクト」ノードの下に「dataObject1」を選択し、右ペインの「引数」ノードの下に「getDetail」ノードにドラッグして接続します。

図 8-78 「データ・アソシエーション」



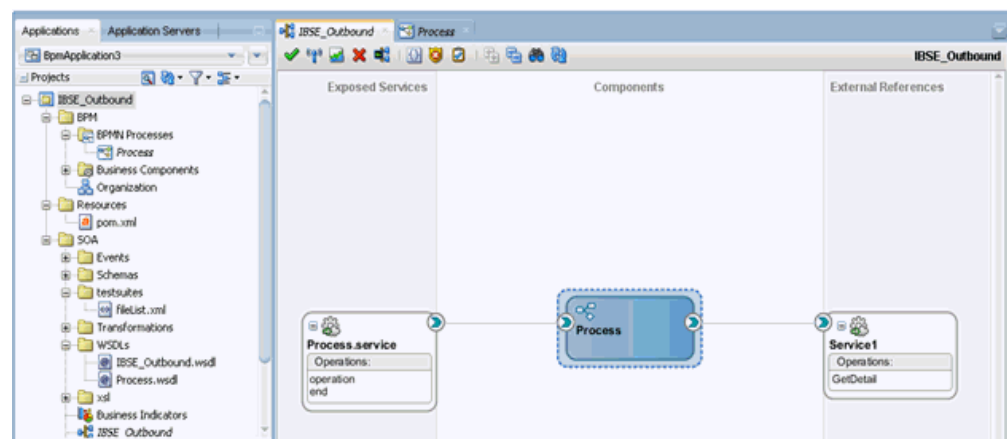
30. 「出力」タブをクリックし、左ペインの「引数」ノードの下の「detailResponse」を選択し、「データ・オブジェクト」ノードの下の「dataObject2」にドラッグして接続します。
31. 図 8-79 に示すように、「OK」をクリックします。

図 8-79 「出力」タブ



32. 表示された「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・バーで「すべて保存」アイコンをクリックし、構成された新規のアウトバウンド 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、左ペインの「IBSE_Outbound」ノードをダブルクリックします。

図 8-80 「Composite.xml」ノード



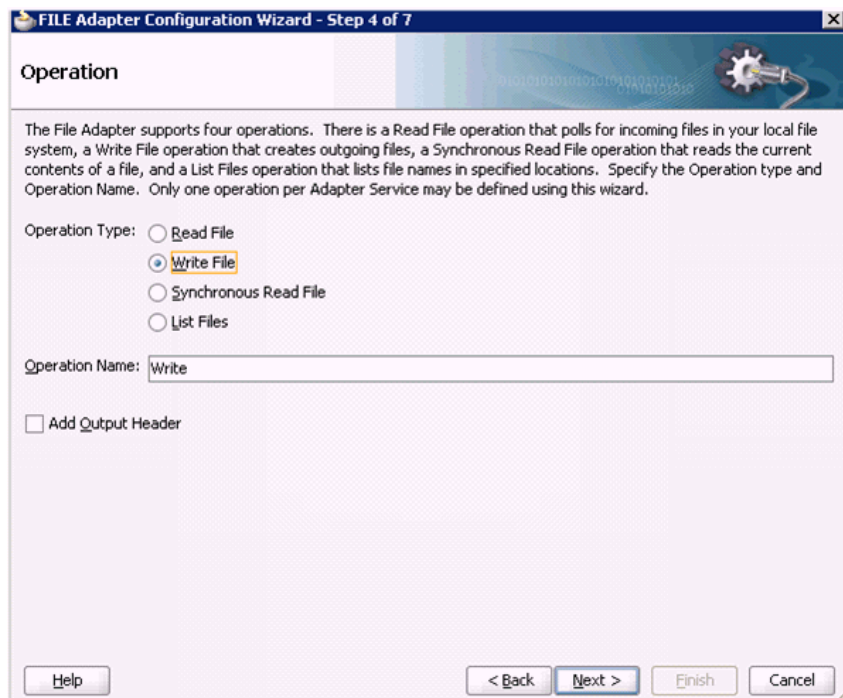
8.7.2.3 書込み操作のファイル・アダプタの作成

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、次のステップを実行します。

1. 「テクノロジ・アダプタ」 ペインから「外部参照」 ペインに「ファイル・アダプタ」 コンポーネントをドラッグ・アンド・ドロップし、「ファイル・アダプタ」 の名前を入力します。
2. 表示された「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、表示された「Operation」 ページで、「操作タイプ」 オプションのリストから「Write File」 を選択し、「次へ」 をクリックします。

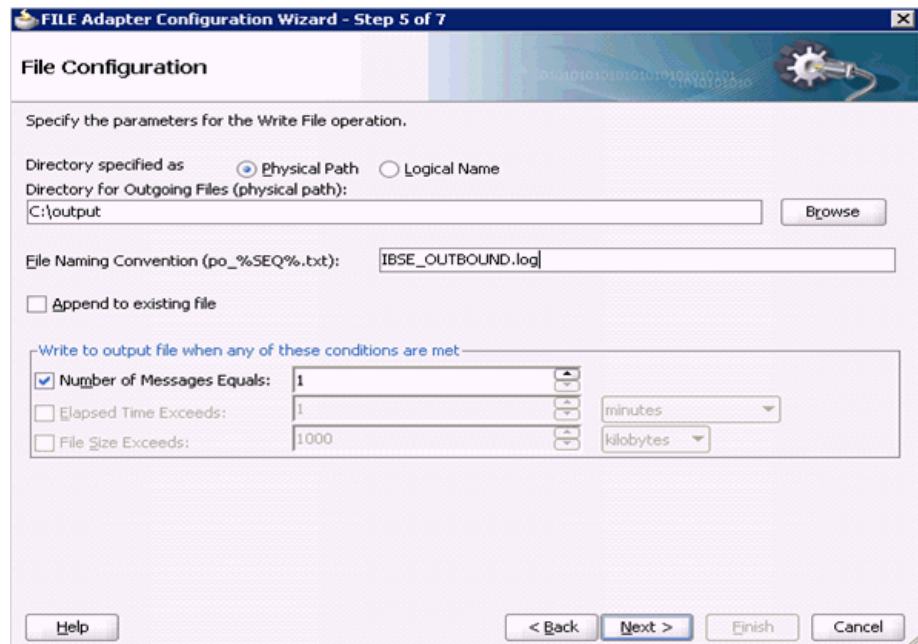
図 8-81 「Operation」 ペイン



「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、出力ファイルの書込み先となるファイル・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。

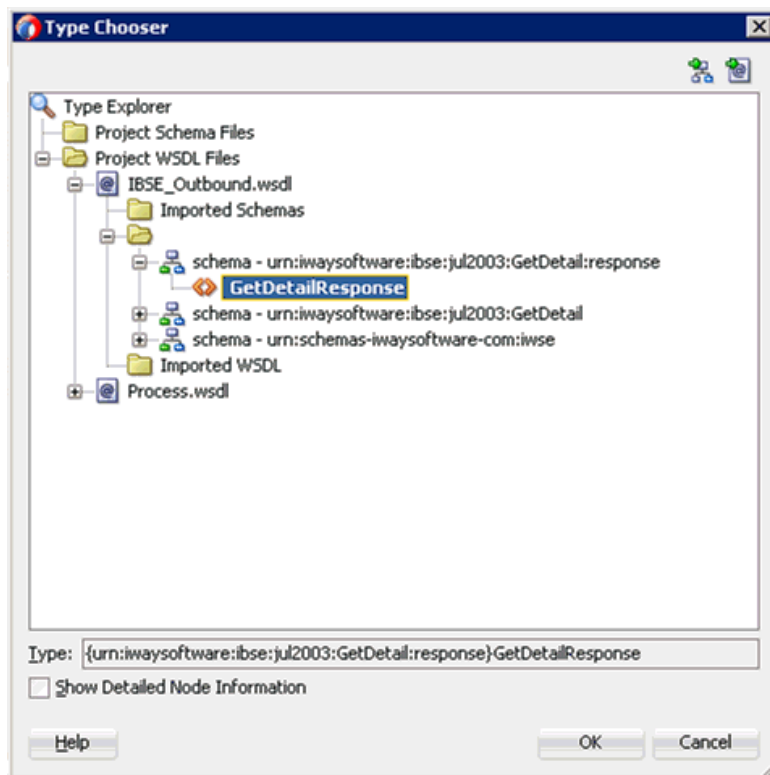
図 8-82 「ファイル構成」 ペイン



「メッセージ」 ペインが表示されます。

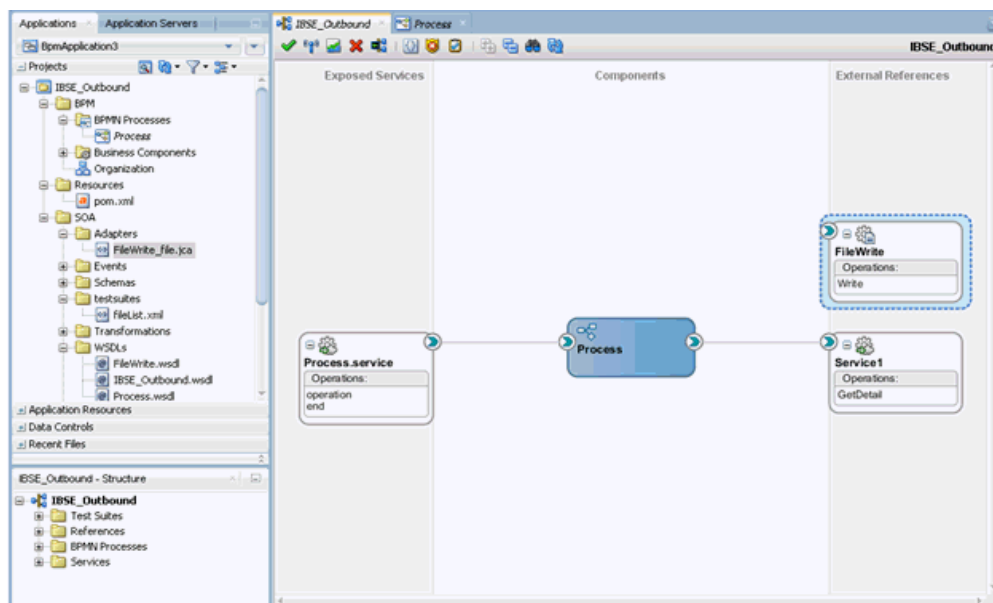
8. 「URL」 フィールドの右の「参照」をクリックします。
9. 表示された「タイプ・チューザ」ウィンドウで、「プロジェクトの WSDL ファイル」、「IBSE_Outbound.wsdl」、「インライン・スキーマ」の順に展開し、「GetDetailresponse」を選択します。
10. 図 8-83 に示すように、「OK」をクリックします。

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. 表示された「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。

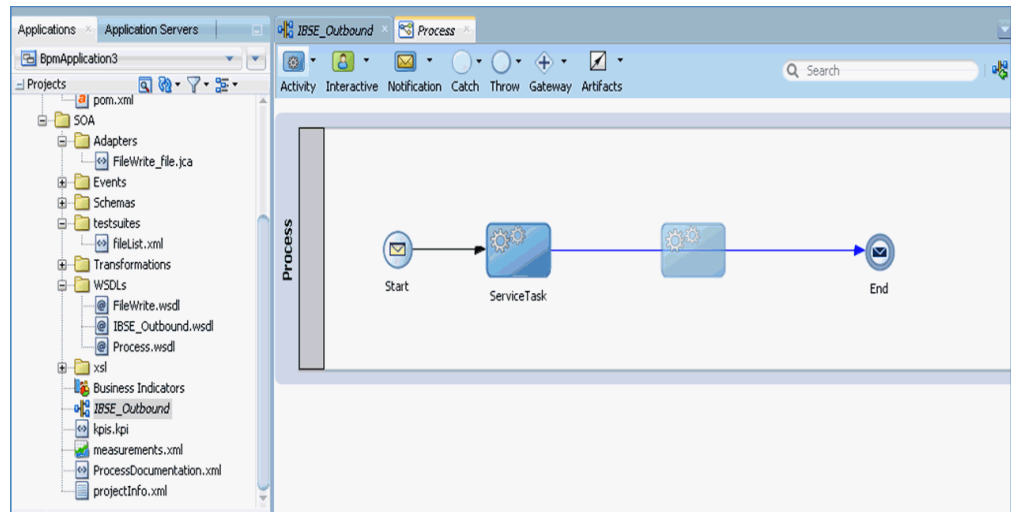
図 8-84 「Composite.xml」 タブ



14. 「アクティビティ」アイコンをクリックします。

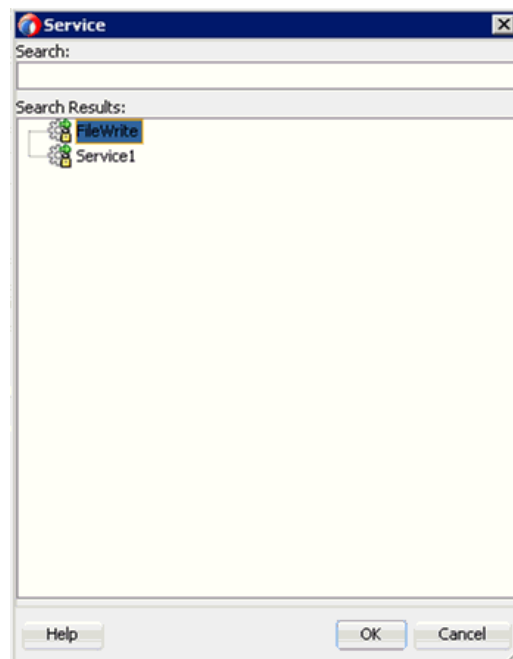
15. 図 8-85 に示すように、「サービス・タスク」と「終了」イベント・コンポーネント間の伝送路に「アクティビティ」アイコンをドロップします。

図 8-85 「アクティビティ」アイコン



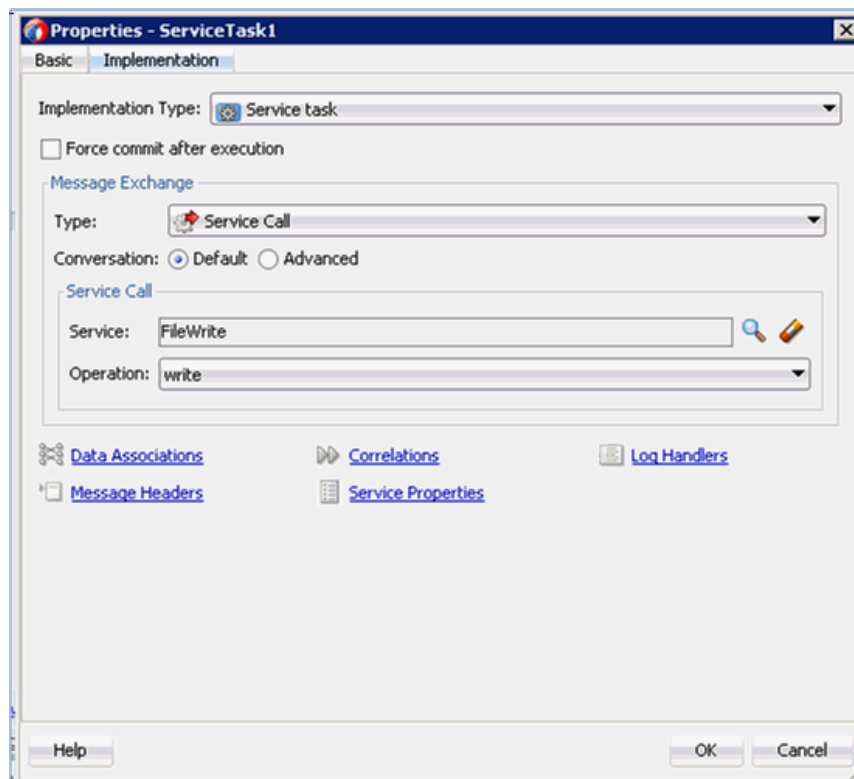
16. 表示された「プロパティ - ServiceTask1」ウィンドウで、「実装」タブをクリックします。
17. 「メッセージ交換」セクションの「タイプ」ドロップダウン・リストから「サービス・コール」を選択します。
18. 「サービス」フィールドの右の「参照」アイコンをクリックします。
19. 図 8-86 に示すように、作成された書込み操作用のサービスを選択し、「OK」をクリックします。

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」



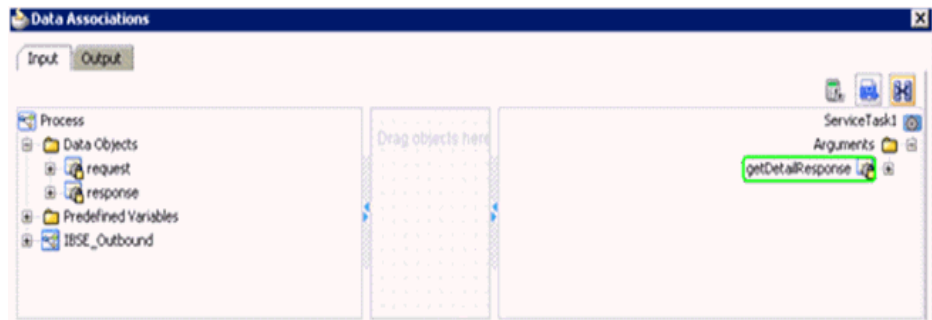
21. 図 8-88 に示すように、「入力」タブで、右上隅の「XSL 変換」アイコンをクリックします。

図 8-88 「XSL 変換」アイコン



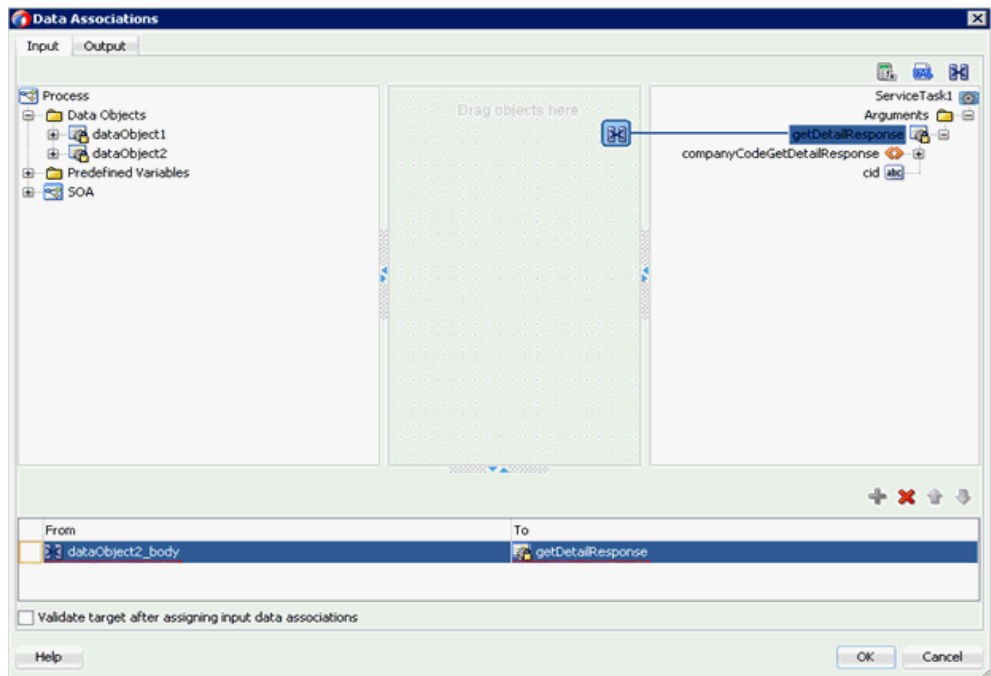
22. 図 8-89 に示すように、「XSL 変換」アイコンを「getDetailResponse」ノードにドラッグ・アンド・ドロップします。

図 8-89 「getDetailResponse」ノード



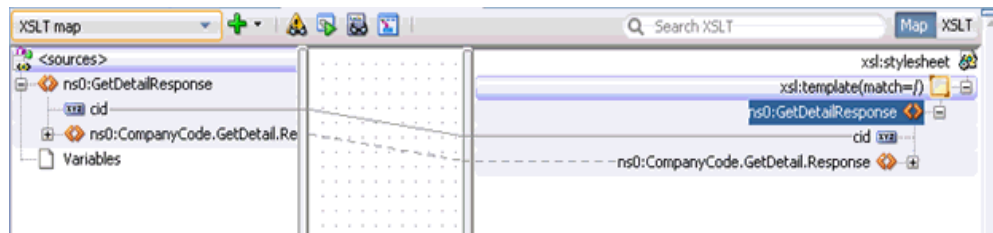
23. 表示されたトランスフォーメーションの作成ウィンドウで、「ソース」セクションの「dataObject2」を選択して、右矢印記号をクリックします。
24. 「OK」をクリックすることで、「ターゲット」ドロップダウン・リストで選択されたデフォルト値および「作成」フィールドのデフォルト名を受け入れます。
25. 図 8-90 に示すように、「データ・アソシエーション」ウィンドウで「OK」をクリックします。

図 8-90 「データ・アソシエーション」ウィンドウ



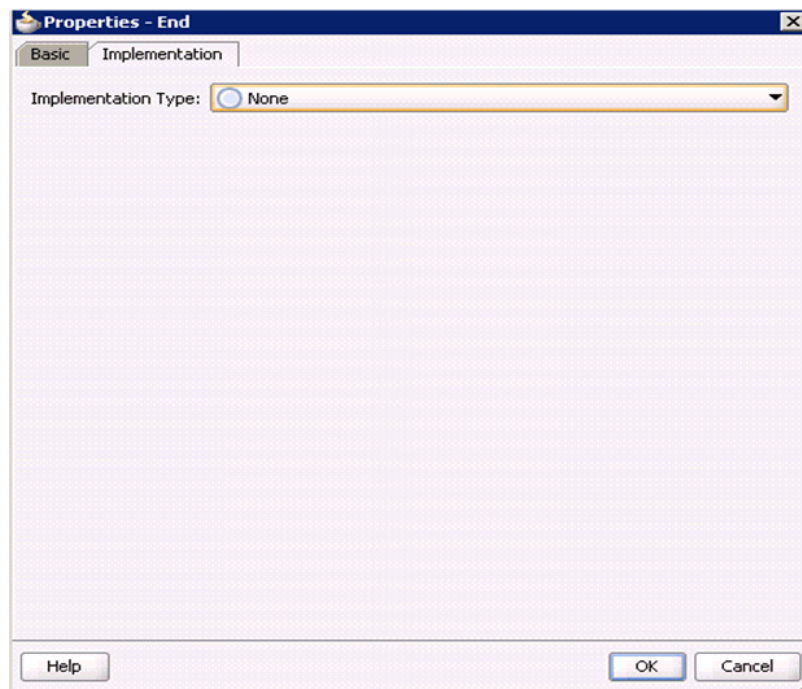
26. 「プロパティ - ServiceTask1」ウィンドウで、「OK」をクリックします。
27. 「response_body.xml」タブで、「ns0:GetDetailResponse」ソース要素を「ns0:GetDetailResponse」ターゲット要素にマッピングします。
28. 図 8-91 に示すように、表示された「自動マップ・プリファレンス」ウィンドウで、デフォルト値を維持し、「OK」をクリックします。

図 8-91 「自動マップ・プリファレンス」



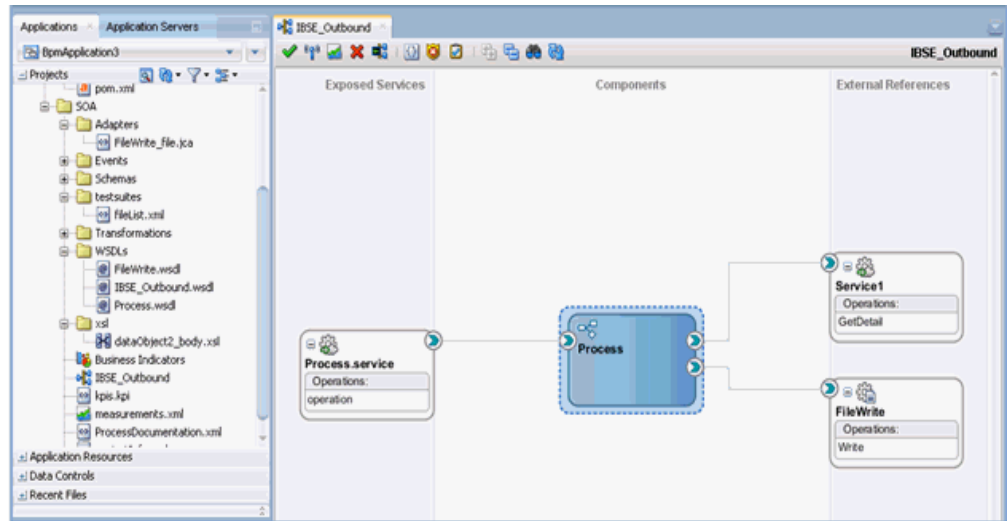
29. 「プロセス」ワークスペース領域に戻り、「終了」イベント・コンポーネントをダブルクリックします。
30. 「プロパティ - 終了」ウィンドウで、「実装」タブをクリックします。
31. 「実装タイプ」ドロップダウン・リストから「なし」を選択します。
32. 図 8-92 に示すように、「OK」をクリックします。

図 8-92 「実装」タブ



33. 図 8-93 に示すように、メニュー・バーで「すべて保存」アイコンをクリックし、構成された新規のアウトバウンド 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」](#)と同じ手順に従うことができます。

デプロイされたら、[8.4.5 項「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動」](#)に定義されたように入力 XML を起動できます。

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処理の構成

注意： リリース 12c (12.2.1.0.0) では、sbconsole を使用した Oracle Service Bus 用のアウトバウンドおよびインバウンド・プロセスの構成方法が変更されています。

sbconsole を使用して Oracle Service Bus 用のプロセスを作成する場合は、『Oracle Fusion Middleware Application Adapters リリース・ノート for 12c (12.2.1.0.0)』の第2章「sbconsole を使用した Oracle Service Bus 用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。

Oracle Application Adapter for SAP R/3 は Oracle Service Bus (OSB) にシームレスに統合されており、Web サービスの統合を可能にします。OSB は、サービス指向アーキテクチャ (SOA) に基づいています。Web Service Definition Language (WSDL) ドキュメントとして公開されるアダプタ・サービスを使用します。

この章の内容は以下のとおりです。

- 9.1 項「Application Adapter と Oracle Service Bus の統合の概要」
- 9.2 項「sbconsole を使用したアウトバウンド処理の構成 (J2CA 構成)」
- 9.3 項「sbconsole を使用したインバウンド処理の構成 (J2CA 構成)」
- 9.4 項「sbconsole を使用したアウトバウンド処理の構成 (BSE 構成)」
- 9.5 項「Oracle Service Bus を使用した同期 SAP R/3 イベントの作業 (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 Application Adapter と Oracle Service Bus の統合の概要

Oracle Service Bus (OSB) と統合するには、Oracle Application Adapter for SAP R/3 を OSB と同じ Oracle WebLogic Server にデプロイする必要があります。また、基になるアダプタ・サービスは、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) の両方のサービス用に Oracle Adapter アプリケーション・エクスペローラ (アプリケーション・エクスペローラ) でデザインタイム中に生成される WSDL ファイルとして公開される必要があります。

9.2 sbconsole を使用したアウトバウンド処理の構成 (J2CA 構成)

この項では、J2CA 構成に sbconsole を使用して、アウトバウンド処理を構成する方法について説明します。

Application Adapters インストールの次のフォルダに、このアウトバウンドの使用ケース・シナリオのサンプル・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、次のトピック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「Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・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 の公開」
- 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・サービスを使用した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、プロジェクト・フォルダを作成するには、次のステップを実行します

1. 構成済の 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 を入力して、Oracle Service Bus コンソールを起動します。

```
http://hostname:port/sbconsole
```

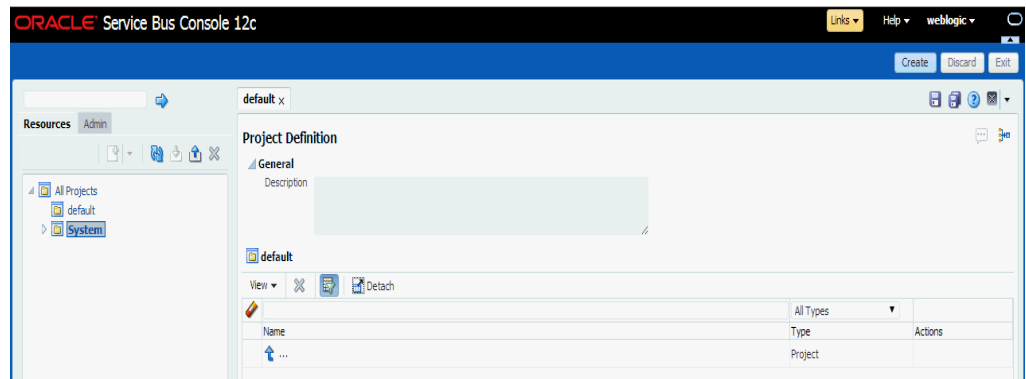
ここで、*hostname* は Oracle WebLogic Server が稼働しているマシンの名前、*port* は使用しているドメインのポートです。

Oracle Service Bus コンソールのログオン・ページが表示されます。

3. 有効なユーザー名および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。

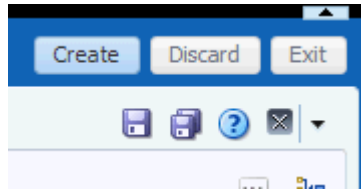
図 9-1 に示すように、Oracle Service Bus コンソールのホーム・ページが表示されます。

図 9-1 Oracle Service Bus コンソールのホーム・ページ



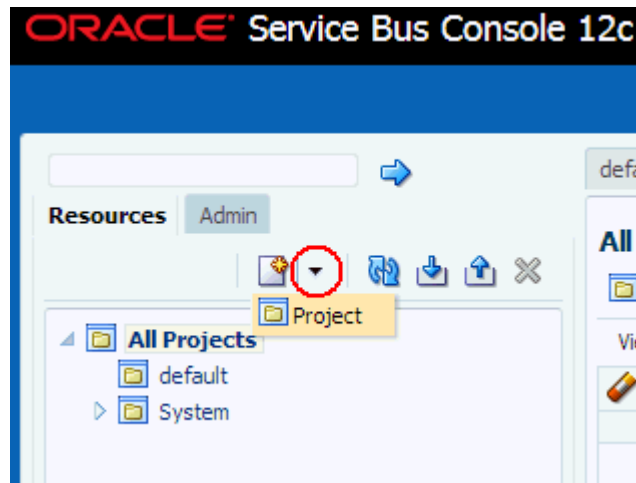
4. 図 9-2 に示すように、Oracle Service Bus セッションの右ペインで「作成」をクリックします。

図 9-2 Oracle Service Bus セッション



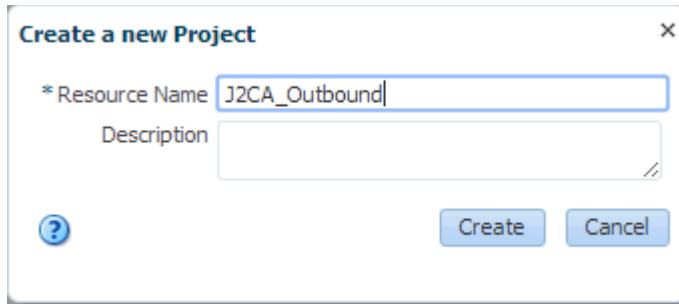
5. 図 9-3 に示すように、「すべてのプロジェクト」を選択し、左ペインで下矢印をクリックして、「プロジェクト」を選択します。

図 9-3 「すべてのプロジェクト」フォルダ



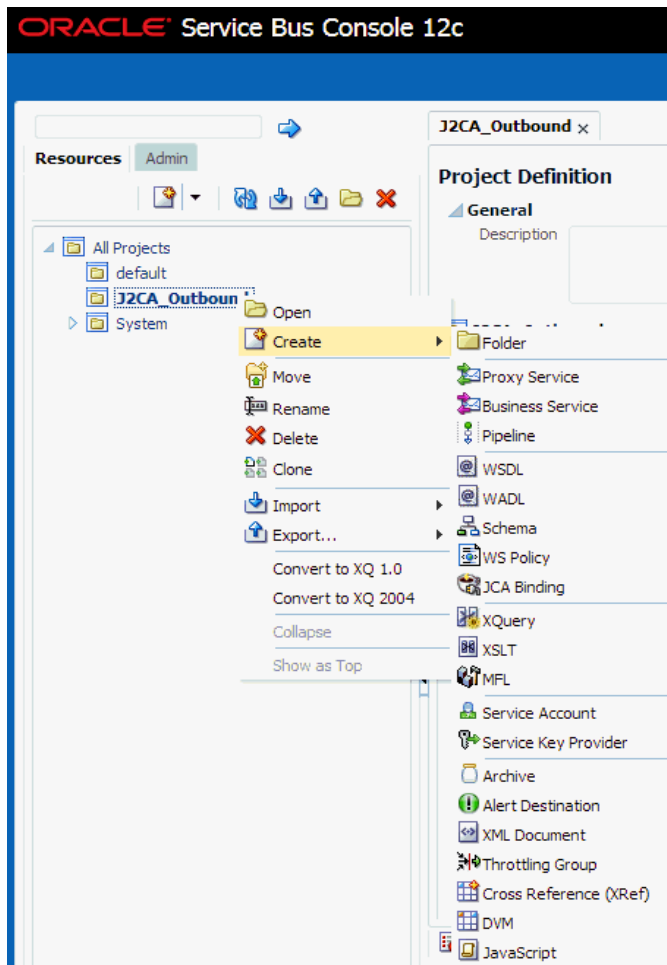
- 図 9-4 に示すように、「新規プロジェクトの作成」ウィンドウが表示されます。

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」フィールドで新規プロジェクトに有効な名前 (J2CA_Outbound など) を入力し、「作成」をクリックします。
新規プロジェクトが正常に作成され、リストに表示されます。
7. 図 9-5 に示すように、新規作成したプロジェクトを右クリックし、「作成」を選択し、「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. ステップ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。

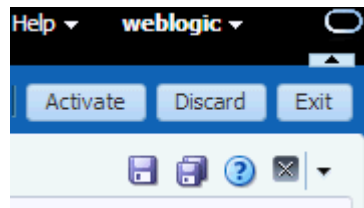
図 9-6 に示すように、「Business Service」、「Proxy Service」、「Wsdls」の各フォルダが左ペインのプロジェクト・ノードの下に表示されます。

図 9-6 プロジェクト・ノード



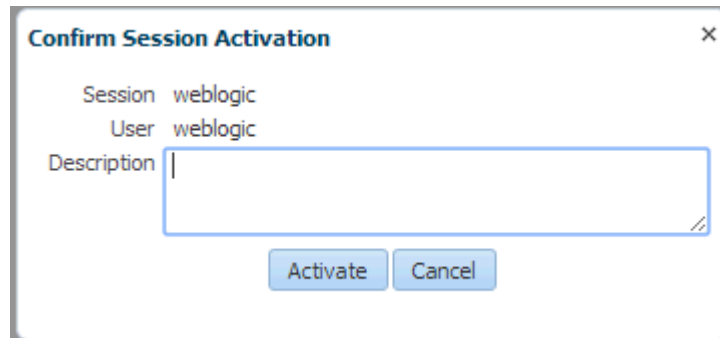
10. 図 9-7 に示すように、Oracle Service Bus セッションの右ペインで「アクティブ化」をクリックします。

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・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内容を保存します。

図 9-8 セッション・アクティブ化の確認ウィンドウ



9.2.2 Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・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 (プロジェクトまたはフォルダ) に直接公開する前に、OSB ユーザーは次の手順を行う必要があります。

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移動します。
`<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。
 このコマンドは、OSB Console を WSDL を公開するためのアプリケーション・エクスプローラから Oracle WebLogic Server API へのアクセス用クラス・パスを設定します。
4. コマンド・プロンプト・ウィンドウを開けたままにします。
5. 次のディレクトリに移動します。
`<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 Console に WSDL を公開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 の公開

アプリケーション・エクスプローラから Oracle Service Bus に WSDL を公開するには、次のステップを実行します

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、SAP R/3 ターゲットに接続します。
 詳細については、[第 4 章「Oracle Application Adapter for SAP R/3 の構成」](#) 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** の順に展開します。
3. [図 9-9](#) に示すように、「GetDetail」メソッドを右クリックし、メニューから「アウトバウンド JCA サービスの作成 (リクエスト/レスポンス)」を選択します。

図 9-9 「アウトバウンド JCA サービスの作成 (リクエスト/レスポンス)」オプション

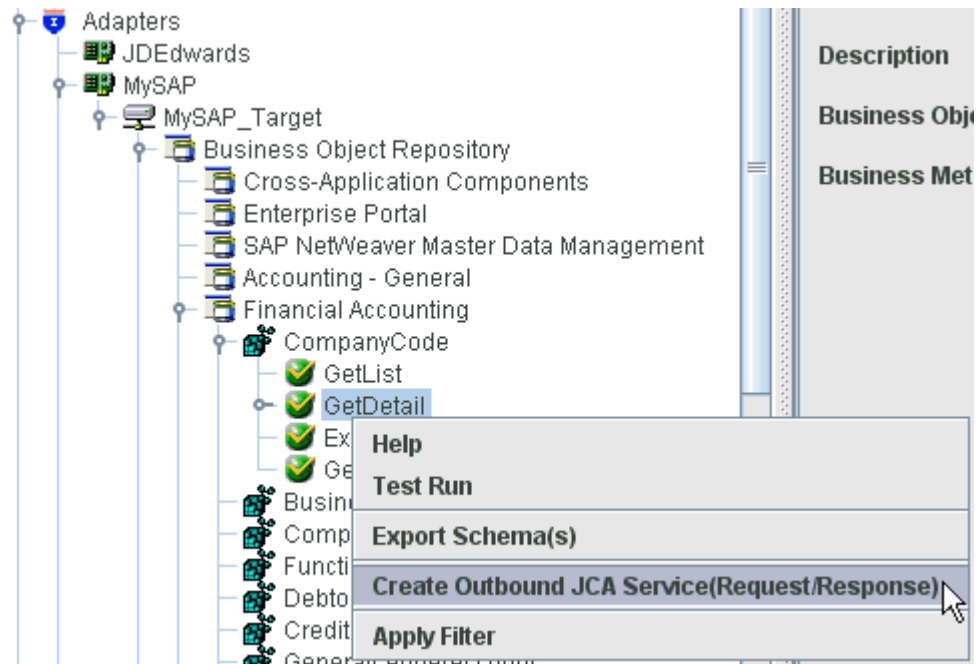
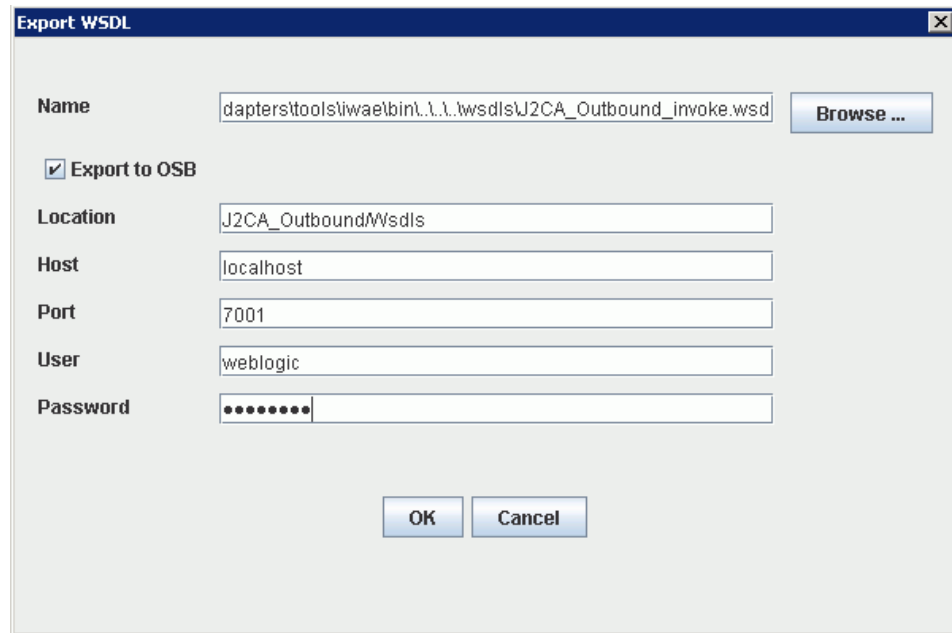


図 9-10 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

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 WSDL ファイルのデフォルトのファイル名が指定されます。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 ドキュメントの公開先となる Oracle Service Bus 内のフォルダ名を入力します。

ロケーションは、Oracle Service Bus プロジェクト名と、さらにオプションで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フォルダ名はいずれもスラッシュ文字の「/」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 により使用されるポートを入力します。
9. 「ユーザー」フィールドで、Oracle Service Bus へのアクセス用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 へのアクセス用のパスワードを入力します。
11. 「OK」をクリックします。

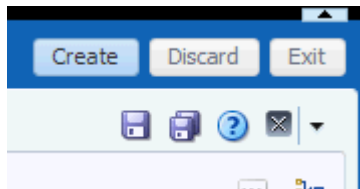
WSDL が「WSDL のエクスポート」ダイアログで指定されている場所に公開され、Oracle Service Bus でビジネス・サービスまたはプロキシ・サービスとともに使用できるようになります。

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、次のステップを実行します。

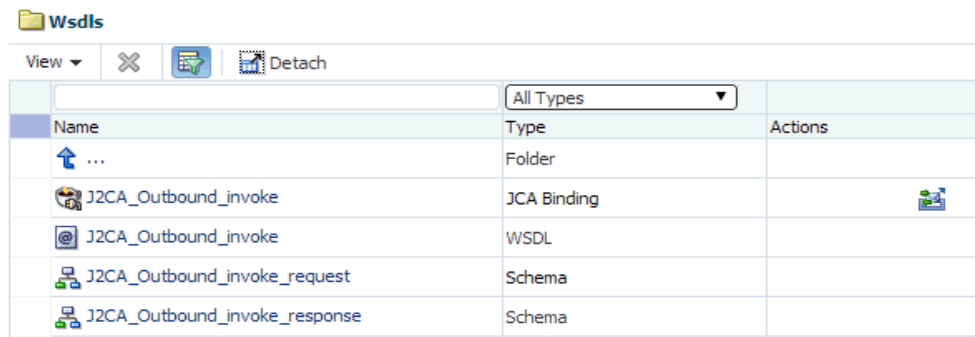
1. Oracle Service Bus コンソールを開きます。図 9-11 に示すように、Oracle Service Bus セッションの右ペインで「作成」をクリックします。

図 9-11 「作成」ボタン



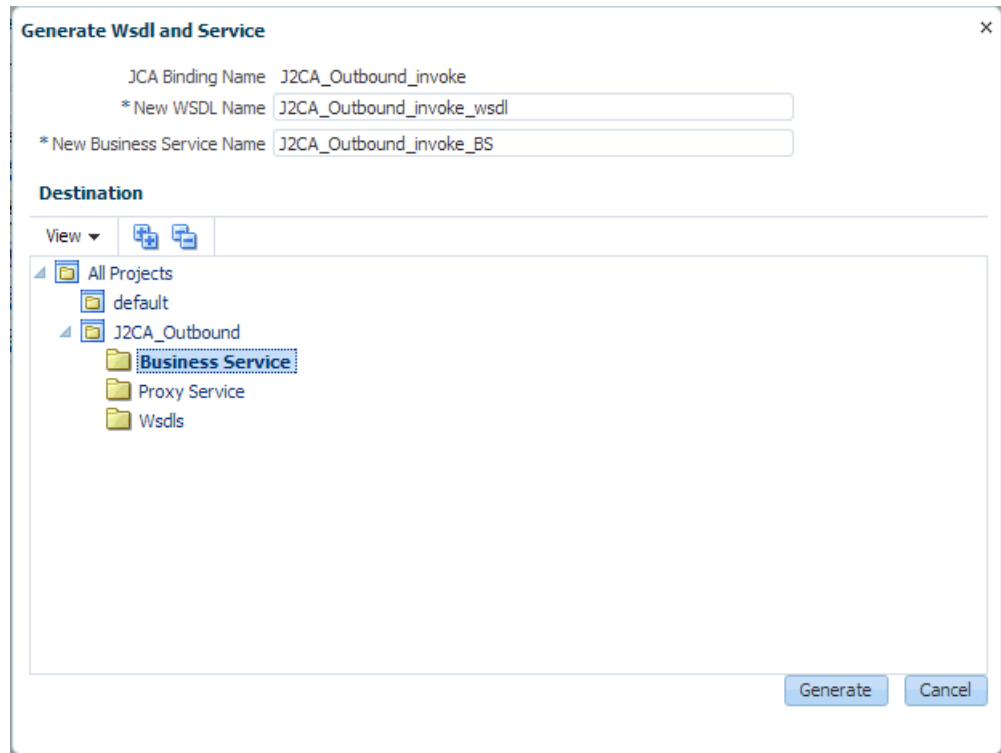
2. 図 9-12 に示すように、左ペインで作成済の WSDL フォルダ (Wsdls など) をダブルクリックし、エクスポートされた WSDL が右ペインにリストされていることを確認します。

図 9-12 「Wsdls」フォルダ



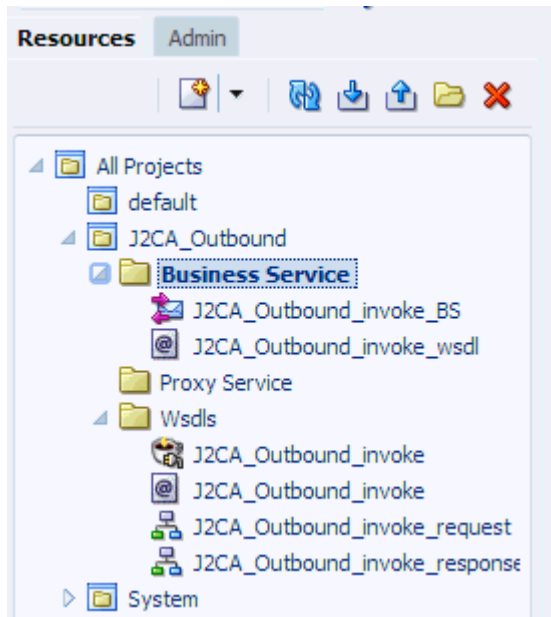
3. 「アクション」列で、「JCA バインド」に対応するアイコンをクリックします。
図 9-13 に示すように、「WSDL とサービスの生成」ウィンドウが表示されます。

図 9-13 「WSDL とサービスの生成」 ウィンドウ



4. 新規 WSDL の名前および新規ビジネス・サービス名を対応するフィールドに入力します。
5. 「宛先」領域で、ビジネス・サービスに指定されている使用可能なプロジェクトおよびサブフォルダを選択します。
6. 「生成」をクリックします。
7. プロジェクト・エクスプローラで、「**Business Service**」を展開します。図 9-14 に示すように、生成された WSDL とビジネス・サービスがリストされていることを確認してください。

図 9-14 「Business Service」フォルダ

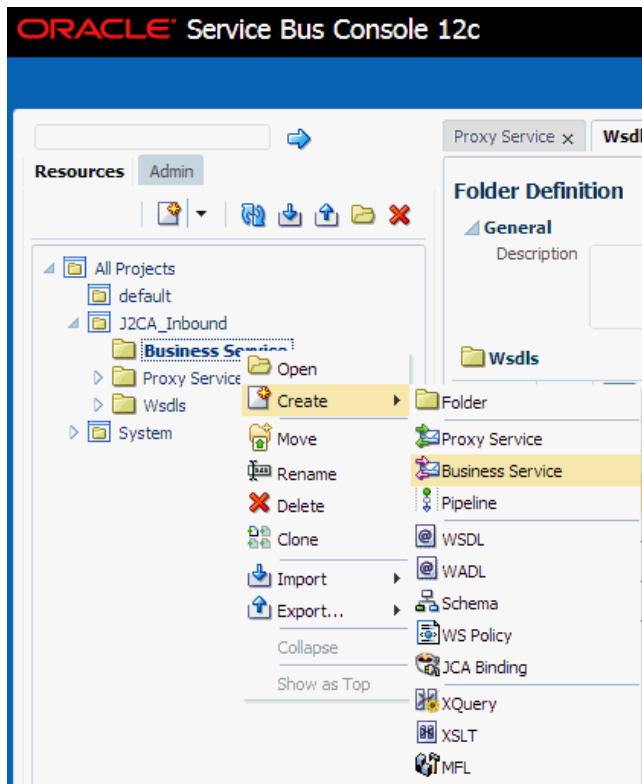


9.2.5 ファイル・タイプ・ビジネス・サービスの構成

ファイル・タイプのビジネス・サービスを構成するには、次のステップを実行します。

1. 図 9-15 に示すように、左ペインで作成済の「Business Service」フォルダを右クリックし、「作成」を選択し、「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」フィールドにビジネス・サービスの名前を指定し、サービス定義の「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、「次へ」をクリックします。

図 9-16 サービス定義

The screenshot shows the 'Create Business Service' dialog box. At the top, there are three tabs: 'Create', 'Type', and 'Transport'. The 'Create' tab is selected. Below the tabs, the 'Create Service' section has a '*Resource Name' field with the value 'File_Out' and a 'Description' field. The 'Service Definition' section has two radio buttons: 'WSDL Based Service' (unselected) and 'Transport' (selected). Under 'WSDL Based Service', there are fields for 'Name', 'Path', and 'Port/Binding'. Under 'Transport', there is a dropdown menu currently showing 'file'. At the bottom, there are four buttons: 'Back', 'Next', 'Create', and 'Cancel'.

3. 「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」が「XML」に設定され、「レスポンス・タイプ」が「なし」に設定されています。図 9-17 に示すように、「次へ」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 the 'Create Business Service' dialog box with the 'Type' tab selected. The progress bar at the top indicates the current step. Under 'Service Type', 'Messaging Service' is selected. The 'Request Type' is set to 'XML', 'Schema Name' is empty, 'Element/Type' is empty, and 'Response Type' is set to 'None'. Navigation buttons 'Back', 'Next', 'Create', and 'Cancel' are at the bottom.

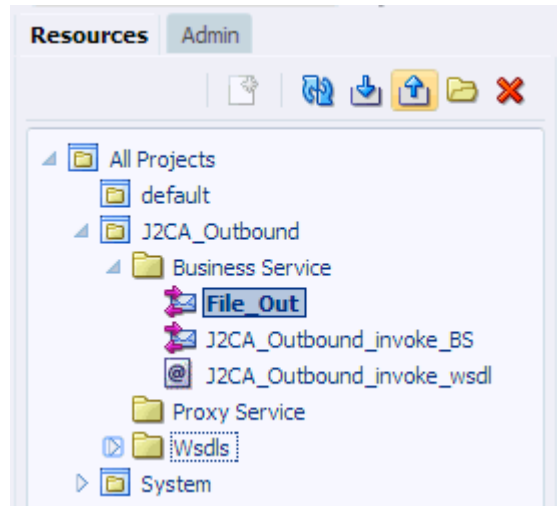
4. 「エンドポイント URI」フィールドで、ファイル・システム上の宛先フォルダへのパスを入力します。
5. 図 9-18 に示すように、「作成」をクリックします。

図 9-18 「トランスポート」ページ

The screenshot shows the 'Create Business Service' dialog box with the 'Transport' tab selected. The progress bar at the top indicates the current step. Under 'Transport', 'Protocol' is set to 'file' and 'Load Balancing Algorithm' is set to 'Round Robin'. Under 'Endpoint URIs', there is a list with one entry: 'file:///c:/output'. Navigation buttons 'Back', 'Next', 'Create', and 'Cancel' are at the bottom.

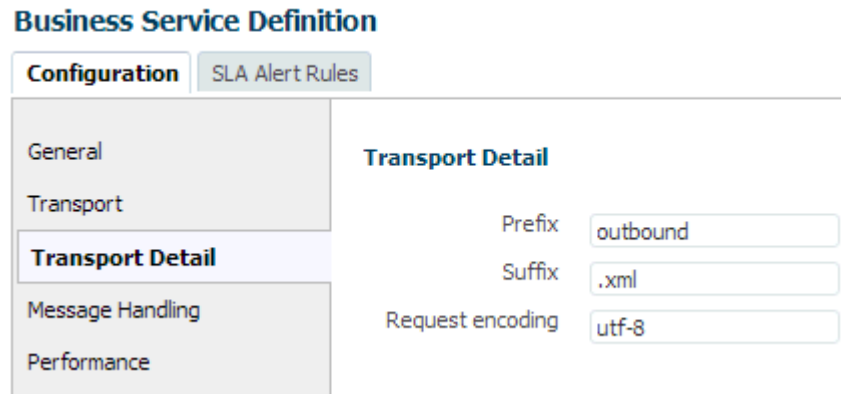
図 9-19 に示すように、ビジネス・サービス「File_Out」が作成され、「Business Service」の下にリストされます。

図 9-19 「File_Out」ビジネス・サービス



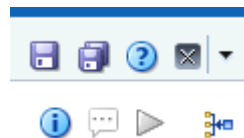
6. 図 9-20 に示すように、「File_Out」をダブルクリックし、左ペインで「トランスポートの詳細」をクリックし、受信する出力ファイルの接頭辞および接尾辞を入力します。

図 9-20 「トランスポートの詳細」



7. 図 9-21 に示すように、右隅にある「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」または「すべて保存」アイコン

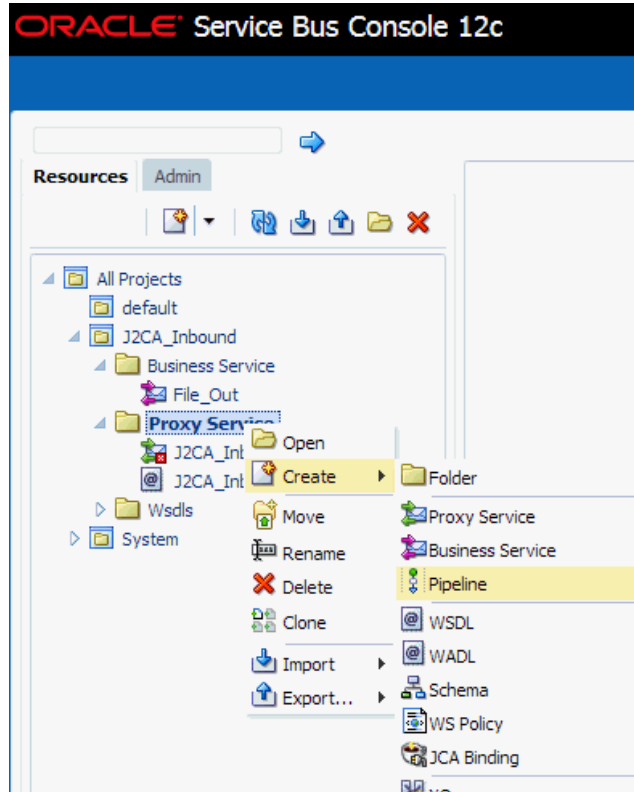


9.2.6 プロキシ・サービスを使用したパイプラインの構成

パイプラインを構成するには、次のステップを実行します。

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、「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



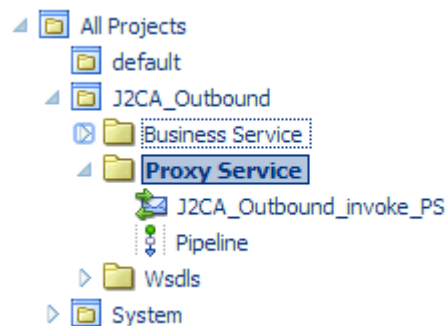
パイプラインの作成ウィンドウが表示されます。

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・サービス名を変更したい場合は、図 9-23 に示すように、変更して「**トランスポート**」を「**ファイル**」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

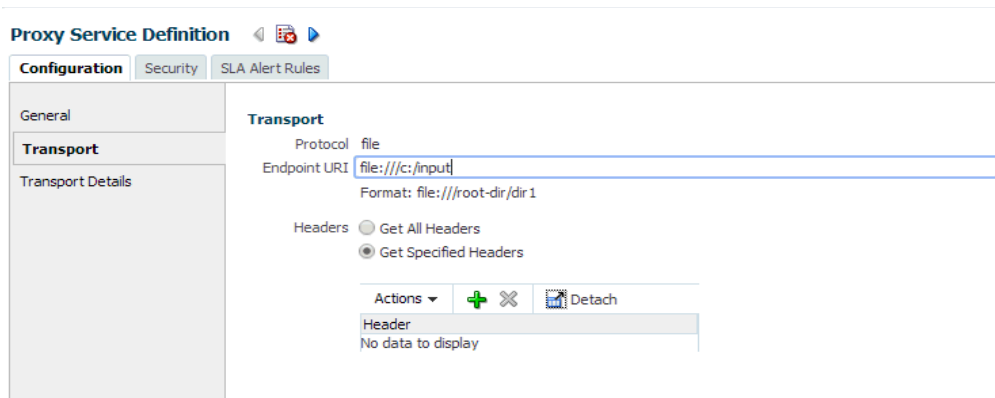
図 9-24 に示すように、作成されたパイプラインとプロキシ・サービスが「Proxy Service」の下にリストされます。

図 9-24 パイプライン・ノード



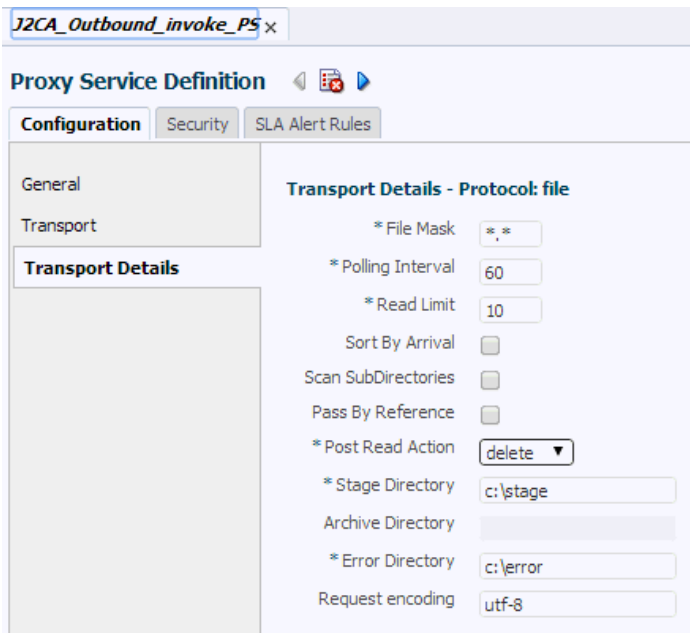
- 作成済のプロキシ・サービスをダブルクリックし、左ペインで「トランスポート」をクリックします。図 9-25 に示すように、「エンドポイント URI」フィールドに入力ロケーションを指定します。

図 9-25 「トランスポート」



4. 図 9-26 に示すように、左ペインで「トランスポートの詳細」をクリックし、「ステージ・ディレクトリ」フィールドと「エラー・ディレクトリ」フィールドにロケーションを指定します。

図 9-26 「トランスポートの詳細」



5. 図 9-27 に示すように、右隅にある「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



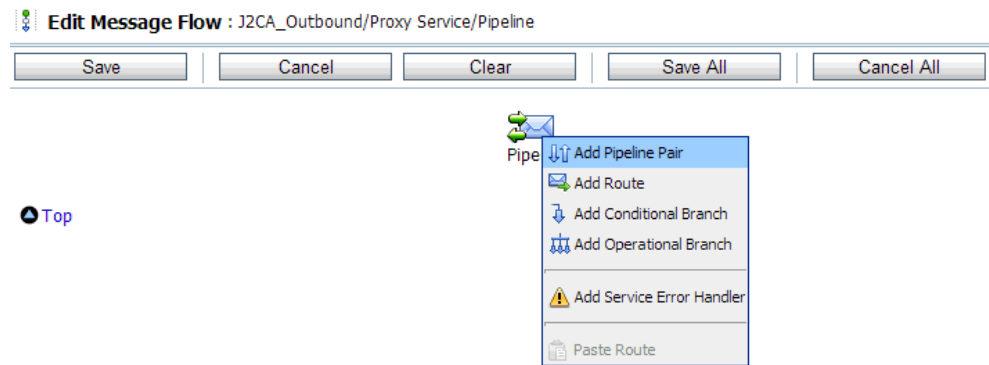
6. 図 9-28 に示すように、「パイプライン」ノードをダブルクリックし、右ペインでメッセージ・フローを開くアイコンをクリックしてメッセージ・フローを開きます。

図 9-28 メッセージ・フローを開くアイコン



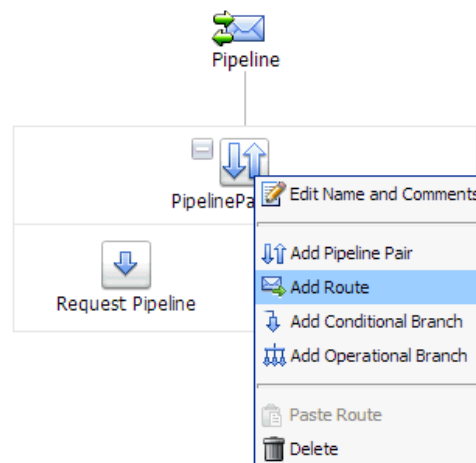
7. 図 9-29 に示すように、「プロキシ・サービス」アイコンをクリックし、メニューから「パイプライン・ペアの追加」を選択します。

図 9-29 「パイプライン・ペアの追加」オプション



8. 図 9-30 に示すように、「PipelinePairNode1」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。

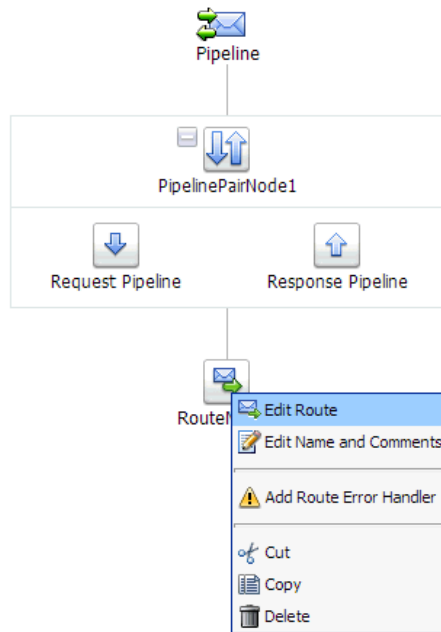
図 9-30 「ルートの追加」オプション



「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、メニューから「ルートの編集」を選択します。

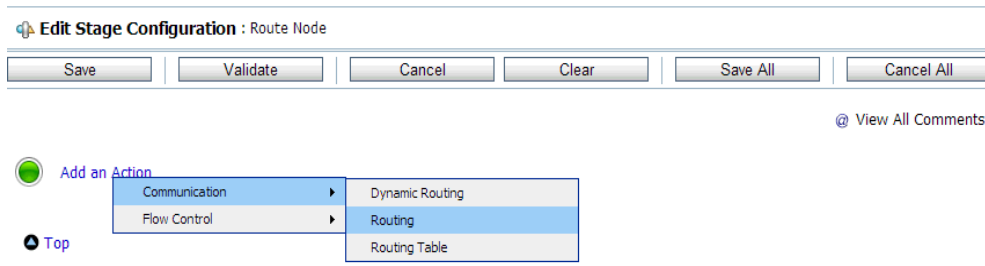
図 9-31 「ルート編集」オプション



「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、「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、「<サービス>」をクリックします。

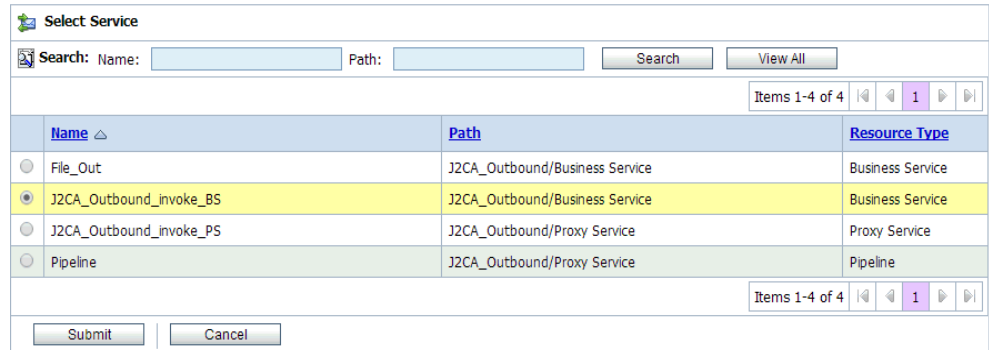
図 9-33 アクション



「サービスの選択」ダイアログが表示されます。

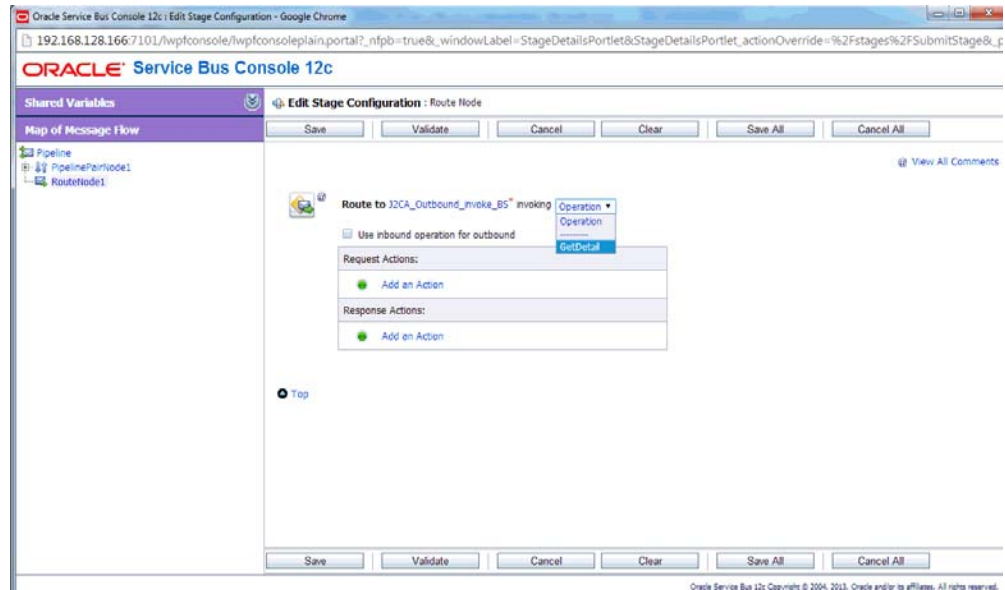
12. 図 9-34 に示すように、SAP R/3 用に構成された WSDL タイプのビジネス・サービスを選択し、「発行」をクリックします。

図 9-34 「サービスの選択」ダイアログ



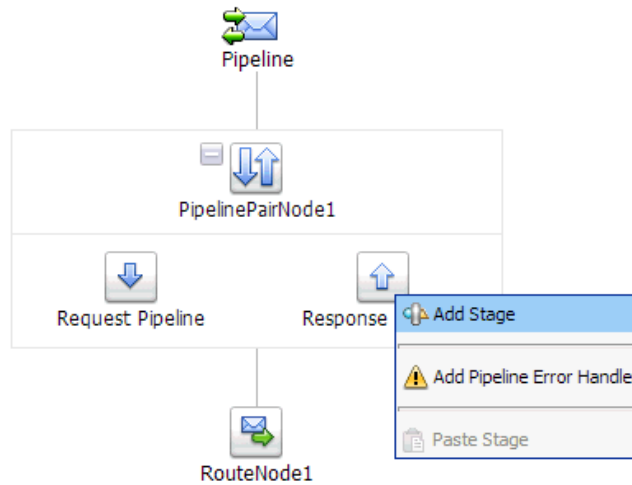
13. 図 9-35 に示すように、リストから操作属性として SAP R/3 ビジネス・オブジェクトの名前 (「GetDetail」など) を選択し、「保存」をクリックします。

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、メニューから「ステージの追加」を選択します。

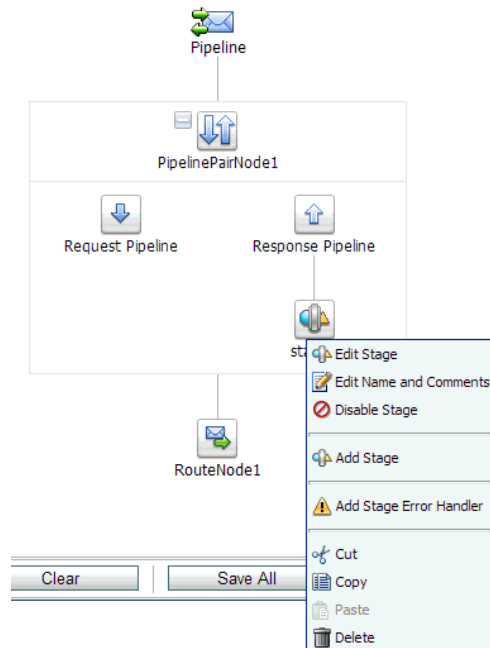
図 9-36 「レスポンス・パイプライン」アイコン



ステージ 1 アイコンが「リクエスト・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、「Stage1」アイコンをクリックし、メニューから「ステージの編集」を選択します。

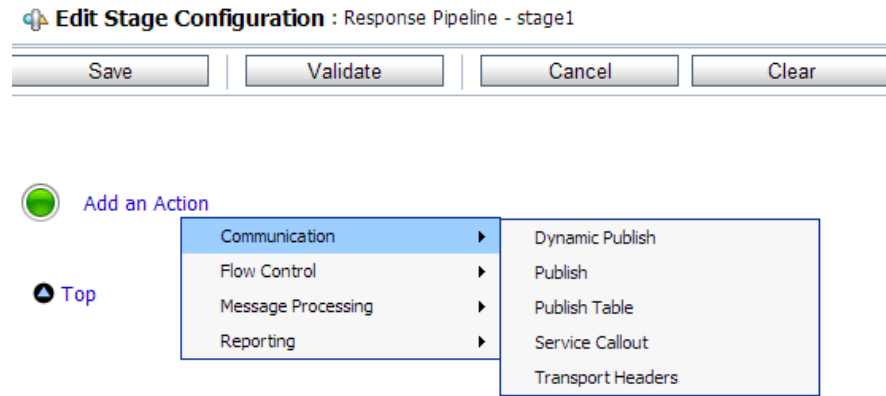
図 9-37 「ステージの編集」オプション



「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。

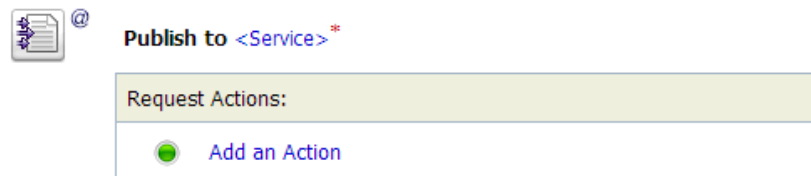
16. 図 9-38 に示すように、「アクションの追加」をクリックし、「通信」を選択し、「パブリッシュ」をクリックします。

図 9-38 「ステージ構成の編集」ワークスペース領域



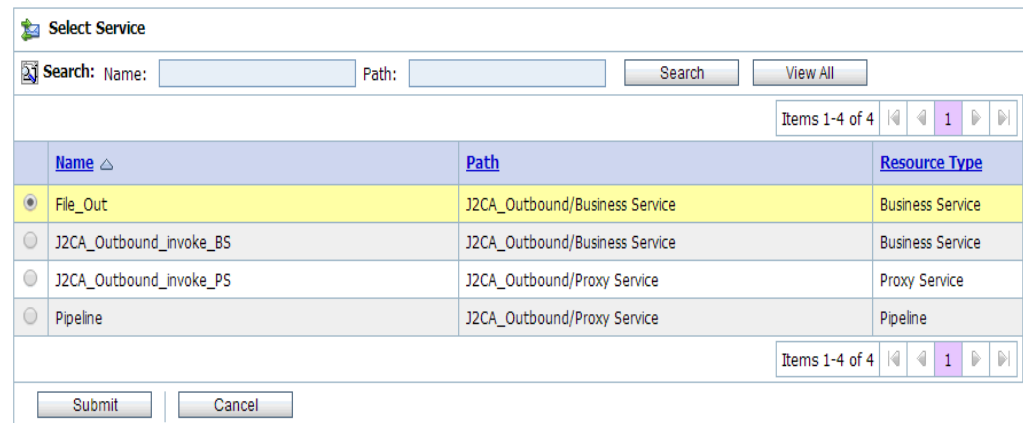
17. 図 9-39 に示すように、「<サービス>」をクリックします。

図 9-39 <サービス> アクション



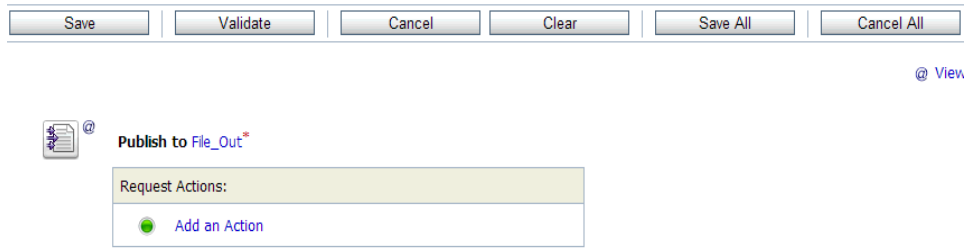
18. 図 9-40 に示すように、「サービスの選択」ダイアログで、ファイル・タイプのビジネス・サービスを選択し、「発行」をクリックします。

図 9-40 「サービスの選択」ダイアログ



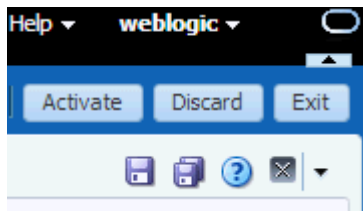
19. 図 9-41 に示すように、「すべて保存」をクリックします。

図 9-41 「すべて保存」ボタン



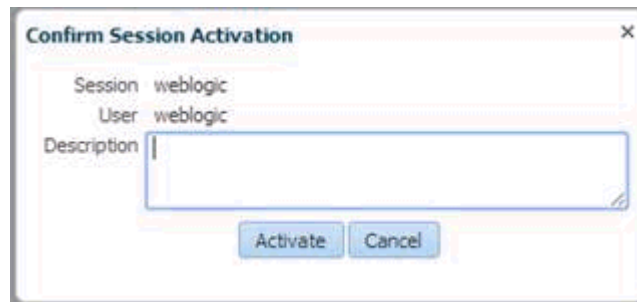
20. 図 9-42 に示すように、Oracle Service Bus セッションの右ペインで「アクティブ化」をクリックします。

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内容を保存します。

図 9-43 セッション・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 ファイルをコピーおよび貼付けます。出力が構成済の出力ロケーション (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処理の構成 (J2CA 構成)

この項では、J2CA 構成に sbconsole を使用して、インバウンド処理を構成する方法について説明します。

Application Adapters インストールの次のフォルダに、このインバウンドの使用ケース・シナリオのサンプル・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
```

この節では、次のトピック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「Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・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 の公開」
- 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 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 Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定

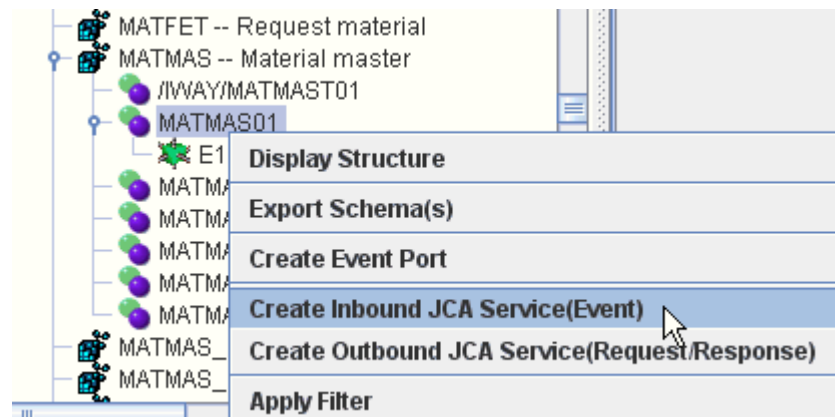
Oracle Service Bus 統合のためにアプリケーション・エクスプローラのクラス・パスを設定する方法の詳細は、9-6 ページ 9.2.2 項「Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定」を参照してください。

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 の公開

アプリケーション・エクスプローラから Oracle Service Bus に WSDL を公開するには、次のステップを実行します

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、SAP R/3 ターゲットに接続します。
詳細については、第 4 章「Oracle Application Adapter for SAP R/3 の構成」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、4-22 ページの 4.10 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、ターゲットで「MATMAS -- Material master」ノードを展開します。

図 9-44 インバウンド JCA サービスの作成 (イベント) オプション



4. 図 9-44 に示すように、「MATMAS01」を右クリックし、メニューからインバウンド JCA サービスの作成 (イベント) 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 WSDL ファイルのデフォルトのファイル名が指定されます。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・サービス用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」の検証の 3 つのチェック・ボックスも使用できます。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 ドキュメントの公開先となる Oracle Service Bus 内のフォルダ名を入力します。
 ロケーションは、Oracle Service Bus プロジェクト名と、さらにオプションで 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フォルダ名はいずれもスラッシュ文字の「/」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 により使用されるポートを入力します。
12. 「ユーザー」フィールドで、Oracle Service Bus へのアクセス用のユーザー名を入力します。

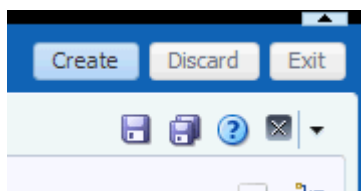
13. 「パスワード」フィールドで、Oracle Service Bus へのアクセス用のパスワードを入力します。
14. 「OK」をクリックします。
インバウンド WSDL が、「WSDL のエクスポート」ダイアログで指定されている場所に公開され、Oracle Service Bus でプロキシ・サービスとともに使用できるようになります。

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、次のステップを実行します。

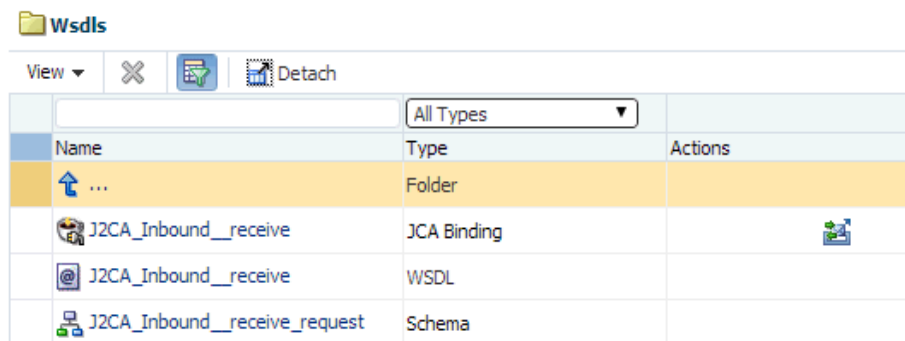
1. Oracle Service Bus コンソールを開きます。図 9-46 に示すように、Oracle Service Bus セッションの右ペインで「作成」をクリックします。

図 9-46 「作成」ボタン



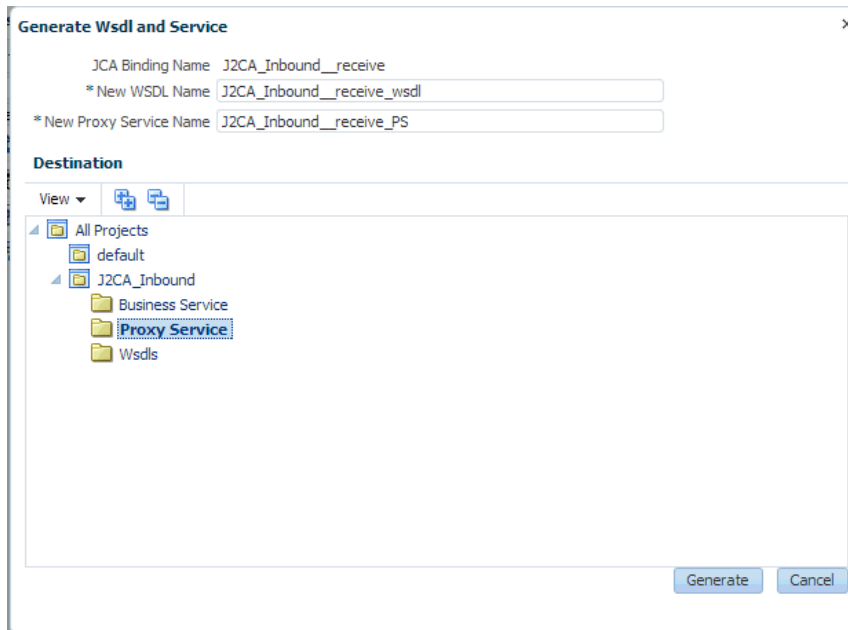
2. 図 9-47 に示すように、左ペインで作成済の WSDL フォルダ (Wsdls など) をダブルクリックし、エクスポートされた WSDL が右ペインにリストされていることを確認します。

図 9-47 エクスポートされた WSDL



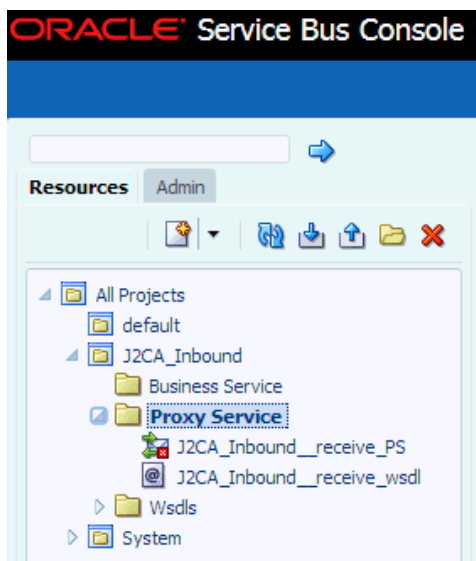
3. 「アクション」列で、「JCA バインド」に対応するアイコンをクリックします。
図 9-48 に示すように、「WSDL とサービスの生成」ページが表示されます。

図 9-48 「WSDL とサービスの生成」 ページ



4. 新規 WSDL の名前および新規プロキシ・サービス名を対応するフィールドに入力します。
5. 「宛先」領域で、プロキシ・サービスに指定されている使用可能なプロジェクトおよび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 9-49 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 とプロキシ・サービスがリストされていることを確認します。

図 9-49 生成された WSDL

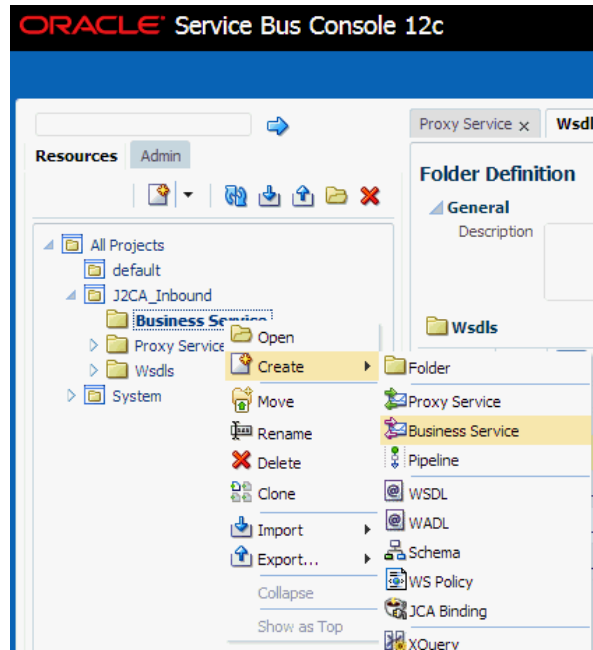


9.3.5 ファイル・タイプ・ビジネス・サービスの構成

ファイル・タイプのビジネス・サービスを構成するには、次のステップを実行します。

1. 図 9-50 に示すように、左ペインで作成済の「Business Service」フォルダを右クリックし、「作成」を選択し、「ビジネス・サービス」をクリックします。

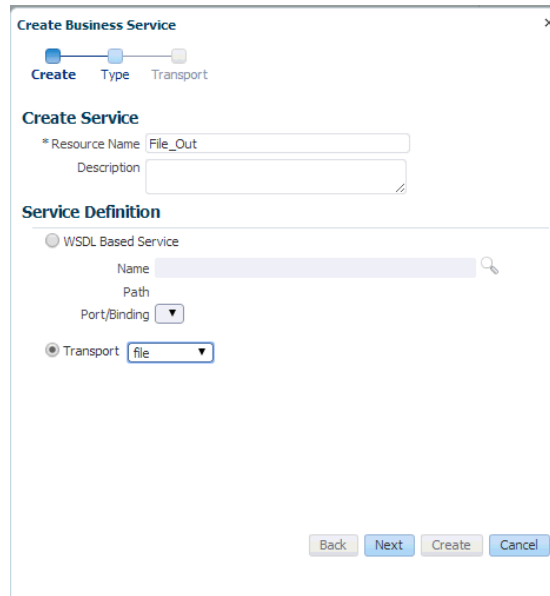
図 9-50 「Business Service」フォルダ



「ビジネス・サービスの作成」ウィンドウが表示されます。

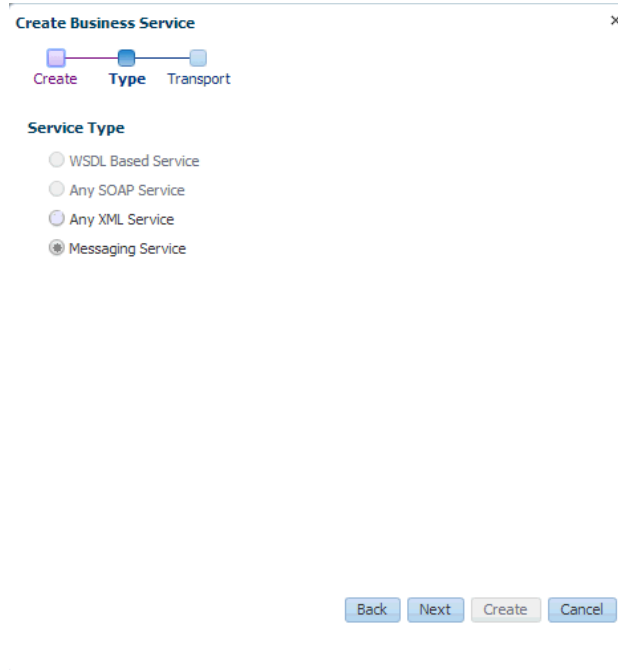
2. 図 9-51 に示すように、「リソース名」フィールドにビジネス・サービスの名前を指定し、サービス定義領域の「トランスポート」ドロップダウン・リストから「ファイル」オプションを選択します。

図 9-51 「ビジネス・サービスの作成」ウィンドウ



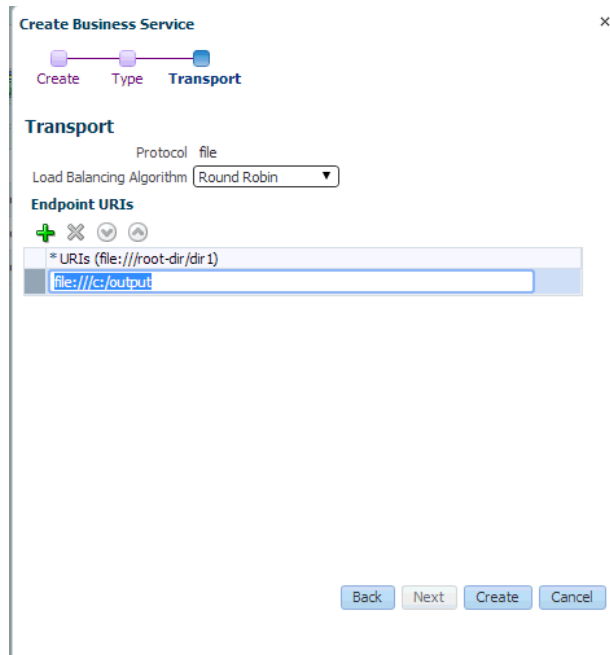
3. 「次へ」をクリックします。
4. 図 9-52 に示すように、「サービス・タイプ」領域で、サービス・タイプとして「メッセージ・サービス」を選択します。

図 9-52 「サービス・タイプ」領域



5. 「次へ」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。

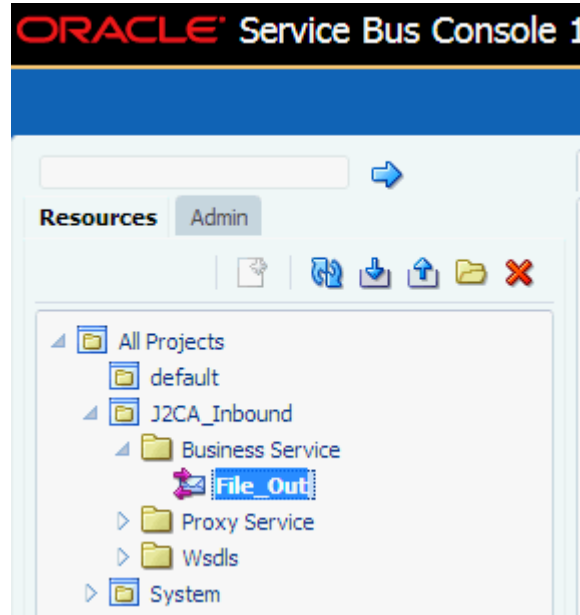
図 9-53 「トランスポート」ページ



6. 「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。

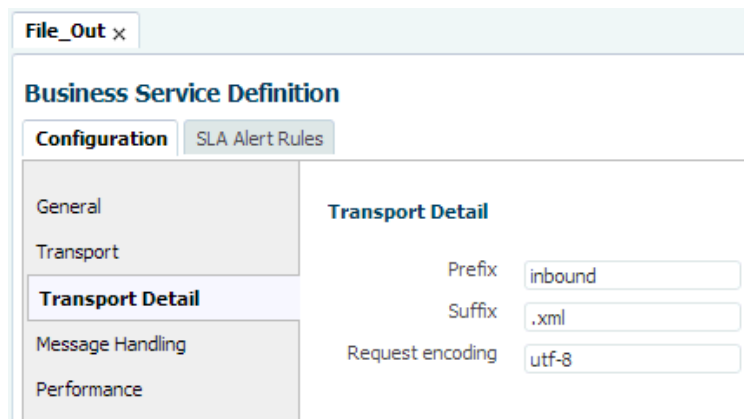
図 9-54 に示すように、ビジネス・サービス「File_Out」が作成され、「Business Service」の下にリストされます。

図 9-54 「File_Out」ビジネス・サービス



7. 図 9-55 に示すように、「File_Out」をダブルクリックし、左ペインで「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および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、右隅にある「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」アイコン

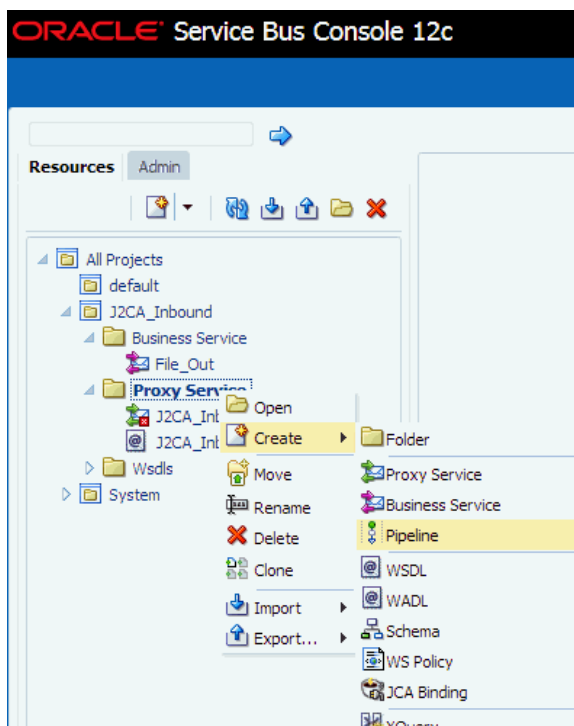


9.3.6 パイプラインの構成

パイプラインを構成するには、次のステップを実行します。

1. 図 9-57 に示すように、作成済のプロキシ・サービスを右クリックし、「作成」を選択し、「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. 図 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** ベースの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. 図 9-59 に示すように、「検索」アイコンをクリックし、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

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・ボックスを選択解除し、「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

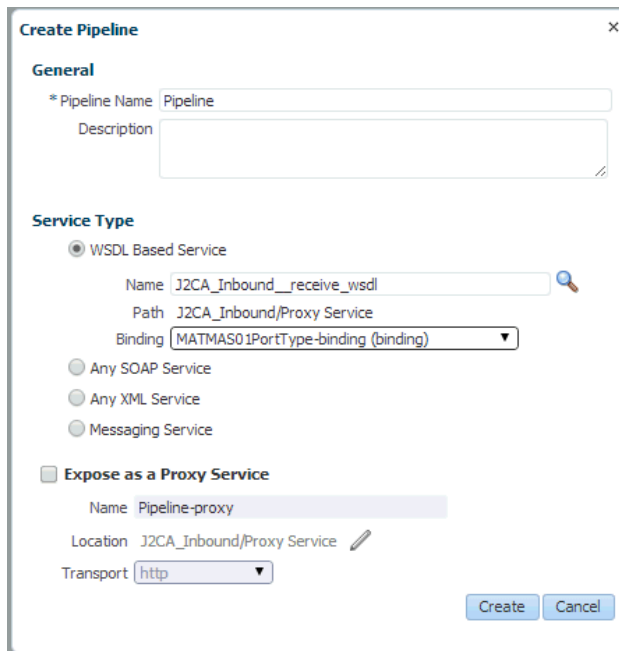
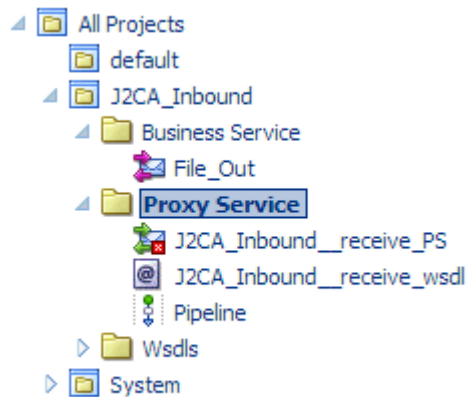


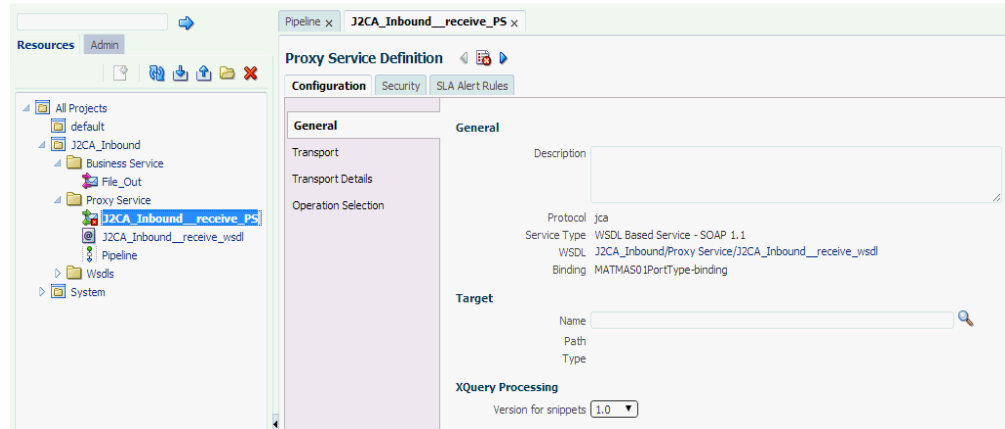
図 9-61 に示すように、パイプラインが作成され、「Proxy Service」の下にリストされます。

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、左ペインで「Proxy Service」の下の「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

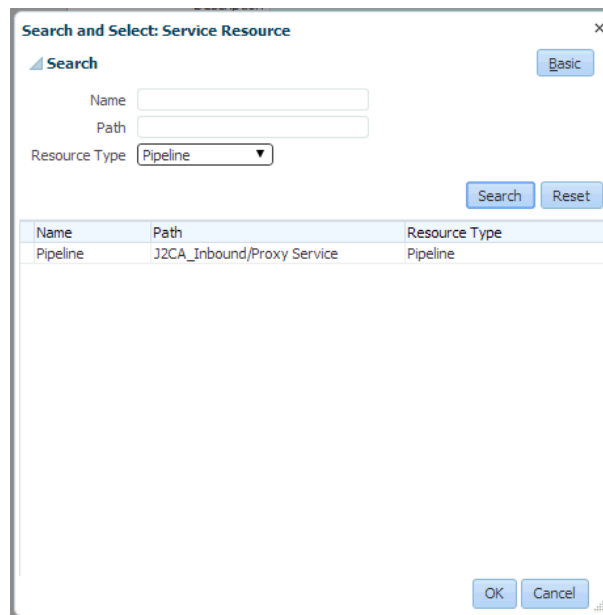


検索と選択: サービス・リソース・ウィンドウが表示されます。

6. 「リソース・タイプ」ドロップダウン・リストから「パイプライン」を選択し、「検索」ボタンをクリックします。

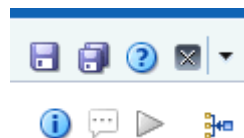
図 9-63 に示すように、パイプラインがリストされます。

図 9-63 検索と選択: サービス・リソース・ウィンドウ



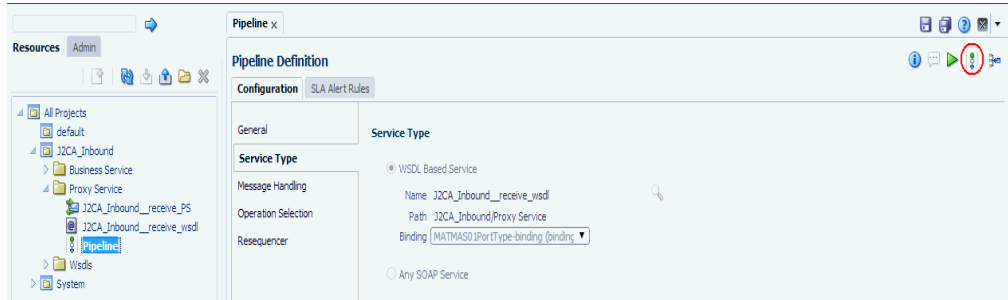
7. このパイプラインを選択して、「OK」をクリックします。
8. 図 9-64 に示すように、右隅にある「保存」または「すべて保存」アイコンをクリックします。

図 9-64 「保存」および「すべて保存」アイコン



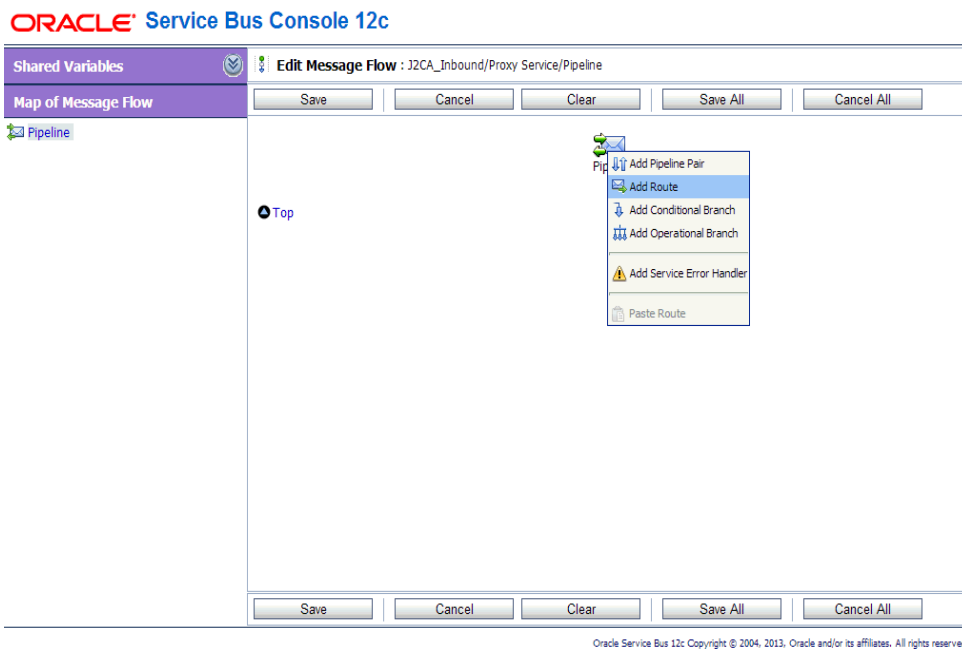
9. 図 9-65 に示すように、左ペインで「Proxy Service」フォルダの下の「Pipeline」をダブルクリックし、右ペインで下向きのアイコンをクリックしてメッセージ・フローを開きます。

図 9-65 メッセージ・フロー



10. 図 9-66 に示すように、表示された「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。

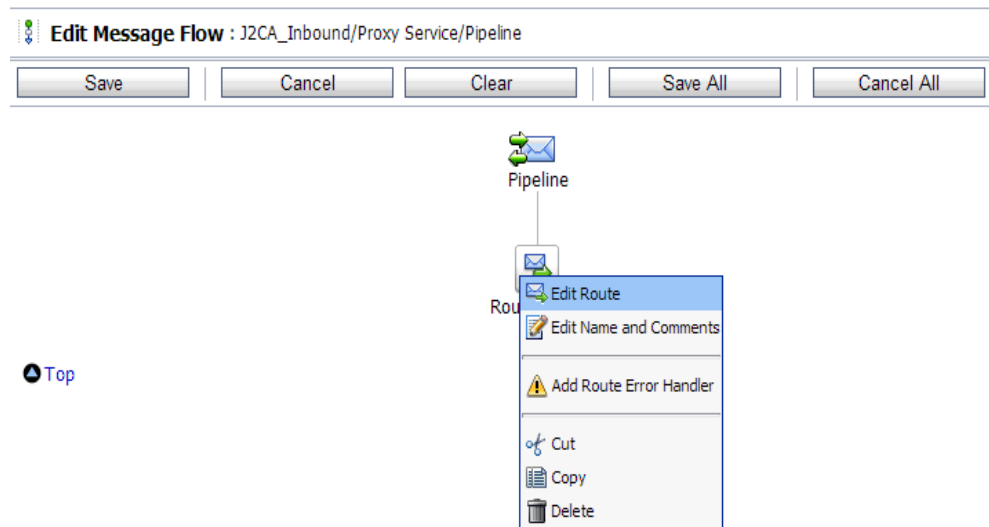
図 9-66 「ルートの追加」オプション



「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、メニューから「ルートの編集」を選択します。

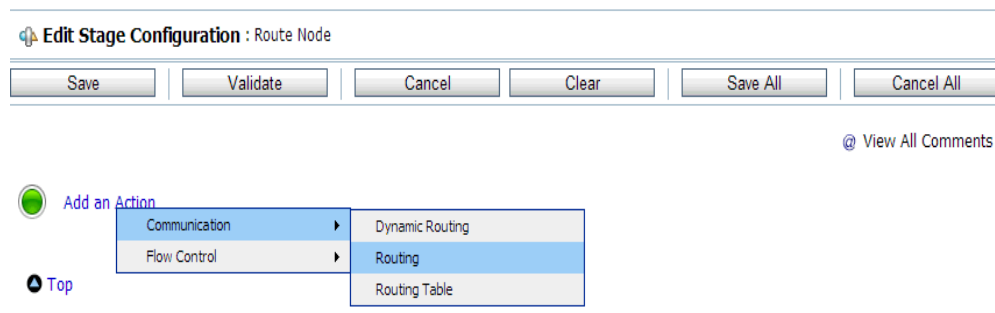
図 9-67 「ルートの編集」オプション



「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。

12. 図 9-68 に示すように、「アクションの追加」をクリックし、メニューから「通信」を選択し、「ルーティング」をクリックします。

図 9-68 「ステージ構成の編集」ワークスペース



13. 図 9-69 に示すように、「<サービス>」をクリックします。

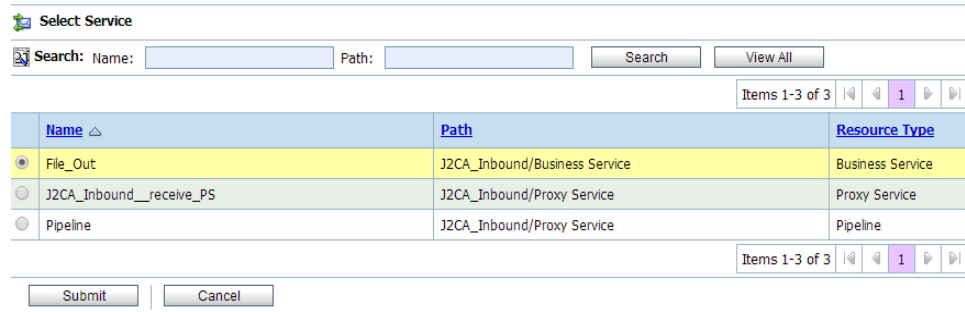
図 9-69 サービスのルート・アクション



「サービスの選択」ダイアログが表示されます。

14. 図 9-70 に示すように、「File_Out」ビジネス・サービスを選択し、「発行」をクリックします。

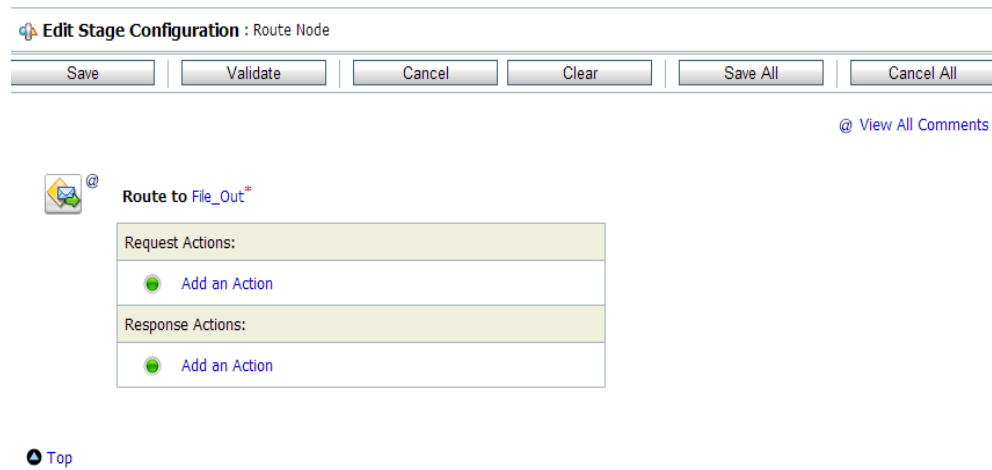
図 9-70 「サービスの選択」ダイアログ



「ステージ構成の編集」ワークスペース領域に戻ります。

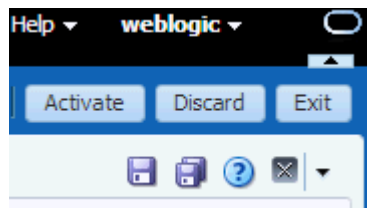
15. 図 9-71 に示すように、「すべて保存」をクリックします。

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で「アクティブ化」をクリックします。

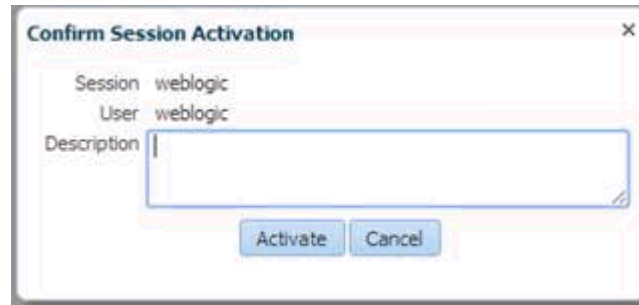
図 9-72 「アクティブ化」ボタン



セッション・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内容を保存します。

図 9-73 セッション・アクティブ化の確認ウィンドウ



18. SAP R/3 システムからのイベントをトリガーし、構成済の出力ロケーションで受信されたことを確認します。

9.4 sbconsole を使用したアウトバウンド処理の構成 (BSE 構成)

この項では、BSE 構成に sbconsole を使用して、アウトバウンド処理を構成する方法について説明します。

Application Adapters インストールの次のフォルダに、このアウトバンドの使用ケース・シナリオのサンプル・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
```

この節では、次のトピック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「Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・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 の公開」
- 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・サービスを使用した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 [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 Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定

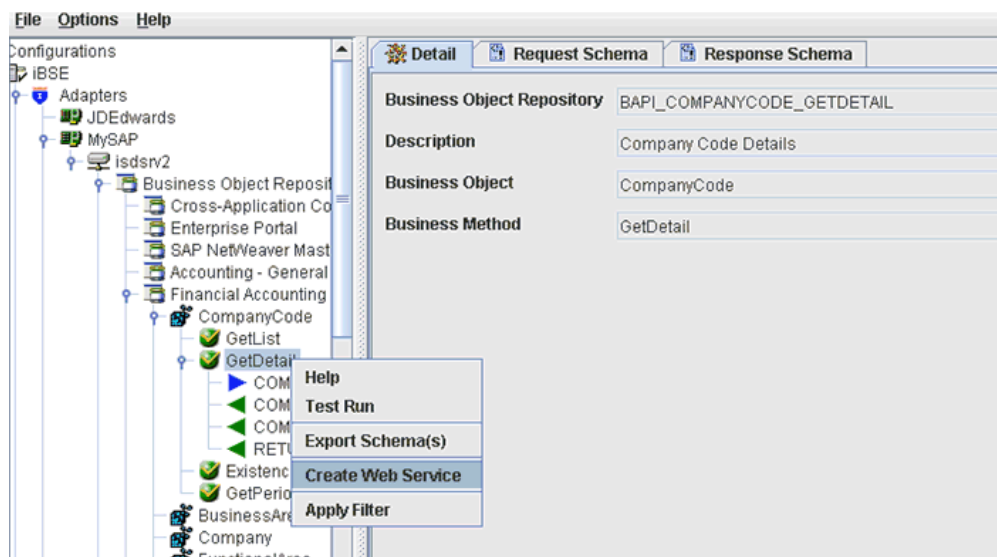
Oracle Service Bus 統合のためにアプリケーション・エクスプローラのクラス・パスを設定する方法の詳細は、9-6 ページ [9.2.2 項「Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定」](#) を参照してください。

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 の公開

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 を公開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、SAP R/3 ターゲットに接続します。

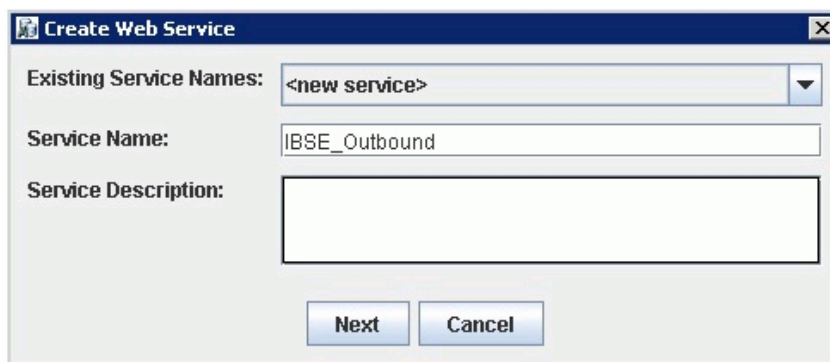
図 9-74 「Web サービスの作成」 オプション



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・オブジェクトの順に展開します。
3. 図 9-74 に示すように、「GetDetail」メソッドを右クリックし、メニューから「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て、「次へ」をクリックします。
5. 次に表示されるダイアログで、「OK」をクリックします。

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

6. 新規の Web サービスを右クリックし、メニューから「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 WSDL ファイルのデフォルトのファイル名が指定されます。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 ドキュメントの公開先となる場所を入力します。
ロケーションは、Oracle Service Bus プロジェクト名と、さらにオプションで 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フォルダ名はいずれもスラッシュ文字の「/」で区切る必要があります。
9. 「ホスト」フィールドで、Oracle Service Bus が稼働し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 へのアクセス用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 へのアクセス用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ている場所に公開され、Oracle Service Bus でビジネス・サービスまたはプロキシ・サービスとともに使用できるようになります。

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・タイプのビジネス・サービスを構成する方法の詳細は、9-10 ページ 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た WSDL タイプ・ビジネス・サービスの構成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、次のステップを実行します。

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

2. ビジネス・サービスの名前を入力し、サービス定義領域で WSDL ベースのサービス・オプションを選択し、「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

Name	Path	Namespace
BSE_Outbound_invoke	BSE_Outbound/...	urn:schemas-iwa...

3. 「検索」ボタンをクリックし、BSE アウトバウンド WSDL を選択し、「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。

4. 「次へ」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、「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

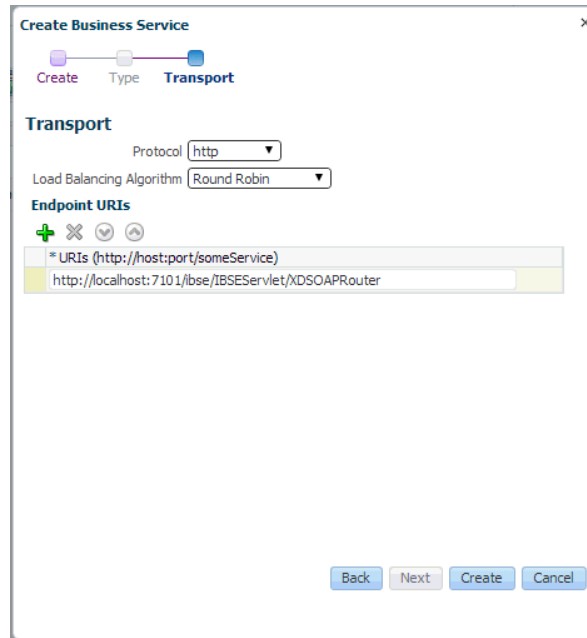
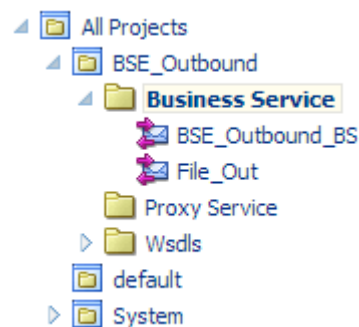


図 9-80 に示すように、作成された WSDL ベースのビジネス・サービスが「Business Service」フォルダの下にリストされます。

図 9-80 WSDL ベースのビジネス・サービス

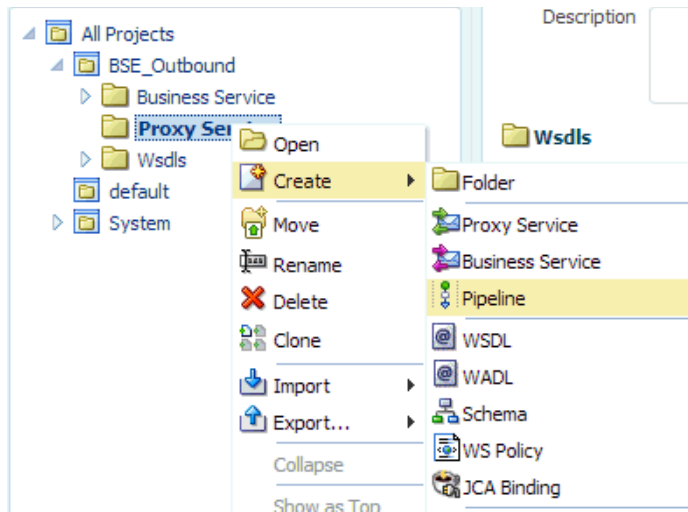


9.4.6 プロキシ・サービスを使用したパイプラインの構成

この項では、Oracle Service Bus コンソールを使用したプロキシ・サービスの構成方法について説明します。

1. 図 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、「パイプライン」をクリックします。

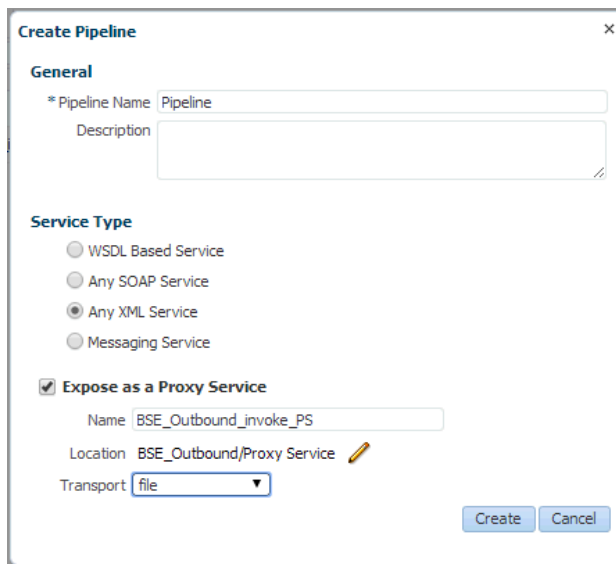
図 9-81 「パイプライン」オプション



パイプラインの作成ウィンドウが表示されます。

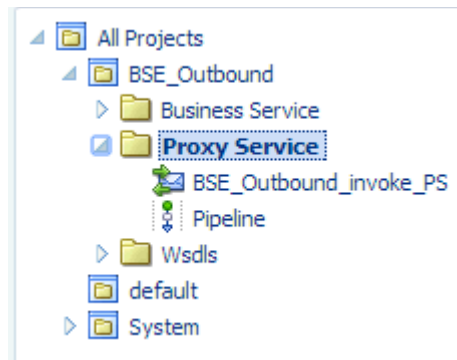
- 「パイプライン名」フィールドに名前を入力します。デフォルトでは、**プロキシ・サービスとして公開**が選択されています。プロキシ・サービス名を変更したい場合は、[図 9-82](#) に示すように、変更して「トランスポート」を「ファイル」に設定し、「作成」をクリックします。

図 9-82 パイプラインの作成ウィンドウ



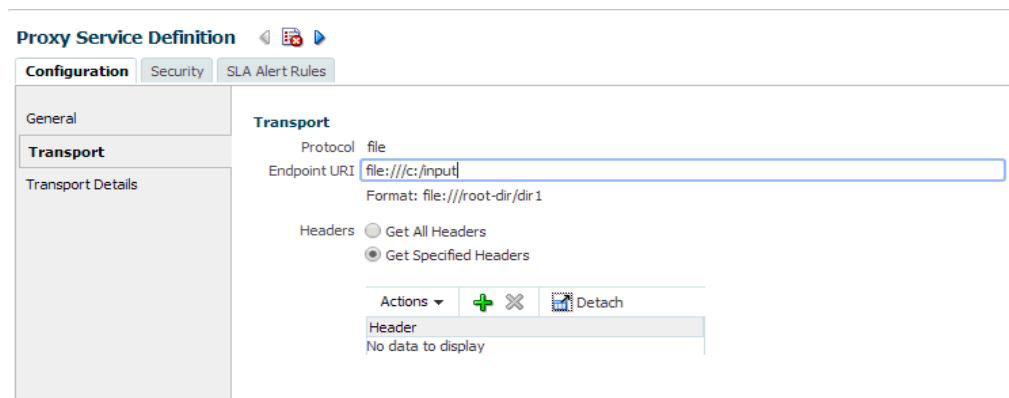
[図 9-83](#) に示すように、作成されたパイプラインとプロキシ・サービスが「Proxy Service」の下にリストされます。

図 9-83 パイプライン・ノード



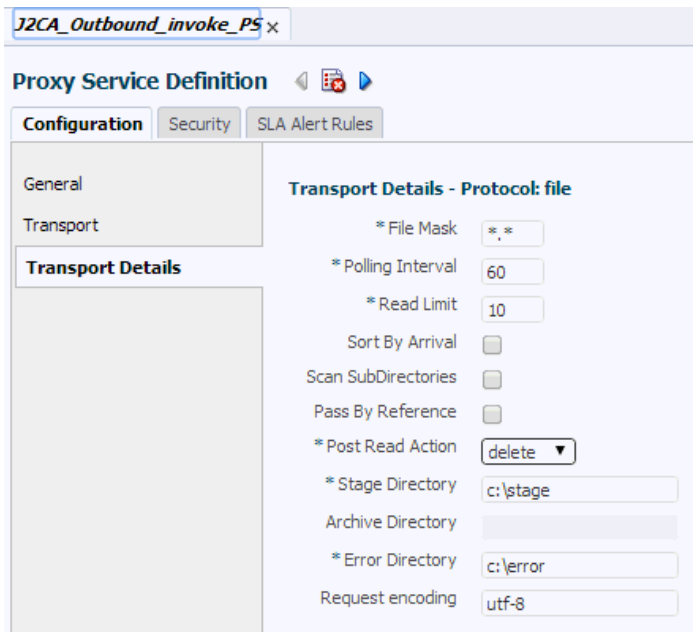
- 作成済のプロキシ・サービスをダブルクリックし、左ペインで「トランスポート」をクリックします。図 9-84 に示すように、「エンドポイント URI」フィールドに入力ロケーションを指定します。

図 9-84 「トランスポート」



- 図 9-85 に示すように、左ペインで「トランスポートの詳細」をクリックし、「ステージ・ディレクトリ」フィールドと「エラー・ディレクトリ」フィールドにロケーションを指定します。

図 9-85 「トランスポートの詳細」



5. 図 9-86 に示すように、右隅にある「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



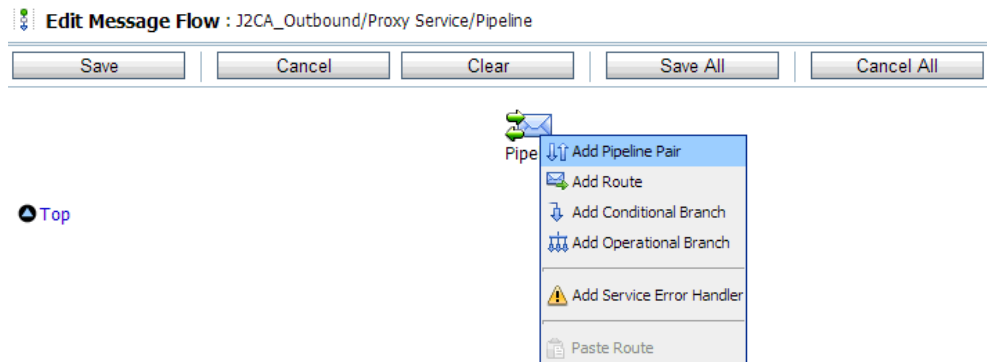
6. 図 9-87 に示すように、「パイプライン」ノードをダブルクリックし、右ペインでメッセージ・フローを開くアイコンをクリックしてメッセージ・フローを開きます。

図 9-87 メッセージ・フローを開くアイコン



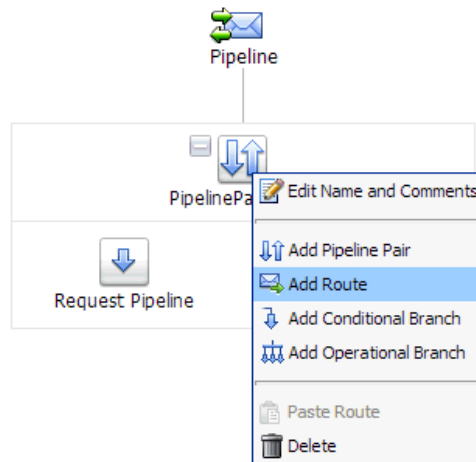
7. 図 9-88 に示すように、「プロキシ・サービス」アイコンをクリックし、メニューから「パイプライン・ペアの追加」を選択します。

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、メニューから「ルートへの追加」を選択します。

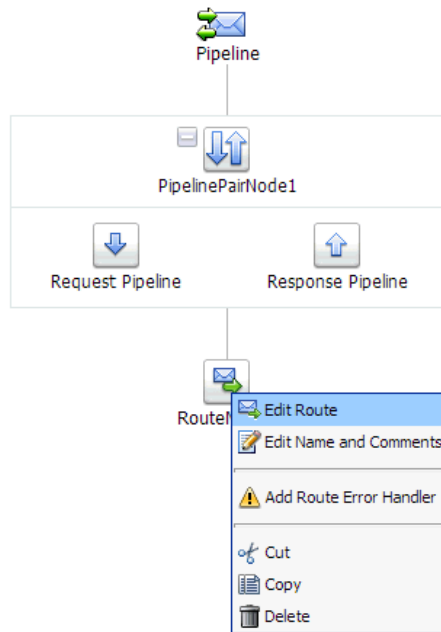
図 9-89 「ルートへの追加」オプション



「RouteNode1」アイコンが「PipelinePairNode1」アイコンの下に追加されます。

9. 図 9-90 に示すように、「RouteNode1」アイコンをクリックし、メニューから「ルートへの編集」を選択します。

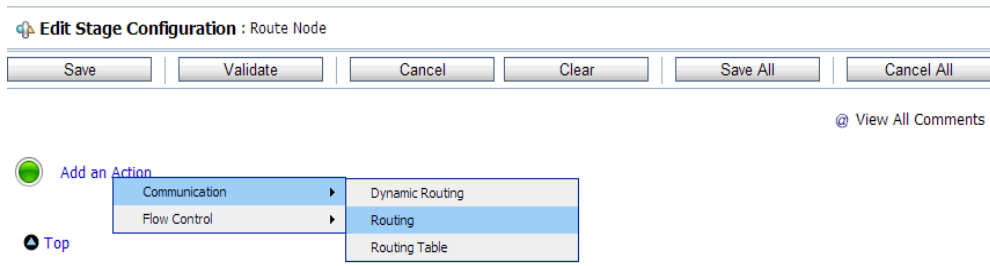
図 9-90 「ルート編集」オプション



「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 9-91 に示すように、「アクションの追加」をクリックし、「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、「<サービス>」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、「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

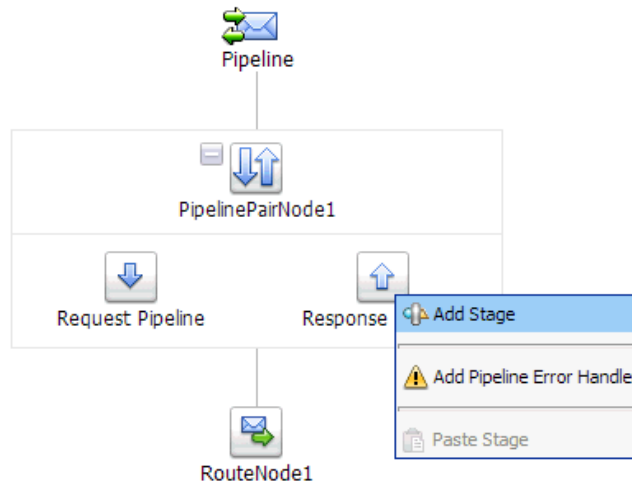
Name	Path	Resource Type
BSE_Outbound_BS	BSE_Outbound/Business Service	Business Service
BSE_Outbound_invoke_PS	BSE_Outbound/Proxy Service	Proxy Service
File_Out	BSE_Outbound/Business Service	Business Service
Pipeline	BSE_Outbound/Proxy Service	Pipeline

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、メニューから「ステージの追加」を選択します。

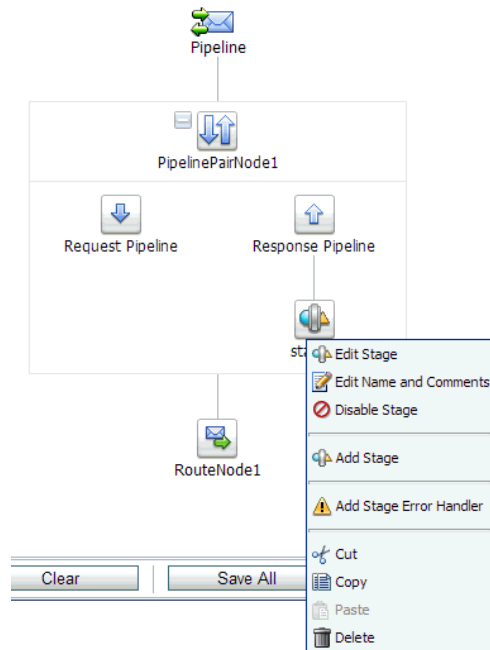
図 9-95 「レスポンス・パイプライン」アイコン



ステージ 1 アイコンが「リクエスト・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、「Stage1」アイコンをクリックし、メニューから「ステージの編集」を選択します。

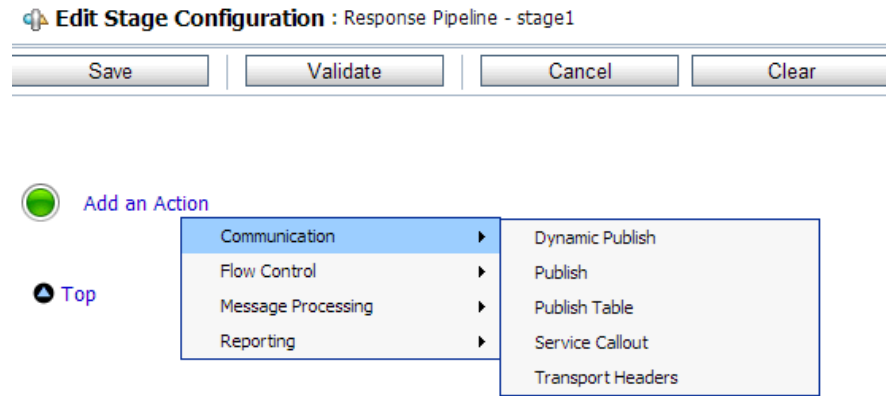
図 9-96 「ステージの編集」オプション



「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。

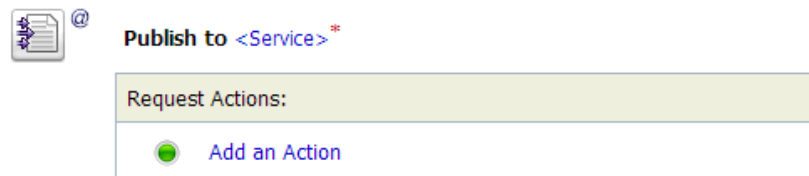
16. 図 9-97 に示すように、「アクションの追加」をクリックし、「通信」を選択し、「パブリッシュ」をクリックします。

図 9-97 「ステージ構成の編集」ワークスペース領域



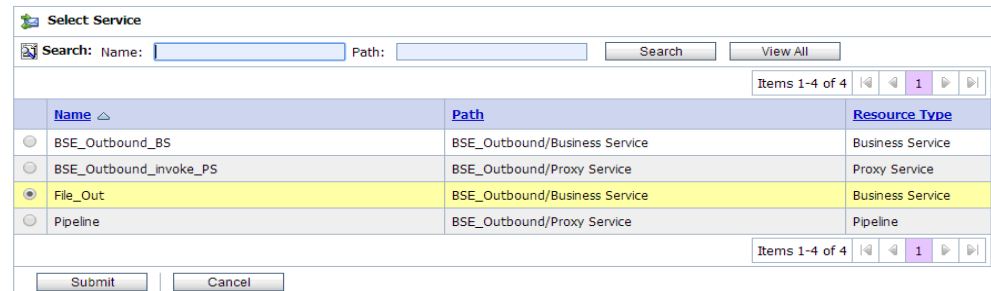
17. 図 9-98 に示すように、「<サービス>」をクリックします。

図 9-98 <サービス> アクション



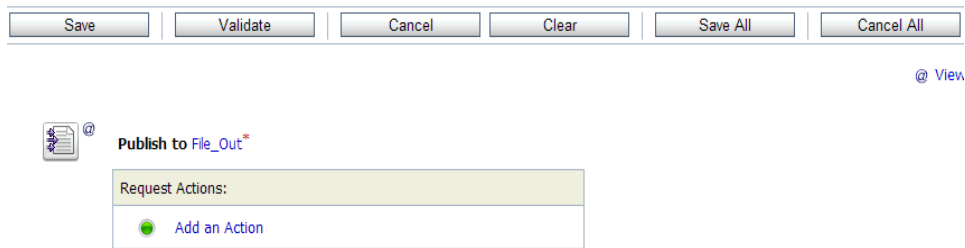
18. 図 9-99 に示すように、「サービスの選択」ダイアログで、ファイル・タイプのビジネス・サービスを選択し、「発行」をクリックします。

図 9-99 「サービスの選択」ダイアログ



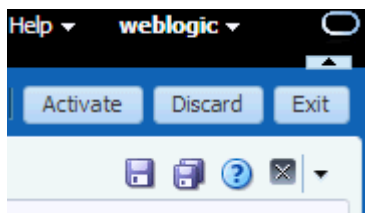
19. 図 9-100 に示すように、「すべて保存」をクリックします。

図 9-100 「すべて保存」 ボタン



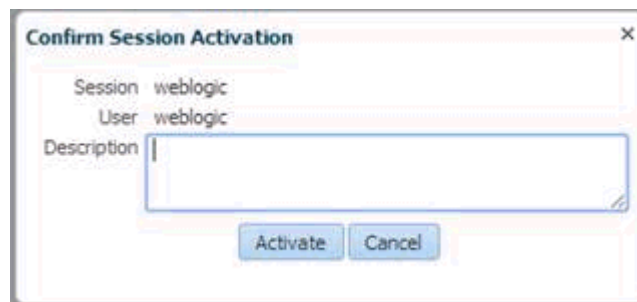
20. 図 9-101 に示すように、Oracle Service Bus セッションの右ペインで「アクティブ化」をクリックします。

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内容を保存します。

図 9-102 セッション・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 ファイルをコピーおよび貼付けます。
出力が構成済の出力ロケーション (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 イベントの作業 (J2CA 構成)

この項では、Oracle Service Bus を使用して同期 SAP R/3 イベントを作業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： 同期イベントはリプライ・スキーマのみを所有しているオブジェクトでサポートされ、IDoc ではサポートされていません。

Application Adapters インストール後の次のフォルダに、この使用ケース・シナリオのサンプル・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_Synchronous_Events
```

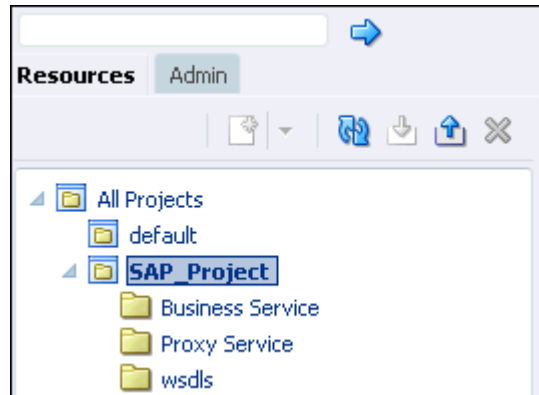
この節では、次のトピックについて説明します。

- 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「メッセージ用の 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、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



詳細については、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 に対して 2 つのアダプタ・ターゲットと 1 つのチャネルを作成します。

詳細については、第 4 章「Oracle Application Adapter for SAP R/3 の構成」を参照してください。

9.5.3 メッセージ用の WSDL ファイルの生成

次のステップを実行して、メッセージ用の WSDL ファイルを生成します。

1. WSDL ファイルを生成する前に、チャネルが開始し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 Business Object** を選択します)。
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 Business Object** を選択します)。

詳細については、[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 の公開」](#) 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 の公開」](#) 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、左ペインで作成済の WSDL フォルダを選択します。

[図 9-104](#) に示すように、右ペインに WSDL ファイル、JCA バインド・ファイル、XSD ファイルがリストされます。

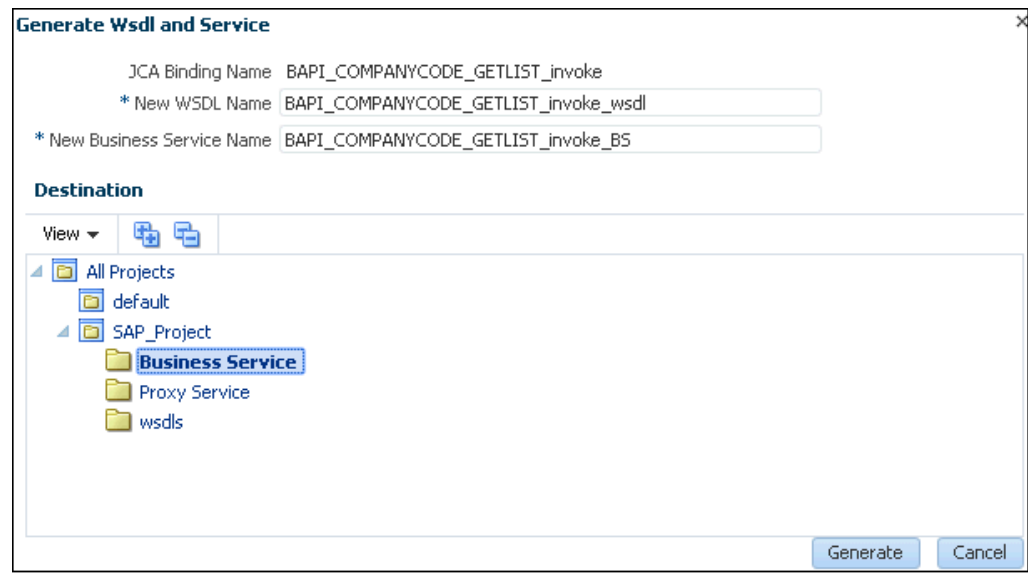
図 9-104 WSDL フォルダの内容

Name	Type	Actions
...	Folder	
BAPI_COMPANYCODE_GETLIST_invoke	JCA Binding	
BAPI_COMPANYCODE_GETLIST_invoke	WSDL	
BAPI_COMPANYCODE_GETLIST_invoke_request	Schema	
BAPI_COMPANYCODE_GETLIST_invoke_response	Schema	
BAPI_COMPANYCODE_GETLIST_receive	JCA Binding	
BAPI_COMPANYCODE_GETLIST_receive	WSDL	
BAPI_COMPANYCODE_GETLIST_receive_request	Schema	
BAPI_COMPANYCODE_GETLIST_receive_response	Schema	

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。

[図 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。

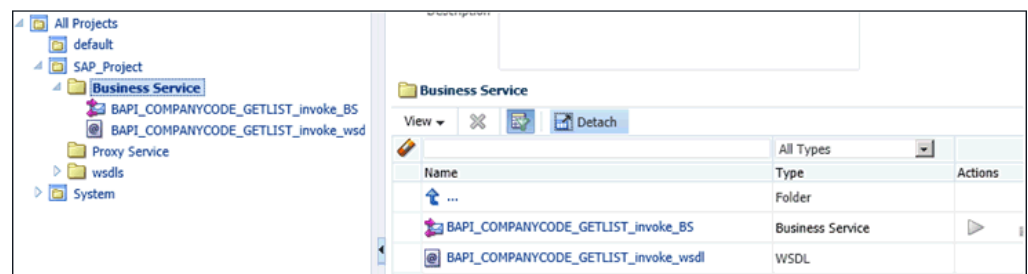
図 9-105 「WSDL とサービスの生成」 ページ



3. 新規 WSDL の名前および新規ビジネス・サービス名を対応するフィールドに入力します。
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、左ペインでプロジェクト・フォルダ (SAP_Project など) の下にある「Business Service」フォルダを選択し、WSDL とサービス・リソースが正しく作成されていることを確認します。

図 9-106 WSDL とサービス・リソースの確認

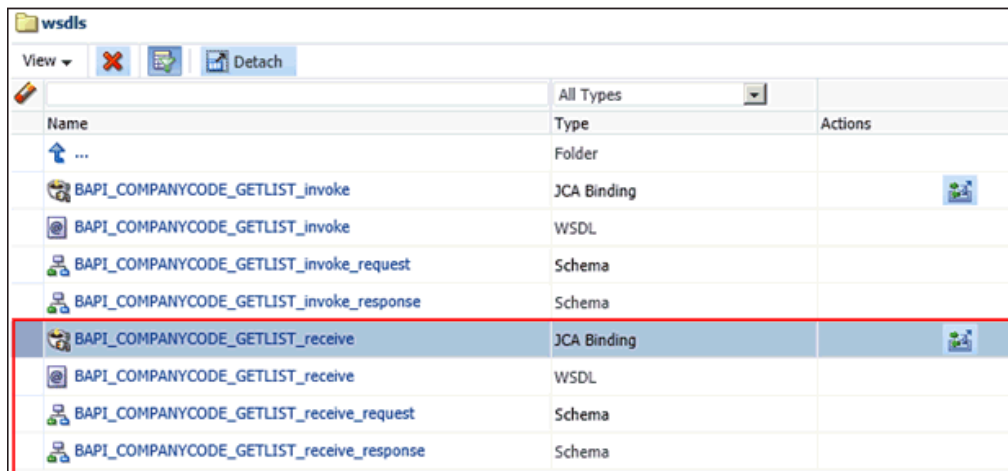


9.5.5 Oracle Service Bus コンソールでのプロキシ・サービスの作成

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、作成済の WSDL フォルダを選択します。
 図 9-107 に示すように、右ペインに WSDL ファイル、JCA バインド・ファイル、XSD ファイルがリストされます。

図 9-107 WSDL フォルダの内容



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。

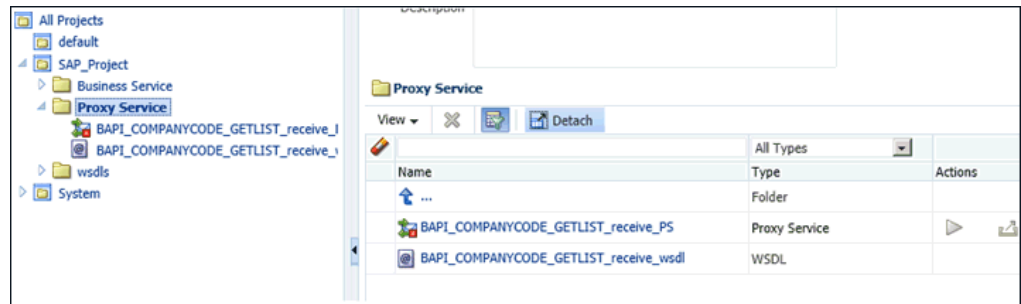
図 9-108 「WSDL とサービスの生成」ページ



3. 新規 WSDL の名前および新規プロキシ・サービス名を対応するフィールドに入力します。
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、左ペインでプロジェクト・フォルダ (SAP_Project など) の下にある「Proxy Service」フォルダを選択し、WSDL とサービス・リソースが正しく作成されていることを確認します。

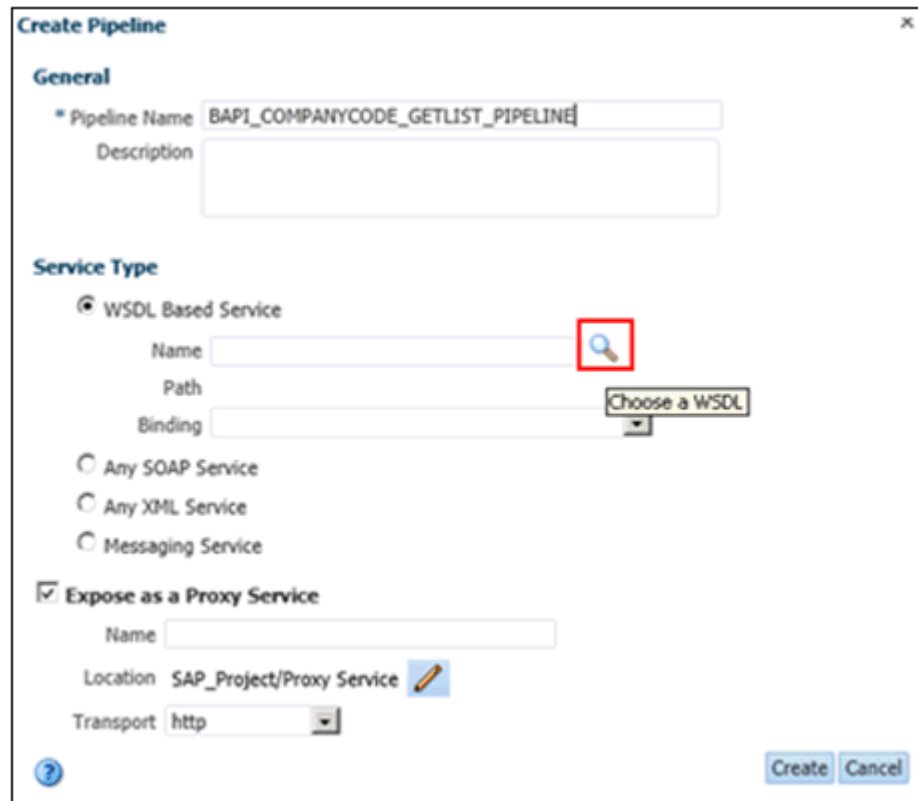
図 9-109 WSDL とサービス・リソースの確認



6. 「Proxy Service」を右クリックし、「作成」を選択し、「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。

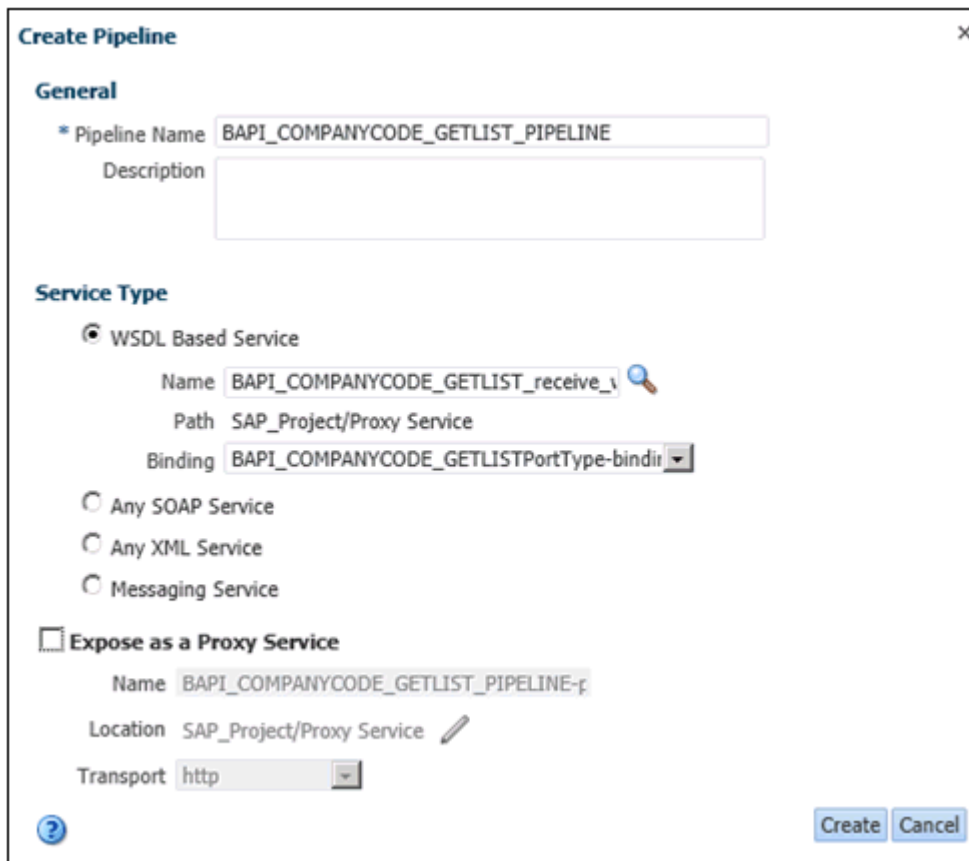
図 9-110 パイプラインの作成ウィンドウ



7. パイプライン名を入力し、サービス・タイプとして **WSDL** ベースのサービスを選択し、「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。
8. 「検索」をクリックし、表示されるリストで、「Proxy Service」の下にある「BAPI_COMPANYCODE GETLIST_receive_wsdI」を選択し、「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

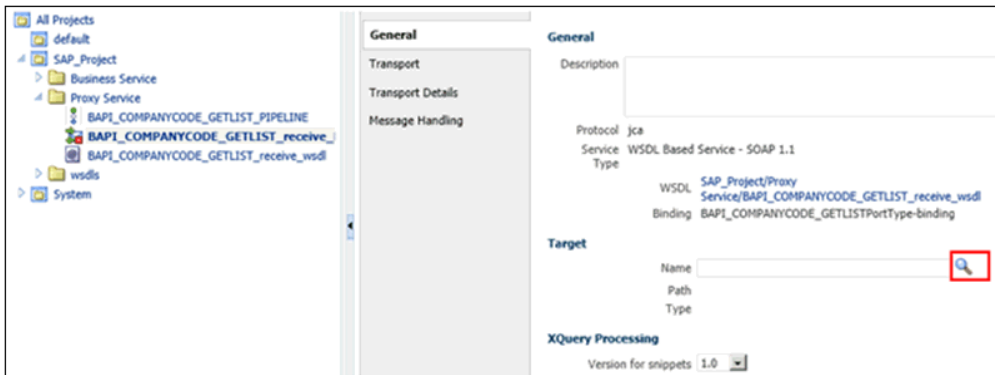


9. プロキシ・サービスとして公開オプションを選択解除し、「作成」をクリックします。

パイプラインが作成され、「Proxy Service」フォルダの下にリストされます。

10. 図 9-112 に示すように、「Proxy Service」フォルダの下の「BAPI_COMPANYCODE_GETLIST_receive_PS」をダブルクリックし、左ペインの「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、表示されるリストで、「Proxy Service」の下にある「BAPI COMPANYCODE_GETLIST_PIPELINE」を選択し、「OK」をクリックします。
12. 左隅にある「保存」または「すべて保存」をクリックします。

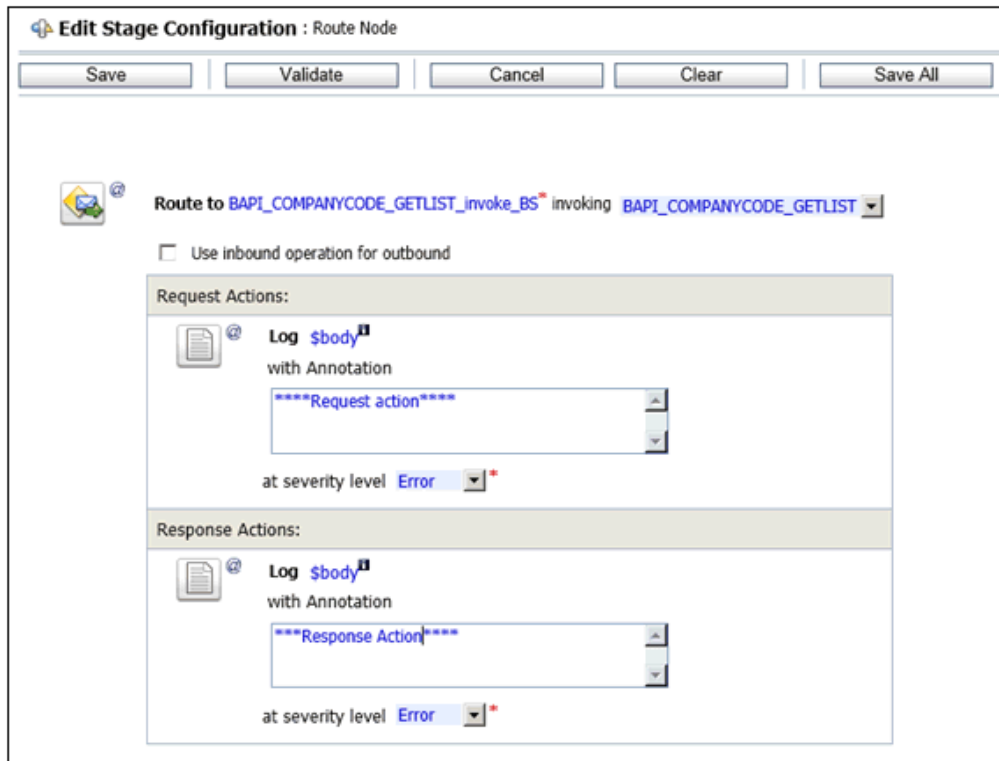
9.5.6 プロキシ・サービスでのパイプラインの構成

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」フォルダの下にある作成済のパイプラインをクリックし、右ページ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、表示された「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、メニューから「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、メニューから「通信」を選択し、「ルーティング」をクリックします。
5. <サービス>をクリックします。
6. 作成済のビジネス・サービスを選択し、「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」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. <ログ式>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、「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」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. <ログ式>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、「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、最終的なパイプライン構造を示します。

図 9-113 最終的なパイプライン構造



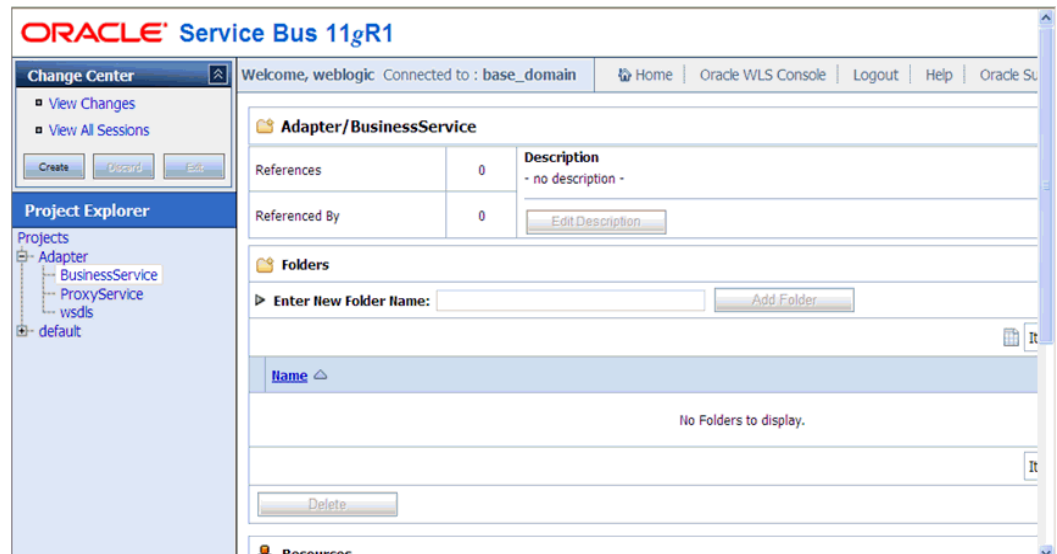
16. 「保存」をクリックします。
「メッセージ・フローの編集」ワークスペース領域に戻ります。
17. 「保存」をクリックします。
18. 画面の上部にある「アクティブ化」をクリックします。
表示されるセッション・アクティブ化の確認ウィンドウで、「アクティブ化」をクリックします。

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 構成用に 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。

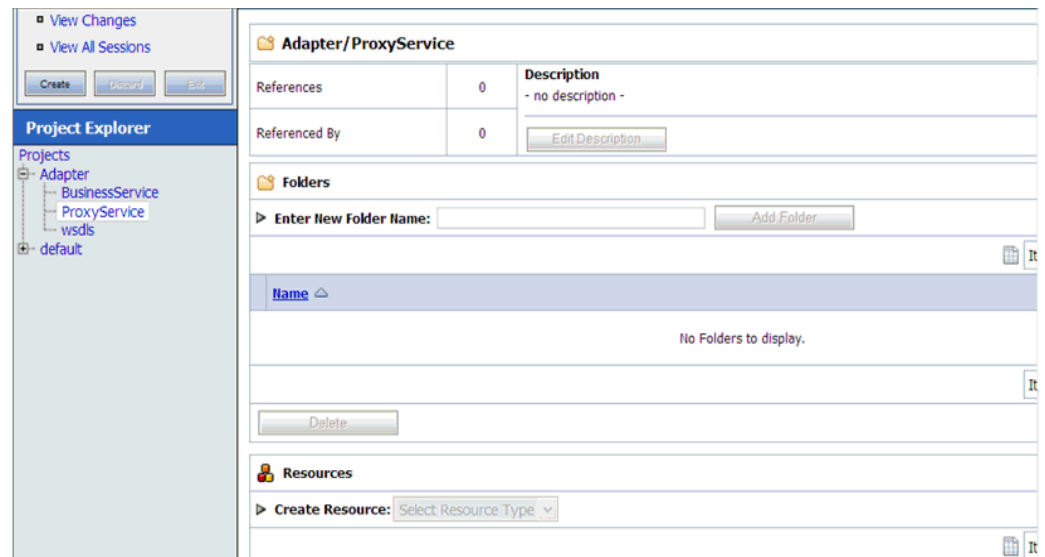
1. Oracle Service Bus を起動し、必要なプロジェクト・フォルダを作成します。
詳細については、[9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」](#)を参照してください。
2. アプリケーション・エクスプローラから WSDL を生成し、作成済のプロジェクト・フォルダに公開します。公開された WSDL を使用して、ビジネス・サービスを作成します。
詳細については、[9.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL の公開」](#)を参照してください。
3. [図 9-114](#) 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



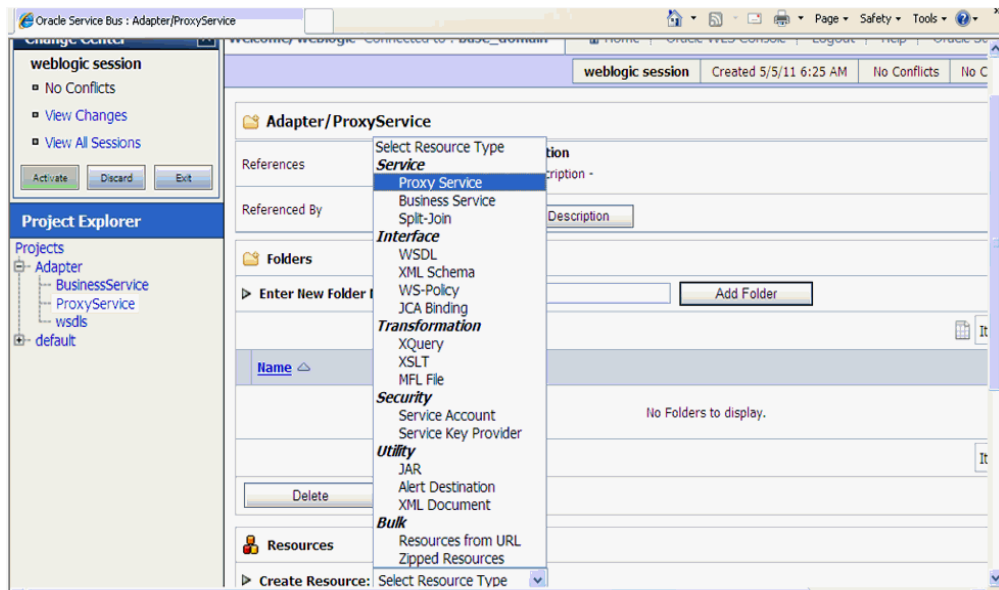
4. 図 9-115 に示すように、左ペインで「ProxyService」プロジェクト・フォルダを選択し、「作成」をクリックします。

図 9-115 プロキシ・サービス



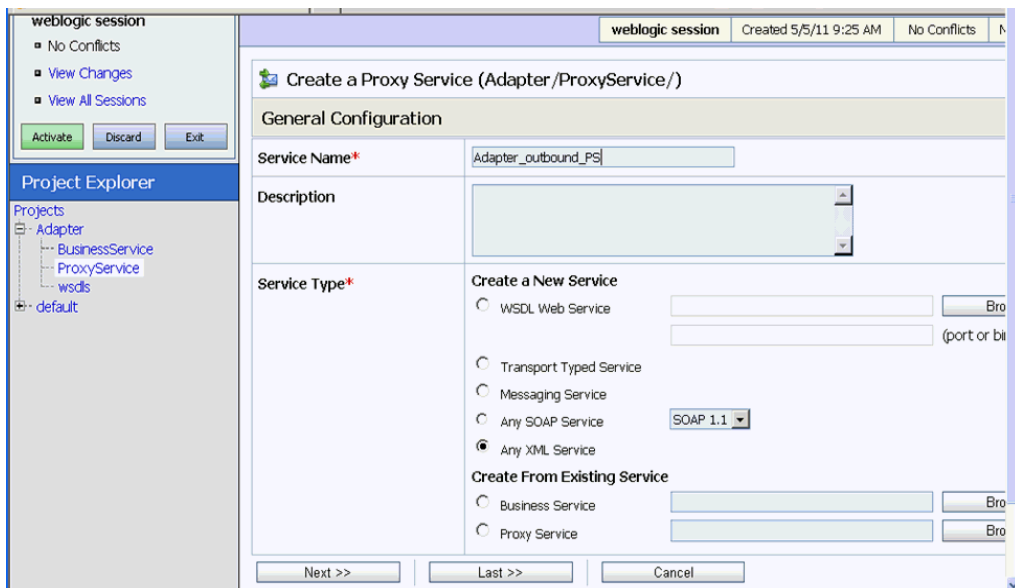
5. 図 9-116 に示すように、右ペインで「リソースの作成」リストから「プロキシ・サービス」を選択します。

図 9-116 「リソースの作成」メニュー



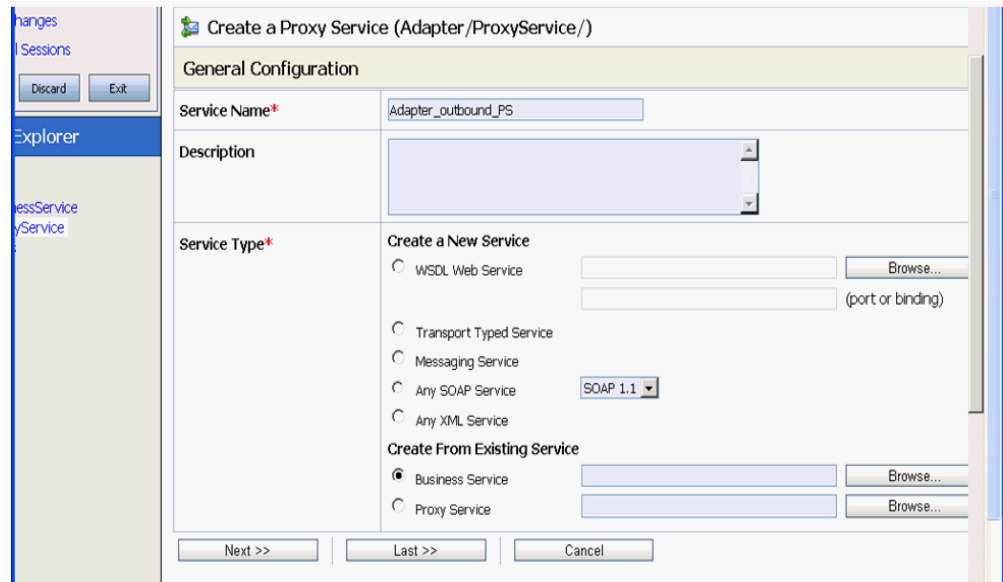
6. 図 9-117 に示すように、「サービス名」フィールドで適切な名前を入力します。

図 9-117 「サービス名」



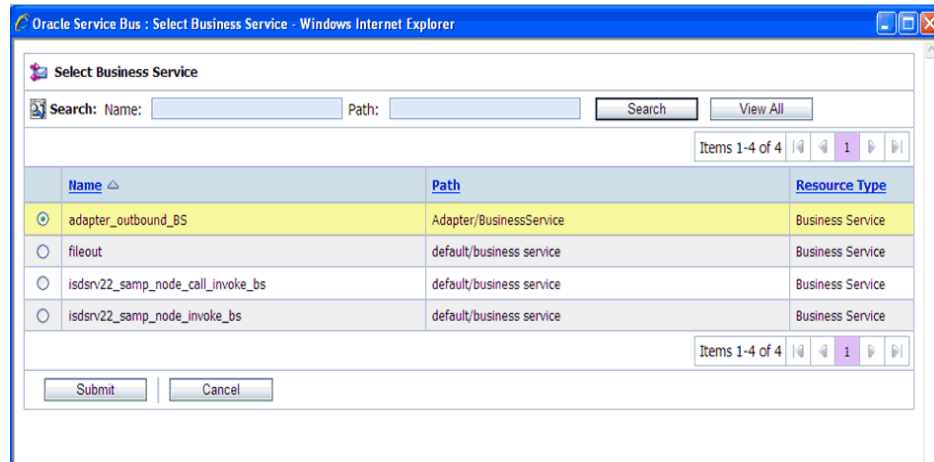
7. 図 9-118 に示すように、「サービス・タイプ」セクションの「既存のサービスから作成」で「ビジネス・サービス」ラジオ・ボタンを選択し、「参照」をクリックします。

図 9-118 ビジネス・サービス



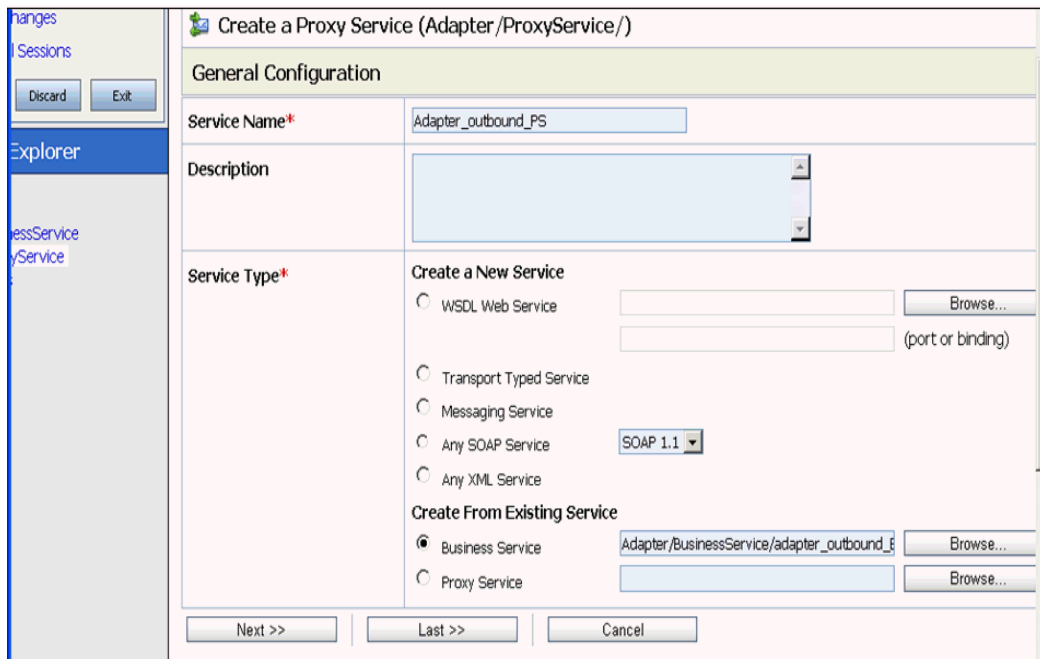
8. 図 9-119 に示すように、既存のビジネス・サービスを選択し、「発行」をクリックします。

図 9-119 既存のビジネス・サービス



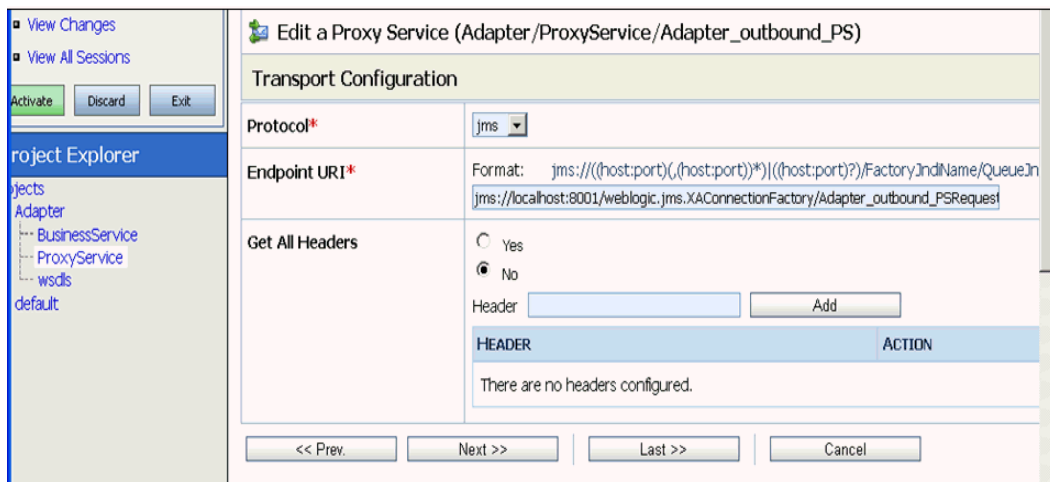
9. 図 9-120 に示すように、「次へ」をクリックします。

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、「次へ」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。
- a. 「宛先タイプ」セクションで「キュー」を選択します。
 - b. 「レスポンスが必要」チェック・ボックスを有効にします。
 - c. 「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。
 - d. 「レスポンス URI」フィールドに、トランスポート構成で使用したエンドポイント URI を入力し、「リクエスト」を「レスポンス」に変更します。
例を以下に示します。

```
jms://localhost:8001/weblogic.jms.XAConnectionFactory/Adapter_outbound_PSResponse
```

図 9-122 プロキシ・サービスの編集

12. 「次へ」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、「次へ」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、「次へ」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
	Buffer Type
	<input type="radio"/> Memory Buffer <input type="radio"/> Disk Buffer
	Compression
	<input type="checkbox"/> Enabled

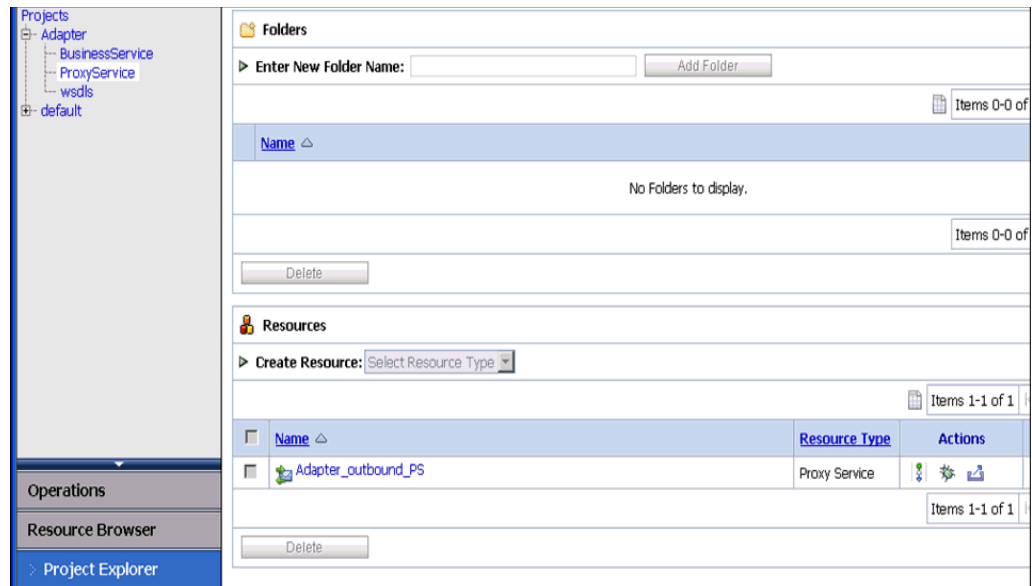
15. 図 9-125 に示すように、「保存」をクリックします。

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

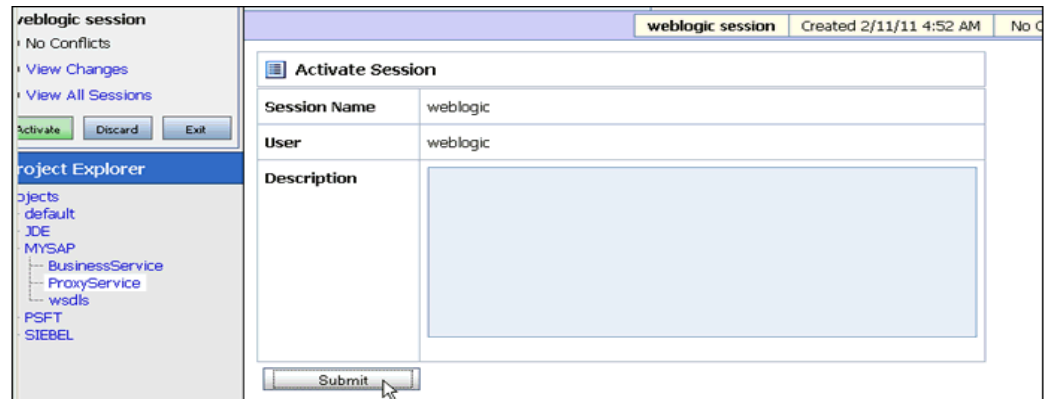
図 9-126 に示すように、作成されたプロキシ・サービスが保存されます。

図 9-126 プロキシ・サービス



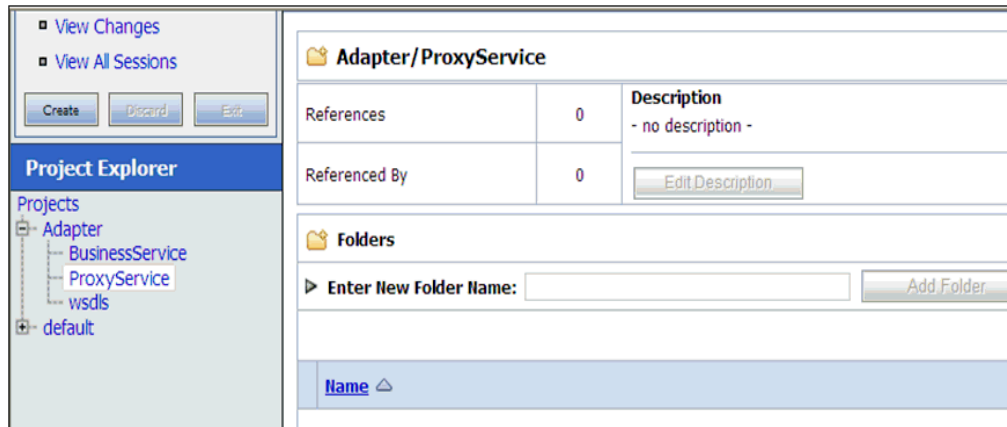
16. 図 9-127 に示すように、左ペインで「アクティブ化」をクリックし、「発行」をクリックします。

図 9-127 「セッションのアクティブ化」



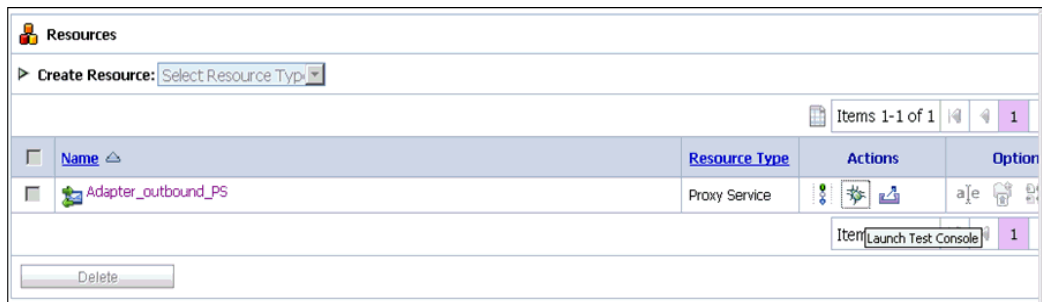
17. 図 9-128 に示すように、左ペインで「Projects」フォルダの下にある「ProxyService」をクリックします。

図 9-128 Adapter/ProxyService



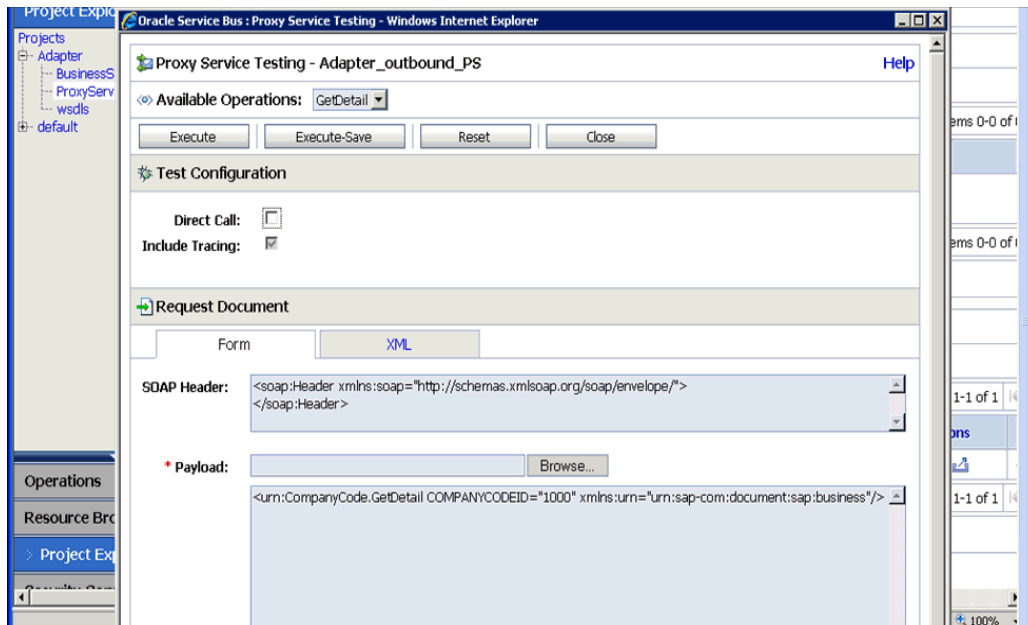
18. 図 9-129 に示すように、作成済のプロキシ・サービスの「テスト・コンソールの起動」アイコンをクリックします。

図 9-129 「テスト・コンソールの起動」アイコン



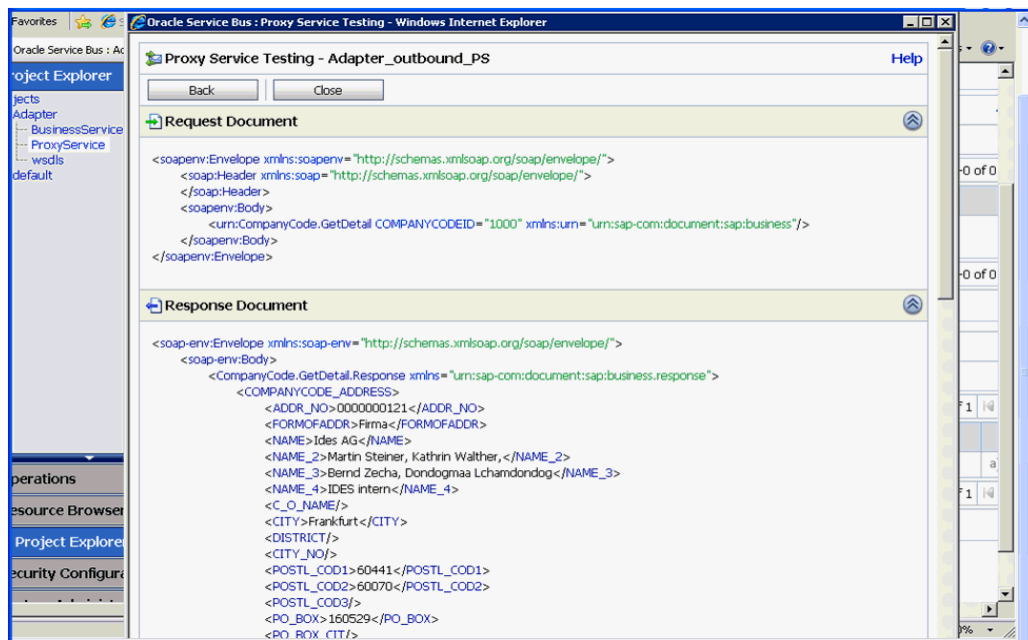
19. 図 9-130 に示すように、「ペイロード」に入力値を指定し、「直接呼出し」ボックスを選択解除し、「実行」をクリックします。

図 9-130 プロキシ・サービスのテスト



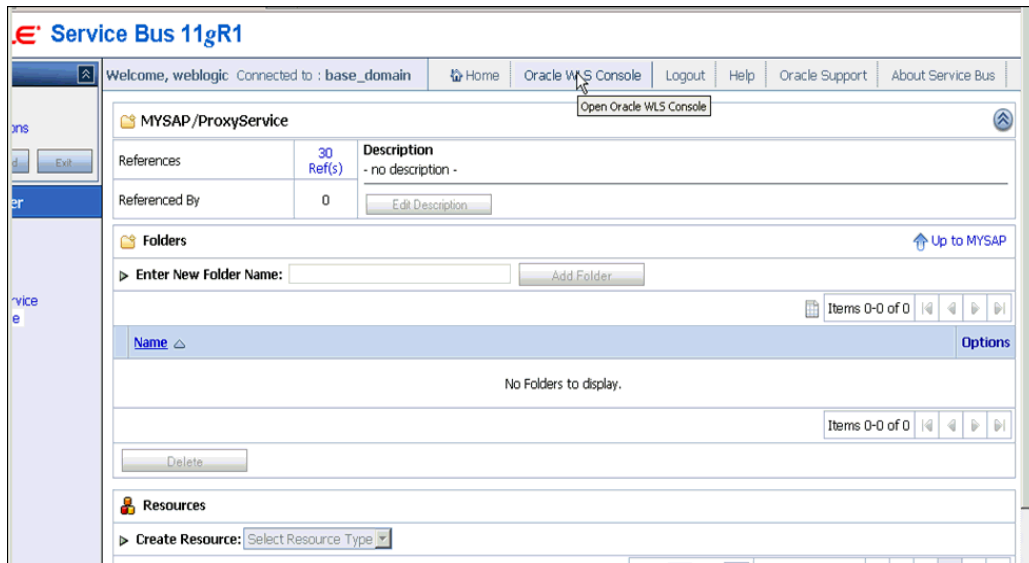
20. 図 9-131 に示すように、レスポンス・ドキュメントをレビューし、「閉じる」をクリックします。

図 9-131 レスポンス・ドキュメント



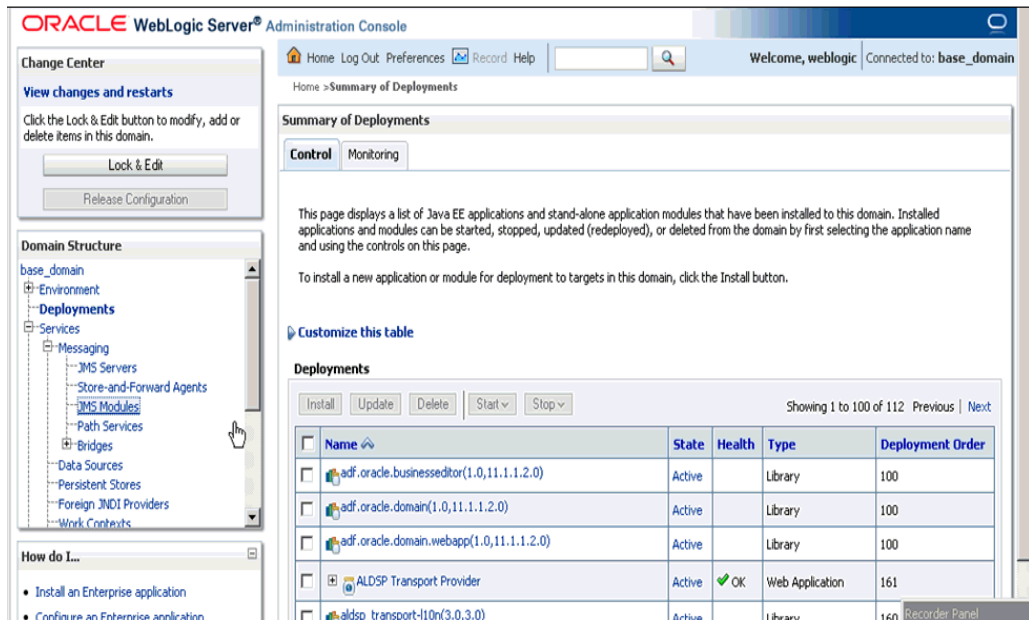
21. 図 9-132 に示すように、「Oracle WLS コンソール」タブをクリックします。

図 9-132 ProxyService



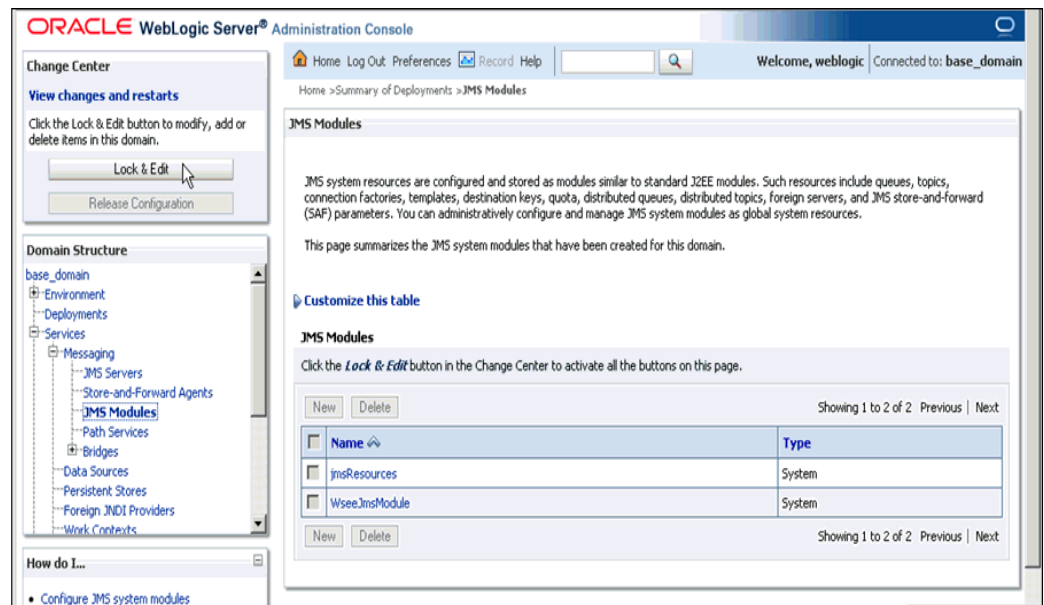
22. 図 9-133 に示すように、「Oracle WLS コンソール」で「サービス」を展開し、「メッセージング」を展開し、「JMS モジュール」をクリックします。

図 9-133 「Oracle WLS コンソール」



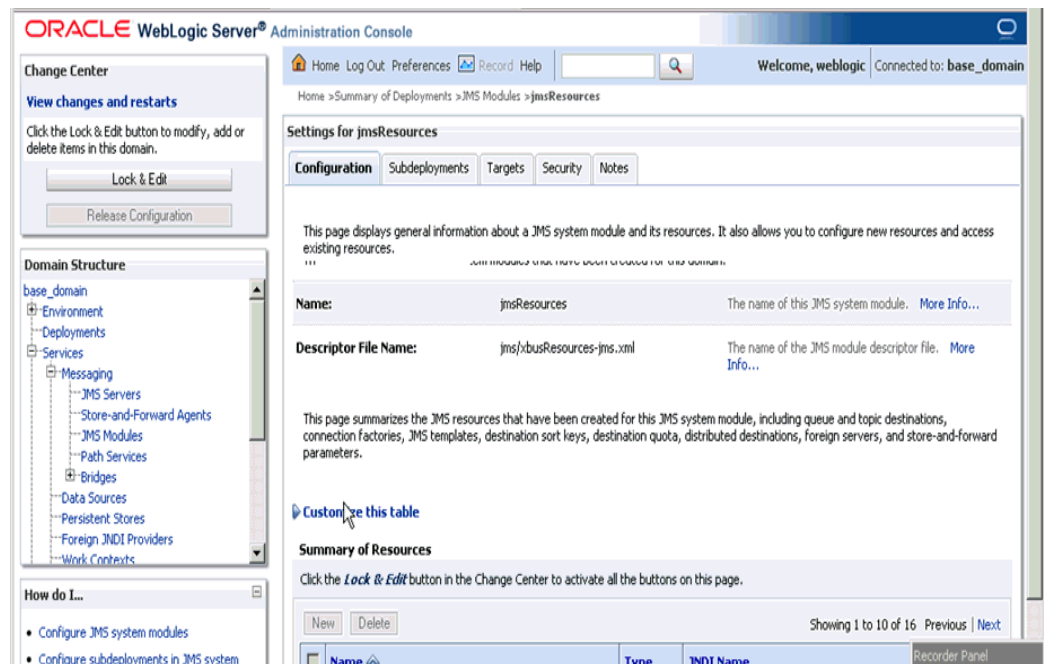
23. 図 9-134 に示すように、「jmsResources」をクリックします。

図 9-134 「JMS モジュール」



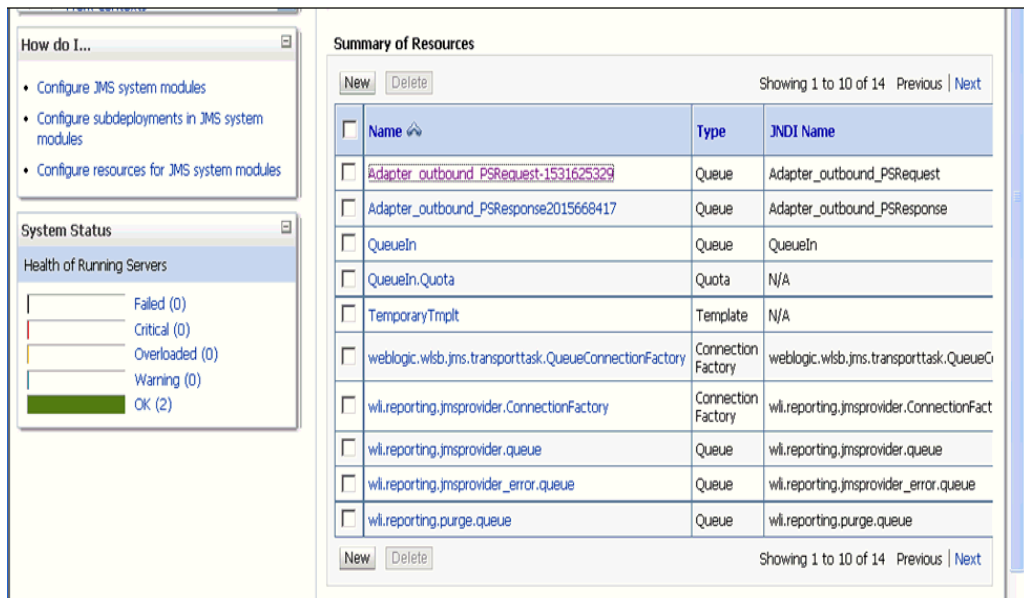
24. 図 9-135 に示すように、「ロックして編集」をクリックします。

図 9-135 構成設定



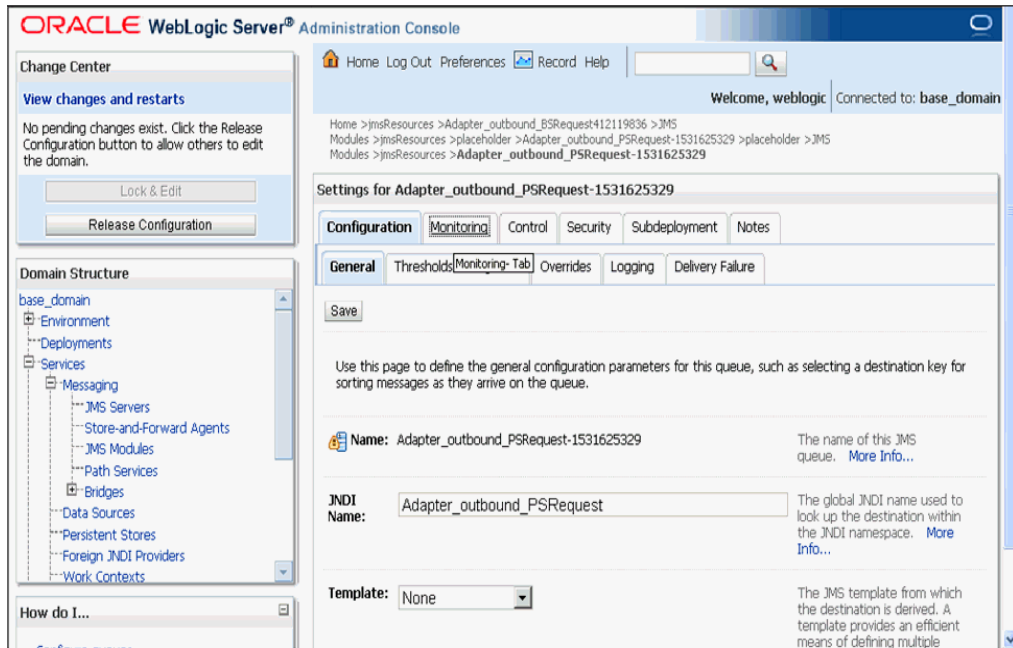
25. 図 9-136 に示すように、適切なリクエスト・リンク (**Adapter_outbound_PSRequest** など) をクリックします。

図 9-136 Adapter_outbound_PSRequest



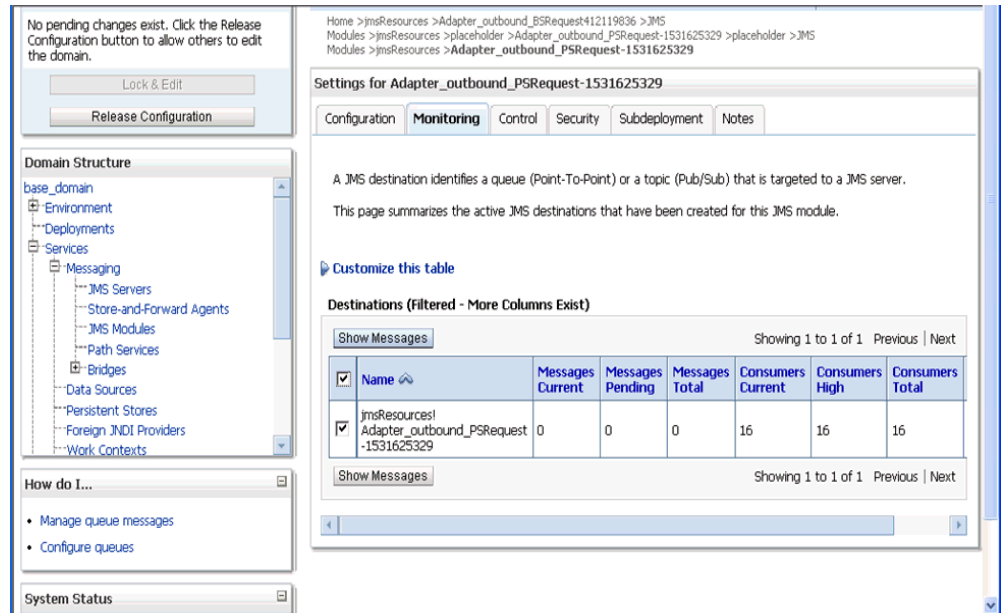
26. 図 9-137 に示すように、「モニタリング」タブをクリックします。

図 9-137 「モニタリング」タブ



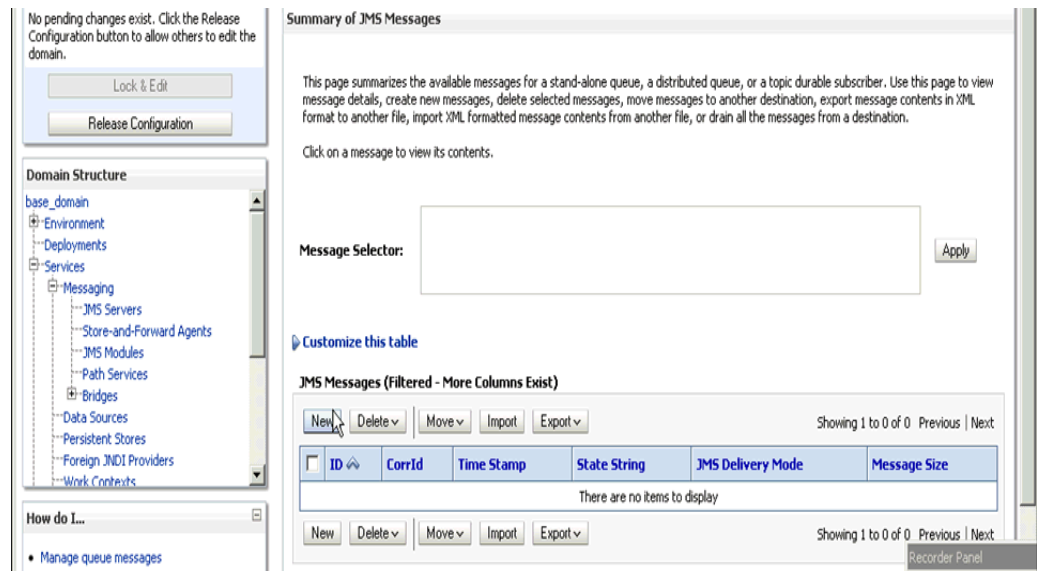
27. 図 9-138 に示すように、チェック・ボックスを有効にし、「メッセージの表示」をクリックします。

図 9-138 アダプタ設定



28. 図 9-139 に示すように、「次へ」をクリックします。

図 9-139 JMS メッセージ



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

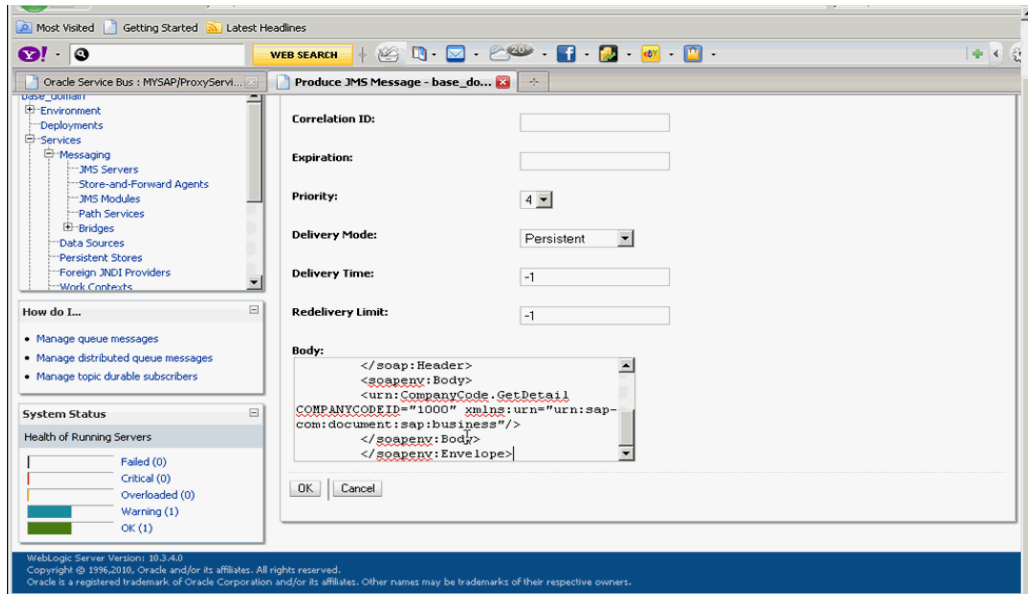
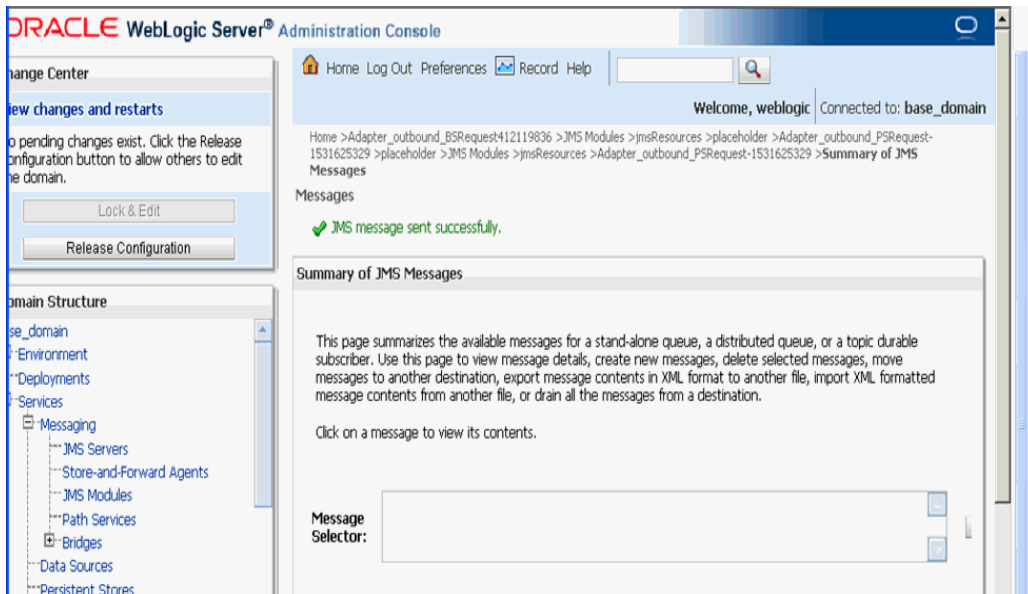


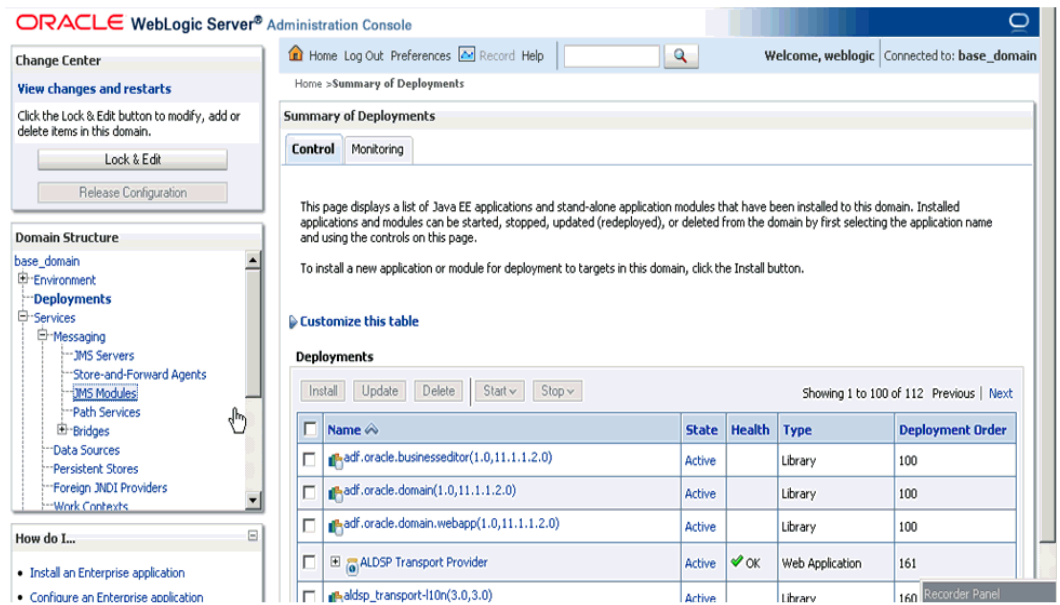
図 9-141 に示すように、成功を伝えるメッセージが表示されます。

図 9-141 JMS の成功を伝えるメッセージ



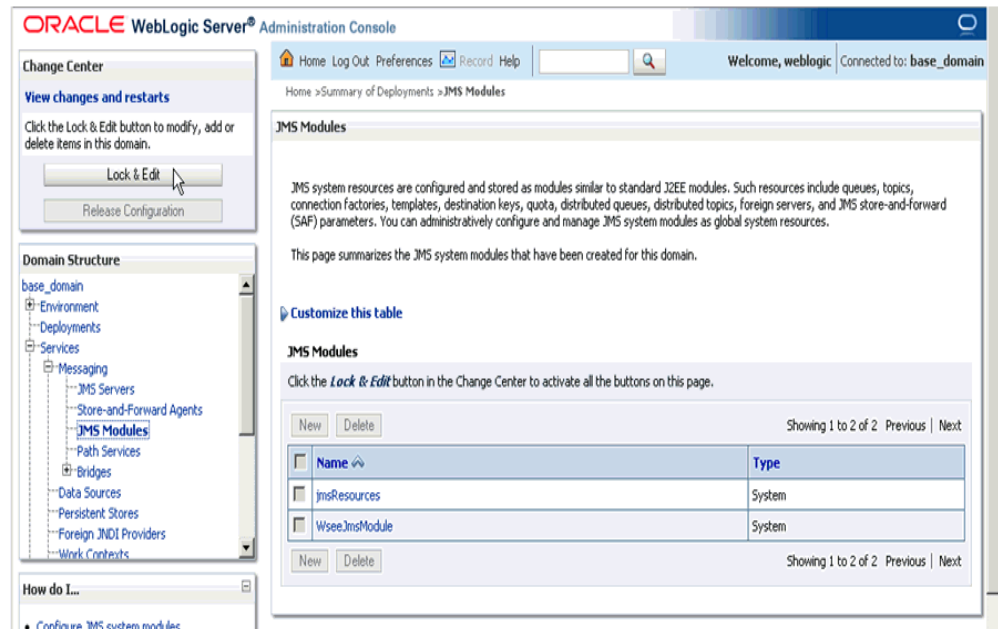
30. 図 9-142 に示すように、「Oracle WLS コンソール」で「サービス」を展開し、「メッセージング」を展開し、「JMS モジュール」をクリックします。

図 9-142 「JMS モジュール」



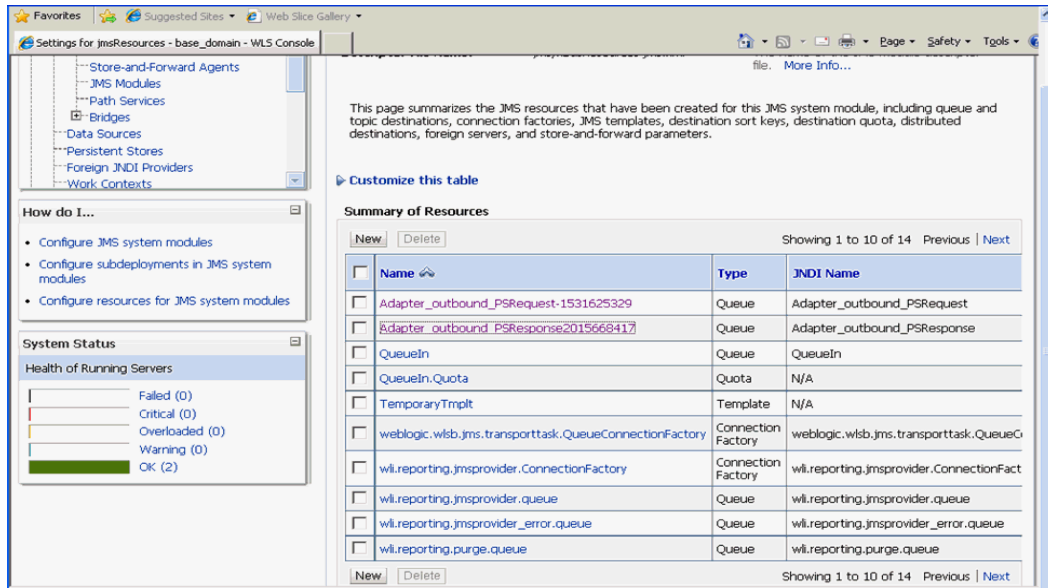
31. 図 9-143 に示すように、「jmsResources」をクリックします。

図 9-143 jmsResources



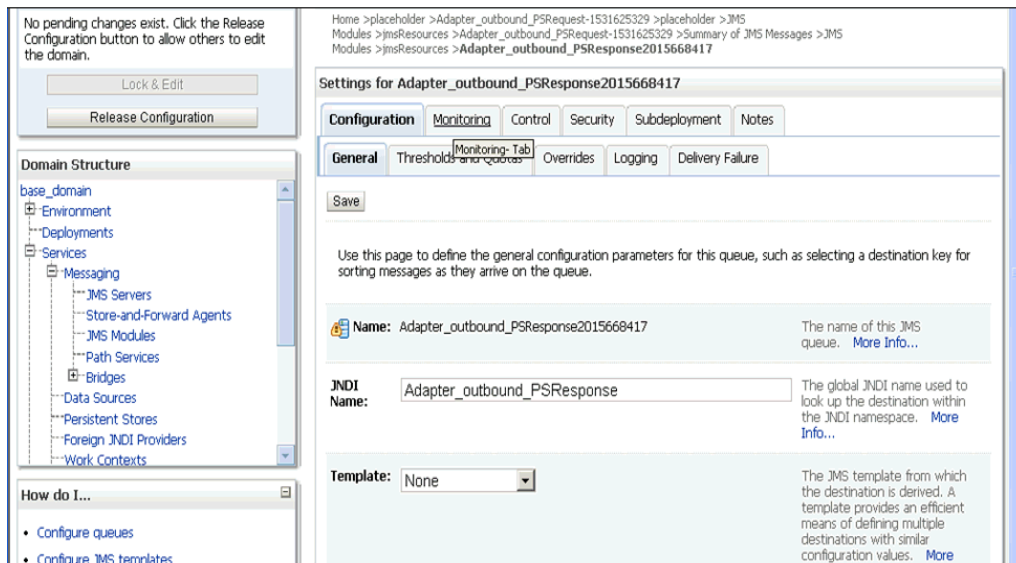
32. 図 9-144 に示すように、適切なレスポンス・リンク (Adapter_outbound_PSResponse など) をクリックします。

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、「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 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」リンクをクリックします。

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)

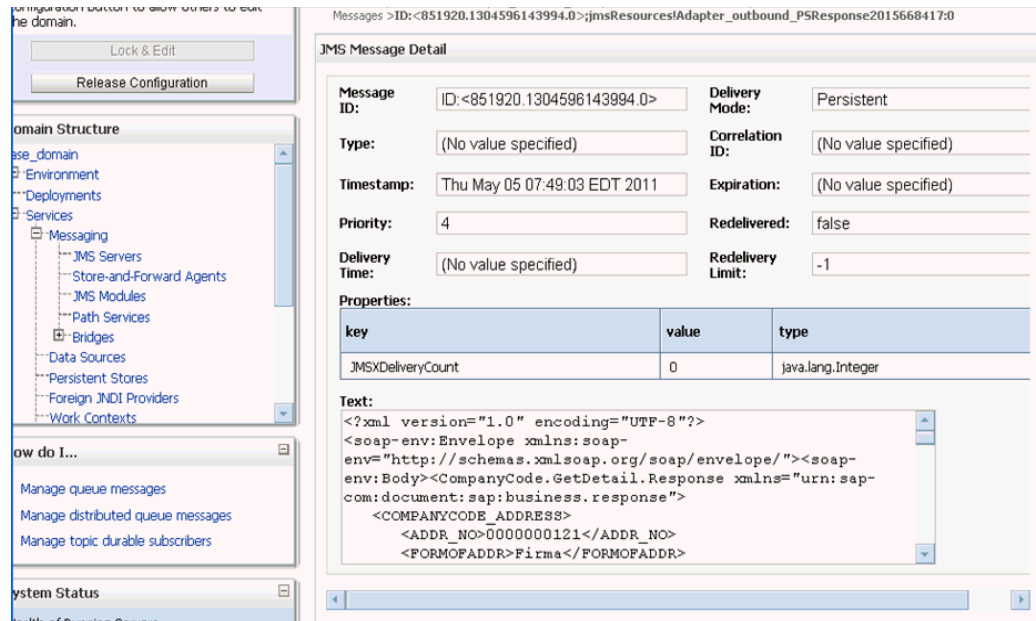
New Delete Move Import Export Showing 1 to 1 of 1 Previous Next

<input type="checkbox"/>	ID	Corrid	Time Stamp	State String	JMS Delivery Mode	Message Size
<input type="checkbox"/>	ID:<851920.1304596143994.0>		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

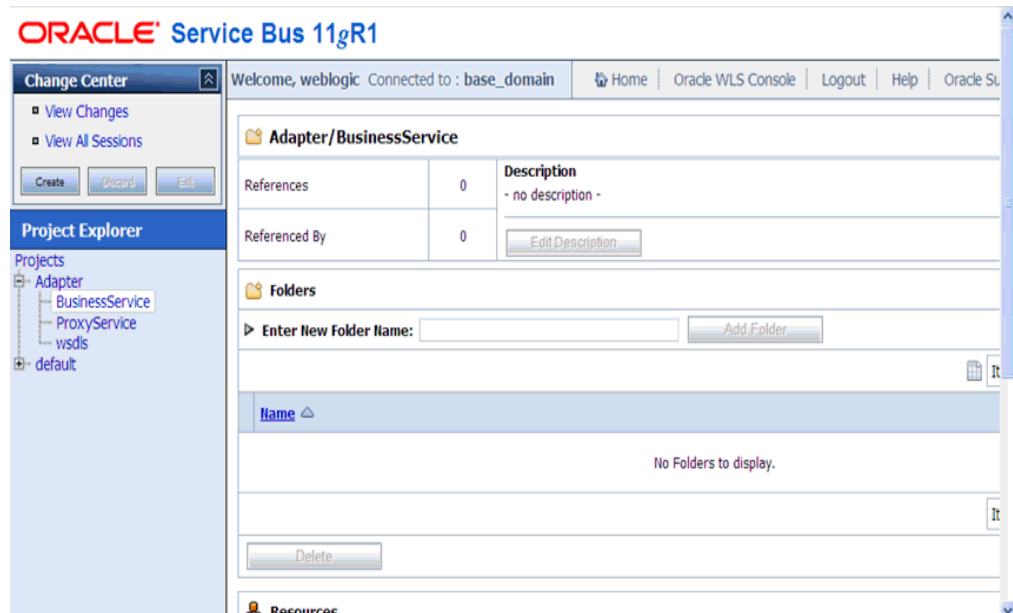


9.7 Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成)

この項では、J2CA 構成用に 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。

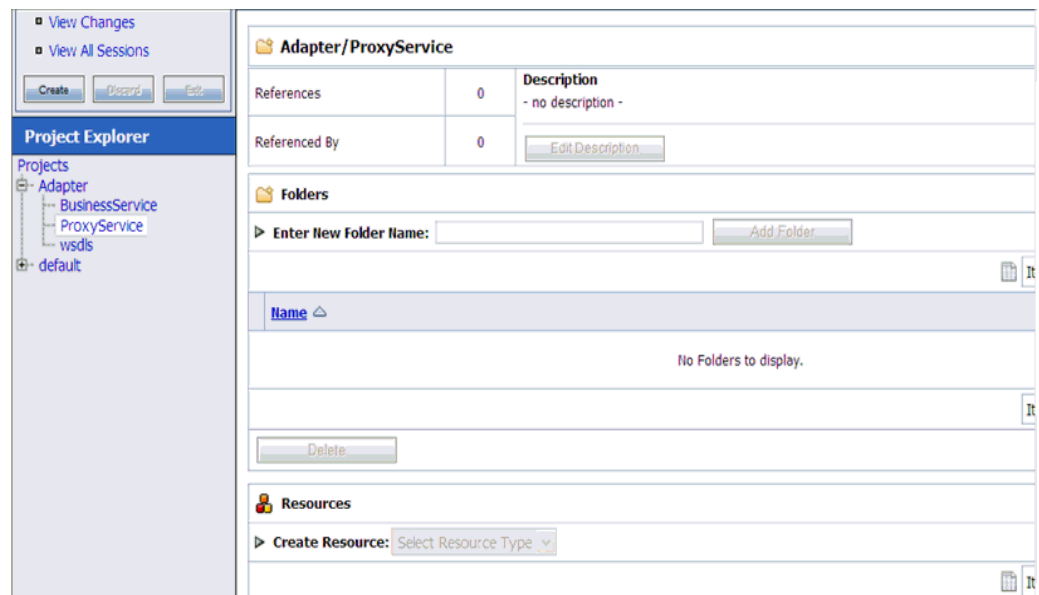
1. Oracle Service Bus を起動し、必要なプロジェクト・フォルダを作成します。
詳細については、[9.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」](#)を参照してください。
2. アプリケーション・エクスプローラから WSDL を生成して作成済のプロジェクト・フォルダに公開し、公開された WSDL を使用してビジネス・サービスを作成します。
詳細については、[9.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL の公開」](#)を参照してください。
3. [図 9-149](#) 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



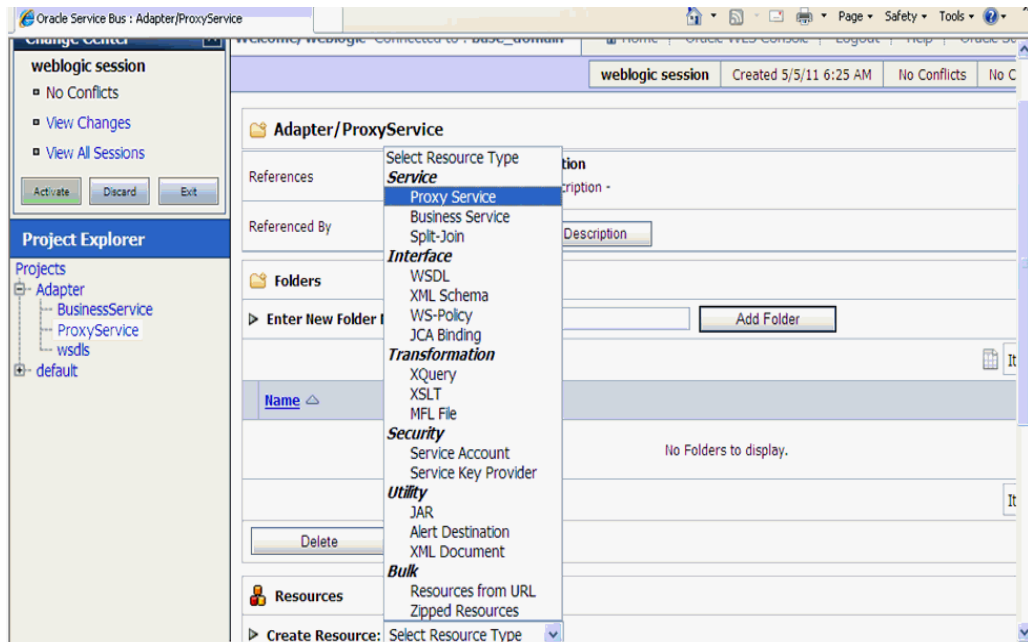
4. 図 9-150 に示すように、プロジェクト・エクスプローラで「ProxyService」プロジェクト・フォルダを選択し、「作成」をクリックします。

図 9-150 プロジェクト・エクスプローラ



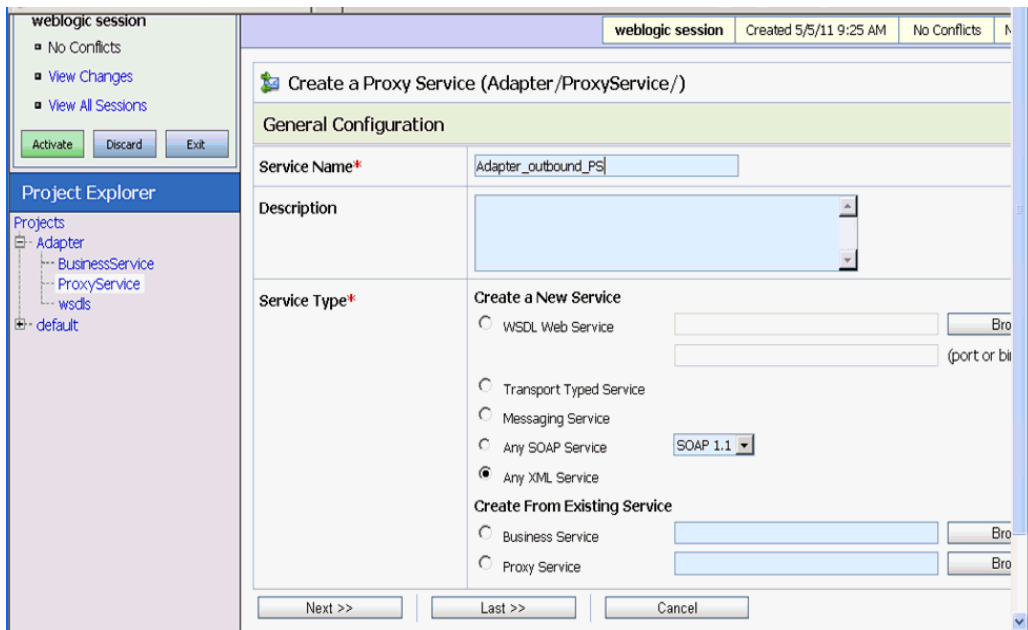
5. 図 9-151 に示すように、右ペインの「リソースの作成」リストで「プロキシ・サービス」を選択します。

図 9-151 プロキシ・サービス



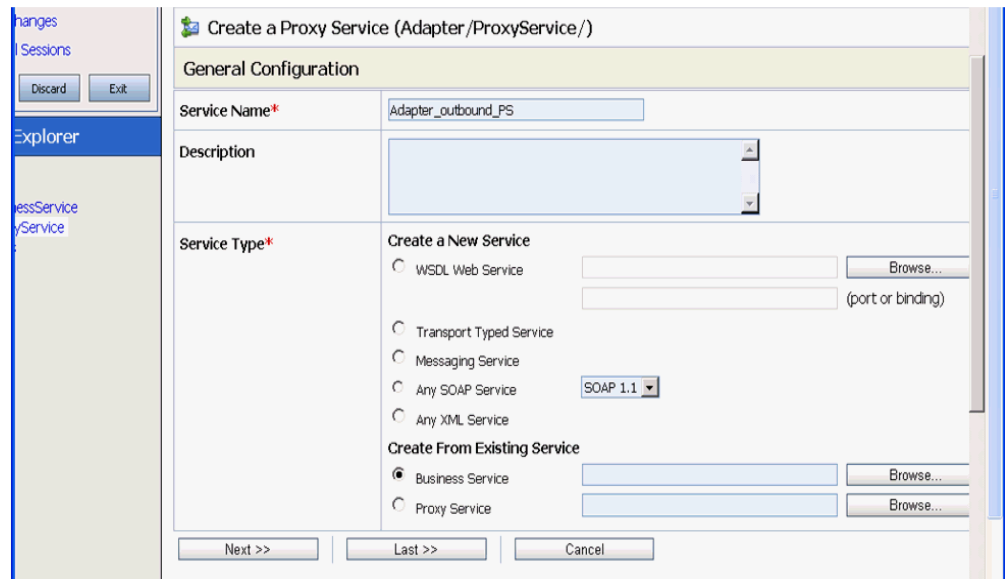
6. 図 9-152 に示すように、「サービス名」フィールドに適切な名前を入力します。

図 9-152 「サービス名」



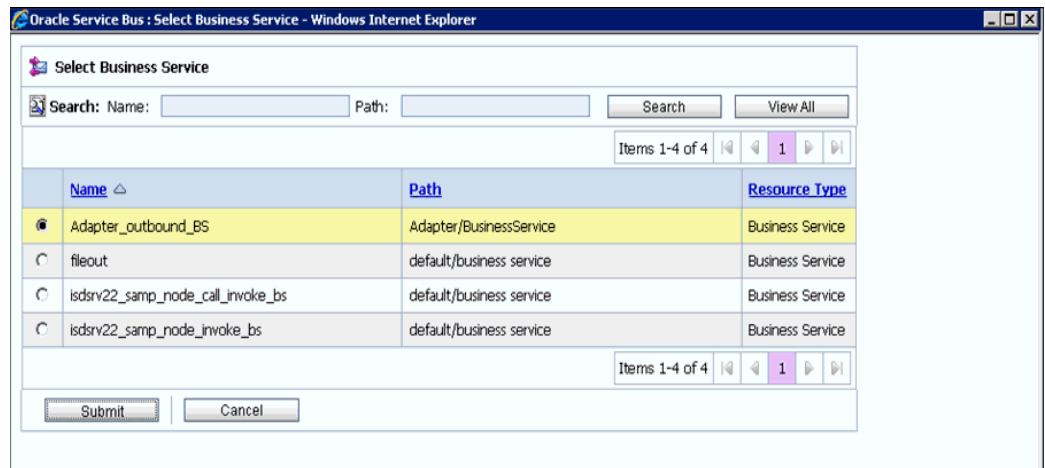
7. 図 9-153 に示すように、「サービス・タイプ」セクションの「既存のサービスから作成」で「ビジネス・サービス」ラジオ・ボタンを選択し、「参照」をクリックします。

図 9-153 「全般的な構成」



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



9. 図 9-155 に示すように、「次へ」をクリックします。

図 9-155 「全般的な構成」

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」リストで「http」を選択し、「次へ」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、「次へ」をクリックします。

図 9-157 HTTP トラnsポート構成

The screenshot shows the 'Create a Proxy Service (Adapter/ProxyService/Adapter_outbound_PS)' configuration wizard. The 'Project Explorer' on the left shows a tree structure with 'Adapter' expanded to show 'BusinessService', 'ProxyService', and 'wsdls'. The main configuration area is titled 'HTTP Transport Configuration' and includes the following settings:

- HTTPS required:**
- Authentication:**
 - None
 - Basic
 - Client Certificate
 - Custom Authentication (See Advanced Settings)
- Dispatch Policy:** default (dropdown menu)
- Request Encoding:** (text input field)
- Response Encoding:** (text input field)
- Advanced Settings:** (section header)

Navigation buttons at the bottom include '<< Prev.', 'Next >>', 'Last >>', and 'Cancel'.

12. 図 9-158 に示すように、「次へ」をクリックします。

図 9-158 「操作選択構成」

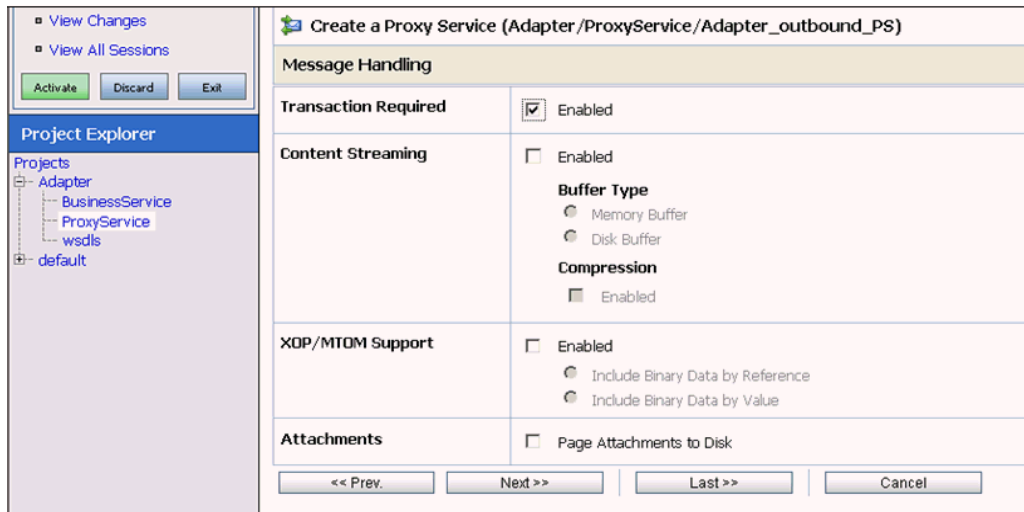
The screenshot shows the 'Create a Proxy Service (Adapter/ProxyService/Adapter_outbound_PS)' configuration wizard. The 'Project Explorer' on the left is the same as in the previous screenshot. The main configuration area is titled 'Operation Selection Configuration' and includes the following settings:

- Enforce WS-I Compliance:**
- Selection Algorithm:**
 - Transport Header
 - SOAPAction Header
 - WS-Addressing
 - SOAP Header
 - SOAP Body Type

Navigation buttons at the bottom include '<< Prev.', 'Next >>', 'Last >>', and 'Cancel'.

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、「次へ」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」

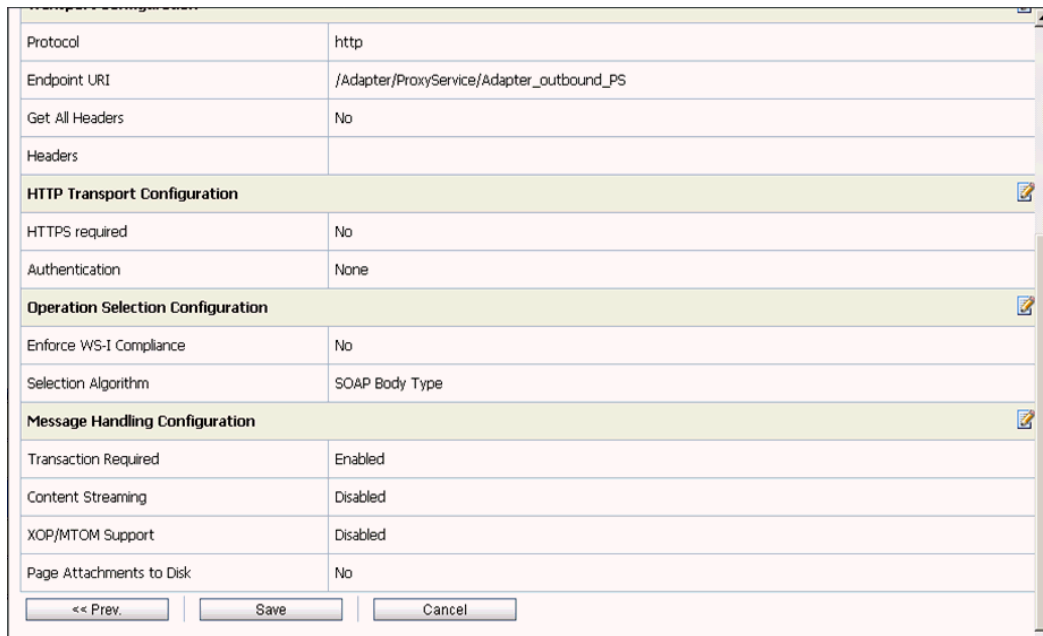
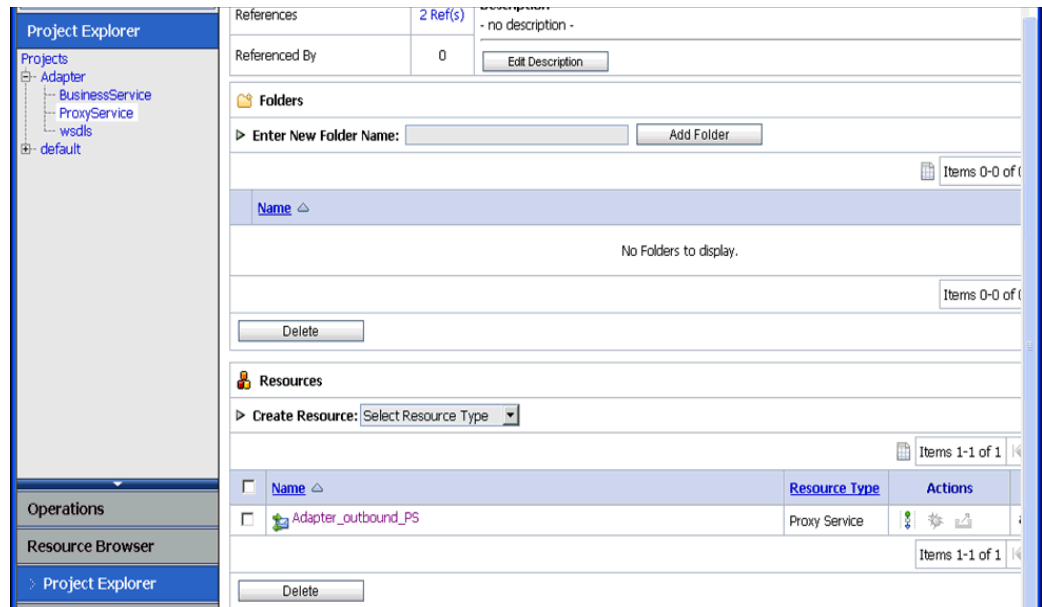


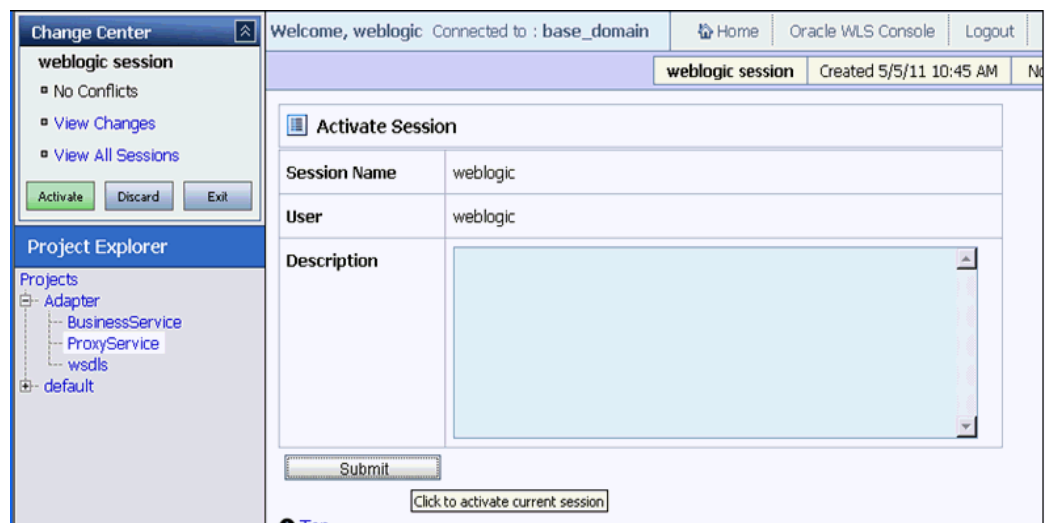
図 9-161 に示すように、作成されたプロキシ・サービスが保存されます。

図 9-161 プロキシ・サービス



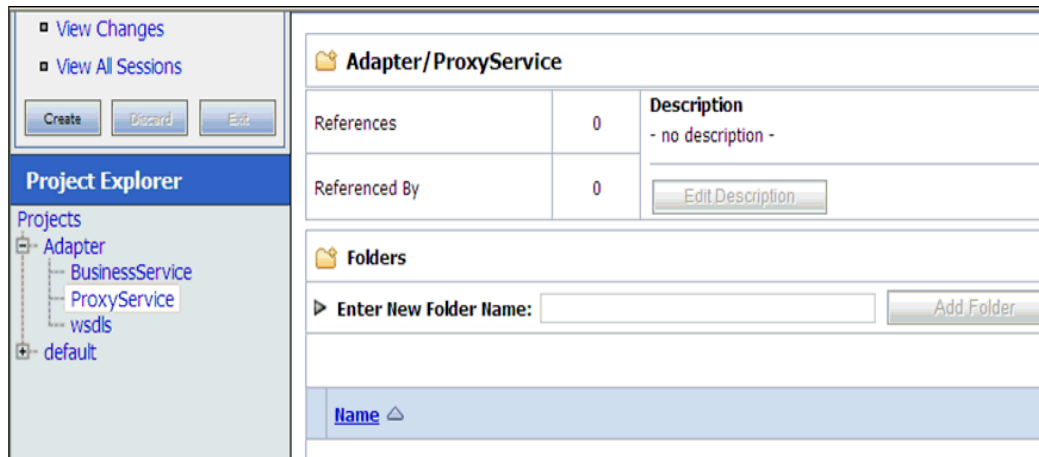
15. 図 9-162 に示すように、左ペインで「アクティブ化」をクリックし、右ペインで「発行」をクリックします。

図 9-162 「セッションのアクティブ化」



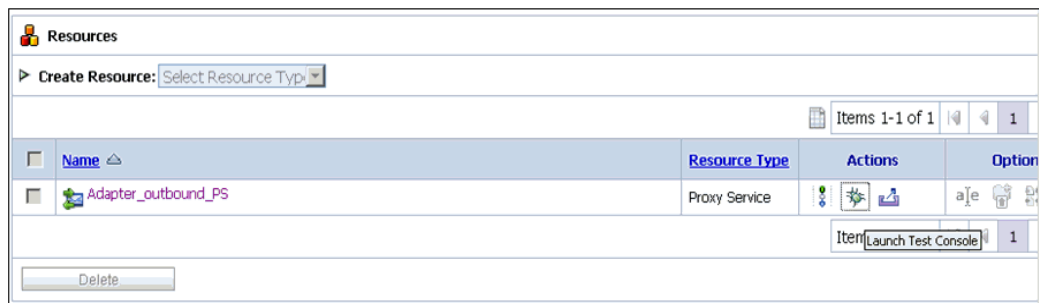
16. 図 9-163 に示すように、左ペインで「Projects」フォルダにある「ProxyService」をクリックします。

図 9-163 ProxyService



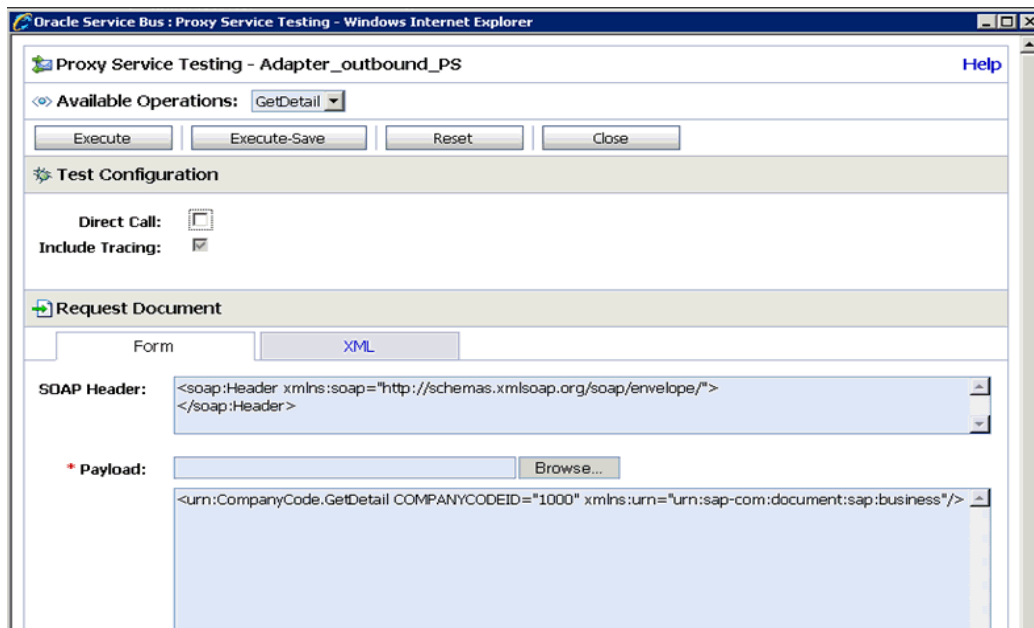
17. 図 9-164 に示すように、作成済のプロキシ・サービスの「テスト・コンソールの起動」アイコンをクリックします。

図 9-164 「テスト・コンソールの起動」



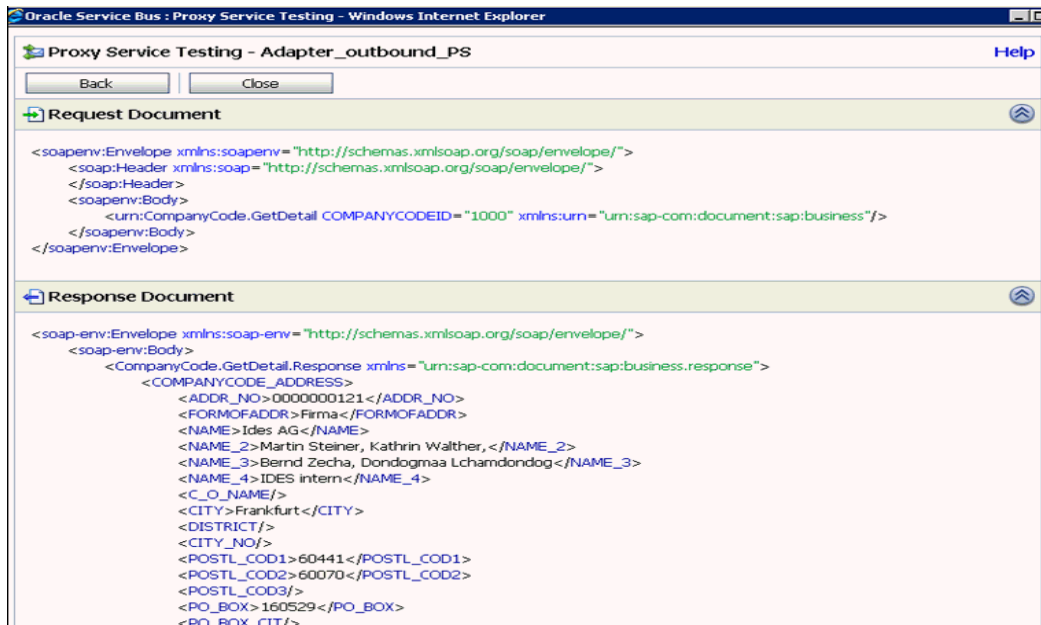
18. 図 9-165 に示すように、「直接呼出し」チェック・ボックスを選択解除し、「ペイロード」に入力値を指定し、「実行」をクリックします。

図 9-165 プロキシ・サービスのテスト



19. 図 9-166 に示すように、レスポンス・ドキュメントをレビューします。

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、Web サービスの統合を可能にします。

この章の内容は以下のとおりです。

- 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。

Application Adapters インストールの次のフォルダに、このアウトバウンドのユースケース・シナリオのサンプル・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
```

この節では、次のトピック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

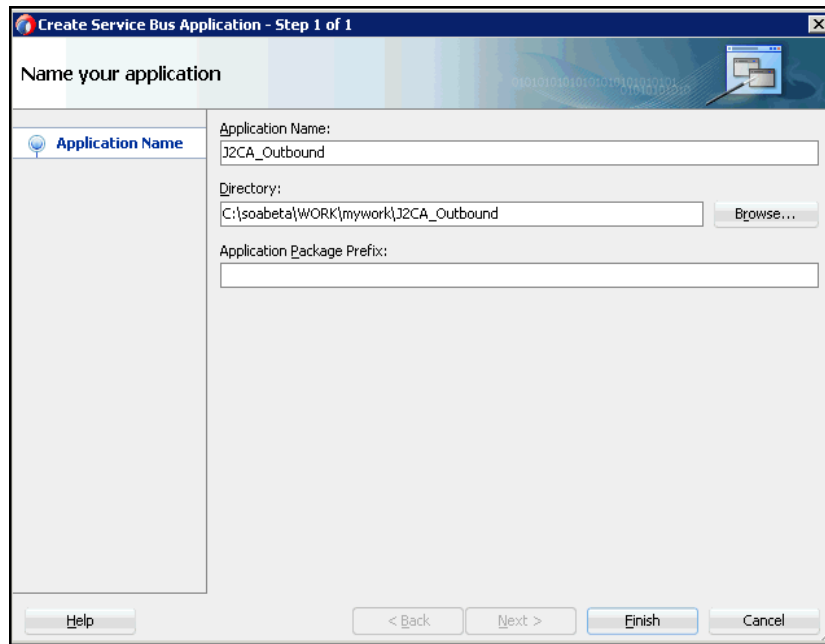
OSB アウトバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して、各 WSDL ファイルを生成する必要があります。詳細については、4-18 ページの 4.8 項「WSDL の生成 (J2CA 構成のみ)」を参照してください。

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 で次の手順を実行します。

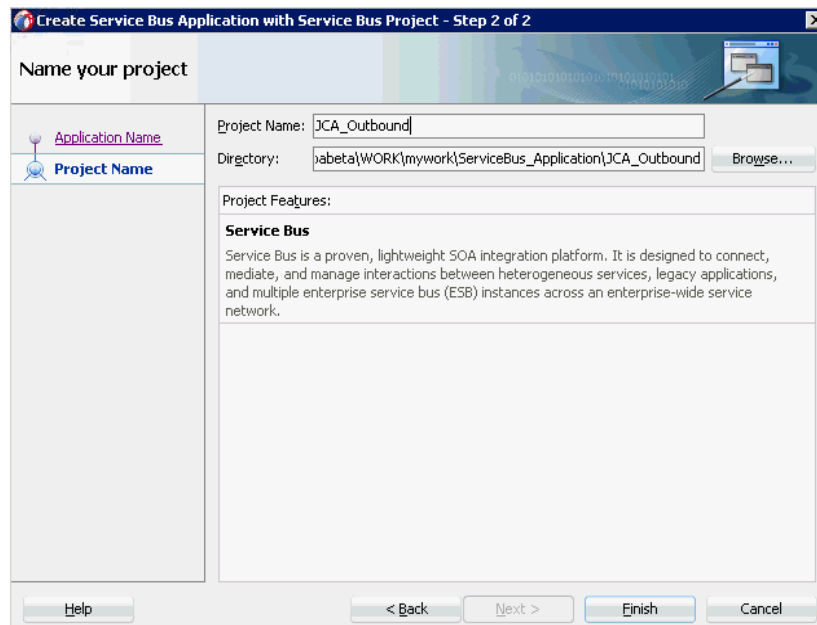
1. 新しい OSB アプリケーションを作成します。
2. 図 10-1 に示すように、OSB アプリケーションの名前 (J2CA_Outbound など) を入力し、「終了」をクリックします。

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。次のトピック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. 図 10-3 に示すように、「Service Bus」コンポーネント・ペインから外部サービス・ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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

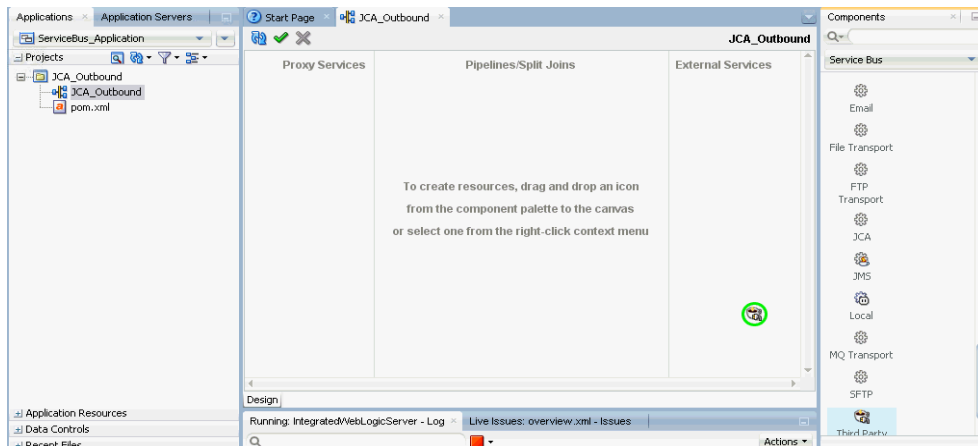
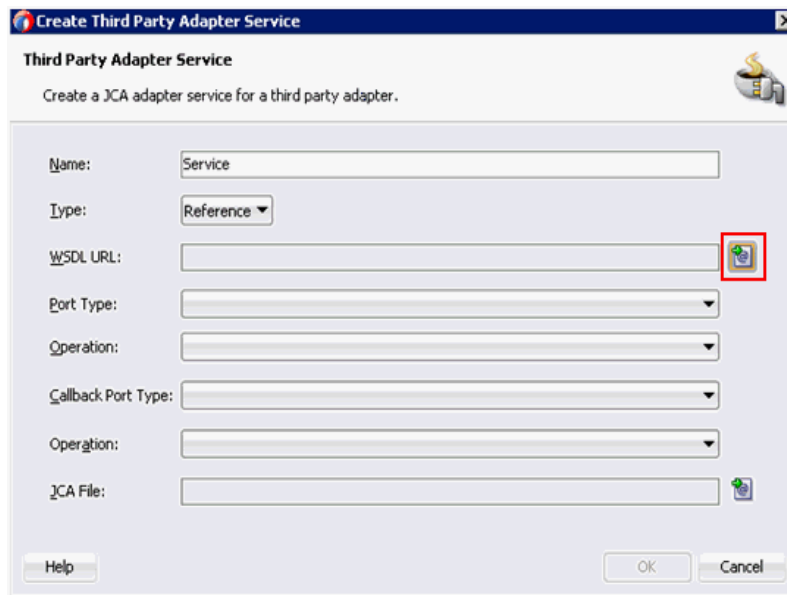


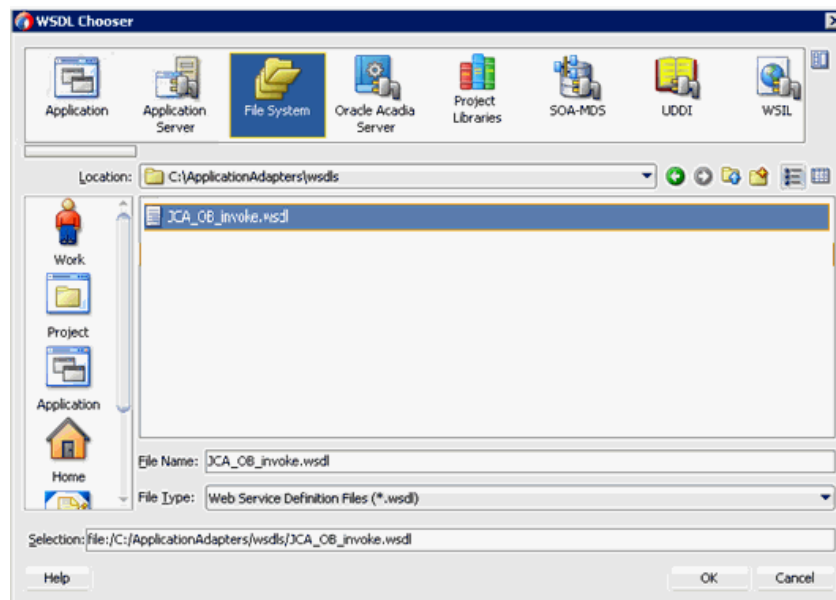
図 10-4 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

図 10-4 「サード・パーティ・アダプタ・サービスの作成」ペイン



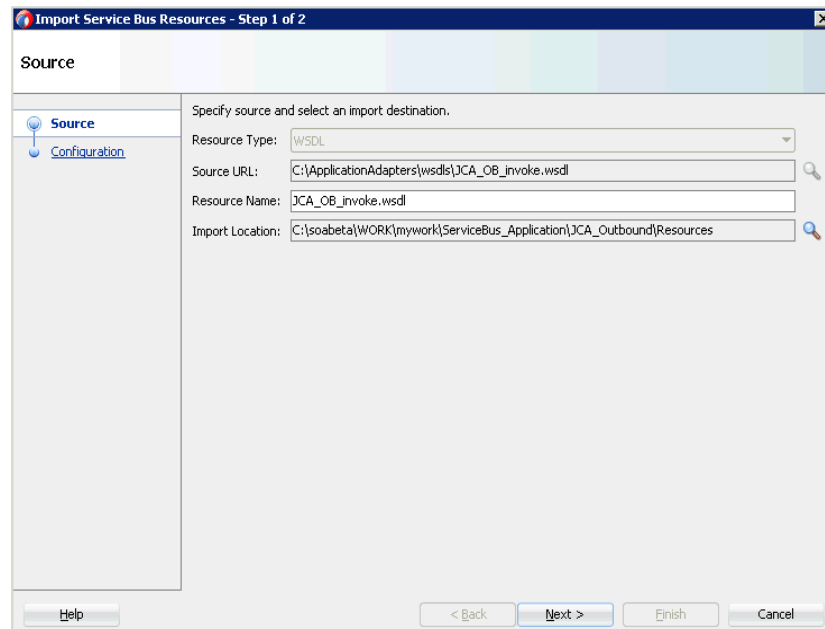
2. ビジネス・サービス名として使用するサード・パーティ・アダプタ・サービスの適切な名前を入力します。
3. 「タイプ」ドロップダウン・リストで「参照」が選択されていることを確認します (デフォルト)。
4. 「WSDL URL」フィールドの右の既存の WSDL の検索アイコンをクリックします。
 図 10-5 に示すように、「WSDL チューザ」ダイアログが表示されます。

図 10-5 「WSDL チューザ」 ダイアログ



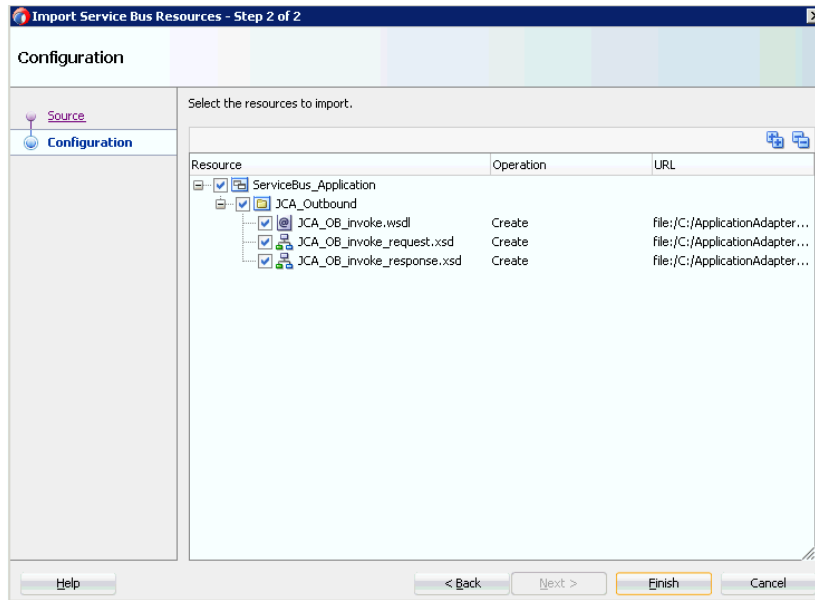
5. 「ファイル・システム」タブを選択し、WSDL ディレクトリでアウトバウンド WSDL ファイルを参照して選択します。
6. 「OK」をクリックします。
サービス・バス・リソースのインポート・ダイアログが表示されます。
7. 図 10-6 に示すように、「次へ」をクリックします。

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。

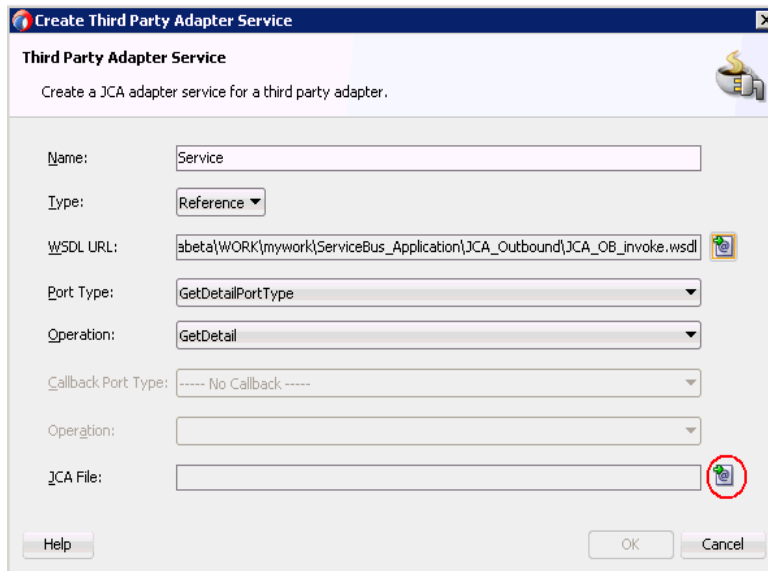
図 10-7 「構成」 ペイン



「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。

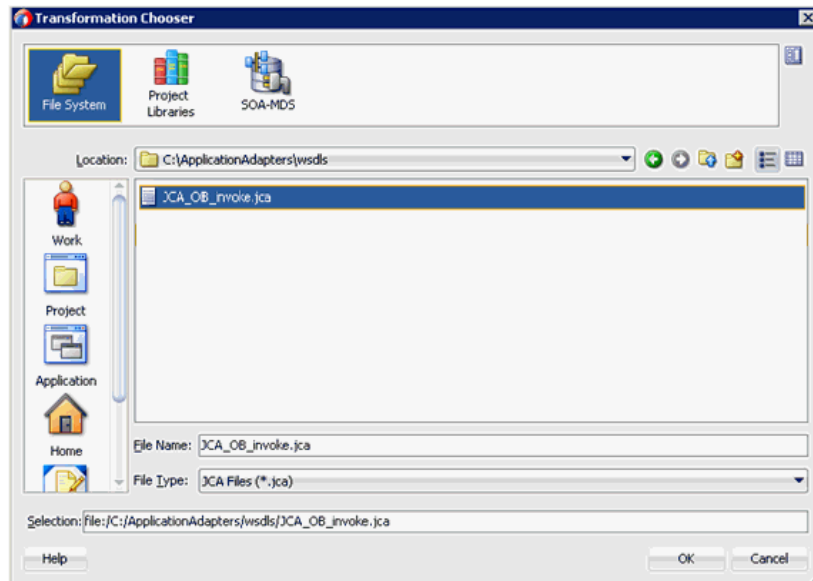
9. 10-8 に示すように、「JCA ファイル」フィールドの右の JCA ファイルの検索アイコンをクリックします。

図 10-8 JCA ファイルの検索アイコン



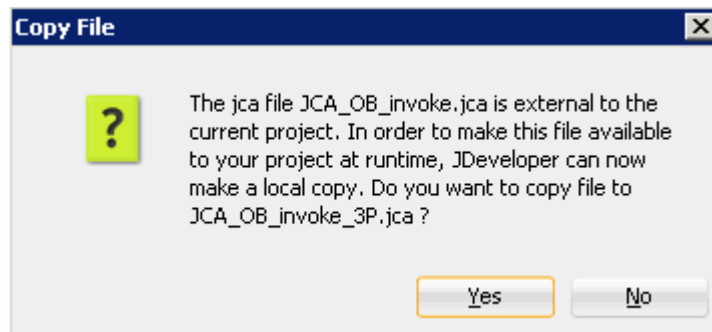
10-9 に示すように、トランスフォーメーション・チューザ・ダイアログが表示されます。

図 10-9 トランスフォーメーション・チューザ・ダイアログ



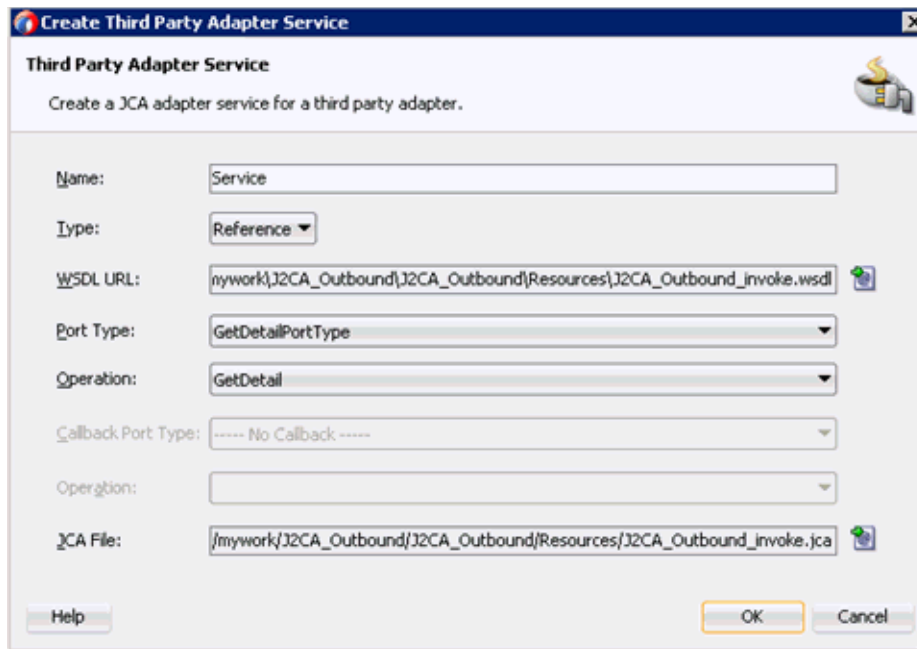
10. WSDL ディレクトリから JCA プロパティ・ファイルを選択します。
11. 「OK」をクリックします。図 10-10 に示すように、「ファイルのコピー」メッセージが表示されます。

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

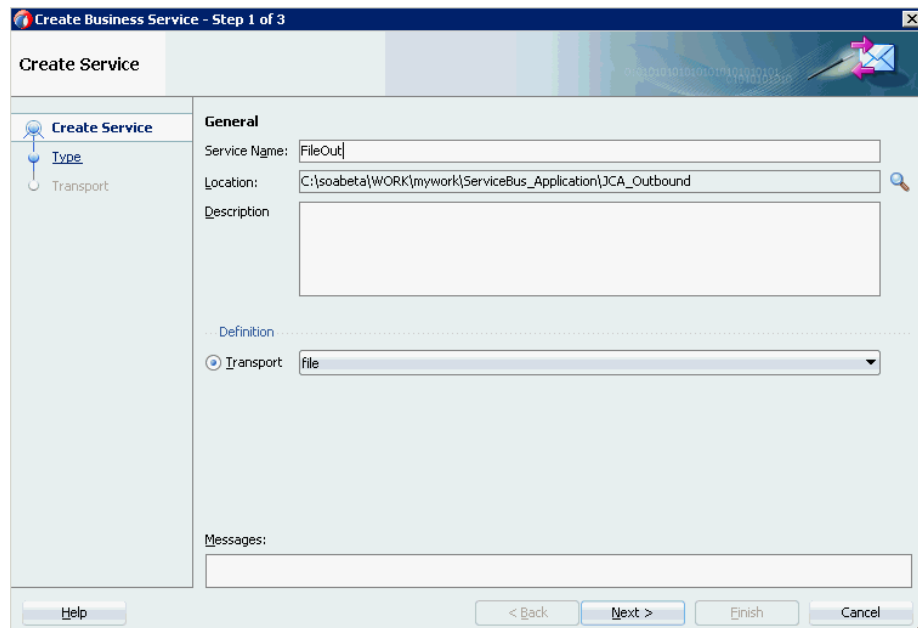
外部サービス・ペインにビジネス・サービス・コンポーネントが作成されます。

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、「次へ」をクリックします。

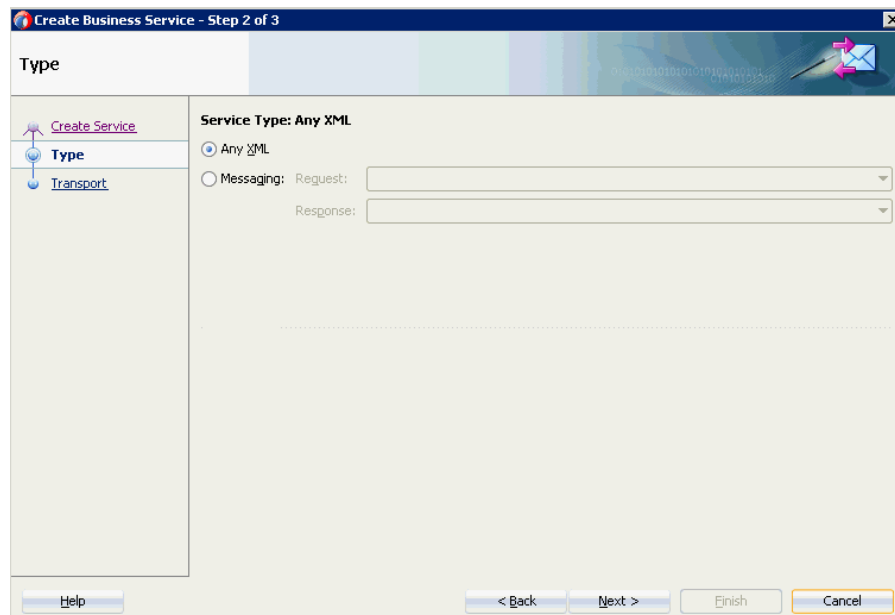
図 10-12 「サービスの作成」 ペイン



「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、「次へ」をクリックします。

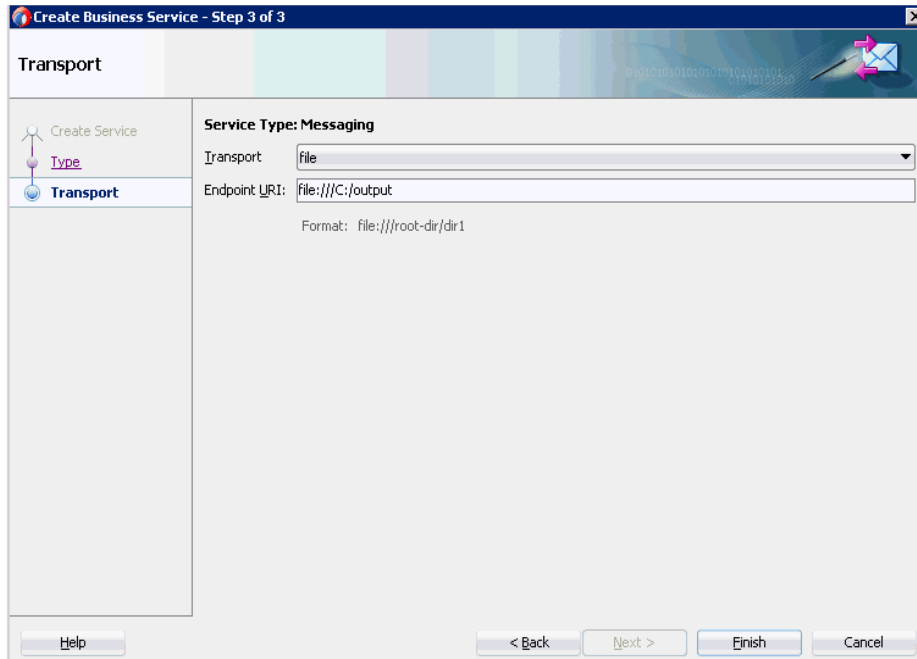
図 10-13 「タイプ」 ペイン



「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。

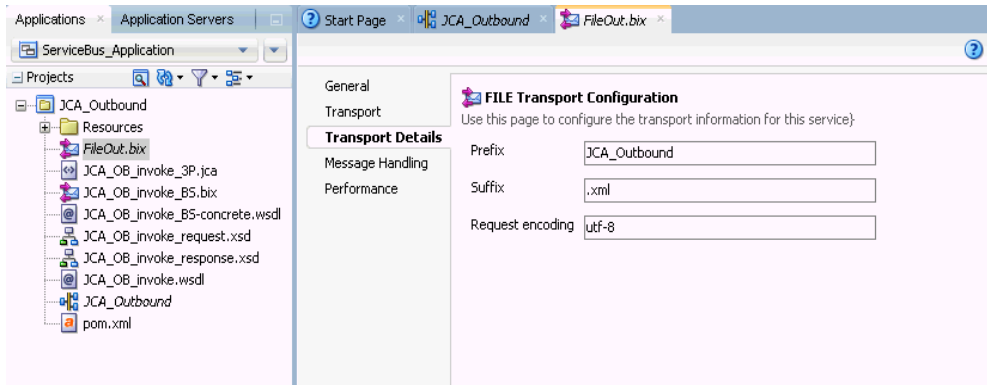
図 10-14 「トランスポート」 ペイン



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」

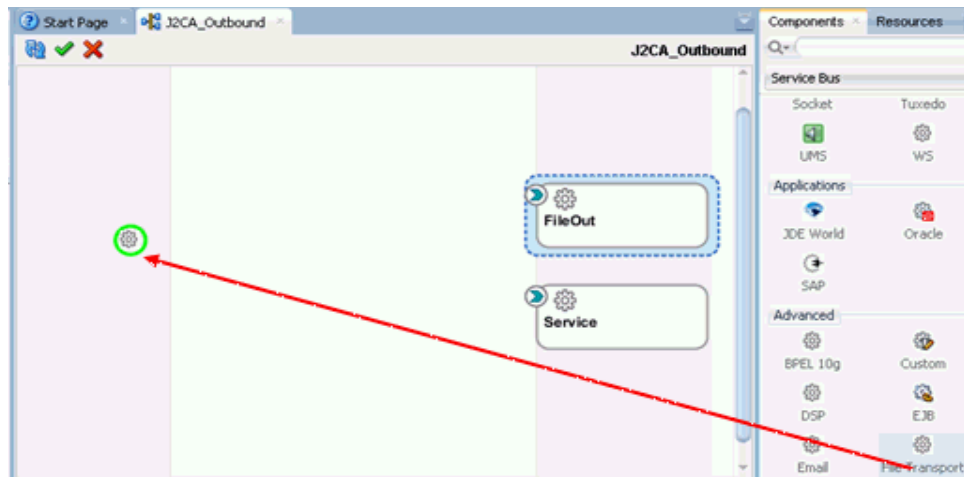


10.1.2.3 パイプラインを含むプロキシ・サービスの作成

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。

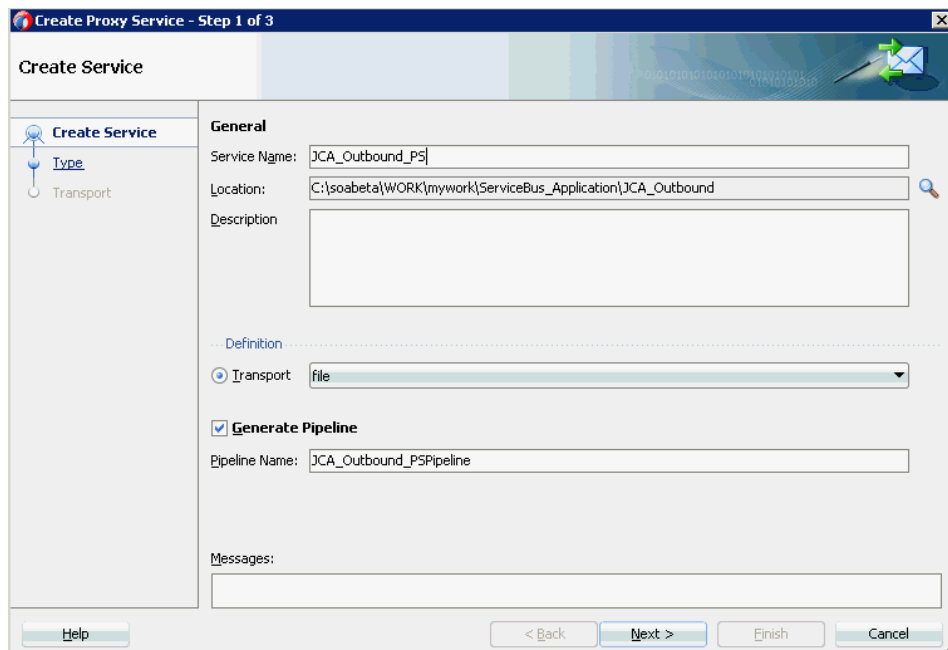
図 10-16 ファイル・トランスポート・コンポーネント



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、「次へ」をクリックします。

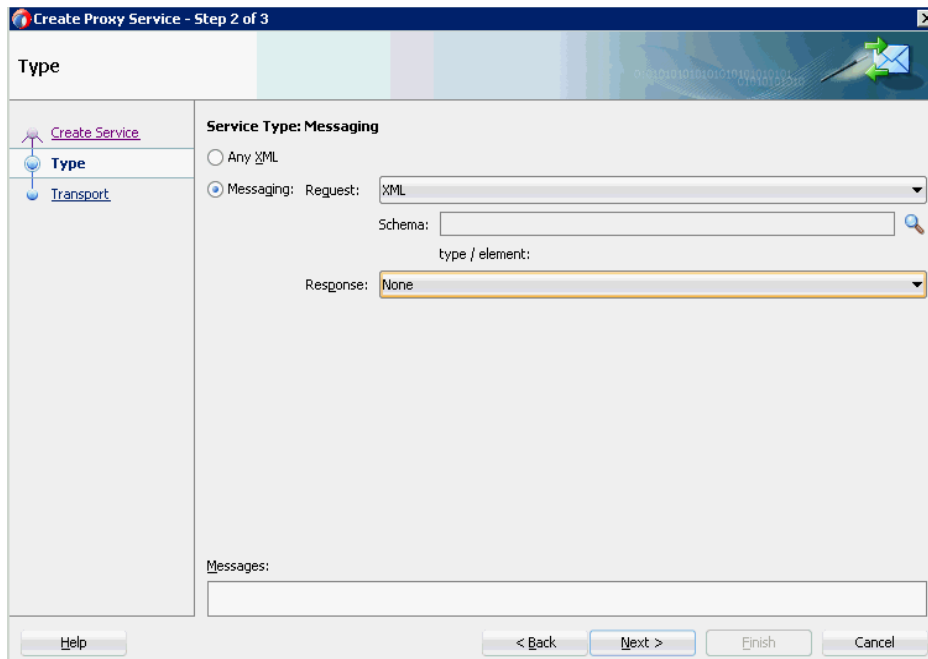
図 10-17 「サービスの作成」ペイン



「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、「次へ」をクリックします。

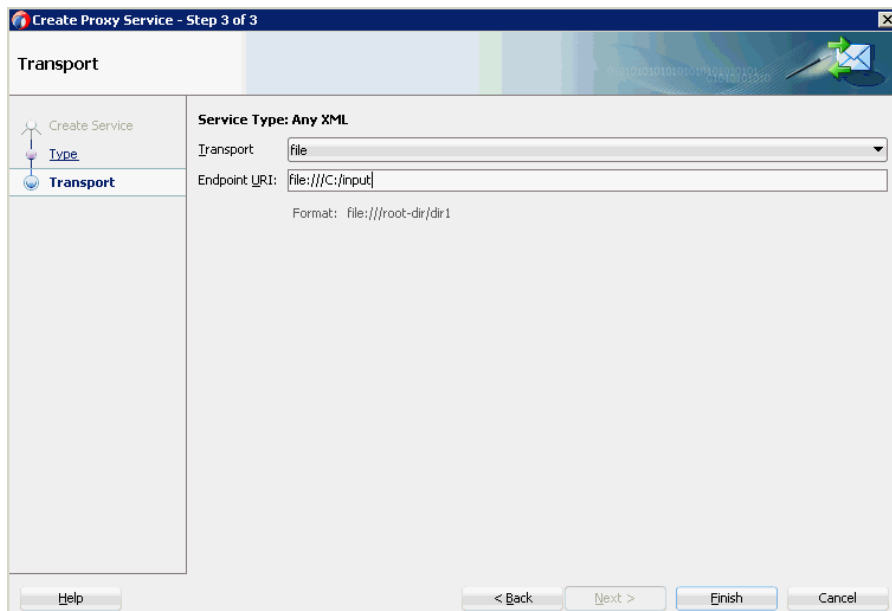
図 10-18 「タイプ」 ペイン



「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。

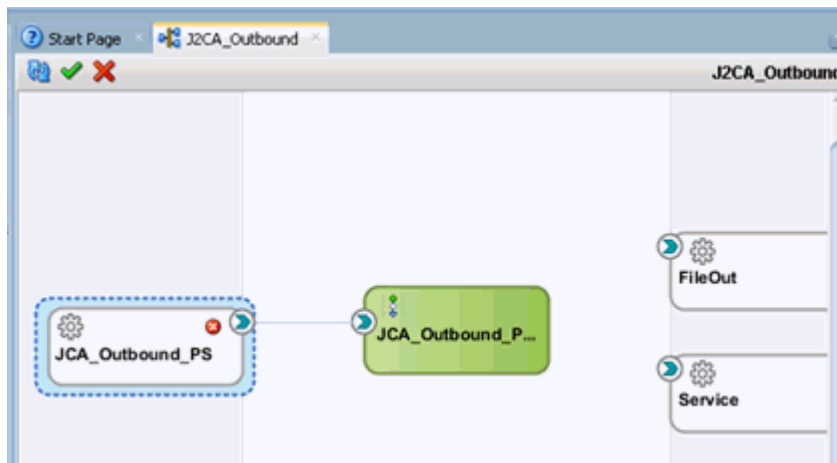
図 10-19 「トランスポート」 ウィンドウ



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。

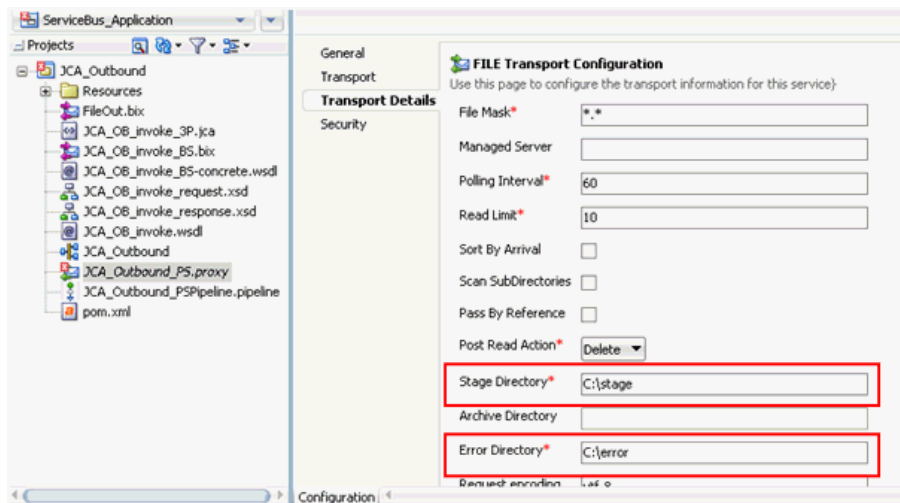
6. 図 10-20 に示すように、作成されたプロキシ・サービス (JCA_Outbound_PS など) をダブルクリックします。

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」および「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



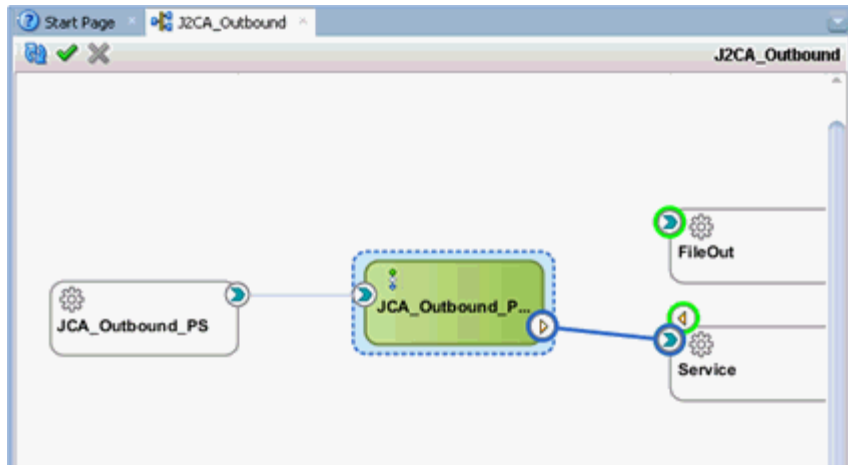
8. プロキシ・サービスの構成ページを保存して閉じます。

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、次のステップを実行します。

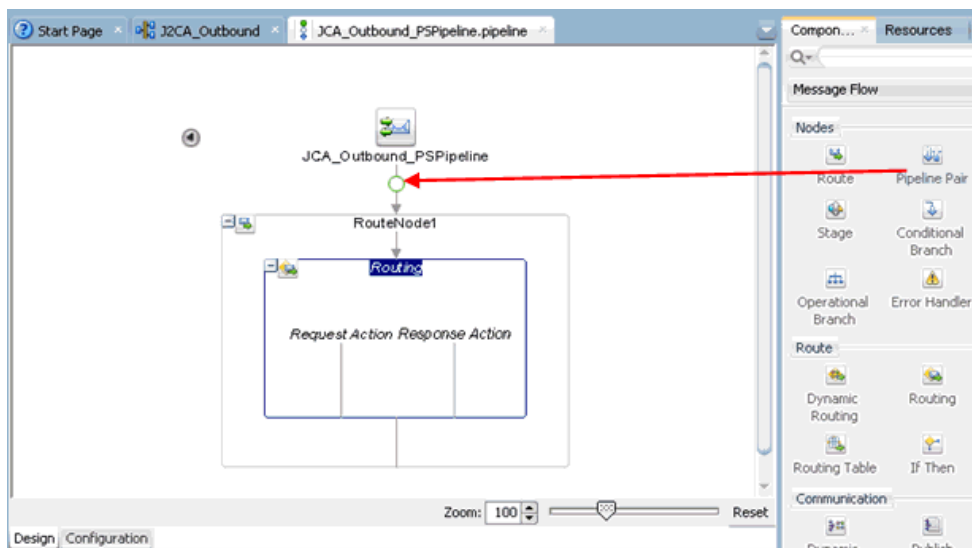
1. 図 10-22 に示すように、パイプラインをビジネス・サービス (Service など) に接続します。

図 10-22 ビジネス・サービスのパイプライン



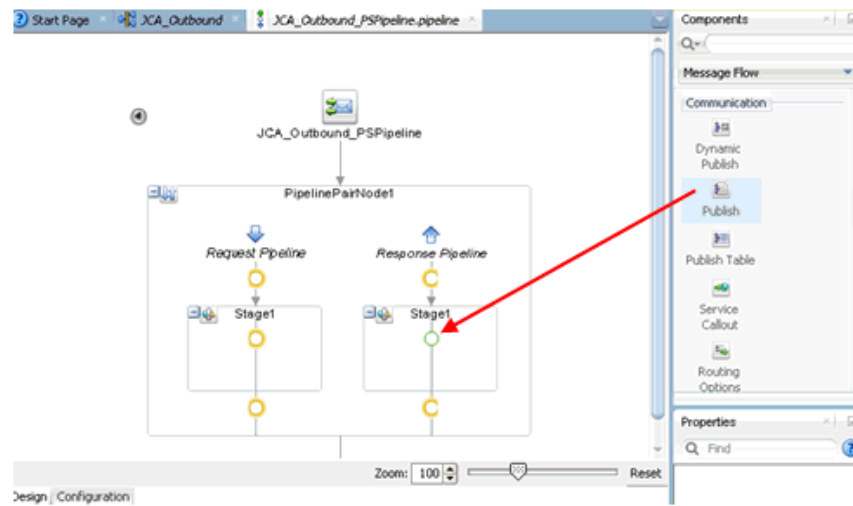
2. パイプライン/分割結合ペインで、パイプライン (JCA_Outbound_PSPipeline など) をダブルクリックします。
パイプライン構成ページが表示されます。
3. 図 10-23 に示すように、パイプライン・ペア・ノードを「ノード」ペインからパイプライン (JCA_Outbound_PSPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 10-23 パイプライン・ペア・ノード



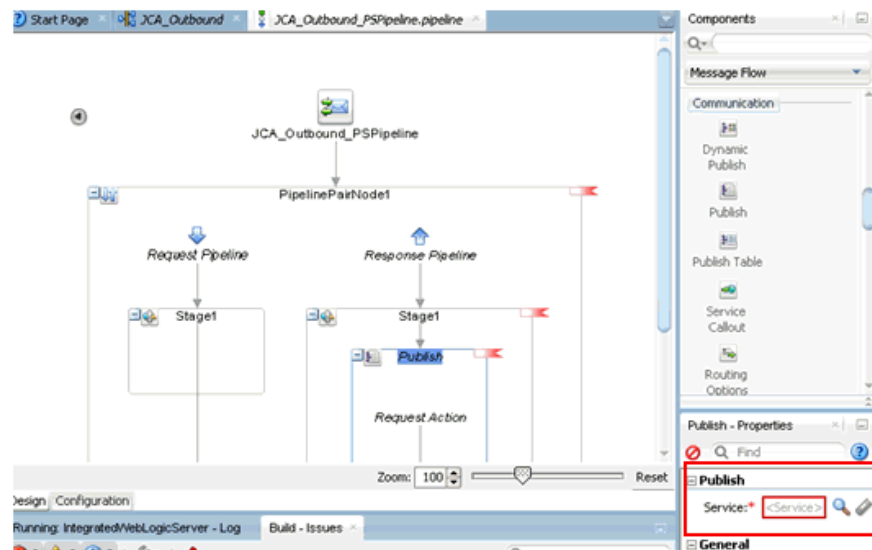
4. 図 10-24 に示すように、「パブリッシュ」ノードを「通信」ペインから「レスポンス・パイプライン」の「ステージ 1」の下の領域にドラッグ・アンド・ドロップします。

図 10-24 「パブリッシュ」ノード



5. 図 10-25 に示すように、プロパティのパブリッシュの右ペインで、「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

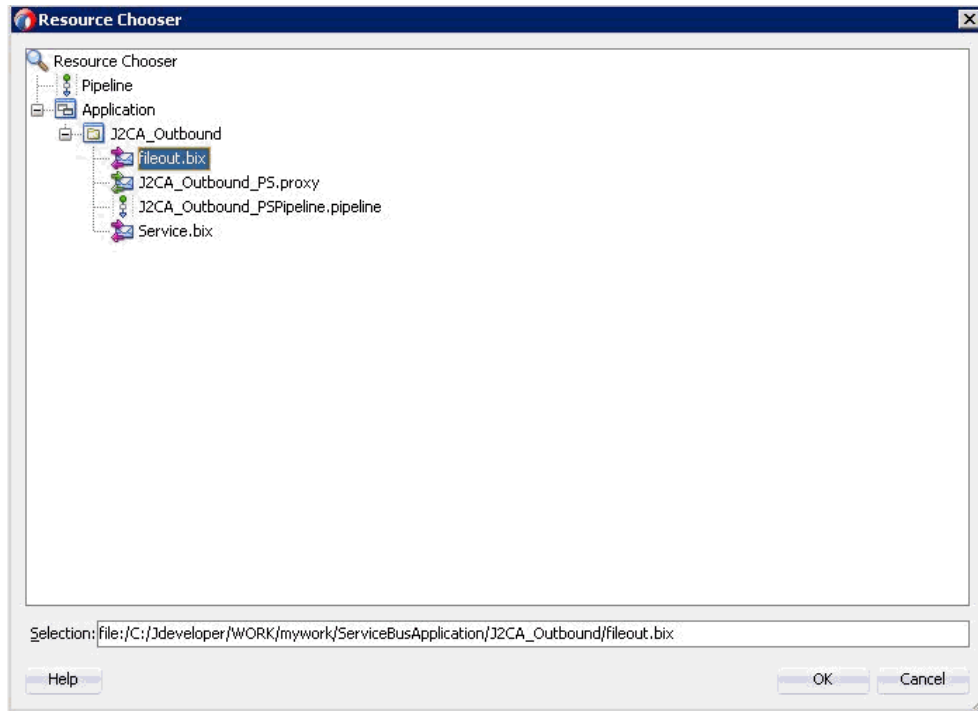
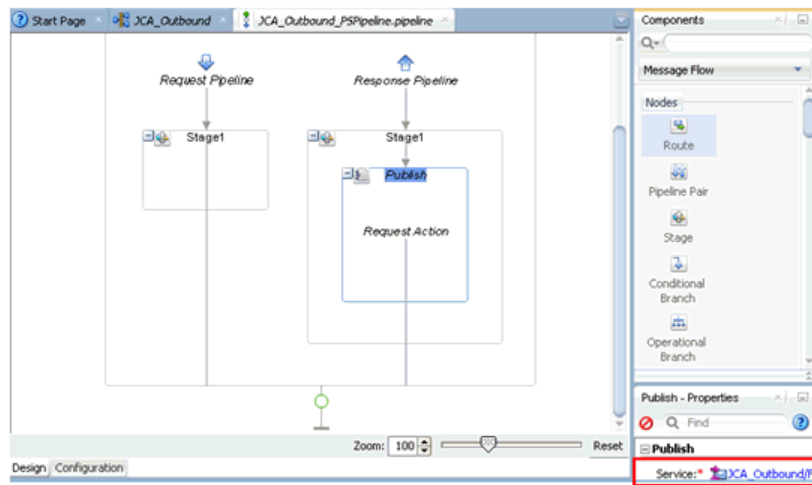


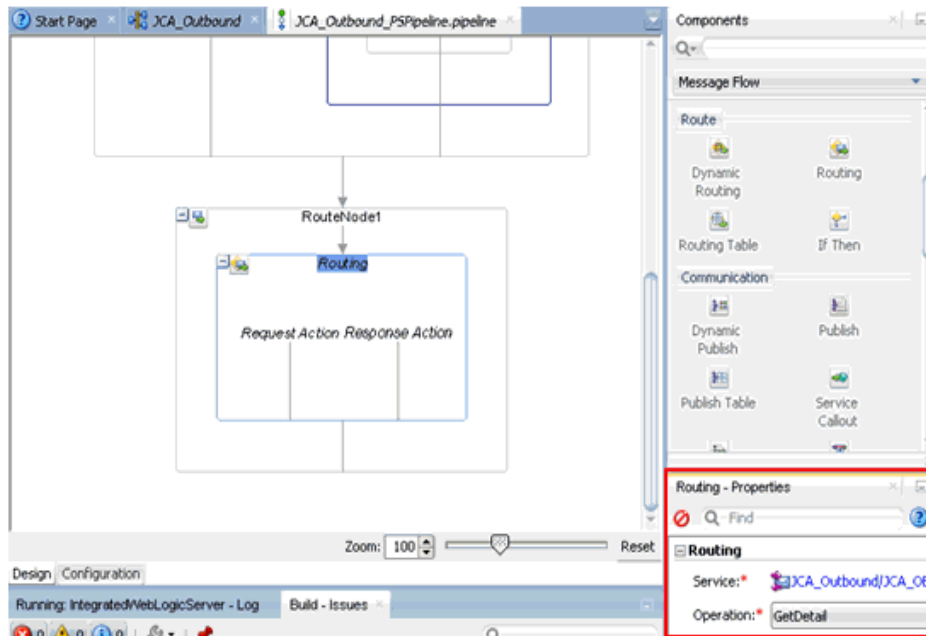
図 10-27 に示すように、選択したサービスが右ペインの「パブリッシュ」ペインに構成されます。

図 10-27 「パブリッシュ」ペイン



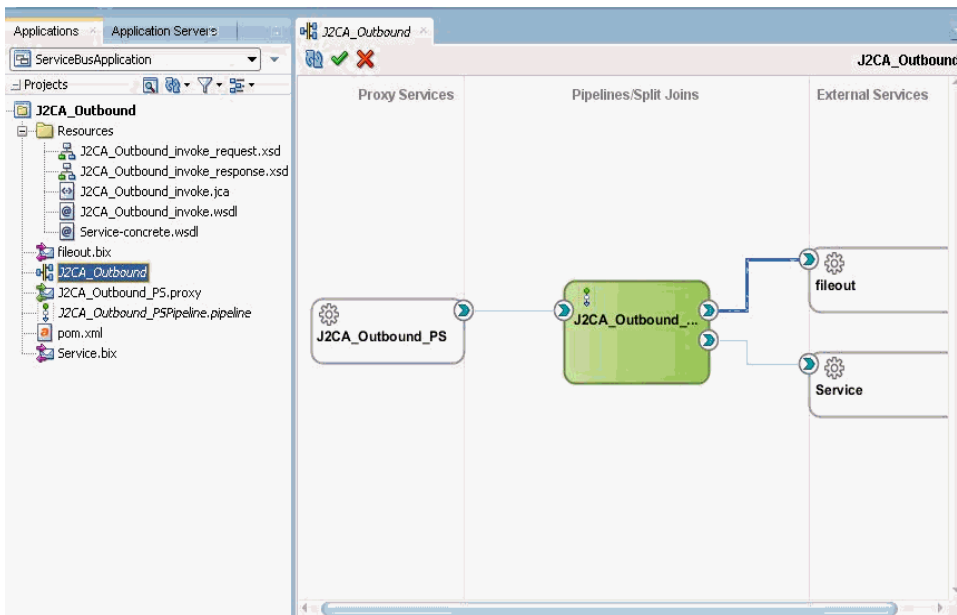
7. 図 10-28 に示すように、「ルーティング」をクリックして、サービスが正しく選択されていることを確認します。

図 10-28 パイプライン構成



8. パイプライン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 (J2CA_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

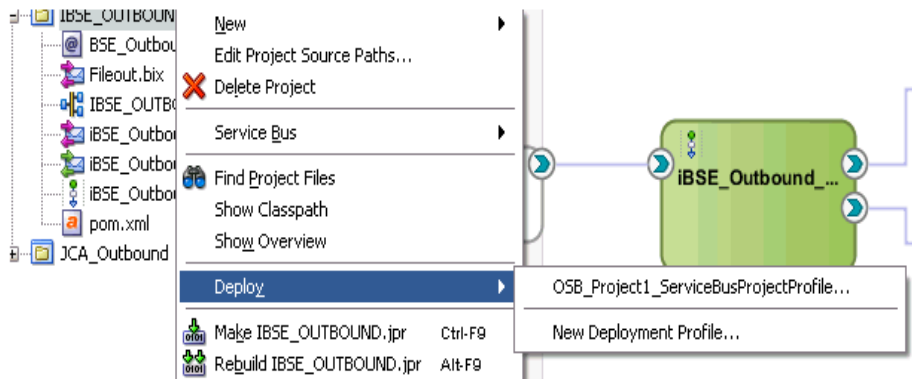


10.1.3 OSB アウトバウンド・プロセスのデプロイ

OSB アウトバウンド・プロセスをデプロイするには、次のステップを実行します。

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。

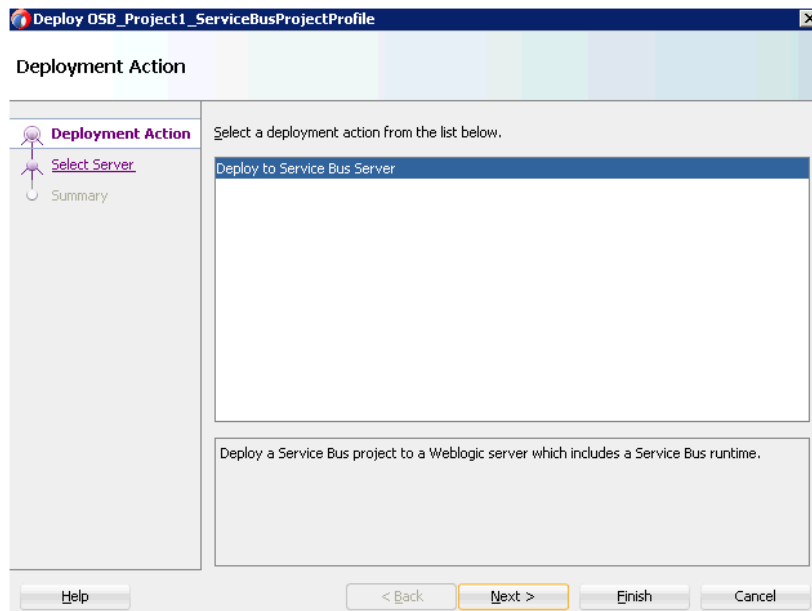
図 10-30 「デプロイ」オプション



「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、「次へ」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、「次へ」をクリックします。

図 10-32 「サーバーの選択」 ページ

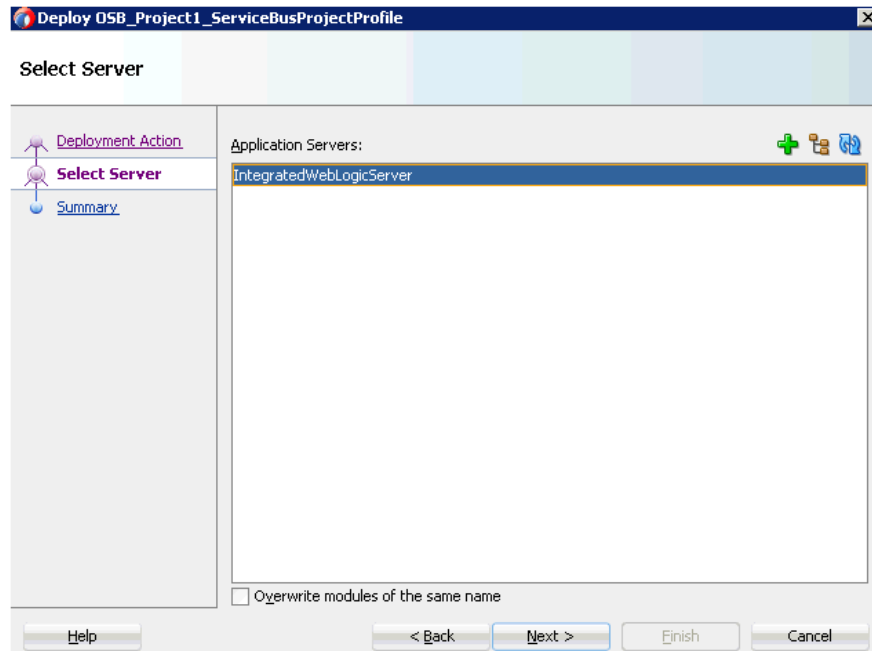
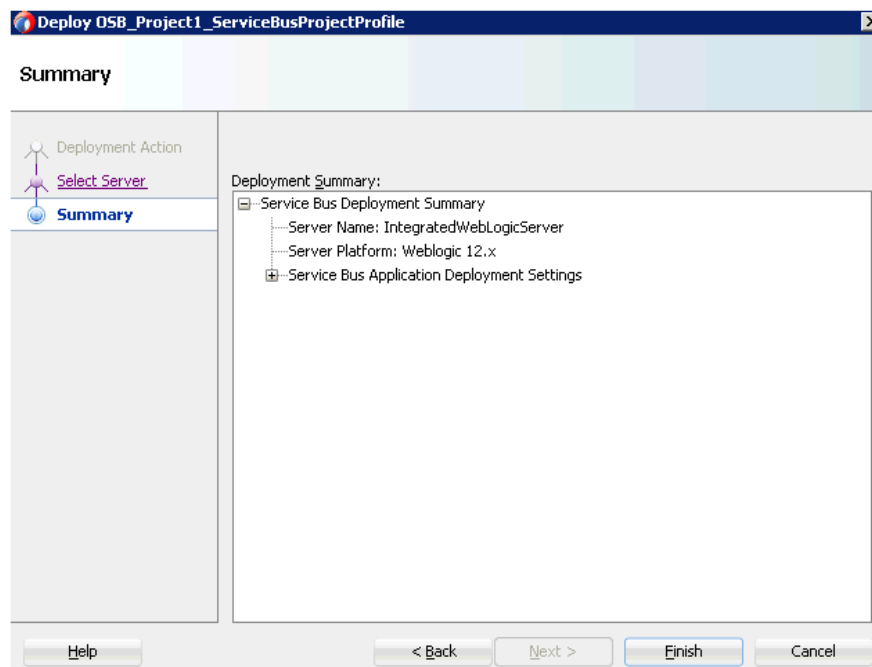


図 10-33 に示すように、「サマリー」 ページが表示されます。

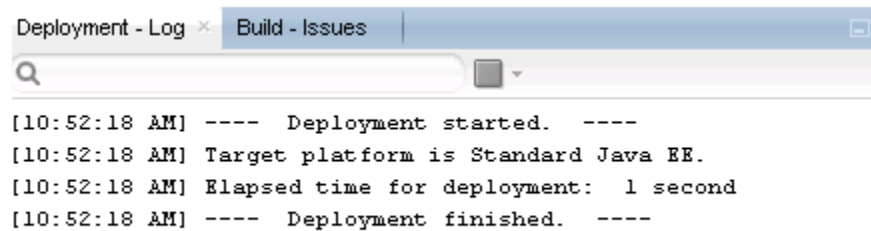
図 10-33 「サマリー」 ページ



- プロジェクトに使用可能なすべてのデプロイメント情報をレビューおよび確認し、「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



```

Deployment - Log x Build - Issues
[10:52:18 AM] ---- Deployment started. ----
[10:52:18 AM] Target platform is Standard Java EE.
[10:52:18 AM] Elapsed time for deployment: 1 second
[10:52:18 AM] ---- Deployment finished. ----
    
```

5. 構成済の入力フォルダ (C:\input など) に入力 XML ファイルをコピーおよび貼付けます。

出力が構成済の出力ロケーション (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 SAP R/3 システムへの OSB インバウンド・プロセスを構成する方法 (J2CA 構成用) について説明します。

Application Adapters インストレーションの次のフォルダに、このインバウンドのユースケース・シナリオのサンプル・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
```

この節では、次のトピック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。詳細については、6-37 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。詳細については、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。次のトピック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」

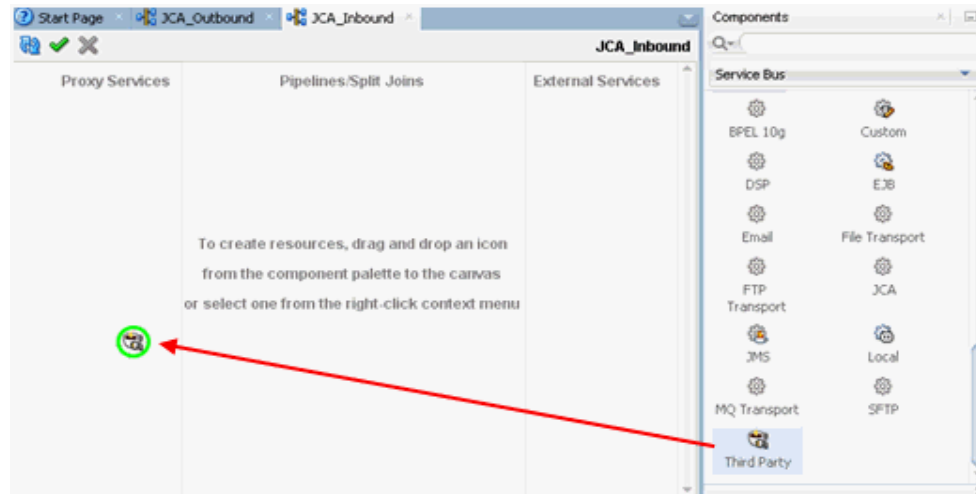
- 10.2.2.4 項「ルーティング・ルール」の構成

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、次のステップを実行します。

1. 図 10-35 に示すように、「Service Bus」コンポーネント・ペインから「プロキシ・サービス」に「サード・パーティ・アダプタ」コンポーネントをドラッグ・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・サービスで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 (デフォルト)。
4. 図 10-36 に示すように、「WSDL URL」フィールドの右の既存の WSDL の検索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

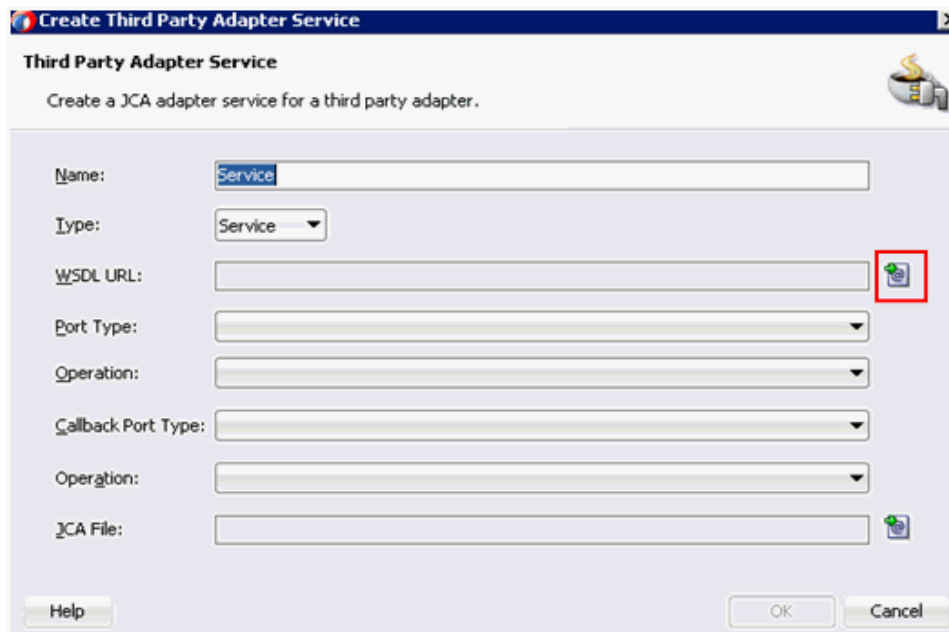
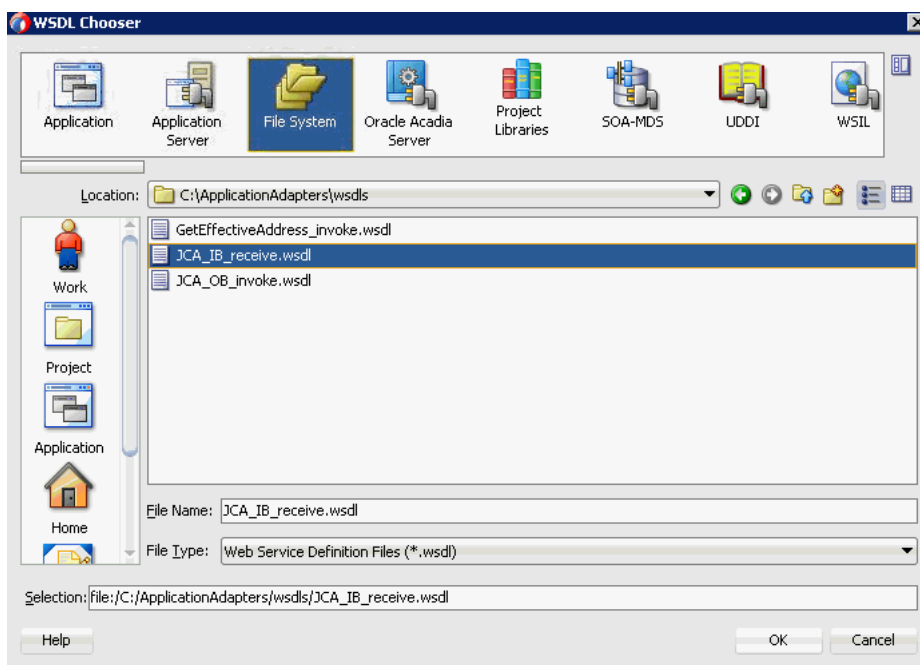


図 10-37 に示すように、「WSDL チューザ」 ダイアログが表示されます。

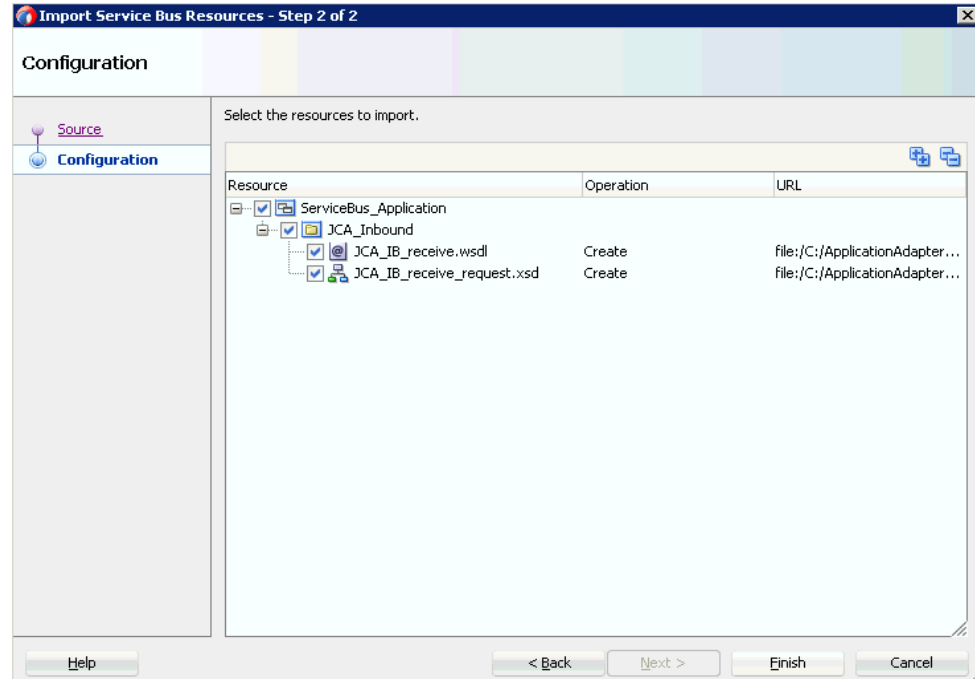
図 10-37 「WSDL チューザ」 ダイアログ



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。
サービス・バス・リソースのインポート・ダイアログが表示されます。
7. 「次へ」 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. 図 10-39 に示すように、「JCA ファイル」フィールドの右の JCA ファイルの検索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

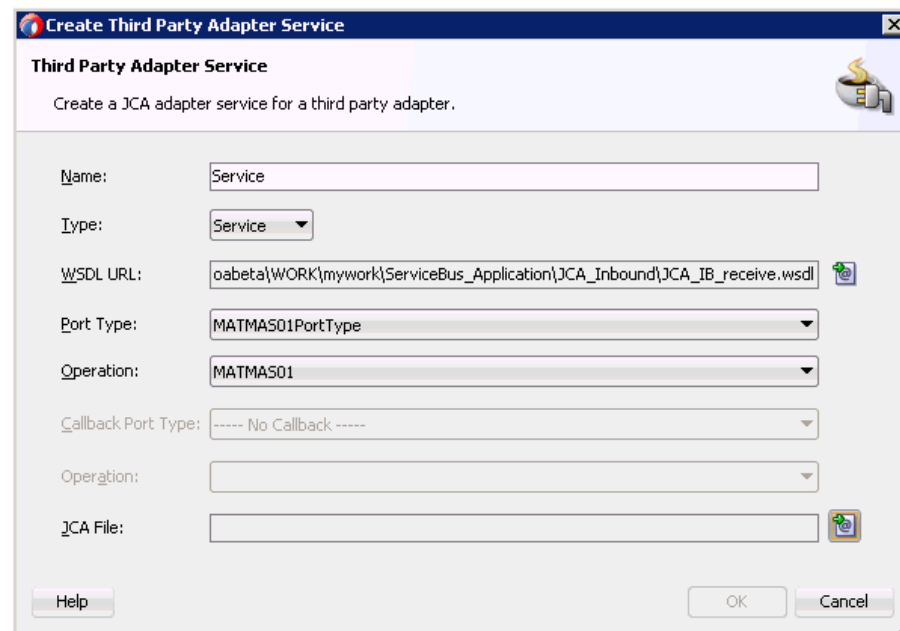
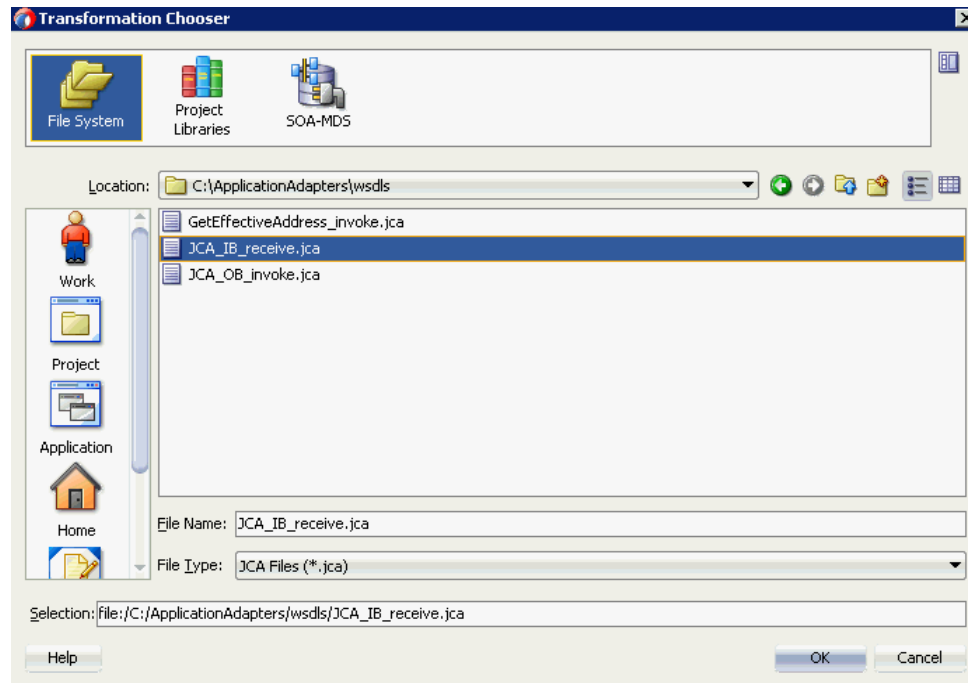


図 10-40 に示すように、トランスフォーメーション・チューザ・ダイアログが表示されます。

図 10-40 トランスフォーメーション・チューザ・ダイアログ

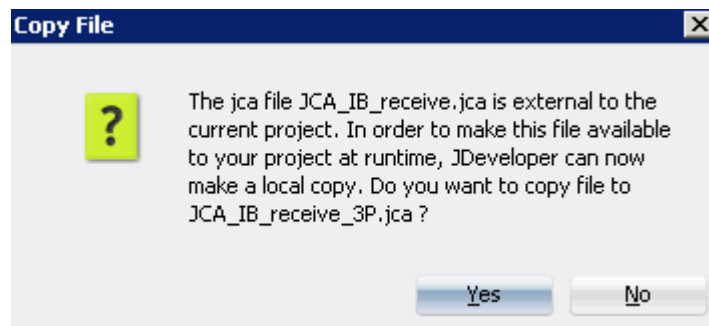


10. WSDL ディレクトリから JCA プロパティ・ファイルを選択します。

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

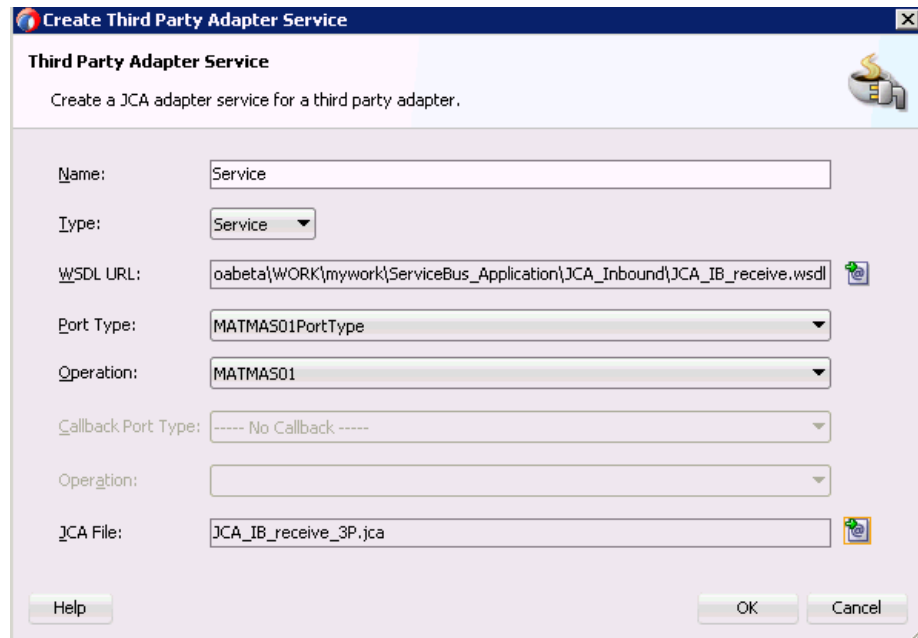


12. 「はい」をクリックします。

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

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。

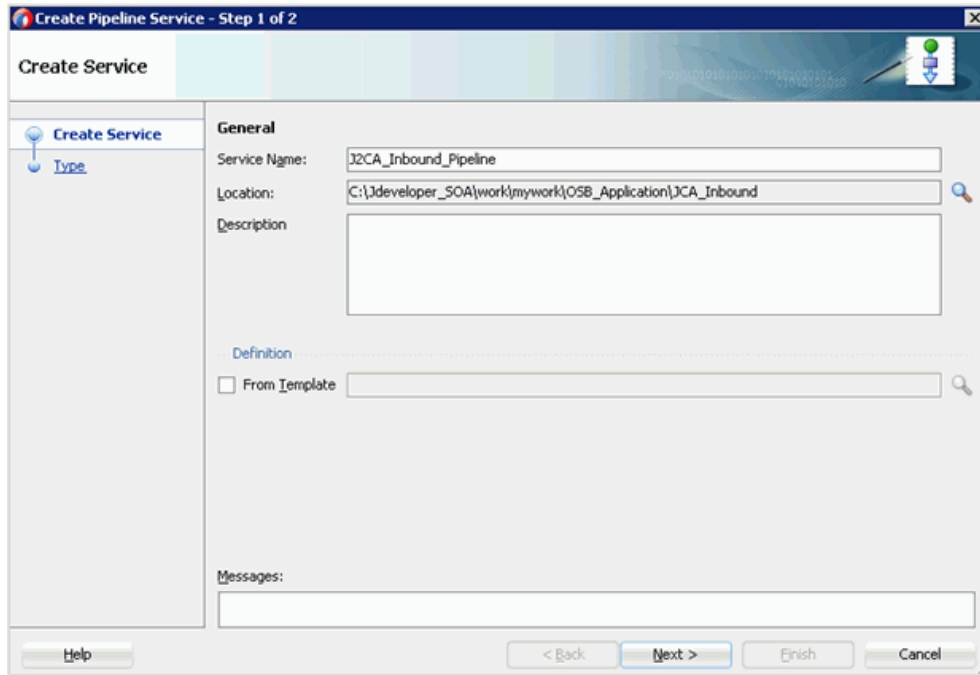
サード・パーティ・アダプタ・サービス・コンポーネントが「プロキシ・サービス」ペインに作成されます。

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。

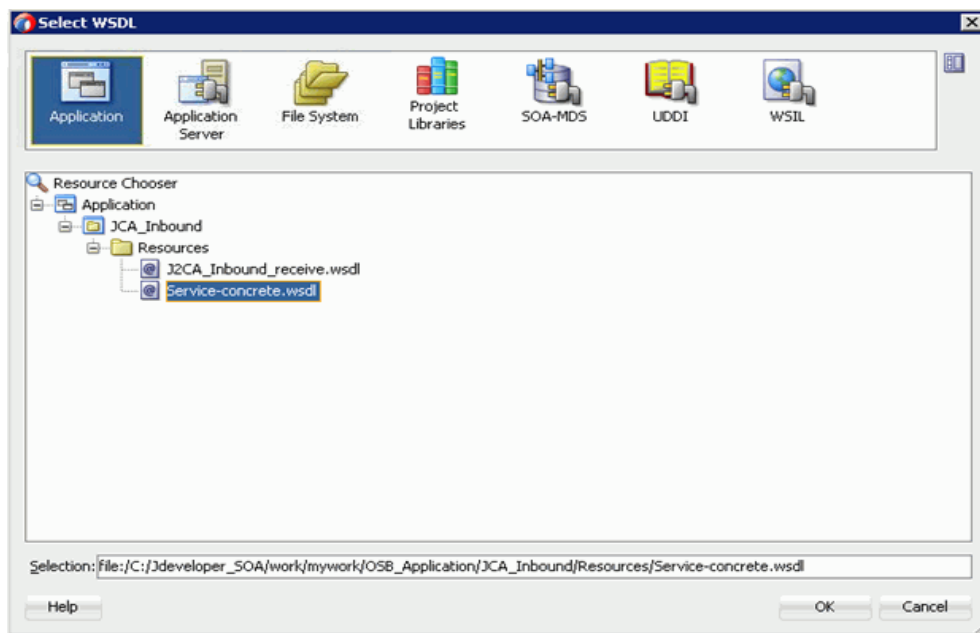
1. 「Service Bus」の下で「リソース」をクリックします。
2. パイプラインをパイプライン/分割結合ペインにドラッグ・アンド・ドロップします。
3. 図 10-43 に示すように、パイプラインの名前を指定して「次へ」をクリックします。

図 10-43 「サービスの作成」 ページ



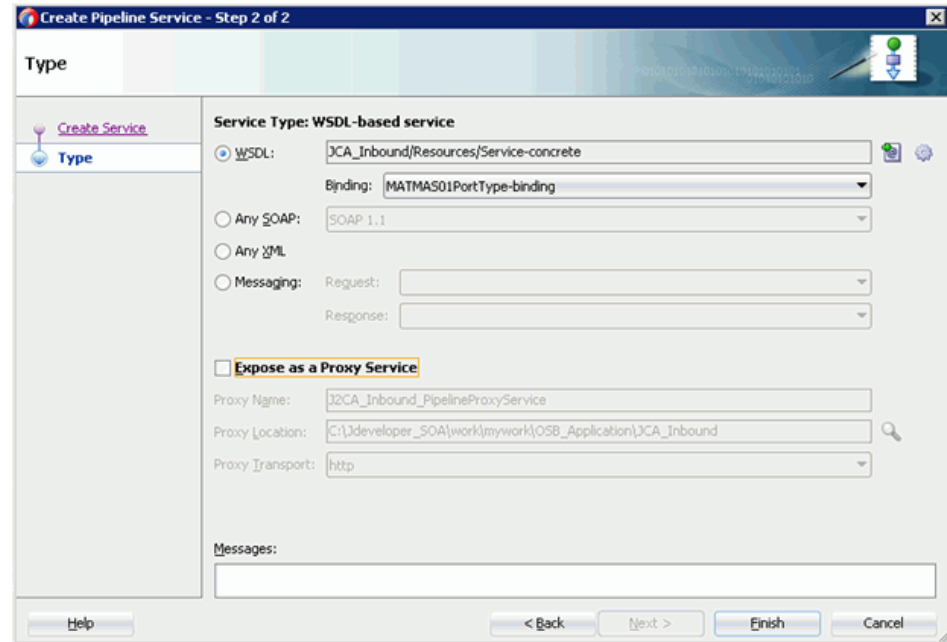
4. パイプライン・サービスの作成ウィンドウで、「WSDL」を選択して「WSDL URL」をクリックします。
5. 図 10-44 に示すように、「WSDL チューザ」ウィンドウで「アプリケーション」を選択し、適切な OSB オブジェクトの「service-concrete.wsdl」を選択して、「OK」をクリックします。

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



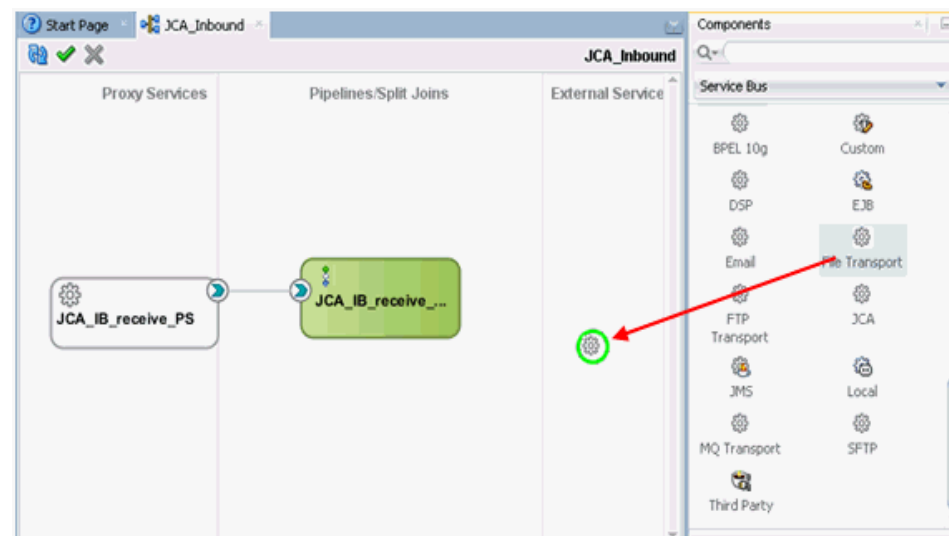
7. プロキシ・サービスをパイプライン/分割結合ペインにドラッグ・アンド・ドロップします。

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・ビジネス・サービスを作成するには、次のようにします。

1. 図 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

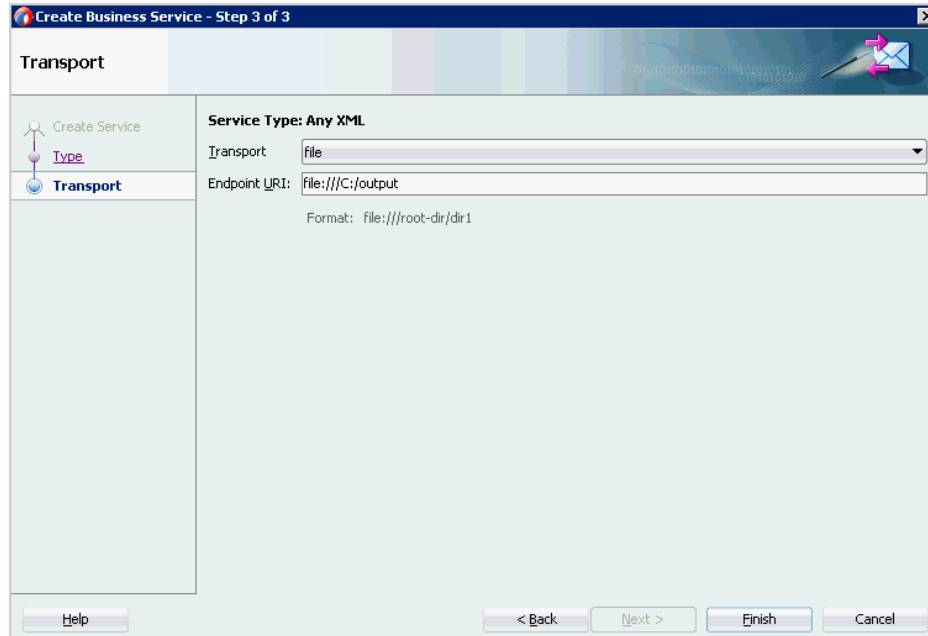


「ビジネス・サービスの作成」ダイアログが表示されます。

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、「次へ」をクリックします。
表示される「タイプ」オプションがデフォルトで選択されています。

3. 「次へ」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。

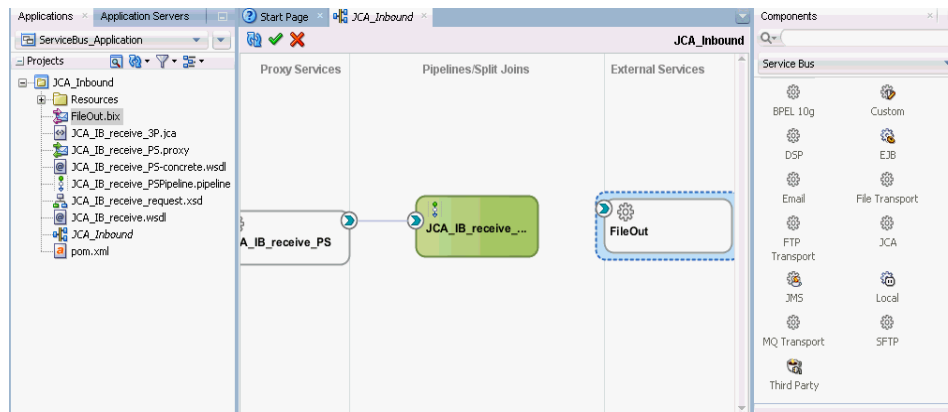
図 10-47 「トランスポート」ペイン



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。

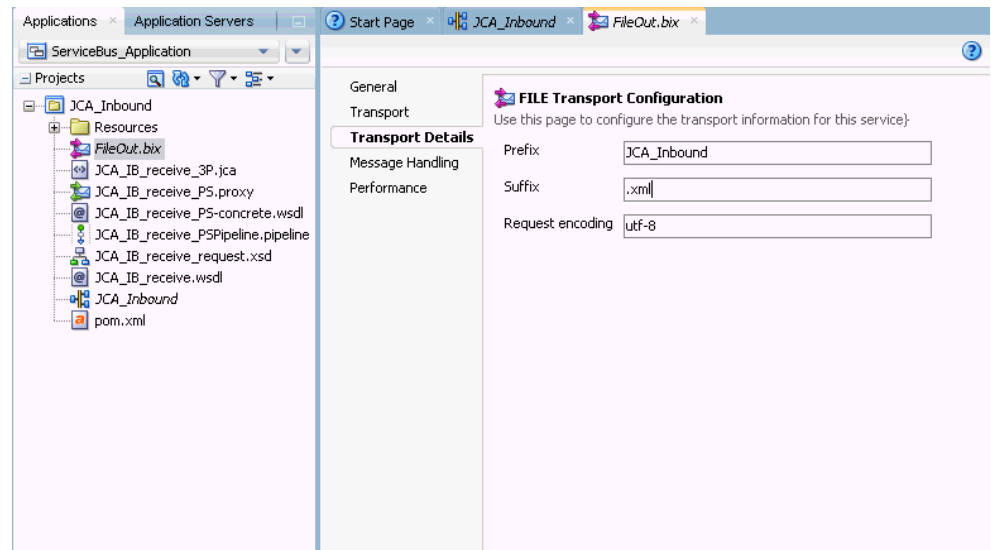
図 10-48 FileOut ビジネス・サービス



「構成」ページが表示されます。

6. 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・トランスポート構成



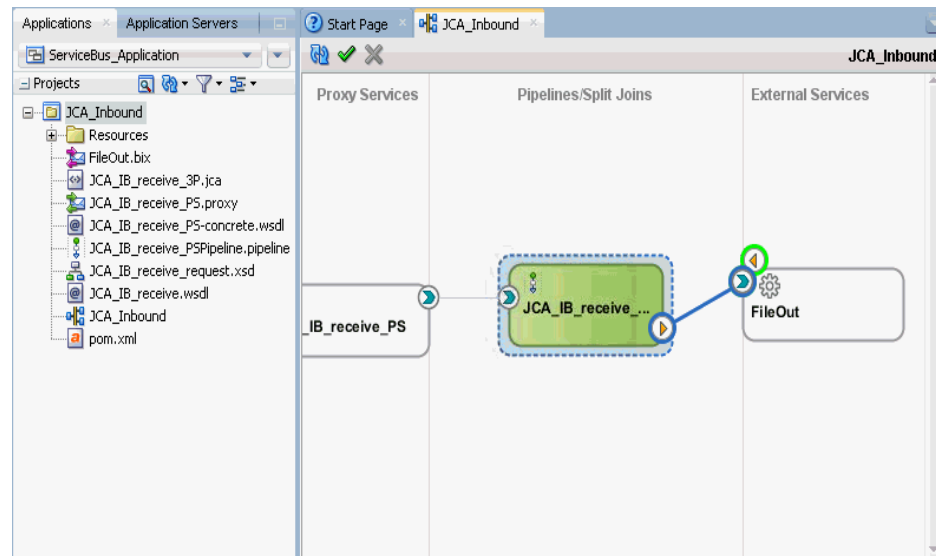
7. 「構成」 ページを保存して閉じます。

10.2.2.4 ルーティング・ルールの構成

ルーティング・ルールを構成するには、次のステップを実行します。

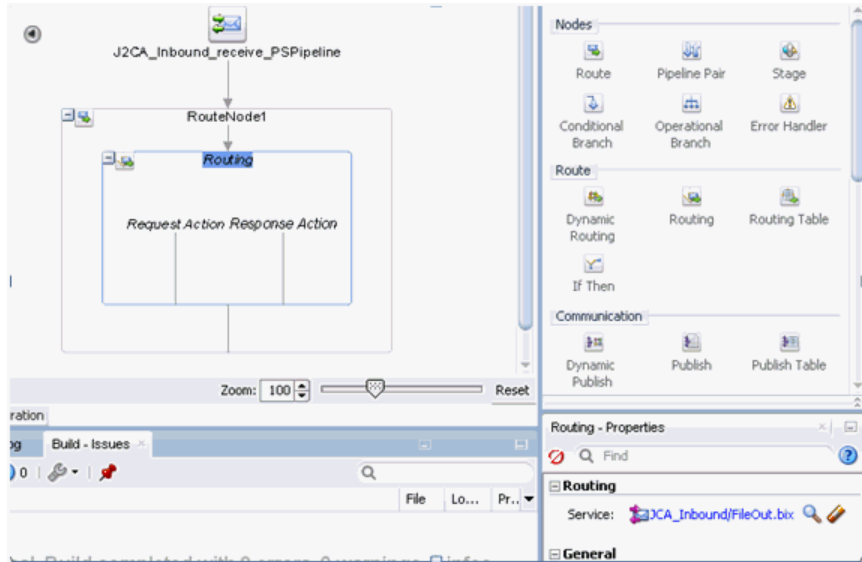
1. 図 10-50 に示すように、パイプライン (JCA_IB_receive_PSPipeline など) とファイル・タイプ・ビジネス・サービス (FileOut など) との間に接続を作成します。

図 10-50 プロキシと FileOut のマッピング



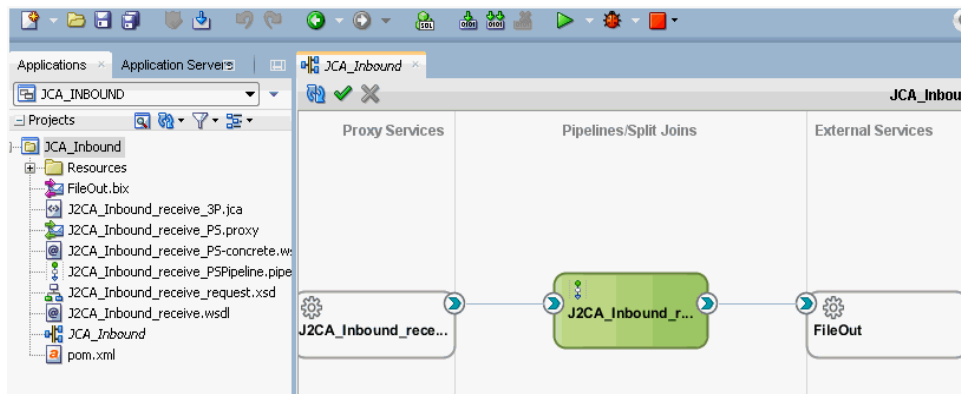
2. パイプライン (JCA_IB_receive_PSPipeline など) をダブルクリックします。
3. 図 10-51 に示すように、「ルーティング」ペインをクリックし、ファイル・タイプ・ビジネス・サービス (FileOut など) が「サービス」フィールドに正しく構成されていることを確認します。

図 10-51 「ルーティング」 ペイン



4. パイプライン構成ページを保存して閉じます。
5. 図 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 構成用) について説明します。

Application Adapters インストールの次のフォルダに、このアウトバウンドのユースケース・シナリオのサンプル・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
```

この節では、次のトピック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。詳細については、6-64 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。詳細については、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。次のトピックについて説明します。

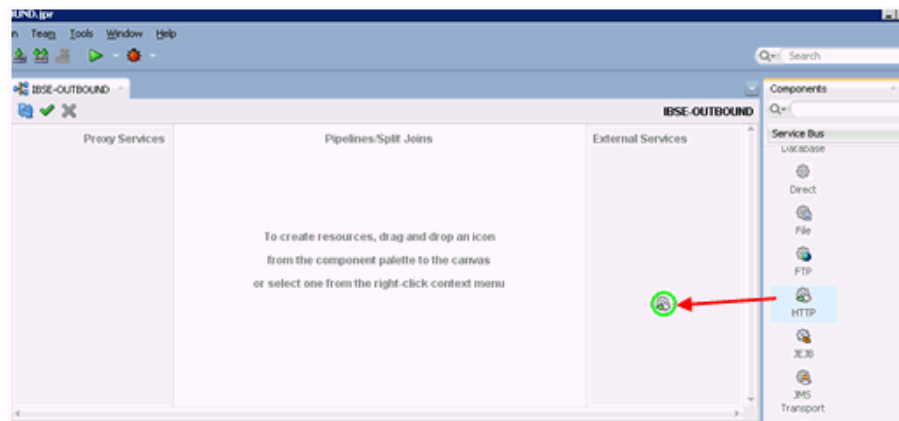
- 10.3.2.1 項「WSDL ベースのビジネス・サービスの構成」
- 10.3.2.2 項「パイプラインを含むプロキシ・サービスの作成」
- 10.3.2.3 項「ファイル・トランスポート・タイプ・ビジネス・サービスの構成」
- 10.3.2.4 項「ルーティング・ルールの構成」

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。

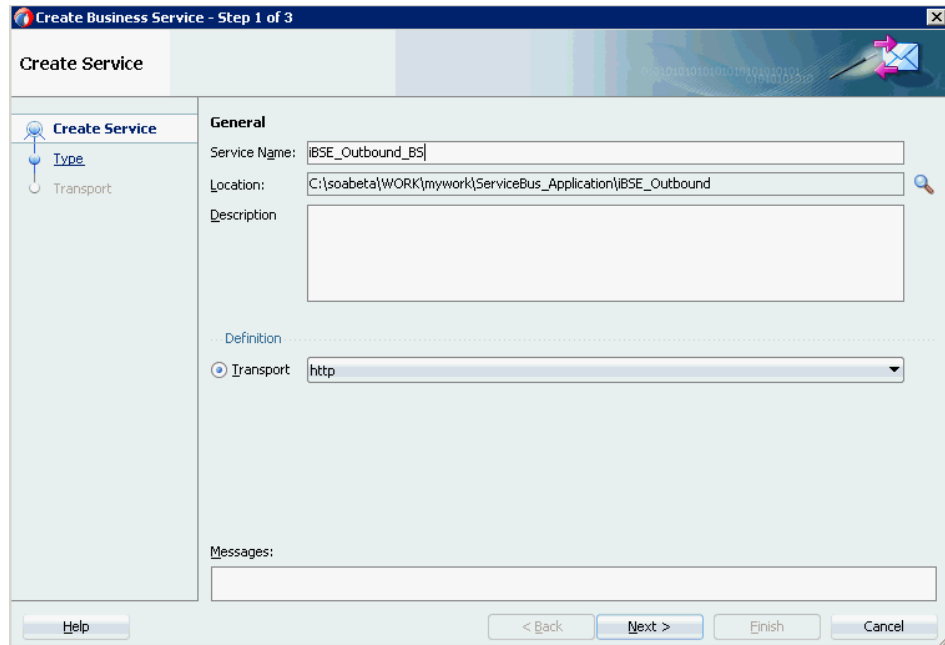
図 10-53 「HTTP」コンポーネント



「ビジネス・サービスの作成」ウィンドウが表示されます。

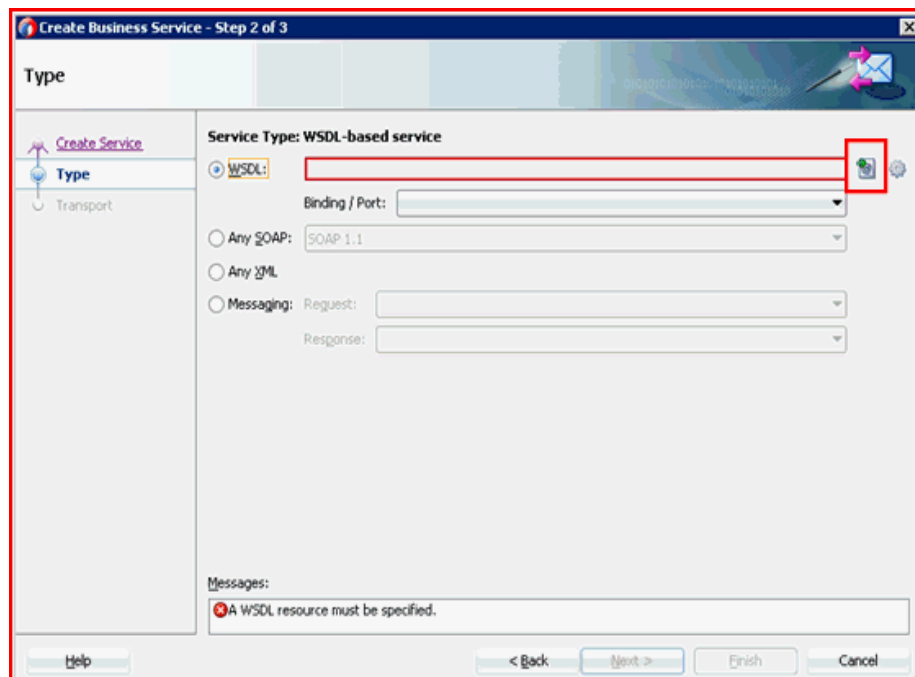
2. [図 10-54](#) に示すように、「サービス名」フィールドにビジネス・サービスに使用する任意の名前を入力し、「次へ」をクリックします。

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。

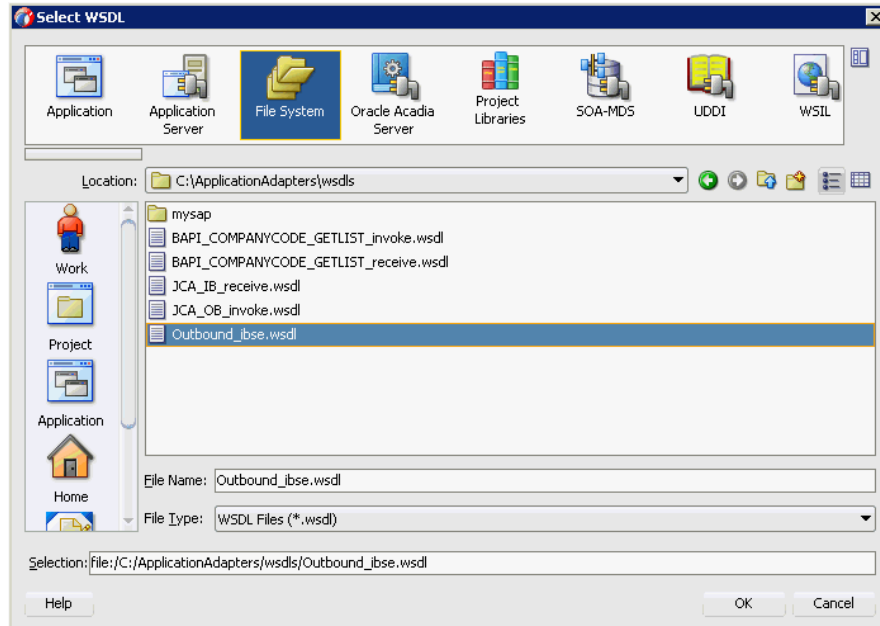
図 10-55 「タイプ」 ペイン



「WSDL の選択」ウィンドウが表示されます。

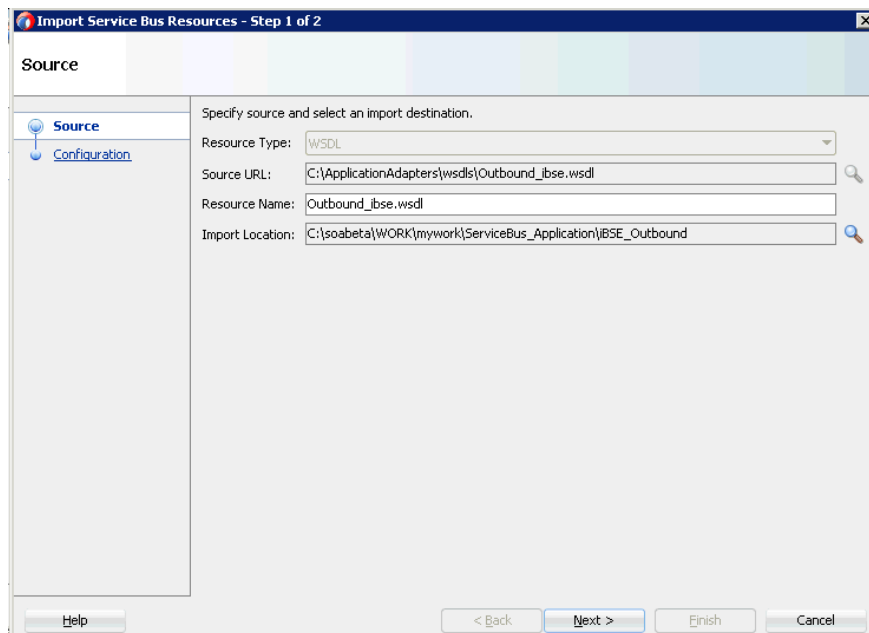
4. 図 10-56 に示すように、「ファイル・システム」フォルダ・アイコンを選択し、WSDL の場所から iBSE WSDL ファイルを参照して選択し、「OK」をクリックします。

図 10-56 「WSDL の選択」ウィンドウ



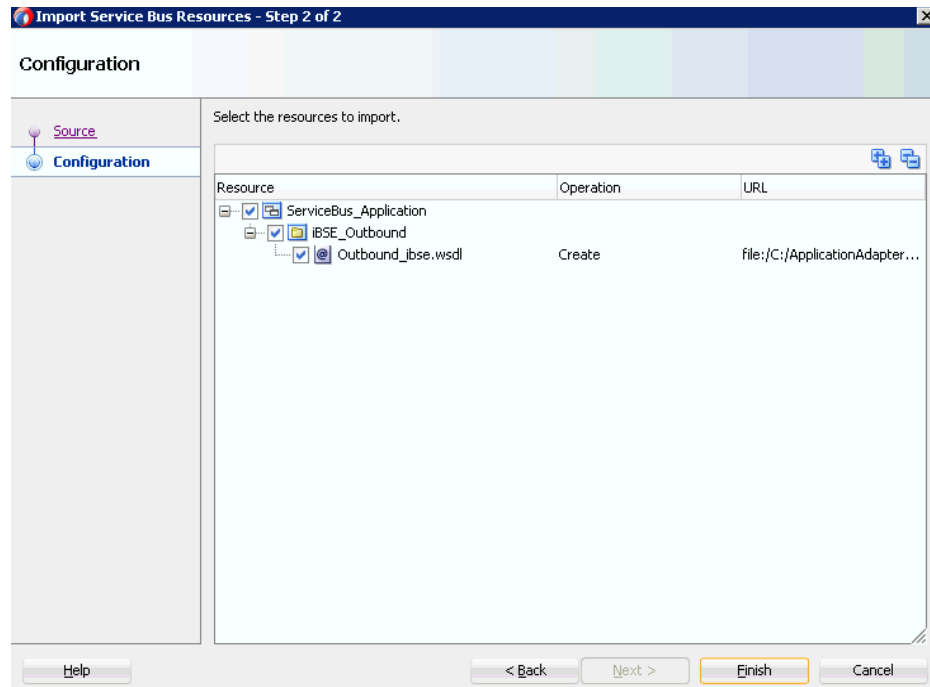
5. 図 10-57 に示すように、表示される「ソース」ペインで「次へ」をクリックします。

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。


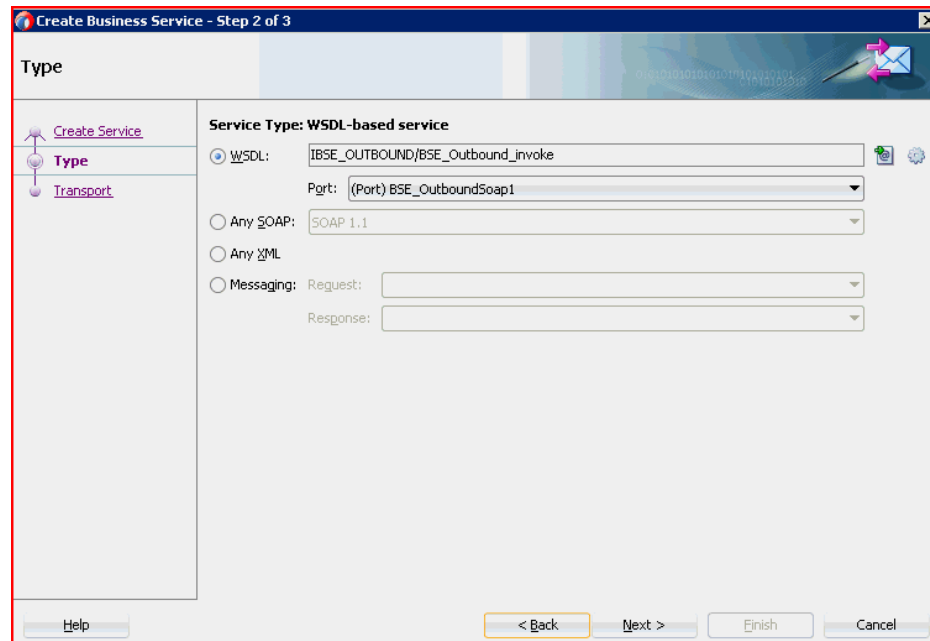
7.  10-59 に示すように、表示される「タイプ」ペインで「次へ」をクリックします。

図 10-59 「タイプ」 ペイン




8.  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

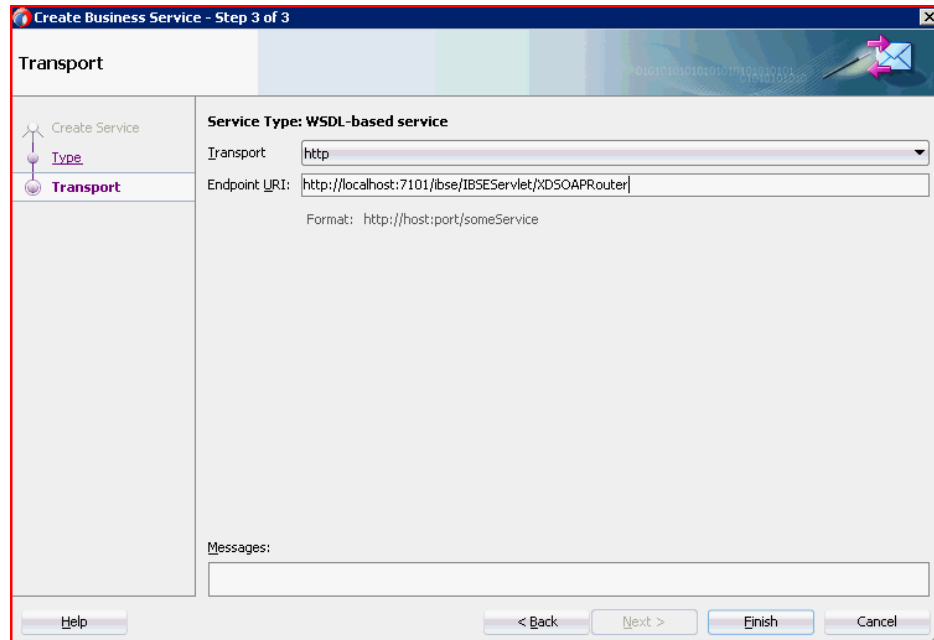
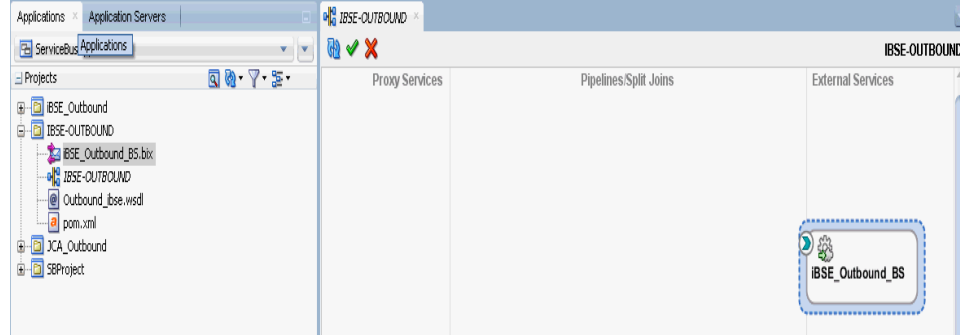


図 10-61 に示すように、外部サービス・ペインにビジネス・サービスが作成されて表示されます。

図 10-61 外部サービス・ペイン

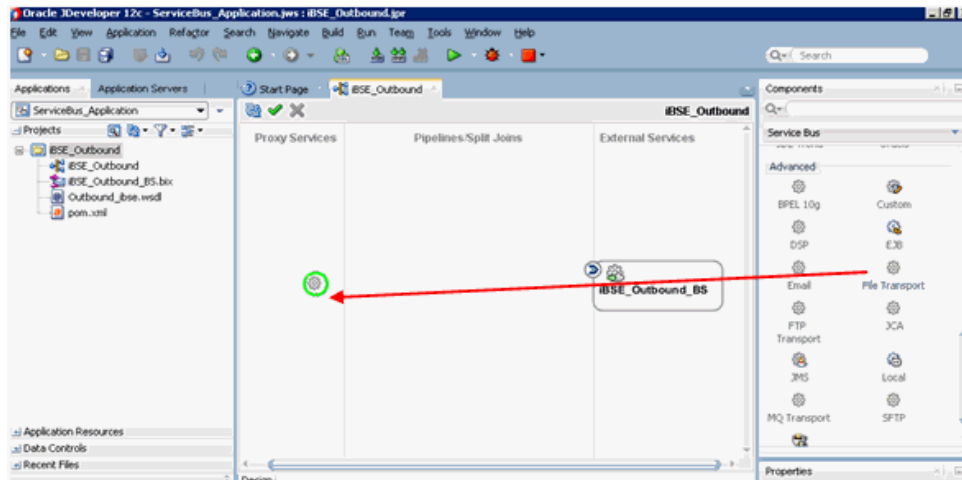


10.3.2.2 パイプラインを含むプロキシ・サービスの作成

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。

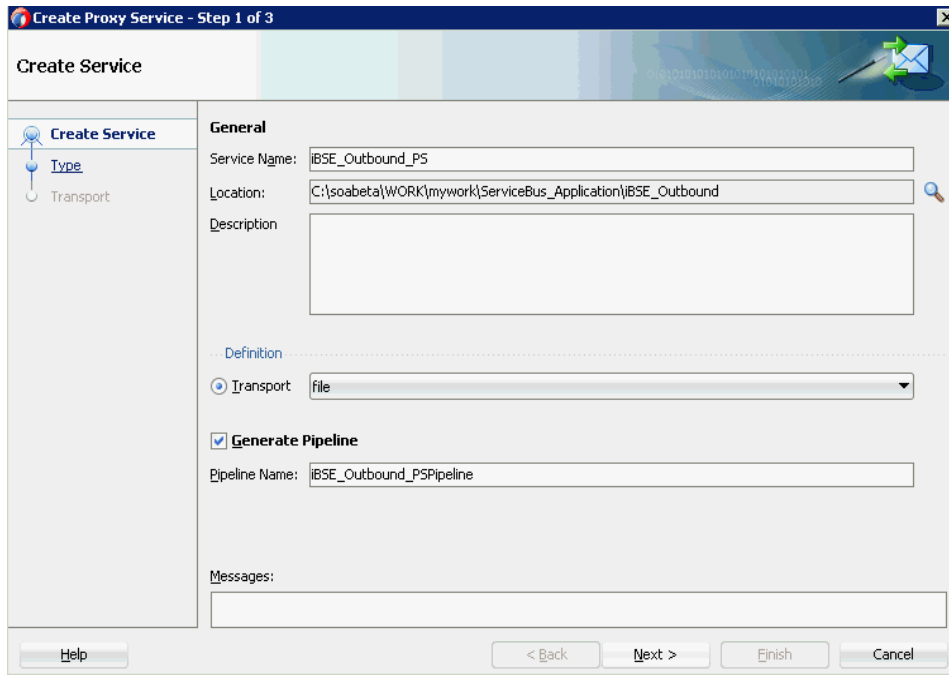
図 10-62 ファイル・トランスポート・コンポーネント



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、「次へ」をクリックします。

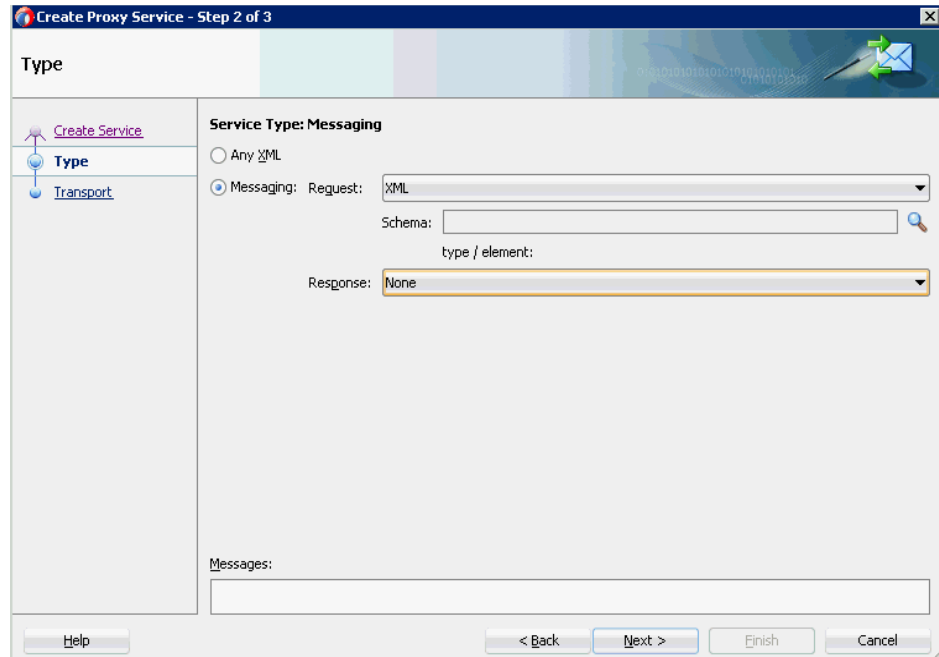
図 10-63 「サービスの作成」ペイン



「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、「次へ」をクリックします。

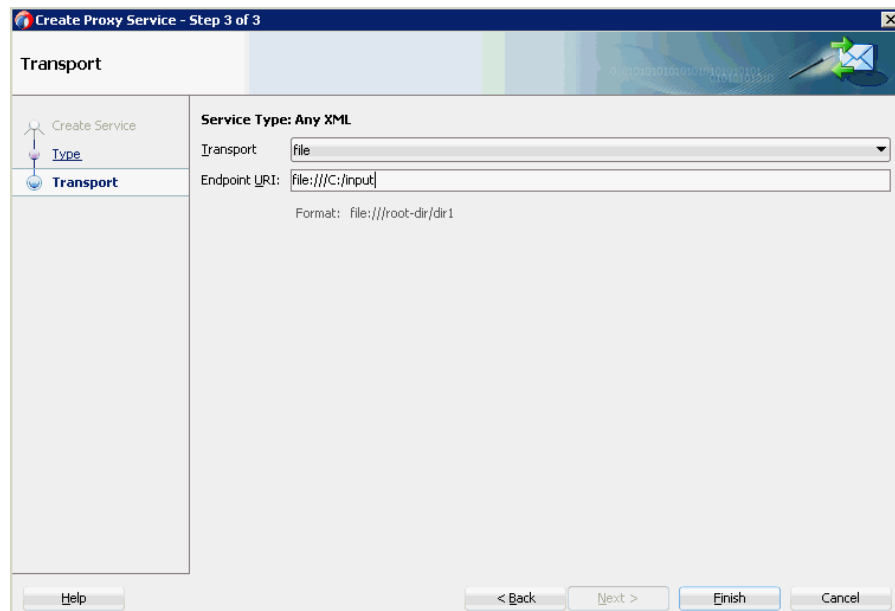
図 10-64 「タイプ」 ペイン



「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。

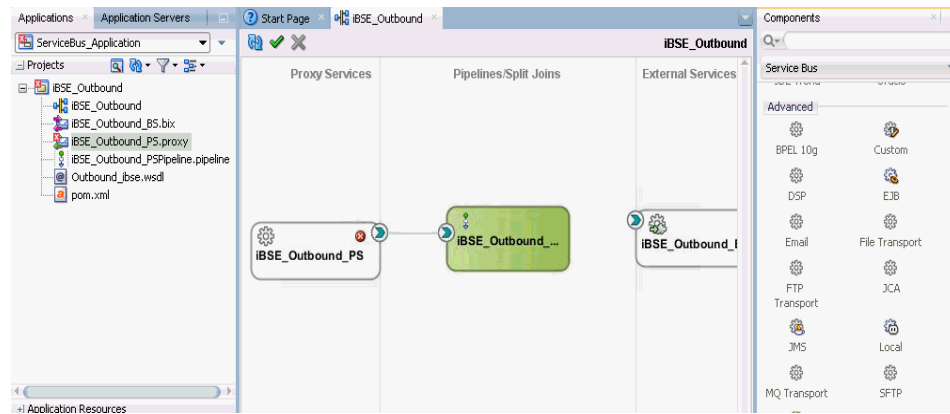
図 10-65 「トランスポート」ウィンドウ



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。

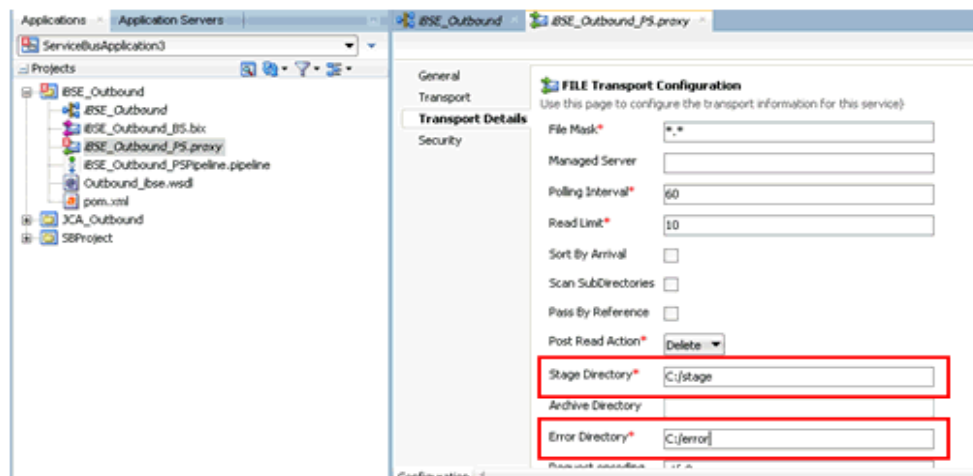
6. 図 10-66 に示すように、作成されたプロキシ・サービス (iBSE_Outbound_PS など) をダブルクリックします。

図 10-66 プロキシ・サービスの編集



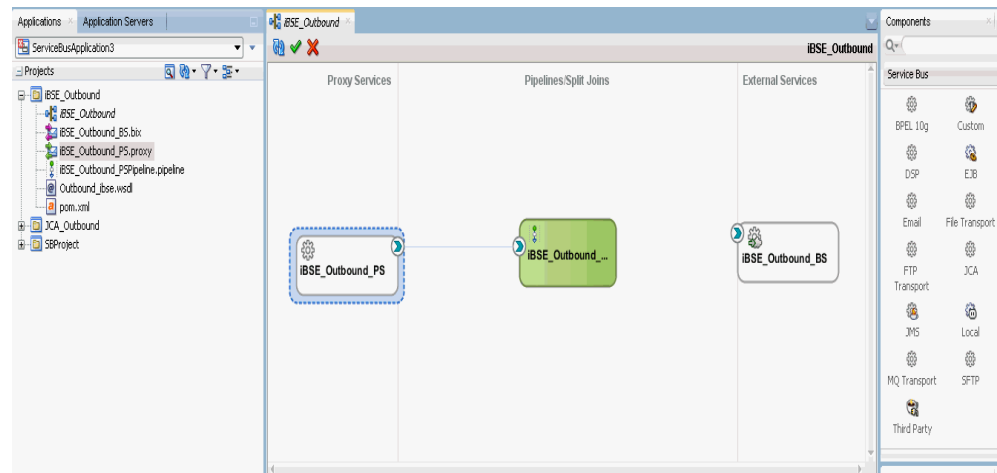
7. 図 10-67 に示すように、表示されるプロキシ・サービスの構成ページで「トランスポートの詳細」を選択し、「ステージ」および「エラー・ディレクトリ」に値を指定します。

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

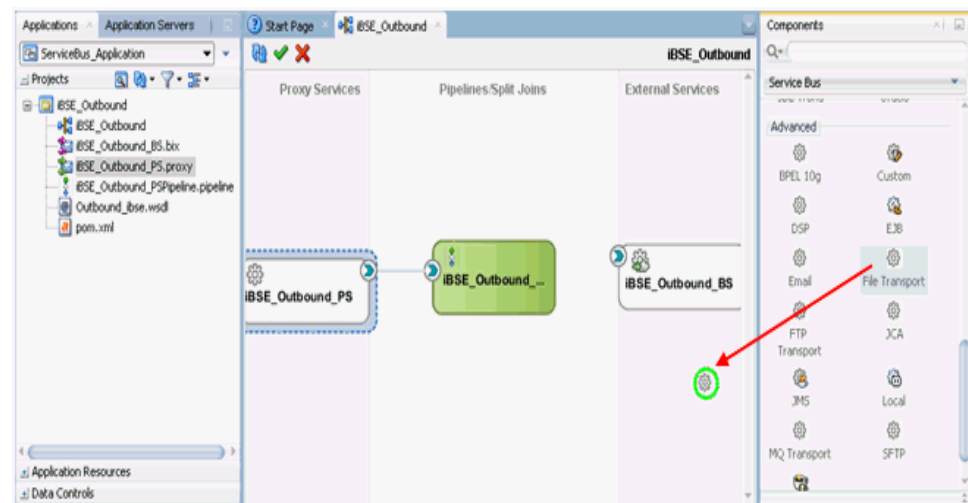


10.3.2.3 ファイル・トランスポート・タイプ・ビジネス・サービスの構成

ファイル・トランスポート・ビジネス・サービスを作成するには、次のようにします。

1. 図 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」ページから外部サービス・ペインにドラッグ・アンド・ドロップします。

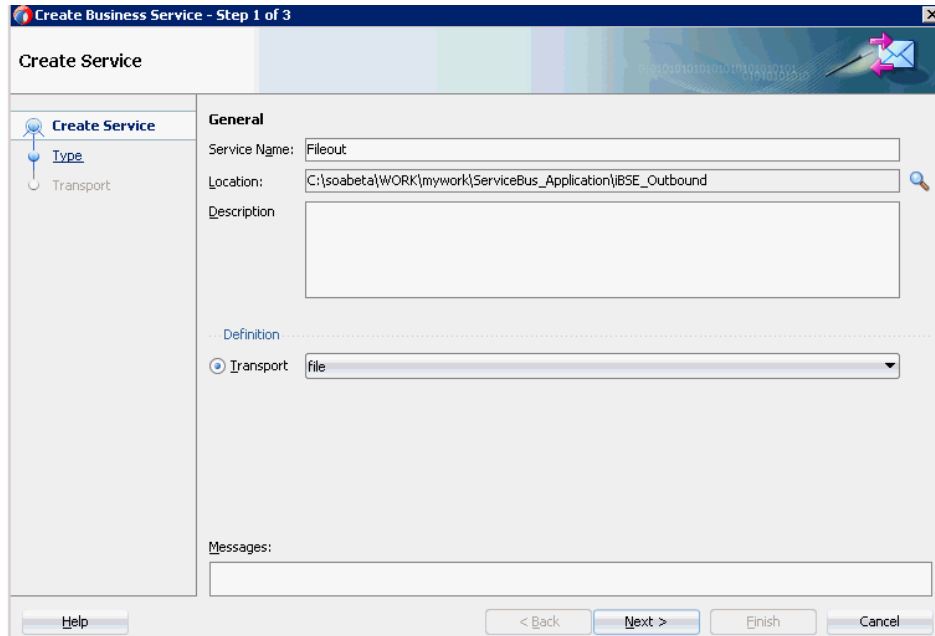
図 10-69 ファイル・トランスポート・コンポーネント



「ビジネス・サービスの作成」ダイアログが表示されます。

2. 図 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、「次へ」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. 「次へ」 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

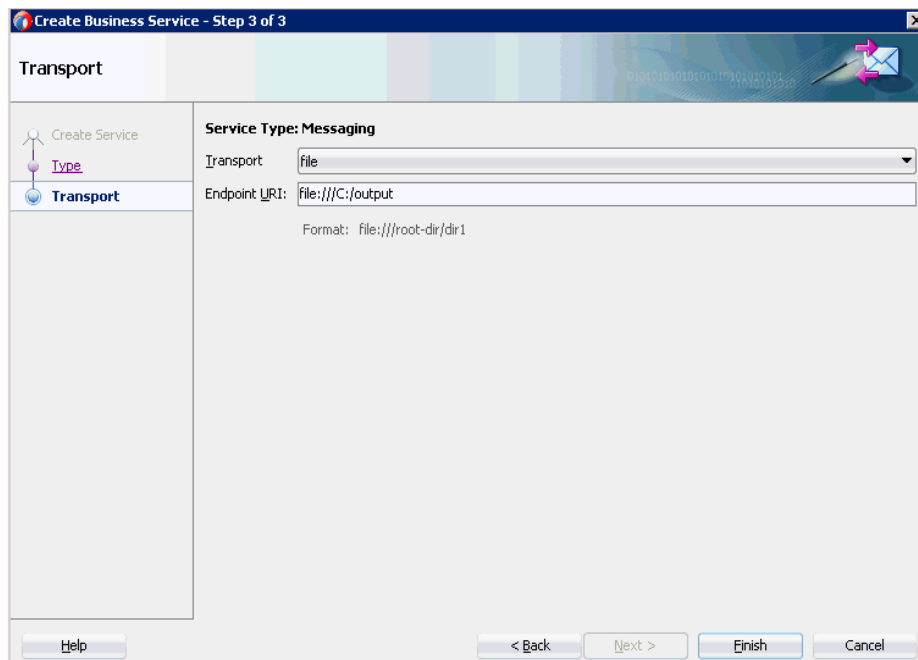
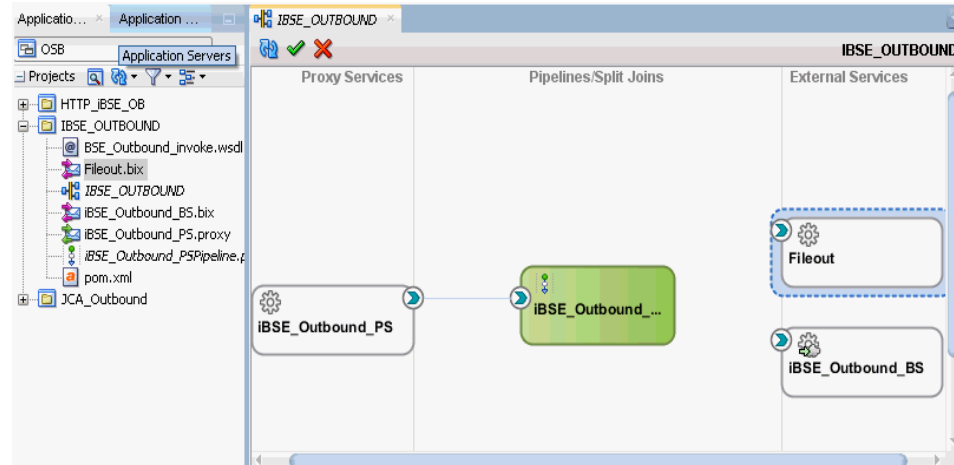


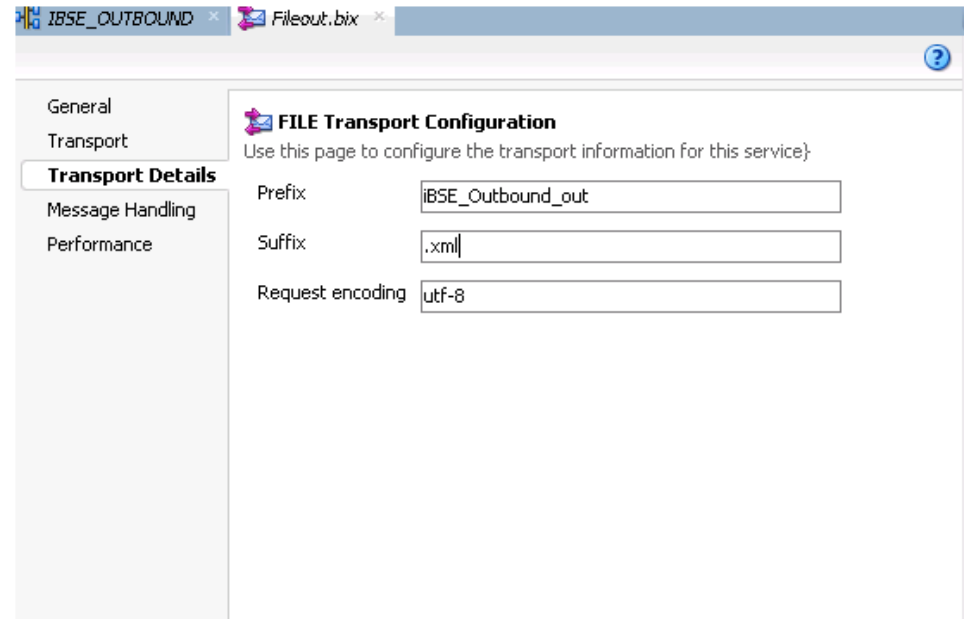
図 10-72 に示すように、ファイル・トランスポート・ビジネス・サービス Fileout が作成されて表示されます。

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」



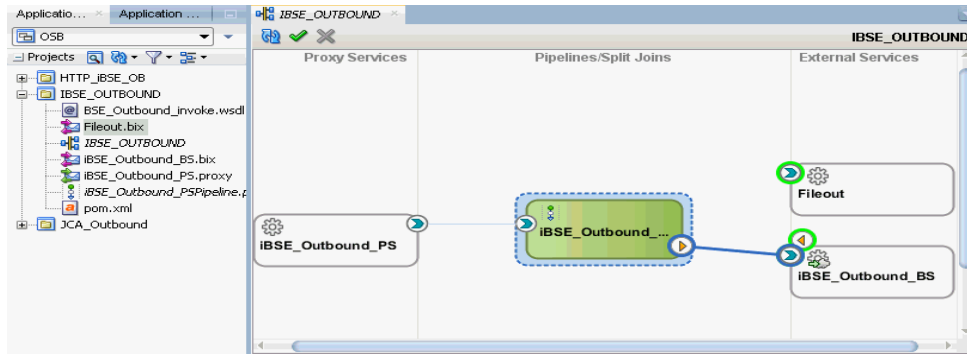
6. 構成ページを保存して閉じ、overview.xml (iBSE_Outbound など) をダブルクリックします。

10.3.2.4 ルーティング・ルールの構成

ルーティング・ルールを構成するには、次のステップを実行します。

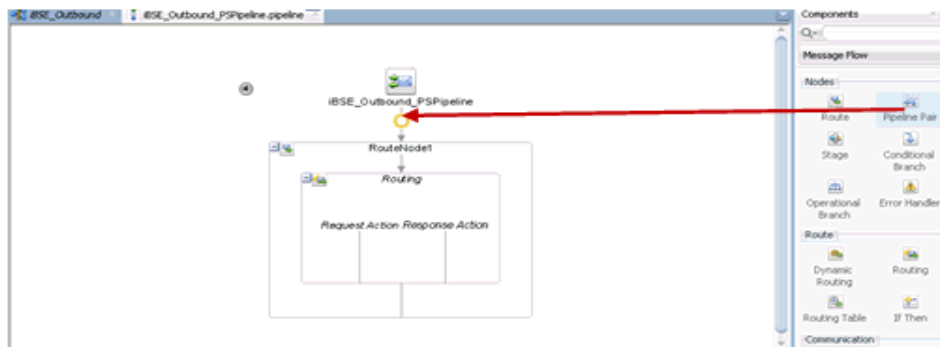
1. 図 10-74 に示すように、「パイプライン」コンポーネント (iBSE_Outbound_PSPipeline など) と WSDL ベースのビジネス・サービス (iBSE_Outbound_BS など) との間に接続を作成します。

図 10-74 「パイプライン」コンポーネント



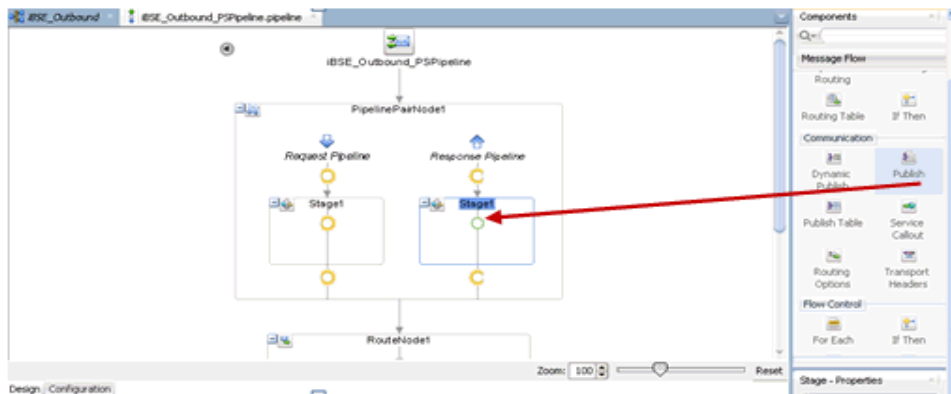
2. パイプライン/分割結合ペインで、「パイプライン」コンポーネント (iBSE_Outbound_PSPipeline など) をダブルクリックします。
3. 図 10-75 に示すように、パイプライン・ペア・ノードを「ノード」ペインから「パイプライン」(iBSE_Outbound_PSPipeline など) と「RouteNode1」の間の領域にドラッグ・アンド・ドロップします。

図 10-75 パイプライン・ペア・ノード



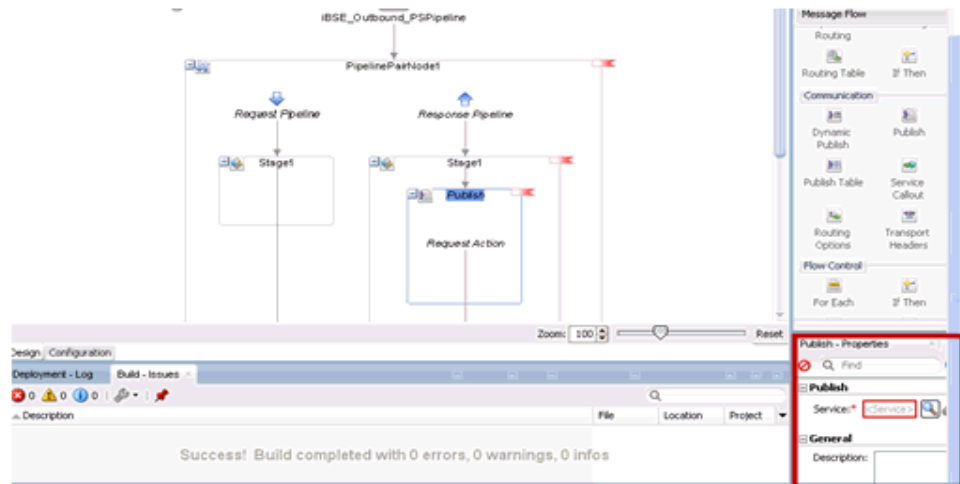
4. 図 10-76 に示すように、「パブリッシュ」ノードを「通信」ペインから「レスポンス・パイプライン」の「ステージ 1」の下の領域にドラッグ・アンド・ドロップします。

図 10-76 「パブリッシュ」ノード



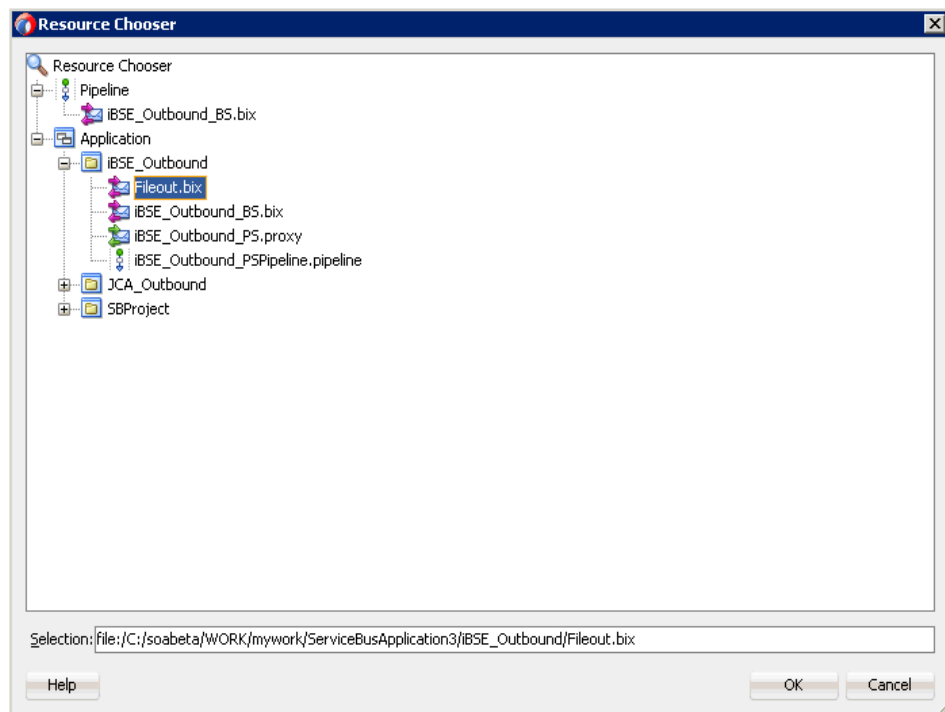
5. 図 10-77 に示すように、プロパティのパブリッシュの右ペインで、「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 10-77 「参照」アイコン



- 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。

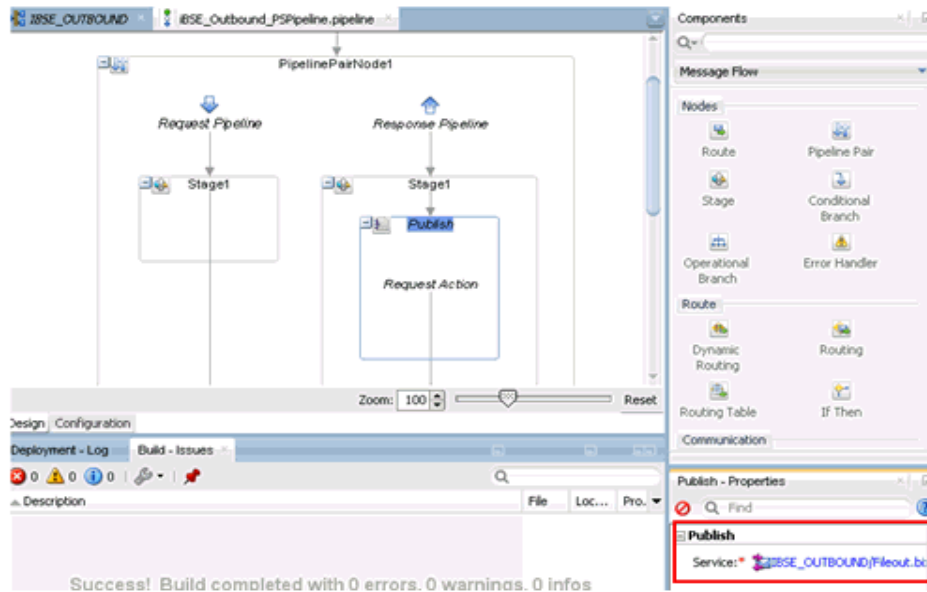
図 10-78 リソース・チューザ



パイプライン構成ページに戻ります。

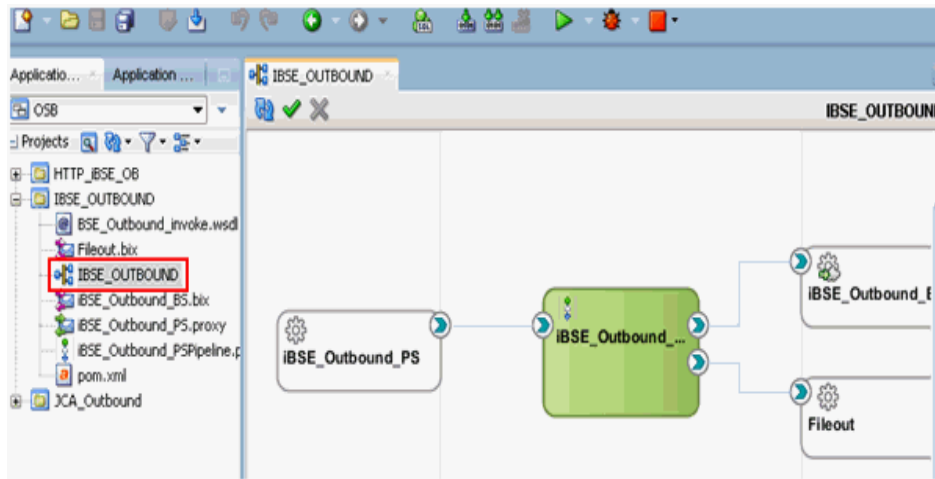
- 図 10-79 に示すように、選択したサービスが右ペインの「パブリッシュ」ペインに構成されます。

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。

Application Adapters 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
```

この節では、次のトピック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。

次のステップを実行して、メッセージ用の WSDL ファイルを生成します。

1. 最初の SAP アダプタ・ターゲット (target1 など) およびチャンネルを作成します。WSDL ファイルを生成する前に、チャンネルが開始し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、チャンネルおよび WSDL ファイルの作成の詳細は、[第 4 章「Oracle Application Adapter for SAP R/3 の構成」](#) 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。次のトピック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」](#)

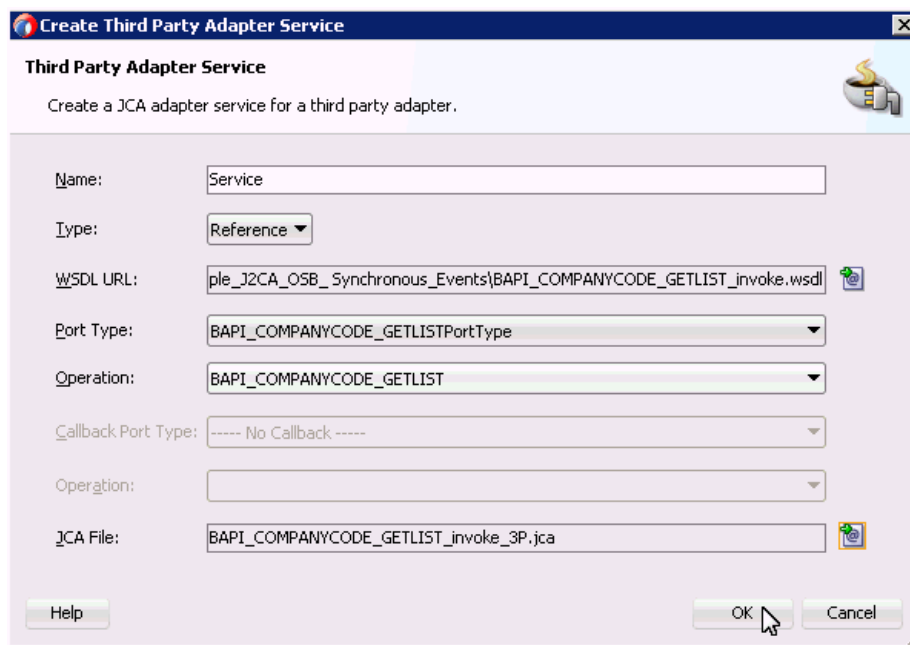
10.4.2.1 アウトバウンドおよびインバウンド WSDL に対するサード・パーティ・アダプタ・サービス・コンポーネントの構成

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. 「Service Bus」コンポーネント・ペインから外部サービス・ペインに「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドの右の既存の WSDL の検索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。
サービス・バス・リソースのインポート・ダイアログが表示されます。
 6. 「次へ」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」フィールドの右の JCA ファイルの検索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。
JCA プロパティ・ファイルのコピーがプロジェクト・フォルダ内に作成されます。
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

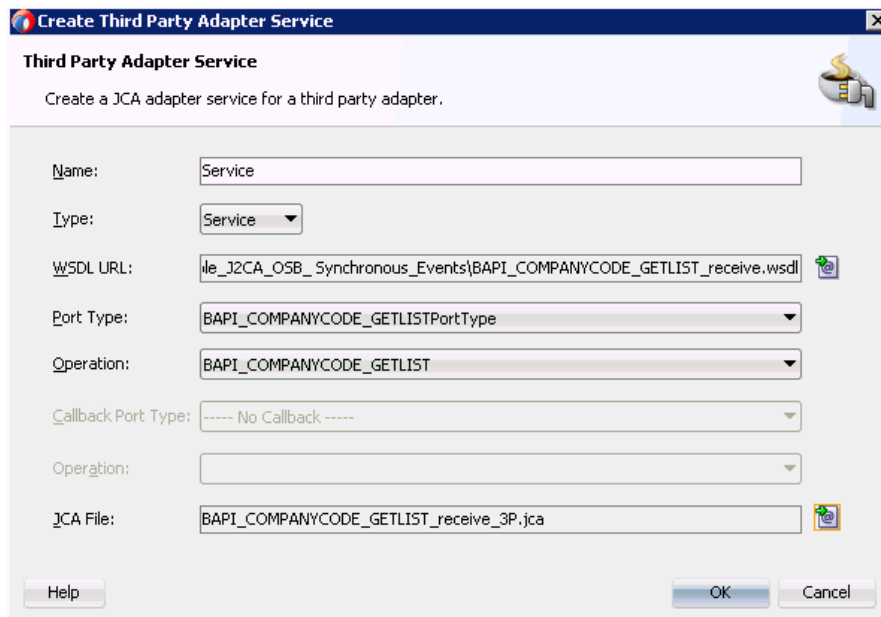


12. 「OK」をクリックします。

外部サービスにビジネス・サービスが作成されます。

13. 「Service Bus」コンポーネント・ペインから「プロキシ・サービス」に「サード・パーティ・アダプタ」コンポーネントをドラッグ・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」フィールドの右の既存の WSDL の検索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 ファイルを参照して選択します。
17. 「OK」をクリックします。
サービス・バス・リソースのインポート・ダイアログが表示されます。
18. 「次へ」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」フィールドの右の JCA ファイルの検索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。
JCA プロパティ・ファイルのコピーがプロジェクト・フォルダ内に作成されます。
[図 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、次のステップを実行します。

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、「次へ」をクリックします。
3. 「WSDL」ボタンを選択し、フィールドの近くにある 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。

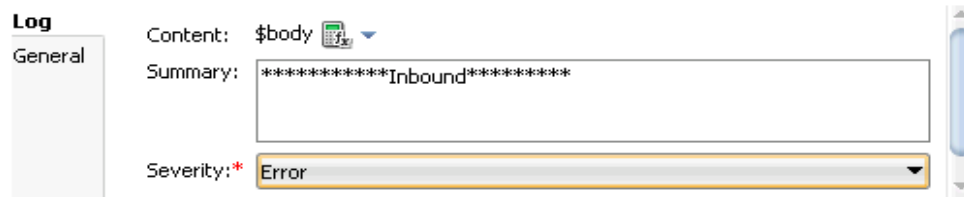
6. プロキシ・サービスとして公開のチェック・ボックスの選択を解除します。
7. 「終了」をクリックします。
プロキシ・サービスが作成されます。
8. プロキシ・サービスとパイプラインを接続してから、変更を保存します。

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、次のステップを実行します。

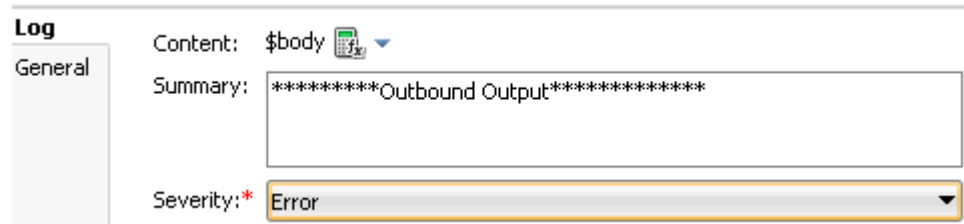
1. パイプラインとビジネス・サービスを接続します。
2. パイプライン / 分割結合ペインで、パイプラインをダブルクリックします。
パイプライン構成ページが表示されます。
3. 「ログ」ノードをコンポーネントの「レポート」ペインから「リクエスト・アクション」の下領域にドラッグ・アンド・ドロップします。
4. ログ・プロパティ・ペインで、「コンテンツ」フィールドの <expression> の近くにあるアイコンをクリックします。
5. XQuery 式ビルダー・ページで、「式」フィールドの下領域のテキスト領域に \$body と入力し、「OK」をクリックします。
6. 図 10-83 に示すように、「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



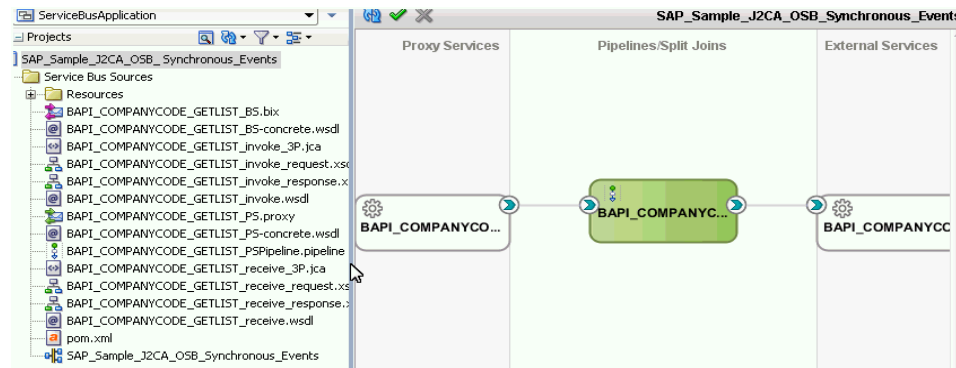
7. 「ログ」ノードをコンポーネントの「レポート」ペインから「レスポンス・アクション」の下領域にドラッグ・アンド・ドロップします。
8. ログ・プロパティ・ペインで、「コンテンツ」フィールドの <expression> の近くにあるアイコンをクリックします。
9. XQuery 式ビルダー・ページで、「式」フィールドの下領域のテキスト領域に \$body と入力し、「OK」をクリックします。
10. 図 10-84 に示すように、「サマリー」フィールドに必要な任意のテキスト・メッセージを入力し (オプション)、「ログ」領域の下領域の「重大度」レベルを示すドロップダウン・リストで「エラー」を選択します。

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、パイプライン構成ページを保存して閉じます。

図 10-85 パイプライン構成ページ



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

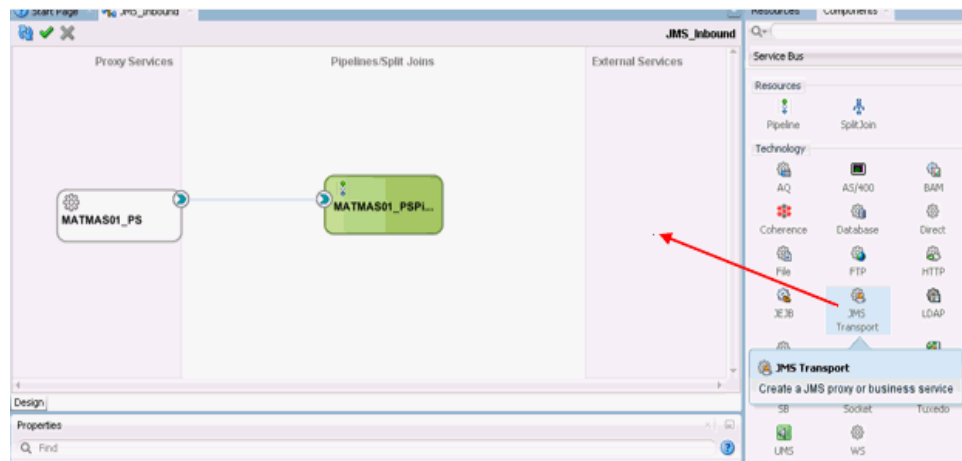
OSB プロセスをデプロイする方法の詳細は、10-17 ページの 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」を参照してください。

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 SAP R/3 システムへの JMS インバウンド・プロセスを構成する方法 (J2CA 構成用) について説明します。

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。詳細については、6-37 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。詳細については、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、7-14 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。詳細については、10-25 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SPORT・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SPORT・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。

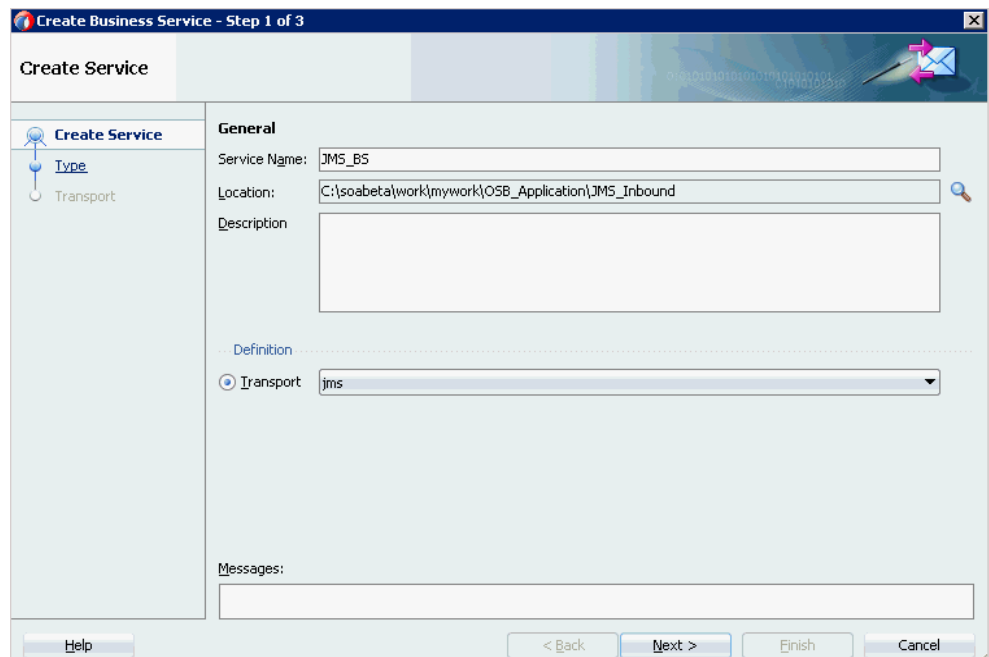
図 10-86 JMS トランスポート・コンポーネント



「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、「次へ」をクリックします。

図 10-87 「サービスの作成」ペイン

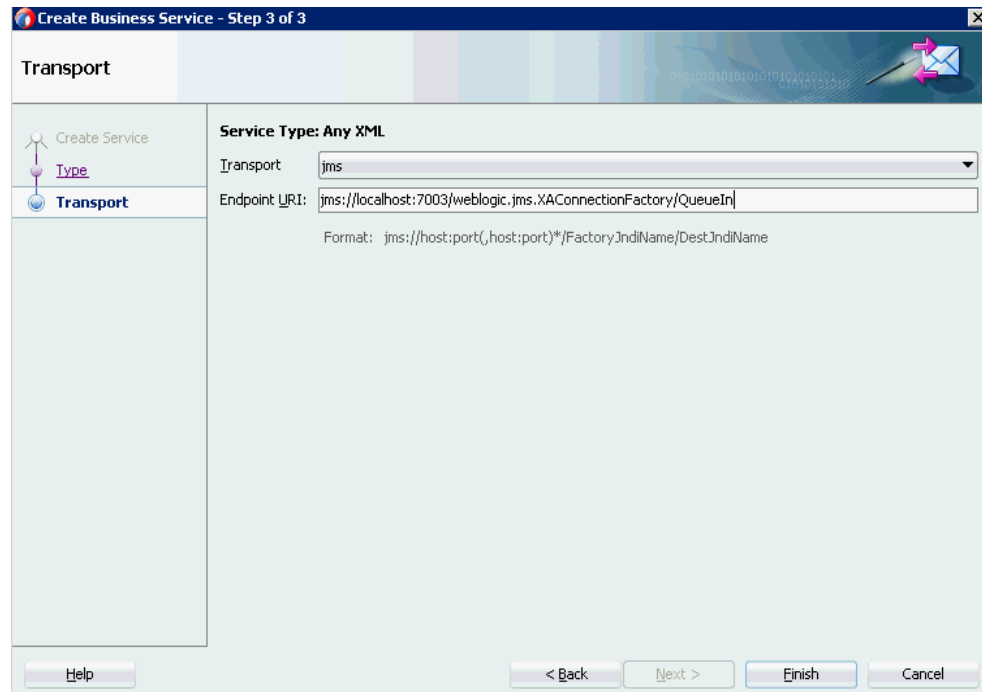


- c. 表示される「タイプ」ウィンドウで「任意の XML」を選択し、「次へ」をクリックします。

「トランスポート」ウィンドウが表示されます。

- d. 図 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。

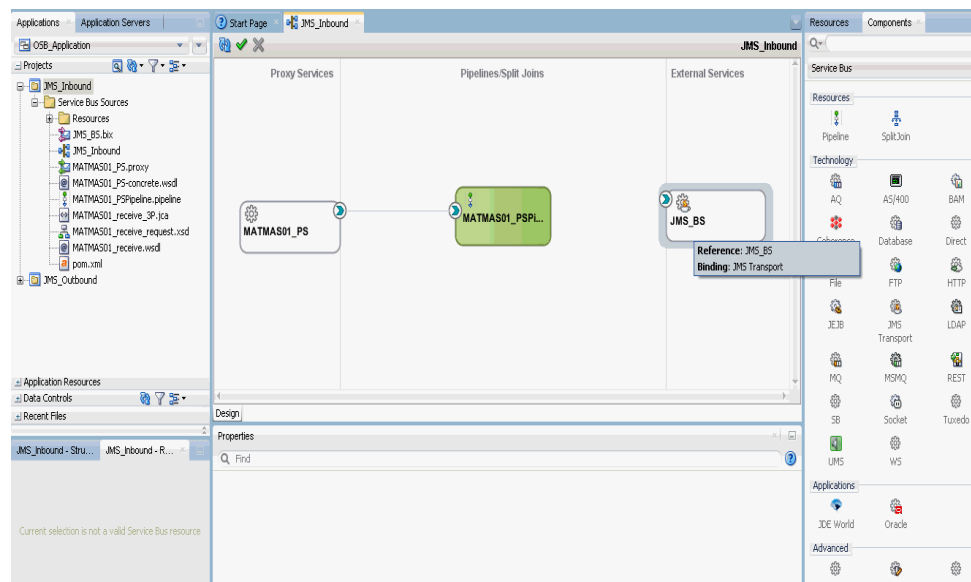
図 10-88 「トランスポート」ウィンドウ



JMS ビジネス・サービスが作成されて表示されます。

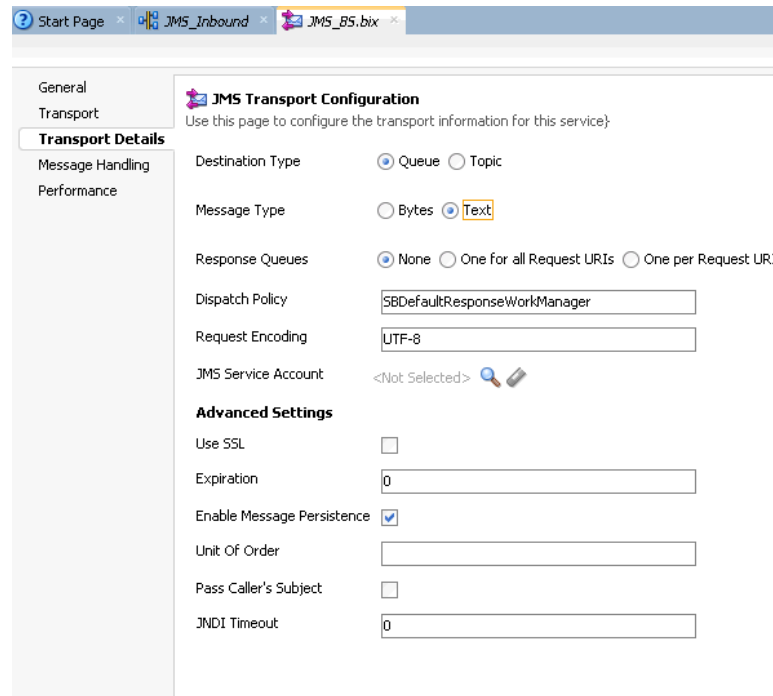
- e. 図 10-89 に示すように、「JMS_BS」をダブルクリックします。

図 10-89 JMS ビジネス・サービス



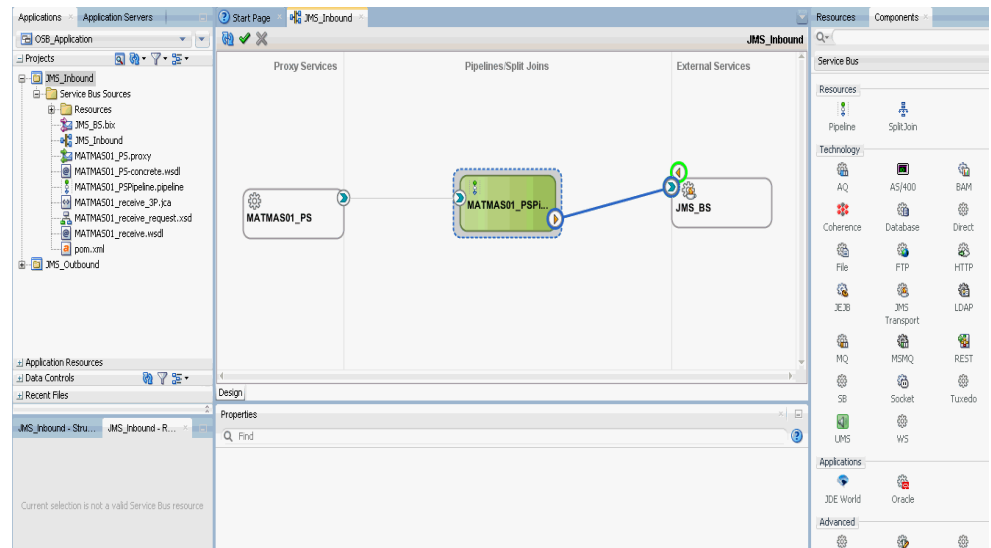
- f. 図 10-90 に示すように、表示されるビジネス・サービスの構成ページの「トランスポートの詳細」タブで、次のパラメータを指定します。

図 10-90 JMS トランスポート構成



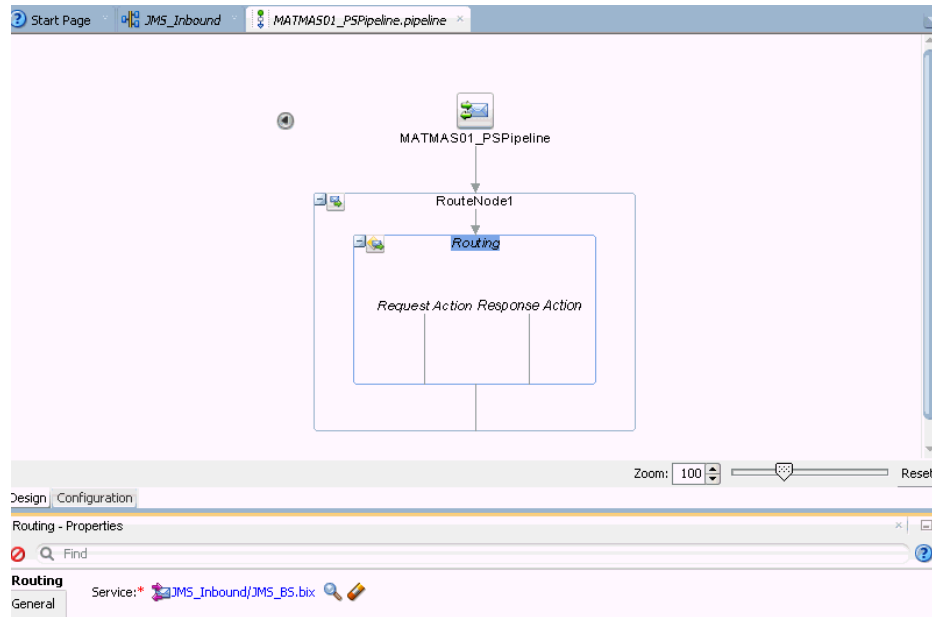
- g. 「宛先タイプ」セクションで、「キュー」を選択します。
- h. 「メッセージ・タイプ」セクションで、「テキスト」を選択します。
6. ビジネス・サービスの構成ページを保存して閉じます。
7. 図 10-91 に示すように、「パイプライン」(xxxx_PSPipeline など) と JMS ビジネス・サービス (JMS_BS など) との間に接続を作成します。

図 10-91 「構成」ページ



8. 「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、パイプライン構成ページが表示されます。

図 10-92 パイプライン構成

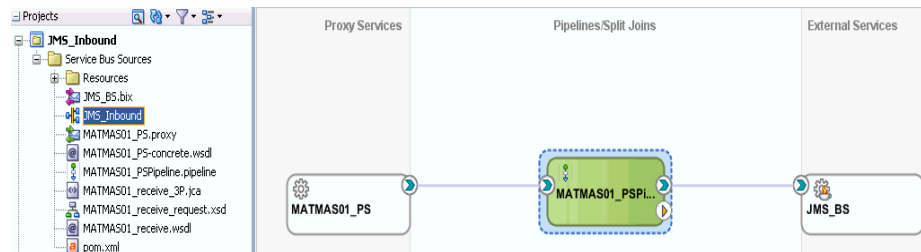


9. 詳細が正しく構成されていることを確認し、パイプライン構成ページを保存して閉じます。

コンポジット・エディタ・ウィンドウに戻ります。

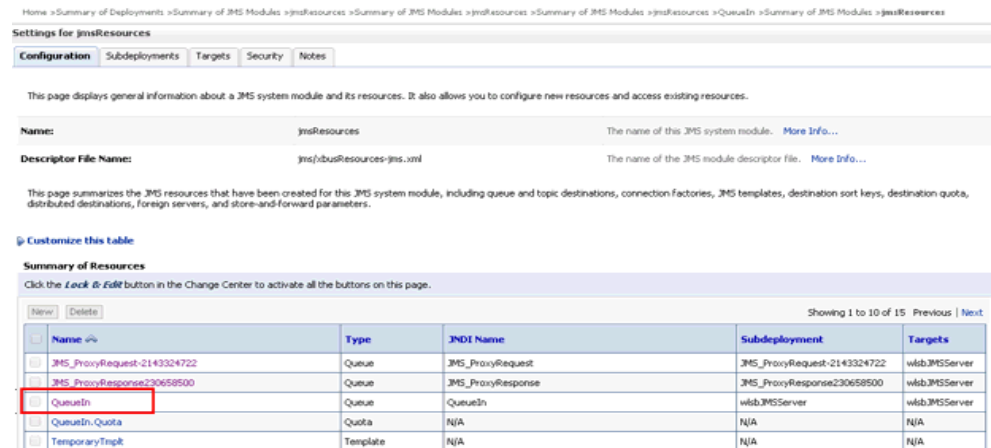
10. 図 10-93 に示すように、メニュー・バーの「すべて保存」をクリックして、OSB JMS プロセスを保存します。

図 10-93 「すべて保存」アイコン



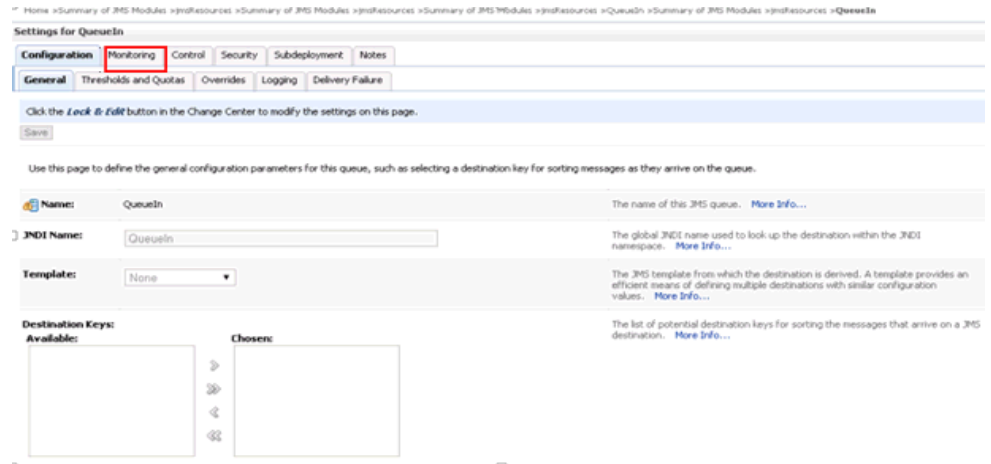
11. OSB JMS インバウンド・プロセスをデプロイします。詳細については、10-30 ページの 10.2.3 項「OSB インバウンド・プロセスのデプロイ」を参照してください。
12. プロセスが正常にデプロイされたら、イベント・メッセージをトリガーします。詳細については、6-54 ページの 6.5.5 項「SAP R/3 でのイベントのトリガー」を参照してください。
13. Oracle WLS コンソールにログオンします。
14. Oracle WLS コンソールで「サービス」を展開し、「メッセージング」をクリックして、「JMS モジュール」を選択してから「jmsResources」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



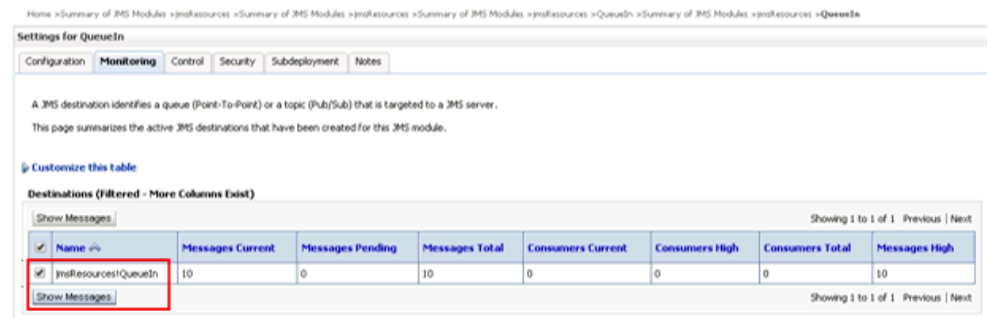
16. 図 10-95 に示すように、「モニタリング」タブをクリックします。

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
ID:<785570.1400633930707.0>		Tue May 20 17:58:50 PDT 2014	visible	Persistent	13320
ID:<785570.1400633930774.0>		Tue May 20 17:58:50 PDT 2014	visible	Persistent	4473
ID:<785570.1400633930794.0>		Tue May 20 17:58:50 PDT 2014	visible	Persistent	12961
ID:<785570.1400633932186.0>		Tue May 20 17:58:52 PDT 2014	visible	Persistent	33583
ID:<785570.1400633932958.0>		Tue May 20 17:58:52 PDT 2014	visible	Persistent	33548
ID:<785570.1400633933037.0>		Tue May 20 17:58:53 PDT 2014	visible	Persistent	25156
ID:<785570.1400633933871.0>		Tue May 20 17:58:53 PDT 2014	visible	Persistent	18775
ID:<785570.1400633934379.0>		Tue May 20 17:58:54 PDT 2014	visible	Persistent	16671
ID:<785570.1400633934420.0>		Tue May 20 17:58:54 PDT 2014	visible	Persistent	14488
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:

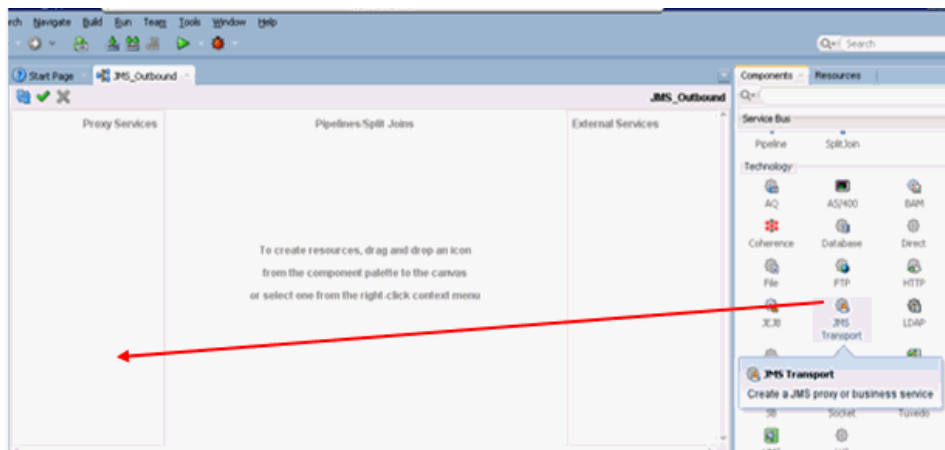
```
<?xml version="1.0" encoding="UTF-8"?>
<MATHAS01 tid="AC1301D72A27537B3D4600B9" xmlns="urn:sap-com:document:sap:idoc">
  <IDOC BEGIN="1">
    <EDI_DC40 SEGMENT="1">
      <TABNAM>EDI_DC40</TABNAM>
      <MANDT>800</MANDT>
      <DOCNUM>000000005013447</DOCNUM>
    </EDI_DC40 SEGMENT="1">
  </IDOC 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 構成用) 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。詳細については、6-8 ページの 6.4.1 項「リクエスト/レスポンス型のサービス用の 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。詳細については、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、7-14 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。詳細については、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。

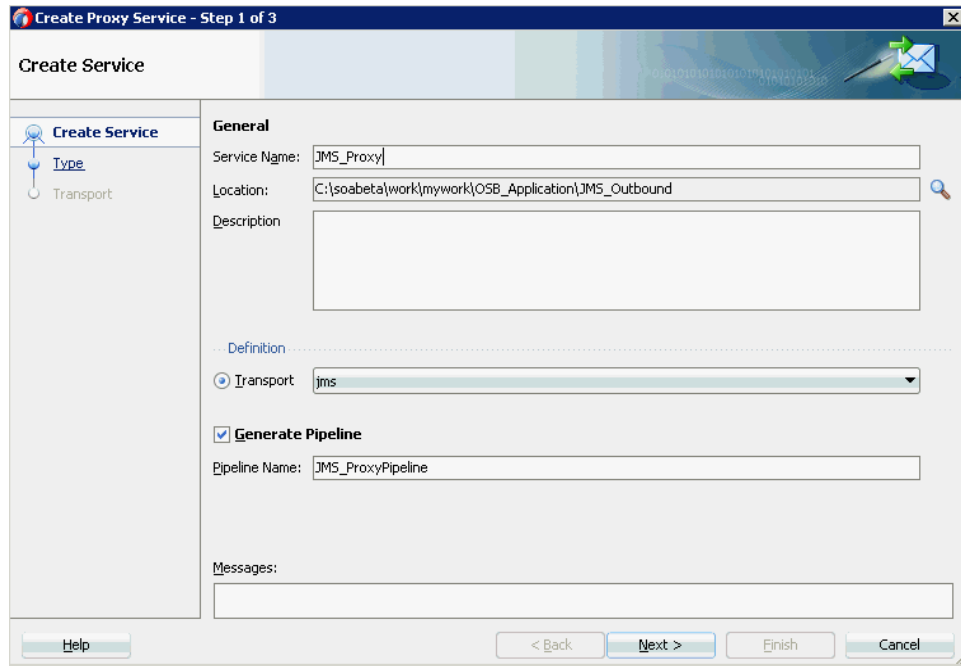
図 10-99 JMS トランスポート・コンポーネント



「ビジネス・サービスの作成」ダイアログが表示されます。

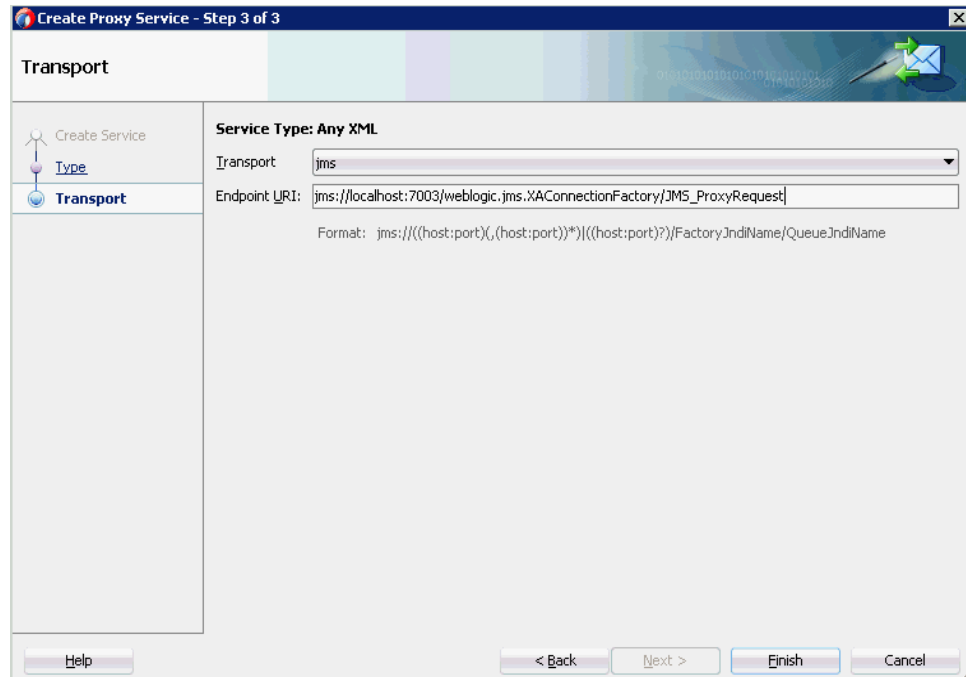
- b. 「サービス名」フィールドに、プロキシ・サービスに使用する任意の名前 (JMS_Proxy など) を入力します。デフォルトでは、パイプラインの生成が選択されています。
- c. 図 10-100 に示すように、「次へ」をクリックします。

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、「次へ」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。

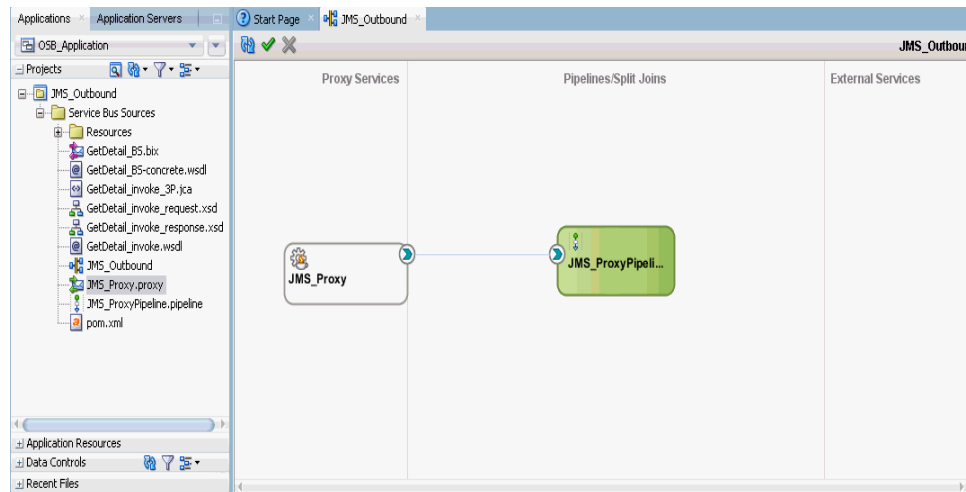
図 10-101 「トランスポート」ウィンドウ



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。

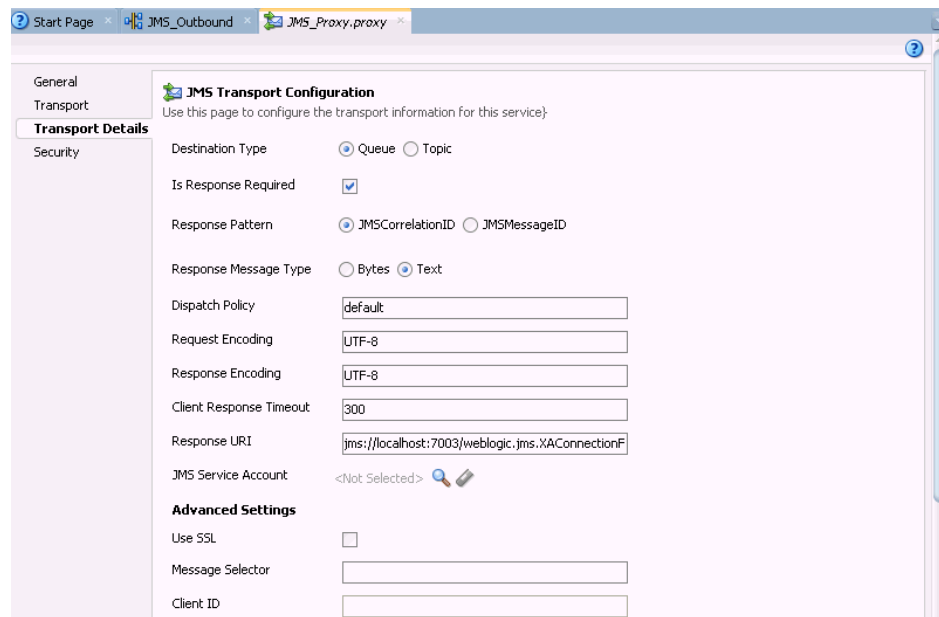
- f. 図 10-102 に示すように、作成されたプロキシ・サービス (JMS_Proxy など) をダブルクリックします。

図 10-102 JMS プロキシ・サービス



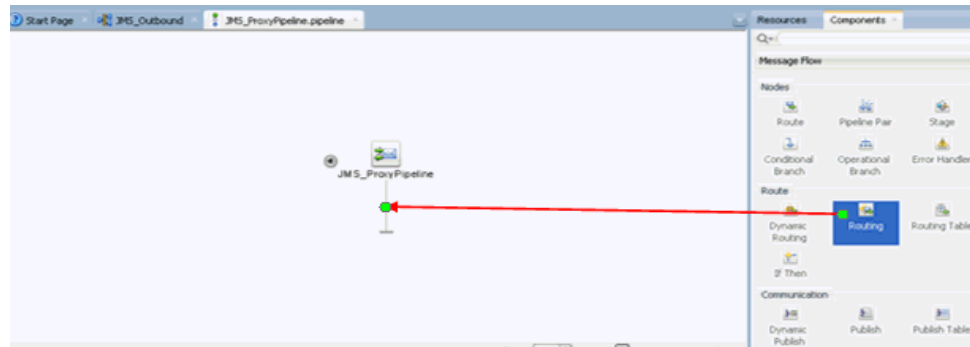
- g. 図 10-103 に示すように、表示される「プロキシ・サービス」の構成ページの「トランスポートの詳細」タブで、次のパラメータを指定します。

図 10-103 JMS トランスポート構成



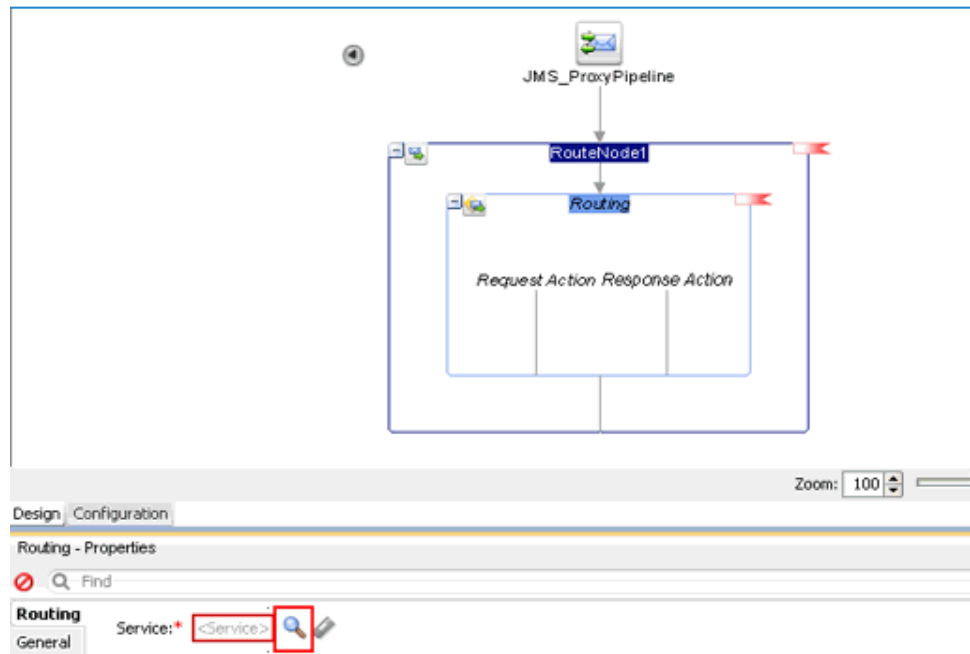
- h. 「宛先タイプ」セクションで、「キュー」を選択します。
 - i. 「レスポンスが必要」チェック・ボックスを選択します。
 - j. 「レスポンス・メッセージ・タイプ」セクションで、「テキスト」を選択します。
 - k. 「レスポンス URI」フィールドに、JMS トランスポート構成で使用したエンドポイント URI を入力し、「リクエスト」を「レスポンス」に変更します。
(`jms://localhost:7003/weblogic.jms.XAConnectionFactory/JMS_ProxyResponse` など)。
6. プロキシ・サービスの構成ページを保存して閉じます。
 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



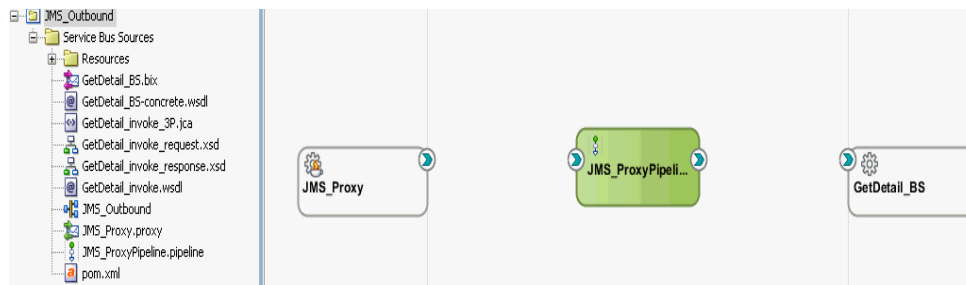
- c. 図 10-105 に示すように、パイプライン構成ページで「ルーティング」を選択し、プロパティのルーティング・ペインの「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 10-105 サービスの参照



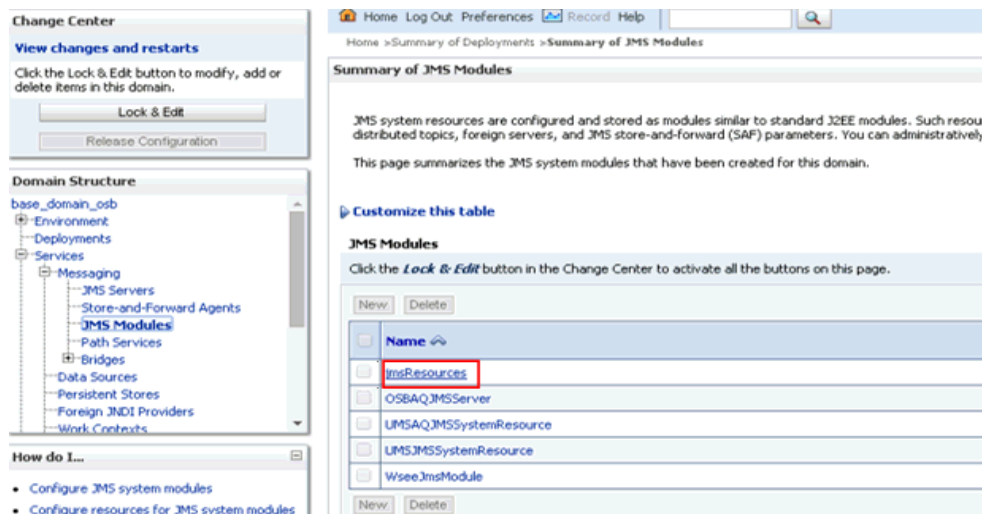
- d. 表示されるリソース・チューザ・ウィンドウで、WSDL ベースのビジネス・サービス (xxxxx_BS.bix など) を選択して、「OK」をクリックします。
パイプライン構成ページに戻ります。
- e. パイプライン構成ページを保存して閉じます。
コンポジット・エディタ・ウィンドウに戻ります。
- f. 図 10-106 に示すように、メニュー・バーの「すべて保存」をクリックして、OSB JMS プロセスを保存します。

図 10-106 「トランスポート」ウィンドウ



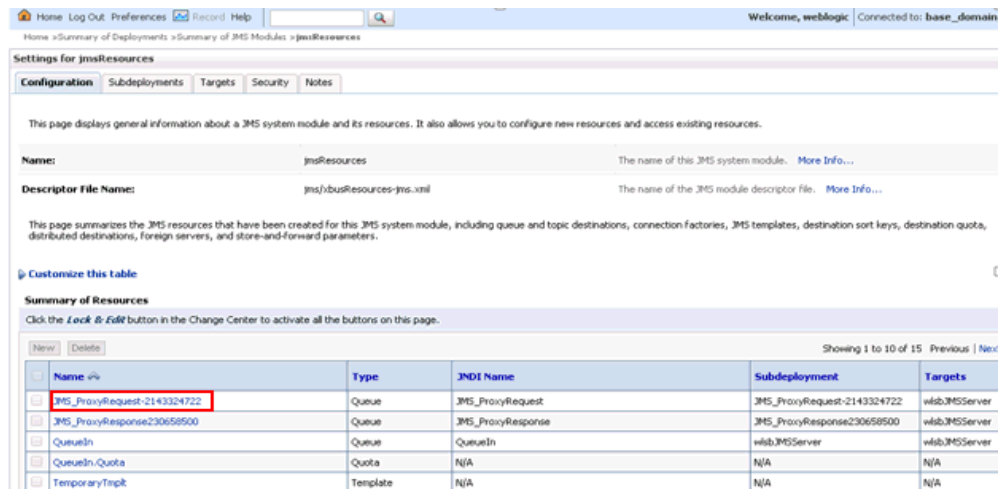
8. OSB JMS アウトバウンド・プロセスをデプロイします。詳細については、10-17 ページの 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」を参照してください。
9. プロセスが正常にデプロイされたら、Oracle WLS コンソールにログオンします。
10. 図 10-107 に示すように、Oracle WLS コンソールで「サービス」を展開し、「メッセージング」をクリックして、「JMS モジュール」を選択してから「jmsResources」をクリックします。

図 10-107 JMS リソース



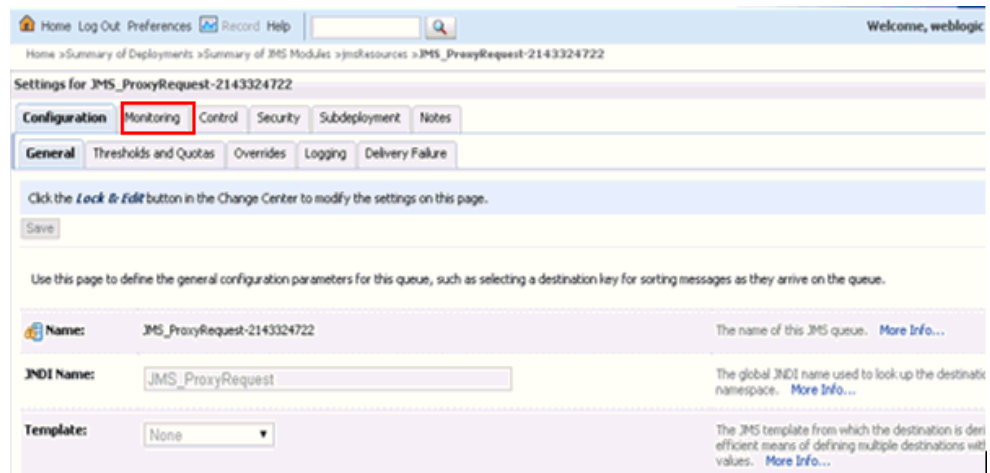
11. 図 10-108 に示すように、適切なリクエスト・リンク (JMS_ProxyRequest など) をクリックします。

図 10-108 JMS_ProxyRequest リンク



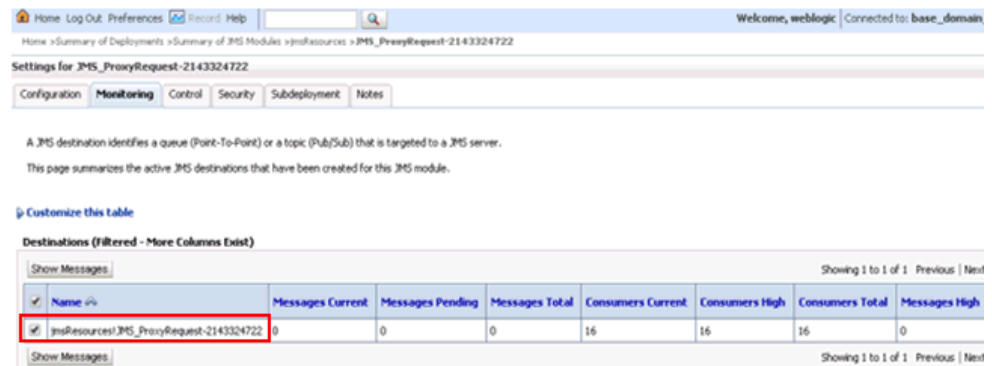
12. 図 10-109 に示すように、「モニタリング」タブをクリックします。

図 10-109 「モニタリング」タブ



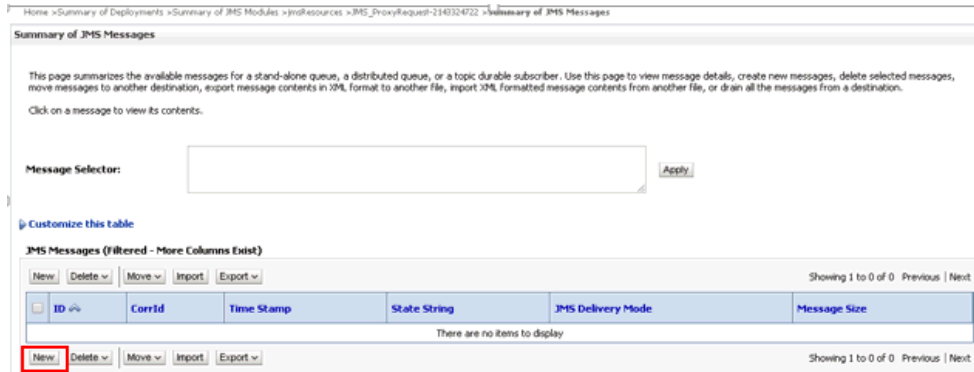
13. 図 10-110 に示すように、チェック・ボックスを選択して「メッセージの表示」ボタンをクリックします。

図 10-110 「メッセージの表示」ボタン



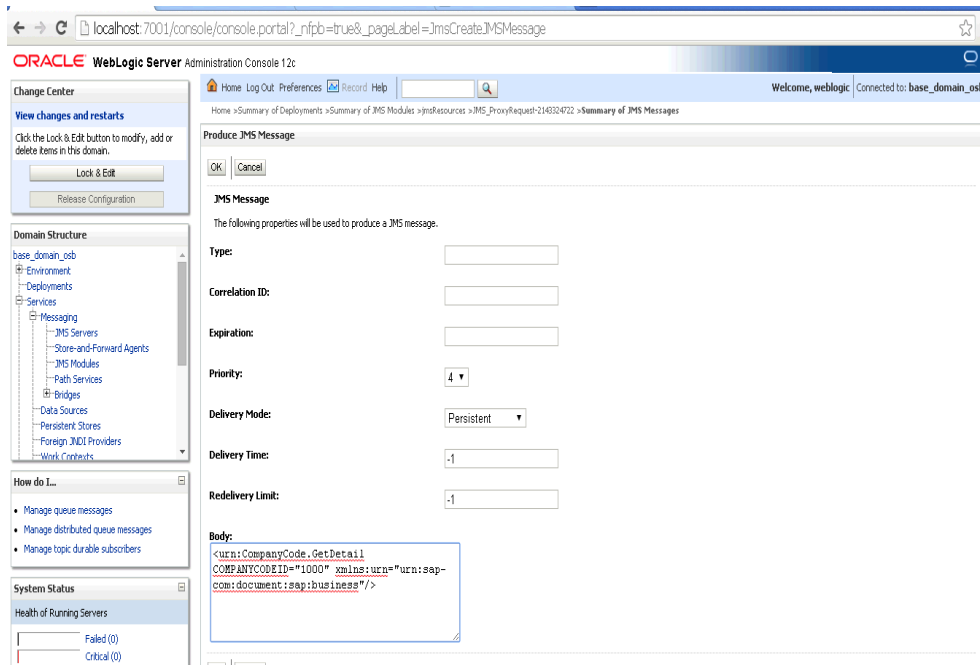
14. 図 10-111 に示すように、「次へ」をクリックします。

図 10-111 JMS メッセージ



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



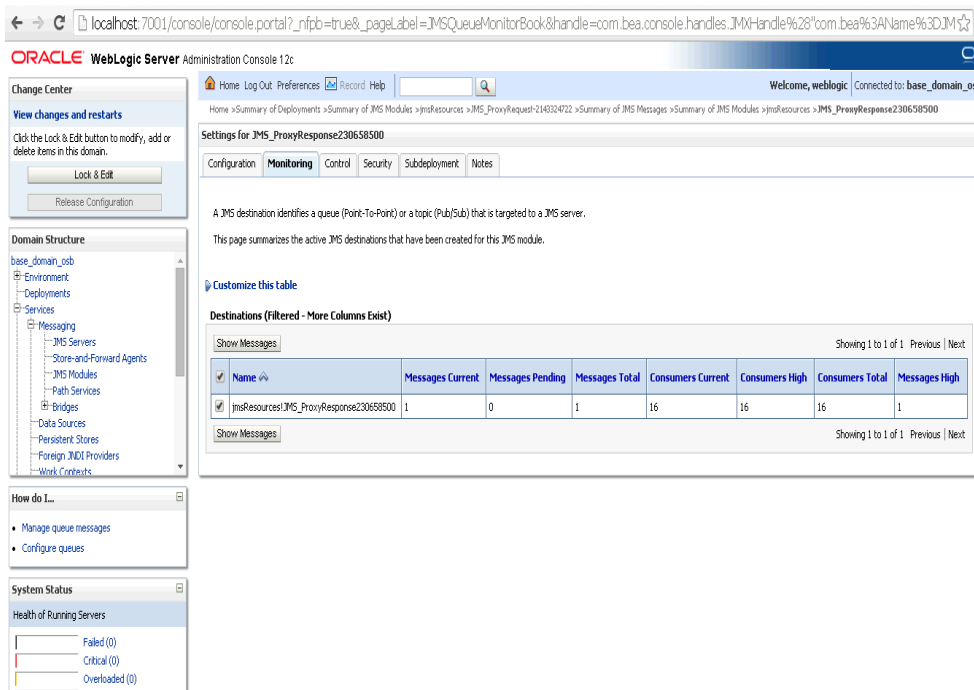
16. Oracle WLS コンソールで「サービス」を展開し、「メッセージング」をクリックして、「JMS モジュール」を選択してから「jmsResources」をクリックします。

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



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

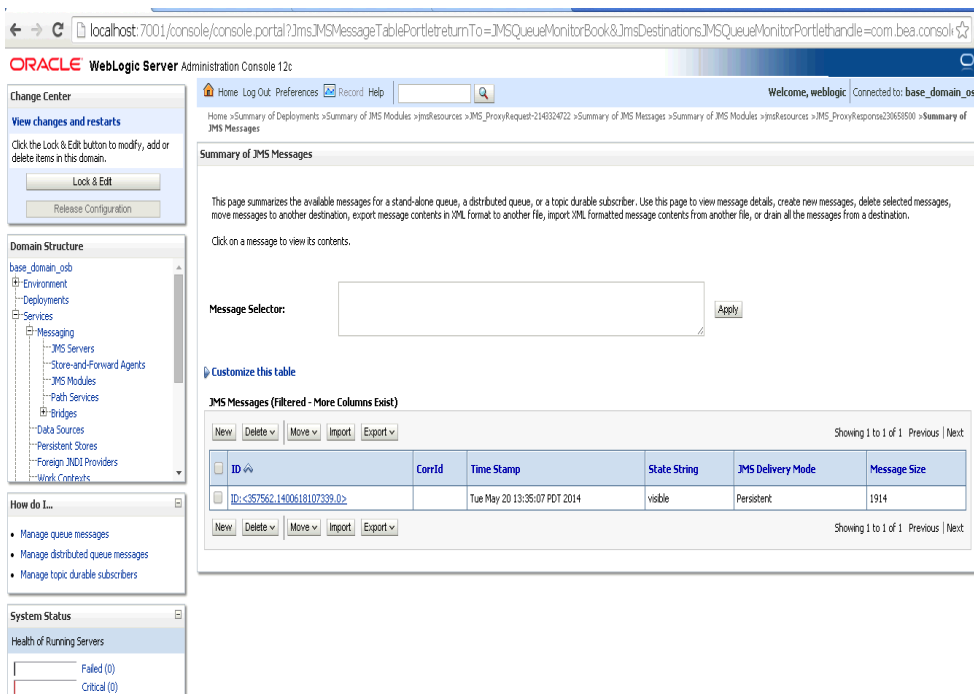
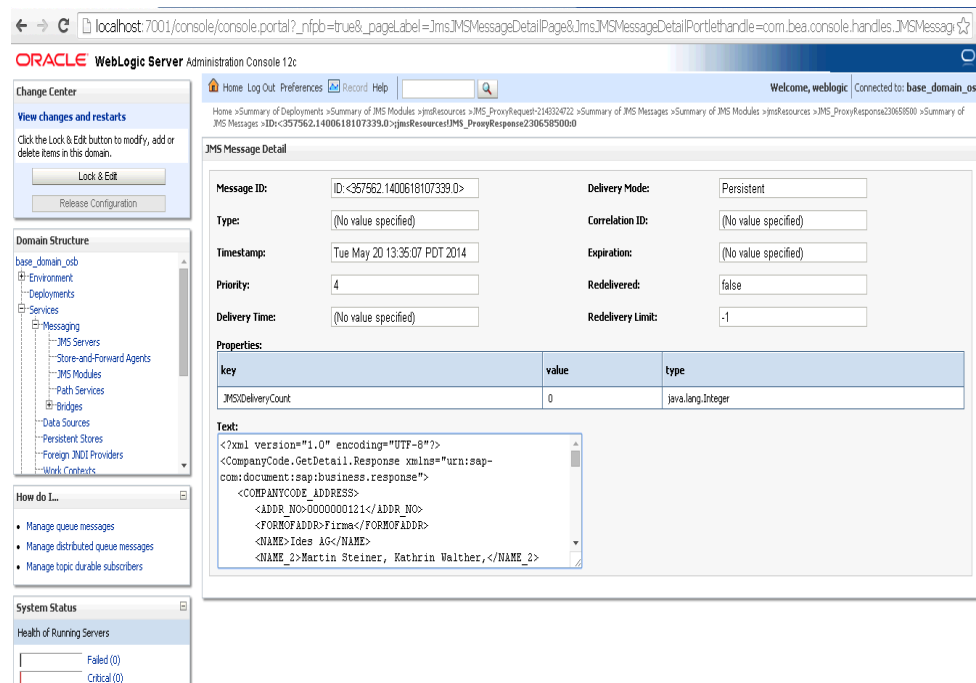


図 10-115 に示すように、「テキスト」フィールドの下にレスポンス・ドキュメントが表示されます。

図 10-115 レスポンス・ドキュメント

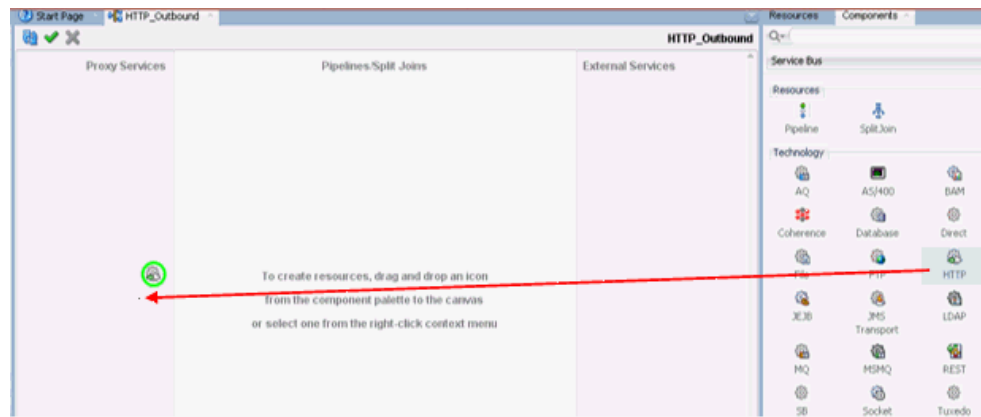


10.7 JDeveloper を使用した HTTP アウトバウンド・プロセスの構成 (J2CA 構成)

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 構成用) 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・エクステンサを使用し、各 WSDL ファイルを生成する必要があります。詳細については、6-8 ページの [6.4.1 項「リクエスト/レスポンス型のサービス用の 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。詳細については、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、7-14 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。

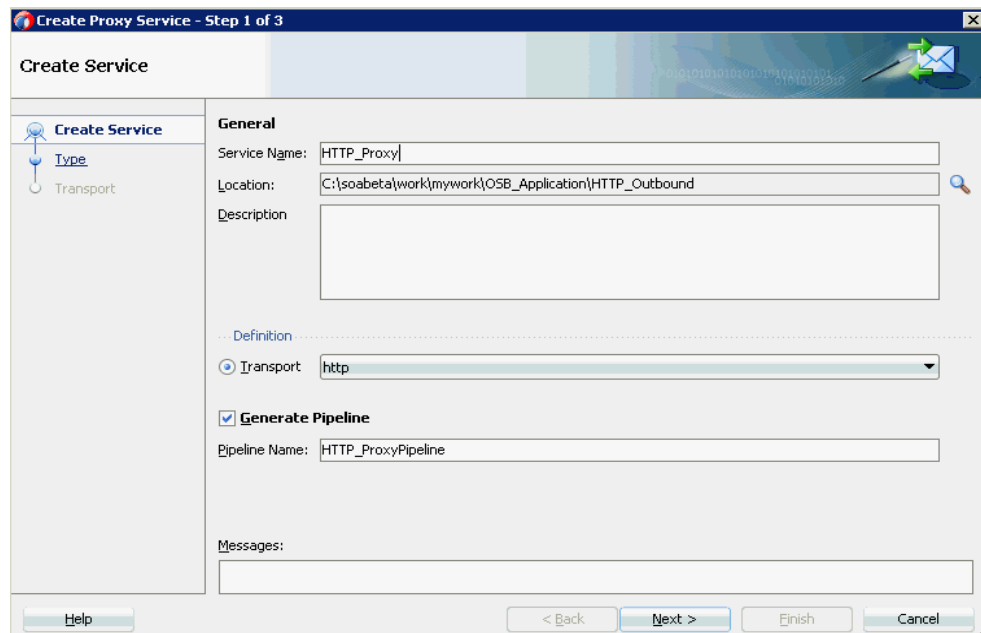
図 10-116 「HTTP」コンポーネント



プロキシ・サービスの作成ダイアログが表示されます。

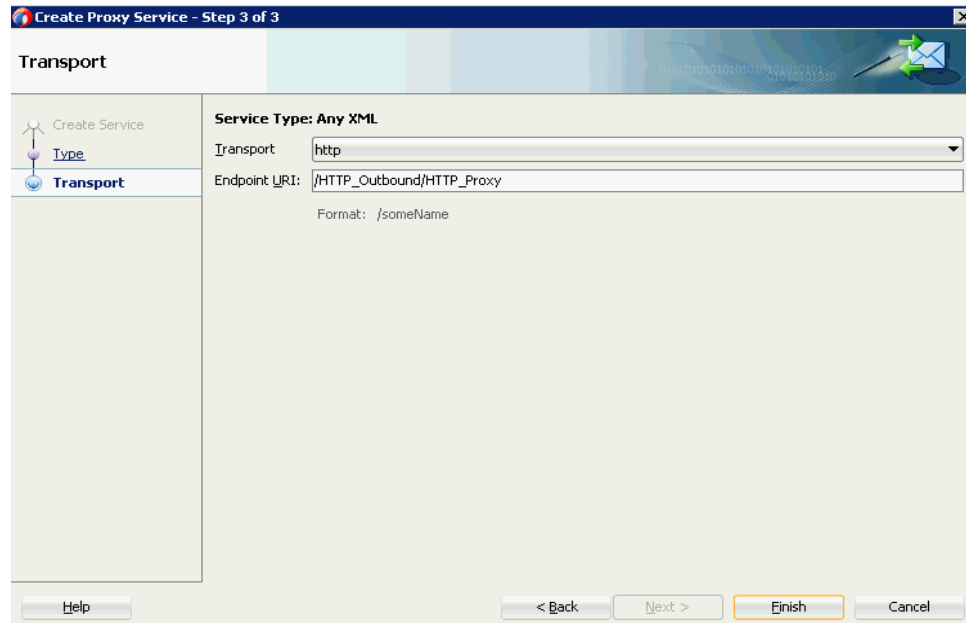
- b. 「サービス名」フィールドに、プロキシ・サービスに使用する任意の名前 (HTTP_Proxy など) を入力します。デフォルトでは、パイプラインの生成が選択されています。
- c. 図 10-117 に示すように、「次へ」をクリックします。

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、「次へ」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、デフォルト値を選択したままにし、「終了」をクリックします。

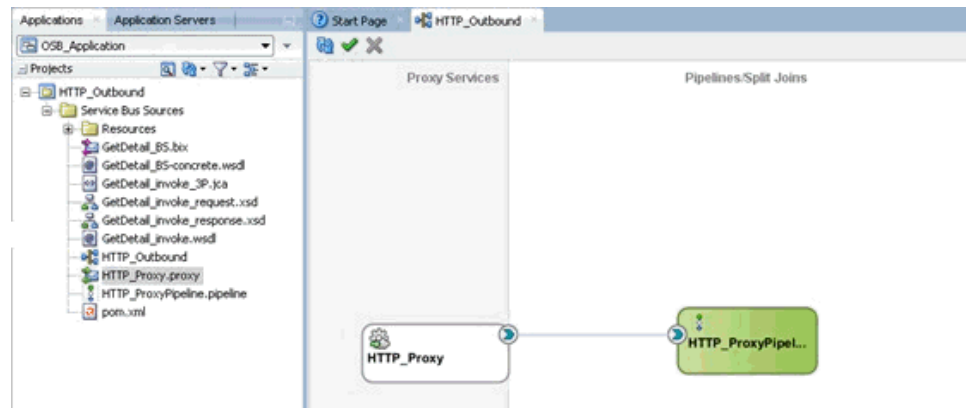
図 10-118 「トランスポート」 ウィンドウ



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。

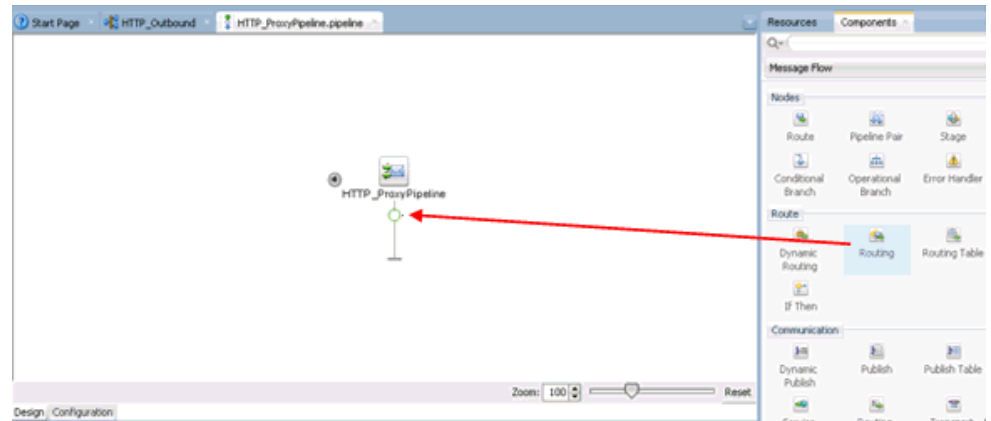
図 10-119 プロキシ・サービス



パイプライン構成ページが表示されます。

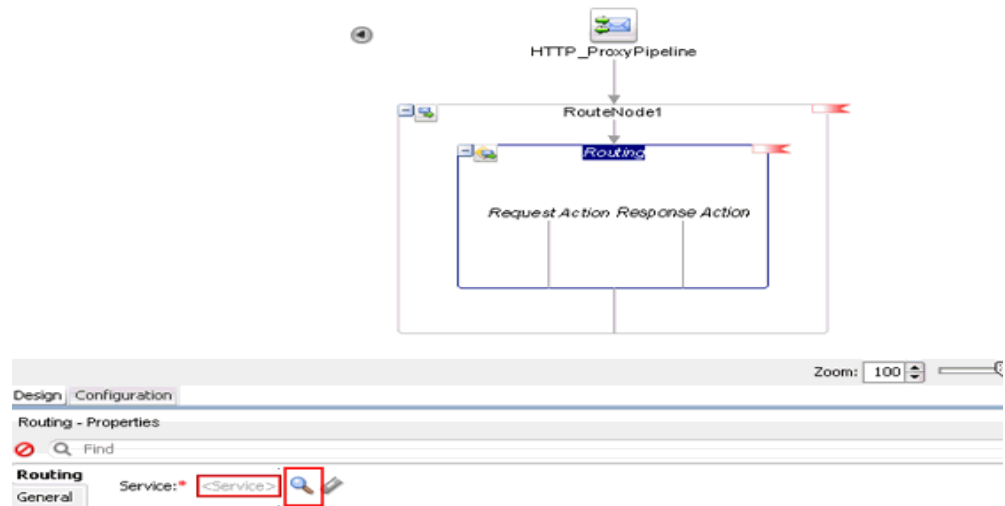
5. ルーティング・ルールを構成し、次の手順を続行します。
 - a. 図 10-120 に示すように、「ルーティング」コンポーネントを「ルート」セクションからパイプライン (HTTP_ProxyPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 10-120 「ルーティング」 コンポーネント



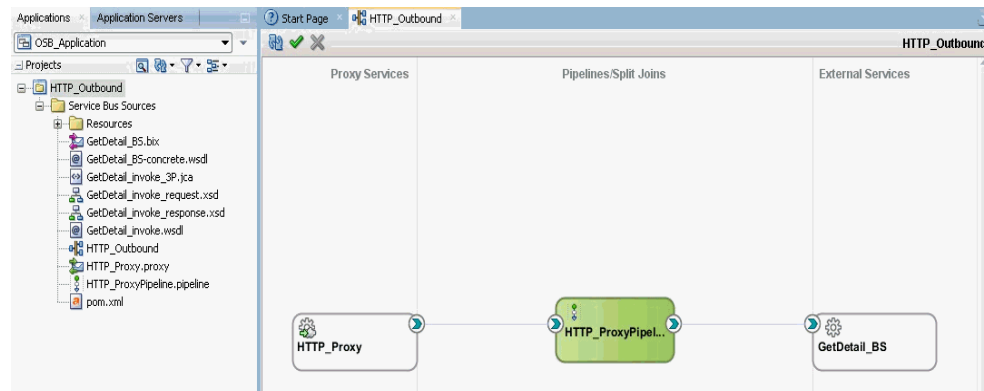
- b. 図 10-121 に示すように、パイプライン構成ページで「ルーティング」を選択し、プロパティのルーティング・ペインの「サービス」フィールドの右側にある「参照」アイコンをクリックします。

図 10-121 サービスの参照



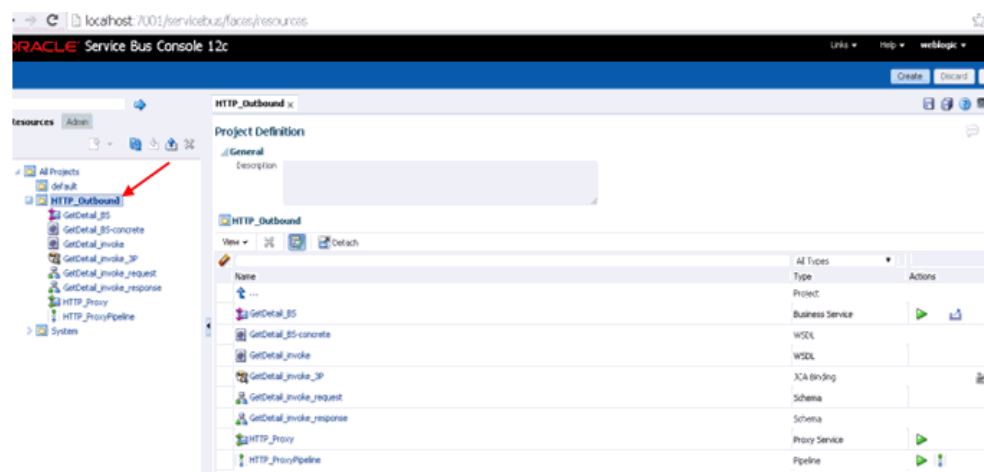
- c. 表示されるリソース・チューザ・ウィンドウで、WSDL ベースのビジネス・サービス (xxxxx_BS.bix など) を選択して、「OK」をクリックします。
パイプライン構成ページに戻ります。
- d. パイプライン構成ページを保存して閉じます。
コンポジット・エディタ・ウィンドウに戻ります。
- e. 図 10-122 に示すように、メニュー・バーの「すべて保存」をクリックして、OSB HTTP プロセスを保存します。

図 10-122 「トランスポート」ウィンドウ



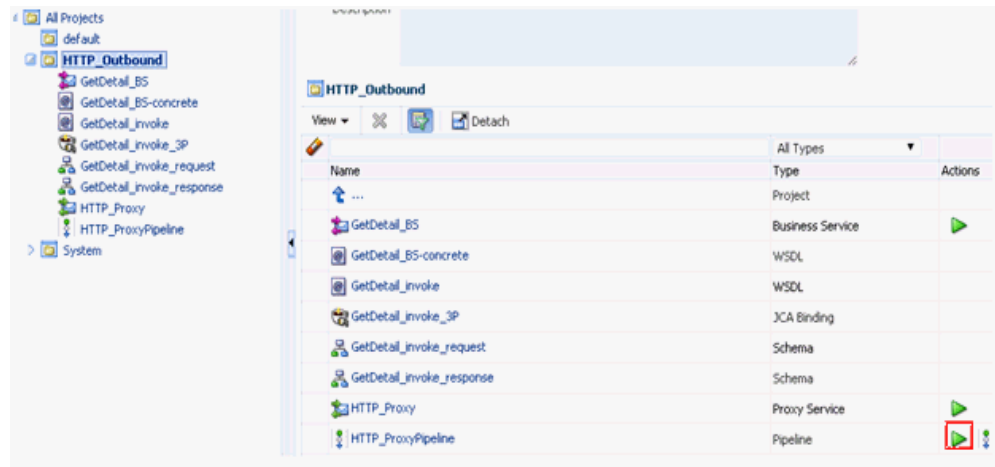
6. OSB HTTP アウトバウンド・プロセスのデプロイ詳細については、10-17 ページの 10.1.3 項「OSB アウトバウンド・プロセスのデプロイ」を参照してください。
7. プロセスが正常にデプロイされたら、Service Bus コンソールにログオンします。
8. 図 10-123 に示すように、Service Bus コンソールで、デプロイされた HTTP アウトバウンド・プロジェクト (HTTP_Outbound など) をクリックします。

図 10-123 Service Bus コンソール



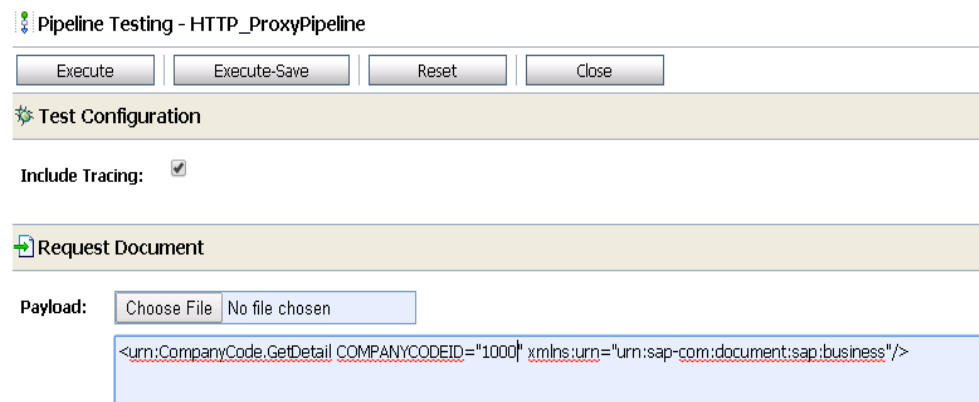
9. 図 10-124 に示すように、作成されたパイプラインの OSB コンソールのテスト・アイコンをクリックします。

図 10-124 OSB コンソールのテスト・アイコン



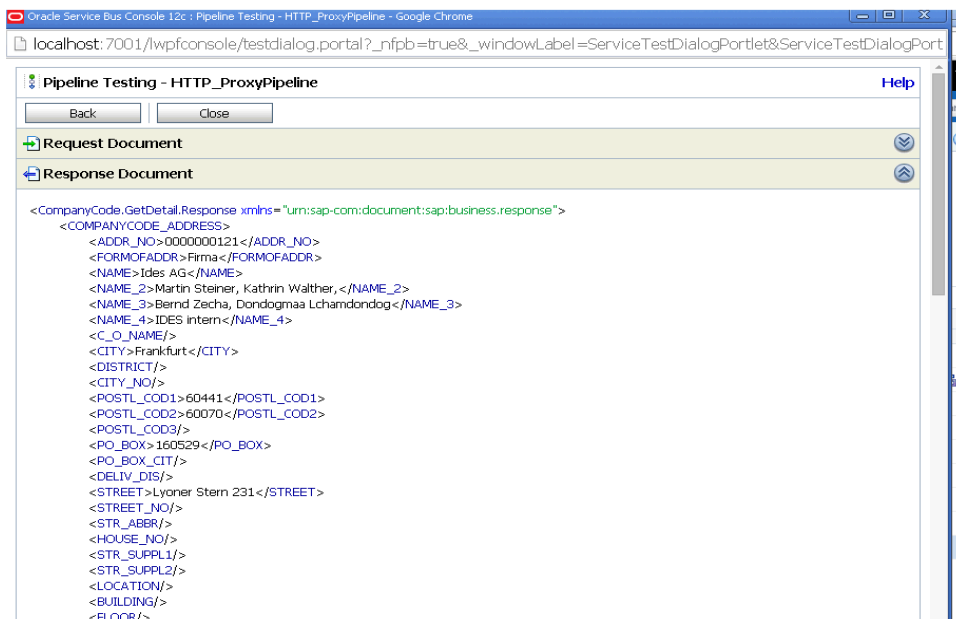
10. 図 10-125 に示すように、OSB コンソールのテスト・ページで、入力 XML を指定して「実行」ボタンをクリックします。

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。この章の内容は次のとおりです。

- 11.1 項「ロギング機能の構成」
- 11.2 項「診断機能の構成」
- 11.3 項「SOA デバッグ機能の構成」
- 11.4 項「例外フィルタの構成」
- 11.5 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」
- 11.6 項「JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング」
- 11.7 項「テキストからの Unicode 文字のフィルタリング」

11.1 ロギング機能の構成

Oracle 12c (12.2.1.0.0) では、J2CA および BSE アダプタ・ログは、次の場所で入手可能な `{server-name}-diagnostic.log` ファイルの Oracle ログで更新されます。

```
<ORACLE_HOME>\user_projects\domains\base_domain\servers\<server_Name>\logs
```

注意： J2CA 用アプリケーション・エクスプローラのログ・ファイルは、`<ADAPTER_HOME>\config\xxxxxx\log` フォルダの下に作成されます。xxxxxx はアプリケーション・エクスプローラで作成された J2CA 構成の名前です。アプリケーション・エクスプローラでの各 J2CA 構成には、名前付き J2CA 構成フォルダの下に対応するログ・フォルダが存在します。

この項では、ロギング機能の構成方法について説明します。次のトピックについて説明します。

- 11.1.1 項「J2CA コネクタ・アプリケーションのログ・ファイル管理の構成」
- 11.1.2 項「Business Services Engine (BSE) のログ・ファイル管理の構成」

11.1.1 J2CA コネクタ・アプリケーションのログ・ファイル管理の構成

J2CA コネクタ・アプリケーションのログ・ファイル管理は、次の場所で定義されるログ出力によって制御されます。

```
<ORACLE_HOME>\user_projects\domains\base_
domain\config\fmwconfig\servers\${server-name}\logging.xml
```

新しいログ出力が EM コンソールから管理される場合、このファイルに追加する必要があります。

次に例を示します。

```
<logger name='oracle.soa.adapter.iwaf' level='NOTIFICATION:1'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.classloader' level='NOTIFICATION:1'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.connection' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.connection.IAEAdapter'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.connection.Sample'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.connection.MySAP' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.inbound' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.inbound.IAEAdapter'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.inbound.Sample' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.inbound.MySAP' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.outbound' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.outbound.IAEAdapter'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.outbound.Sample' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.outbound.MySAP' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.transaction' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.transaction.IAEAdapter'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.transaction.Sample'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.transaction.MySAP'
useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.IAEAdapter' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.Sample' useParentHandlers='true' />
<logger name='oracle.soa.adapter.iwaf.MySAP' useParentHandlers='true' />
```

これにより、「oracle.soa.adapter.iwaf」の下にあるすべてのログ出力のロギング・レベルが、Oracle によるデフォルトの設定レベルである「NOTIFICATION:1 (INFO)」に設定されます。

次のステップを実行して、すべてのログ出力のロギング・レベルを EM コンソールから構成することもできます。

1. 構成済の 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 を入力して、Oracle WebLogic Server Enterprise Manager Console を起動します。

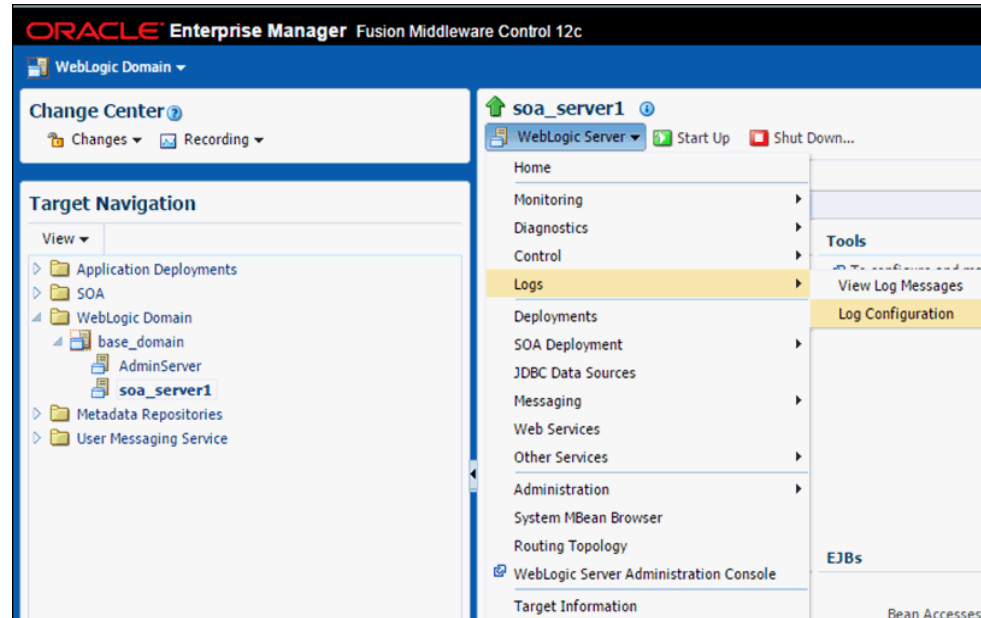
```
http://host name:port/em
```

ここで、host name は Oracle WebLogic Server が稼働しているシステムの名前で、port は稼働している Oracle WebLogic Server のポートです。Oracle WebLogic Server のデフォルト・ポートは 7001 です。ただし、この値はインストールによって異なる場合があります。

3. 管理者権限を持つアカウントを使用して、Oracle WebLogic Server 管理コンソールにログインします。

- 「ターゲット・ナビゲーション」ペインの下で、「WebLogic ドメイン」をクリックし、「ドメインが作成されました」を選択して、適切なサーバー（「管理対象サーバー」または「統合サーバー」）をクリックします。
- 図 11-1 に示すように、「サーバー」ペインで、「WebLogic Server」を展開し、「ログ」を選択して、「ログ構成」をクリックします。

図 11-1 「ログ構成」オプション



- 図 11-2 に示すように、「ログ・レベル」タブの下で、「oracle.soa.adapter.iwaf」が表示されるまで Oracle ルート・ログ出力を展開します。

図 11-2 「ログ・レベル」タブ

Log Configuration
Use this page to configure basic and advanced log configuration settings.

Log Levels | Log Files | QuickTrace

This page allows you to configure the log level for both persistent loggers and active runtime loggers. Persistent loggers are loggers that are saved in a configuration file and are active when the component is started. The log levels for these loggers are persisted across component restarts. Runtime loggers are automatically created during runtime and become active when the component is exercised. For example, oracle.j2ee.ejb.deployment.Logger is a runtime logger that becomes active when an EJB module is deployed. Log levels for runtime loggers are not persisted across component restarts.

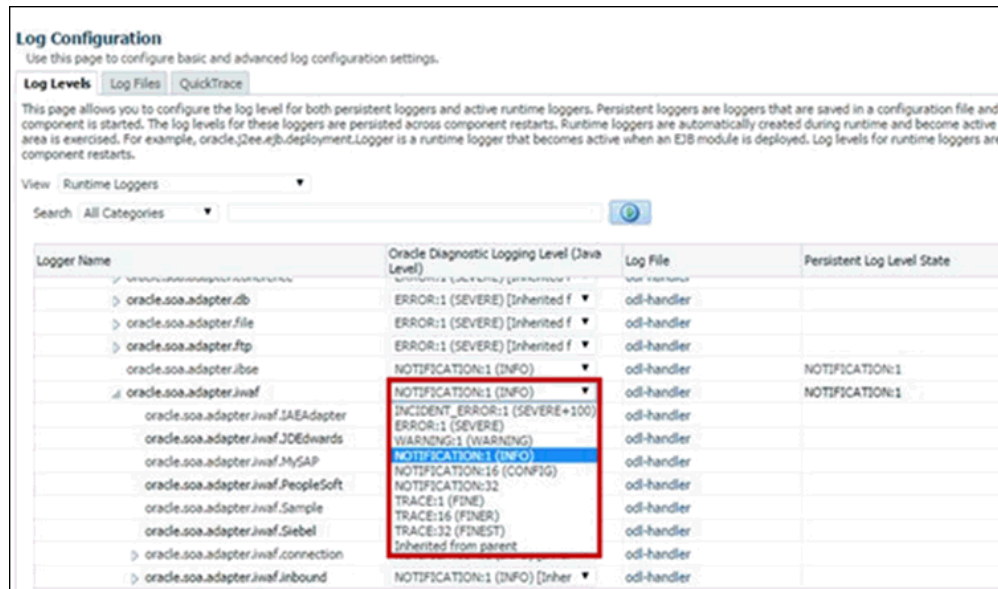
View: Runtime Loggers

Search: All Categories

Logger Name	Oracle Diagnostic Logging Level (Java Level)	Log File	Persistent Log Level State
oracle.soa.adapter.db	ERROR:1 (SEVERE) [Inherited from oracle.soa.adapter]	odl-handler	
oracle.soa.adapter.file	ERROR:1 (SEVERE) [Inherited from oracle.soa.adapter]	odl-handler	
oracle.soa.adapter.ftp	ERROR:1 (SEVERE) [Inherited from oracle.soa.adapter]	odl-handler	
oracle.soa.adapter.ibse	NOTIFICATION:1 (INFO)	odl-handler	NOTIFICATION:1
oracle.soa.adapter.iwaf	NOTIFICATION:1 (INFO)	odl-handler	NOTIFICATION:1
oracle.soa.adapter.iwaf.IAEdapter	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	
oracle.soa.adapter.iwaf.JDEdwards	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	
oracle.soa.adapter.iwaf.MySAP	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	
oracle.soa.adapter.iwaf.PeopleSoft	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	
oracle.soa.adapter.iwaf.Sample	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	
oracle.soa.adapter.iwaf.Siebel	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	
oracle.soa.adapter.iwaf.connection	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	
oracle.soa.adapter.iwaf.inbound	NOTIFICATION:1 (INFO) [Inherited from oracle.soa.adapter.iwaf]	odl-handler	

7. 図 11-3 に示すように、「Oracle Diagnostic Logging レベル (Java レベル)」列で、「oracle.soa.adapter.iwaf」ドロップダウン・リストから必要なログ・レベルを選択します。

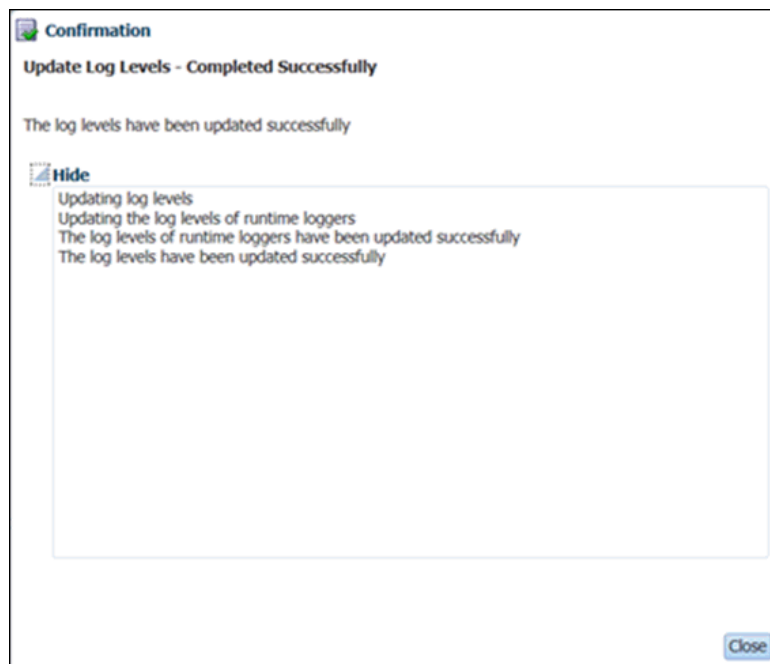
図 11-3 「Oracle Diagnostic Logging レベル (Java レベル)」列



8. 「適用」をクリックします。

図 11-4 に示すように、更新が正常に完了したことを示す確認メッセージが表示されます。

図 11-4 確認メッセージ



9. 「閉じる」をクリックします。

次の表は、「ログ・レベル」プロパティが `{server-name}-diagnostic.log` で EM コンソール内の対応する「ログ・レベル」プロパティ設定に基づいてどのように更新されるかを示しています。

表 11-1 「ログ・レベル」プロパティ設定

EM コンソールで設定された ログ・レベル	<code>{server-name}-diagnostic.log</code> で更新された ログ・レベル
ERROR:1	ERROR
WARNING:1	WARNING
NOTIFICATION:1	NOTIFICATION
NOTIFICATION:16	NOTIFICATION
NOTIFICATION:32	NOTIFICATION
TRACE:1	NOTIFICATION
TRACE:16	NOTIFICATION
TRACE:32	TRACE:32

注意： EM コンソールでログ・レベルを「TRACE:32」と設定すると、`{server-name}-diagnostic.log` でログ・レベルが「TRACE:32」と表示されたログ (アダプタ、アダプタから受信されたレスポンス、および他の追加詳細に渡された入力を表示) に「FINEST」の詳細が表示されます。

開発環境およびテスト環境の場合、「TRACE:32」は、すべてのログ詳細が表示される、望ましいログ・レベルです。本番環境の場合、望ましいログ・レベルは「ERROR」です。

ログ・メッセージはディスク・ファイルに書き込まれ、ファイル・パスは「ログ・ファイル」タブで参照できます。「ログ・ファイル」表の「ハンドラ名」は、「ログ・レベル」表の「ログ・ファイル」の名前に対応しています。図 11-5 に示すように、「oracle.soa.adapter」の下の階層にあるすべてのログ出力は、現在「odl-handler」によって処理されています。

図 11-5 「ログ構成」ペイン

The screenshot shows the 'Log Configuration' page with the 'Log Files' tab selected. It contains a table with the following data:

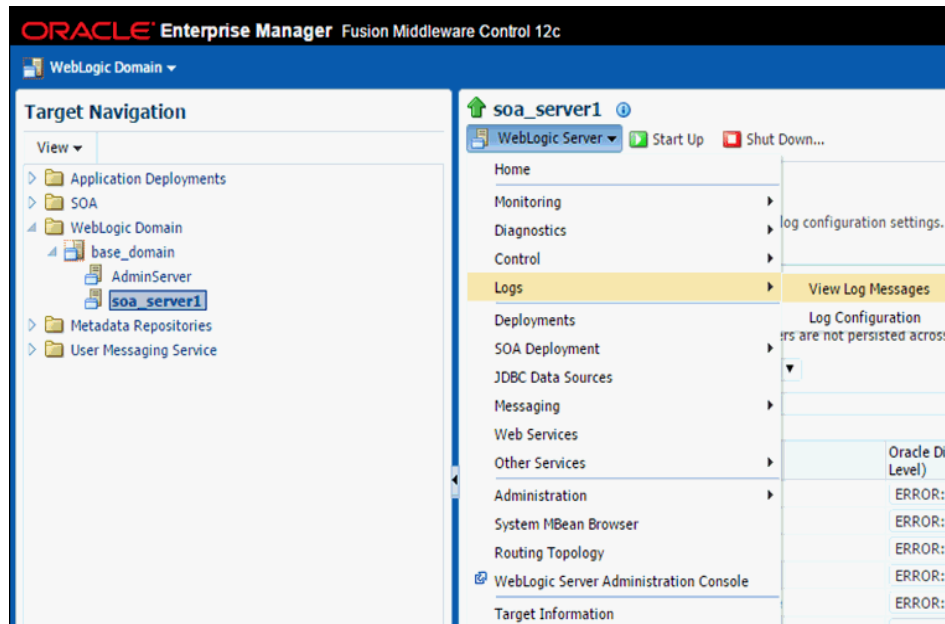
Handler Name	Log Path	Log File Format
em-log-handler	<code>\${domain.home}/servers/\${weblogic.Name}/sysman/log/emoms.log</code>	Oracle Diagnostics Logging
em-trc-handler	<code>\${domain.home}/servers/\${weblogic.Name}/sysman/log/emoms.trc</code>	Oracle Diagnostics Logging
odl-handler	<code>\${domain.home}/servers/\${weblogic.Name}/logs/\${weblogic.Name}-diagnostic.log</code>	Oracle Diagnostics Logging
owsm-message-ha...	<code>\${domain.home}/servers/\${weblogic.Name}/logs/owsm/msglogging/diagnostic.log</code>	Oracle Diagnostics Logging
soa-tracking-trc-h...	<code>\${domain.home}/servers/\${weblogic.Name}/logs/\${weblogic.Name}-soa-tracking.trc</code>	Oracle Diagnostics Logging

ログは、次の場所で入手できる `{server-name}-diagnostic.log` で更新されます。

`<ORACLE_HOME>\user_projects\domains\base_domain\servers\<server_Name>\logs`

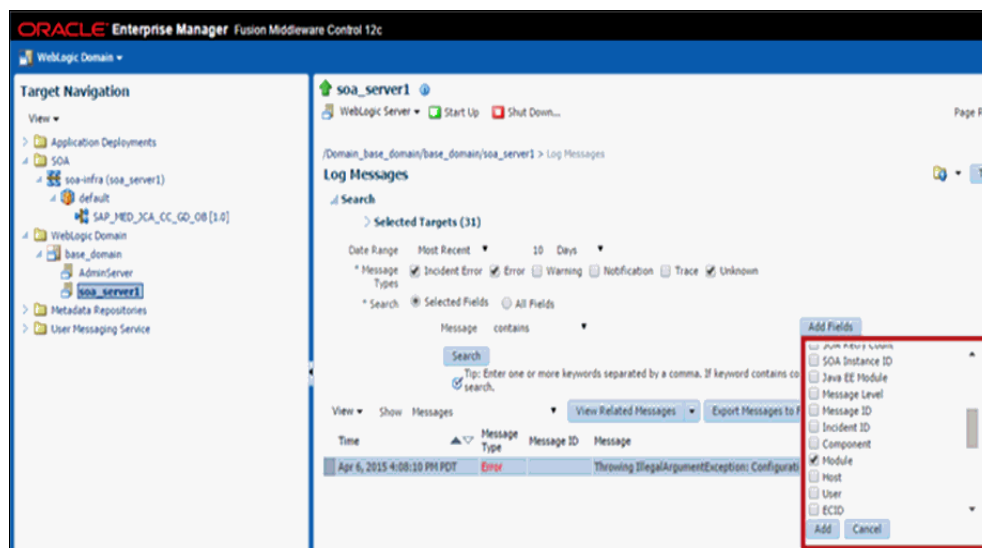
10. 図 11-6 に示すように、「サーバー」ペインで、「WebLogic Server」ドロップダウン・リストをクリックし、「ログ」を選択し、「ログ・メッセージの表示」をクリックすると、EM コンソールにログ・メッセージが表示されます。

図 11-6 「ログ・メッセージの表示」オプション



11. 「ログ・メッセージ」ペインで、必要な検索基準を入力します。図 11-7 に示すように、対象のログ出力の名前が含まれる「モジュール」フィールドを検索基準に追加することもできます。

図 11-7 「モジュール」フィールド



12. 「追加」をクリックします。
13. 図 11-8 に示すように、「モジュール」フィールドに対象のログ出力の名前を入力し、必要に応じて追加のメッセージ・タイプ（「警告」、「通知」、「トレース」など）を選択して、「検索」をクリックします。

図 11-8 「ログ・メッセージ」 ペイン

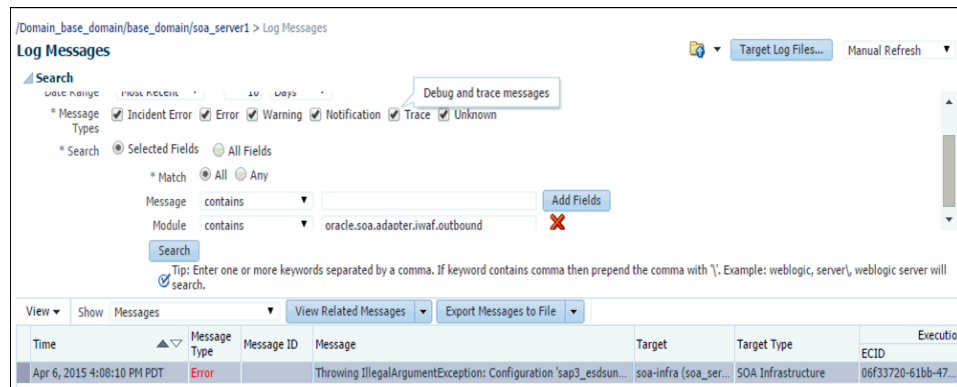
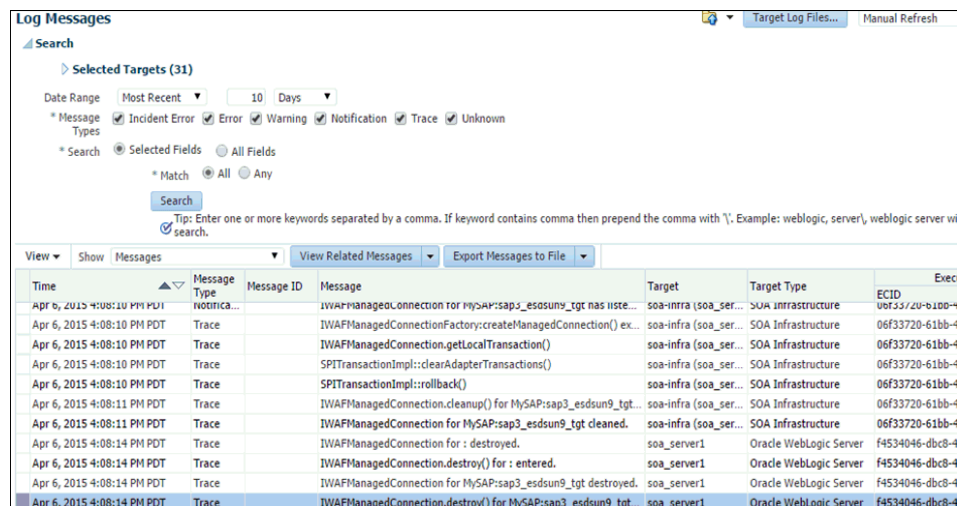


図 11-9 に示すように、指定されたログ出力のメッセージが、検索基準の下の表に表示されます。

図 11-9 ログ出力のメッセージ



14. 表の任意の行を選択します。図 11-10 に示すように、同じ詳細を取得するには、「関連メッセージの表示」ドロップダウン・リストをクリックし、「ECID」(実行コンテキスト ID)を選択します。

図 11-10 「ECID」 オプション

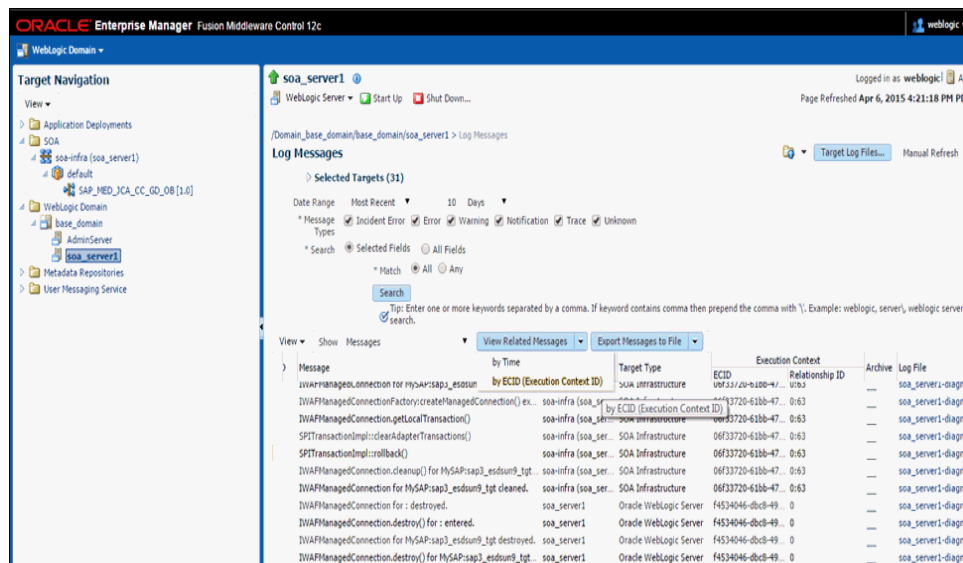
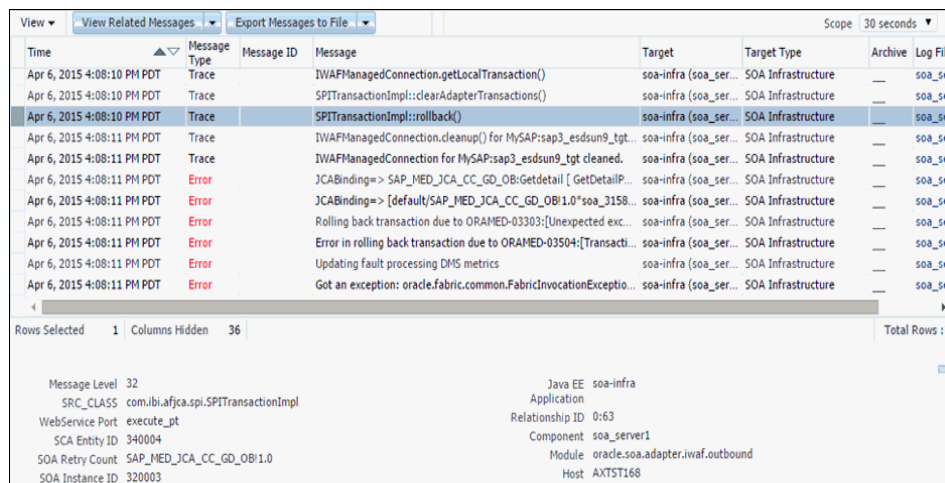


図 11-11 に示すように、詳細が表示されます。

図 11-11 メッセージの詳細



11.1.2 Business Services Engine (BSE) のログ・ファイル管理の構成

J2CA と同様に、BSE の場合も、ログ・ファイル管理は次の場所で定義されるログ出力によって制御されます。

```
<ORACLE_HOME>\user_projects\domains\base_
domain\config\fmwconfig\servers\${server-name}\logging.xml
```

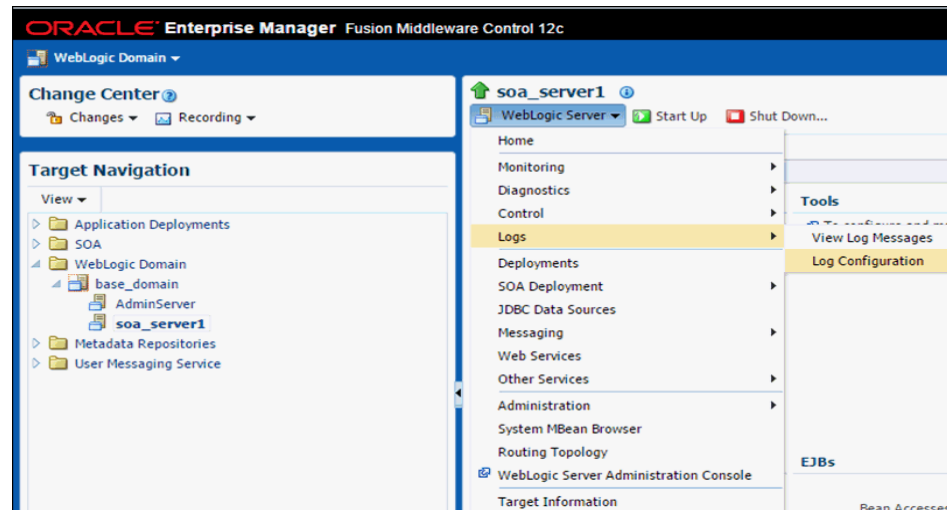
次の構文により、「oracle.soa.adapter.ibse」の下にあるすべてのログ出力のロギング・レベルが、Oracle によるデフォルトの設定レベルである「NOTIFICATION:1 (INFO)」に設定されます。

```
<logger name='oracle.soa.adapter.ibse' level='NOTIFICATION:1'
useParentHandlers='true' />
```

次のステップを実行して、すべてのログ出力のロギング・レベルを EM コンソールから構成することもできます。

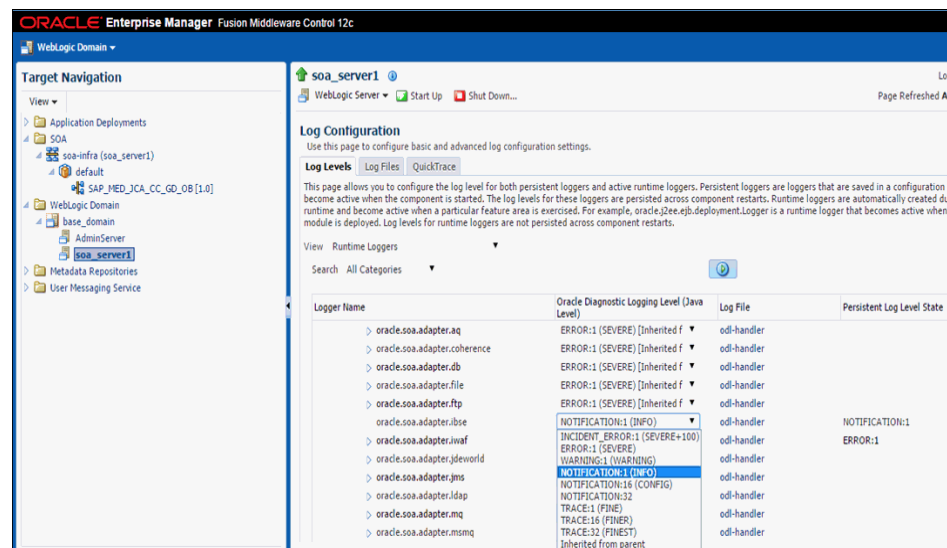
- 11.1.1 項「J2CA コネクタ・アプリケーションのログ・ファイル管理の構成」の説明に従って、ステップ 1 から 5 を繰り返します。
- 図 11-12 に示すように、「ログ・レベル」タブの下で、ログ出力名「oracle.soa.adapter.ibse」が表示されるまで Oracle ルート・ログ出力を展開します。

図 11-12 「ログ・レベル」タブ



- 図 11-13 に示すように、「Oracle Diagnostic Logging レベル (Java レベル)」列で、「oracle.soa.adapter.ibse」ドロップダウン・リストから必要なログ・レベルを選択します。

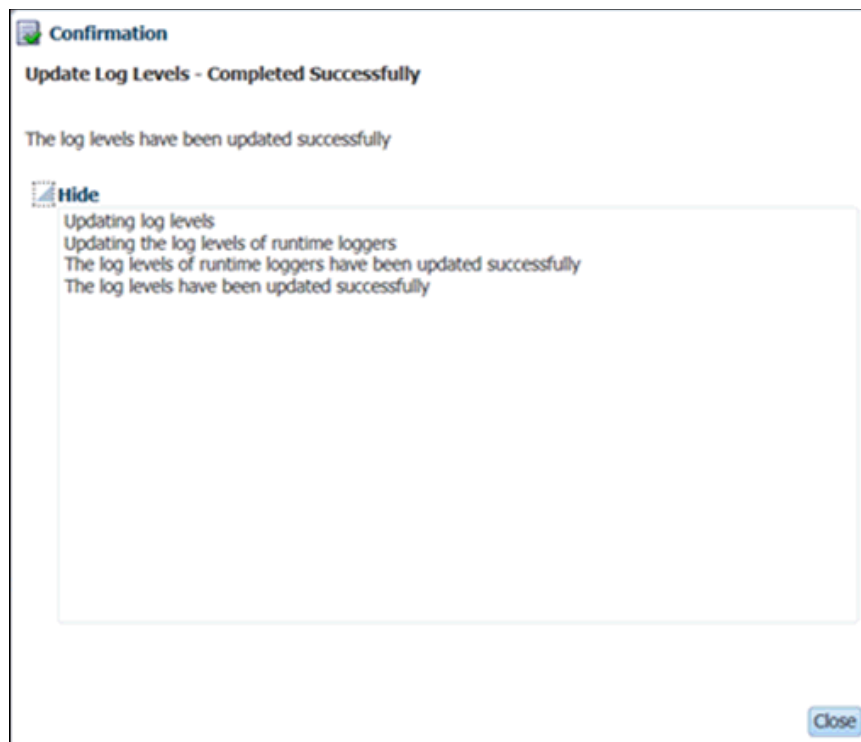
図 11-13 「Oracle Diagnostic Logging レベル (Java レベル)」列



- 「適用」をクリックします。

図 11-14 に示すように、更新が正常に完了したことを示す確認メッセージが表示されます。

図 11-14 確認メッセージ



5. 「閉じる」をクリックします。

次の表は、「ログ・レベル」プロパティが `{server-name}-diagnostic.log` で EM コンソール内の対応する「ログ・レベル」プロパティ設定に基づいてどのように更新されるかを示しています。

表 11-2 「ログ・レベル」プロパティ設定

EM コンソールで設定された「ログ・レベル」	<code>{server-name}-diagnostic.log</code> で更新された「ログ・レベル」
ERROR:1	ERROR
WARNING:1	WARNING
NOTIFICATION:1	NOTIFICATION
NOTIFICATION:16	NOTIFICATION
NOTIFICATION:32	NOTIFICATION
TRACE:1	NOTIFICATION
TRACE:16	NOTIFICATION
TRACE:32	TRACE:32

ログの検証については、[11.1.1 項「J2CA コネクタ・アプリケーションのログ・ファイル管理の構成」](#)に示す J2CA ロギング機能のステップ 11 から 18 を参照してください。

11.2 診断機能の構成

この項では、Oracle Fusion Middleware Application Adapters for Oracle WebLogic Server の診断機能を構成および使用方法について説明します。次のトピックが含まれています。

- 11.2.1 項「サポートされるプロトコル」

診断機能は、アダプタのエンドポイント・ヘルス・ステータスを取得して (利用可能な場合)、対応するアラートを Oracle アダプタ・フレームワークに提供します。これにより、そのアラートを EM コンソールに表示できるようになります。

注意： 診断機能でサポートされているのはインバウンド・アダプタ・プロセスのみです。

次の項に進む前に、インバウンド・プロセスがデプロイされていることを確認してください。

この情報をアダプタについて表示するには、次のステップを実行します。

1. Oracle WebLogic Server を起動し、Web ブラウザで次の URL を入力して、Oracle WebLogic Server Enterprise Manager Console を起動します。

`http://host name:port/em`

ここで、*host name* は Oracle WebLogic Server が稼働しているシステムの名前で、*port* は稼働している Oracle WebLogic Server のポートです。

2. 管理者権限を持つアカウントを使用して、Oracle WebLogic Server 管理コンソールにログインします。
3. 右ペインで、「SOA」、「soa-Infra (server_name)」、「デフォルト」、デプロイ済インバウンド・プロセスの順に展開します。
4. プロセスをクリックし、[図 11-15](#) に示すように、右ペインの「サービスと参照」セクションに表示されているサービスを選択します。

図 11-15 「サービスと参照」セクション

The screenshot shows the Oracle WebLogic Server Enterprise Manager Console interface. The top navigation bar includes 'SOA Composite' and 'Page Refreshed Apr 23, 2015 12:29:57 PM PDT'. Below the navigation bar, there are tabs for 'Dashboard', 'Composite Definition', 'Flow Instances', 'Unit Tests', and 'Policies'. The main content area is divided into two sections: 'Components' and 'Services and References'. The 'Components' section shows a table with one row: 'Mediator1' of type 'Mediator'. The 'Services and References' section shows a table with three rows: 'Service' (Type: JCA Adapter, Usage: Service, Total Messages: 0, Average Processing Time: 0.000), 'FileReference' (Type: JCA Adapter, Usage: Reference, Total Messages: 0, Average Processing Time: 0.000).

Components	
Name	Component Type
Mediator1	Mediator

Services and References				
Name	Type	Usage	Total Messages	Average Processing Time (sec)
Service	JCA Adapter	Service	0	0.000
FileReference	JCA Adapter	Reference	0	0.000

5. 「アダプタ・レポート」タブで、「レポートの有効化」チェック・ボックスを選択し、「スナップショット・レポート」を展開して、EIS の停止時間の詳細を表示します。[図 11-16](#) に示すように、「エラー・サマリー」をクリックするとスタック・トレースが表示されます。

図 11-16 スタック・トレース

The screenshot shows the SOA Composite console for a SAMPACC (Custom Adapter). The 'Adapter Reports' section is active, displaying 'EIS Connection Downtime' statistics. The table below shows a connection reset event for the 'soa_server1' node.

Node	EIS Connection Status	Managed Connections	Most Recent Time Stamp
		Currently Open, Average Number Used, Currently Free, Maximum Pool Size	Last Message Publication, Last Service Activation
No data to display			

Server Name	Start EIS Downtime	End EIS Downtime	Error Summary	Total EIS Connection Downtime (Seconds)	Data Location
soa_server1	Apr 10, 2015 4:12:20 AM	Apr 10, 2015 4:14:04 AM	Connection reset	103	

11.2.1 サポートされるプロトコル

この項では、Oracle Fusion Middleware Application Adapter for SAP R/3 (SAP JCo 3.0) でサポートされるプロトコルについて説明します。次のトピックで構成されています。

- 11.2.1.1 項「Oracle Fusion Middleware Application Adapter for SAP R/3 のエンドポイント・ステータス」

11.2.1.1 Oracle Fusion Middleware Application Adapter for SAP R/3 のエンドポイント・ステータス

Oracle Fusion Middleware Application Adapter for SAP R/3 (インバウンド) では、SAP API プロトコルを使用します。

Oracle Fusion Middleware Application Adapter for SAP R/3 (インバウンド) は、イベントについてサーバーをポーリングします。その後、アダプタは、サーバーが次のいずれかの状態であることを Oracle Adapter Framework に通知します。

- ダウン。使用可能でないことが確認された場合です。
- 実行中。通信が使用可能である場合です。

11.3 SOA デバッグ機能の構成

この項では、Oracle Application Adapter for SAP R/3 の SOA デバッグ機能を構成および使用する方法について説明します。次のトピックについて説明します。

- 11.3.1 項「SOA デバッガの使用のガイドライン」
- 11.3.2 項「前提条件」
- 11.3.3 項「Oracle JDeveloper での BPEL プロセスのデバッグ」
- 11.3.4 項「Oracle JDeveloper での OSB プロセスのデバッグ」

注意： 現在 SOA デバッグ機能がサポートされているのは J2CA 構成のみです。BSE 構成では使用できません。

SOA については、この機能の説明には BPEL プロセスを使用します。メディエータ・プロセスと BPM プロセスについても同様です。

11.3.1 SOA デバッガの使用のガイドライン

この項では、SOA デバッガの使用のガイドラインについて説明します。

1. SOA デバッガに接続できるクライアントは一度に1つのみです。
2. アダプタ・エンドポイント・エラーは、Oracle JDeveloper の SOA デバッガには表示されません。これらのエラーはログ・ファイルに記録されます。

注意： SOA デバッガは現在、BPEL、メディエータ、BPM、OSB の各プロセスで、開発モードでのみ使用できます。

11.3.2 前提条件

Oracle JDeveloper で *IntegratedWebLogicServer* ドメインと BPEL プロセスがすでに作成されていることを確認してください。

11.3.3 Oracle JDeveloper での BPEL プロセスのデバッグ

この項では、Oracle JDeveloper で BPEL プロセスをデバッグする方法について説明します。次のトピックについて説明します。

- 11.3.3.1 項「Oracle JDeveloper でのアウトバウンド BPEL プロセスのデバッグ」
- 11.3.3.2 項「Oracle JDeveloper でのインバウンド BPEL プロセスのデバッグ」

11.3.3.1 Oracle JDeveloper でのアウトバウンド BPEL プロセスのデバッグ


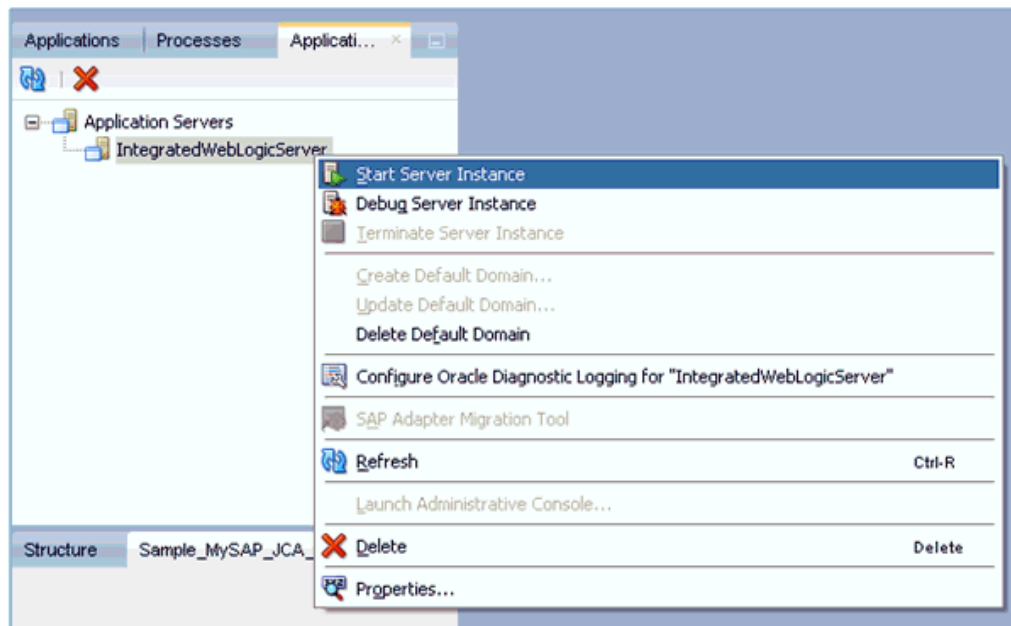
1. Oracle JDeveloper を開きます。
2. *IntegratedWeblogicServer* ドメインを起動します。
 - a. 左ペインの「アプリケーション・サーバー」タブをクリックします。
 - b.  11-17 に示すように、「アプリケーション・サーバー」ノードの下の「**IntegratedWeblogicServer**」を右クリックし、コンテキスト・メニューから「サーバー・インスタンスの起動」を選択します。

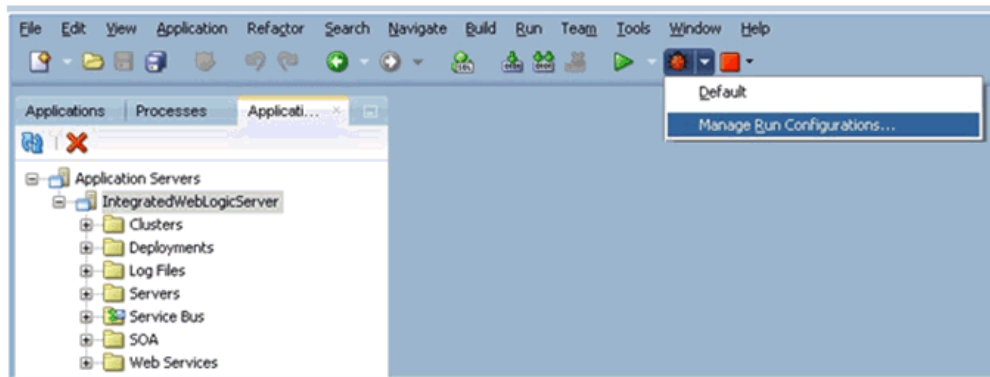
図 11-17 サーバー・インスタンスの起動



3. デバッグ環境を設定します。

- a. 図 11-18 に示すように、「デバッグ」アイコンの横の下矢印をクリックし、コンテキスト・メニューから「実行構成の管理」を選択します。

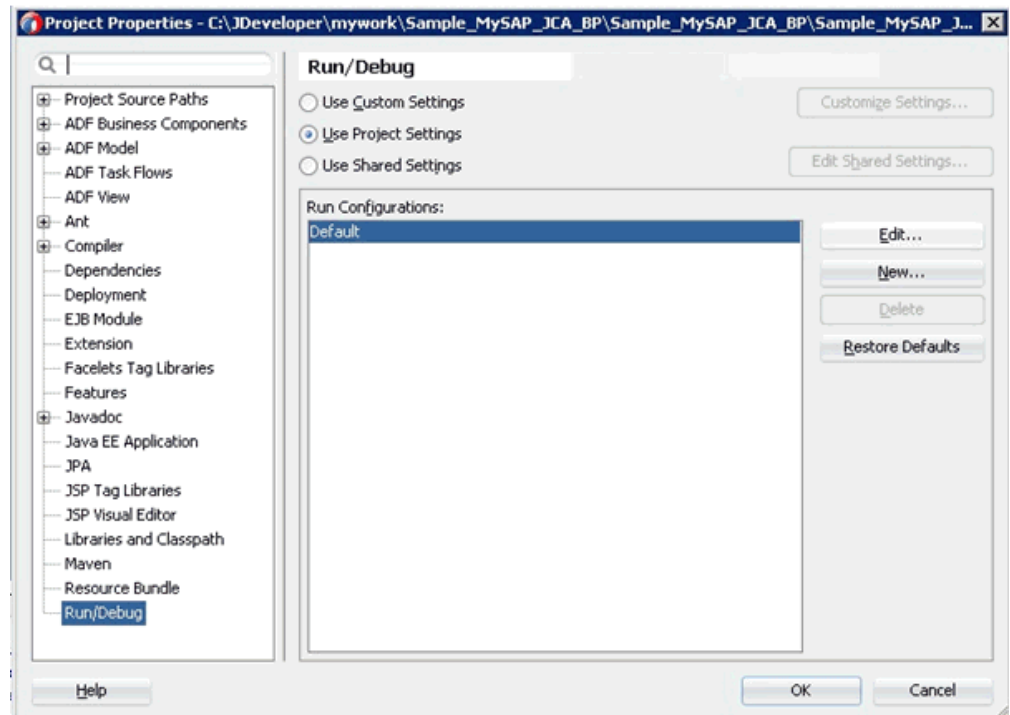
図 11-18 実行構成の管理



- b. または、プロジェクトを右クリックし、「プロジェクト・プロパティ」を選択します。

図 11-19 に示すように、「プロジェクト・プロパティ」ダイアログが表示されます。

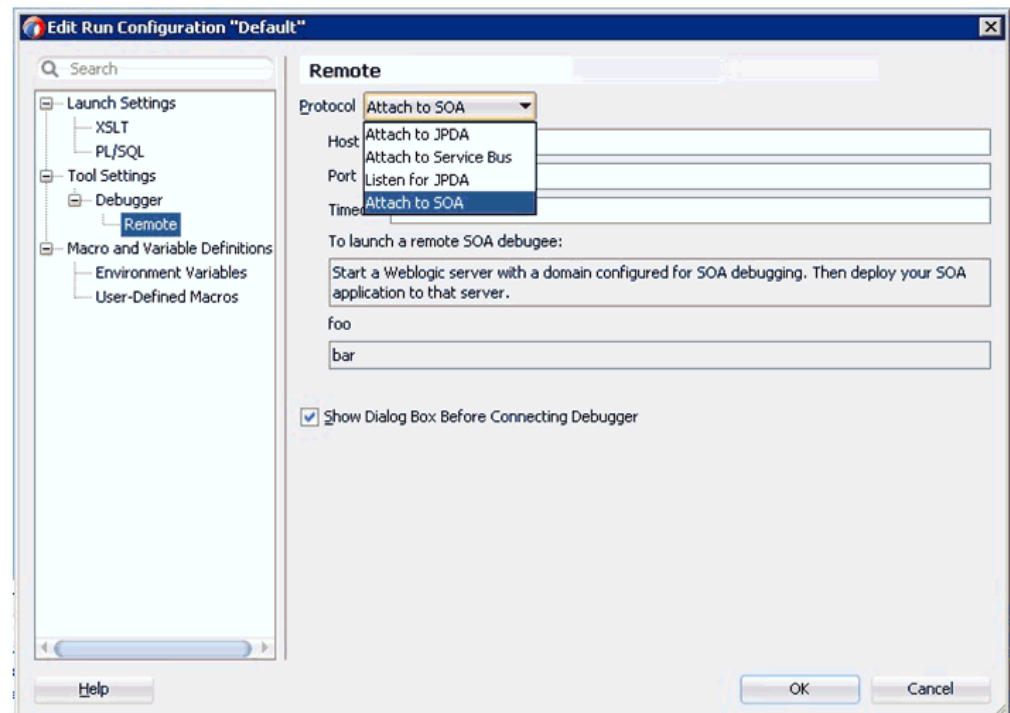
図 11-19 「プロジェクト・プロパティ」ダイアログ



- c. 左ペインで「実行/デバッグ」を選択し、「実行/デバッグ」ペイン（「実行構成」領域）で「編集」をクリックします。

図 11-20 に示すように、「実行構成の編集」ダイアログが表示されます。

図 11-20 「実行構成の編集」ダイアログ




- d. 左ペインで、「ツール設定」、「デバッガ」の順に展開し、「リモート」をクリックします。
 - e. 「プロトコル」リストから、「SOA にアタッチ」を選択します。
 - f. 「ホスト」と「ポート」はデフォルト値のままにしておきます。
 - g. 「OK」をクリックします。
4. プロジェクトをデプロイします。
- a. プロジェクトを右クリックし、「デプロイ」を選択します。
「デプロイメント・アクション」ダイアログが表示されます。
デプロイするアプリケーションを選択し、「次へ」をクリックします。
「構成のデプロイ」ダイアログが表示されます。
 - b. 「次へ」をクリックします。
「アプリケーション・サーバー」ダイアログが表示されます。
 - c. 「IntegratedWebLogicServer」を選択し、「終了」をクリックしてデプロイを完了します。
 - d. 次のステップに進む前に、プロジェクトのデプロイが完了し、エラーや問題が発生していないことを確認してください。
5. BPEL プロセスを SOA デバッガに接続します。
- a.  11-21 に示すように、左ペインの「アプリケーション」タブで、既存の BPEL プロセスのコンポジット XML またはコンポジット・プロジェクトを右クリックし、コンテキスト・メニューから「デバッグ」を選択します。

図 11-21 デバッグする BPEL プロセスを選択

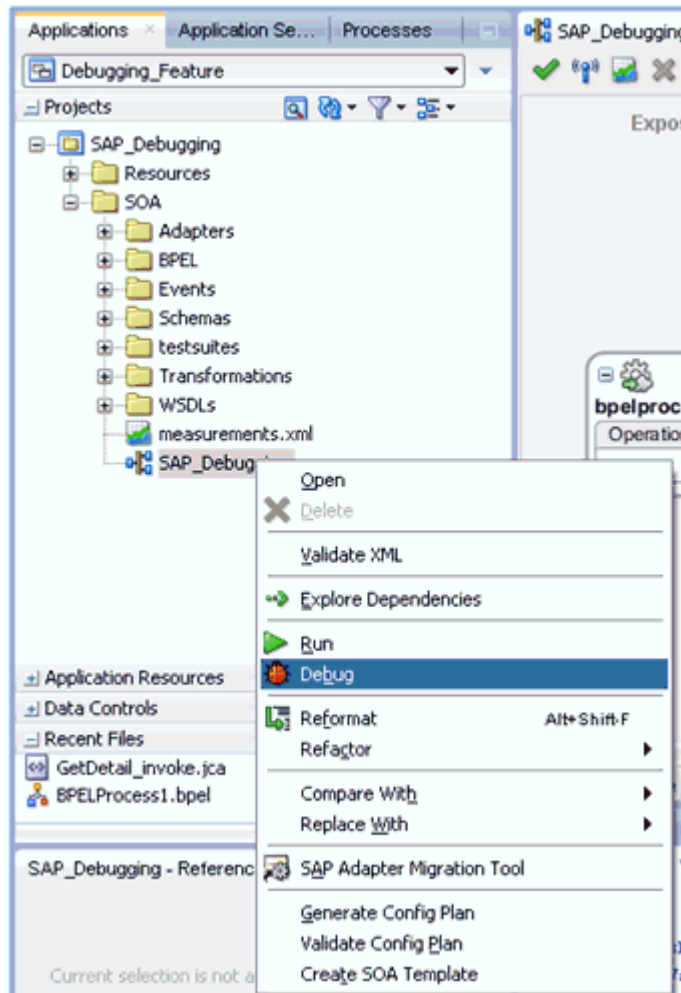
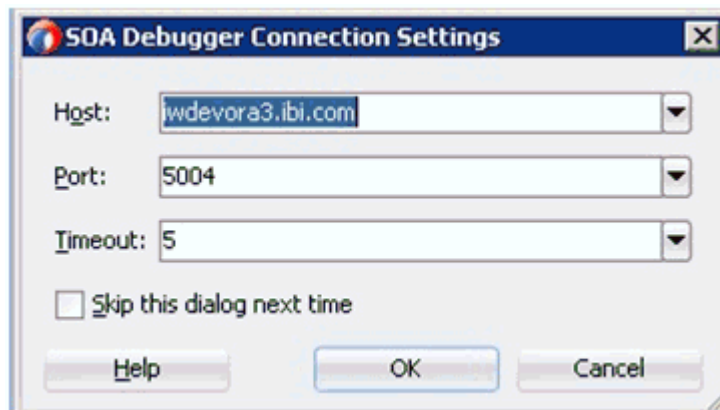


図 11-22 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 11-22 「SOA デバッガ接続設定」ダイアログ



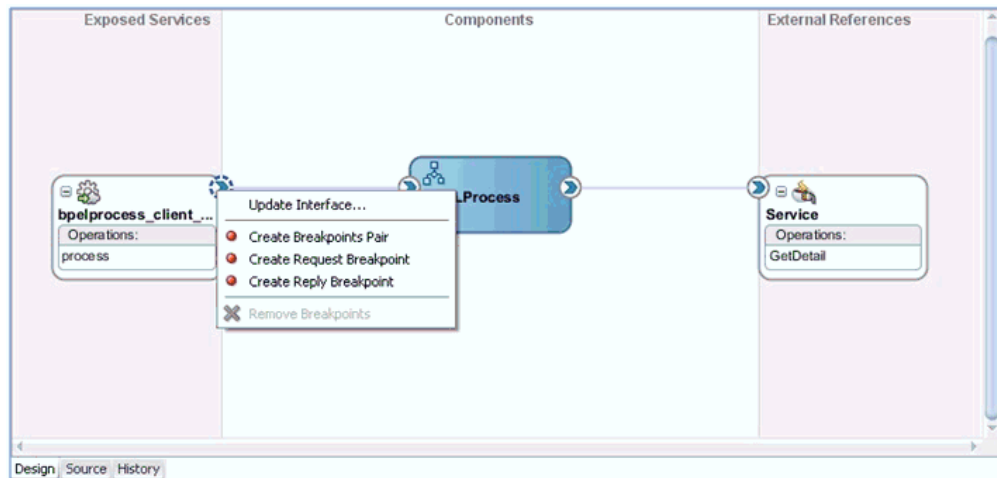
b. 「OK」をクリックします。

BPEL プロセスが SOA デバッガに接続されると、デバッグ・ログに次のメッセージが表示されます。

デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しようとしています。
 デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しました。
 デバッガ・プロセスの仮想マシンは SOA デバッガです。

6. ブレークポイントを設定してデバッグを開始します。
 - a. 図 11-23 に示すように、コンポーネントを右クリックし、設定するブレークポイント・タイプを選択します。

図 11-23 ブレークポイントの選択



「ブレークポイント・ペアの作成」：リクエスト/リプライ (アウトバウンド/インバウンド) 相互作用に対してはこのブレークポイント・タイプを設定します。リクエストとリプライの両方が重要となるシナリオに便利です。

「リクエスト・ブレークポイントの作成」：リクエスト (アウトバウンド) 相互作用に対してはこのブレークポイント・タイプを設定します。リクエストのみが重要となるシナリオに便利です。

「リプライ・ブレークポイントの作成」：リプライ (インバウンド) 相互作用に対してはこのブレークポイント・タイプを設定します。リプライのみが重要となるシナリオに便利です。

「WS デバッグの開始」：デバッグ・セッションを開始するにはこのブレークポイント・タイプを設定します。デバッグ・セッションには、たとえば、Web サービスから BPEL プロセスへの開始 SOAP リクエストから、アダプタ参照バインディング・コンポーネントまでが含まれます。

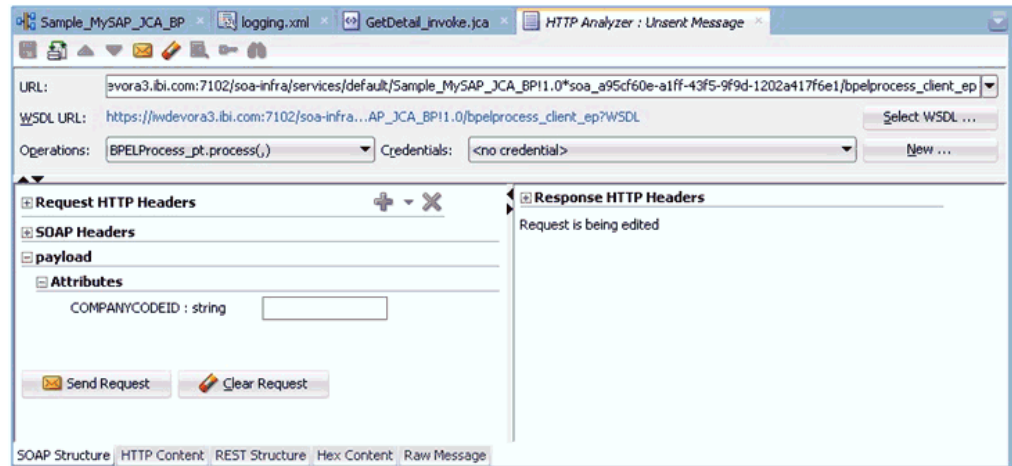
- b. ブレークポイントを設定したら、図 11-24 に示すように、右側のハンドルを右クリックして「WS デバッグの開始」を選択します。

図 11-24 WS デバッグの開始



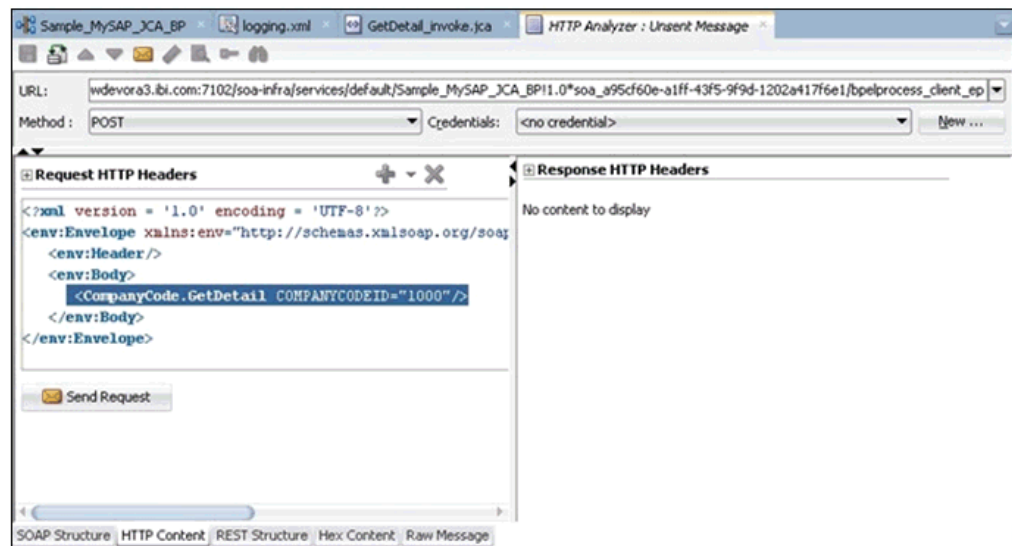
図 11-25 に示すように、「HTTP アナライザ」ダイアログが表示されます。

図 11-25 「HTTP アナライザ」ダイアログ



- c. 下のタブから「HTTP コンテンツ」を選択します。ここで、図 11-26 に示すように、ペイロードをコピーして本文に貼り付けます。

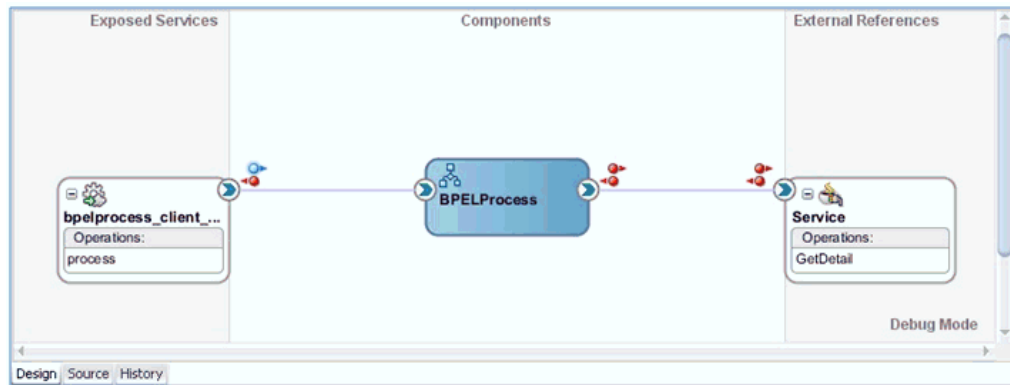
図 11-26 「HTTP コンテンツ」を選択



- d. 「リクエストの送信」をクリックします。

図 11-27 に示すように、指定したブレークポイントで BPEL プロセスが停止し、青く点滅します。






図 11-27 BPEL プロセスがブレークポイントで停止



- e. 図 11-28 に示すように、使用可能な「ステップ」オプションを使用してデバッグ・プロセスをステップ実行します。

図 11-28 「ステップ」オプション



アイコン	説明
	デバッグ・セッションを終了または連結解除します。
	フレームをステップ実行します。 これにより、次のブレークポイント (ブレークポイントが設定された BPEL プロセスの受信アクティビティなど) に移動します。ブレークポイントがない場合は、すべてのフレームをステップ実行して最初のフレームに戻ります。 [F8] を押してフレームをステップ実行することもできます。
	次の有効な位置までトレース実行します。 これは、新しいフレームである場合もあれば、同じフレームの別の位置である場合もあります。 [F7] を押してフレームをトレース実行することもできます。
	フレームをステップ・アウトします。 このオプションは、BPEL の Scope アクティビティまたは Sequence アクティビティを処理するためにのみ使用されます。スコープの処理が完了すると、プロセスの次のスコープまたはアクティビティで一時停止します。[Shift] + [F7] を押すこともできます。
	ステップ操作を再開します。 [F9] を押して再開することもできます。

7. リクエストのペイロードとヘッダー情報を表示します。

- a. 「ウィンドウ」をクリックし、「デバッグ」、「データ」の順に選択します。

Oracle Application Adapter for SAP R/3 のリクエスト・ブレークポイントで停止したときのヘッダー情報のサンプルを図 11-29 に示します。

図 11-29 ヘッダー情報

Name	Value	Decla...	Type	Hex V...	Address	ID
(x) completionPersistPolicy	null		osb.Variable			
(x) euid	148f5c57-bd1a-4571-81b0-2f9e2c29ccc2...		osb.Variable			
(x) masterConversationId	null		osb.Variable			
(x) MESH_METRICS	null		osb.Variable			
(x) nativePayload			osb.Variable			
(x) replyToPortType	null		osb.Variable			
(x) transaction	Name=[EJB com.collaxa.cube.engine.ejb.l...		osb.Variable			
(x) transport.http.remoteAddress	172.19.25.18		osb.Variable			
(x) XID	BEA1-0AD7677B5C6B9E472B1E(1799957...		osb.Variable			

Oracle Application Adapter for SAP R/3 のレスポンス・ペイロードのサンプルを図 11-30 に示します。

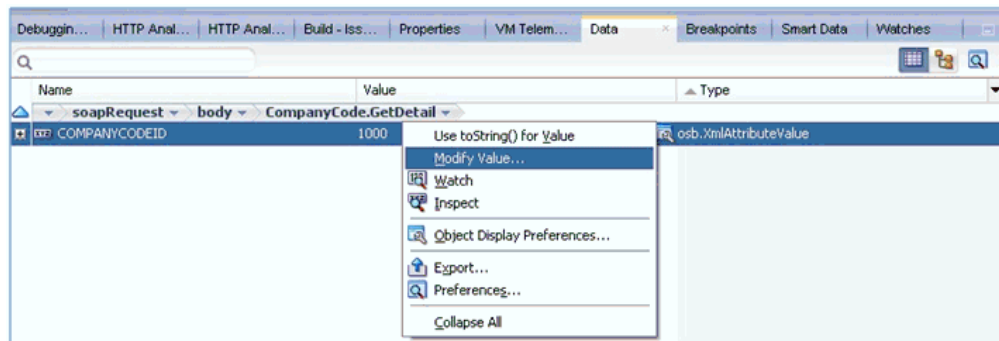
図 11-30 レスポンス・ペイロードのサンプル

```
<CompanyCode.GetDetail.Response xmlns="urn:sap-com:document:sap:business:response"
>
  <COMPANYCODE_ADDRESS>
    <ADDR_NO>0000000121</ADDR_NO>
    <FORMOFADDR>Firma</FORMOFADDR>
    <NAME>Ides AG</NAME>
    <NAME_2>Martin Steiner, Kathrin Walther,</NAME_2>
    <NAME_3>Bernd Zecha, Dondogmaa Lchamdondog</NAME_3>
    <NAME_4>IDES intern</NAME_4>
    <C_O_NAME/>
    <CITY>Frankfurt</CITY>
    <DISTRICT/>
    <CITY_NO/>
    <POSTL_COD1>60441</POSTL_COD1>
    <POSTL_COD2>60070</POSTL_COD2>
    <POSTL_COD3/>
    <PO_BOX>160529</PO_BOX>
    <PO_BOX_CITY/>
    <DELIV_DIS/>
    <STREET>Lyoner Stern 231</STREET>
    <STREET_NO/>
    <STR_ABBR/>
    <HOUSE_NO/>
    <STR_SUPPL1/>
    <STR_SUPPL2/>
    <LOCATION/>
    <BUILDING/>
    <FLOOR/>
    <ROOM_NO/>
    <COUNTRY>DE</COUNTRY>
    <LANGU>D</LANGU>
    <REGION>06</REGION>
    <SORT1>IDES</SORT1>
    <SORT2/>
    <TIME_ZONE>CET</TIME_ZONE>
    <TAXCURCODE/>
    <ADR_NOTES/>
    <COMM_TYPE/>
    <TEL1_NUMBER/>
    <TEL1_EXT/>
    <FAX_NUMBER/>
    <FAX_EXTENS/>
  </COMPANYCODE_ADDRESS>
  <COMPANYCODE_DETAIL>
    <COMP_CODE>1000</COMP_CODE>
    <COMP_NAME>IDES AG</COMP_NAME>
    <CITY>Frankfurt</CITY>
    <COUNTRY>DE</COUNTRY>
    <CURRENCY>EUR</CURRENCY>
    <LANGU>D</LANGU>
    <CHRT_ACCTS>INT</CHRT_ACCTS>
    <FY_VARIANT>K4</FY_VARIANT>
    <VAT_REG_NO>DE129456789</VAT_REG_NO>
    <COMPANY>001000</COMPANY>
    <ADDR_NO>0000000121</ADDR_NO>
    <COUNTRY_ISO>DE</COUNTRY_ISO>
    <CURRENCY_ISO>EUR</CURRENCY_ISO>
    <LANGU_ISO>DE</LANGU_ISO>
  </COMPANYCODE_DETAIL>
  <RETURN>
    <TYPE/>
    <CODE/>
    <MESSAGE/>
    <LOG_NO/>
    <LOG_MSG_NO>000000</LOG_MSG_NO>
  </RETURN>
</CompanyCode.GetDetail.Response>
```

注意： 図 11-30 のペイロードの表示は、画面サイズによって制限されています。実際には、BPEL プロセスのブレークポイントでは完全なペイロードが表示されます。画面をスクロールすると、ペイロードのすべての要素を表示できます。

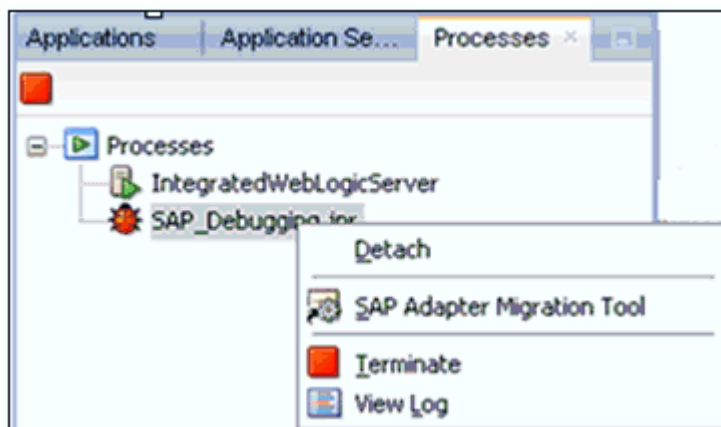
8. リクエスト・ペイロードのコンテンツを変更します。
 - a. 図 11-31 に示すように、SOAP リクエストを展開し、変更するフィールドを右クリックして、コンテキスト・メニューから「値の変更」を選択します。

図 11-31 値の変更



9. デバッグ・セッションを終了または接続解除します。
 - a. 「ウィンドウ」、「プロセス」の順にクリックします。図 11-32 に示すように、「プロセス」タブでプロセスを右クリックし、「接続解除」または「終了」を選択します。

図 11-32 デバッグ・セッションの接続解除



- b. 次のいずれかのオプションを選択します。
 - 「接続解除」：デバッグ・プロセスを終了せずに SOA デバッガを削除します。
 - 「終了」：デバッグ・プロセスを終了します。

11.3.3.2 Oracle JDeveloper でのインバウンド BPEL プロセスのデバッグ

1. インバウンド BPEL プロセスをデプロイします。

- a. プロジェクトを右クリックし、「デプロイ」を選択します。
「デプロイメント・アクション」ダイアログが表示されます。
 - b. 「アプリケーション・サーバーへのデプロイ」を選択し、「次へ」をクリックします。
「構成のデプロイ」ダイアログが表示されます。
 - c. 「次へ」をクリックします。
「サーバーの選択」ダイアログが表示されます。
 - d. デプロイするサーバーを選択し、「終了」をクリックします。
 - e. 次のステップに進む前に、プロジェクトのデプロイが完了し、エラーや問題が発生していないことを確認してください。
2. BPEL プロセスを SOA デバッガに接続します。
- a. 図 11-33 に示すように、インバウンド・コンポジットを右クリックし、「デバッグ」を選択します。

図 11-33 デバッグする BPEL プロセスを選択

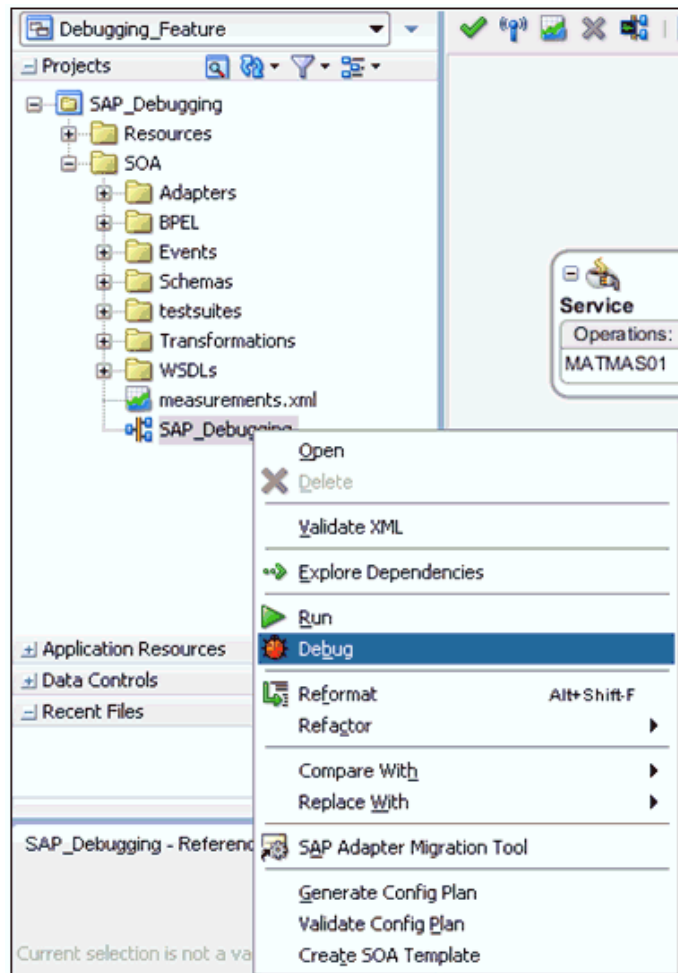
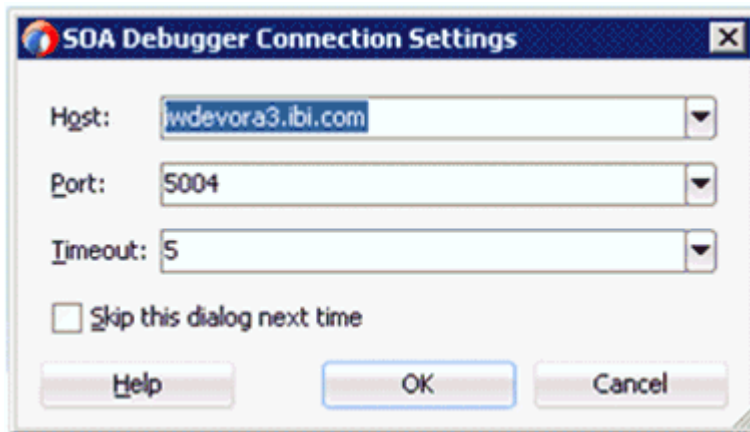


図 11-34 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 11-34 「SOA デバッガ接続設定」ダイアログ



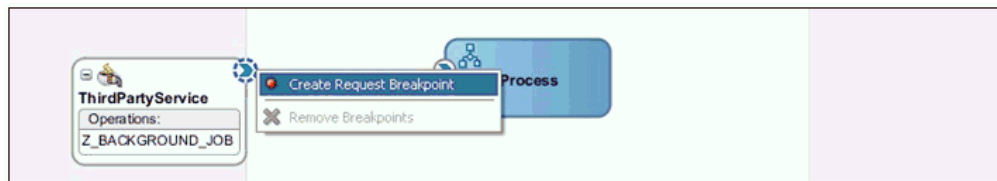
- b. 「OK」をクリックします。

BPEL プロセスが SOA デバッガに接続されると、デバッグ・ログに次のメッセージが表示されます。

デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しようとしています。
 デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しました。
 デバッガ・プロセスの仮想マシンは SOA デバッガです。

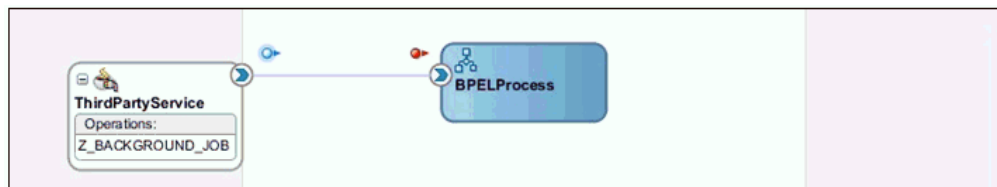
3. 図 11-35 に示すように、ブレークポイントを設定します。

図 11-35 ブレークポイントの設定



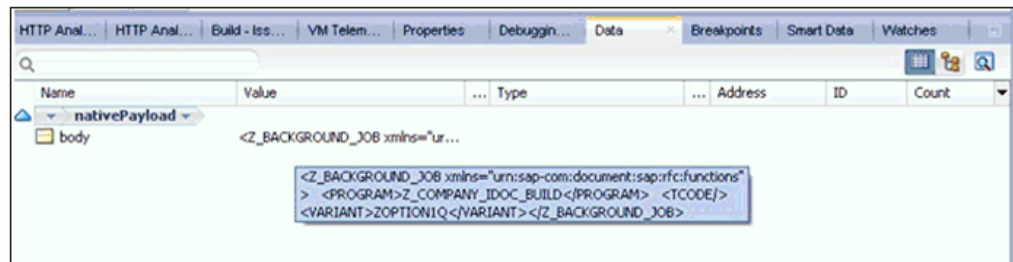
4. ブレークポイントが設定されて、インバウンド・プロセスから (SAP からトリガーされるなどして) メッセージが受信されると、図 11-36 に示すように、指定したブレークポイントでプロセスが停止します。

図 11-36 ブレークポイントでプロセスを停止



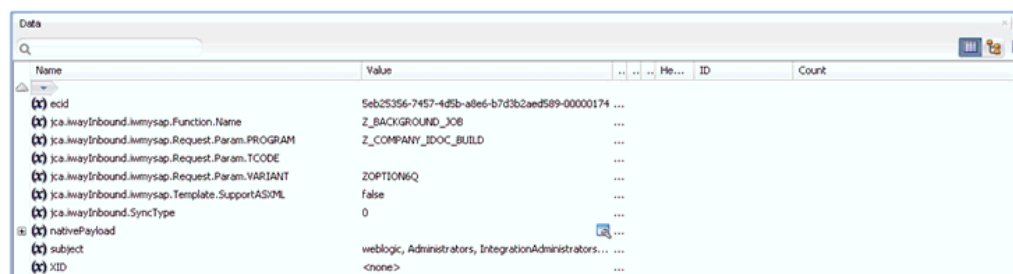
5. リクエストのペイロードとヘッダー情報を表示します。
- a. 図 11-37 に示すように、「ウィンドウ」をクリックし、「デバッガ」、「データ」の順に選択します。

図 11-37 デバッガのデータ



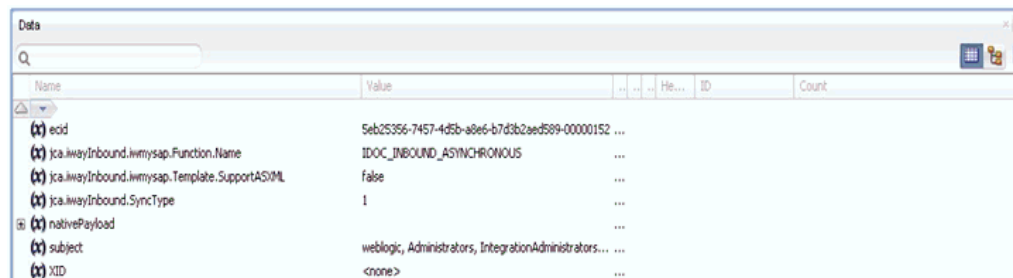
- b. インバウンド・リクエスト・ブレイクポイント (Z_BACKGROUND_JOB) に対して表示されるヘッダー情報を図 11-38 に示します。

図 11-38 インバウンド・リクエスト・ブレイクポイント (Z_BACKGROUND_JOB)



- c. インバウンド・リクエスト・ブレイクポイント (IDOC MATMAS01) に対して表示されるヘッダー情報を図 11-39 に示します。

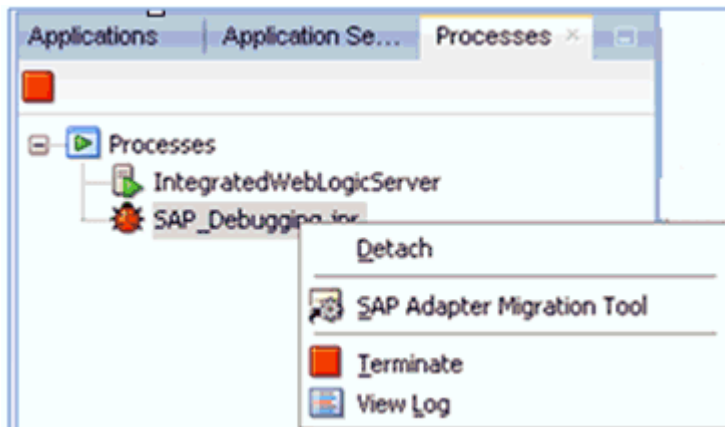
図 11-39 インバウンド・リクエスト・ブレイクポイント (IDOC MATMAS01)



6. デバッグ・セッションを終了または連結解除します。

- a. 「ウィンドウ」、「プロセス」の順にクリックします。図 11-40 に示すように、「プロセス」タブでプロセスを右クリックし、「連結解除」または「終了」を選択します。

図 11-40 デバッグ・セッションの連結解除



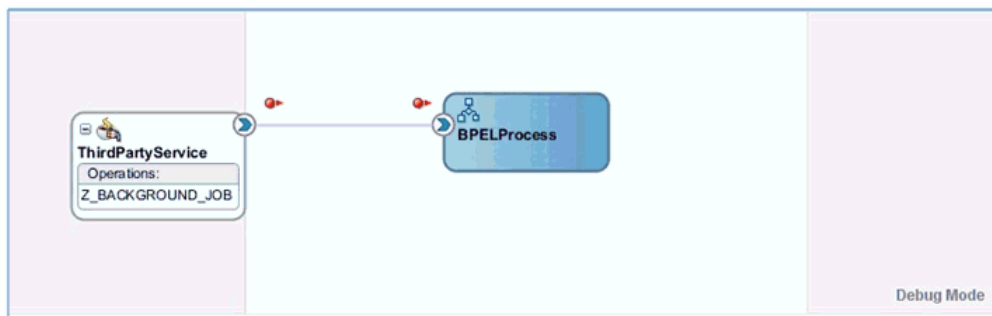
- b. 次のいずれかのオプションを選択します。

「連結解除」: デバッグ・プロセスを終了せずに SOA デバッガを削除します。

「終了」: デバッグ・プロセスを終了します。

図 11-41 に示すように、プロセスが連結解除されて表示されます。

図 11-41 連結解除された BPEL プロセス



11.3.4 Oracle JDeveloper での OSB プロセスのデバッグ

この項では、Oracle JDeveloper で OSB プロセスをデバッグする方法について説明します。次のトピックについて説明します。

- 11.3.4.1 項「前提条件」
- 11.3.4.2 項「Oracle JDeveloper でのアウトバウンド OSB プロセスのデバッグ」
- 11.3.4.3 項「Oracle JDeveloper でのインバウンド OSB プロセスのデバッグ」

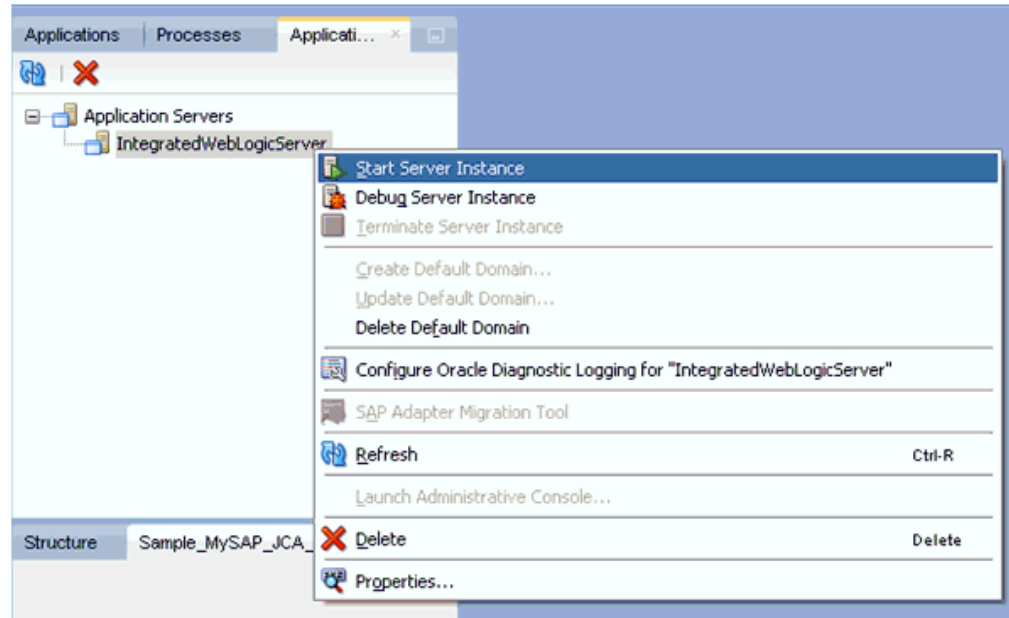
11.3.4.1 前提条件

Oracle JDeveloper で、*IntegratedWebLogicServer* ドメインと、ファイル出力を含む OSB プロセスがすでに作成されていることを確認してください。

注意: アプリケーション名とプロジェクト名が同じであることを確認してください。

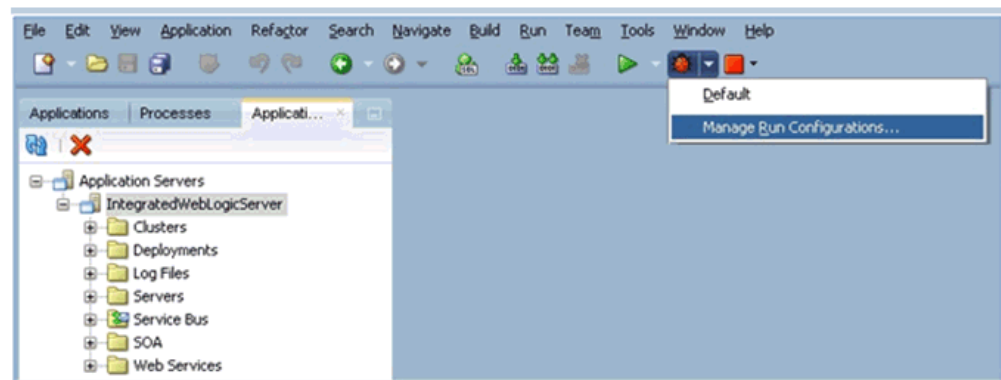
1. Oracle JDeveloper を開き、*IntegratedWebLogicServer* または *startWebLogic.cmd* を起動します。
 - a. 左ペインの「アプリケーション・サーバー」タブをクリックします。
 - b. 図 11-42 に示すように、「アプリケーション・サーバー」ノードの下の「**IntegratedWebLogicServer**」を右クリックし、コンテキスト・メニューから「サーバー・インスタンスの起動」を選択します。

図 11-42 サーバー・インスタンスの起動



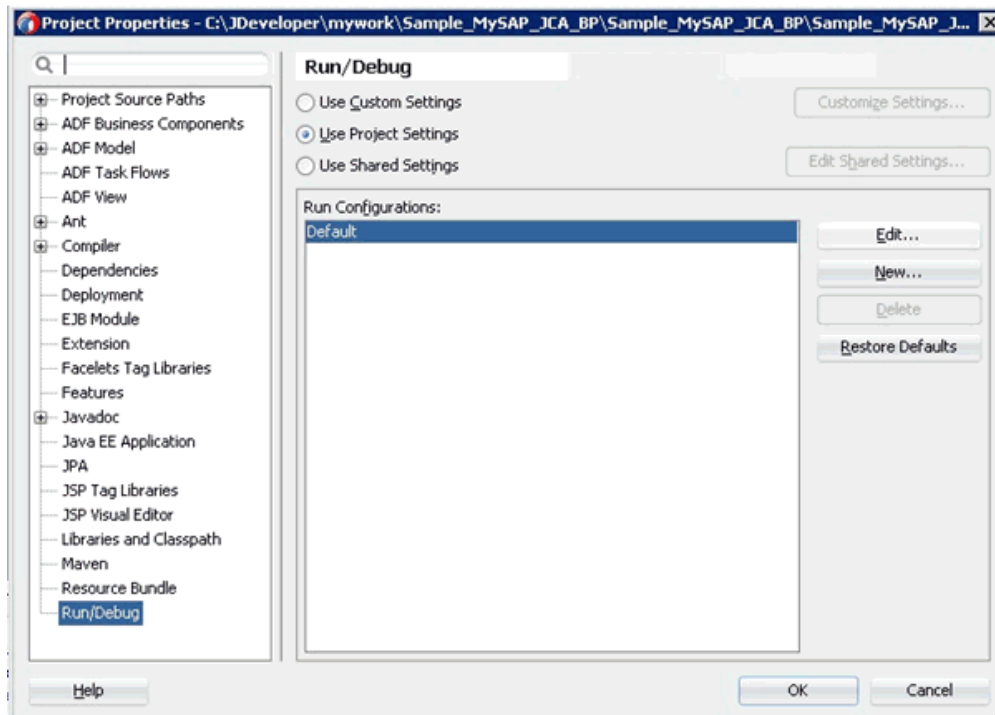
- c. または、*startWebLogic.cmd* を使用して、コマンド・プロンプトから Oracle WebLogic Server を起動します。
2. デバッグ環境を設定します。
 - a. 図 11-43 に示すように、「デバッグ」アイコンの横の下矢印をクリックし、コンテキスト・メニューから「実行構成の管理」を選択します。

図 11-43 実行構成の管理



- b. または、プロジェクトを右クリックし、「プロジェクト・プロパティ」を選択します。
- 図 11-44 に示すように、「プロジェクト・プロパティ」ダイアログが表示されます。

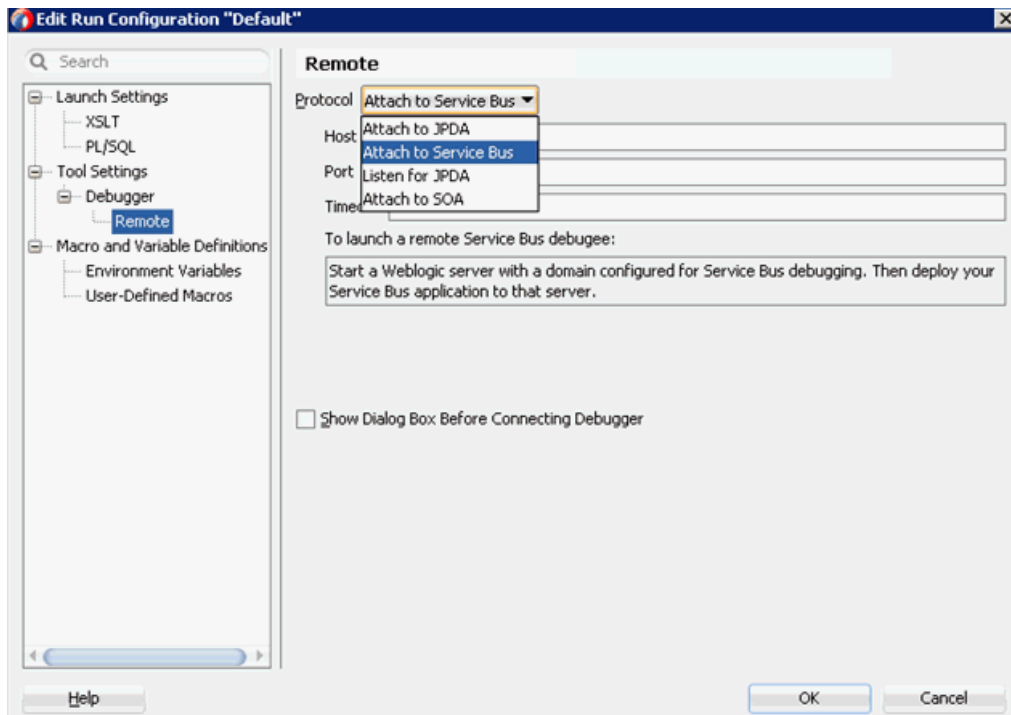
図 11-44 「プロジェクト・プロパティ」ダイアログ



- c. 左ペインで「実行/デバッグ」を選択し、「実行/デバッグ」ペイン（「実行構成」領域）で「編集」をクリックします。

図 11-45 に示すように、「実行構成の編集」ダイアログが表示されます。

図 11-45 「実行構成の編集」ダイアログ



- d. 左ペインで、「ツール設定」、「デバッガ」の順に展開し、「リモート」をクリックします。
- e. 「プロトコル」リストから、「Service Bus にアタッチ」を選択します。
- f. 「ホスト」と「ポート」はデフォルト値のままにしておきます。
- g. 「OK」をクリックします。

11.3.4.2 Oracle JDeveloper でのアウトバウンド OSB プロセスのデバッグ

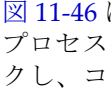
1. プロジェクトをデプロイします。
 - a. プロジェクトを右クリックし、「デプロイ」を選択します。
「デプロイメント・アクション」ダイアログが表示されます。
 - b. デプロイするアプリケーションを選択し、「次へ」をクリックします。
「構成のデプロイ」ダイアログが表示されます。
 - c. 「次へ」をクリックします。
「アプリケーション・サーバー」ダイアログが表示されます。
 - d. 「IntegratedWebLogicServer」を選択し、「終了」をクリックしてデプロイを完了します。
 - e. 次のステップに進む前に、プロジェクトのデプロイが完了し、エラーや問題が発生していないことを確認してください。
2. OSB プロセスを SOA デバッガに接続します。
 - a.  11-46 に示すように、左ペインの「アプリケーション」タブで、既存の OSB プロセスのコンポジット XML またはコンポジット・プロジェクトを右クリックし、コンテキスト・メニューから「デバッグ」を選択します。

図 11-46 デバッグする OSB プロセスを選択

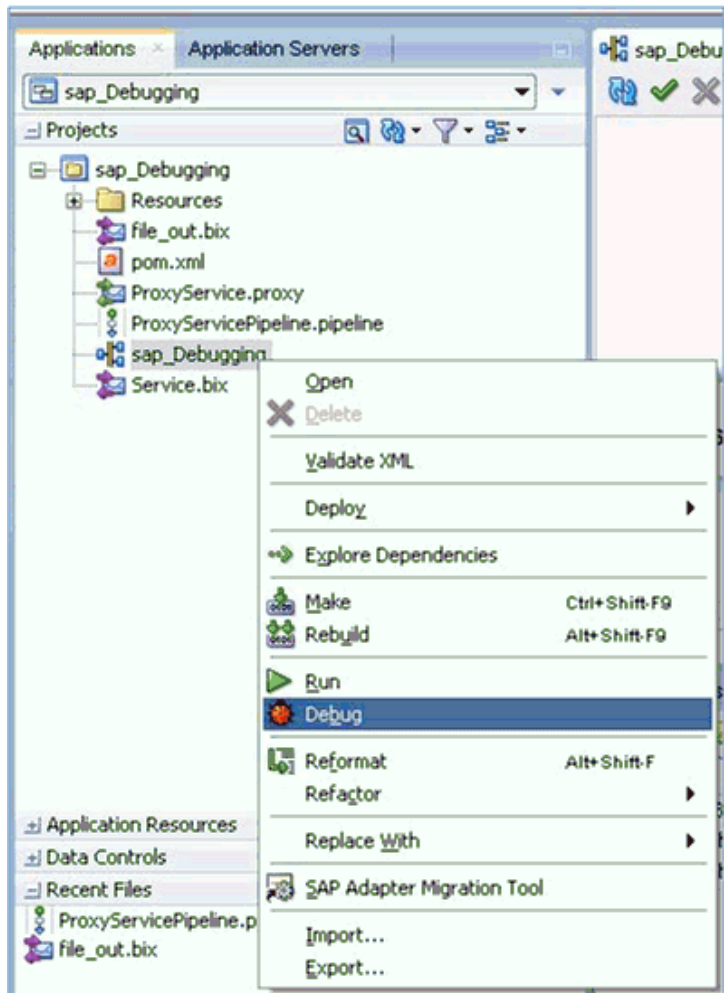
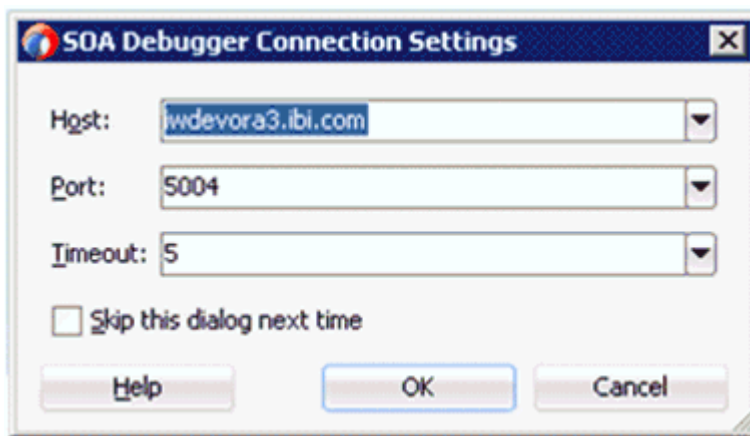


図 11-47 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 11-47 「SOA デバッガ接続設定」ダイアログ



b. 「OK」をクリックします。

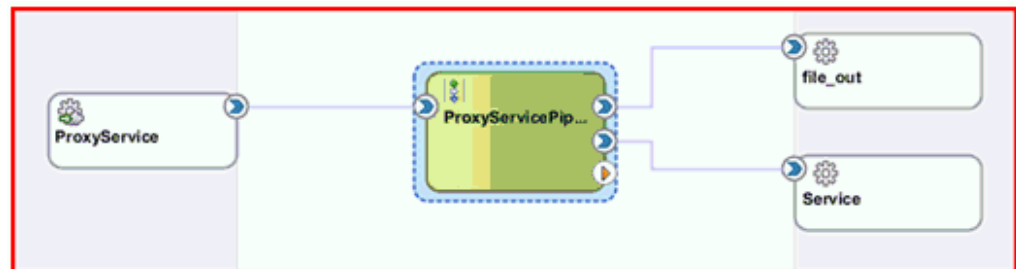
OSB プロセスが SOA デバッガに接続されると、デバッグ・ログに次のメッセージが表示されます。

デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しようとしています。
 デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しました。
 デバッガ・プロセスの仮想マシンは SOA デバッガです。

3. ブレークポイントを設定します。

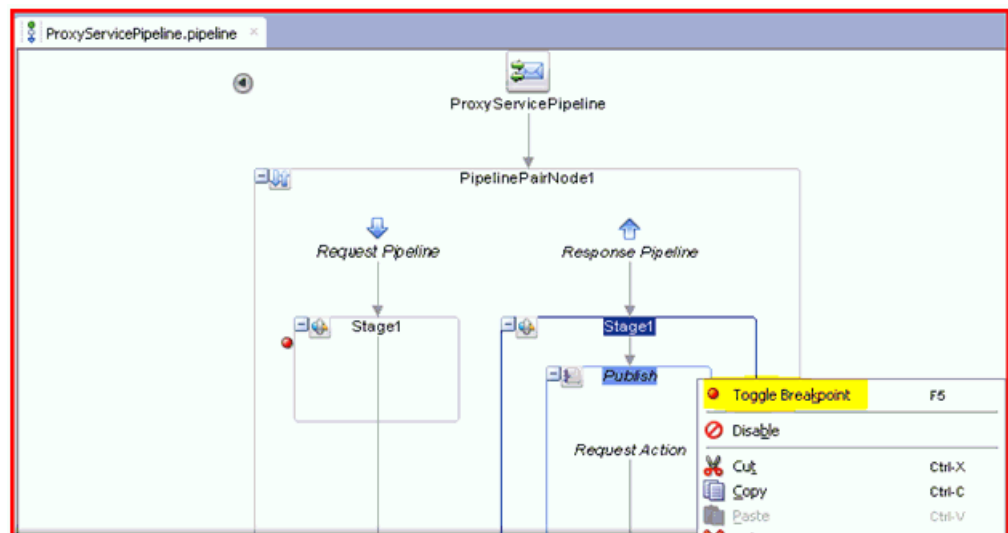
- a. 図 11-48 に示すように、パイプラインをダブルクリックしてエディタで開きます。

図 11-48 ProxyServicePipeline



- b. ブレークポイントを追加するノードが表示されるまでアクションを展開します。この例では、図 11-49 に示すように、PipelinePairNode1 を展開し、「ステージ 1」ノード（「リクエスト・パイプライン」の下）を右クリックして、コンテキスト・メニューから「ブレークポイントの切替え」を選択します。

図 11-49 ブレークポイントの切替え



このステップを「パブリッシュ」ノード（「レスポンス・パイプライン」の下）に対して繰り返します。

ノードの横に、ブレークポイントが設定されたことを示す赤いアイコンが表示されます。

注意： ブレークポイントを無効にするには、ノードを右クリックして「ブレークポイントの無効化」を選択します。

ブレークポイントを削除するには、ノードを右クリックして再度「ブレークポイントの切替え」を選択します。

4. デバッグを開始します。
 - a. 図 11-50 に示すように、「アプリケーション・ナビゲータ」でパイプラインを右クリックし、「デバッグ」を選択します。

図 11-50 「デバッグ」を選択

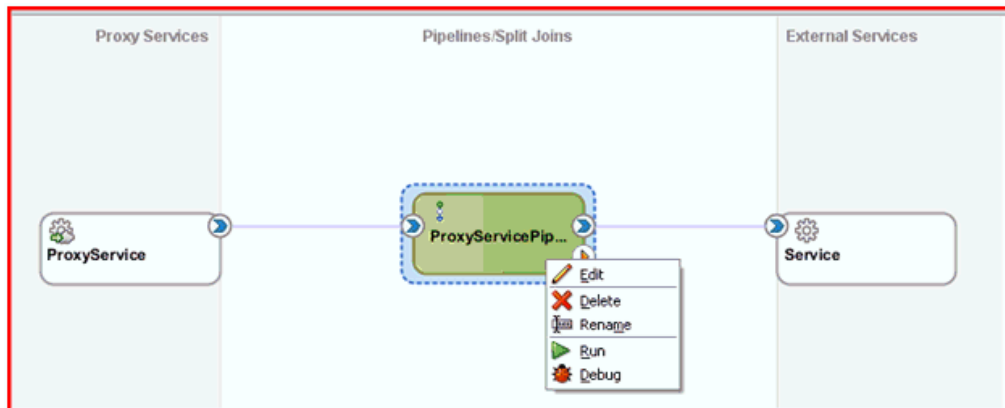
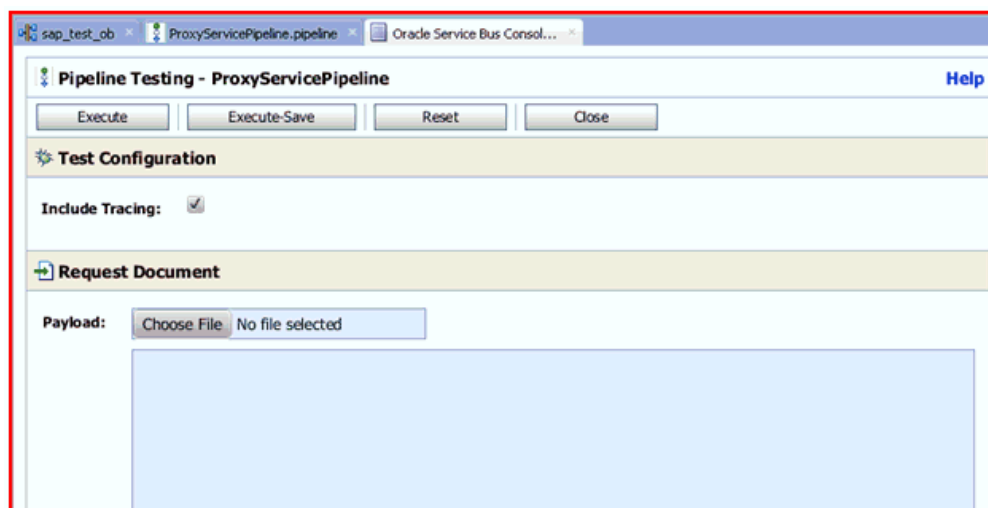


図 11-51 に示すように、プロセスが統合サーバーにデプロイされて、「構成のテスト」ペインが表示されます。

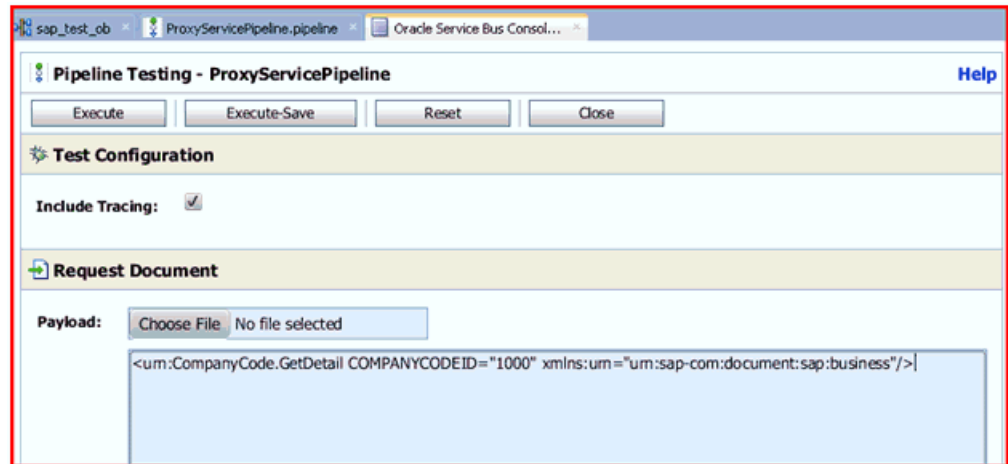
図 11-51 「構成のテスト」ペイン



注意： 現在実行されているドメインがない場合は、「デフォルト・ドメインの作成」ダイアログが表示されます。統合サーバーの接続情報を入力し、「OK」をクリックします。このプロセスには数分かかる場合があります。

- b. 図 11-52 に示すように、「構成のテスト」ペインで、「リクエスト・ドキュメント」領域にテスト・データを入力し、必要に応じて追加の入力を構成します。

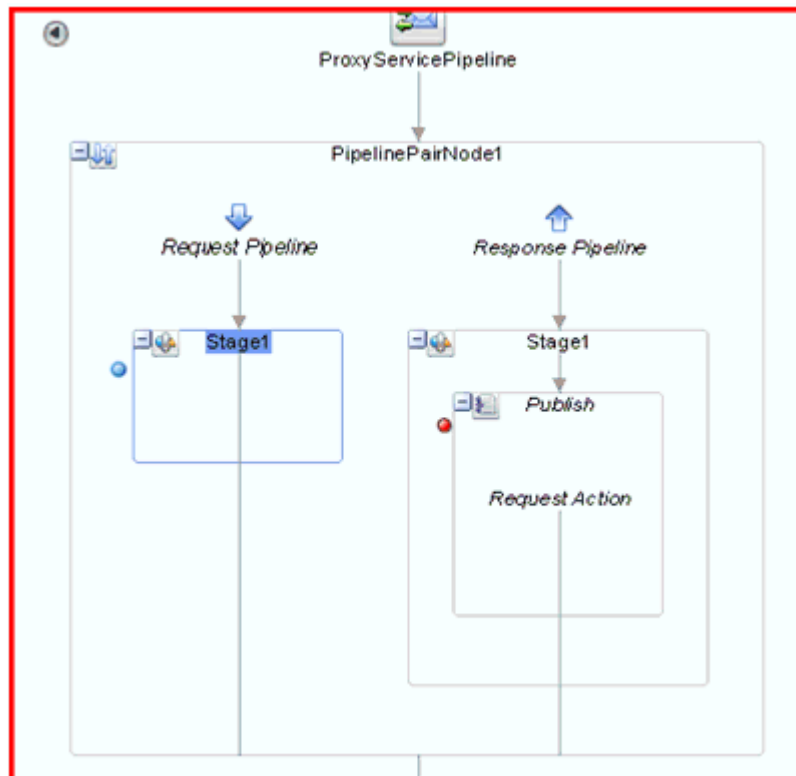
図 11-52 リクエスト・ドキュメントのテスト



- c. 「実行」をクリックします。

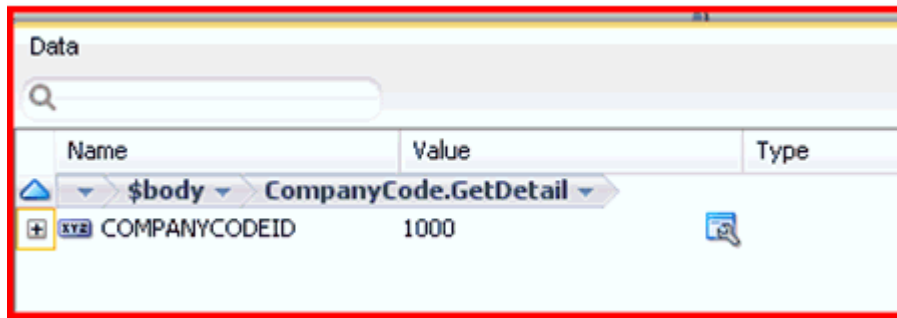
「構成のテスト」ペインでコマンドが実行されますが、図 11-53 に示すように、指定したブレークポイントで OSB プロセスが停止し、青く点滅します。

図 11-53 OSB プロセスがブレークポイントで停止



- d. 図 11-54 に示すように、下部の「データ」タブで、入力されたことを確認します。






図 11-54 「データ」タブ



- e. 図 11-55 に示すように、使用可能な「ステップ」オプションを使用してデバッグ・プロセスをステップ実行します。

図 11-55 「ステップ」オプション

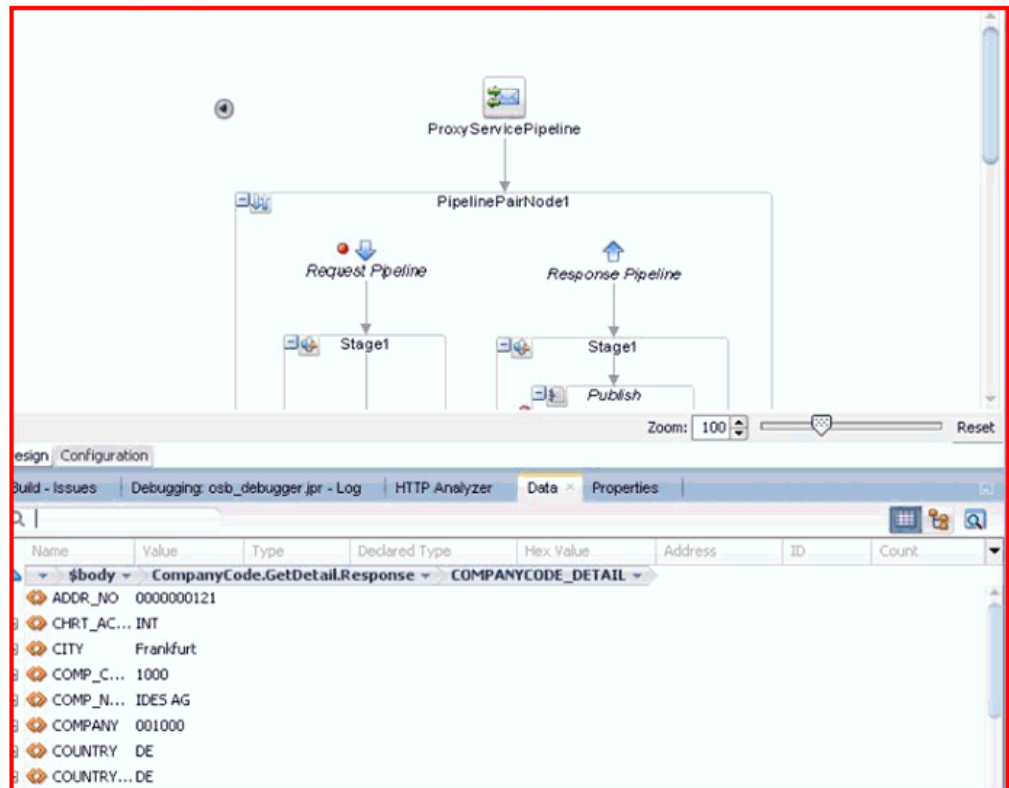


アイコン	説明
	デバッグ・セッションを終了または連結解除します。
	フレームをステップ実行します。 これにより、次のブレークポイント (ブレークポイントが設定された OSB プロセスの受信アクティビティなど) に移動します。ブレークポイントがない場合は、すべてのフレームをステップ実行して最初のフレームに戻ります。 [F8] を押してフレームをステップ実行することもできます。
	次の有効な位置までトレース実行します。 これは、新しいフレームである場合もあれば、同じフレームの別の位置である場合もあります。 [F7] を押してフレームをトレース実行することもできます。
	フレームをステップ・アウトします。 このオプションは、OSB の Scope アクティビティまたは Sequence アクティビティを処理するためにのみ使用されます。スコープの処理が完了すると、プロセスの次のスコープまたはアクティビティで一時停止します。 [Shift] + [F7] を押すこともできます。
	ステップ操作を再開します。 [F9] を押して再開することもできます。

- f. 「ステップ実行」を使用して、次のブレークポイント (この例では「レスポンス・パイプライン」) に移動します。

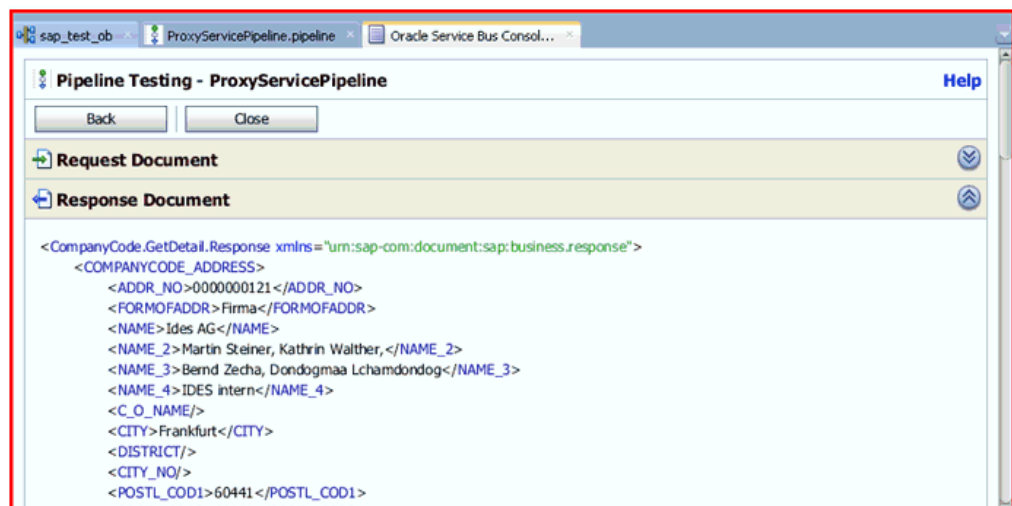
図 11-56 に示すように、「データ」タブで出力を確認できます。

図 11-56 「データ」タブの出力



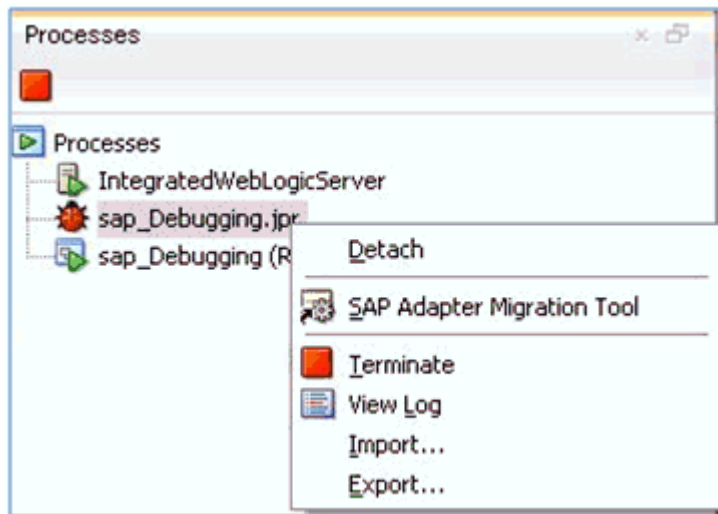
また、図 11-57 に示すように、「Oracle Service Bus コンソール」でレスポンスを確認することもできます。

図 11-57 Oracle Service Bus コンソール



5. デバッグ・セッションを終了または連結解除します。
 - a. 「ウィンドウ」、「プロセス」の順にクリックします。図 11-58 に示すように、「プロセス」タブでプロセスを右クリックし、「連結解除」または「終了」を選択します。

図 11-58 デバッグ・セッションの連結解除



- b. 次のいずれかのオプションを選択します。
 - 「連結解除」: デバッグ・プロセスを終了せずに SOA デバッガを削除します。
 - 「終了」: デバッグ・プロセスを終了します。

11.3.4.3 Oracle JDeveloper でのインバウンド OSB プロセスのデバッグ

1. インバウンド OSB プロセスを作成します。

アプリケーション名とプロジェクト名が同じであることを確認してください。
2. プロジェクトをデプロイします。
 - a. プロジェクトを右クリックし、「デプロイ」を選択します。

「デプロイメント・アクション」ダイアログが表示されます。
 - b. デプロイするアプリケーションを選択し、「次へ」をクリックします。

「構成のデプロイ」ダイアログが表示されます。
 - c. 「次へ」をクリックします。

「アプリケーション・サーバー」ダイアログが表示されます。
 - d. 「IntegratedWebLogicServer」を選択し、「終了」をクリックしてデプロイを完了します。
 - e. 次のステップに進む前に、プロジェクトのデプロイが完了し、エラーや問題が発生していないことを確認してください。
3. OSB プロセスを SOA デバッガに接続します。
 - a. 図 11-46 に示すように、左ペインの「アプリケーション」タブで、既存の OSB プロセスのコンポジット XML またはコンポジット・プロジェクトを右クリックし、コンテキスト・メニューから「デバッグ」を選択します。

図 11-59 デバッグする OSB プロセスを選択

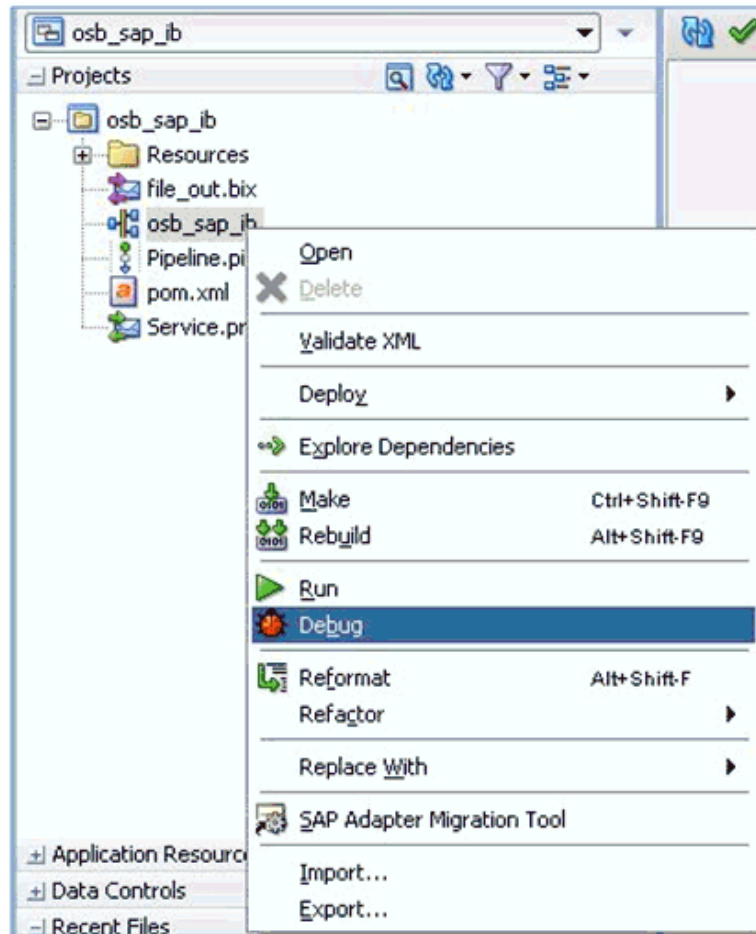
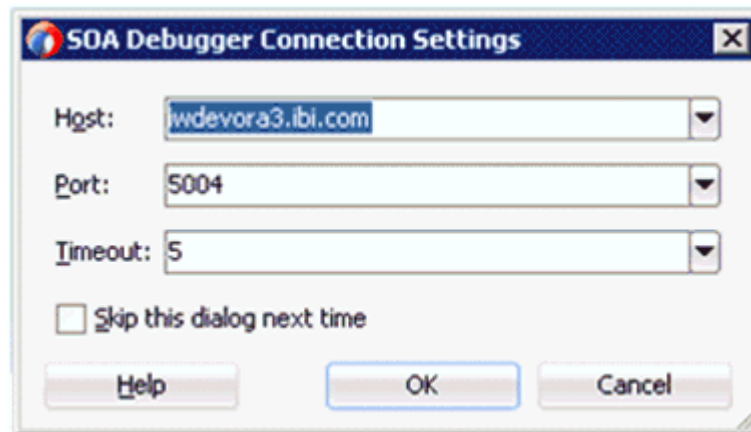


図 11-60 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 11-60 「SOA デバッガ接続設定」ダイアログ



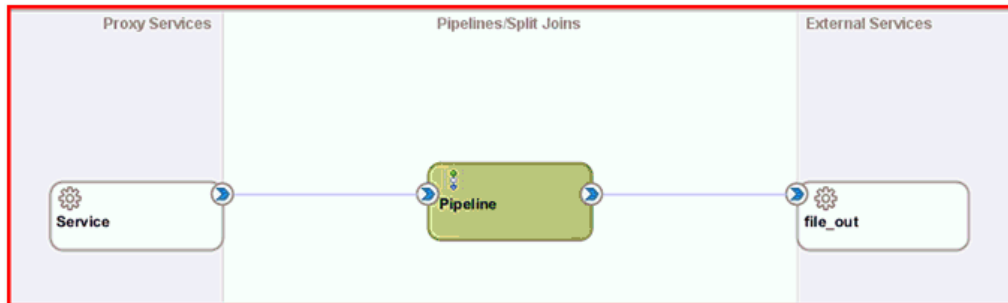
- b. 「OK」をクリックします。

OSB プロセスが SOA デバッガに接続されると、デバッグ・ログに次のメッセージが表示されます。

デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しようとしています。
 デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しました。
 デバッガ・プロセスの仮想マシンは SOA デバッガです。

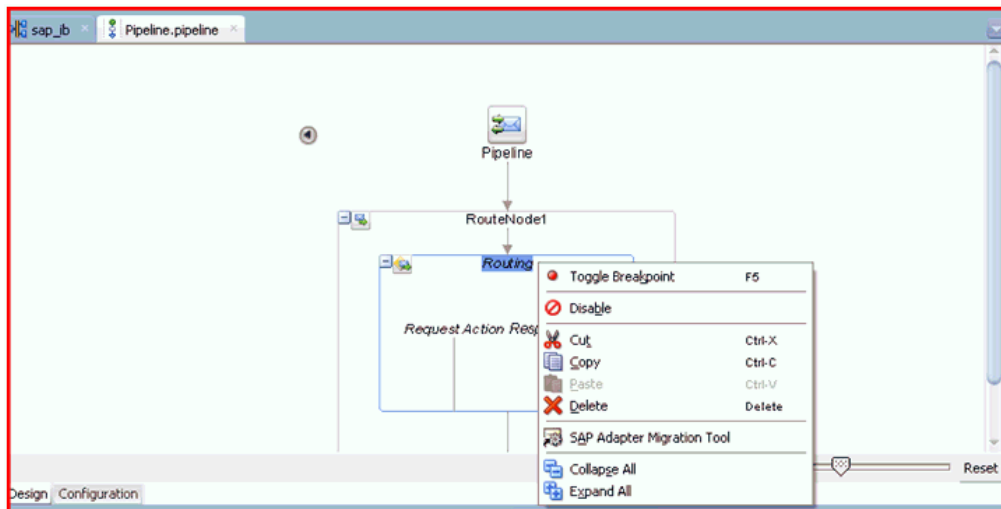
4. ブレークポイントを設定します。
 - a. 図 11-48 に示すように、パイプラインをダブルクリックしてエディタで開きます。

図 11-61 ProxyServicePipeline



- b. ブレークポイントを追加するノードが表示されるまでアクションを展開します。この例では、「ルーティング」が表示されるまで展開します。図 11-62 に示すように、「ルーティング」ノードを右クリックし、コンテキスト・メニューから「ブレークポイントの切替え」を選択します。

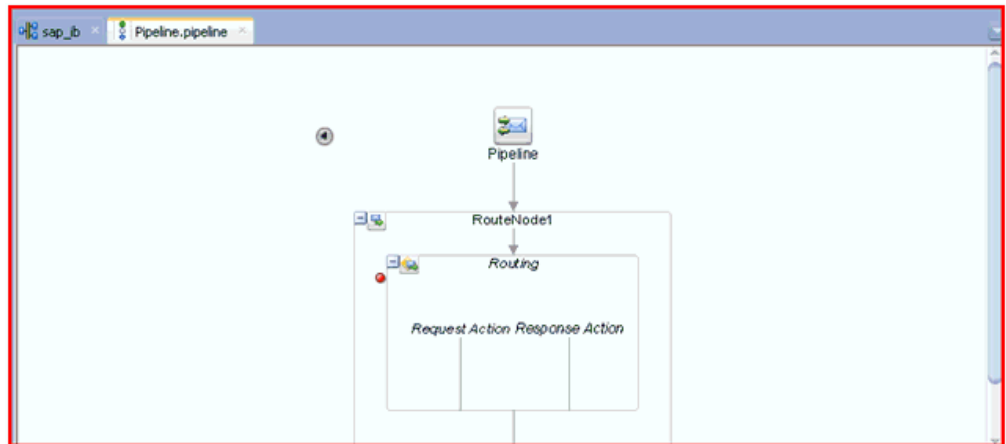
図 11-62 ブレークポイントの切替え



このステップを、ブレークポイントを追加する各ノードに対して繰り返します。この例では、ブレークポイントを設定するのは「ルーティング」ノードのみです。

図 11-63 に示すように、ブレークポイントが設定されたことを示す赤いアイコンがノードの横に表示されます。

図 11-63 設定されたブレイクポイントの赤いアイコン



注意: ブレイクポイントを無効にするには、ノードを右クリックして「ブレイクポイントの無効化」を選択します。

ブレイクポイントを削除するには、ノードを右クリックして再度「ブレイクポイントの切替え」を選択します。

5. デバッグを開始します。
 - a. 図 11-64 に示すように、「アプリケーション・ナビゲータ」でパイプラインを右クリックし、「デバッグ」を選択します。

図 11-64 「デバッグ」を選択

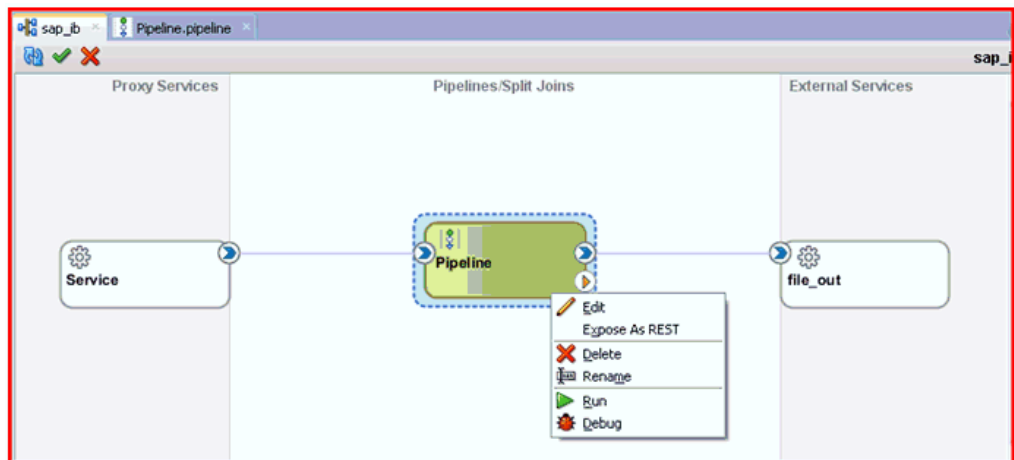
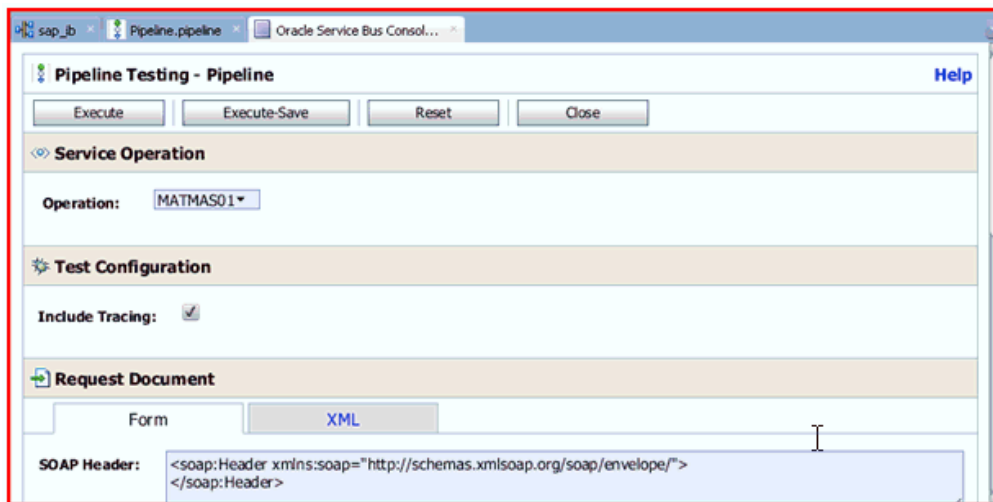


図 11-65 に示すように、プロセスが統合サーバーにデプロイされて、「構成のテスト」ペインが表示されます。

図 11-65 「構成のテスト」 ペイン



注意： 現在実行されているドメインがない場合は、「デフォルト・ドメインの作成」ダイアログが表示されます。統合サーバーの接続情報を入力し、「OK」をクリックします。このプロセスには数分かかる場合があります。


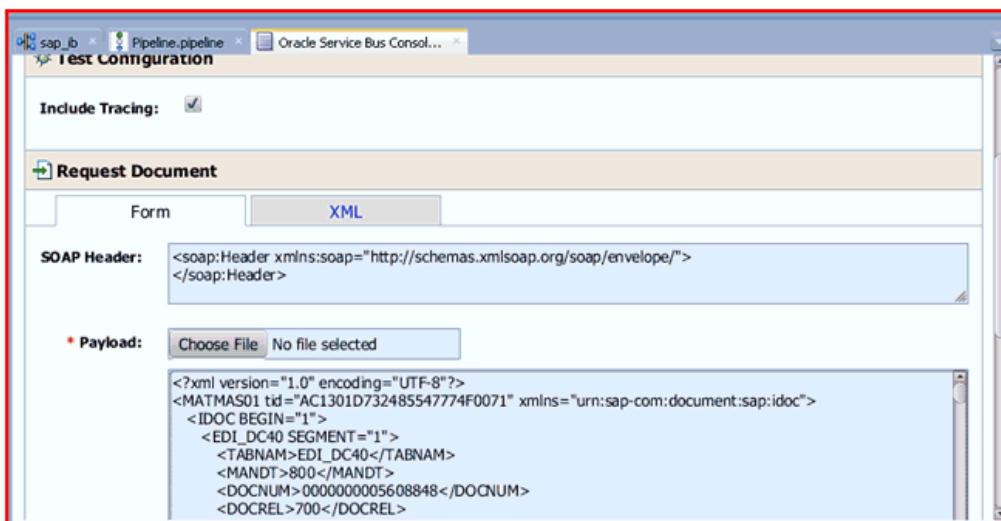
- b.  11-66 に示すように、「構成のテスト」ペインで、「リクエスト・ドキュメント」領域にテスト・データを入力し、必要に応じて追加の入力を構成します。

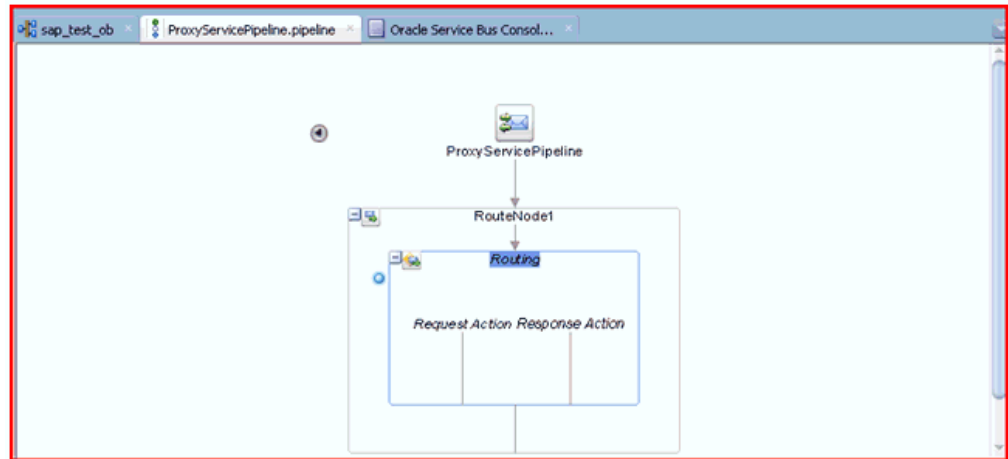
図 11-66 リクエスト・ドキュメントのテスト



- c. 「実行」をクリックします。

「構成のテスト」ペインでコマンドが実行されますが、 11-67 に示すように、指定したブレークポイントで OSB プロセスが停止し、青く点滅します。






図 11-67 OSB プロセスがブレークポイントで停止



- d. 図 11-68 に示すように、使用可能な「ステップ」オプションを使用してデバッグ・プロセスをステップ実行します。

図 11-68 「ステップ」オプション



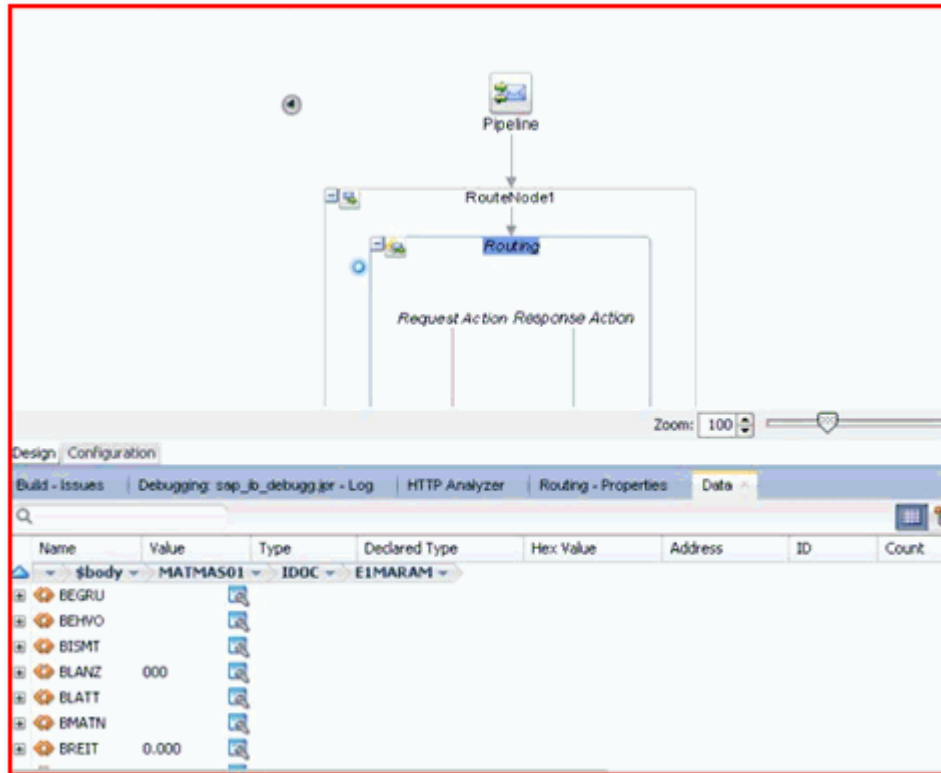
アイコン	説明
	デバッグ・セッションを終了または連結解除します。
	フレームをステップ実行します。 これにより、次のブレークポイント (ブレークポイントが設定された OSB プロセスの受信アクティビティなど) に移動します。ブレークポイントがない場合は、すべてのフレームをステップ実行して最初のフレームに戻ります。 [F8] を押してフレームをステップ実行することもできます。
	次の有効な位置までトレース実行します。 これは、新しいフレームである場合もあれば、同じフレームの別の位置である場合もあります。 [F7] を押してフレームをトレース実行することもできます。
	フレームをステップ・アウトします。 このオプションは、OSB の Scope アクティビティまたは Sequence アクティビティを処理するためにのみ使用されます。スコープの処理が完了すると、プロセスの次のスコープまたはアクティビティで一時停止します。[Shift] + [F7] を押すこともできます。
	ステップ操作を再開します。 [F9] を押して再開することもできます。

- e. 「ステップ実行」を使用して、プロセスの実行を完了します。

注意： この例にはブレークポイントが1つしかないため、「ステップ実行」を使用するとプロセスの実行が完了します。

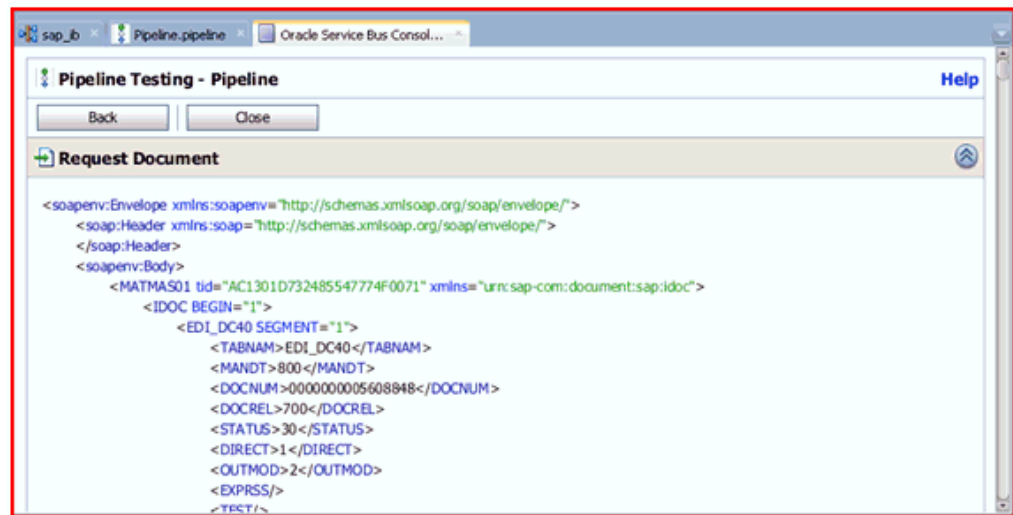
図 11-56 に示すように、「データ」タブでレスポンス・ドキュメントを確認できます。

図 11-69 「データ」タブの出力



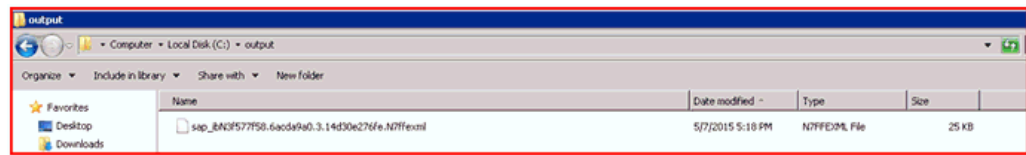
また、図 11-70 に示すように、「Oracle Service Bus コンソール」でレスポンスを確認することもできます。

図 11-70 Oracle Service Bus コンソール



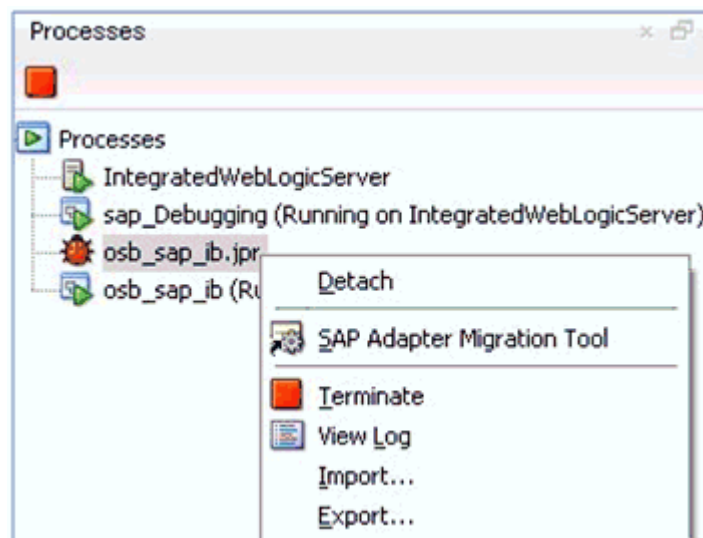
出力は、図 11-71 に示すように、構成済の出力場所にもあります。

図 11-71 構成済の出力場所



6. デバッグ・セッションを終了または連結解除します。
 - a. 「ウィンドウ」、「プロセス」の順にクリックします。図 11-72 に示すように、「プロセス」タブでプロセスを右クリックし、「連結解除」または「終了」を選択します。

図 11-72 デバッグ・セッションの連結解除



- b. 次のいずれかのオプションを選択します。

「**連結解除**」: デバッグ・プロセスを終了せずに SOA デバッガを削除します。

「**終了**」: デバッグ・プロセスを終了します。

11.4 例外フィルタの構成

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、次のトピックについて説明します。

■ 11.4.1 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。

11.4.1 例外フィルタの構成

この項では、次のトピックについて説明します。

- 11.4.1.1 項「WSDL ファイルの生成」
- 11.4.1.2 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」
- 11.4.1.3 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」
- 11.4.1.4 項「12c でのデプロイメントに関する既知の問題の調整」
- 11.4.1.5 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、11-45 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、11-45 ページ「[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、11-46 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、11-57 ページ「[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。

11.4.1.1 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。

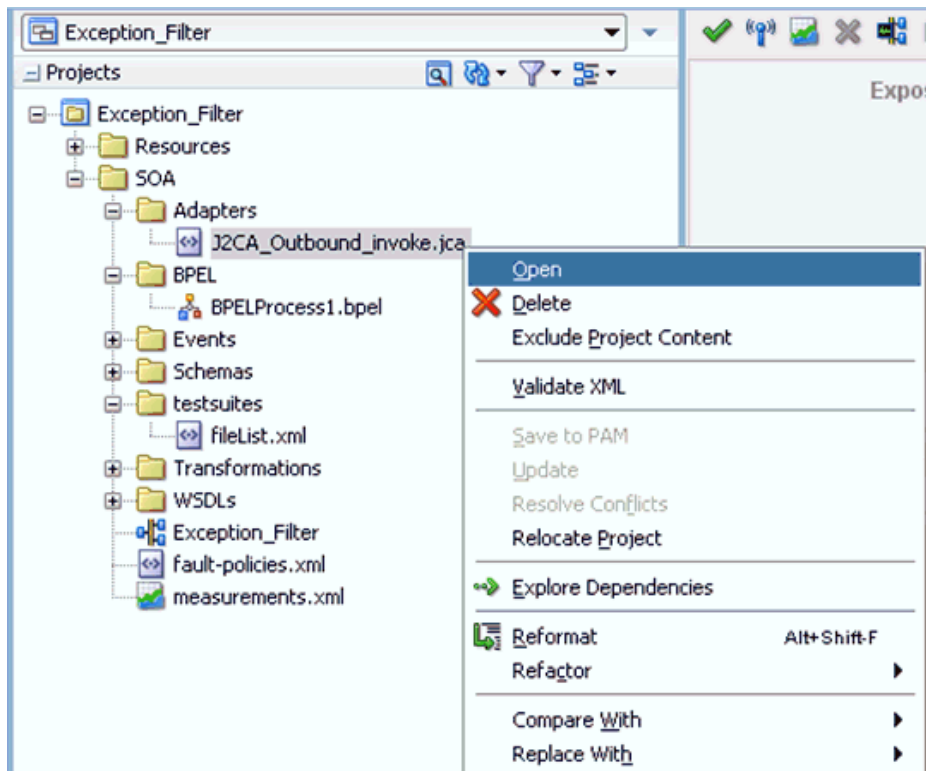
1. アプリケーション・エクスプローラを開き、J2CA 構成を作成します。
詳細は、4-6 ページの「[J2CA 用の構成の作成](#)」を参照してください。
2. SAP R/3 アダプタのターゲットを作成してから、そのターゲットに接続します。
詳細は、4-8 ページの「[SAP R/3 用の接続 \(ターゲット\) の確立](#)」を参照してください。
3. 適切なオブジェクト (CompanyCode ビジネス・オブジェクトの GetDetail BAPI など) の WSDL を生成します。
詳細は、4-18 ページ「[WSDL の生成 \(J2CA 構成のみ\)](#)」を参照してください。

11.4.1.2 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、6-9 ページ「[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、6-11 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、6-11 ページ「[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. [図 11-73](#) 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 11-73 「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.SAPExceptionFilter"/></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。

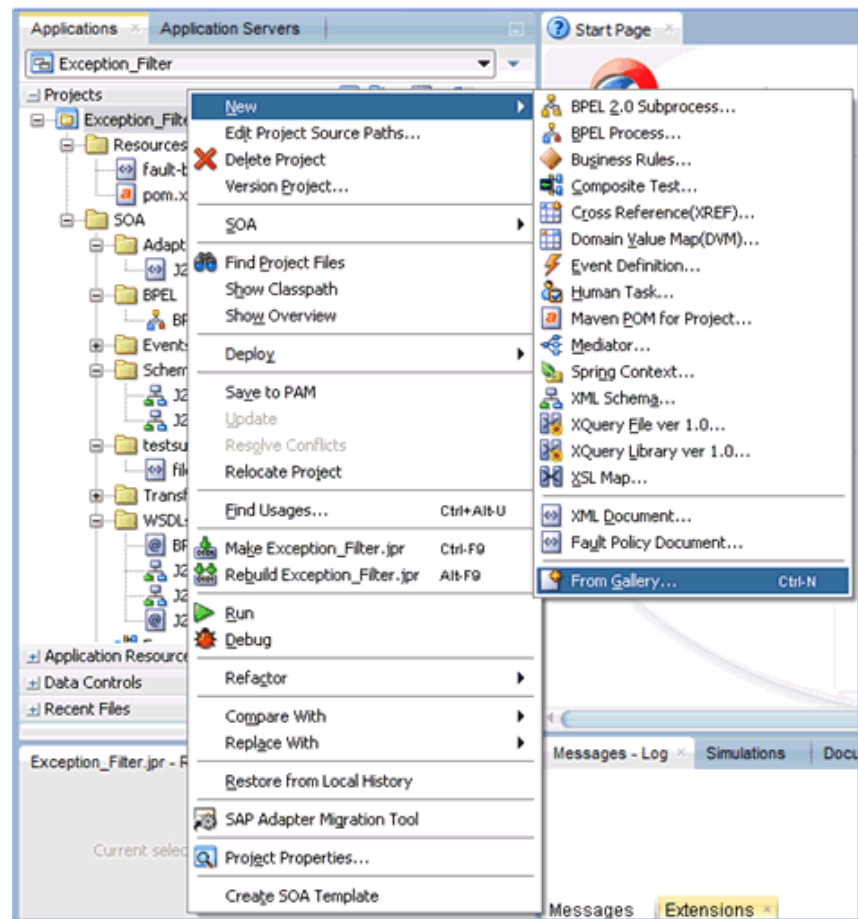
詳細は、6-11 ページ「[BPEL アウトバウンド・プロセスの定義](#)」を参照してください。

11.4.1.3 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。

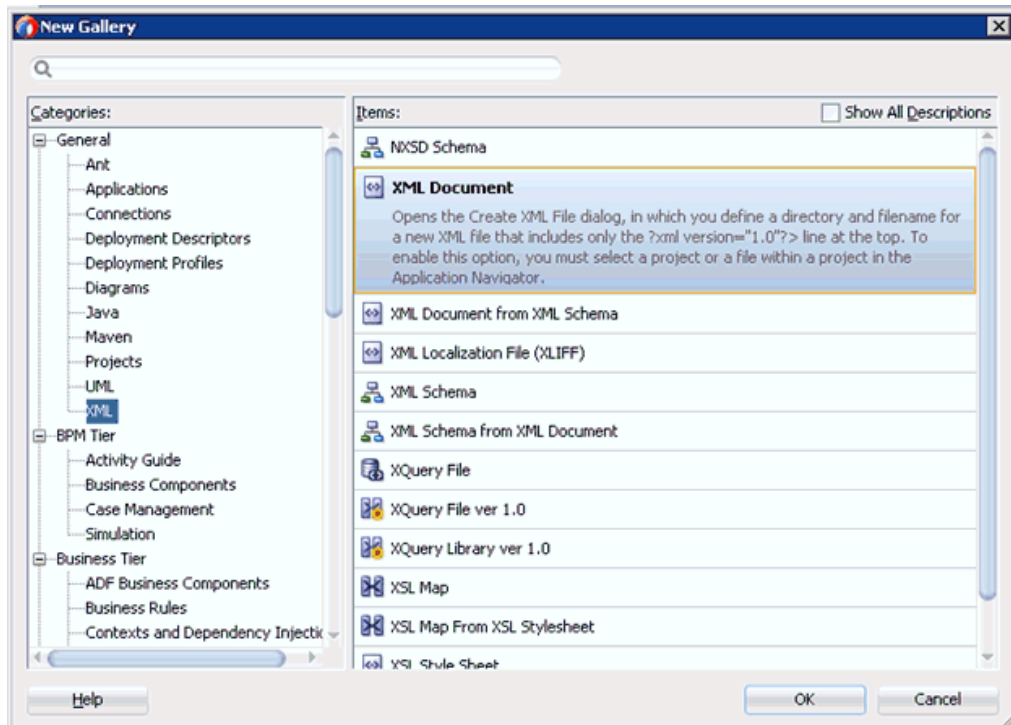
1. 図 11-74 に示すように、作成された SOA プロジェクト (Exception_Filter など) を右クリックして「新規」を選択し、「ギャラリーから」をクリックします。

図 11-74 「アプリケーション」タブ



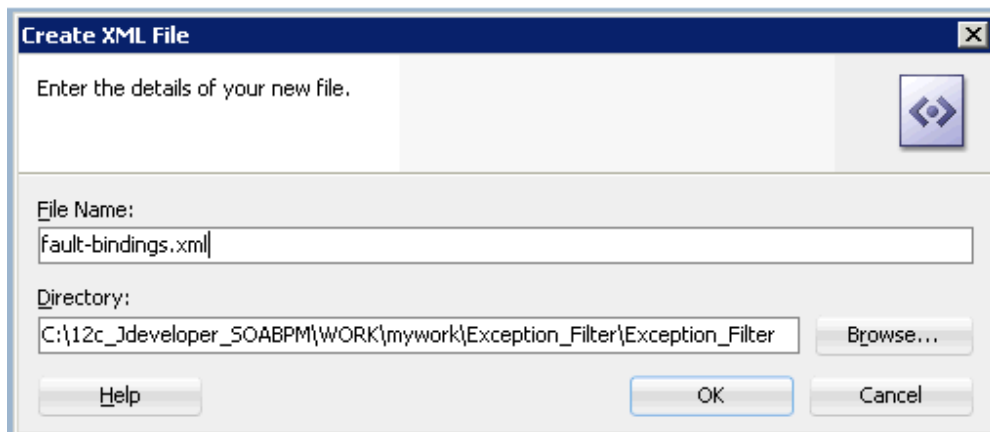
「新規ギャラリー」ダイアログが表示されます。図 11-75 に示すように、「一般」カテゴリの下に「XML」をクリックします。

図 11-75 「新規ギャラリー」 ダイアログ



- 「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 11-76 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 11-76 XML ファイルの作成ダイアログ



- 「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。
- 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、11-49 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>

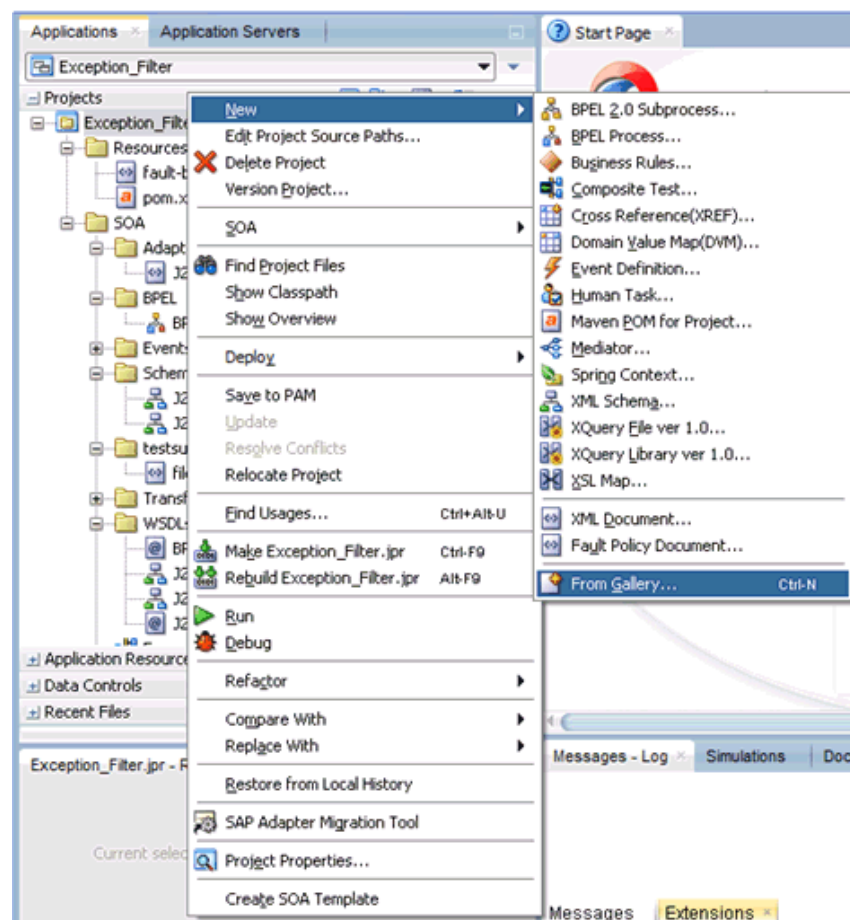
</faultPolicyBindings>
```

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。

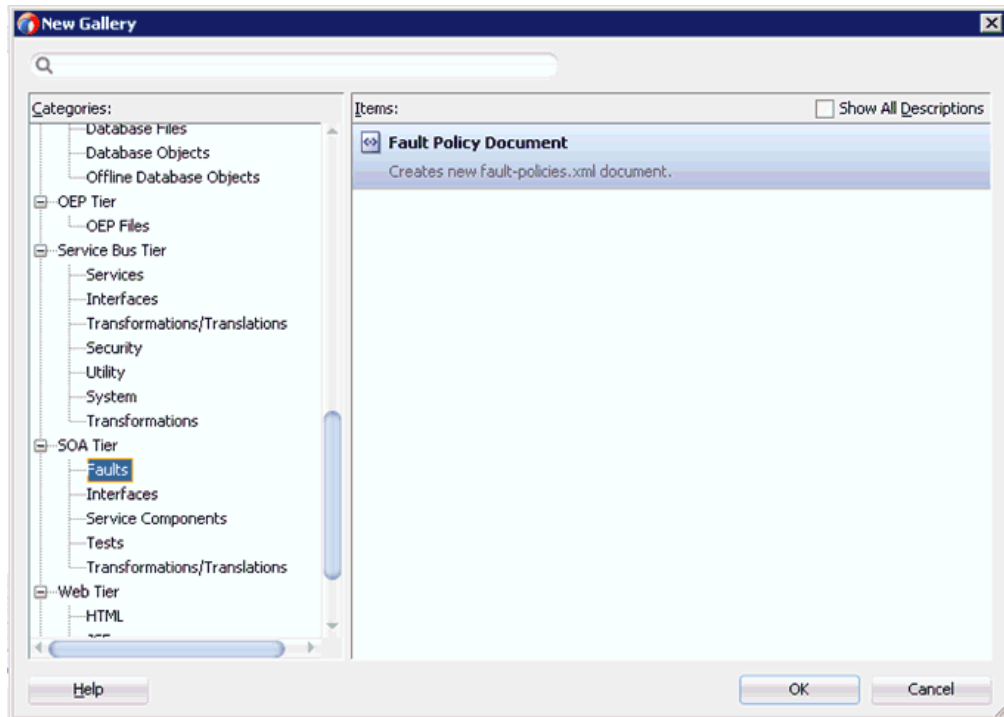
1. 図 11-77 に示すように、作成された SOA プロジェクト (Exception_Filter など) を右クリックして「新規」を選択し、「ギャラリーから」をクリックします。

図 11-77 「アプリケーション」タブ



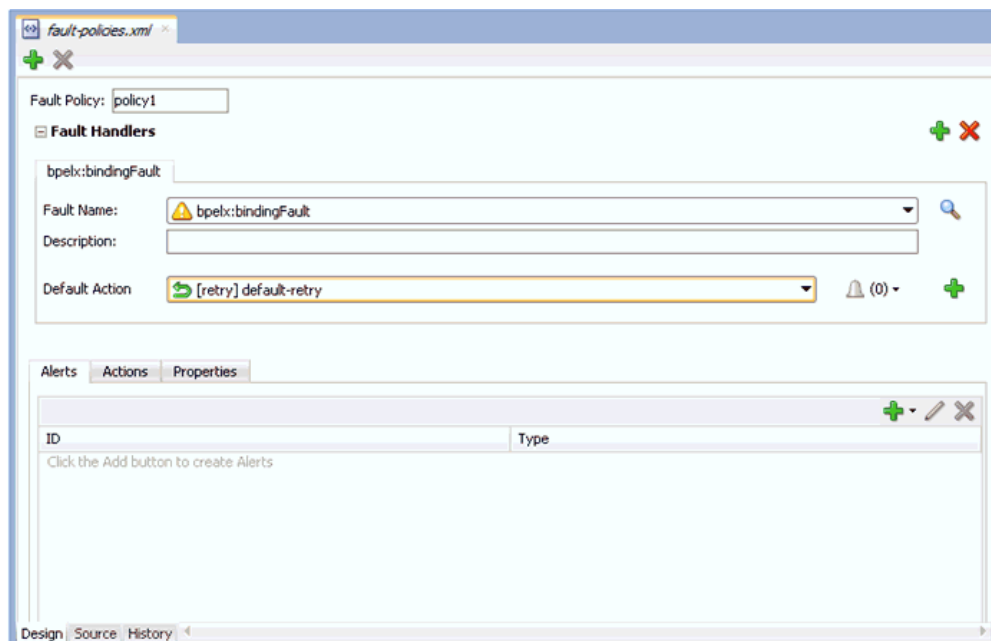
「新規ギャラリー」ダイアログが表示されます。図 11-78 に示すように、「SOA 層」カテゴリの下の「フォルト」を選択します。

図 11-78 「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 11-79 に示すように、「fault-policies.xml」タブの「フォルト名」ドロップダウン・リストで「**bpelx:bindingFault**」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「**[retry] default-retry**」を選択します。

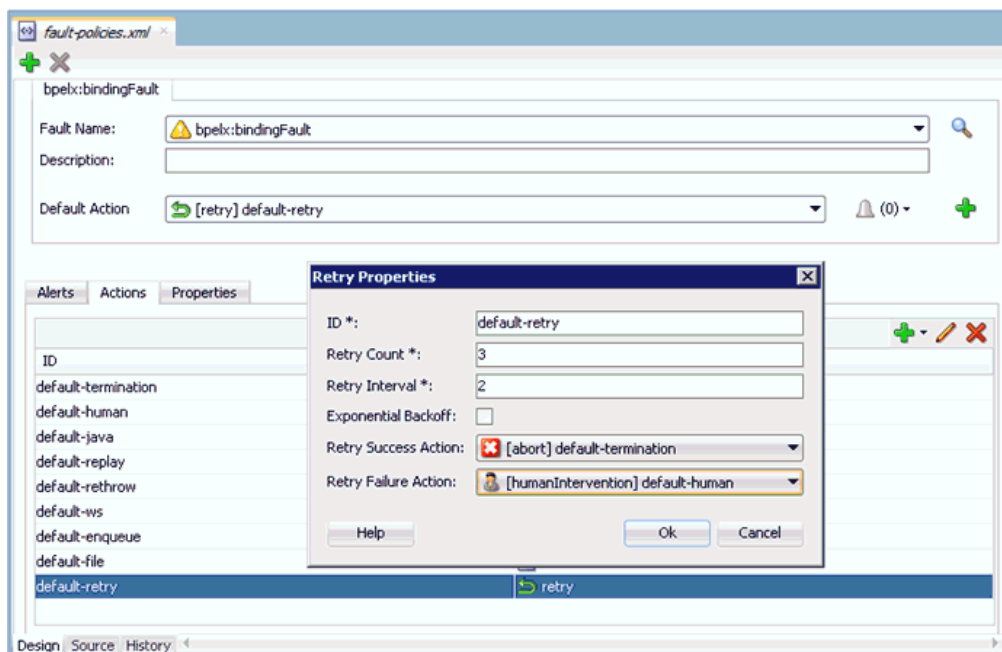
図 11-79 「fault-policies.xml」 タブ



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。

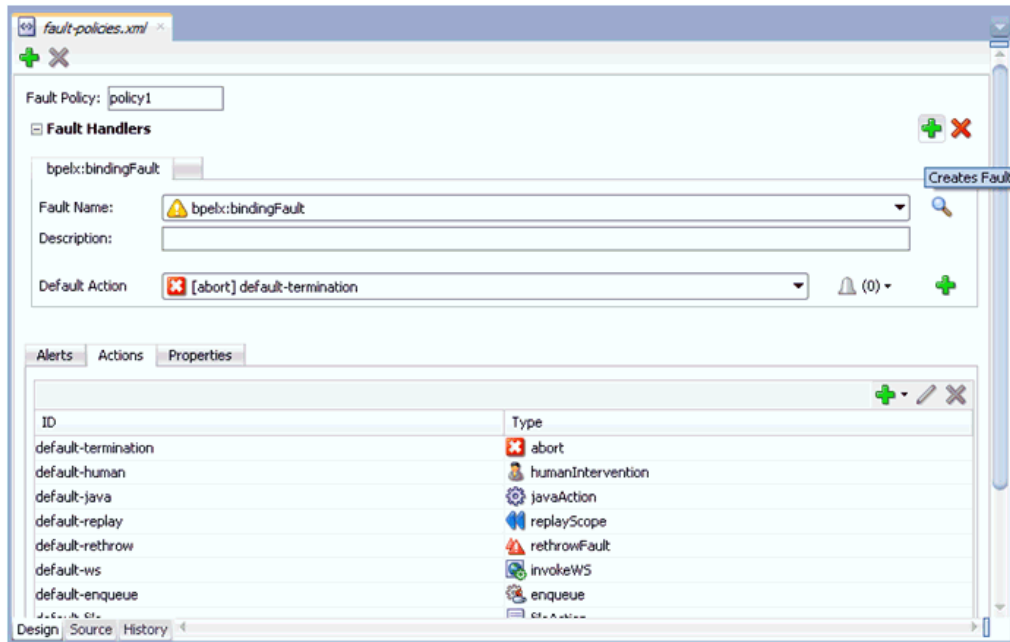
図 11-80 に示すように、「再試行プロパティ」ダイアログ・ボックスが表示されます。

図 11-80 「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. 図 11-81 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 11-81 「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 11-82 に示すように、「アクション」タブで「追加」をクリックし、「再試行」を選択します。

図 11-82 「アクション」タブ

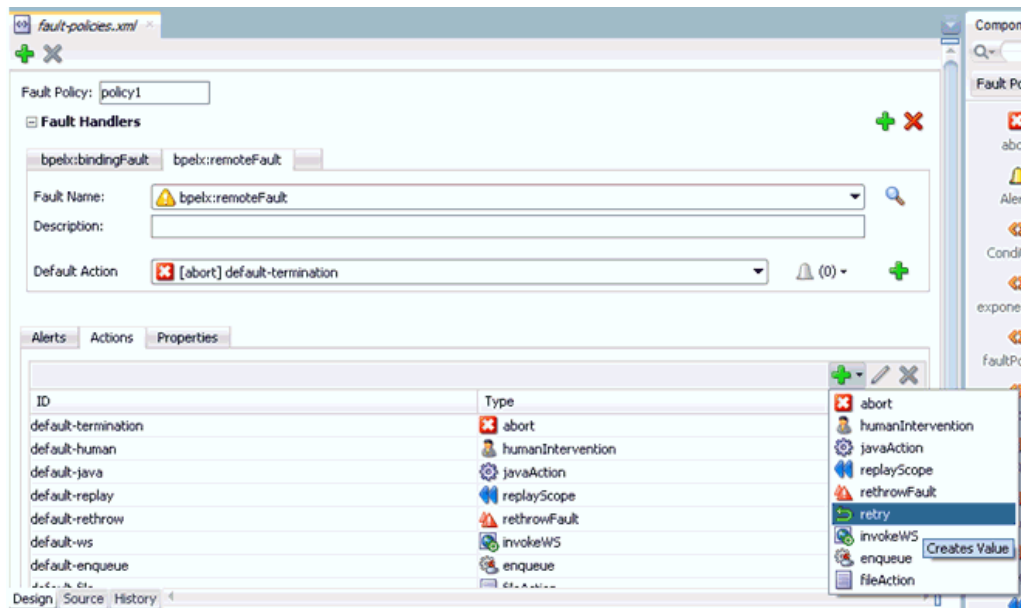
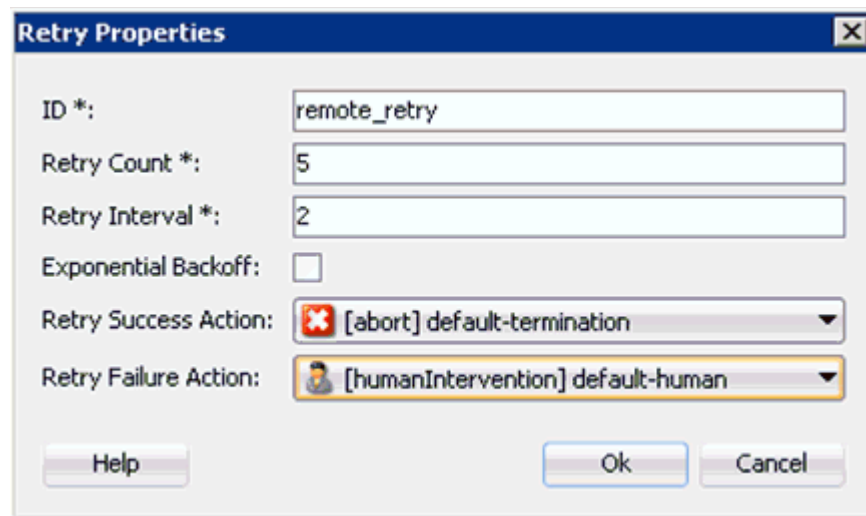


図 11-83 に示すように、「再試行プロパティ」ダイアログが表示されます。

図 11-83 「再試行プロパティ」ダイアログ・ボックス

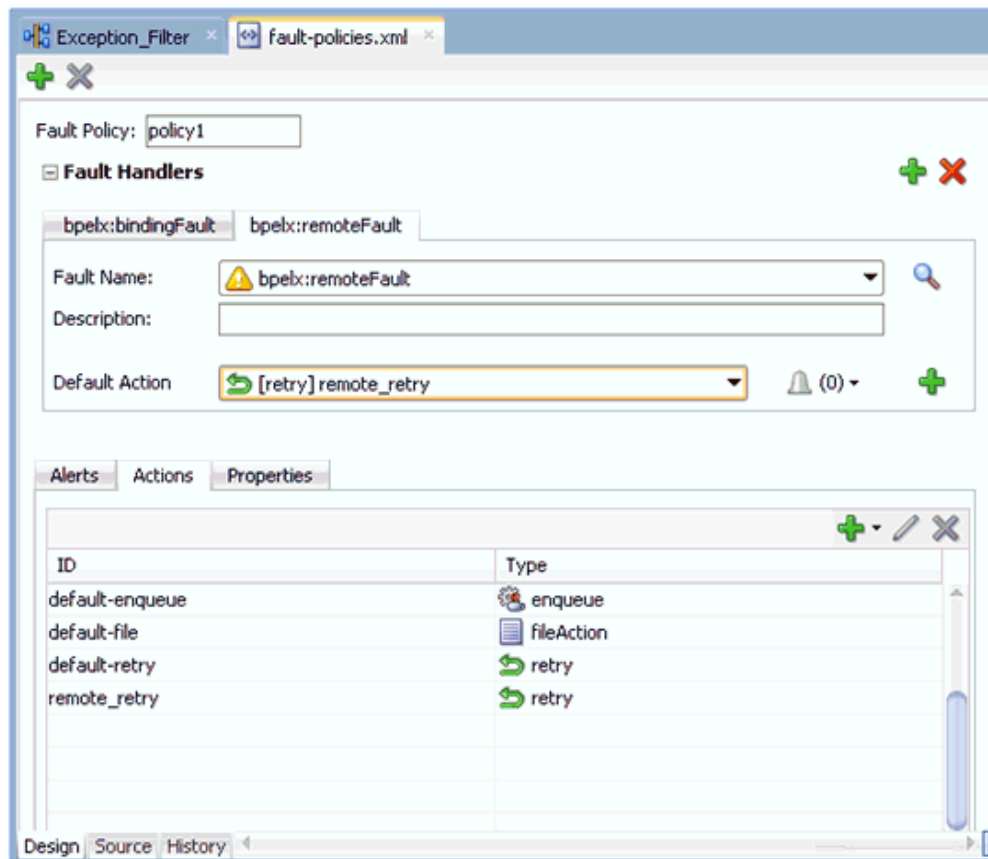


10. 「ID」、「再試行数」および「再試行間隔」の各フィールドに値を指定します。
11. 「再試行成功アクション」ドロップダウン・リストで「**[abort] default-termination**」を選択し、「再試行失敗アクション」ドロップダウン・リストで「**[humanIntervention] default-human**」を選択します。
12. 「OK」をクリックします。

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。

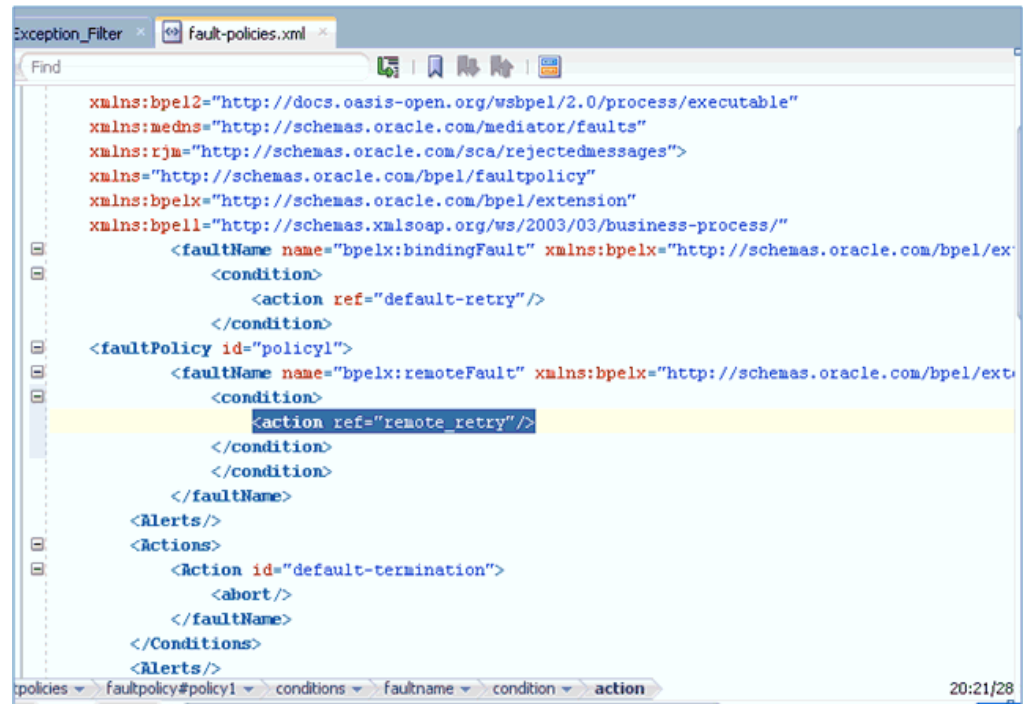
図 11-84 に示すように、「デフォルト・アクション」ドロップダウン・リストで新しく作成された再試行 ID(remote_retry など) を選択します。

図 11-84 「fault-policies.xml」 タブ



13. 「すべて保存」 をクリックします。
14. 図 11-85 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。

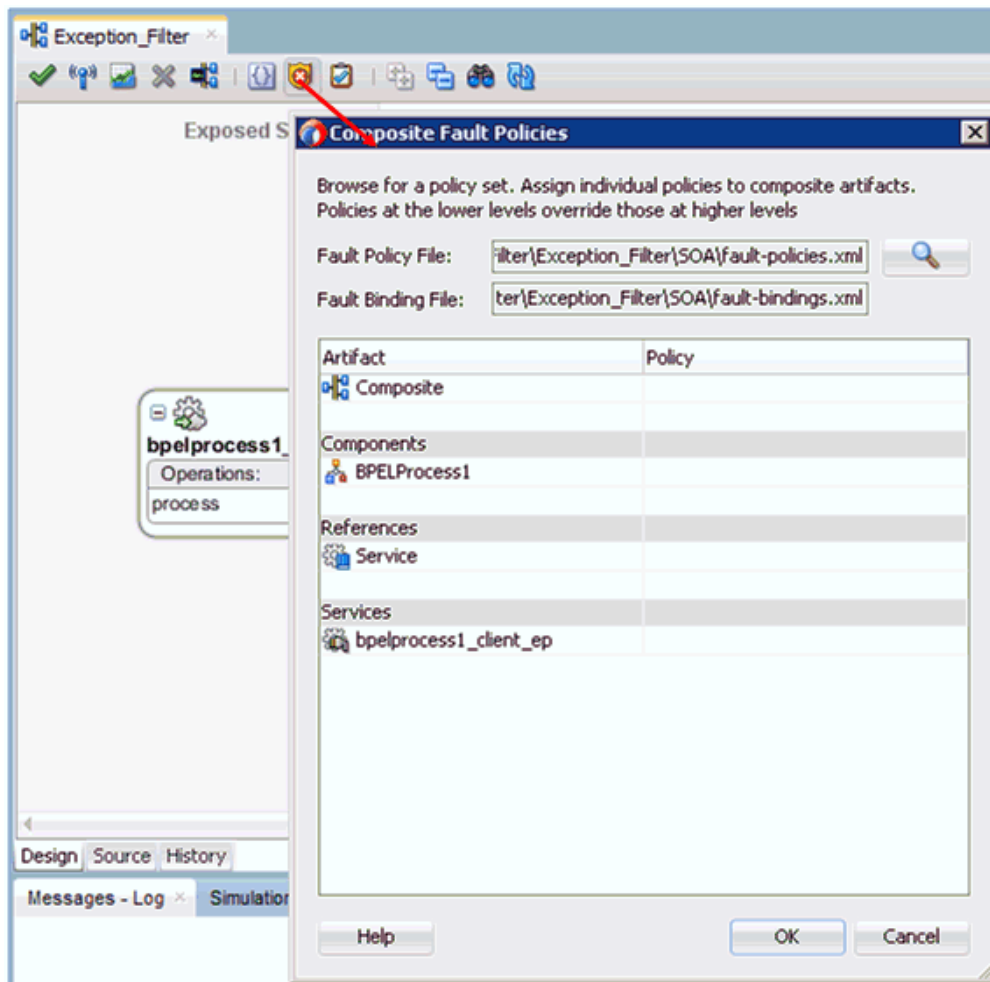
図 11-85 「ソース」タブ



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。

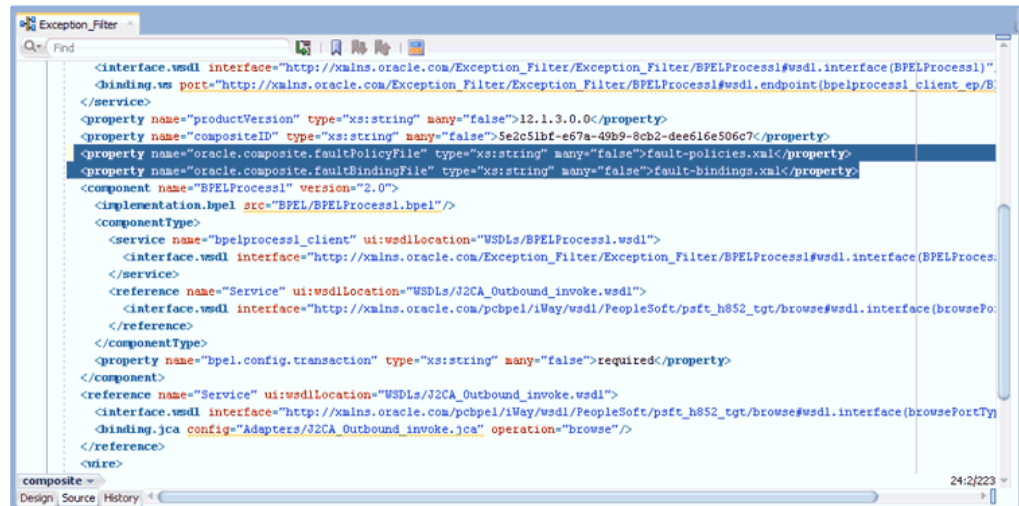
「コンポジット・フォルト・ポリシーの編集」ウィンドウが表示されます。図 11-86 に示すように、フォルト・ポリシーとフォルト・バインディングが正しく選択されていることを確認します。

図 11-86 「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 11-87 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 11-87 「ソース」タブ



11.4.1.4 12c でのデプロイメントに関する既知の問題の調整

12c での既知のデプロイメント問題の調整方法については、6-27 ページの 6.4.3.3 項「12c との既知のデプロイメント問題の調整」を参照してください。

11.4.1.5 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 BPE アウトバウンド・プロセスをデプロイします。
詳細は、6-30 ページ「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、6-33 ページ「Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動」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。

11.5 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、次のトピックについて説明します。

- 11.5.1 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 to 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。

11.5.1 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ます。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。
詳細については、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、11-58 ページの 11.5.1.1 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、11-61 ページの 11.5.1.2 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、11-61 ページ 11.5.1.3 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、11-62 ページ 11.5.1.4 項「EIS 資格証明の検証」を参照してください。

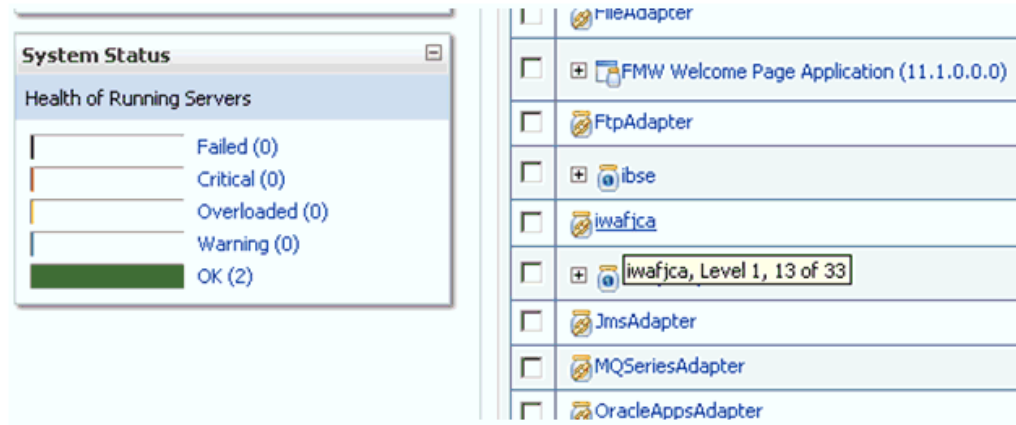
11.5.1.1 Oracle WebLogic Server 資格証明と EIS 資格証明の関連付け

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」セクションにある「デプロイメント」をクリックします。

図 11-88 に示すように、「デプロイメント」ページが表示されます。

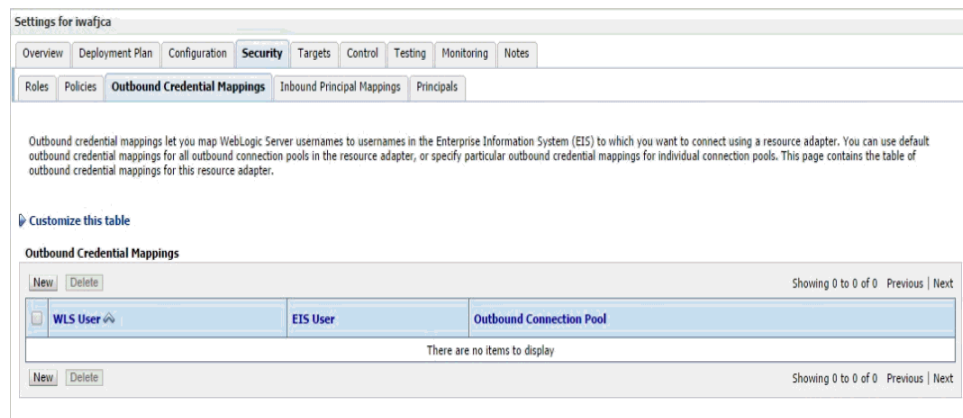
図 11-88 「デプロイメント」ページ



3. 「iwafjca」リソース・アダプタをクリックします。

図 11-89 に示すように、iwafjca の設定ページが表示されます。

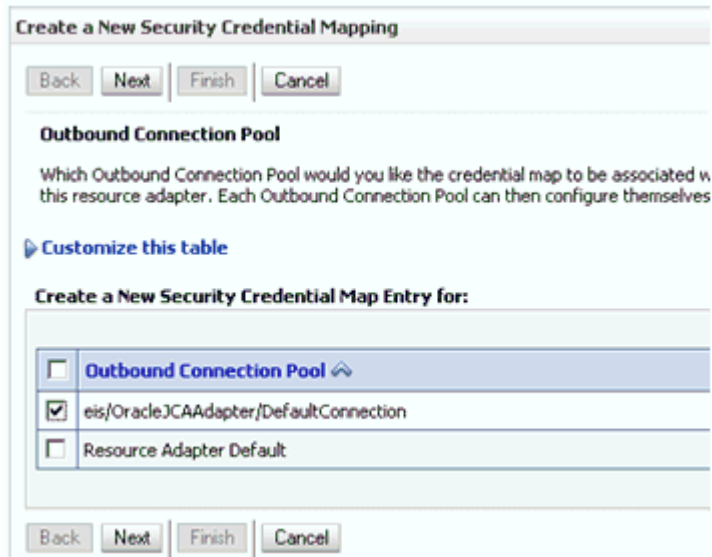
図 11-89 iwafjca の設定ページ



4. 「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。

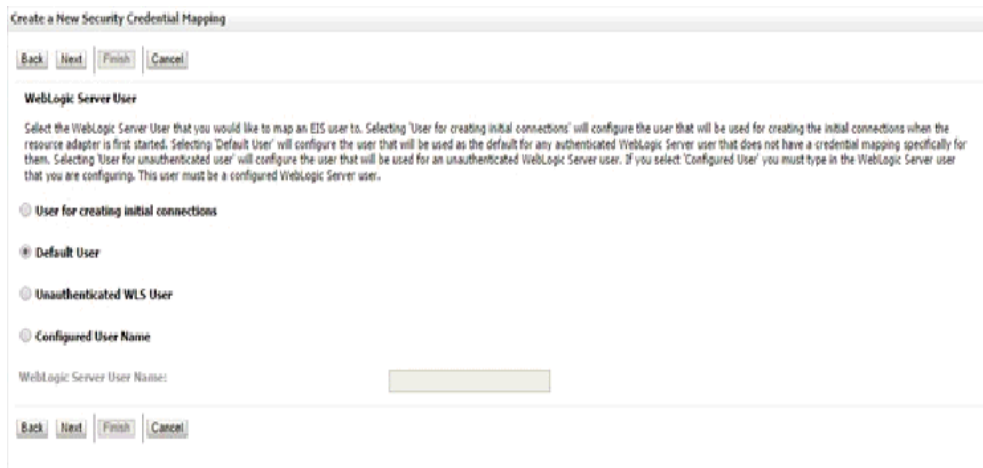
図 11-90 に示すように、「新規セキュリティ資格証明マッピングの作成」ページが表示されます。

図 11-90 「新規セキュリティ資格証明マッピングの作成」 ページ



5. 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
6. 「次へ」をクリックします。
図 11-91 に示すように、「WebLogic Server ユーザー」ページが表示されます。

図 11-91 「WebLogic Server ユーザー」 ページ



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、「次へ」をクリックします。
図 11-92 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 11-92 「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。

11.5.1.2 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、4-6 ページの 4.4.2 項「J2CA 用の構成の作成」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、4-8 ページの 4.5 項「SAP R/3 用の接続 (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、4-18 ページ 4.8 項「WSDL の生成 (J2CA 構成のみ)」を参照してください。

11.5.1.3 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、メディエータおよび 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。

メディエータ・アウトバウンド・プロセスの作成方法の詳細は、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。

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」

11.5.1.4 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、6-33 ページ「[Oracle Enterprise Manager コンソールでの入力 XML ドキュメントの起動](#)」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 IWAManagedConnectionFactory com.ibi.afjca.Util
getPasswordCredential(78) InLoop:
User-iwayqa:Password-ENCR(3109311731831131382333215315332323192322731773172)
FINEST IWAManagedConnectionFactory com.ibi.afjca.Util
getPasswordCredential(90) Use the system PasswordCredential:
User-iwayqa:Password-ENCR(3109311731831131382333215315332323192322731773172)
```

11.6 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、次のトピックについて説明します。

- 11.6.1 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。

11.6.1 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。
詳細については、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、11.6.1.1 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、11.6.1.2 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。
詳細については、11.6.1.3 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。

11.6.1.1 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 11-93 に示すように、左ペインの「ドメイン構造」セクションにある「デプロイメント」をクリックします。

図 11-93 「ドメイン構造」セクション

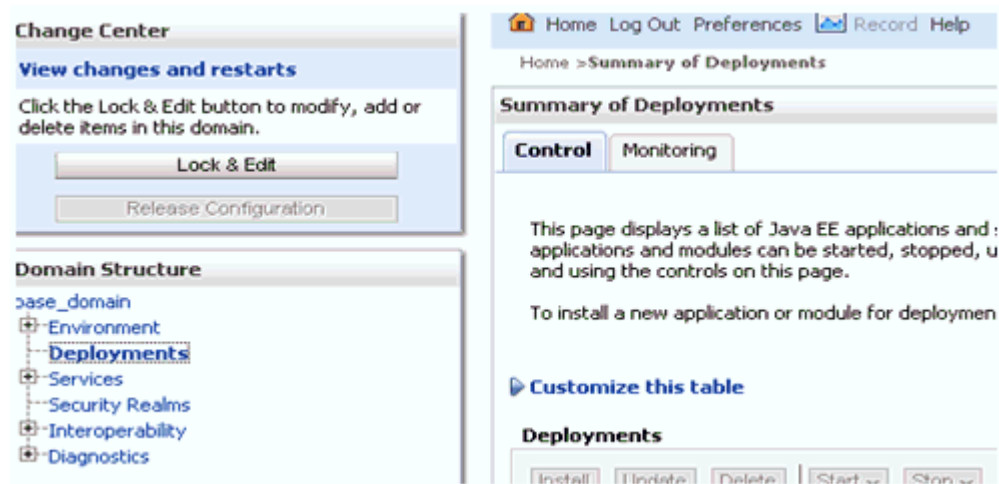
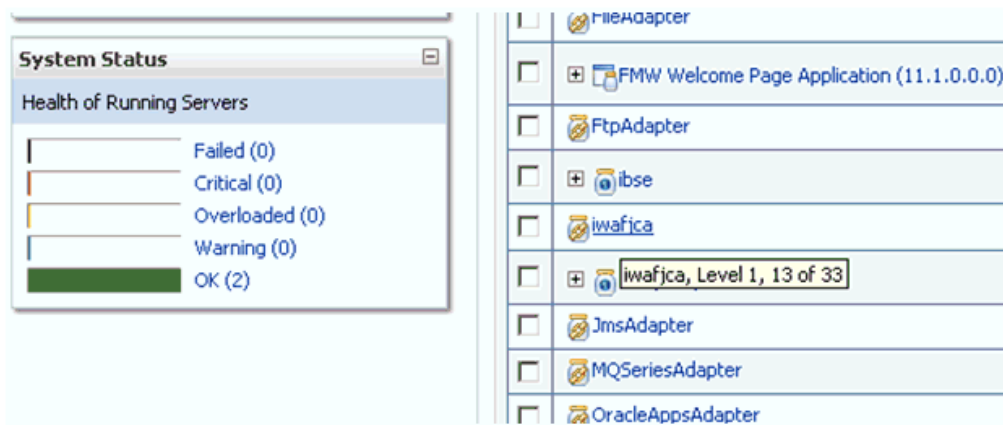


図 11-94 に示すように、「デプロイメント」ページが表示されます。

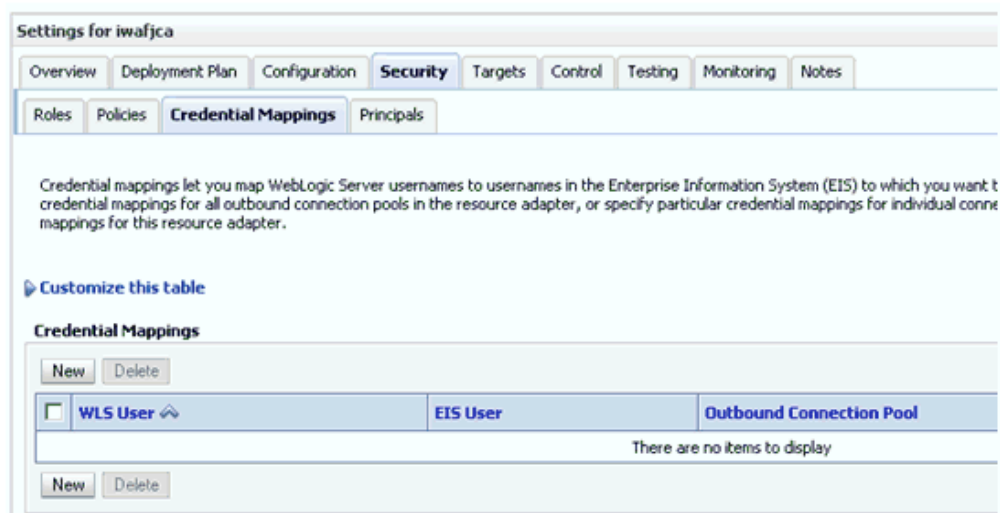
図 11-94 「デプロイメント」ページ



3. 「iwafjca」リソース・アダプタをクリックします。

図 11-95 に示すように、iwafjca の設定ページが表示されます。

図 11-95 iwafjca の設定ページ



4. 「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、「新規」をクリックします。

図 11-96 に示すように、「新規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 11-96 「新規セキュリティ資格証明マッピングの作成」 ページ

The screenshot shows a dialog box titled "Create a New Security Credential Mapping". At the top, there are buttons for "Back", "Next", "Finish", and "Cancel". Below this is the "Outbound Connection Pool" section, which contains a text box explaining that the user should select an Outbound Connection Pool for the credential map. A link "Customize this table" is provided. Below the link is a table titled "Create a New Security Credential Map Entry for:". The table has three rows:

<input type="checkbox"/>	Outbound Connection Pool
<input checked="" type="checkbox"/>	eis/OracleJCAAdapter/DefaultConnection
<input type="checkbox"/>	Resource Adapter Default

At the bottom of the dialog, there are buttons for "Back", "Next", "Finish", and "Cancel".

5. 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
6. 「次へ」をクリックします。
図 11-97 に示すように、「WebLogic Server ユーザー」ページが表示されます。

図 11-97 「WebLogic Server ユーザー」 ページ

The screenshot shows a page titled "Create a New Security Credential Mapping" with a sub-section "WebLogic Server User". It contains a text box explaining the purpose of selecting a WebLogic Server User. Below this are four radio button options:

- User for creating initial connections
- Default User
- Unauthenticated WLS User
- Configured User Name

Below the radio buttons is a text field labeled "WebLogic Server User Name:" with the value "weblogic" entered. At the bottom, there are buttons for "Back", "Next", "Finish", and "Cancel".

7. 「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、「次へ」をクリックします。
図 11-98 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 11-98 「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。

11.6.1.2 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「[Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定](#)」を参照してください。
2. 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、4-6 ページの「[J2CA 用の構成の作成](#)」を参照してください。
3. 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、4-8 ページの「[SAP R/3 用の接続 \(ターゲット\) の確立](#)」を参照してください。
4. 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「[アプリケーション・エクスプローラから Oracle Service Bus への WSDL の公開](#)」を参照してください。

11.6.1.3 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、10-31 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、10-27 ページ 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。

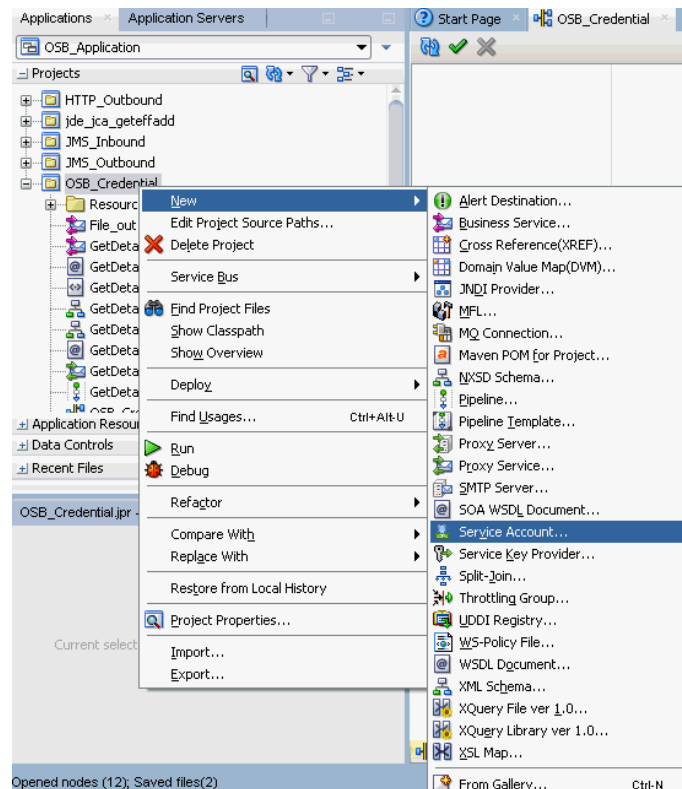
詳細は、10-35 ページ 10.3.2.2 項「パイプラインを含むプロキシ・サービスの作成」を参照してください。

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。

詳細については、を参照してください。10-29 ページ 10.2.2.4 項「ルーティング・ルールの構成」

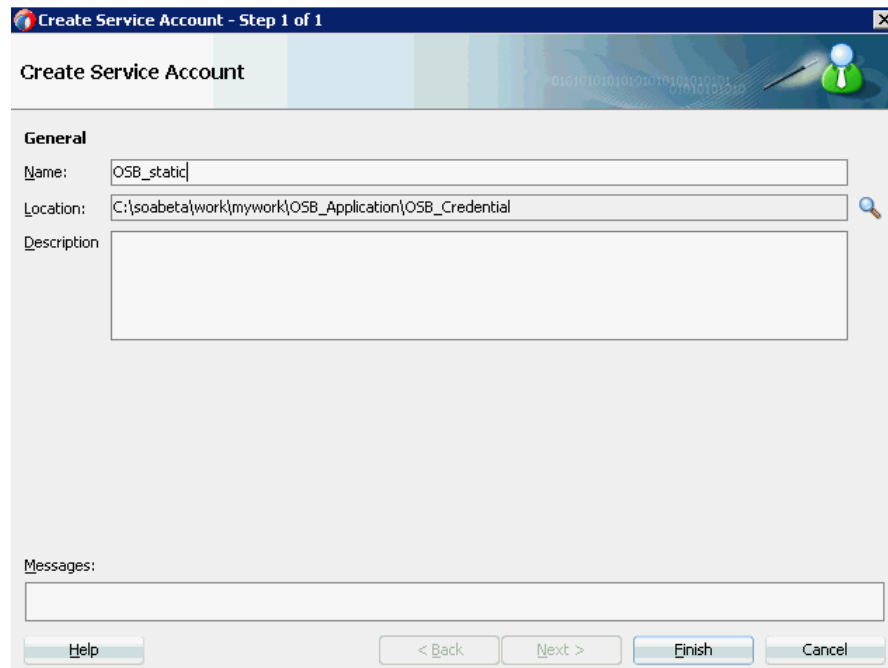
7. サービス・アカウントを構成するには、[図 11-99](#) に示すように、OSB プロジェクトを右クリックして「新規」を選択し、「サービス・アカウント」をクリックします。

図 11-99 「サービス・アカウント」オプションの選択



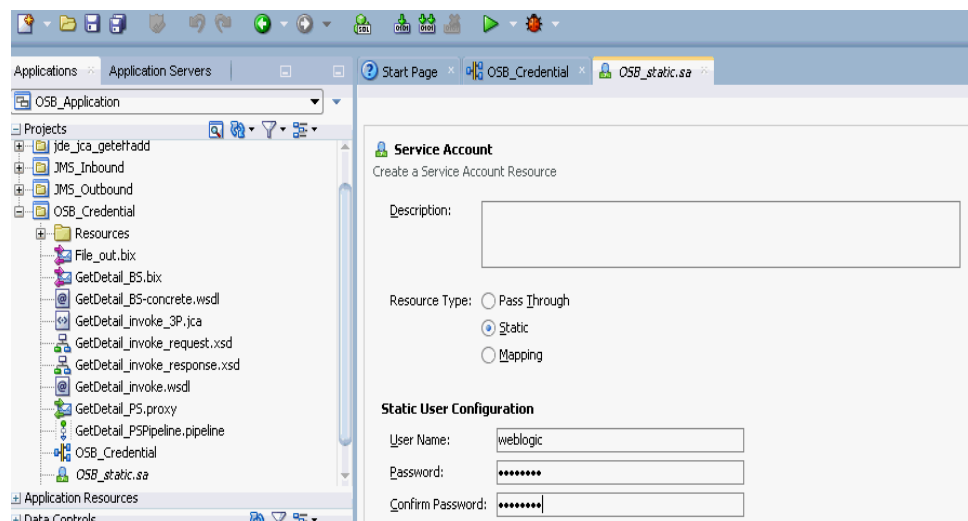
[図 11-100](#) に示すように、サービス・アカウントの作成ペインが表示されます。

図 11-100 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 11-101 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。

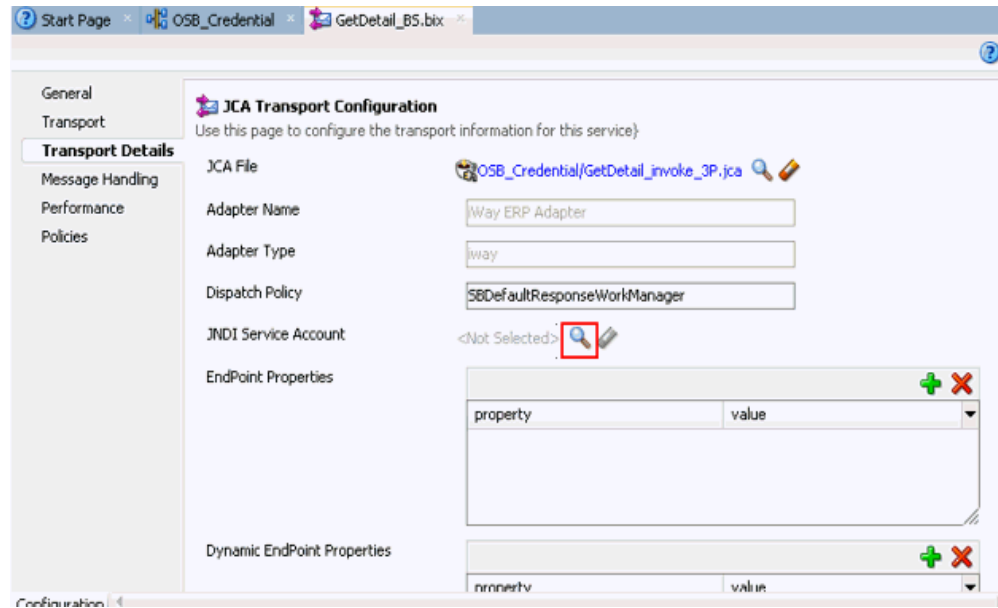
図 11-101 「サービス・アカウント構成」ページ



11. 「構成」ページを保存して閉じます。
12. 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。

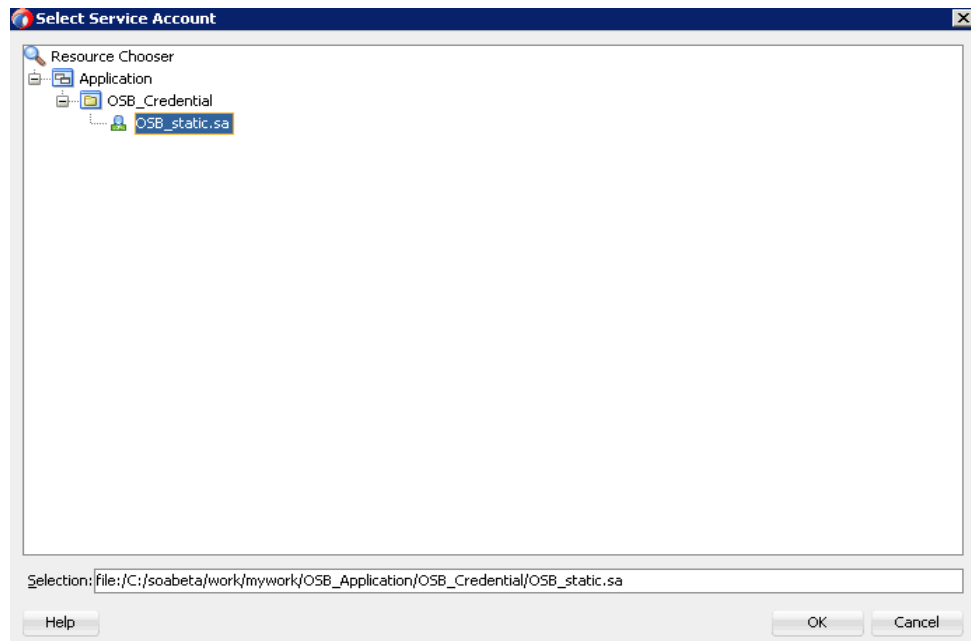
13. 図 11-102 に示すように、「トランスポートの詳細」タブを選択します。

図 11-102 「トランスポートの詳細」タブ



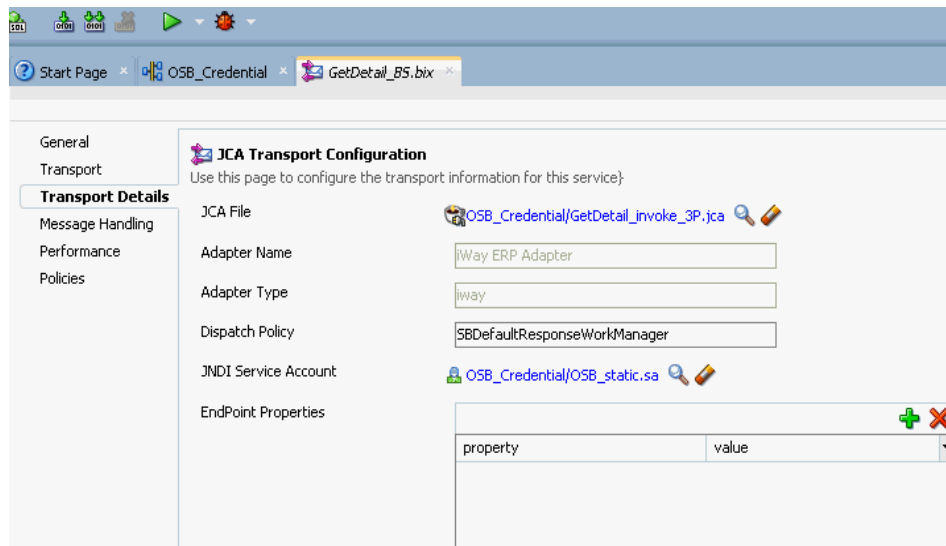
14. 「JNDI サービス・アカウント」セクションで、「参照」アイコンをクリックします。
- 「サービス・アカウントの選択」ウィンドウが表示されます。
15. 図 11-103 に示すように、作成したサービス・アカウントを選択し、「OK」をクリックします。

図 11-103 「サービス・アカウントの選択」



16. 図 11-104 に示すように、構成ページを保存して閉じます。

図 11-104 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 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)
```

11.7 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。

- 11.7.1 項「概要」
- 11.7.2 項「フィルタの有効化」
- 11.7.3 項「置換文字の指定」
- 11.7.4 項「設定ファイルの保存およびサーバーの再起動」
- 11.7.5 項「フィルタの検証」
- 11.7.6 項「クイック・スタート・ガイド」
- 11.7.7 項「既知の制限」

11.7.1 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 文字のすべてのインスタンスがすべて、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。

11.7.2 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。

11.7.3 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: `<substitutionChar></substitutionChar>` 要素タグの `mysap.settings.xml` ファイル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。

11.7.4 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。

11.7.5 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。

11.7.6 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. 11-71 ページの 11.7.3 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。

11.7.7 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。

- 11.7.7.1 項「アダプタ全体でのインストール」
- 11.7.7.2 項「全置換の制限」
- 11.7.7.3 項「SAP システム・タイプ」
- 11.7.7.4 項「文字制限」
- 11.7.7.5 項「ドキュメント・タイプおよび方向」
- 11.7.7.6 項「空白ルールの表示」

- 11.7.7.7 項「SAX パーサー」
- 11.7.7.8 項「置換文字タイプ」
- 11.7.7.9 項「置換文字数」
- 11.7.7.10 項「フォントの使用状況」
- 11.7.7.11 項「空白文字の表示」

11.7.7.1 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。

11.7.7.2 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のすべてのインスタンスを置換します。詳細は、11-71 ページ 11.7.3 項「置換文字の指定」を参照してください。

11.7.7.3 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。

11.7.7.4 文字制限

この機能は、このドキュメント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。

11.7.7.5 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。

11.7.7.6 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、11-74 ページ 11.7.7.11 項「空白文字の表示」を参照してください。

11.7.7.7 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。

11.7.7.8 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ます、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。

11.7.7.9 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。

11.7.7.10 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。

11.7.7.11 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、[図 11-105](#) 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 11-105 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約事項および回避策について説明します。次のトピックが含まれています。

- 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」

この章に記載しているアダプタ特定のエラーは、Oracle Adapter J2CA または Oracle Adapter Business Services Engine (BSE) のいずれかの構成で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 4 つのトピックで 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\iwaeb\bin
```

注意: SEVERE

IWAFManagedConnectionFactory
 com.iwaysoftware.af.container.channel.Channel
 onMessage (277) このチャンネルのサブスライバは存在しません。
 イベントが破棄されました。メッセージ：
 java.lang.IllegalStateException: メッセージをリスナーに
 配信できませんでした

上記のエラーは、アプリケーション・エクスペローラのチャンネルが
 メッセージを受信できない不正な状態にあるため、例外となったこと
 を意味します。

この問題を解決するには、以下に従ってください。

- イベントのランタイム・デプロイメントの前に、すべてのチャネ
 ルを停止し、アプリケーション・エクスペローラを閉じます。
- AE のインスタンスが実行中であり、メッセージを処理するために
 デプロイされた BPM、BPEL または OSB プロセスがない場合、
 SAP から送信された IDoc は、「不正な状態の例外」によって
 SAP に返されますが、適切なプロセス・ハンドラをデプロイする
 と、トランザクション SM58 から再送信できるようになります。

12.1.1 アプリケーション・エクスペローラ

ここでは、アプリケーション・エクスペ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	ソリューション
SAP R/3 がアプリケーション・エクスペロー ラの「アダプタ」ノード・リストに表示され ません。	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。
アプリケーション・エクスペローラから Oracle Application Adapter for SAP R/3 に接 続できません。 アダプタのアクティブ化での問題	次の点を確認します。 <ul style="list-style-type: none"> ■ SAP R/3 が動作していること ■ 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 ■ SAP R/3 のユーザー ID とパスワードが適 切であること
アプリケーション・エクスペローラ経由で 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。	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。
Error getting target [SAP] - java.lang.Exception: Error Logon to SAP System	

エラー	ソリューション
<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.SapAdapterException: 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 が動作中であること、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てください。</p>
<p>SAP R/3 システムが動作中であっても、アプリケーション・エクスプローラ経由で SAP R/3 にシステム接続できません。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.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 [no sapjco3 in java.library.path]. java.library.path </pre>	<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てください。</p>
<p>dll が別のクラスローダに読み込まれています (BSE と J2CA は同じサーバー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>

エラー	ソリューション
<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.<init>(JCATransport.java:62) at com.iwaysoftware.iwae.common.AdapterClient.<init>(AdapterClient.java:85) at com.ibi.bse.ConfigWorker.run(ConfigWorker.java:41) at java.lang.Thread.run(Thread.java:534) Could not create the connection factory. 実行時にログオン失敗エラーが発生します。 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。 java.lang.ClassNotFoundException: org.bouncycastle.jce.provider.BouncyCastleProvider </pre>	<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=<jdk_home>/bin/java, where <jdk_home> 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>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>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	ソリューション
<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>

エラー	ソリューション
<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 文書の構文をチェックし、ビジネス・オブジェクト型が SAP R/3 に存在して正しいこと、およびアクティブ化されていることを確認してください。また、文書名が正しいことを確認してください。</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 文書の構文をチェックし、BAPI の名前が正しいこと、SAP R/3 で利用可能なこと、およびアクティブ化されていることを確認してください。</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>java.lang.RuntimeException: com.sap.mw.jco.JCO\$AbapException: (126) OBJECT_UNKNOWN: Basic type or extension does not exist.</pre>	<p>入力 XML 文書の構文をチェックし、IDoc 型または拡張型が正しいこと、SAP R/3 で利用可能なこと、およびアクティブ化されていることを確認してください。</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: 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	ソリューション
<p>Oracle Adapter J2CA 構成への接続を試行すると、アプリケーション・エクスプローラで次のエラー・メッセージが表示されます。</p> <pre>Could not initialize JCA</pre>	<p>次の例に示すように、右ペインの「詳細」タブで、「ホーム」フィールドで正しいディレクトリが指定されていることを確認します。</p> <pre><ADAPTER_HOME>\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 사용되는インター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 が無効な SOAP リクエストを受信し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 は、この Web サービスに必須の SOAP リクエスト・メッセージで要素を受信しませんでした。

12.2.2 アダプタ特定のエラー処理

この節では、次のトピック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・システムのインターフェースと 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・システムのインターフェースまたは 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、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。

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・プロセスにおける SAP のシステム・ロールについて

Oracle Application Adapter for SAP R/3 は、3 種類のメッセージ・タイプ・シナリオと 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。各ロールには構成オプションがあり、通信を有効化してメッセージを送信するには、これらのオプションを SAP サーバー上とアダプタ上で有効化する必要があります。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 の実動システム環境では、特定の構成ロールを担当する適切な管理者のみが、通信エントリの変更や作成を行う権限を持ちます。この付録では、必要なタスクの管理者用のチェックリストを示します。また、インバウンドおよびアウトバウンド・メッセージ処理の全体的な構成プロセスについても説明します。この情報は、非実動システムの構成時に利用でき、また参照資料としても役立ちます。

この付録の内容は以下のとおりです。

- [A.1 項「アダプタの機能」](#)
- [A.2 項「SAP インバウンド・メッセージの処理」](#)
- [A.3 項「SAP アウトバウンド・メッセージの処理」](#)

A.1 アダプタの機能

この項では、次のトピック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 の実装環境では、SAP Java Connector (JCo) を使用します。これは、ネイティブの c 通信レイヤーとの Java Native Interface (JNI) を介して通信を行う Java インタフェースです。RFC (リモート関数呼出し) API は、SAP システム上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) と TCP/IP プロトコルを実装します。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 に送信できるように現在の SAP メッセージ・タイプにマーシャリングされます。SAP サーバーはこのメッセージを現在のランタイム・オブジェクトにアンマーシャルして、そのオブジェクトを実行します。レスポンス・メッセージは RFC にマーシャリングされてアダプタに返され、アンマーシャルと XML へのシリアライズが行われます。この処理はシームレスで非常に高速です。資格証明が正しくなかったり、ドキュメントのエンコードや実際のアプリケーションにエラーがあると (所定のフィールドに誤った値を入力するなど)、エラーが発生する可能性があります。

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、そのノードは、リクエストに応じるためのサーバーとして機能します。リクエストがノードからのアウトバウンドである場合、そのノードは、サーバーがリクエストに応じるのを待つクライアントとして機能します。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

他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・ロールはリクエスト/レスポンス・メッセージを同期型で処理します。クライアント・システムは、メッセージ処理を実行するのではなく有効化します。メッセージはクライアント上で始まり、クライアント上で終わります。アダプタは、Oracle から SAP にメッセージを送信するときにはクライアントとして機能します。

クライアント接続には、以下のタイプがあります。

- アプリケーション・サーバー。これは単一の SAP アプリケーション・サーバーに接続されます。
- メッセージ・サーバー。これは、分散負荷処理のための、SAP アプリケーション・サーバー・グループに対する SAP ログオン・ロード・バランスのメッセージ・サーバーに接続されます。

どちらの接続タイプも接続プールと呼ばれる特殊な接続を使用します。これについては、この付録の後半で説明します。

クライアント・メッセージには、以下のタイプがあります。

- BAPI メッセージ。これは、XML 属性としてパラメータを備えたオブジェクト形式のメッセージです。
- RFC メッセージ。これは、XML 要素としてパラメータを備えた RFC 形式のメッセージです。
- Intermediate Documents (IDoc)。これは、位置区切りのフラット・テキストまたは XML 要素のパラメータを送信する ALE メッセージ・システムです。

A.1.3 サーバー

他のシステムから接続リクエストを受け取るシステムはサーバーとなります。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。サーバーが NOT_IMPLEMENTED の例外を適用して、所定のメッセージに対する処理を行わない場合は、メッセージ・リクエストが拒否されます。サーバーが処理できるメッセージのタイプは、以下の 2 つです。

- インバウンド・メッセージ。これには、メッセージ・リクエストとそのリクエストのパラメータが含まれます。
- 同期メッセージ。これは、インバウンド・メッセージを選択し、SAP に渡す前に追加作業を実施するか、別のシステムに渡すか、あるいは追加処理のための処理を行います。

SAP アプリケーション・サーバーは、同期メッセージ処理が終了するのを待ってから処理を続行します。

サーバー接続には、以下のタイプがあります。

- 処理モード (リクエスト)。これは、メッセージ・タイプとパラメータを単に受け取るだけです。
- レスポンス。これは、同期処理を実行します。

サーバー・メッセージには、以下のタイプがあります。

- RFC メッセージ。これは、要素としてパラメータを備えた RPC 形式のメッセージです。
- Intermediate Documents (IDoc)。これは、位置区切りのフラット・テキストまたは XML 要素のパラメータを送信する ALE メッセージ・システムです。

A.2 SAP インバウンド・メッセージの処理

この項では、次のトピック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」](#)

リクエスト/レスポンス (クライアント) メッセージを Remote Function Call (RFC) または Business Application Interface (BAPI) オブジェクト対応の SAP システムに送信するには、有効なログオン資格証明セットが必要です。必須パラメータのリストなど、アダプタ・ターゲットの構成方法の詳細は、2-3 ページ「[SAP R/3 ログオン・パラメータの特定](#)」を参照してください。アダプタ・ターゲットの有効値については、SAP システム管理者に確認してください。

初回のログオンを行う前に、以下の値を取得します。

- 有効な SAP 接続パラメータ
- SAP 通信ユーザー (SAP GUI DIALOG ユーザーはセキュリティ上のお勧めできません)
- アダプタが実施すると想定されるタスクのための正しい SAP 認可

SAP 認可

サーバーに接続する前に、適切な SAP 認可を取得しておく必要があります。これは、同じユーザー ID への RFC および DIAG アクセスを許可しないための、SAP のセキュリティ推奨事項です。

RFC アクセスの保護に使用される基本的な認可オブジェクトは以下のとおりです。

S_RFC: 機能グループを保護します (認可オブジェクト・クラス AAAB 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:** 有効値は "FUGR" のみです。
 - **認可フィールド:** RFC_NAME
 - **認可値:** アクティブ化された機能グループの名前。例: ("ABCD") または "*"
 - **認可フィールド:** ACTVT
 - **認可値:** 有効値は 16 (実行) のみです。

個々の関数モジュールの実行可能コードには、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。機能やアプリケーションのユーザーに、その機能や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、機能開発者またはアプリケーション制作者の要件です。管理者に問い合わせ、追加の認可が必要かどうかを判断してください。

以下の認可オブジェクトを使用すると、個々の表をアクセス保護することができます。

S_TABU_DIS: 表を保護します (認可オブジェクト・クラス BC_A 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 ("Authorization Group")。
 - 認可値: Table など: MARA はグループ MA に含まれます。
 - 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詳細な説明については、SAP のシステム・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正しい資格証明を指定し、正しい認可が SAP アプリケーション・システムに組み込まれたら、アダプタ初期化の準備が完了です。

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、該当するアプリケーション・ツリー分岐 (*Financial Accounting* など) を展開し、ビジネス・オブジェクト (*Company Code* など) に移動して展開します。オブジェクトをクリックし、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。ランタイムで使用される BAPI はすべて "リリース" ステータスでなければなりません。そうでないと、ランタイム例外が発生します。メソッドを選択し、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、そこにデータを埋め込みます。適切な入力および出力 (*file/html/etc*) 伝送メソッドを使用して、Oracle プロセスを作成します。結果を受け取る処理のために、ドキュメントをアダプタに送信します。

Remote Function Call

アプリケーション・エクスプローラを使用して、該当するアプリケーション・ツリー分岐 (*Financial Accounting* など) を展開し、機能グループ (*0002 - Company Code* など) に移動して展開します。各機能は、異なる処理メソッドを公開します。BAPI は RFC によって実装されるため、すべて RFC オブジェクトとして表されます。これらはこの方法で呼び出され、BAPI 処理で実行されない場合などにイベント処理を可能にします。すべての RFC が BAPI として表されるわけではありません。機能を選択し、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、そこにデータを埋め込みます。適切な入力および出力 (*file/html/etc*) 伝送メソッドを使用して、Oracle プロセスを作成します。結果を受け取る処理のために、ドキュメントをアダプタに送信します。

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 が送信され、SAP R/3 システムに格納されます。この処理を行うには、使用する SAP サーバー上で構成が必要となります。このロールでは、メッセージがトランザクション・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、SAP 内の 2 番目のステップ (トリガーされたワークフロー) 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、次のトピックについて説明します。

- [A.2.1.1 項「アダプタ・チェックリスト」](#)

IDoc を SAP ERP に送信するには、メッセージ・パーティに関する以下の情報が必要です。

- メッセージの送信者

- メッセージの受信者
- 送信するメッセージのタイプ

送信者と受信者は、ALE システムで定義する必要があり、また各パーティごとに、各パーティが受け取るメッセージの種類とその処理方法を定義します。

接続情報には物理接続情報が含まれないため、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単位で定義されます。論理システムは、特定の送信側または受信側に関するすべての情報を格納するコンテナの役目をします。

論理システムが定義されると、それが送受信するさまざま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。どの処理を行うにも、このモデルを事前に定義しておく必要があります。送信側または受信側のエラーを避けるため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・アプリケーションがフィルタを有効化する場合もあります。この場合は、IDoc の特定の部分のみを送信つまり配信することができます。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、依存関係を使用して、複雑なフィルタ・ルールを構成することができます。フィルタを使用した配信モデルの構成に関する詳細説明については、SAP システムのマニュアル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・プロファイルで定義されます。パートナとメッセージごとにプロセス・コードを定義して、(可能であれば)メッセージを異なる方法で処理することができます。

A.2.1.1 アダプタ・チェックリスト

メッセージを SAP に送信するには、ALE 管理者が (ログオン資格証明の後に) 以下のものを割り当てる必要があります。

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・タイプと IDoc 基本タイプ
- SAP ERP システムでは、ALE 管理者が以下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・プロファイルを作成します。
 - a. 各メッセージ・タイプごとに、メッセージを処理する SAP のプロセス・コードを割り当てます。

例

- (SAP トランザクション SALE によって) 論理システム "ORACLIENT" が作成されます。
- (SAP トランザクション BD64 によって) 配信モデル "ZORACLNT" が作成されます。
- メッセージ・タイプ "MATMAS" が、送信側パートナ "ORACLIENT" と受信側パートナ "T90ALECNT" (中央ホストの ALE コンポーネント) 用に追加されます。
- (SAP トランザクション WE20 によって) パートナ・プロファイル "ORACLIENT" が作成され、インバウンド・メッセージ "MATMAS" がプロセス・コード "MATM" とともに割り当てられます。

注意： 以下に参照情報として、インバウンド・メッセージ用に SAP ERP を構成する手順を示します。包括的な詳細情報については、SAP ERP のマニュアル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、以下の手順を実行する必要があります。

1. 論理システムの構成
2. 配信モデルの構成
3. インバウンドのパートナー・プロファイルの定義

A.2.2 論理システムの構成

この項では、次のトピックについて説明します。

- [A.2.2.1 項「SAP トランザクション・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、メッセージの送信側と受信側に関する構成情報を格納するためのコンテナ・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・ポイントが処理できるメッセージと、その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るためです。

A.2.2.1 SAP トランザクション・ショートカット

- `/n` - 現在のトランザクションを終了し、新規トランザクションを直ちに開始します。
- `/o` - 現在のトランザクション・コンテキストを保持し、新規トランザクションを新規ウィンドウで開始します。

A.2.2.2 論理システムの定義

論理システムの定義は、「カスタマイズのための SAP 実装ガイド (IMG)」にエントリを構築することで作成されます。

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 まで移動します。

1. SALE トランザクションに移動します。
IMG の表示ウィンドウに、ALE 構成ツリーが表示されます。
2. 次のステップを実行します。
 - a. 基本設定ノードを展開します。
 - b. 論理システムを展開します。
 - c. 論理システムの定義をクリックします。
 - d. SAP R/3 ユーザーに対して若干異なる画面が表示される場合があるため、追加ノード受信側 / 送信側システムを展開します。

"注意: この表はクロスクライアントです" 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・データに対する変更が SAP ERP サーバーの全ユーザーに適用されることを知らせるために表示されます。これは、ログオン・クライアントのみに限定される大半のアプリケーション・データとは対照的です。論理システムのエントリには変更を加えないでください。新規情報の追加のみにしてください。これを行うには、適切な権限を持つ 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。
変更ビューの論理システム・ウィンドウが表示されます。
4. **新規エントリ**・ボタンをクリックします。
新規エントリ: 追加されたエントリの概要ウィンドウが表示されます。
5. 論理システム列に論理システム (ORACLETDS など) を入力し、名前列に説明を記載します。

注意: フィールドは黄色で強調表示されるので、必須フィールドであることがわかります。

6. 上部のメニュー・バーにある「**保存**」アイコンをクリックします。
ワークベンチ・リクエストのプロンプト・ダイアログが表示されます。

注意: システムの使用状況に応じて、前回のワークベンチ・リクエスト番号がダイアログに表示されます。これは使用せずに、次のステップに進んでください。

7. **新規リクエスト**・アイコンをクリックし、新規リクエストを作成します。
8. リクエスト・システムに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、変更の必要はありません。

ワークベンチ・リクエストのプロンプト・ダイアログが再表示され、ワークベンチで前回作成されたリクエスト番号とともに、新規リクエスト番号が入力されます。これは使用しないでください。
9. チェック・マークをクリックし、新規リクエスト番号を作成します。

論理システム・エントリ・ウィンドウがリフレッシュされ、新規論理システムの情報が表示されます。新規論理システムが青色で強調表示され、このステップが正常に完了したことを示します。

A.2.3 論理システム用の配信モデルの構成

この項では、次のトピック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、送信側論理システム、受信側論理システム、送受信されたメッセージのタイプが含まれます。所定の送信側または受信側に有効な配信モデルがない場合は、メッセージが処理されず、エラー・ロケーションにルーティングされます。SAP BAPI オブジェクトおよびフィルタリングを含む、配信モデルの詳細なトピックについては、SAP のマニュアル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、以下の手順を実行します。

1. **bd64** トランザクションを実行します。

配信モデルの表示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. **配信モデル**をクリックし、**処理モードの切替え**を選択します。
画面がリフレッシュされ、モデル・ビュー・ウィンドウが配信モデルの変更に切り替わります。
4. 使用可能なメニュー・ボタンから、**モデル・ビューの作成**を選択します。
モデル・ビューの作成ダイアログが表示されます。
5. 「短縮テキスト」フィールドにモデル・ビューの名前を入力し、技術名フィールドに名前を入力します。これは説明としての役割も果たします。
先頭文字が Z の名前は、SAP のシステム更新時に上書きされないため、この命名規則を使用することをお勧めします。これは、モデル・ビューの名前にすぎません。
6. 緑色のチェック・マークの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、メインの配信モデルの変更ウィンドウに戻ります。構成した配信モデルがリストに追加されます。
モデル・ビューは、「保存」アイコンをクリックするまでは一時トランザクション・ストレージに保管されます(エントリが明るい色で表示されます)。システム接続が失われるかタイムアウトになると、エントリが失われます。ベスト・プラクティスとして、変更内容が失われるのを防ぐため、配信モデルの構成プロセス中は各ダイアログの後に変更内容を保存してください。
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、**メッセージ・タイプの追加**ボタンをクリックします。
メッセージ・タイプの追加ダイアログが表示されます。
9. 次のステップを実行します。
 - a. 「送信者」フィールドに、Oracle システムの送信側論理システムの名前 (ORACLETDS など) を入力します。詳細については、A-6 ページ「[論理システムの構成](#)」を参照してください。
 - b. 「受信者」フィールドに、SAP システム用のクライアント ALE 論理システムの名前を入力するか、ドロップダウンから選択します (T90CLNT090 など)。
 - c. 「メッセージ・タイプ」フィールドに、使用するメッセージのタイプを入力します (MATMAS など)。
各フィールドの右にあるドロップダウン・アイコンをクリックすると、利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・マークのアイコンをクリックして情報を入力します。
メインの配信モデルの変更ウィンドウがリフレッシュされます。
10. 「**保存**」をクリックします。
11. ALE 階層ツリーを、最近作成した配信モデルまで移動します。送信側と受信側が、送信側、受信側、およびメッセージとのグラフィカル関係で表されます。
同じシステムに対する後続のメッセージ・タイプについて、モデル・ビューを送信側の名前まで展開し、その名前にカーソルを置きます。次に、**メッセージ・タイプの追加**ボタンをクリックします。すべてのフィールドに値が自動的に入力されます。この後、メッセージ・タイプを追加することができます。

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、処理対象のメッセージと、メッセージの処理方法を定義したものです。これにより、パートナの処理詳細を柔軟に構成することができます。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つだけ設定できます。パートナ・プロファイルを定義するには、以下の手順を実行します。

1. 画面の左上にあるコマンド・バーに「/nwe20」と入力して、**we20** を実行します。
パートナ・プロファイル・ウィンドウが表示されます。
2. **パートナ・タイプ LS** を選択し、中央のメニュー・バーで白いアイコンをクリックします。また、上部のメニューから「**作成**」を選択することもできます。
画面の右側が空白のパートナ・プロファイル・フォームに変わります。
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、「**作成**」アイコンをクリックします。
新しいプロファイルで画面が変更されます。
4. 埋込みテキスト "Inbound parmtrs" エントリが表示されるまでスクロール・ダウンします。
5. 緑色の「プラス記号」アイコンをクリックします。
パートナ・プロファイル:インバウンド・パラメータ・ダイアログが表示されます。
6. メッセージ・タイプとして「**MATMAS**」と入力します。
7. プロセス・コード・ドロップダウンを使用して、MATMAS に対して「**MATM**」を選択します。
8. プロセス・モードとして「**即時**」または「**背景**」を選択します。

注意: 正しい設定値については、SAP 管理者に問い合わせてください。

9. メニュー・バーで「**保存**」アイコンをクリックし、MATMAS のインバウンド・パートナ・プロファイル・エントリを保存します。
10. メニュー・バーで「**戻る**」(左向き矢印) ボタンをクリックし、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さらに追加するには、A-7 ページ「**論理システム用の配信モデルの構成**」以降のステップを繰り返します。別の論理システムとメッセージ・タイプを追加するには、A-6 ページ「**論理システムの構成**」以降のステップを繰り返します。

これで、適切な Oracle プロセスを構成した後、IDoc を送信できるようになります。

SAP IDoc モニター・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のほか、送信側、受信側、データを確認できます。詳細については、SAP のオンライン・ヘルプ・ページで IDoc 表示に関するトピックを参照してください。

A.3 SAP アウトバウンド・メッセージの処理

Oracle Application Adapter for SAP R/3 は、インタフェース・スタイルのリモート関数モジュール、BAPI ALE、またはトランザクション ID を持つトランザクション RFC メッセージを使用して、SAP メッセージからのアウトバウンドを処理します。メッセージ・タイプには Remote Function Call または IDoc を使用できます。SAP JCo を

通した処理はすべて同期型 (リクエスト/レスポンス) ですが、メッセージ処理は非同期型 (配信のみ) でも同期型 (イベント/リプライ) でも可能です。同期型では、オーケストレーションされたプロセスがタスクを実行し、待機中の SAP 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・システムから送信することができます。IDoc は、システム・トランザクションまたは標準プログラムを通して送信されます。SAP の BAPI イベントは、特別にプログラミングされ、ALE インタフェース用に有効化されていない限り、内部で使用されます。

SAP のイベントは、イベントの処理タスクを実行するイベント・ハンドラとのイベント・リンケージによって有効化されます。イベントの送信メカニズムはすべて SAP システム上で有効化され、アダプタが SAP イベント・ハンドラの終了処理の受信側になります。

アウトバウンド・メッセージは、アダプタ内ではイベントとして定義されます。イベントは、非同期型 (メッセージの配信のみ) または同期型 (リプライを返す) になります。アダプタのインバウンド・イベントを双方向 (イベント・リプライ) にするには、受信側プロセスでプロセス・フロー内にリクエスト・ハンドラを実装する必要があります。リクエスト・ハンドラは入ってきた SAP データを受け取り、データを操作し、そのデータを有効なイベント・リプライ形式にして、SAP に返すための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、Remote Function Call を受け取るには、SAP RFC 宛先とプログラム ID のみが必要です。IDoc 処理の場合、RFC を通してメッセージ・イベントを受信するには、論理システム、配信モデル、パートナー・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 のアウトバウンドに関する用語

- イベント・クリエーター - JCO/RFC クライアント。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・リプライ (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。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。

この項では、次のトピック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 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 の構成」](#)
- [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「パートナー・プロファイルの定義」](#)
- [A.3.9 項「SAP R/3 ALE 構成のテスト」](#)

アウトバウンド処理は、よくイベントとして実装されます。SAP アプリケーション・サーバー上には、定義済イベントのステージが多数あります。イベント処理は、リンケージによってまとめられます。デフォルトでは、実動 SAP サーバー上で有効化されないリンケージもあります。イベントをアクティブ化し、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・ハンドラは、メッセージの処理を実装します。

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コンポーネント上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、さまざまなイ

イベント・タイプがあります。最も良く知られているイベント・ハンドラは、品目マスタや顧客マスタなどの SAP 関連オブジェクトによって実装される ALE メッセージ管理です。これらのオブジェクト内の変更によって、アウトバウンド・イベント IDoc が生成される場合があります。

イベント作成は、通常は SAP によって実装されますが、多くのオブジェクトはイベント処理用に有効化することができます。ただし、これには詳細な構成やプログラミングが必要な場合があります。SAP イベントのタイプおよびそれらの構成方法については、SAP のマニュアル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、Remote Function Call を受信するために SAP RFC の宛先とプログラム ID が必要です。

IDoc 処理の場合、tRFC を経由してメッセージ制御イベントを受信するための、論理システム、配信モデル、パートナー・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 のアウトバウンドに関する用語

- JCO/RFC クライアント。RFC リクエストを作成するシステムであり、オプションとしてレスポンスを待ちます。
- JCO/RFC サーバー。RFC リクエストを受信し、リクエストを処理するシステムであり、オプションとしてレスポンスを受信します。

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ゲートウェイ・サーバー上で実行されるプロセスです。ゲートウェイ・プロセスは、SAP への接続をすべて管理します。SAP へのサーバー接続が受け入れられるには、事前にサーバー接続を SAP アプリケーションで登録プログラムとして構成し、必要なプログラム識別子を提示する必要があります。

サーバー接続はそれ自体をゲートウェイに提示し、プログラム識別子を公開します。プログラム識別子が登録済プログラム ID のリスト内で見つかった場合は、ゲートウェイ・サーバーが、接続を "受け入れる" サーバーへの接続を提供します。このプログラム ID は SAP 内で RFC 宛先とリンクされ、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します。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。プログラム ID は、SAP への接続を行うアプリケーション・サーバーでも公開され、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。セキュリティ推奨事項として、プログラム ID を実装するアダプタ・インスタンスへのアクセス権を持つ 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けることをお勧めします。リクエストはすべて RFC 宛先を通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、RFC コール・リクエストの着信を待機します。RFC サーバー・プログラムは、特定の SAP システムに対してではなく、SAP ゲートウェイ上のプログラム ID で自身を登録します。SAPGUI で、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

この項では、次のトピックについて説明します。

- A.3.2.1 項「SAP RFC アウトバウンドのアウトライン」
- 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・インスタンスが自身をゲートウェイに提示している場合、ゲートウェイは接続を提供し、ロード・バランシング・モードで機能を開始します。ゲートウェイは専用のアルゴリズムを使用し、要求と合計処理時間に応じて各サーバーにさまざまなメッセージを送信します。これにより、スキーマやアプリケーションによってメッセージが検証されるときに、予測不能な結果が生じる可能性があります。

SAP は、1 つの SAP プログラム ID を使用して Oracle で複数のイベントを構成する場合に、イベント・データのロード・バランシングを行います。たとえば、複数の Remote Function Call または BAPI が同じプログラム ID (ORACLETDS など) を使用し、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 は 1 つのリクエストを 1 つのリスナーに送信し、次のリクエストを別のリスナーに送信します。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。これは SAP アプリケーション開発の専有メカニズムであり、接続の合計スループット、待ち状態の回数などを比較することで動作する場合があります。たとえば、1 つの接続で 9 つのメッセージを受け取り、2 番目の接続で 1 つのメッセージを受け取るとします。9 つのメッセージのうち 5 つがスキーマ検証で拒否され、もう一方の接続で受け取る 1 つのメッセージがスキーマ検証で拒否された場合、ユーザーには SAP イベント処理メッセージが失われたように見える可能性があります。

サーバー内のロード・バランシング (SAP からアダプタへのインバウンド) の状況は、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で処理されます。そのため、SAP システムによって接続のロード・バランシングが行われます。このパフォーマンスは調整できません。

A.3.2.1 SAP RFC アウトバウンドのアウトライン

1. RFC 宛先を作成します (SM59 経由)。
 - a. 宛先名を入力します。
 - b. 登録済みプログラム 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/Mediator を起動します。
 - e. BPEL または Mediator チャンネルのステップ b から同じ登録済みプログラム ID 値を入力します。
 - f. チャンネルを起動します。
 - g. SAP でテスト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 経由)。

ABAP ワークベンチ認可が必要です。

 - a. 「宛先」フィールドにステップ 1a からの RFC 宛先名を入力します。
 - b. 「関数」フィールドに有効な SAP RFC 関数名を入力します。
 - c. 関数のパラメータにデータを入力します。
 - d. 「実行」(時計) ボタンをクリックします。

データが BPEL チャンネルに到着します。

A.3.2.2 SAP IDoc イベント

IDoc を SAP から受け取る前に、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 用のチャンネルを構成しておく必要があります。さらに、SAP アプリケーション・サーバー上の構成も必要です。

1. RFC 宛先を作成するか (SM59 経由)、あるいは前のセクションで定義された宛先と同じ宛先を使用します (SAP RFC アウトバウンドのアウトライン)。
 - a. 宛先名を入力します。
 - b. 登録済みプログラム 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/Mediator を起動します。
 - e. BPEL または Mediator チャンネルのステップ b から同じ登録済みプログラム ID 値を入力します。
 - f. チャンネルを起動します。
 - g. SAP でテストボタンを使用して接続をテストします。
2. A-6 ページ「[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「[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 を経由して 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 を経由して SAP アウトバウンド・パートナー・プロファイルを作成します。

A.3.3 SAP GUI で RFC 宛先とプログラム ID の登録

この項では、次のトピック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 システムで、SAP R/3 イベント・アダプタに次のコールまたはインタフェースを発行できるようにするには、RFC 宛先の下でプログラム ID を登録する必要があります。

- Remote Function Call (RFC)
- Business Application Programming Interface (BAPI) - BAPI には外部イベントはありません。RFC フォームを使用してください。
- Intermediate Document (IDoc)

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 とイベント・ハンドラで構成されます。

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、以下の手順を実行します。

1. SAP GUI を起動して SAP R/3 システムにログインします。
2. 「ツール」、「管理」、「ネットワーク」、RFC 宛先の順に選択します。

3. **SM59** トランザクションを実行します。
RFC 接続の構成ウィンドウが表示されます。
4. **TCP/IP 接続**を選択し、「作成」(白紙のアイコン)をクリックします。
RFC 宛先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 宛先フィールドに、名前(ORACLETDS など)を入力します。
このフィールド値は大/小文字の区別があります。送信側プログラムと同じ大/小文字を使用してください。
 - b. 「接続タイプ」フィールドに、宛先タイプ TCP/IP を表す T を入力します。
 - c. 「説明」フィールドに、簡単な説明を入力します。
6. ツール・バーから「保存」アイコン(ディスク)をクリックするか、トップ・メニュー・バーから「接続」、「保存」を順に選択します。
ウィンドウがリフレッシュされ、第2のパネルに技術設定タブが強調表示されます。
7. **登録済サーバー・プログラム・ボタン**をすぐにクリックします。
画面が再びリフレッシュされます。
8. 大/小文字を区別したプログラム ID を入力します。これは、アダプタ・チャンネルでの名前と同じ名前に一致させる必要があります。
9. ツール・バーから「保存」アイコン(ディスク)をクリックするか、トップ・メニュー・バーから「接続」、「保存」を順に選択します。
10. **MDMP と Unicode** タブを選択し、「Unicode」ボタンをクリックします。
11. ツール・バーから「保存」アイコン(ディスク)をクリックするか、トップ・メニュー・バーから「接続」、「保存」を順に選択します。
RFC 宛先とプログラム ID が定義されます。関数コール API の宛先パラメータを使用して、任意の Remote Function Call (RFC) をサーバーに送信できるようになります。これは逆方向のコールなので、関数のデータをマーシャリング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、以下の手順を実行します。

1. Oracle Application Adapter for SAP R/3用のBPELプロセス/チャンネルでプログラム ID を入力し、チャンネルを開始します。
2. SAP GUI に移行して SE37 に移動します。
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、「テスト実行」(ツール)アイコンをクリックします。
関数モジュールのテスト:初期画面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、「実行」(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、そのメタデータをキャッシュに入れる必要があるため、必ず初回は最も実行時間がかかり、それ以降は実行時間が短くなります。

A.3.3.3 同期 RFC コール

同期 RFC コールは、SAP 側で非同期と同じ方法で定義されます。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 から配信されたデータを受け取って処理し、それを関数形式で SAP に返すプロセスをオーケストレーションします。形式エラーが発生すると、リクエストは直ちに終了します。以下に擬似コードの例を示します。

```
関数 : MY_COMPANY_GETLIST DESTINATION 'ORACLETDS'

tables mycompanies
      field1: name(20)
      field2: company(20)
```

同期型の Get List コールは、空の表 *mycompanies* を BPEL サーバーに渡し、SAP は待機します。BPEL のオーケストレーションには、以下の 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
```

この SQL 文は Oracle データベース上で実行されます。結果セットは、オーケストレーション内の代入文によって返され、*mycompanies* 表に渡されます。なお、このプロセスの間中、SAP アプリケーションは待機しています。SAP サーバーにはオンライン・プロセスの最大時間が設定されている場合があり、その場合は最大時間に達すると SAP によって自動的にプロセスが終了されます。

関数は、同期コールを介して SAP に戻され、テーブルはコーリング・プロセスにデータを返します。

この実行時間が接続プールの実行時間よりも長いと、タイムアウトが発生する可能性があります。タイプ、長さ、小数桁数に関する形式例外が発生する可能性があります。

A.3.4 イベント・アダプタ用の Application Link Embedding の構成

SAP R/3 イベント・アダプタは、SAP R/3 から IDoc (中間ドキュメント) を受け取ります。SAP R/3 システムが IDoc を SAP R/3 イベント・アダプタに送信するよう構成するには、ALE (Application Link Embedding) 構成を使用して以下のことを行います。

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. パートナとメッセージ・タイプについて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

ポートは、メッセージの送信先を識別します。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、それを RFC 宛先にリンクすることで、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、次の手順を実行します。

1. ALE 構成で、「ツール」、ビジネス通信、IDoc ベース、IDoc、ポート定義の順に選択します。

WE21 トランザクションを実行することもできます。

IDoc 処理のポート・ウィンドウが表示されます。

2. 「ポート」の下の左ペインで、**トランザクション RFC** を選択し、「作成」(白紙のアイコン)をクリックします。
3. **ポート名の生成**をクリックします。
システムはポート名を生成します。
4. このポート (通常 4) を通じて送信する IDoc のバージョンを入力します。
5. 作成した RFC 宛先の名前 (ORACLETDS など) を入力または選択するか、選択リストに最初の数文字とアスタリスクを入力します。
6. セッションを保存して、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: 以前に定義した論理システムを再使用する場合は、この項を省略することができます。

論理システムは、メッセージの送信側と受信側に関する構成情報を格納するためのコンテナ・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・ポイントが処理できるメッセージと、その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るためです。

この項では、次のトピックについて説明します。

- [A.3.6.1 項「SAP トランザクション・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・ショートカット

- `/n` - 現在のトランザクションを終了し、新規トランザクションを直ちに開始します。
- `/o` - 現在のトランザクション・コンテキストを保持し、新規トランザクションを新規ウィンドウで開始します。

A.3.6.2 論理システムの定義

論理システムの定義は、「カスタマイズのための SAP 実装ガイド (IMG)」にエントリーを構築することで作成されます。

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 まで移動します。

1. SALE トランザクションに移動します。
IMG の表示ウィンドウに、ALE 構成ツリーが表示されます。
2. 次のステップを実行します。
 - a. **基本設定**ノードを展開します。
 - b. **論理システム**を展開します。
 - c. **論理システムの定義**をクリックします。
 - d. SAP R/3 ユーザーに対して若干異なる画面が表示される場合があるため、追加ノード**受信側 / 送信側システム**を展開します。

"注意: この表はクロスクライアントです" 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・データに対する変更が SAP ERP サーバーの全ユーザーに適用されることを知らせるために表示されます。こ

れは、ログオン・クライアントのみに限定される大半のアプリケーション・データとは対照的です。論理システムのエントリーには変更を加えないでください。新規情報の追加のみにしてください。これを行うには、適切な権限を持つ 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。
変更ビューの論理システム・ウィンドウが、強調表示なしで表示されます。
4. **新規エントリー**・ボタンをクリックします。
新規エントリー: 追加されたエントリーの概要ウィンドウが表示されます。
5. 論理システム列に論理システム (ORACLETDS など) を入力し、名前列に説明を記載します。

注意: フィールドは黄色で強調表示されるので、必須フィールドであることがわかります。

6. 上部のメニュー・バーにある「**保存**」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。
ワークベンチ・リクエストのプロンプト・ダイアログが表示されます。

注意: システムの使用状況に応じて、前回のワークベンチ・リクエスト番号がダイアログに表示されます。これは使用せずに、次のステップに進んでください。

7. **新規リクエスト**・アイコンをクリックし、新規リクエストを作成します。
8. リクエスト・システムにエントリー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、変更の必要はありません。
ワークベンチ・リクエストのプロンプト・ダイアログが再表示され、ワークベンチで前回作成されたリクエスト番号とともに、新規リクエスト番号が入力されます。これは使用しないでください。
9. チェック・マークをクリックし、新規リクエスト番号を作成します。
新規論理システムが青色で強調表示され、このステップが正常に完了したことを示します。

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、送信側論理システム、受信側論理システム、送受信されたメッセージのタイプが含まれます。所定の送信側または受信側に有効な配信モデルがない場合は、メッセージが処理されず、エラー・ロケーションにルーティングされます。SAP BAPI オブジェクトおよびフィルタリングを含む、配信モデルの詳細なトピックについては、SAP のマニュアルを参照してください。

この項では、次のトピック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、以下の手順を実行します。

1. 画面の左上にあるコマンド・バーに「/nbd64」と入力して、**bd64** を実行します。

配信モデルの表示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. 上部のメニューから、**配信モデル**をクリックし、リストから**処理モードの切替え**を選択します。
画面がリフレッシュされ、メインの配信モデルの変更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、**モデル・ビューの作成**を選択します。
モデル・ビューの作成ダイアログが表示されます。
5. 「短縮テキスト」フィールドにモデル・ビューの名前を入力し、技術名フィールドに名前を入力します。これは説明としての役割も果たします。
先頭文字が Z の名前は、SAP のシステム更新時に上書きされないため、この命名規則を使用することをお勧めします。これは、モデル・ビューの名前にすぎません。
6. 緑色のチェック・マークの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、メインの配信モデルの変更ウィンドウに戻ります。構成した配信モデルがリストに追加されます。
モデル・ビューは、「保存」アイコンをクリックするまでは一時トランザクション・ストレージに保管されます (エントリが明るい色で表示されます)。システム接続が失われるかタイムアウトになると、エントリが失われます。ベスト・プラクティスとして、変更内容が失われるのを防ぐため、配信モデルの構成プロセス中は各ダイアログの後に変更内容を保存してください。
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、**メッセージ・タイプの追加**ボタンをクリックします。
メッセージ・タイプの追加ダイアログが表示されます。
9. ALE 中央インスタンス発信者を入力します。
この詳細については、SAP 管理者に問い合わせてください。
10. 「受信者」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： 以前に定義したパートナ・プロファイルを再使用する場合は、この項を省略することができます。

パートナ・プロファイルは、論理システムごとに、処理対象のメッセージと、メッセージの処理方法を定義したものです。これにより、パートナの処理詳細を柔軟に構成することができます。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 つだけ設定できます。パートナ・プロファイルを定義するには、以下の手順を実行します。

1. 画面の左上にあるコマンド・バーに「/nwe20」と入力して、**we20** を実行します。
パートナ・プロファイル・ウィンドウが表示されます。
2. **パートナ・タイプ LS** を選択し、中央のメニュー・バーで白いアイコンをクリックします。また、上部のメニューから「**作成**」を選択することもできます。
画面の右側が空白のパートナ・プロファイル・フォームに変わります。
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、「**作成**」アイコンをクリックします。
新しいプロファイルで画面が変更されます。
4. 「メッセージ・タイプフィールドに、メッセージ・タイプを入力します。
5. 受信者ポート・フィールドにポート番号を入力します。
6. **IDoc をすぐに転送** ボタンを選択するか、**IDoc の収集** を選択し、バッチ・サイズを指定し、bd87 を使用して処理対象の IDoc をリリースします。
7. 基本タイプ (MATMAS の場合は MATMAS05 など) を入力します。
8. 「**保存**」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り、マスター・データ IDoc を任意の論理システム (たとえば、イベント・アダプタ) に送信できるようになります。

この項では、次のトピック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 バージョンの処理」](#)

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、以下の手順を実行します。

1. 顧客送信ウィンドウで、「出力タイプ」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. 論理システム・フィールドに、論理システム (ORACLETDS など) を入力します。
3. 「**実行**」をクリックします。

SAP R/3 イベント・アダプタは、IDoc を XML 形式で受け取ります。イベント・アダプタからのレスポンスは想定されません。

確認メッセージが表示されます。

同じ数の通信 IDoc も生成されます。通信 IDoc カウントとマスター IDoc カウントの数が異なる場合は、一部の IDoc が構文チェックで不合格になっている可能性があります。IDoc アプリケーション開発者と一緒に確認してください。通常、IDoc はアダプタ・サーバーにほぼ即座に表示されます。IDoc が表示されない場合は、以下のことを確認してください。

1. パートナ・プロファイルで IDoc の収集オプションが選択されている (bd87 に移動し、それらを処理対象としてリリースします)。
2. 他のサーバーが同じプログラム ID を公開していない。

A.3.9.2 SAP IDoc イベント

IDoc を SAP から受け取る前に、アプリケーション・エクスプローラを使用して Oracle Application Adapter for SAP R/3 用のチャンネルを構成しておく必要があります。さらに、SAP アプリケーション・サーバー上の構成も必要です。

1. RFC 宛先を作成するか (SM59 経由)、あるいは前のセクションで定義された宛先と同じ宛先を使用します (SAP RFC アウトバウンドのアウトライン)。
 - a. 宛先名を入力します。
 - b. 登録済みプログラム 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/Mediator を起動します。
 - e. BPEL または Mediator チャンネルのステップ b から同じ登録済みプログラム ID 値を入力します。
 - f. チャンネルを起動します。
 - g. SAP でテストボタンを使用して接続をテストします。
2. A-6 ページ「[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「[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 を経由して 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 を経由して SAP アウトバウンド・パートナー・プロファイルを作成します。

A.3.9.3 IDoc バージョンの処理

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なければなりません。

アダプタには、IDoc リリース・バージョンに関する以下の 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 - 送信側アプリケーションは、正しいバージョンを出力するために IDoc 制御レコードを修正する必要があります。これが行われない場合は、SAP ログオン・リリースにデフォルト設定されます。
- ユーザー入力

ユーザー入力フィールドでは、2 種類の入力が可能です。入力が SAP リリース (700 など) の場合は、そのリリースのメタデータが使用されます。

このフィールドの値が空白の場合は、リリース済か否かを問わず、SAP にある最も高いバージョンのメタデータが返されます。不明な IDoc や最近の IDoc を使用している場合は、これが最善のオプションとなることがよくあります。リリース済のセグメント・リスト内でセグメント名が見つからない場合、SAP は (アダプタを通して) "不明なセグメント" というメッセージを返します。この場合に IDoc を処理するには、そのセグメントをリリースするか、アダプタ設定を、値が空白の "ユーザー入力" に変更する必要があります。

用語集

アダプタ

1つの電子的インタフェースを別の電子的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)一般的な接続性を提供します。

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。

ビジネス・サービス

Web サービスとしても知られます。Web サービスは、独立型でモジュール化された機能で、オープン標準を使用してネットワーク間で公開およびアクセスが可能です。コンポーネント別のインタフェースの実装で、実行可能なエンティティです。

チャンネル

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接続を表します。1つのチャンネルは、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・ポートをバインドします。

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です。

ポート

アダプタにより公開される特定のビジネス・オブジェクトを特定の配置と関連付けます。この場合の配置とは、プロトコルおよびイベント・データの場所を定義する URL です。ポートはイベント使用のエンドポイントを定義します。

索引

B

BAPI (Business Application Programming Interfaces), 12-4
BAPI リクエスト, 1-5
BAPI 例外パラメータ, 4-11, 4-25, 6-40
BPEL Process Manager
 OracleAS Adapter for MySAP ERP および, 6-1, 8-1
 アダプタとの統合, 6-1, 8-1
BPEL デザイナ, 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

DEBMS05
 検索, 6-42
DEBMS05 の検索, 6-42

E

Eclipse. JDeveloper を参照
「Edi バージョン」パラメータ, 4-10
EJB (Enterprise Java Bean), 5-1
Enterprise Java Bean (EJB), 5-1

I

「iBSE URL」フィールド, 4-6
IDoc (Intermediate Documents), 12-4
IDoc リクエスト, 1-5
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, 5-2
J2CA サービス, 6-37
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

MySAP Easy Access ウィンドウ, 6-54
MySAP ERP
 接続, 1-4, 6-8, 6-42 ~ 6-43
 接続先, 4-8 ~ 4-15, 12-2, 12-7
 切断元, 4-15
 統合, 1-4
MySAP ERP から切断, 4-15
MySAP ERP でのイベントのトリガー, 6-54
MySAP ERP トレース・パラメータ, 4-25, 6-40, 4-10
MySAP ERP ビジネス・プロセス, 1-4
MySAP ERP ビジネス関数, 4-8, 4-18 ~ 4-21
 スキーマの作成, 4-18
MySAP ERP プラットフォーム, 1-4
MySAP ERP への接続, 4-8 ~ 4-15, 6-8, 6-42 ~ 6-43, 12-2, 12-7
MySAP Workbench, 6-54
「MySAP」ノード, 6-42 ~ 6-43
MySAP プログラム ID (識別子), 6-54

O

OC4J-ra.xml ファイル, 5-2 ~ 5-4
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 デザイナ。BPEL Designer、
JDeveloper または Oracle JDeveloper を参照
Oracle's Unified Method (OUM), xi
OracleAS Adapter
 J2CA, 5-1
 インストール・ディレクトリ, 5-4
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 および, 6-1, 8-1
 構成, 4-1 ~ 4-29
 デプロイ, 1-2, 6-2, 8-2
 トラブルシューティング, 12-1 ~ 12-8
OracleAS Adapter J2CA
 アプリケーション・エクスペローラおよび, 12-5
 接続先, 12-5
OracleAS Adapter J2CA への接続, 12-5
OracleAS Adapter アプリケーション・エクスペローラ。
 アプリケーション・エクスペローラを参照
Oracle データベース, 5-4

P

Process Manager。BPEL Process Manager を参照

R

RFC リクエスト, 1-5

S

sapjco3.dll ファイル, 12-2
sapjco3.jar ファイル, 12-2 ~ 12-3
「SNC パートナ」パラメータ, 4-13
「SNC 名」パラメータ, 4-13
「SNC モード」パラメータ, 4-13
「SNC ライブラリ・パス」パラメータ, 4-13
「SNC レベル」パラメータ, 4-13
SOAP エージェント, 12-6
SOAP フォルト, 12-6
SOAP リクエスト, 12-6 ~ 12-8
 エラーおよび, 12-6 ~ 12-8
SOA(サービス指向アーキテクチャ), 6-1, 8-1

W

Web Service Definition Language (WSDL), 4-18 ~
 4-19
Web サービス, 1-2
 作成, 4-20 ~ 4-21
 テスト, 4-21
 デプロイ, 12-6 ~ 12-8
 統合, 6-1, 8-1
 トラブルシューティング, 12-5 ~ 12-8
 リポジトリ・プロジェクト, 4-5
「Web サービスの作成」ダイアログ・ボックス, 4-20
Web サービスのテスト, 4-21
Web サービス・プロジェクト
 作成, 4-5

Web サービス名, 4-20
WSDL (Web Service Definition Language), 4-18 ~
 4-19
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,
 6-44
WSDL ファイル, 6-1, 8-1
 アプリケーション・エクスペローラおよび, 6-37

X

XML スキーマ
 作成, 4-18
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 サービスの作成 (リクエスト / レ
 スポンス), 6-9
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 ~ 4-29, 5-2
 デプロイ, 1-5, 5-1 ~ 5-2, 6-2, 8-2
 トラブルシューティング, 12-1 ~ 12-8
アダプタ構成
 更新, 5-2
アダプタ・タイプ
 リソース, 1-2, 5-1 ~ 5-2
「アダプタ」ノード, 6-42
アダプタの構成, 4-1 ~ 4-29, 5-2
アダプタの構成の更新, 5-2
アダプタのデプロイ, 6-2, 8-2, 5-1 ~ 5-2
アダプタの例外, 12-4 ~ 12-6
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 ~ 4-29, 5-2
 デプロイ, 5-1 ~ 5-2
 トラブルシューティング, 12-1 ~ 12-8
アプリケーション・エクスペローラ, 1-2, 6-7, 6-37
 J2CA 構成, 5-2
 OracleAS Adapter J2CA および, 12-5
 WSDL ファイルおよび, 6-37
 アプリケーション・システム, 4-8
 スキーマ, 4-18
 チャンネル, 4-23
 テストおよび, 12-3
 デバッグおよび, 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, 4-9
「アプリケーション・サーバー」パラメータ, 4-10
アプリケーション・システム
 アプリケーション・エクスペローラ, 4-8
 サポート, 4-8
アプリケーション・アダプタ
 デプロイ, 6-2, 8-2

い

イベント, 1-2, 4-22, A-1
 構成, 4-22
 作成, 6-42 ~ 6-45
 トリガー, 6-54
イベント・アダプタ
 構成, 4-22
イベント処理, A-1
イベント統合
 検証, 6-54
イベントの作成, 6-42 ~ 6-45
イベント・ポート
 作成, 6-42 ~ 6-45
イベント・メッセージ, 6-37
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, 6-37

え

エラー・メッセージ, 12-2 ~ 12-8
 ターゲット・システムおよび, 12-6

お

オブジェクト・タイプ, 12-4

か

管理対象 ConnectionFactory パラメータ, 5-4
管理対象コネクタ・ファクトリ, 5-2
管理対象コネクタ・ファクトリ・オブジェクト, 5-2
 複数, 5-4

き

「既存のサービス名」リスト, 4-20

く

クライアント・パラメータ, 6-40
「クライアント」パラメータ, 4-10, 4-25

け

権限
 構成, 12-5
「言語」パラメータ, 4-10, 4-25
言語パラメータ, 6-40

こ

構成
 上書き, 5-4
 接続先, 4-7
 定義, 4-5 ~ 4-7
「構成」ノード, 4-5 ~ 4-6
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, 6-40
「コード・ページ」パラメータ, 4-10, 4-25

顧客送信ウィンドウ, 6-54

顧客マスター・データ, 6-54

コネクタ

 Oracle Application Server へのデプロイ, 5-1

コネクタ・ファクトリ, 5-2

コネクタ・ファクトリ・オブジェクト, 5-2

 複数, 5-4

さ

サービス, 1-2

 作成, 4-20 ~ 4-21

 テスト, 4-21

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, 4-6 ~ 4-7

サービス名, 4-20

サーブレット, 5-1

し

「システム」タブ, 4-10, 4-25, 6-39

「システム番号」パラメータ, 4-10

「詳細」タブ, 4-11, 4-25, 6-40

「新規構成」ダイアログ・ボックス, 4-5 ~ 4-7

す

スキーマ

 アプリケーション・エクスプローラ, 4-18

 作成, 4-18

せ

「制限」ページ, 6-54

セキュリティタブ, 4-13

接続

 確立, 4-8 ~ 4-15, 6-43

 削除, 4-15

 閉じる, 4-15

「接続」ダイアログ・ボックス, 4-15, 6-43

接続パラメータ, 4-9 ~ 4-15, 4-25 ~ 4-26, 12-2

 BAPI 例外処理, 4-11

 「Edi バージョン」, 4-10

 MySAP ERP トレース, 4-10

 「SNC パートナ」, 4-13

 SNC 名, 4-13

 「SNC モード」, 4-13

 「SNC ライブラリ・パス」, 4-14, 4-26, 6-41

 「SNC レベル」, 4-13

 「アプリケーション・サーバー」, 4-10

 「クライアント」, 4-10, 4-25, 6-40

 「言語」, 4-10

 コード・ページ, 4-10

 システム, 4-10, 4-25, 6-39

 「システム番号」, 4-10

 詳細, 4-11, 4-25, 6-40

 セキュリティ, 4-13

接続プール・サイズ, 4-11
接続プール名, 4-11
待機中コミット, 4-11
「パスワード」, 4-10, 4-25, 6-40
「ユーザー」, 4-10, 4-25, 6-40
ログオン・チケット (SSO2), 4-13
ログオン・チケット (X509), 4-13
「接続プール・サイズ」パラメータ, 4-11, 4-25
接続プール・サイズ・パラメータ, 6-40
「接続プール名」パラメータ, 4-11, 4-25
接続プール名パラメータ, 6-40
「説明」フィールド, 4-9, 4-20

た

ターゲット
削除, 4-15
接続, 6-43
接続先, 4-15, 12-7
切断元, 4-15
定義, 4-8
編集, 4-16
ターゲット・システム
エラーおよび, 12-6
「ターゲット・タイプ」リスト, 4-9
「ターゲットの追加」ダイアログ・ボックス, 4-9
ターゲットの編集, 4-16
「待機中コミット」パラメータ, 4-11, 4-25
待機中コミット・パラメータ, 6-40

ち

チャンネル, 4-22
アプリケーション・エクスペローラ, 4-23
開始, 4-28, 6-42
削除, 4-29
作成, 4-23, 6-37 ~ 6-42
停止, 4-28, 6-42
テストおよびデバッグ, 6-42
編集, 4-29
「チャンネル」ノード, 4-24, 6-38
チャンネルの作成, 6-37 ~ 6-42
「チャンネルの追加」ダイアログ・ボックス, 4-23
チャンネルの編集, 4-29
チャンネル。リスナーも参照

て

データベース
Oracle, 5-4
接続, 5-4
データベース接続
開く, 5-4
デザインタイム, 6-37
構成, 6-7, 6-37

と

トラブルシューティング, 12-1 ~ 12-8
BSE, 12-5

Web サービス, 12-5 ~ 12-8
アプリケーション・エクスペローラ, 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, 4-9

に

入力 XML ドキュメント, 12-4

の

ノード
MySAP, 6-42 ~ 6-43
アダプタ, 6-42
構成, 4-5 ~ 4-6
接続済, 4-15, 6-43
切断, 4-16
チャンネル, 4-24, 6-38
ポート, 6-38
ノードのリスト, 12-2
ノード・リスト, 12-2

は

パスワード, 4-10, 4-25, 5-4, 6-40
パスワード・パラメータ, 6-40
「パスワード」パラメータ, 4-10, 4-25
パラメータ・タイプ
構成, 5-4
接続, 4-25 ~ 4-26, 12-2
チャンネル構成, 4-29

ひ

ビジネス・イベント, 1-2
ビジネス関数, 4-8, 4-18 ~ 4-21
スキーマの作成, 4-18
ビジネス・サービス
作成, 4-20 ~ 4-21
テスト, 4-21
ビジネス・プロセス, 1-4

ふ

フォルト・コード要素, 12-6
フォルト文字列要素, 12-6
プログラム ID (識別子), 6-54
プロパティ, 5-4

ほ

ポート
作成, 6-42 ~ 6-45
「ポート」ノード, 6-38
「ポート番号」パラメータ, 4-6
「ホスト名」パラメータ, 4-6

め

「メソッド名」フィールド, 4-20
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,
4-23
メッセージ・タイプ
 イベント, 6-37
メディアエータ・アウトバウンド・プロセス, 7-2
メディアエータ・インバウンド・プロセス, 7-13

ゆ

「ユーザー」タブ, 4-10, 4-25, 6-40
ユーザー・パラメータ, 6-40
「ユーザー」パラメータ, 4-10, 4-25, 12-5

ら

ライセンス, 4-20
ライセンスおよびメソッド・ダイアログ・ボックス,
4-20
「ライセンス」フィールド, 4-20

り

リクエスト
 実行, 12-4
リクエスト・スキーマ, 4-18
リクエスト処理
 BAPI, 1-5
 IDoc, 1-5
 RFC, 1-5
リスナー, 1-5, 4-7
リスナー。チャンネルも参照
リソース・アダプタ, 5-1 ~ 5-2
 デプロイ, 1-2
リポジトリ・プロジェクト
 Web サービス, 4-5
 作成, 4-5 ~ 4-7
リポジトリ・プロジェクトの作成, 4-5 ~ 4-7
リモート関数モジュール, 12-4

れ

レスポンス・スキーマ, 4-18

ろ

「ログオン・チケット (SSO2)」パラメータ, 4-13
「ログオン・チケット (X509)」パラメータ, 4-13
ログ・ファイル, 12-1
ログ・レベル
 上書き, 5-4
論理システム, 6-54

