

Oracle® Fusion Middleware Application Adapters

Application Adapter for Siebel ユーザーズ・ガイド for 12c (12.2.1.2.0)

E85899-01

2016 年 12 月

Siebel システムおよび開発アプリケーションとの統合方法について説明します。

Oracle Fusion Middleware Oracle WebLogic Server 12c (12.2.1.2.0) Application Adapter 12c (12.2.1.2.0) for Siebel ユーザーズ・ガイド

E85899-01

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のもとに使用し、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	ix
対象読者	ix
ドキュメントのアクセシビリティについて	ix
関連ドキュメント	ix
表記規則	x
1 紹介	
アダプタの機能	1-1
Oracle Adapter Business Services Engine (BSE) のアーキテクチャ	1-2
Oracle Adapter J2CA の一般的なアーキテクチャ	1-3
Siebel アプリケーション・モデル	1-4
Siebel との統合	1-4
Siebel EAI アーキテクチャとの統合	1-5
Oracle Application Adapter for Siebel でのアプリケーション・エクスプローラの使用	1-6
BSE と Oracle Adapter J2CA デプロイメントの差異	1-6
サンプル・プロジェクト	1-7
クイック・スタート・ガイド	1-8
インストール	1-8
サード・パーティ・ライブラリ・ファイルのコピー	1-8
構成	1-9
WebLogic Server のデプロイメントおよび統合	1-10
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	1-10
SOA Suite でのサービス・コンポーネントの使用	1-10
Oracle Service Bus の使用	1-11
その他の機能	1-11
2 Oracle Application Server Adapter for Siebel 構成	
アプリケーション・エクスプローラの起動	2-2
リポジトリ設定の構成	2-2
リポジトリ構成の作成	2-3
BSE 用の構成の作成	2-3
J2CA 用の構成の作成	2-4
BSE または J2CA 構成への接続	2-5

Siebel 用の接続 (ターゲット) の確立	2-5
Siebel へのターゲットの定義.....	2-6
定義済のターゲットへの接続.....	2-9
Siebel からの切断.....	2-9
ターゲットの編集.....	2-10
Siebel のターゲットの削除.....	2-10
アプリケーション・システム・オブジェクトの表示	2-10
メタデータの表示.....	2-11
XML スキーマの作成	2-12
Siebel スキーマに関する考慮事項.....	2-13
Siebel のビジネス・オブジェクトまたはビジネス・サービス用の XML スキーマの作成...	2-13
Siebel のビジネス・コンポーネントまたはビジネス・サービス用の XML スキーマの作成..	2-14
特定の Siebel オブジェクトの検索.....	2-15
指定した順序でのフィールドの戻し.....	2-15
QueryWithView の使用.....	2-16
統合オブジェクトの使用に関する Siebel の前提条件	2-16
Siebel 統合オブジェクト用のスキーマの作成	2-17
Siebel 統合オブジェクト用の Siebel XDR または XSD スキーマの作成.....	2-17
Siebel の統合オブジェクト (IO) ノードの作成	2-19
Siebel 統合オブジェクト用の XML スキーマの作成.....	2-21
Siebel ビジネス・サービス用のサービス・ノードの作成	2-22
Web サービスの作成およびテスト (BSE 構成のみ)	2-24
Web サービスの作成.....	2-24
Web サービスのテスト.....	2-25
WSDL の生成 (J2CA 構成のみ)	2-27
イベント・アダプタの構成	2-28
チャンネルの作成および変更.....	2-29

3 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	3-1
アダプタのデプロイメント	3-1
アダプタ構成の更新	3-2
管理対象コネクタ・ファクトリ・オブジェクトの作成.....	3-3
複数の管理対象コネクタ・ファクトリ・オブジェクトの作成.....	3-4
追加の接続ファクトリ値用の WSDL ファイルの変更.....	3-6

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

概要	4-1
アダプタのデプロイメント	4-2
新規のアプリケーション・サーバー接続の構成	4-2
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)	4-7
リクエスト / レスポンス・サービス用の WSDL の生成.....	4-8
SOA 用の空のコンポジットの作成.....	4-9
BPEL アウトバウンド・プロセスの定義.....	4-11

BPEL アウトバウンド・プロセスのデプロイ	4-28
Oracle Enterprise Manager Console での入力 XML ドキュメントの起動	4-31
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	4-33
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)	4-34
イベント統合のための WSDL の生成	4-34
SOA 用の空のコンポジットの作成	4-41
BPEL インバウンド・プロセスの定義	4-42
BPEL インバウンド・プロセスのデプロイ	4-48
Siebel でのイベントのトリガー	4-49
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)	4-75
Web サービスを使用するリクエスト/レスポンス型サービス用の WSDL ファイルの生成...	4-76
SOA 用の空のコンポジットの作成	4-77
BPEL アウトバウンド・プロセスの定義	4-77

5 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	5-2
メディエータ・アウトバウンド・プロセスの構成 (J2CA 構成)	5-2
SOA 用の空のコンポジットの作成	5-2
メディエータ・アウトバウンド・プロセスの定義	5-3
メディエータ・アウトバウンド・プロセスのデプロイ	5-10
Oracle Enterprise Manager Console での入力 XML ドキュメントの起動	5-11
メディエータ・インバウンド・プロセスの構成 (J2CA 構成)	5-11
SOA 用の空のコンポジットの作成	5-11
メディエータ・インバウンド・プロセスの定義	5-11
メディエータ・アウトバウンド・プロセスの構成 (BSE 構成)	5-17
SOA 用の空のコンポジットの作成	5-18
メディエータ・アウトバウンド・プロセスの定義	5-18

6 Oracle SOA Suite での BPM 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
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)	6-2
BPM 用の空のコンポジットの作成	6-2
BPM アウトバウンド・プロセスの定義	6-3
12c のデプロイメントに関する既知の問題の調整	6-25
BPM アウトバウンド・プロセスのデプロイ	6-25
Oracle Enterprise Manager Console での入力 XML ドキュメントの起動	6-26
トランスフォーメーションを使用したイベント統合のためのインバウンド BPM プロセスの設計 (J2CA 構成)	6-27
BPM 用の空のコンポジットの作成	6-27
BPM インバウンド・プロセスの定義	6-28

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).....	6-42
BPM 用の空のコンポジットの作成.....	6-42
BPM アウトバウンド・プロセスの定義.....	6-42

7 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.....	7-1
sbconsole を使用したアウトバウンド処理の構成 (J2CA 構成).....	7-2
Oracle Service Bus の起動およびプロジェクト・フォルダの作成.....	7-2
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.....	7-6
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.....	7-6
WSDL ベースのビジネス・サービスの構成.....	7-7
ファイル・タイプ・ビジネス・サービスの構成.....	7-9
プロキシ・サービスでのパイプラインの構成.....	7-13
sbconsole を使用したインバウンド処理の構成 (J2CA 構成).....	7-22
Oracle Service Bus の起動およびプロジェクト・フォルダの作成.....	7-22
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.....	7-22
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.....	7-23
WSDL ベースのプロキシ・サービスの構成.....	7-24
ファイル・タイプ・ビジネス・サービスの構成.....	7-26
パイプラインの構成.....	7-29
sbconsole を使用したアウトバウンド処理の構成 (BSE 構成).....	7-36
Oracle Service Bus の起動およびプロジェクト・フォルダの作成.....	7-36
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.....	7-36
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.....	7-37
ファイル・タイプ・ビジネス・サービスの構成.....	7-38
WSDL ベースのビジネス・サービスの構成.....	7-38
プロキシ・サービスでのパイプラインの構成.....	7-40
Oracle Service Bus を使用した JMS プロキシ・サービスの構成 (J2CA 構成).....	7-49
Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成).....	7-64

8 JDeveloper を使用した Oracle Service Bus のアウトバウンド処理およびインバウンド処理の構成

JDeveloper を使用した OSB アウトバウンド処理の構成 (J2CA 構成).....	8-1
OSB 用の Service Bus アプリケーションの作成.....	8-2
OSB アウトバウンド処理の定義.....	8-3
OSB アウトバウンド処理のデプロイ.....	8-17
JDeveloper を使用した OSB インバウンド処理の構成 (J2CA 構成).....	8-19
OSB 用の Service Bus アプリケーションの作成.....	8-19
OSB インバウンド処理の定義.....	8-19

OSB インバウンド処理のデプロイ	8-27
JDeveloper を使用した OSB アウトバウンド処理の構成 (BSE 構成)	8-28
OSB 用の Service Bus アプリケーションの作成	8-28
OSB アウトバウンド処理の定義	8-28
OSB アウトバウンド処理のデプロイ	8-42
JDeveloper を使用した JMS インバウンド処理の構成 (J2CA 構成)	8-42
JDeveloper を使用した JMS アウトバウンド処理の構成 (J2CA 構成)	8-48
JDeveloper を使用した HTTP アウトバウンド処理の構成 (J2CA 構成)	8-55
9 主な機能	
ロギング機能の構成	9-1
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	9-1
Business Services Engine (BSE) のログ・ファイル管理の構成	9-8
診断機能の構成	9-11
サポートされるプロトコル	9-12
SOA デバッグ機能の構成	9-13
SOA デバッガの使用のガイドライン	9-13
前提条件	9-13
Oracle JDeveloper での BPEL プロセスのデバッグ	9-13
Oracle JDeveloper での OSB プロセスのデバッグ	9-26
例外フィルタ	9-44
例外フィルタの構成	9-44
Oracle SOA Suite (BPEL、メディエータまたは BPM) 用の資格証明マッピング	9-55
資格証明マッピングの構成	9-56
JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング	9-60
資格証明マッピングの構成	9-61
10 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	10-1
Oracle Application Adapter for Siebel に関する一般的な使用上の注意	10-1
アプリケーション・エクスプローラ	10-2
Siebel	10-3
Oracle Adapter J2CA	10-4
BSE エラー・メッセージ	10-5
BSE での一般的なエラー処理	10-5
アダプタ固有のエラー処理	10-5
A Siebel ワークフローの使用	
概要	A-1
Siebel ワークフロー	A-1
ポリシーを使用した Siebel EAI ワークフローの起動	A-2
Siebel ワークフロー - アウトバウンド	A-2
Siebel ワークフロー - インバウンド	A-3

Siebel ワークフローの作成	A-3
MQSeries トランスポートを使用したイベント用の Siebel ワークフローの作成	A-4
ファイル・トランスポートを使用したイベント用の Siebel ワークフローの作成	A-8
HTTP トランスポートを使用したイベント用の Siebel ワークフローの作成	A-13
MQSeries トランスポートを使用したサービス用の Siebel ワークフローの作成	A-16
ファイル・トランスポートを使用したサービス用の Siebel ワークフローの作成	A-21
HTTP トランスポートを使用したサービス用の Siebel ワークフローの作成	A-25

用語集

索引

はじめに

『Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for Siebel ユーザーズ・ガイド』をご利用いただきありがとうございます。このドキュメントでは、Siebel システムおよび開発アプリケーションとの統合方法について説明します。

対象読者

このドキュメントは、Siebel システムおよび開発アプリケーションとの統合を行う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 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ー変数を示します。
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 Siebel を介して Siebel システムに接続します。Oracle Application Adapter for Siebel によって Siebel システムとの接続と対話が可能になります。この章の内容は、次のとおりです。

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> は次の場所を指します。

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

-
- [1.1 項「アダプタの機能」](#)
 - [1.2 項「Siebel アプリケーション・モデル」](#)
 - [1.3 項「Siebel との統合」](#)
 - [1.4 項「Oracle Application Adapter for Siebel でのアプリケーション・エクスポーラの使用」](#)
 - [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 Siebel は、Siebel システムと他のアプリケーション、データベースまたは外部ビジネス・パートナー・システムとの間でリアルタイムにビジネス・データを交換する手段を提供します。[アダプタ](#)によって、Siebel とのインバウンドおよびアウトバウンドの処理が外部アプリケーションで可能になります。

Oracle Application Adapter for Siebel 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。Web サービス・サーブレットとしてデプロイすることもでき、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「Oracle Adapter Business Services Engine (BSE) のアーキテクチャ」
- 1.1.2 項「Oracle Adapter J2CA の一般的なアーキテクチャ」

Oracle Application Adapter for Siebel では、XML メッセージを使用して、非 Siebel アプリケーションがサービスやイベントを使用して Siebel と通信したり、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は、次のように定義されます。

- 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 Siebel ビジネス・イベントを開始できるようにします。
- イベント(インバウンド処理とも呼ばれる): Siebel ビジネス・イベントの発生時にのみアプリケーションが Siebel データにアクセスできるようにします。

イベント機能をサポートするため、チャンネルがサポートされます。**チャンネル**とは、バックエンドまたは他のタイプのシステムの特定インスタンスへの構成済の接続のことです。

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。チャンネル・コンポーネントは、ファイル・リーダー、HTTP リスナーまたは MQ リスナーです。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。

Oracle Application Adapter for Siebel:

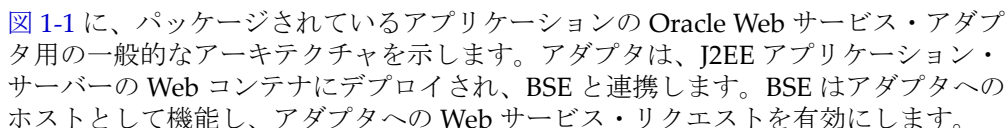
- Siebel ビジネス・サービス、ビジネス・コンポーネントおよび統合オブジェクトに対する同期および非同期の双方向のメッセージ交換がサポートされます。
- Siebel オブジェクト・マネージャを使用した Siebel メタデータの検索および XML スキーマまたは Web サービスの構築用の GUI ツールである Oracle WebLogic Server Adapter アプリケーション・エクスペローラ (アプリケーション・エクスペローラ) が含まれます。
- Siebel トランスポート (MQSeries、ファイルおよび HTTP) がサポートされます。MSMQ メッセージングもサポートされます。
- Oracle Adapter J2CA 用 XML スキーマ。
- BSE 用 Web サービス

Oracle Application Adapter for Siebel は、ビジネス・オブジェクト、ビジネス・コンポーネント、ビジネス・サービスおよび統合オブジェクトにより、23 のすべての Siebel Industry Applications (SIA) をサポートします。Siebel Industry Applications には、保険、先端技術、自動車、通信、メディア、金融サービス、生命科学、製造および消費財などの各種産業が含まれます。

Siebel Industry Applications は、ビジネス・オブジェクト、ビジネス・コンポーネント、ビジネス・サービスおよび統合オブジェクト形式の追加ビジネス・ロジックにより、特定のビジネス要件および特定産業のプロセスに合わせて調整されます。Oracle Application Adapter for Siebel はメタデータを公開および生成し、これらの産業固有オブジェクトと対話します。

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

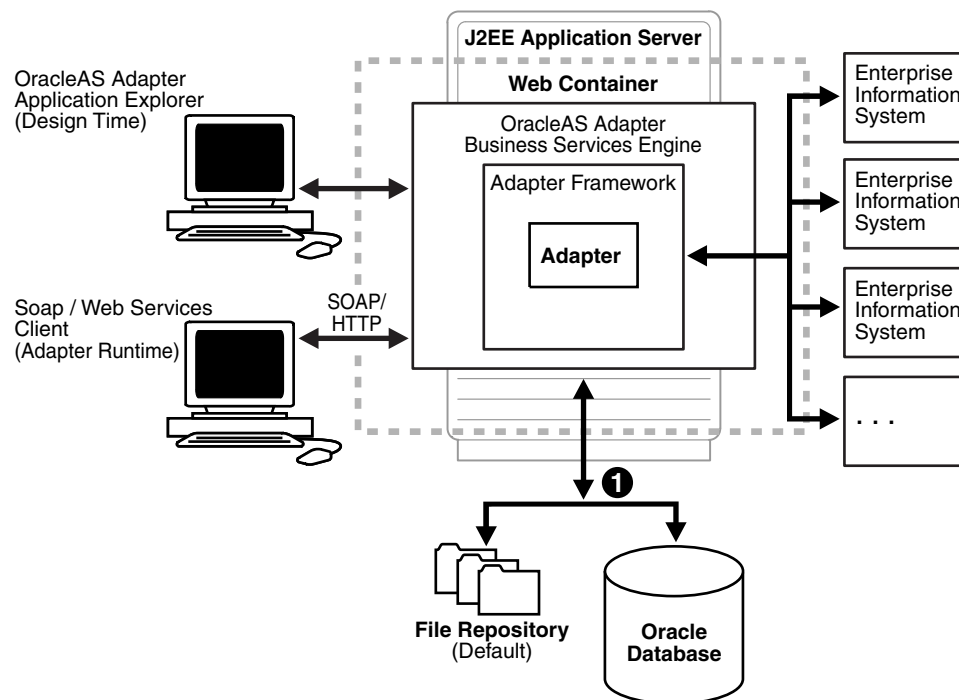
1.1.1 Oracle Adapter Business Services Engine (BSE) のアーキテクチャ

 **図 1-1** に、パッケージされているアプリケーションの Oracle Web サービス・アダプタ用の一般的なアーキテクチャを示します。アダプタは、J2EE アプリケーション・サーバーの Web コンテナにデプロイされ、BSE と連携します。BSE はアダプタへのホストとして機能し、アダプタへの Web サービス・リクエストを有効にします。

BSE とともにデプロイされるデザインタイム・ツールのアプリケーション・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およびサービスの構成を行います。これらの操作の実行中に作成されるメタデータは、BSE により、リポジトリ内に格納されます。

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。

図 1-1 Oracle Adapter Business Services Engine (BSE) の一般的なアーキテクチャ



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

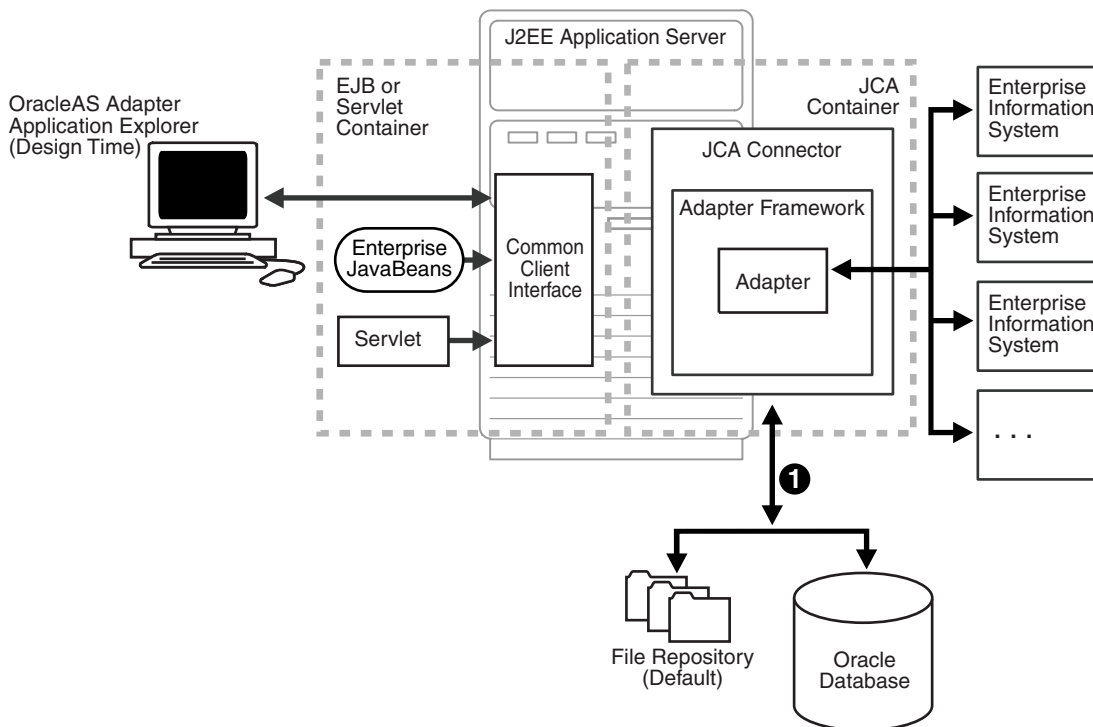
注意： BSE 用のファイル・リポジトリは本番環境では使用しないでください。

1.1.2 Oracle Adapter J2CA の一般的なアーキテクチャ

図 1-2 に、パッケージされているアプリケーションの Oracle Adapter J2CA 用の一般的なアーキテクチャを示します。The Oracle Adapter J2CA は標準の J2CA コンテナにデプロイされ、アダプタのホスト・コンテナとして機能します。コネクタは、リポジトリで構成されます。

アプリケーション・エクスプローラは、コネクタとともに動作する設計ツールで、これを使用して、アダプタ接続の構成、EIS オブジェクトの参照およびサービスの構成を行います。これらの操作の実行中に作成されるメタデータは、コネクタにより、リポジトリに格納されます。リポジトリには、ファイル・システムまたは Oracle データベースを使用できます。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記述子が関連付けられています。Oracle WebLogic Server デプロイメント記述子 ra.xml を編集して、複数のコネクタ・ファクトリを作成できます。詳細は、第 3 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 WebLogic Server Adapter J2CA 汎用アーキテクチャ



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

1.2 Siebel アプリケーション・モデル

Siebel エンタープライズ・アプリケーションは、基礎となるデータベースへの依存性を取り除くデータ抽象化レイヤーを定義します。これは、データベース構造を表す中間のビジネス・コンポーネントとビジネス・オブジェクトを使用して実現されます。ビジネス・コンポーネントは通常データベース内の表を表します。ビジネス・オブジェクトは関連するビジネス・コンポーネントのグループです。

指定されたビジネス・コンポーネントから、そのコンポーネントに対して定義された別のコンポーネントとの関係をナビゲートできます。コンポーネントの関係をたどるために使用されるパスは、ナビゲーション・パスと呼ばれます。たとえば特定アカウントのアドレスをすべて取得する必要がある場合、アカウントとアドレス間の親/子関係をたどってアドレスを取得できます。ナビゲーション・パスを使用することで、Siebel システム内に定義されたほぼすべてのビジネス・コンポーネントの関係をたどることができます。

Siebel では統合オブジェクトは Siebel ビジネス・コンポーネントと似ていますが、より複雑で階層的なデータ関係を表します。

1.3 Siebel との統合

Oracle Application Adapter for Siebel を使用して、アカウントの追加/更新などの Siebel ビジネス・プロセスを開始したり、このアダプタを統合の一部として使用して、Siebel と非 Siebel システムを接続できます。Oracle Application Adapter for Siebel は双方向であり、Siebel が送出する Siebel XML ドキュメントを受信して Siebel からのイベントを検出できます。

この項では、次のトピックについて説明します。

■ 1.3.1 項「Siebel EAI アーキテクチャとの統合」

Siebel XML ドキュメントを使用して Siebel と統合する場合、アダプタ・アプリケーションの開発者は既存の Siebel 統合オブジェクトを使用するか、使用する新規の Siebel 統合オブジェクトを Siebel ワークフロー内に作成する必要があります。ワークフローはインバウンドまたはアウトバウンドの Siebel XML を処理し、様々なトランスポート (MQSeries、ファイルおよび HTTP など) を使用して外部システムとトランザクションを交換します。Siebel ワークフローは、通常、Siebel の管理者または開発者により Siebel ワークフロー管理画面を使用して作成されます。

Java データ Bean または COM データ・インタフェースを使用して直接 Siebel と統合する場合、Oracle Application Adapter for Siebel に Siebel 統合オブジェクトや Siebel ワークフローは必要ありません。そのかわり、Siebel ビジネス・サービスおよび Siebel ビジネス・コンポーネントを直接実行します。

次の表に、Siebel のオブジェクトとプロセスを示します。

表 1-1 Siebel のオブジェクトとプロセス

Siebel のオブジェクト	API またはトランスポート	プロセス
ビジネス・サービス	Java データ Bean (Siebel バージョン 6.3-8.0) Com データ・インタフェース (Siebel バージョン 6.01-6.2)	サービス
ビジネス・コンポーネント	Java データ Bean (Siebel バージョン 6.3-8.0) Com データ・インタフェース (Siebel バージョン 6.01-6.2)	サービス
統合オブジェクト	ファイル HTTP MQSeries MQ 読取り	イベント、サービス イベント、サービス イベント、サービス サービス

1.3.1 Siebel EAI アーキテクチャとの統合

Siebel は、Siebel EAI(Enterprise Application Integration) フレームワークと Business Integration Manager 機能を使用して他のアプリケーションおよびシステムとの統合を可能にします。Oracle Application Adapter for Siebel は、Siebel フレームワーク内での操作時に、Siebel EAI フレームワークを使用し、様々な統合アクセス方法を利用して最大限の柔軟性と機能を提供します。

Oracle Application Adapter for Siebel は次の統合アクセス方法をサポートします。

- Siebel ビジネス・コンポーネントまたは Siebel ビジネス・サービスを含むサービスに対する Siebel Java データ Bean。
- Siebel ビジネス・コンポーネントまたは Siebel ビジネス・サービスを含むサービスに対する Siebel COM データ・インタフェース。
- Siebel 統合オブジェクトを含むイベントおよびサービスに対する Siebel XML。

1.4 Oracle Application Adapter for Siebel でのアプリケーション・エクスペローラの使用

アプリケーション・エクスペローラは、エクスペローラ・メタファを使用して Siebel システムのビジネス・サービス、ビジネス・オブジェクト、ビジネス・コンポーネントおよび統合オブジェクトを参照します。エクスペローラにより、関連オブジェクト用の XML スキーマおよび Web サービスを作成できます。Oracle Application Adapter for Siebel を介して Siebel にアクセスする外部アプリケーションは、XML スキーマまたは Web サービスを使用して外部アプリケーションとアダプタとの間のデータの受渡しを行います。

アプリケーション・エクスペローラは、Siebel により提供されるインタフェースと Siebel アプリケーション・システムに関する詳細な情報を使用して、ビジネス・オブジェクト・メタデータにアクセスします。オブジェクトを選択した後、アプリケーション・エクスペローラは XML スキーマまたは Web サービスを生成して Oracle Application Adapter for Siebel とともに使用するオブジェクトを定義できます。

アプリケーション・エクスペローラの主な機能は次のとおりです。

- 様々なアプリケーション・システムに接続したり、探索する機能。
- アプリケーション・システム・オブジェクト・メタデータへのアクセス。
- XML スキーマおよび Web サービスを生成するためのポイントアンドクリック処理。

関連項目：

- *Oracle Application Server Adapter コンセプト・ガイド*
- *Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストラクション・ガイド*

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

Oracle Application Adapter for Siebel を Oracle SOA Suite コンポーネント (BPEL、メディアータ、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 のデプロイメント間の差異には、次の 2 つの要素があります。これらの要素を理解することは、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。
2. 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

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およびインバウンド統合シナリオを示している Oracle Application Adapter for Siebel のサンプル・パッケージは、Application Adapters インストールでパッケージされています。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
アウトバウンド BPEL プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPEL\J2CA\Outbound_Project
インバウンド BPEL プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPEL\J2CA\Inbound_Project
アウトバウンド BPEL プロセス (BSE)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPEL\BSE\Outbound_Project
アウトバウンド・メディエータ・プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\Mediator\J2CA\Outbound_Project
インバウンド・メディエータ・プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\Mediator\J2CA\Inbound_Project
アウトバウンド・メディエータ・プロセス (BSE)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\Mediator\BSE\Outbound_Project
アウトバウンド BPM プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPM\J2CA\Siebel_Sample_J2CA_BPM_Outbound_Project
インバウンド BPM プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPM\J2CA\Inbound_Project
アウトバウンド BPM プロセス (BSE)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPM\BSE\Outbound_Project
アウトバウンド OSB sbconsole プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB\J2CA\Siebel_Sample_J2CA_OSB_Outbound_Project
インバウンド OSB sbconsole プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB\J2CA\Siebel_Sample_J2CA_OSB_Inbound_Project
アウトバウンド OSB sbconsole プロセス (BSE)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB\BSE\Siebel_Sample_BSE_OSB_Outbound_Project
アウトバウンド OSB Jdeveloper プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB_Jdeveloper\J2CA\Siebel_Sample_J2CA_OSB_Outbound_Project

サンプル・プロジェクト 場所

インバウンド OSB Jdeveloper プロセス (J2CA)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB_Jdeveloper\J2CA\Siebel_Sample_J2CA_OSB_Inbound_Project
アウトバウンド OSB Jdeveloper プロセス (BSE)	<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB_Jdeveloper\BSE\Siebel_Sample_BSE_OSB_Outbound_Project

1.7 クイック・スタート・ガイド

この項では、Oracle Application Adapter for Siebel をインストールして構成し、即座に使用するための基本的な手順について説明します。内容は次のとおりです。

- [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「その他の機能」](#)

1.7.1 インストール

Oracle Application Adapter for Siebel 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 Adapters 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

アダプタのインストールが完了したら、Siebel に必要なサード・パーティ・ライブラリ・ファイル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
```

Siebel バージョン 6.2 以下のエンコーディング設定と前提条件の詳細は、[第 2 章「Oracle Application Server Adapter for Siebel 構成」](#)にある次のトピックを参照してください。

- UNIX プラットフォームでのエンコーディング・サポート
- 必要なエンコーディング・オプションの追加 (すべての UNIX プラットフォーム)
- バージョン 6.2 以下の Siebel 接続の前提条件

1.7.3 構成

<ADAPTER_HOME>に移動し、次の変更を加えます。

1. 表 1-2 に示すように、*iwafjca.rar\META-INF\ra.xml* を開き、指定した *config-property-name* パラメータの下に次の値を追加します。

表 1-2

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。 jca_sample

2. 表 1-3 に示すように、*ibse.war\WEB-INF\web.xml* を開き、指定した *param-name* パラメータの下に次の値を追加します。

表 1-3

Param-Name	Param-Value
<i>ibseroot</i>	<ADAPTER_HOME>\ibse.war 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\ibse.war ■ OSB の場合 C:\12c_OSB\osb\3rdparty\ApplicationAdapters\ibse.war
<i>IWay.home</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>Iway.config</i>	構成の名前。次に例を示します。 IBSE

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提供されます。データベース・リポジトリの構成と一般的な構成情報の詳細は、[第2章「Oracle Application Server Adapter for Siebel 構成」](#) および [第3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。

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、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、デプロイしたアダプタ・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の作成の詳細は、[第3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。

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

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

- アプリケーション・エクスプローラの起動：[2.1 項「アプリケーション・エクスプローラの起動」](#)
- BSE 構成の作成：[2.3.1 項「BSE 用の構成の作成」](#)
- J2CA 構成の作成：[2.3.2 項「J2CA 用の構成の作成」](#)
- 作成済みの構成の接続：[2.3.3 項「BSE または J2CA 構成への接続」](#)
- ターゲットの作成と接続：[2.4 項「Siebel 用の接続 \(ターゲット\) の確立」](#)
- 統合オブジェクトの使用：[2.7 項「統合オブジェクトの使用に関する Siebel の前提条件」](#)、[2.8 項「Siebel 統合オブジェクト用のスキーマの作成」](#) および [2.9 項「Siebel の統合オブジェクト \(IO\) ノードの作成」](#)
- サービス・ノードの使用：[2.10 項「Siebel ビジネス・サービス用のサービス・ノードの作成」](#)
- Web サービスの作成とテスト：[2.11 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：[2.12 項「WSDL の生成 \(J2CA 構成のみ\)」](#)
- チャネルの作成と使用：[2.13 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 Siebel は、BPEL、メディエータ、BPM など、SOA Suite のサービス・コンポーネントと統合されます。JDeveloper で必要なプロセスが作成され、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、次の項を参照してください。

- [第4章「Oracle SOA Suite での BPEL サービス・コンポーネントの統合」](#)
- [第5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6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 Siebel は Oracle Service Bus (OSB) と統合され、Web サービスの簡易な統合を可能にします。Oracle Service Bus コンソールで必要なプロセスが作成されます。JDeveloper でプロセスを作成し、SOA サーバーにデプロイすることもできます。

OSB コンソールの使用の詳細は、第 7 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 8 章「[JDeveloper を使用した Oracle Service Bus のアウトバウンド処理およびインバウンド処理の構成](#)」を参照してください。

1.7.8 その他の機能

次に、その他の機能のリストと、このユーザーズ・ガイドの関連する項を示します。

- 例外フィルタの構成: [9.4 項「例外フィルタ」](#)
- 資格情報のマッピングの構成:
 - [9.5 項「Oracle SOA Suite \(BPEL、メディエータまたは BPM\) 用の資格証明マッピング」](#)
 - [9.6 項「JDeveloper を使用した Oracle Service Bus \(OSB\) の資格証明マッピング」](#)

Oracle Application Server Adapter for Siebel 構成

この章では、Oracle Application Adapter for Siebel の構成方法および Siebel ビジネス・オブジェクトのスキーマの作成方法について説明します。内容は次のとおりです。

- 2.1 項「アプリケーション・エクスプローラの起動」
- 2.2 項「リポジトリ設定の構成」
- 2.3 項「リポジトリ構成の作成」
- 2.4 項「Siebel 用の接続 (ターゲット) の確立」
- 2.5 項「アプリケーション・システム・オブジェクトの表示」
- 2.6 項「XML スキーマの作成」
- 2.7 項「統合オブジェクトの使用に関する Siebel の前提条件」
- 2.8 項「Siebel 統合オブジェクト用のスキーマの作成」
- 2.9 項「Siebel の統合オブジェクト (IO) ノードの作成」
- 2.10 項「Siebel ビジネス・サービス用のサービス・ノードの作成」
- 2.11 項「Web サービスの作成およびテスト (BSE 構成のみ)」
- 2.12 項「WSDL の生成 (J2CA 構成のみ)」
- 2.13 項「イベント・アダプタの構成」

バージョン 6.2 以下の Siebel 接続の前提条件

Siebel バージョン 6.2 以下の場合のみ、次の手順を実行して、J2CA 構成用 COM 接続を使用して Siebel システムに接続する必要があります。

1. アダプタがインストールされているのと同じシステムに Siebel シック・クライアントをインストールします。
2. 同じシステムにデータベース・クライアント (Microsoft SQL Server または Oracle) をインストールします。
3. アダプタの lib フォルダの Siebel .DLL ファイル (iwsiebel.local.dll および iwsiebel.core.dll) をアプリケーション・サーバーのパスに追加する必要があります。
4. uagent.cfg ファイルを編集して、データ・ソース・パラメータ値を "local" から "server" に変更します。

uagent.cfg ファイルは、次の Siebel シック・クライアント・フォルダにあります。
c:\sea\client\bin

- 適切なパラメータを使用して SEA MSQI のデータ・ソースを編集します。
Windows では、「コントロールパネル」、「管理ツール」、「データ ソース (ODBC)」の順にアクセスすると、データ・ソースを編集できます。
- アダプタ・ターゲット接続を作成する場合、次のターゲット・タイプを使用します。
Siebel 6.2- (ローカルの COM アクセス実装)
- アダプタ・ターゲット接続を作成する場合、次のように `uagent.cfg` ファイルのフル・パスを指定します。
`c:\sea\client\bin\uagent.cfg`

2.1 アプリケーション・エクスプローラの起動

アプリケーション・エクスプローラを起動する手順

- 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。
- コマンド・プロンプト・ウィンドウを開きます。
- 次のディレクトリに移動します。
`<ADAPTER_HOME>\user_projects\domains\base_domain\bin`
- `setDomainEnv.cmd` (Windows) または `../setDomainEnv.sh` (UNIX/Linux) を実行します。
このコマンドは、Oracle WebLogic Server 環境でアプリケーション・エクスプローラ用のクラス・パスとその他の環境変数を設定します。また、アプリケーション・エクスプローラで Oracle WebLogic Server API にアクセスし、WSDL ファイルを Oracle Service Bus (OSB) Console にパブリッシュすることもできます。
- コマンド・プロンプトを開けたままにします。
- 次のディレクトリに移動します。
`<ADAPTER_HOME>\tools\iwae\bin`
- `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。新規ターゲットを Siebel システムに定義する準備が整いました。

注意: `iwae.sh` ファイルを UNIX または Linux プラットフォームで実行する前に、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
```

2.2 リポジトリ設定の構成

リポジトリには、構成の詳細、アダプタのターゲット、チャンネルに関する情報、およびその他の構成情報が格納されます。BSE および J2CA リポジトリ設定の詳細は、*Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストール・ガイド* (Oracle 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。

2.3 リポジトリ構成の作成

アプリケーション・エクスプローラを Oracle Application Adapter for Siebel とともに使用する前に、レ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。

この項では、次のトピックについて説明します。

- 2.3.1 項「BSE 用の構成の作成」
- 2.3.2 項「J2CA 用の構成の作成」
- 2.3.3 項「BSE または J2CA 構成への接続」

リポジトリは、設計時にアプリケーション・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るメタデータの格納に使用されます。リポジトリ内の情報は実行時にも参照されます。

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

2.3.1 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 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。

この項では、次のトピックについて説明します。

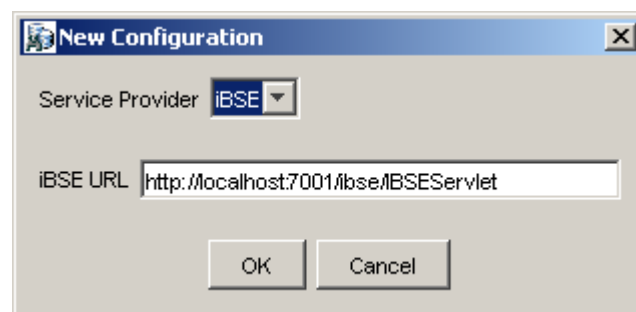
- 2.3.1.1 項「BSE 用の新規構成の定義」

2.3.1.1 BSE 用の新規構成の定義

BSE 用の新規構成を作成する手順

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 (SampleConfig など) を入力し、「OK」をクリックします。
[図 2-1](#) に示すように、「新規構成」ダイアログが表示されます。

図 2-1 「新規構成」ダイアログ



4. 「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、または次のフォーマットで別の URL と置き換えます。

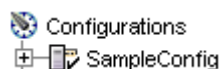
http://host name:port/ibse/IBSEServlet

host name は Oracle WebLogic Server がインストールされたシステムで、port は Oracle WebLogic Server がリスニングしている HTTP ポートの番号です。

6. 「OK」をクリックします。

図 2-2 に示すように、新規構成を示すノードがルートの「構成」ノードの下に表示されます。

図 2-2 SampleConfig ノード



2.3.2 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 用の新規構成を定義する手順

1. アプリケーション・エクスプローラを起動します。
2. 図 2-3 に示すように、「構成」を右クリックし、「新規」を選択します。

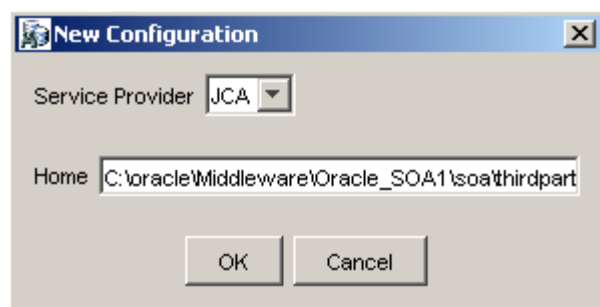
図 2-3 「構成」ノード



図 2-4 に示すように、「新規構成」ダイアログが表示されます。

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

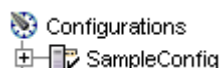
図 2-4 「新規構成」ダイアログ



4. 「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 2-5 に示すように、新規構成を示すノードがルートの「構成」ノードの下に表示されます。

図 2-5 SampleConfig ノード



Oracle Adapter J2CA 構成フォルダがアダプタのインストールに基づき次の場所に保存されます。

```
<ADAPTER_HOME>\config\configuration_name
```

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

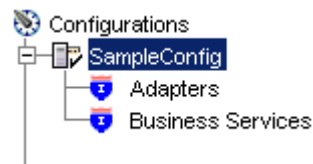
2.3.3 BSE または J2CA 構成への接続

新規構成に接続する手順

1. 接続する構成 (SampleConfig など) を右クリックします。
2. 「接続」を選択します。

「アダプタ」 (Web サービスとも呼ばれる) のノードが表示されます。「ビジネス・サービス」ノードは、BSE 構成の場合にのみ表示されます。J2CA 構成に接続している場合は、「ビジネス・サービス」ノードは表示されません。図 2-6 に、SampleConfig という名前の BSE 構成の例を示します。

図 2-6 「構成」ノードの下に表示される新規 SampleConfig 構成



- 「アダプタ」ノードを使用して、Siebel とのインバウンド対話を作成します。たとえば、「アダプタ」ノードを使用して Siebel を更新するサービスを構成します。
- 「イベント」ノード (J2CA 構成の場合のみ) を使用して、Siebel で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 構成の場合のみ) を使用して、「アダプタ」ノードで作成された 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、Siebel への新規ターゲットを定義する準備ができました。

2.4 Siebel 用の接続 (ターゲット) の確立

Siebel のビジネス・サービス、ビジネス・コンポーネントおよび統合オブジェクトを参照するには、Siebel 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、パラメータは自動的に保存されます。

この項では、次のトピックについて説明します。

- 2.4.1 項 「Siebel へのターゲットの定義」
- 2.4.2 項 「定義済のターゲットへの接続」
- 2.4.3 項 「Siebel からの切断」
- 2.4.4 項 「ターゲットの編集」
- 2.4.5 項 「Siebel のターゲットの削除」

重要 (すべての UNIX プラットフォーム): UNIX 環境で BSE または J2CA 構成を使用して Siebel ターゲットへの接続を試行する前に、2-6 ページ 「必要なエンコーディング・オプションの追加 (すべての UNIX プラットフォーム)」 で説明されている追加

手順を実行する必要があります。この項で説明するエンコーディング・オプションの追加を行わないと、エラーが発生し、Siebel ターゲットに接続できなくなります。エラー・メッセージは、次のようにそのエンコーディングがサポートされていないことを示す場合があります。

エラー：アダプタのアクティブ化で問題が発生しました -- UTF-8 はサポートされていません。詳細はログを確認してください。

エラー：ターゲット [Siebel] の取得中にエラーが発生しました -- UTF-8 はサポートされていません。

必要なエンコーディング・オプションの追加 (すべての UNIX プラットフォーム)

Siebel ターゲットに接続する前に、次の手順を実行します。

1. 次の Java ファイル・エンコーディング・オプションを **startWebLogic.sh** ファイルに追加します。

```
JAVA_OPTIONS="${SAVE_JAVA_OPTIONS} -Dfile.encoding=ISO8859_1"
```

startWebLogic.sh ファイルは次のディレクトリにあります。

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

2.4.1 Siebel へのターゲットの定義

Siebel ターゲットの定義に必要な接続パラメータは、次のディレクトリにある **eapps.cfg** ファイルから取得できます。

```
drive:\SiebelRoot\SWEApp\BIN
```

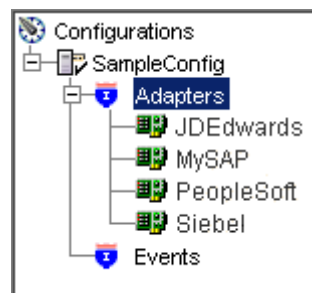
Siebelroot は、Siebel のインストール・ディレクトリです。

J2CA 構成を使用する場合、ターゲットの作成、更新および削除には Oracle WebLogic Server の再起動が必要です。また、Oracle WebLogic Server を再起動する前に、アプリケーション・エクスプローラを閉じることも忘れないでください。

Siebel へのターゲットを定義する手順

1. [図 2-7](#) に示すように、左ペインで「アダプタ」ノードを展開します。

図 2-7 「アダプタ」ノード



2. 「Siebel」ノードを右クリックして「ターゲットの追加」を選択します。「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、新規ターゲットの名前を入力します。
 - b. 「説明」フィールドに、説明を入力します (オプション) 。
 - c. 「ターゲット・タイプ」リストから「**Java データ Bean 接続**」(デフォルト) を選択します。

3. 「OK」をクリックします。

図 2-8 に示すように、「Java データ Bean 接続」ダイアログが表示されます。

図 2-8 「Java データ Bean 接続」ダイアログ

次の手順に示すようにシステム情報を入力します。

- a. 「ゲートウェイ・サーバー」フィールドにサーバーの名前を入力します。デフォルト (通常、2320) 以外のポートを使用するゲートウェイ・サーバーを指定するには、`gateway name:port number` のようにコロンとポート番号を追加します。
- b. 「企業名」フィールドに、適切な名前を入力します。
- c. 「Siebel サーバー」フィールドに Siebel サーバーの名前を入力します。Siebel 7.7、7.8 または 8 のシステムに接続する場合、このフィールドに値を指定しないでください。
- d. 「ユーザー」フィールドに、ユーザー名を入力します。
- e. 「パスワード」フィールドに、ユーザー名に対応するパスワードを入力します。
- f. Siebel のバージョン・リストで、**Siebel 7.7 以上** (デフォルト) または **Siebel 7.5 以下** を選択します。
- g. 図 2-9 に示す「詳細」タブをクリックし、次の点を確認します。

言語

オブジェクト・マネージャ

図 2-9 「Java データ Bean 接続」 タブ

The screenshot shows a dialog box titled "Java Data Bean Connection" with a close button (X) in the top right corner. It has two tabs: "Logon" and "Advanced". The "Advanced" tab is active. The dialog contains the following fields:

- Language**: A text box containing "enu".
- Object Manager***: A text box containing "EAIObjMgr".
- Repository Name**: A text box containing "Siebel Repository".
- Encryption**: A dropdown menu with "None" selected.

At the bottom of the dialog are "OK" and "Cancel" buttons. Below the dialog, a red text note reads: "Fields marked with * are required."

オブジェクト・マネージャ

Siebel 7.0.3 の場合、デフォルトのオブジェクト・マネージャは EAIObjMgr です。Siebel 7.7 の場合、デフォルトは EAIObjMgr_enu です。Siebel 7.7 では、オブジェクト・マネージャ名の末尾に言語拡張子 (_enu など) を追加する必要があります。システムに適用される特定の名前については、Siebel 管理者に確認してください。

リポジトリ名

リポジトリを指定しない場合、利用可能なすべてのリポジトリからオブジェクトの全リストが戻されます。指定したリポジトリが見つからない場合は、空のオブジェクト・リストが戻されます。

指定する構成パラメータは、Siebel クライアント・アプリケーションが Siebel システムに接続するために使用するパラメータです。これらのパラメータの詳細は、Siebel のドキュメントを参照するか、Siebel システム管理者に問い合わせてください。

暗号化

設計時に Siebel アダプタを使用してターゲットを作成する場合の「暗号化」タブに導入されています。このパラメータには、「なし」および「RSA」という 2 つの値があります。デフォルト値は暗号化が実行されない「なし」です。「RSA」を選択すると、指定されたオブジェクト・マネージャへの RSA 暗号化された接続が確立されます。

RSA 暗号化を使用するには、「オブジェクト・マネージャ」に SCCObjMgr_enu が指定される必要があります。

注意: 通常、これらのパラメータは、Siebel サーバーの root/bin/<language> ディレクトリに格納される Siebel 構成ファイルにあります。language は、インストールした言語の Siebel コードです (アメリカ英語の場合 enu)。たとえば、Windows プラットフォーム上の Siebel バージョン 7 以上の場合、Siebel Call Center モジュールについては、これらの値は uagent.cfg ファイルにあります。詳細は、Siebel 管理者に確認するか、Siebel Bookshelf のドキュメントを参照してください。

4. 「OK」をクリックします。

作成したターゲットが、左ペインの「Siebel」ノードの下に表示されます。

2.4.2 定義済のターゲットへの接続

定義済のターゲットに接続する手順

1. 図 2-10 に示すように、「Siebel」ノードを展開し、接続するターゲットの名前をクリックします。

図 2-10 切断されている Siebel ターゲット



2. 左ペインでターゲット名を右クリックし、「接続」を選択します。

図 2-11 に示すように、ターゲット・アイコンが変化し、Siebel システムに接続されたことを示します。

図 2-11 Siebel ターゲット・ノード



Siebel システムで使用可能なビジネス・オブジェクト、ビジネス・サービスおよび統合オブジェクトを参照できます。

2.4.3 Siebel からの切断

異なるアプリケーション・システムに対して開いている複数の接続を維持できますが、使用していない接続は切断することをお勧めします。

Siebel から切断する手順

1. 左ペインで、接続しているターゲットを選択します。
2. ターゲット名を右クリックし、「切断」を選択します。

アプリケーション・システムから切断すると、ターゲットは削除されますが、ノードは残ります。図 2-12 に示すように、左ペインの「SiebelConnection」ノードが変更され、ターゲットが切断されたことが反映されます。

図 2-12 切断されている Siebel ターゲット

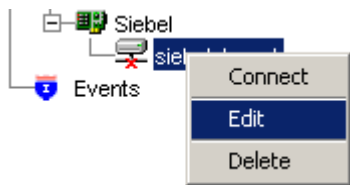


2.4.4 ターゲットの編集

ターゲットを編集する手順

1. 左ペインで、編集するターゲットが切断されていることを確認します。
2. 図 2-13 に示すように、切断されているターゲットを右クリックして「**編集**」を選択します。

図 2-13 「編集」オプション



右側に「編集」ペインが表示されます。

3. ターゲットの情報を変更します。
4. 「OK」をクリックします。

2.4.5 Siebel のターゲットの削除

ターゲットは、切断して閉じるだけでなく、削除できます。ターゲットを削除すると、アプリケーション・エクスプローラの左ペインの Siebel ターゲットのリストにそのノードが表示されなくなります。

ターゲットを削除する場合、Oracle WebLogic Server を再起動してランタイム用にリポジトリを更新する必要があります。

ターゲットを削除する手順

1. 左ペインでターゲットを選択します。
2. ターゲット名を右クリックし、「**削除**」を選択します。
確認のメッセージが表示されます。
3. 「OK」をクリックして、選択したターゲットを削除します。
Siebel 接続ノードが左ペインから削除されます。

2.5 アプリケーション・システム・オブジェクトの表示

アプリケーション・エクスプローラでは、Siebel アプリケーション・システムのすべてのオブジェクトを表示できます。これにより、Siebel データ構造を理解できます。パラメータ、データ型および Siebel データのその他の属性を右ペインで確認できます。

この項では、次のトピックについて説明します。

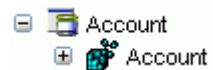
- 2.5.1 項「メタデータの表示」

2.5.1 メタデータの表示

メタデータを表示する手順

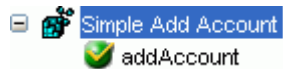
1. アプリケーション・エクスプローラを起動して Siebel システムに接続します。
2. 左ペインで、スキーマの生成対象のコンポーネントを含む「ビジネス・オブジェクト」または「ビジネス・サービス」を展開します。
3. 「ビジネス・オブジェクト」ノードまたは「ビジネス・サービス」ノードを展開します。
4. 「ビジネス・コンポーネント」または「ビジネス・サービス」ノードを展開して、その下のオブジェクトを表示します。
 - 「ビジネス・コンポーネント」の場合、[図 2-14](#) に示すように、該当するノード（「アカウント」など）を選択します。

図 2-14 「アカウント」ノード



- Siebel ビジネス・サービスの場合、[図 2-15](#) に示すように、該当するオブジェクト（「addAccount」など）を選択します。

図 2-15 シンプルなアカウント追加ノード



5. 右ペインで、プロパティ表の「表」行にある省略記号 (...) をクリックします。
[図 2-16](#) に示すように、メタデータ表が右ペインに表示されます。

図 2-16 Siebel オブジェクトのメタデータ表

Name	Type	Required	MultiValued	ReadOnly	Active
Account Co...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Account Con...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Account Mar...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Account Org...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Account Pro...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Account Role	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Account Stat...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Account Trend	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Address Act...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Address Id	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Address Inte...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Agreement E...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Agreement N...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Agreement S...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Agreement S...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Algorithm Type	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Alias	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Annual Reve...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Assignment ...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Assignment ...	string	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Assignment ...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Assignment ...	boolean	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Assignment ...	string	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

2.6 XML スキーマの作成

ビジネス・サービスおよびビジネス・コンポーネント用のサービス・スキーマは、アプリケーション・エクスプローラを使用して作成できます。

この項では、次のトピックについて説明します。

- 2.6.1 項「Siebel スキーマに関する考慮事項」
- 2.6.2 項「Siebel のビジネス・オブジェクトまたはビジネス・サービス用の XML スキーマの作成」
- 2.6.3 項「Siebel のビジネス・コンポーネントまたはビジネス・サービス用の XML スキーマの作成」
- 2.6.4 項「特定の Siebel オブジェクトの検索」
- 2.6.5 項「指定した順序でのフィールドの戻し」
- 2.6.6 項「QueryWithView の使用」

次の項では、J2CA 環境または Web サービス環境で使用するために Oracle Application Adapter for Siebel をデプロイするときに、アダプタ用のスキーマを作成する方法について説明します。Oracle Application Adapter for Siebel を Web サービス環境にデプロイする場合は、2-24 ページ「Web サービスの作成およびテスト (BSE 構成のみ)」を参照してください。

2.6.1 Siebel スキーマに関する考慮事項

レコードを Siebel に挿入する場合、データは、ユーザーが指定することも、デフォルト値または別のシステム生成値を持つよう Siebel を構成することもできます。たとえば、レコードの挿入時、デフォルトでは、Account ビジネス・コンポーネントの通貨コードは 'USD' で、Siebel システムによって生成される ROW_ID などのシステム・フィールドを持ちます。Siebel API では、これは区別されません。したがって、Oracle Application Adapter for Siebel は、ユーザーが入力する必要のある必須フィールドはどれで、Siebel によって移入される必須フィールドはどれかを見ることができません。そのため、アダプタ・スキーマは、要素に `minoccurs=0` を設定することですべての要素をオプションとするよう変更されました。

したがって、すべてのユーザーは、Siebel Tools を使用して必須フィールドを確認し、Siebel サービス (アウトバウンド) に対するペイロード (リクエスト XML ドキュメント) を作成する必要があります。

2.6.2 Siebel のビジネス・オブジェクトまたはビジネス・サービス用の XML スキーマの作成

Siebel のビジネス・サービス・メソッド (Add メソッドなど) およびビジネス・コンポーネント用のスキーマは、アプリケーション・エクスプローラを使用して作成します。スキーマを作成した後、スキーマを使用して、ビジネス・サービスまたはビジネス・コンポーネントのサービス・リクエスト・スキーマとレスポンス・スキーマを生成できます。

Siebel ビジネス・オブジェクトには、1 つ以上の Siebel ビジネス・コンポーネントが含まれます。ビジネス・コンポーネントは、関連するビジネス・オブジェクトをクリックすると表示できます。

図 2-17 に示すように、「Account」ビジネス・オブジェクトを展開して、すべての使用可能なビジネス・コンポーネントを表示できます。

図 2-17 「Account」ビジネス・オブジェクト



2.6.3 Siebel のビジネス・コンポーネントまたはビジネス・サービス用の XML スキーマの作成

ビジネス・コンポーネントまたはビジネス・サービスに対してサービスのリクエスト・スキーマとレスポンス・スキーマを生成する手順

1. アプリケーション・エクスプローラを起動して Siebel システムに接続します。
2. 左ペインで「ビジネス・オブジェクト」または「ビジネス・サービス」ノードを展開します。
3. 「ビジネス・コンポーネント」または「ビジネス・サービス」を展開し、その下にあるオブジェクトを表示します。
 - ビジネス・コンポーネントの場合は、[図 2-18](#) に示すように、「ビジネス・オブジェクト」ノードを展開し、使用するビジネス・コンポーネントを展開してから、使用するノードを展開し、スキーマを作成するメソッドを選択します。

図 2-18 「Account」ビジネス・オブジェクト下で選択されている「insert」メソッド



- Siebel ビジネス・サービスの場合は、[図 2-19](#) に示すように、スキーマの作成対象のオブジェクトを含む「ビジネス・サービス」ノードを展開します。

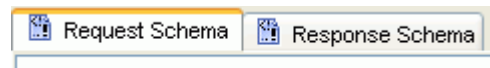
図 2-19 「Add Account」ビジネス・サービス下の「addAccount」オブジェクト



4. ノードを右クリックして「スキーマ」を選択します。
アプリケーション・スキーマが Siebel リポジトリにアクセスし、スキーマを作成します。

[図 2-20](#) に示すスキーマ・タブが右ペインに表示されます。

図 2-20 「リクエスト・スキーマ」タブ



5. スキーマを表示するには、表示するスキーマに対応する省略記号タブをクリックします。

[図 2-21](#) に示すように、スキーマが右ペインに表示されます。

図 2-21 XML スキーマ

```

<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:z="
  <xsd:element name="Siebel">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="insert" type="z:record"/>
      </xsd:sequence>
      <xsd:attribute name="location" type="xsd:string" use="optional"
    </xsd:complexType>
  </xsd:element>
  <xsd:complexType name="record">
    <xsd:sequence>
      <xsd:element name="Account_spcCompetitors" type="xsd:string"
      <xsd:element name="Account_spcCondition" type="xsd:string"
      <xsd:element name="Account_spcMarkets" type="xsd:string"
      <xsd:element name="Account_spcOrganization_spcIntegratio
      <xsd:element name="Account_spcProducts" type="xsd:string"

```

2.6.4 特定の Siebel オブジェクトの検索

アプリケーション・エクスプローラの検索機能を使用すると、Siebel オブジェクトまたはノードをすばやく検索できます。

1. アプリケーション・エクスプローラを起動し、ターゲットを介して Siebel システムに接続します。
2. ターゲットを展開し、「ビジネス・オブジェクト」、「ビジネス・サービス」または「統合オブジェクト」を選択します。
3. 右ペインで、カーソルを「操作」の上に置いて「検索」を選択します。
4. 検索するノードまたはオブジェクトの名前(**Account**など)をテキスト入力ボックスに入力します。
5. 「OK」をクリックします。

検索に一致する Siebel 項目を含むリストが表示されます。

6. 該当する項目を選択します。

アプリケーション・エクスプローラによって該当する項目が検索されます。

2.6.5 指定した順序でのフィールドの戻し

XML スキーマからリクエスト・ドキュメントを作成して Siebel システムを問い合わせる場合、想定されるレスポンスを、問合せで指定した特定のフィールドに制限できません。レスポンスには、指定した順序でフィールドが含まれます。フィールドのセットを指定しない場合、レスポンス・ドキュメントにはセット全体が含まれます。

たとえば、次の問合せはすべてのフィールドを戻します。

```

<m:Siebel location="S/BO/Account/Account/queryWithView" view="AllView">
  <m:select>
    <m:Name>Yelena*</m:Name>
  </m:select>
</m:Siebel>

```

次の問合せは、Name、Location および Account Status の各フィールドのみを含むレスポンスを返します。

```
<m:Siebel location="S/BO/Account/Account/queryWithView" view="AllView">
  <m:select>
    <m>Name>Yelena*</m>Name>
  </m:select>
  <m:field>Name</m:field>
  <m:field>Location</m:field>
  <m:field>Account Status</m:field>
</m:Siebel>
```

2.6.6 QueryWithView の使用

ビジネス・コンポーネントの場合、Oracle Application Adapter for Siebel により挿入、更新、削除および問合せが有効になります。QueryWithView というメソッドも有効になります。表示モードは、Siebel が提供する表示機能です。

QueryWithView を使用すると、Siebel 表示モードをパラメータとして指定できます。API パラメータを使用すると、ユーザーが構成した Siebel 環境に応じて、様々な方法でデータを表示できます。

様々な表示モードに基づいてユーザーがレコードを取得できるようにする場合を除いて、問合せを使用できます。この場合、QueryWithView を使用します。

QueryWithView モードまたは Siebel の表示の概念の詳細は、Siebel 管理者に確認してください。

次のレベルを使用できます。

- 営業担当ビュー
- マネージャ・ビュー
- パーソナル・ビュー
- すべてのビュー
- 組織ビュー
- グループ・ビュー
- カタログ・ビュー
- サブ組織ビュー

2.7 統合オブジェクトの使用に関する Siebel の前提条件

Siebel 統合オブジェクトの XML スキーマを作成する場合、Siebel Tools のスキーマ・ウィザードを使用して、まず XDR スキーマを生成する必要があります。

XDR スキーマは、統合オブジェクトのスキーマを生成する際、アプリケーション・エクスプローラへの入力として使用されます。XDR スキーマの生成後、アプリケーション・エクスプローラは XDR ファイルを使用して XML スキーマを生成します。

以下の点に注意してください。

- **Siebel 7.5 以降の場合** : XSD スキーマを Siebel Tools から直接生成します。これらの XSD スキーマを使用して、アプリケーション・エクスプローラで Web サービスを直接作成します。Siebel Tools を使用して XSD スキーマを生成した後、これを使用して IO ノードおよび Web サービスを作成します。
- **Siebel 7.0 の場合** : Siebel Tools から XSD スキーマを直接生成できません。XDR スキーマのみ作成できます。そのため、Web サービスを作成するには、アプリケーション・エクスプローラでまず XDR スキーマから XSD スキーマを生成する必要があります。

- **Siebel 6.3 より前**のリリースの場合: Siebel Tools のスキーマ・ウィザードは DTD スキーマのみ作成します。アプリケーション・エクスプローラで XML スキーマを作成するための入力としてこれらを使用するには、手動または他のツールを使用してこれらのスキーマを XDR ファイルに変換する必要があります。さらに、SiebelMessage タグ参照を XDR ファイルに含める必要があります。

Oracle Application Adapter for Siebel では、Siebel XML を使用してイベントを処理することによる、Siebel 統合オブジェクトへのアクセスがサポートされます。サポートされているトランスポートを介して Siebel 統合オブジェクトを使用するには、Siebel ワークフローが必要です。

2.8 Siebel 統合オブジェクト用のスキーマの作成

この項では、Siebel 統合オブジェクト用のスキーマの作成方法について説明します。内容は次のとおりです。

- 2.8.1 項「Siebel 統合オブジェクト用の Siebel XDR または XSD スキーマの作成」

2.8.1 Siebel 統合オブジェクト用の Siebel XDR または XSD スキーマの作成

Siebel XDR または XSD スキーマを生成する手順

1. [図 2-22](#) に示すように、Siebel Tools にログオンします。

図 2-22 Siebel Tools メニュー

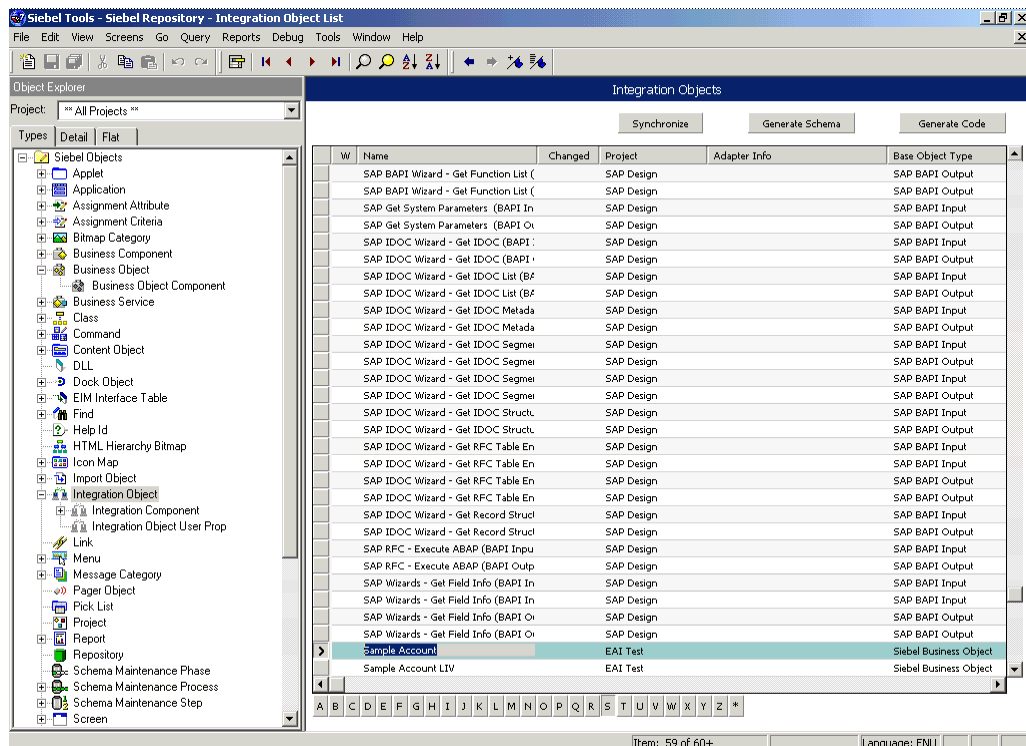


次のステップを実行します。

- a. ユーザー ID とパスワードを入力します。
 - b. リストからデータベースを選択します。
2. 「OK」をクリックします。

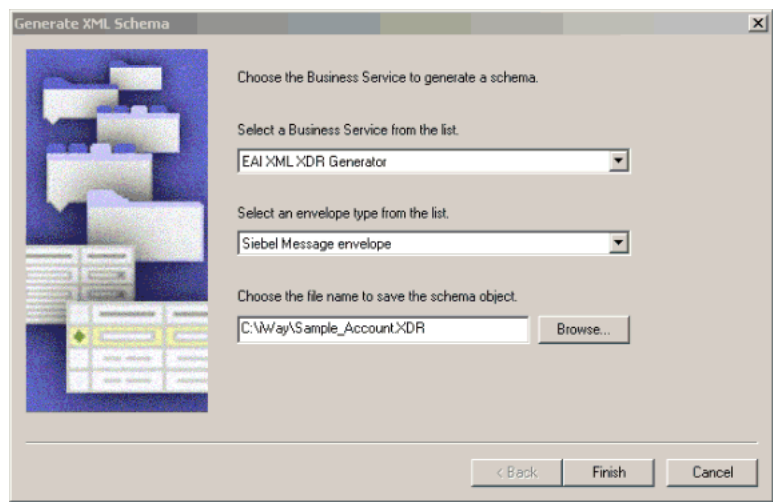
[図 2-23](#) に示すように、Siebel Tools ウィンドウが表示されます。右ペインに統合オブジェクトが表示されます。

図 2-23 Siebel Tools ウィンドウ



3. スキーマを作成するには、統合オブジェクト (Sample Account など) を選択します。
4. 「スキーマ」をクリックします。
 図 2-24 に示すように、「XML スキーマ」ウィザードが表示されます。

図 2-24 「XML スキーマ」ウィザード



次のステップを実行します。

- a. ビジネス・サービスの選択リストから、**EAI XML XDR Generator**(XDR スキーマの場合) または **EAI XML XSD Generator**(XSD スキーマの場合) を選択します (Siebel 7.5 以上の場合)。

- b. エンベロープ・タイプの選択リストから、**Siebel メッセージ・エンベロープ** を選択します。
- c. ファイル名の選択フィールドに、XDR スキーマのファイル名とアプリケーション・エクスプローラでアクセスできるディレクトリを指定します。

注意： XDR または XSD スキーマ・ファイルは、アプリケーション・エクスプローラと同じコンピュータ上のディレクトリに保存する必要があります。

5. 「終了」をクリックします。
6. HTTP を介して着信 XML ドキュメントを受け入れ、EAI XML Converter および EAI Siebel Adapter ビジネス・サービスを使用して Siebel データを挿入 / 更新するワークフローを作成します。

詳細は、付録 A 「[Siebel ワークフローの使用](#)」を参照してください。

7. 次のディレクトリにある `eai.cfg` ファイルを編集します。

```
<siebel_server>/bin/enu
```

8. [HTTP Services] セクションに次の行を追加します。

```
[HTTP Services]
wf = iWayWorkflow
```

9. `Eapps.cfg` ファイルの [EAI_ENU] セクションで次の行が設定されていることを確認します。

```
[EAI_ENU]
EnableExtServiceOnly = True
```

`Eapps.cfg` ファイルは次のディレクトリにあります。

```
<siebel_server>/bin
```

10. 次のコマンドを実行して Siebel Server Manager を使用する名前付きサブシステムを作成します。ここで、EAITEST はステップ 6 で作成されたワークフローの名前です。

```
create named subsystem iWayWorkflow for subsystem
EAITransportDataHandlingSubsys with DispatchWorkflowProcess="EAITEST"
```

これで、アプリケーション・エクスプローラを使用して統合オブジェクト (IO) ノードを作成できます。

2.9 Siebel の統合オブジェクト (IO) ノードの作成

この項では、次のトピックについて説明します。

- [2.9.1 項「Siebel 統合オブジェクト用の XML スキーマの作成」](#)

Siebel の統合オブジェクトノードを作成するには、次の手順を実行します。

1. アプリケーション・エクスプローラで、定義済のターゲットに接続します。ターゲットへの接続方法の詳細は、2-9 ページ「[定義済のターゲットへの接続](#)」を参照してください。

[図 2-25](#) に示すように、アイコン上の「x」が消え、ノード・ターゲットが接続されたことが示されます。

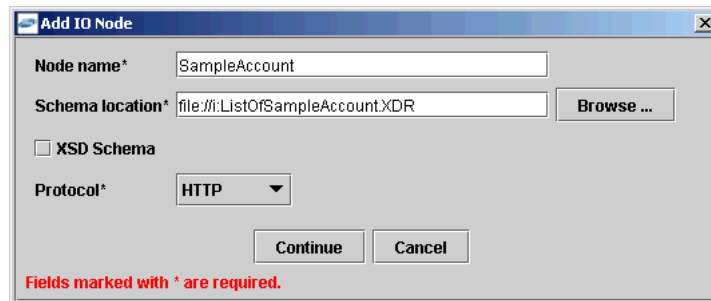
図 2-25 展開された「Siebel」ノード



2. 「統合オブジェクト」ノードを展開して「Sample Account」を選択します。
3. 「Sample Account」ノードを右クリックして IO ノードの追加を選択します。

図 2-26 に示すように、IO ノードの追加ダイアログが表示されます。

図 2-26 IO ノードの追加ダイアログ



以下の点に注意してください。

- **Siebel 7.5 以降の場合** : XSD スキーマを Siebel Tools から直接生成します。アプリケーション・エクスプローラで Web サービスを作成する際、XSD スキーマを使用します。Siebel Tools を使用して XSD スキーマを生成したら、これを使用して IO ノードおよび Web サービスを作成します。
- **Siebel 7.0 の場合** : Siebel Tools から XSD スキーマを直接生成できません。XDR スキーマのみ作成できます。Web サービスを作成する前に、まずアプリケーション・エクスプローラを使用して XDR スキーマから XSD スキーマを生成する必要があります。

注意 : これは、2-17 ページ「Siebel 統合オブジェクト用のスキーマの作成」で生成したスキーマ・ファイルです。

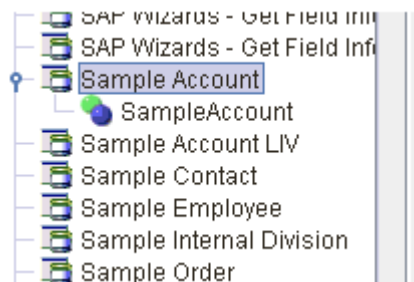
4. 「ノード名」フィールドにノード名 (SampleAccount など) を入力し、「スキーマの場所」フィールドに Sample Account の XDR または XSD ファイルへのパスを入力します。
5. XSD スキーマがすでに生成されている場合、「XSD スキーマ」を選択します。Siebel で生成された XDR スキーマを使用する場合、「XSD スキーマ」オプションを選択しないでください。
6. 「プロトコル」リストからプロトコルを選択します。
7. 「続行」をクリックします。

図 2-27 に示すように、IO ノードの追加ダイアログが表示されます。

図 2-27 IO ノードの追加ダイアログ

8. 次のステップを実行します。
 - a. SWE URL フィールドに、ベース SWE URL を入力します。次に例を示します。
`http://web_server/eai/start.swe`
web_server は、Siebel SWE をホストする Web サーバーの名前です。
 - b. SWE 外部ソース・フィールドに、実行する eai.cfg ファイル内のセクション ([HTTP Services] セクション) を入力します。
 詳細は、2-17 ページ「[Siebel 統合オブジェクト用のスキーマの作成](#)」のステップ 8 を参照してください。
 - c. SWE 外部コマンド・フィールドに、示されているとおりに次のコマンドを入力します。
`é¿çs`
 - d. 「ユーザー名」のフィールドに、Siebel SWE への接続に使用される有効なユーザー名とパスワードを入力します。
 ユーザー名とパスワードは、指定されたワークフローの実行権限を持っている必要があります。
9. 「終了」をクリックします。
 図 2-28 に示すように、新しい IO ノードが統合オブジェクトの「Sample Account」ノードの下にリストされます。

図 2-28 統合オブジェクトの「Sample Account」ノード



これで、XML スキーマを作成できます。

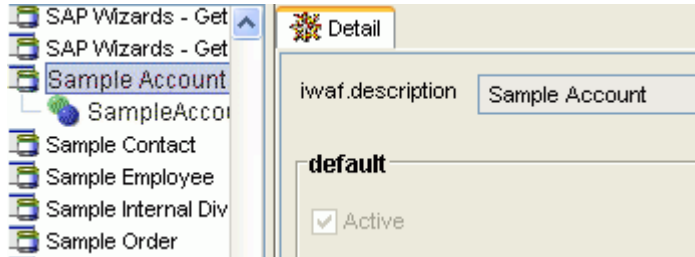
2.9.1 Siebel 統合オブジェクト用の XML スキーマの作成

Siebel 用の統合オブジェクト・ノードを作成した後、アプリケーション・エクスプローラを使用して XML スキーマを作成できます。

XML スキーマを作成する手順

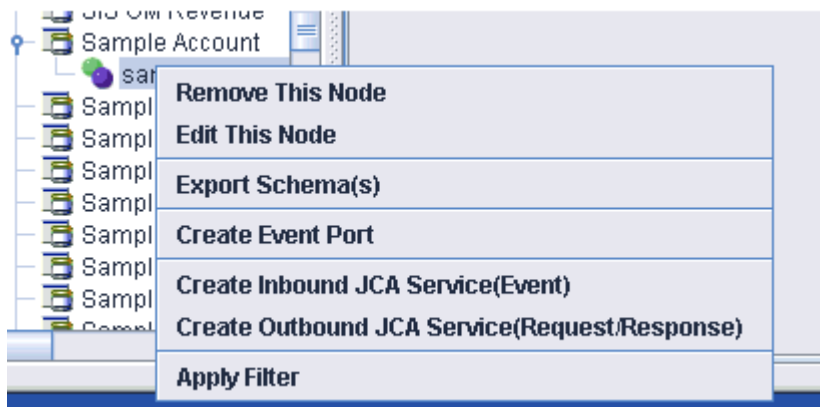
1. 図 2-29 に示すように、アプリケーション・エクスプローラで「統合オブジェクト」ノードを展開し、Siebel システムの統合オブジェクトを参照します。

図 2-29 Siebel 統合オブジェクト・ノード「Sample Account」



2. 下にスクロールし、統合オブジェクト (SampleAccount など) を選択します。
3. 図 2-30 に示すように、作成された統合オブジェクト・ノード (SampleAccount など) を右クリックし、メニューから「スキーマのエクスポート」を選択します。

図 2-30 「スキーマのエクスポート」メニュー・オプション



「エクスポート・ディレクトリの選択」ダイアログが表示されます。

チャンネルの作成処理時、プリパーサー・タブ (スキーマの場所フィールド) で、エクスポートされたイベント・スキーマを指定する必要があります。

4. 「OK」をクリックして、スキーマを保存します。

2.10 Siebel ビジネス・サービス用のサービス・ノードの作成

OracleAS Adapter for Siebel では、階層データ型のメソッド引数を持つメソッドを含むビジネス・サービス用のサービス・ノードを追加できます。

重要な制限:

- アダプタでサポートされるのは、統合オブジェクト階層データ型のみです。
- サービス・ノードを追加するには、統合オブジェクトの XSD スキーマを事前に生成しておく必要があります。Siebel 統合オブジェクト用の XSD スキーマ生成の詳細は、2-17 ページ「Siebel 統合オブジェクト用のスキーマの作成」を参照してください。

- 階層データ型にできるのは、サービス・ノードの追加対象のビジネス・サービス・メソッドのメソッド引数の1つのみです。
- メソッド引数 XMLCharEncoding はサポートされていません。XML ペイロードでこの要素を空白にします。有効な XMLCharEncoding 値 (UTF-8、UTF-16 など) を入力すると、次のエラーが表示されます。

サービスの呼び出しに失敗しました。

サービスを作成する手順

1. 該当するビジネス・サービス・ノードを選択します。
2. サービスを作成するビジネス・サービス・メソッド引数を右クリックし、**サービス・ノードの追加**を選択します。

図 2-31 に示すように、サービス・ノードの追加ダイアログが表示されます。

図 2-31 サービス・ノードの追加ダイアログ

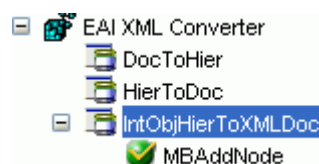
3. 次のステップを実行します。
 - a. サービス・ノード名を指定します。
 - b. 説明を入力します (オプション)。
 - c. XSD スキーマ・ファイルへのフルパス (ファイル名を含む) を指定します。
 - d. XSD スキーマ・ファイルのルート要素を指定します。統合オブジェクトの多くの XSD スキーマでは、ルート要素は SiebelMessage です。
 - e. XSD スキーマが統合オブジェクト用かどうかを指定します。

重要: このチェック・ボックスが選択されていることを確認する必要があります。

4. 「OK」をクリックします。

図 2-32 に示すように、サービス・ノードがビジネス・サービス・オブジェクトの下に表示されます。

図 2-32 ビジネス・サービス・オブジェクトの下にリストされたサービス・ノード



このノードを右クリックすると、Web サービスを作成できます。リクエスト・スキーマおよびレスポンス・スキーマは、右ペインに表示されます。

次の手順は、ビジネス・オブジェクト用の Web サービスの作成方法を説明しています。

2.11 Web サービスの作成およびテスト (BSE 構成のみ)

サブレット BSE を適切に構成したら、アダプタで使用する Siebel オブジェクト用の **ビジネス・サービス** (Web サービスとも呼ばれる) を生成できます。

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。Oracle Adapter J2CA を使用するようアダプタが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。

この項では、次のトピックについて説明します。

- 2.11.1 項「Web サービスの作成」
- 2.11.2 項「Web サービスのテスト」

2.11.1 Web サービスの作成

Siebel ビジネス・オブジェクト用の Web サービスを生成する手順

1. Siebel システムに接続します。
2. 「ビジネス・オブジェクト」ノードを展開します。
3. 図 2-33 に示すように、Web サービスを作成する「ビジネス・コンポーネント」を展開します。

図 2-33 queryWithView メソッドを持つ「Account」ビジネス・オブジェクト



4. オブジェクトを展開し、Web サービスを作成するメソッド(「Account」の下の「QueryWithView」など)を選択します。
5. ビジネス・サービスの作成元となるノードを右クリックし、「ビジネス・サービス」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。

新規の Web サービスのメソッドとして、または既存のもののメソッドとしてビジネス・オブジェクトを追加できます。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はサービス名を指定します。この名前により、「ビジネス・サービス」ノードの下のサービス・リストの Web サービスが識別されます。

- c. サービスの説明を入力します (オプション)。
 - d. 使用可能なライセンスを 1 つ選択します。
6. 「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・ボックスが表示されます。次のステップを実行します。
- a. 「ライセンス」フィールドで、Web サービスに割り当てる 1 つ以上のライセンス・コードを選択します。複数を選択する場合は [Ctrl] キーを押しながら各ライセンスをクリックします。
 - b. 「メソッド名」フィールドでは、デフォルトのメソッド名を選択したままにします。
 - c. 「説明」フィールドに、メソッドの説明を入力します (オプション)。
7. 「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。
8. 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。
9. WSDL ファイルとの名前と WSDL ファイルを保存するファイル・システムの場所を指定します。
10. 「保存」をクリックします。

2.11.2 Web サービスのテスト

Siebel ビジネス・オブジェクトに対して Web サービスを作成した後は、サービスをテストして正常に機能することを確認します。アプリケーション・エクスプローラには Web サービスをテストするためのテスト・ツールが含まれています。

この項では、次のトピックについて説明します。

- [2.11.2.1 項「ビジネス・オブジェクト用の Web サービスのテスト」](#)
- [2.11.2.2 項「ビジネス・サービス用の Web サービスのテスト」](#)
- [2.11.2.3 項「伝播の識別」](#)

2.11.2.1 ビジネス・オブジェクト用の Web サービスのテスト

1. アプリケーション・エクスプローラの左ペインで、「ビジネス・サービス」ノードを展開します。
2. 「サービス」ノードを展開します。
3. [図 2-34](#) に示すように、テストするビジネス・サービスの名前を選択します。

図 2-34 展開された「サービス」ノード



4. サービスの下の「メソッド」ノードを展開し、テストするメソッドを選択します。右ペインにテスト・オプションが表示されます。

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. 「起動」をクリックします。

図 2-35 に示すように、アプリケーション・エクスプローラの「結果」ペインに結果が表示されます。

図 2-35 「結果」ペインの XML の結果

```
<?xml version="1.0" encoding="UTF-8" ?>
- <SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:SOAP-
  ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance">
- <SOAP-ENV:Body>
  - <QueryWithViewResponse
    xmlns="urn:iwaysoftware:ibse:jul2003:QueryWithView"
    cid="638ED68A7082CDA3B0492896446C44D8">
  - <SiebelResponse status="success">
    - <record>
      <Name>SIEBEL1 ACCOUNT</Name>
      <Location>ONE</Location>
    </record>
    - <record>
      <Name>SIEBEL2 ACCOUNT</Name>
      <Location>TWO</Location>
    </record>
    - <record>
      <Name>SIEBEL3</Name>
      <Location>RR</Location>
    </record>
  - <record>
```

2.11.2.2 ビジネス・サービス用の Web サービスのテスト

Siebel ビジネス・サービスに対して Web サービスを作成した後は、サービスをテストして正常に機能することを確認します。アプリケーション・エクスプローラには Web サービスをテストするためのテスト・ツールが含まれています。

1. 「ビジネス・サービス」ノードを展開します。
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。
4. 「メソッド」ノードを展開し、テストするメソッドの名前を選択します。

右ペインにテスト・オプションが表示されます。

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. 適切に入力します。
6. 「起動」をクリックします。

アプリケーション・エクスプローラの「結果」ペインに結果が表示されます。

2.11.2.3 伝播の識別

XMLSPY などのサード・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は、SOAP ヘッダーで指定するユーザー名およびパスワードの値が有効で、Siebel への接続に使用されるものと同じである必要があります。アプリケーション・エクスプローラを使用したターゲットの作成中に Siebel 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 サービスのリクエスト用のもので上書きされます。次は、Web サービス用の WSDL ファイルに含まれるサンプルの SOAP ヘッダーです。

```
<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> タグは必要でないため、SOAP ヘッダーから削除できます。

2.12 WSDL の生成 (J2CA 構成のみ)

Web サービスの Web Service Definition Language (WSDL) 記述により、ホスト・サーバー内の他のサービスでサービスが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 JCA サービスを作成できます。

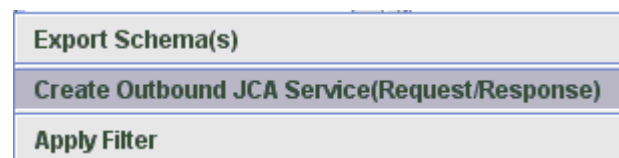
注意： インバウンド JCA サービスの作成 (イベント) オプションは、選択されたノードでイベントがサポートされている場合にのみ使用できます。

リクエスト/レスポンス型サービス用の WSDL ファイルを生成する手順

1. 接続されている Siebel ターゲットの下で、「ビジネス・オブジェクト」、「アカウント」、「アカウント」の順に展開します。オブジェクトに移動し、そのオブジェクトを右クリックします。

図 2-36 に示すように、次のメニューが表示されます。

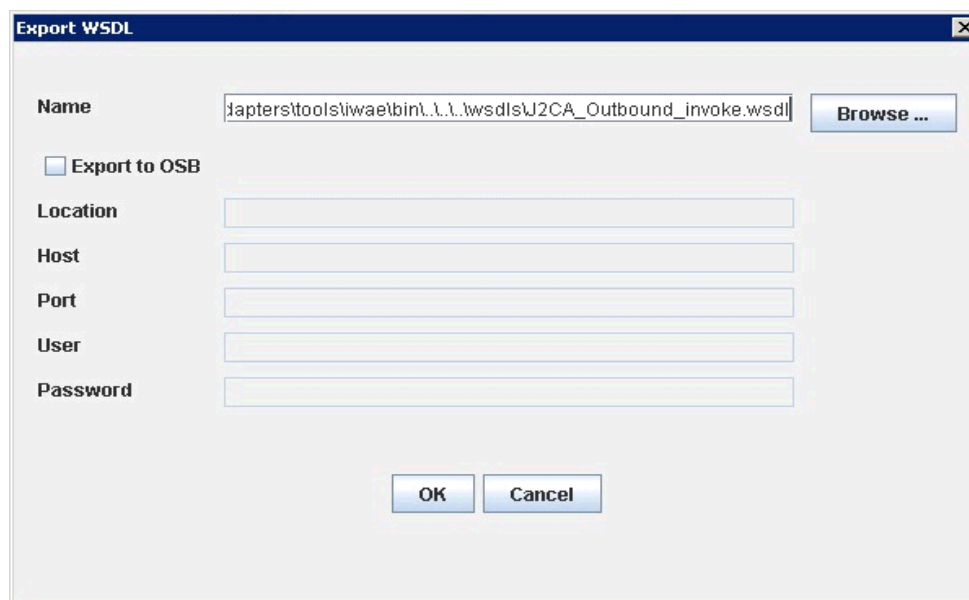
図 2-36 アウトバウンド JCA サービスの作成 (リクエスト/レスポンス) オプション



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 2-37 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 2-37 「WSDL のエクスポート」ダイアログ



3. デフォルトの名前を受け入れるかファイルの名前を指定します (「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/レスポンス型のサービス用に生成された 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たものには「_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/レスポンス用と類似しています。イベント通知用の WSDL を生成するには、先に各イベントのチャンネルを作成する必要があります。

2.13 イベント・アダプタの構成

イベントは、Siebel システムで特定のビジネス条件が満たされるかトリガーされることにより、生成されます。イベントを使用して、アプリケーション内でアクションをトリガーできます。たとえば、顧客情報への更新を反映してデータベースが更新されるようにすることができます。これが起こったときにアプリケーション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です。

この項では、次のトピックについて説明します。

- [2.13.1 項「チャンネルの作成および変更」](#)

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。イベントを構成するには、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は、各イベント・オブジェクトに対して新規のチャンネルを作成し、WSDL を生成するときこのチャンネルを選択する必要があります。チャンネルの作成が必要なのは J2CA 構成の場合のみです。このためには、たとえば、Account と Contact という Siebel オブジェクトを使用する場合、2 つの異なるチャンネルが必要です。

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。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。詳細は、2-29 ページ「[チャンネルの作成および変更](#)」を参照してください。

IO ノードの追加機能は、イベント構成では適用されないことに注意してください。

2.13.1 チャンネルの作成および変更

この項では、次のトピックについて説明します。

- [2.13.1.1 項「HTTP チャンネルの作成」](#)
- [2.13.1.2 項「MQ Series チャンネルの作成」](#)
- [2.13.1.3 項「ファイル・チャンネルの作成」](#)
- [2.13.1.4 項「チャンネルの編集」](#)
- [2.13.1.5 項「チャンネルの削除」](#)

次の手順は、イベントのチャンネルを作成する方法を説明しています。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る必要があります。

チャンネルを作成、変更または削除する場合、ランタイム用に 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。チャンネルとインバウンド WSDL ファイルを正常に作成したら、アプリケーション・エクスペローラを閉じて Oracle WebLogic Server を再起動します。

注意： Oracle Application Adapter for Siebel を BPM、BPEL、メディアータ、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、メディアータ、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・エクスペローラから起動する場合は、ランタイム前に停止してください (BPM、BPEL、メディアータ、または OSB プロセス・コンポーネントと連携する場合)。

次の 3 タイプのチャンネルを使用できます。

- HTTP
- MQ Series
- ファイル

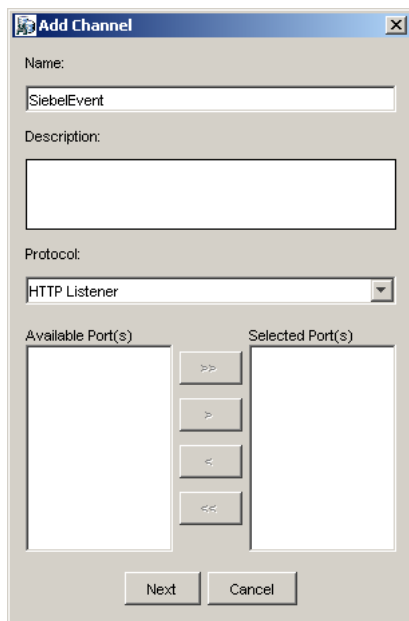
注意： チャンネルは、Oracle Application Adapter for Siebel がインストールされているシステムでのみ構成できます。

2.13.1.1 HTTP チャンネルの作成

HTTP チャンネルを作成する手順

1. 「イベント」ノードをクリックします。
「イベント」ウィンドウが表示されます。左ペインに表示されるアダプタが、イベントをサポートします。
2. 左ペインで「Siebel」ノードを展開します。
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」を右クリックして「チャンネルの追加」を選択します。
図 2-38 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 2-38 「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. チャンネルの名前 (NewChannel など) を入力します。
 - b. 簡単な説明を入力します。
 - c. 「プロトコル」から、「HTTP リスナー」を選択します。
4. 「次へ」をクリックします。
図 2-39 に示すように、HTTP リスナー・ダイアログが表示されます。

図 2-39 HTTP リスナー・ダイアログ

Http Listener

Basic PreParser

Listener port^{*} 8080

Https

Synchronization Type REQUEST

Encoding Type ASCII

OK Cancel

Fields marked with ^{*} are required.

5. 次の表で指定されているようにシステム情報を入力します。

パラメータ	説明
リスナー・ポート	Siebel イベント・データをリスニングするポート。
Https	セキュアな HTTP 接続の場合は Https チェック・ボックスを選択します。このオプションは現在サポートされていません。
同期タイプ	「同期タイプ」リストから REQUEST_RESPONSE (推奨オプション)を選択します。
エンコーディング・タイプ	使用するエンコーディング・タイプをエンコーディング・タイプ・リストから選択します。デフォルトでは、「ASCII」が選択されています。

6. 図 2-40 に示すように、プリパーサー・タブをクリックします。

図 2-40 プリパーサー・タブ

Http Listener

Basic PreParser

Schema location^{*}

OK Cancel

Fields marked with ^{*} are required.

7. アプリケーション・エクスプローラの「スキーマのエクスポート」オプションを使用して統合オブジェクト・ノード用に生成されたスキーマ・ファイルの場所を指定します。

注意： 実行時、Oracle Application Adapter for Siebel は、プリパーサー・タブで指定されたスキーマを使用して Siebel に公開されたドキュメントにネームスペースを追加します。プリパーサー・タブの「スキーマの場所」フィールドが空の場合、Siebel に公開されたドキュメントにネームスペースが含まれないため、BPEL、BPM、OSB、およびメディエータ・プロセスが適切に機能しません。

8. 「OK」をクリックします。

チャンネルの説明、ステータスおよび使用可能なポートが記載されているサマリーが表示されます。これらの情報はすべて作成したチャンネルと関連付けられています。図 2-41 に示すように、チャンネルは左ペインの「チャンネル」ノードの下にも表示されます。

図 2-41 非アクティブな「SiebelHTTP」ノード



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起動する必要があります。

9. チャンネルを右クリックし、「起動」を選択します。

作成したチャンネルがアクティブになります。左ペインでアイコン上に表示されていた「X」が消えます。

10. チャンネルを停止するには、チャンネルを右クリックし、「停止」を選択します。

2.13.1.2 MQ Series チャンネルの作成

MQ Series チャンネルを作成する手順

1. 「イベント」ノードをクリックします。

「イベント」ウィンドウが表示されます。左ペインに表示されるアダプタが、イベントをサポートします。

2. 左ペインで「Siebel」ノードを展開します。

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」ノードを右クリックして「チャンネルの追加」を選択します。

新規チャンネルの追加ペインが表示されます。次のステップを実行します。

- a. チャンネルの名前 (NewChannel など) を入力します。
- b. 簡単な説明を入力します。
- c. 「プロトコル」から、MQ Series リスナーを選択します。

4. 「次へ」をクリックします。

図 2-42 に示すように、MQ Series リスナー・ダイアログが表示されます。

図 2-42 MQ Series リスナー・ダイアログ

The image shows the 'MQ Listener' dialog box with three tabs: 'Request', 'Response', and 'Advanced'. The 'Request' tab is selected. It contains the following fields:

- Queue manager name* (text input)
- MQ server host for MQClient operation* (text input)
- MQ server port for MQClient operation* (text input)
- MQ server channel for MQClient operation* (text input)
- Document type XML
- Request queue name* (text input)

At the bottom, there are 'OK' and 'Cancel' buttons. A red text note at the very bottom reads: 'Fields marked with * are required.'

5. 次の手順に示すようにシステム情報を入力します。
 - a. 「リクエスト」タブで、次のパラメータに値を入力します。

パラメータ	説明
キュー・マネージャ名	MQ サーバーが配置されているホスト (MQ クライアントのみ)。
MQClient 運用のための MQ サーバー・ホスト	ホスト・データベースがリスニングしているポート。
MQClient 運用のための MQ サーバー・ポート	MQ サーバーのキュー・マネージャに接続する番号 (MQ クライアントのみ)。 REQUEST REQUEST_RESPONSE REQUEST_ACK
MQClient 運用のための MQ サーバー・チャンネル	リモートの MQ サーバー・キュー・マネージャに接続するチャンネルの大/小文字を区別した名前 (MQ クライアントのみ)。MQSeries のデフォルト・チャンネル名は、SYSTEM.DEF.SVRCONN です。
ドキュメント・タイプ XML	デフォルト選択のままにします。

パラメータ	説明
リクエスト・キュー名	<p>メッセージがルーティングされ、リクエスト・ドキュメントが受け取られるキュー。キューの名前は大/小文字が区別され、次の書式に従います。</p> <p>Host\queue type\$\qName</p> <p>Host</p> <p>MQ Series のキューイング・システムが実行されているシステムの名前です。</p> <p>queue type</p> <p>プライベート・キューは Active Directory に公開されていないキューで、配置されているローカル・コンピュータ上のみ表示されます。プライベート・キューにアクセスできるのは、キューのフルパス名または書式名を認識するメッセージ・キューイング・アプリケーションのみです。</p> <p>qName</p> <p>メッセージが配置されるキューの名前です。例</p> <p>iwaykxcl\Private\$\siebel</p>

b. 「レスポンス」タブで、次のパラメータに値を入力します。

パラメータ	定義
同期タイプ	「同期タイプ」リストから REQUEST_RESPONSE (推奨オプション)を選択します。

c. 「詳細」タブで、次のパラメータに値を入力します。

パラメータ	定義
メッセージ待機間隔 (msec)	新規入力をチェックする間隔 (ミリ秒)。デフォルトは3秒です。オプション。
操作モード	<p>「順次を選択します。</p> <ul style="list-style-type: none"> ■ 「順次」は、リクエストを1つずつ処理することを示します。 ■ スレッドは、複数のリクエストを同時に処理することを示します。
スレッド制限	スレッド処理を選択した場合、同時に処理可能なリクエストの最大数を示します。

6. 「OK」をクリックします。

チャンネルの説明、ステータスおよび使用可能なポートが記載されているサマリーが表示されます。これらの情報はすべて作成したチャンネルと関連付けられています。チャンネルは左ペインの「チャンネル」ノードの下にも表示されます。

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。イベント構成をアクティブ化するにはチャンネルを再起動する必要があります。

7. チャンネルを右クリックし、「開始」を選択します。

作成したチャンネルがアクティブになります。左ペインでアイコン上に表示されていた「X」が消えます。

8. チャンネルを停止するには、チャンネルを右クリックし、「停止」を選択します。

2.13.1.3 ファイル・チャネルの作成

ファイル・チャネルを作成する手順

1. 「イベント」ノードをクリックします。

「イベント」ウィンドウが表示されます。左ペインに表示されるアダプタが、イベントをサポートします。

2. 左ペインで「Siebel」ノードを展開します。

ポートおよびチャネル・ノードが表示されます。

3. 「チャネル」ノードを右クリックして「チャネルの追加」を選択します。

「チャネルの追加」ダイアログが表示されます。次のステップを実行します。

- a. チャネルの名前 (NewChannel など) を入力します。
- b. 簡単な説明を入力します。
- c. 「プロトコル」から、ファイル・リスナーを選択します。

4. 「次へ」をクリックします。

図 2-43 に示すように、ファイル・リスナー・ダイアログが表示されます。

図 2-43 ファイル・リスナー・ダイアログ

5. 次の手順に示すようにシステム情報を入力します。

- a. 「リクエスト」タブで、次のパラメータに値を入力します。

パラメータ	説明
ポーリング場所	Siebel XML ファイル用のターゲット・システムの場所。
ファイル・マスク	この操作で生成される出力ファイルに使用されるファイル名。

- b. 「レスポンス」タブで、次のパラメータに値を入力します。

パラメータ	定義
同期タイプ	「同期タイプ」リストからから REQUEST_RESPONSE (推奨オプション)を選択します。
レスポンス/Ack ディレクトリ	レスポンスまたは Ack が送信されるディレクトリ。

- c. 「詳細」タブで、次のパラメータに値を入力します。

パラメータ	定義
エラー・ディレクトリ	エラーのあるドキュメントが書き込まれるディレクトリ
ポーリング間隔	新規入力をチェックする間隔 (ミリ秒)。デフォルトは3秒です。オプション。
処理モード	「順次」を選択します。 <ul style="list-style-type: none"> ■ 「順次」は、リクエストを1つずつ処理することを示します。 ■ スレッドは、複数のリクエストを同時に処理することを示します。
スレッド制限	スレッド処理を選択した場合、同時に処理可能なリクエストの最大数を示します。

6. 「OK」をクリックします。

チャンネルの説明とステータスが記載されているサマリーが表示されます。これらの情報はすべて作成したチャンネルと関連付けられています。チャンネルは左ペインの「チャンネル」ノードの下にも表示されます。

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。イベント構成をアクティブ化するにはチャンネルを起動する必要があります。

7. チャンネルを右クリックし、「開始」を選択します。

作成したチャンネルがアクティブになります。

左ペインでアイコン上に表示されていた「X」が消えます。

8. チャンネルを停止するには、チャンネルを右クリックし、「停止」を選択します。

2.13.1.4 チャンネルの編集

チャンネルを編集する手順

1. 左ペインで、編集するチャンネルを選択します。
2. チャンネルを右クリックし、「編集」を選択します。
「チャンネルの編集」ダイアログが表示されます。
3. チャンネル構成に必要な変更を行い、「OK」をクリックします。

2.13.1.5 チャンネルの削除

チャンネルを削除する手順

1. 左ペインで、削除するチャンネルを選択します。
2. チャンネルを右クリックし、「削除」を選択します。
チャンネルが左ペインから消えます。

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 Siebel との統合について説明します。内容は次のとおりです。

- 3.1 項「Oracle WebLogic Server とのアダプタの統合」
- 3.2 項「アダプタのデプロイメント」
- 3.3 項「アダプタ構成の更新」

関連項目：

- *Oracle Application Server Adapter* コンセプト・ガイド

3.1 Oracle WebLogic Server とのアダプタの統合

Oracle Application Adapter for Siebel は、インストール中に 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 環境で動作します。J2CA デプロイメントでは、Common Client Interface (CCI) により、OracleWLS クライアント・アプリケーションはリソース・アダプタと統合されます。

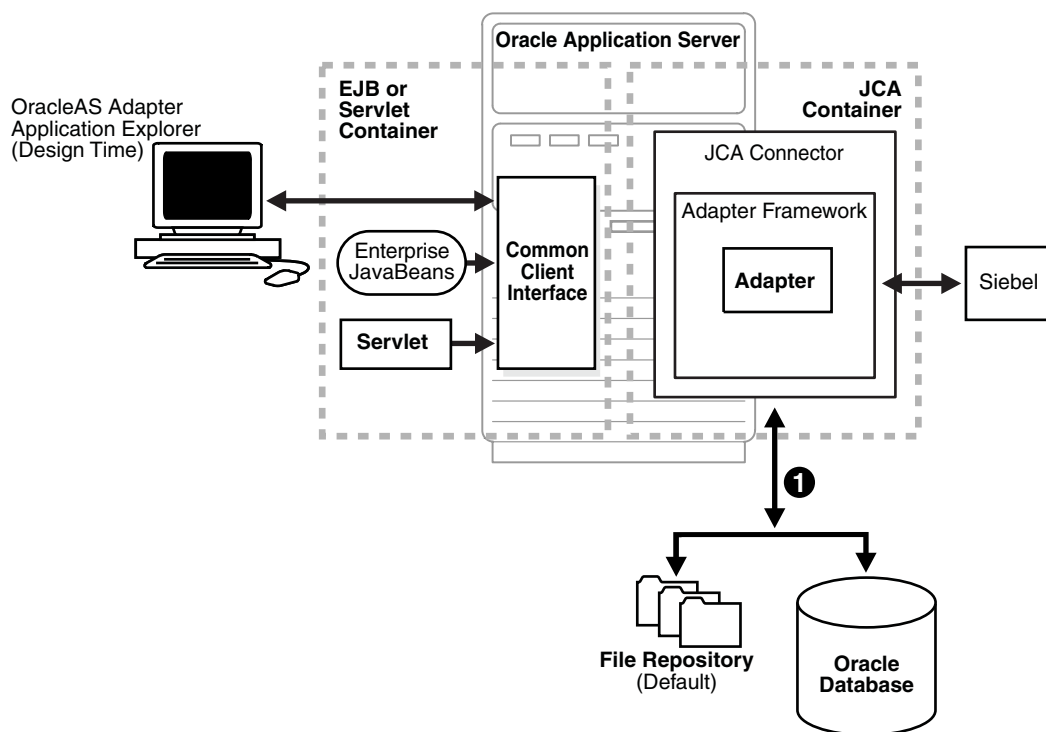
関連項目：

- *Oracle Application Server Adapter* コンセプト・ガイド

3.2 アダプタのデプロイメント

図 3-1 に、Oracle WebLogic Server へのコネクタのデプロイメントを示します。ランタイム・サービス・シナリオでは、Enterprise Java Bean、サーブレットまたは Java プログラム・クライアントによって、J2CA リソース・アダプタへの CCI コールが行われます。アダプタはこれらのコールをリクエストとして処理し、EIS に送信します。その後、EIS のレスポンスがクライアントに返信されます。

図 3-1 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 コンセプト・ガイド](#)

3.3 アダプタ構成の更新

この項では、次のトピックについて説明します。

- [3.3.1 項「管理対象コネクタ・ファクトリ・オブジェクトの作成」](#)
- [3.3.2 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」](#)
- [3.3.3 項「追加の接続ファクトリ値用の WSDL ファイルの変更」](#)

OracleAS Adapter for Siebel の J2CA デプロイメント中、OracleWLS により、次の場所に `ra.xml` と呼ばれるデプロイメント記述子が生成されます。

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

インストールには、`ra.xml` と呼ばれるファイルが 1 つ以上含まれています。この項で説明されている OracleWLS デプロイメント記述子は、前述のディレクトリ内にあります。

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

3.3.1 管理対象コネクタ・ファクトリ・オブジェクトの作成

ra.xml 記述子には、リソース・アダプタに関する OracleWLS に固有のデプロイメント情報が記載されます。たとえば、アプリケーション・エクスプローラ内のデフォルトの jca_sample 構成は、ra.xml ファイル内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>

```

ra.xml ファイルで定義されているパラメータを次の表に示します。

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 Siebel には、アプリケーション・エクスプローラで jca_sample 構成が事前定義されています。
IWayRepoURL	データベースへの接続を開くときに使用される URL。これは Oracle 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されるユーザー名。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。指定された場合は、構成を上書きします。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	ManagedConnectorFactory プロパティによって設定されるレベルを上書きします。

3.3.2 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、さらに <connection-instance> ノードを追加します。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現されます。

```
<?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。必要に応じ、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 ファイルの値をオーバーライドできます。また、J2CA 構成 (jca_sample2 など) が、アプリケーション・エクスペローラ内で作成済である必要があります。

注意： ra.xml ファイルと weblogic-ra.xml ファイルを変更した場合、Oracle WebLogic Server を再起動する必要があります。Oracle WebLogic Server がすでに稼働している場合、Oracle WebLogic Server を停止し、再起動します。

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

3.3.3 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、追加の構成および接続ファクトリ用に生成された WSDL を変更する必要があります。isdsrv2_conn2 構成が含まれている Oracle Application Adapter for Siebel 用のデフォルトの J2CA プロパティ・ファイルを次の例に示します。

J2CA プロパティ・ファイルに、次のデフォルトの接続ファクトリが含まれていることがわかります。eis/OracleJCAAdapter/DefaultConnection

```

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

```

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

次に例を示します。

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

`jca:address` セクションの `location` フィールドの値のみ変更可能です。その他のフィールドまたは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 Siebel では、Process Execution Language (BPEL) Process Manager がシームレスに統合されており、Web サービスの簡易な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。

この章の内容は次のとおりです。

- [4.1 項「概要」](#)
- [4.2 項「アダプタのデプロイメント」](#)
- [4.3 項「新規のアプリケーション・サーバー接続の構成」](#)
- [4.4 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 \(J2CA 構成\)」](#)
- [4.5 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 \(J2CA 構成\)」](#)
- [4.6 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 \(BSE 構成\)」](#)

4.1 概要

Oracle BPEL Process Manager を統合するには、Oracle Application Adapter for Siebel が Oracle BPEL Process Manager と同じ WLS コンテナにデプロイされている必要があります。基になるアダプタ・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) で設計時に生成される WSDL ファイルとして公開される必要があります。詳細は、2-27 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・サービスに適した 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、Oracle 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。

4.2 アダプタのデプロイメント

インストール中、Oracle Application Adapter for Siebel 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。

4.3 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には次を行います。

1. システムで **Oracle JDeveloper** を開きます。
2. 図 4-1 に示すように、メニュー・バーで、「ウィンドウ」をクリックして「アプリケーション・サーバー・ナビゲータ」を選択します。

図 4-1 アプリケーション・サーバー・ナビゲータ

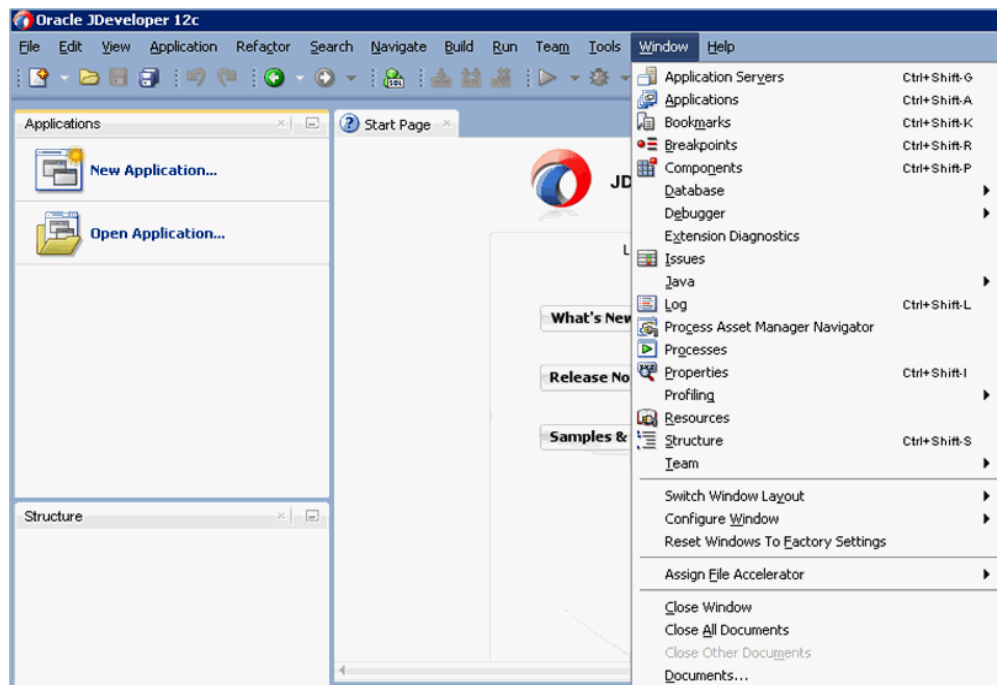
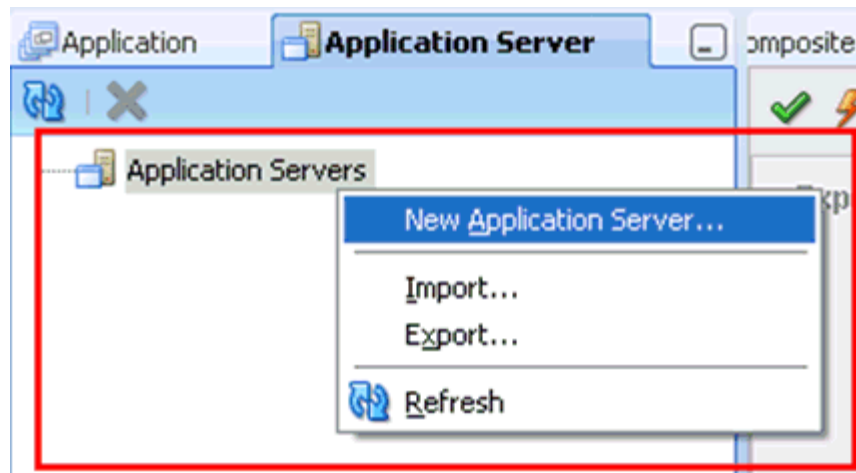


図 4-2 に示すように、「アプリケーション・サーバー」タブが表示されます。

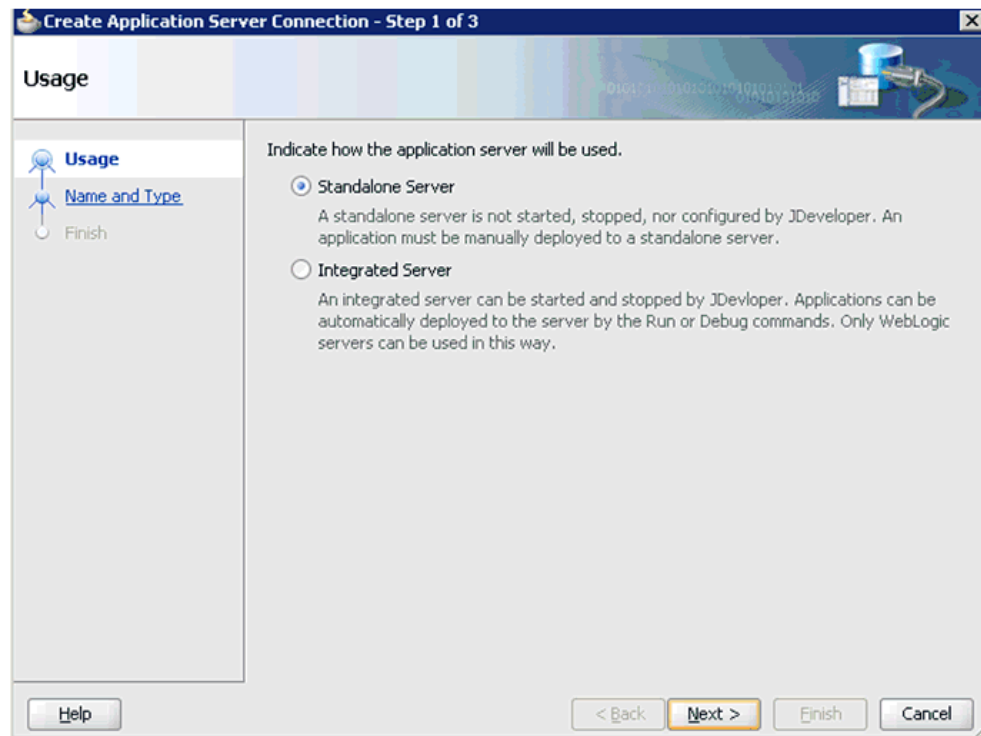
図 4-2 「アプリケーション・サーバー」タブ



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

図 4-3 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。

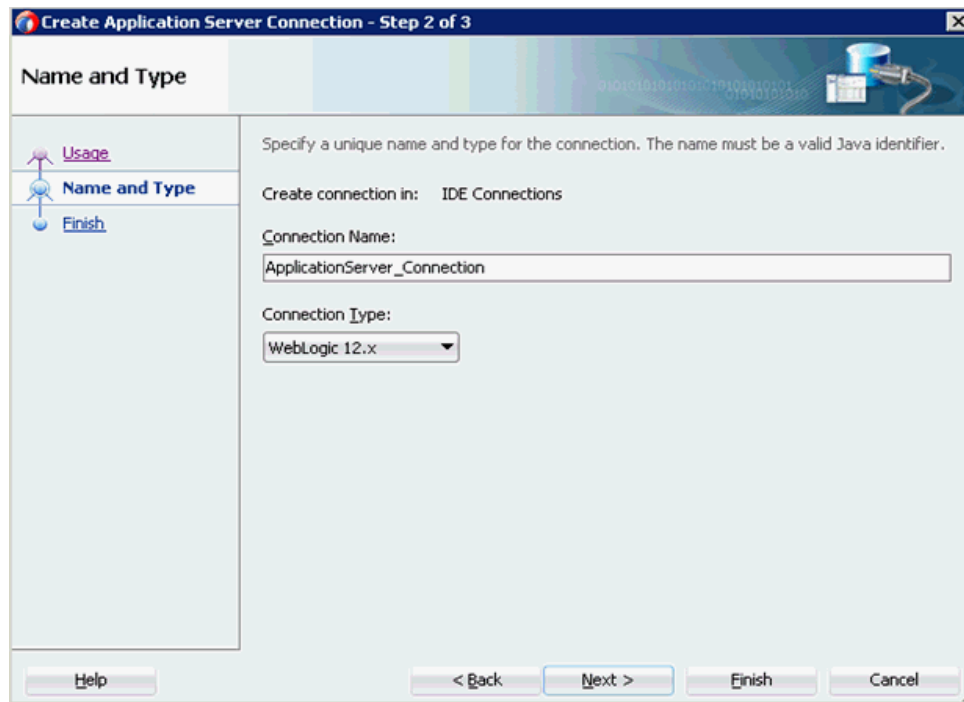
図 4-3 アプリケーション・サーバー接続の作成ウィザード



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 4-4 に示すように、「名前とタイプ」ページが表示されます。

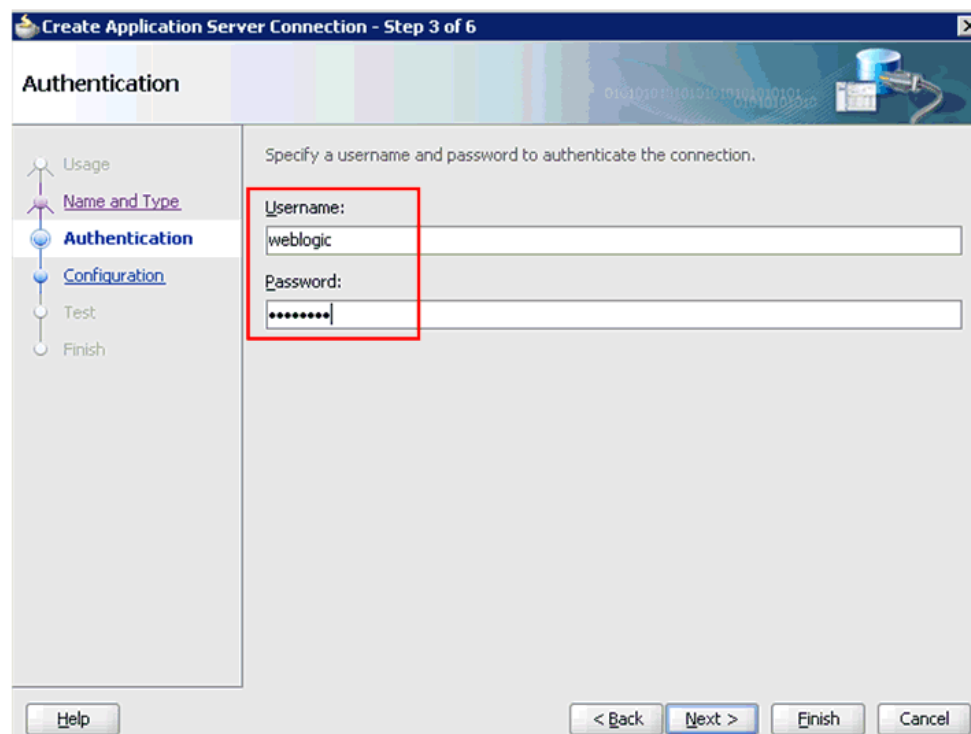
図 4-4 「名前とタイプ」 ページ



5. アプリケーション・サーバー接続の新しい名前を指定し、「次へ」をクリックします。

図 4-5 に示すように、「認証」 ページが表示されます。

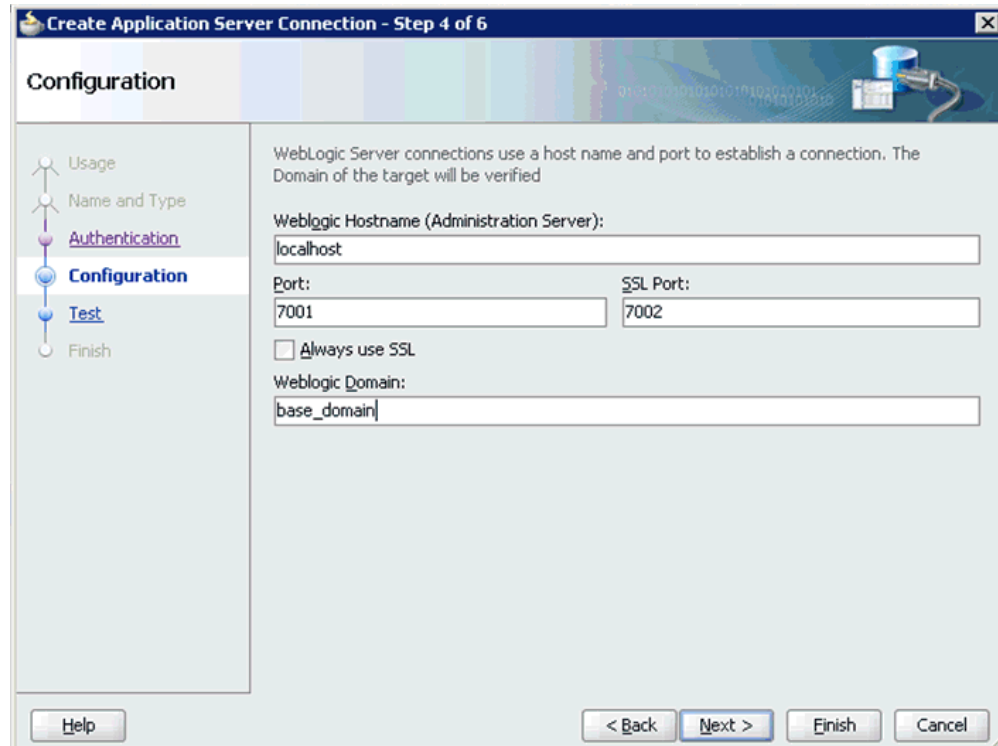
図 4-5 「認証」 ページ



6. 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。
7. 「次へ」をクリックします。

図 4-6 に示すように、「構成」ページが表示されます。

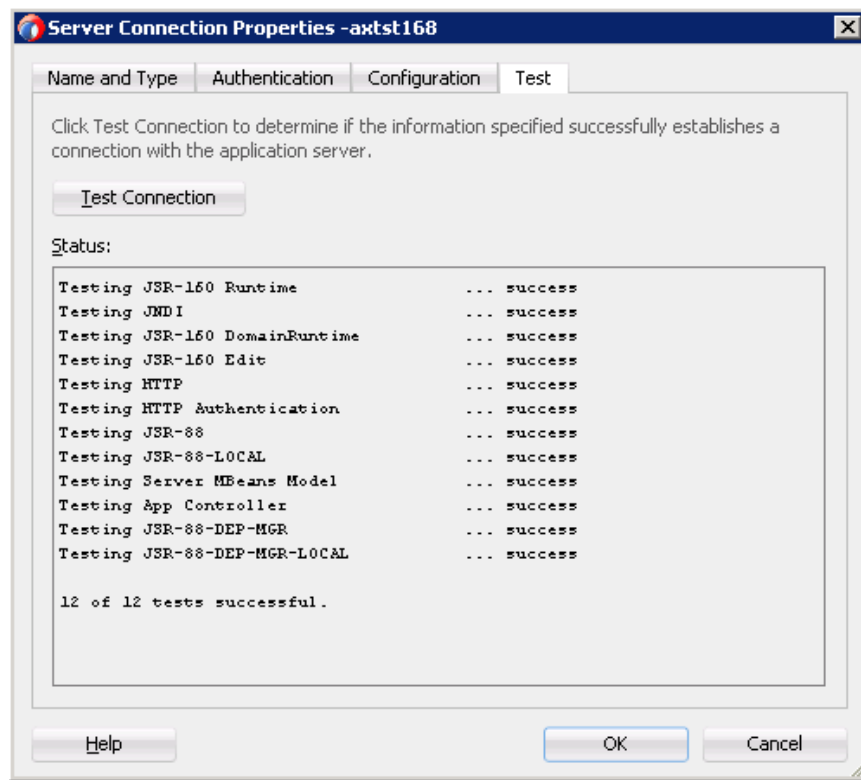
図 4-6 「構成」ページ



8. プロセスのデプロイ先のシステムの IP である Oracle WebLogic のホスト名 (「localhost」など) と、Oracle WebLogic のドメイン (「base_domain」など) を指定します。
9. 「次へ」をクリックします。

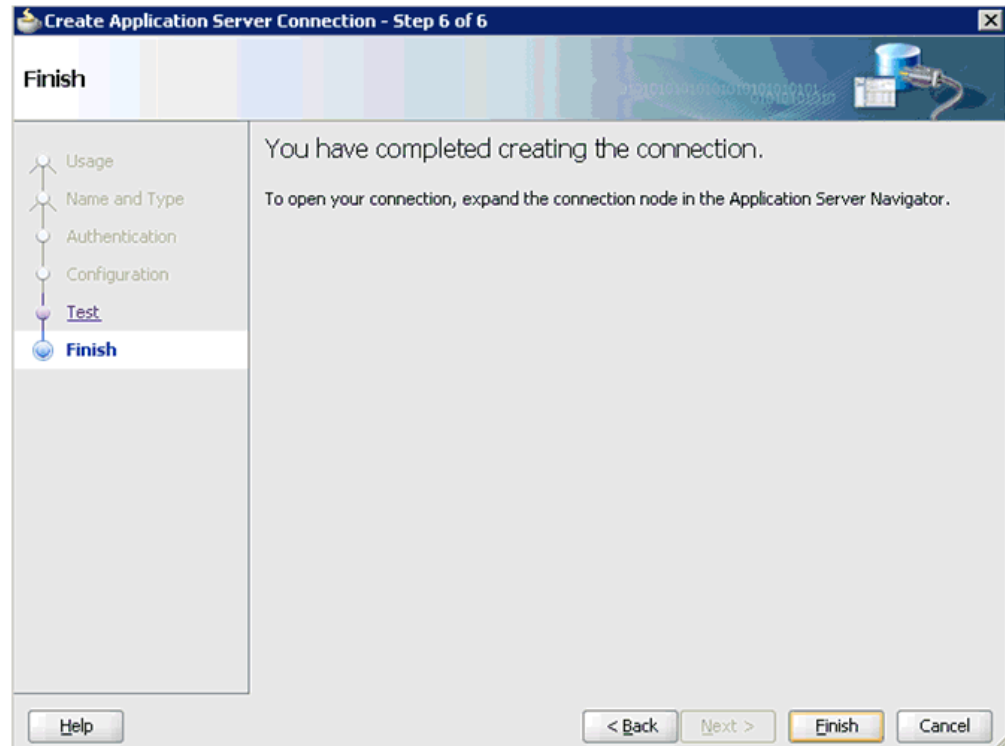
図 4-7 に示すように、「テスト」ページが表示されます。

図 4-7 「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 4-8 に示すように、「終了」ページが表示されます。

図 4-8 「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。

4.4 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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_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 が使用されていません。

この項では、次のトピックについて説明します。

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

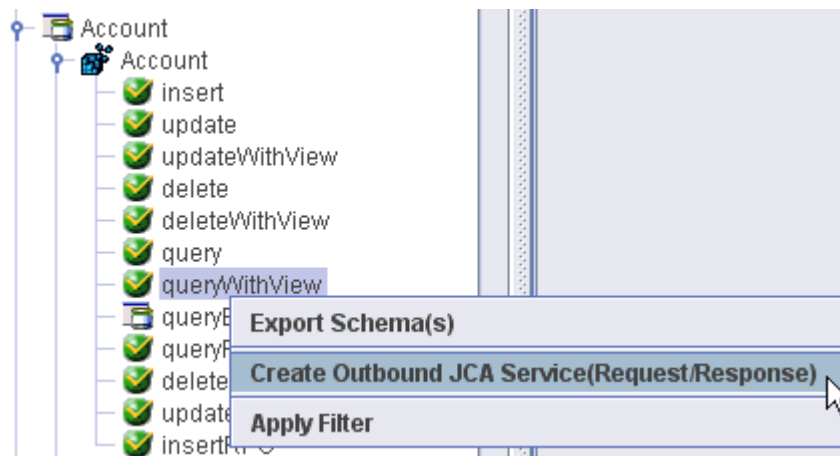
- 4.4.2 項「SOA 用の空のコンポジットの作成」
- 4.4.3 項「BPEL アウトバウンド・プロセスの定義」
- 4.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」
- 4.4.5 項「Oracle Enterprise Manager Console での入力 XML ドキュメントの起動」
- 4.4.6 項「アウトバウンド BPEL およびアウトバウンド・メディエータ・プロセスのテスト」

4.4.1 リクエスト/レスポンス・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対応する WSDL ファイルを生成する必要があります。リクエスト/レスポンス・サービス用の WSDL を生成するには、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、定義済の Siebel ターゲットに接続するか、新規ターゲットを作成します。
 アプリケーション・エクスプローラの起動およびターゲットへの接続の詳細は、2-2 ページ 2.1 項「アプリケーション・エクスプローラの起動」および 2-9 ページ「定義済のターゲットへの接続」を参照してください。
2. 接続先の Siebel ターゲットを展開します。
3. 図 4-9 に示すように、「ビジネス・オブジェクト」、「アカウント」、「アカウント」の順に展開します。

図 4-9 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



4. 「queryWithView」を右クリックしてアウトバウンド 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 4-10 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 4-10 「WSDL のエクスポート」 ダイアログ

5. デフォルトの名前を受け入れるかファイルの名前を指定します (「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/レスポンス型のサービス用に生成された WSDL ファイルの名前の末尾には「_invoke」が付きます。

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できました。

4.4.2 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、次のステップを実行します。

1. 新規の SOA アプリケーションを作成します。
2. 図 4-11 に示すように、新規の SOA アプリケーションの名前を入力し、「次へ」をクリックします。

図 4-11 「アプリケーションの名前付け」 ページ

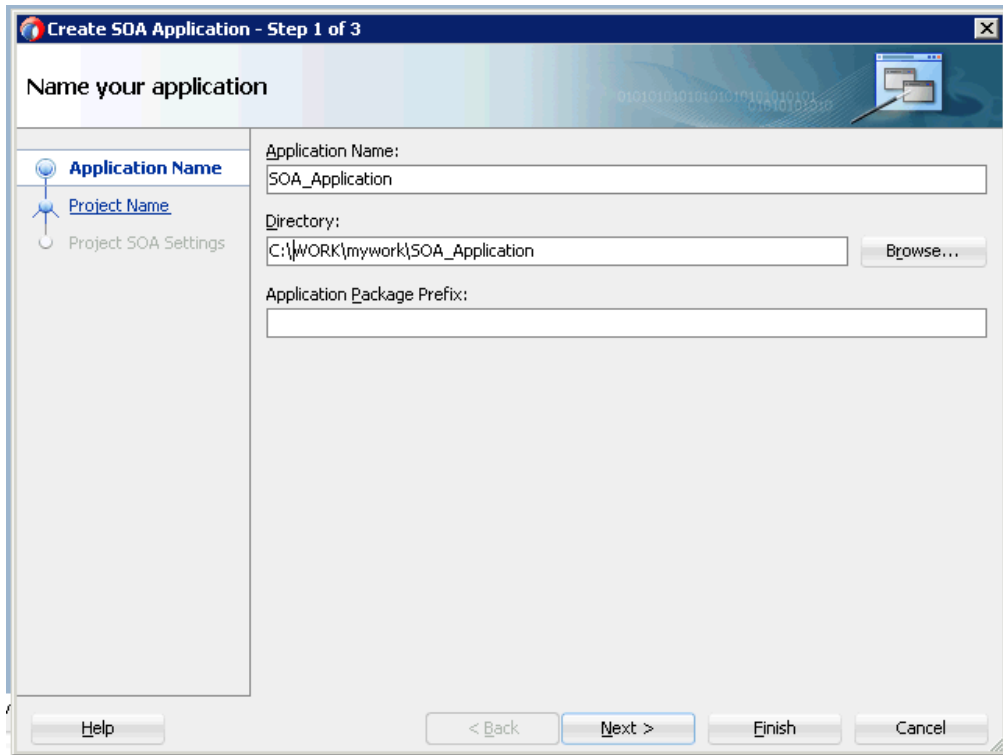
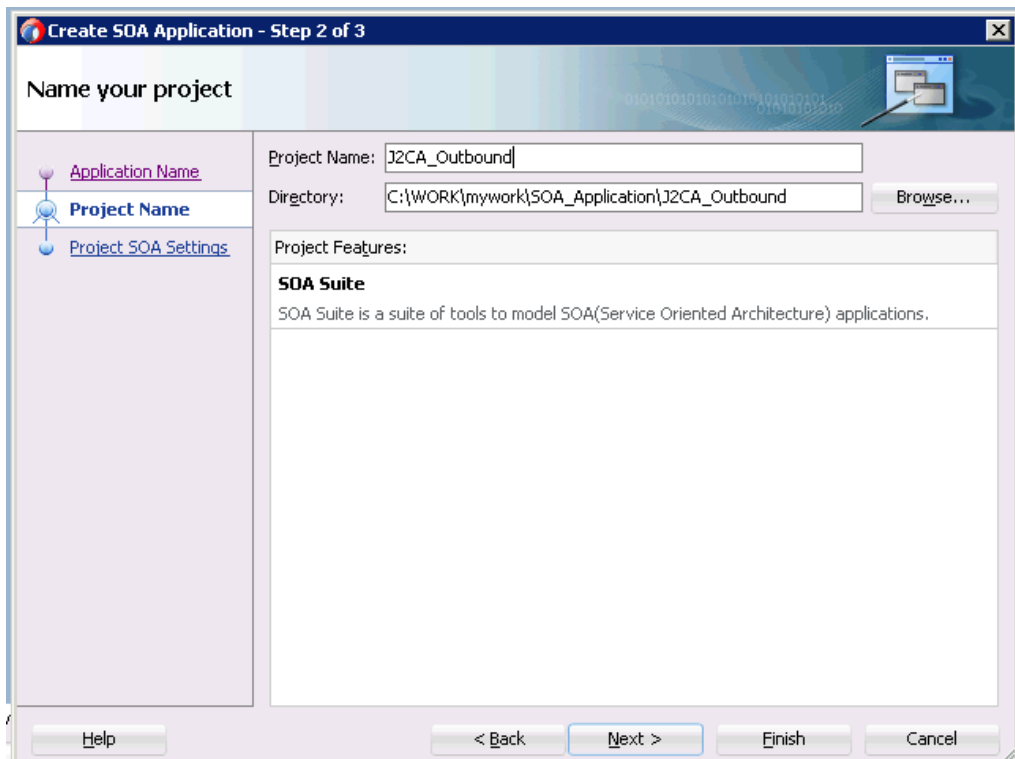


図 4-12 に示すように、「プロジェクトの名前付け」 ページが表示されます。

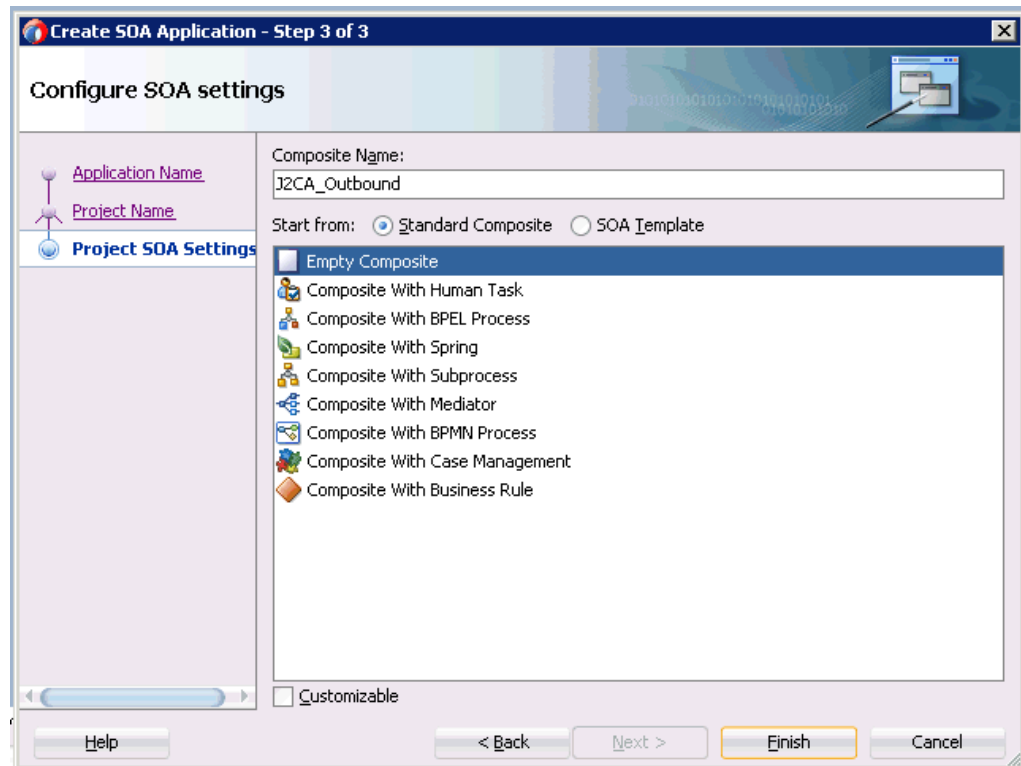
図 4-12 「プロジェクトの名前付け」 ページ



- プロジェクトの名前を入力し、「次へ」をクリックします。

図 4-13 に示すように、「SOA」ページが表示されます。

図 4-13 「SOA」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。

4.4.3 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。

- 4.4.3.1 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」
- 4.4.3.2 項「アウトバウンド BPEL プロセス・コンポーネントの構成」
- 4.4.3.3 項「12c のデプロイメントに関する既知の問題の調整」

4.4.3.1 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 4-14 に示すように、「サービス・アダプタ」ペインから「サード・パーティ・アダプタ」コンポーネントを「外部参照」ペインにドラッグ・アンド・ドロップします。

図 4-14 「サード・パーティ・アダプタ」コンポーネント

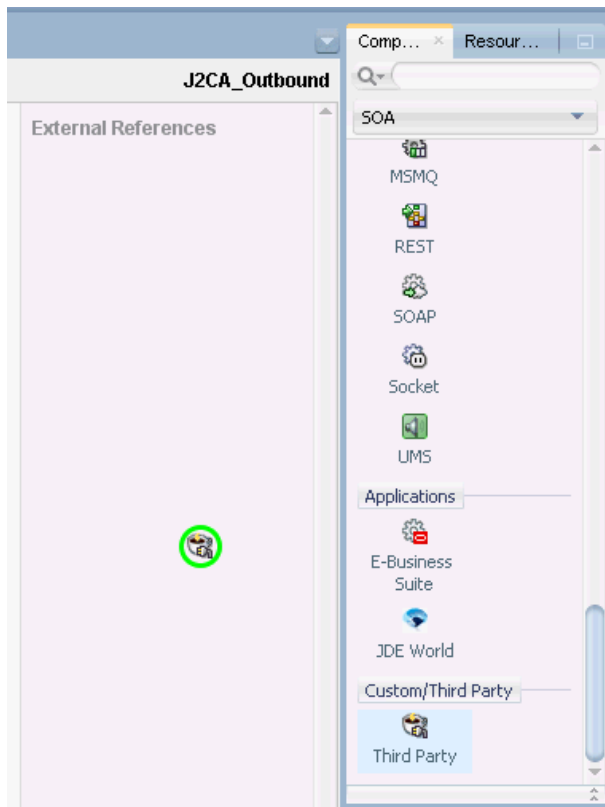
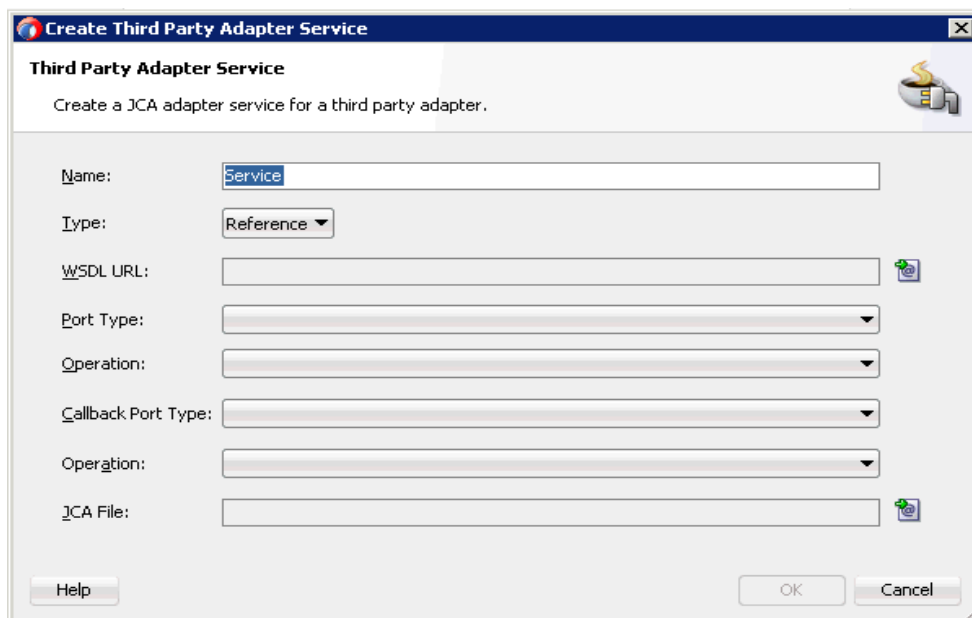


図 4-15 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

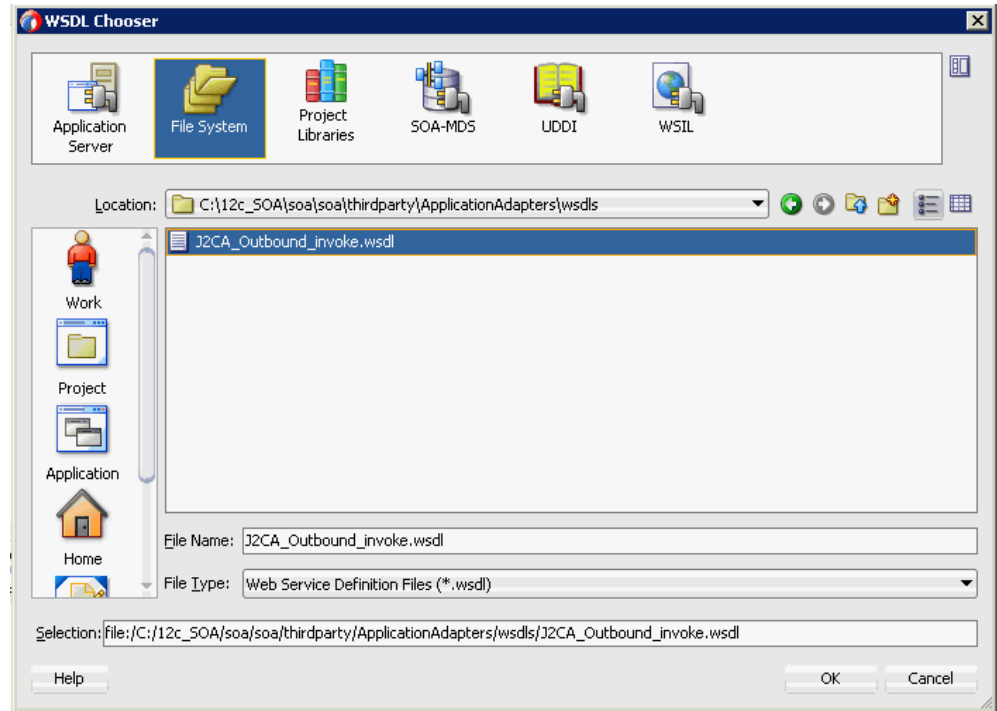
図 4-15 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



2. 「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)。

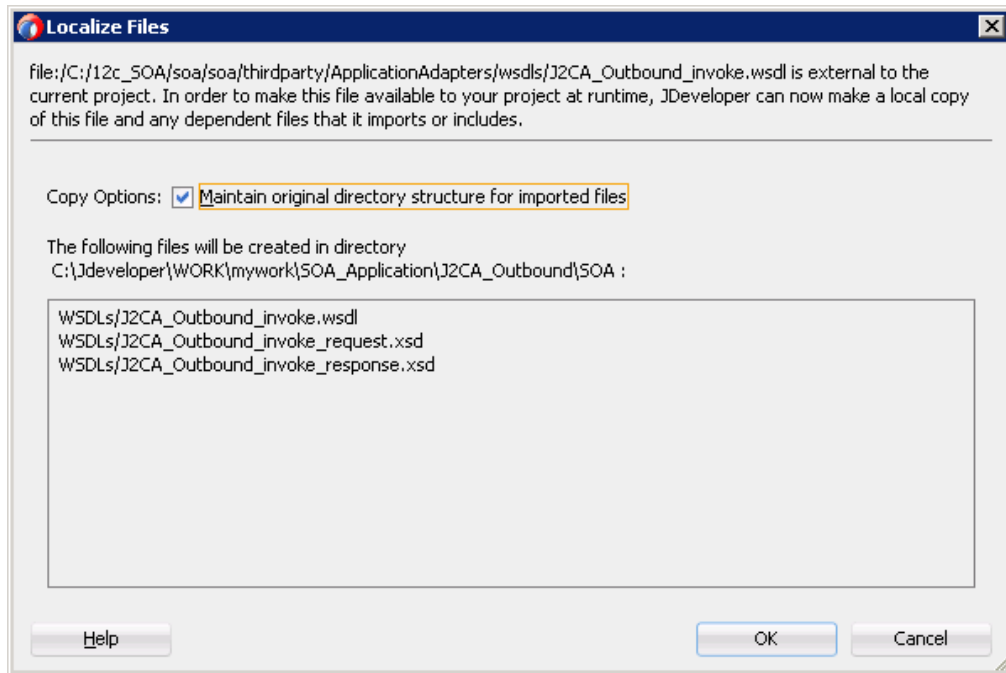
3. 「WSDL URL」フィールドの右の**既存の WSDL を検索**アイコンをクリックします。
図 4-16 に示すように、「WSDL チューザ」ダイアログが表示されます。

図 4-16 「WSDL チューザ」ダイアログ



4. 次のディレクトリでアウトバウンド WSDL ファイルを特定し、選択します。
<ADAPTER_HOME>\wsdl
5. 「OK」をクリックします。
図 4-17 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 4-17 「ファイルのローカライズ」ダイアログ

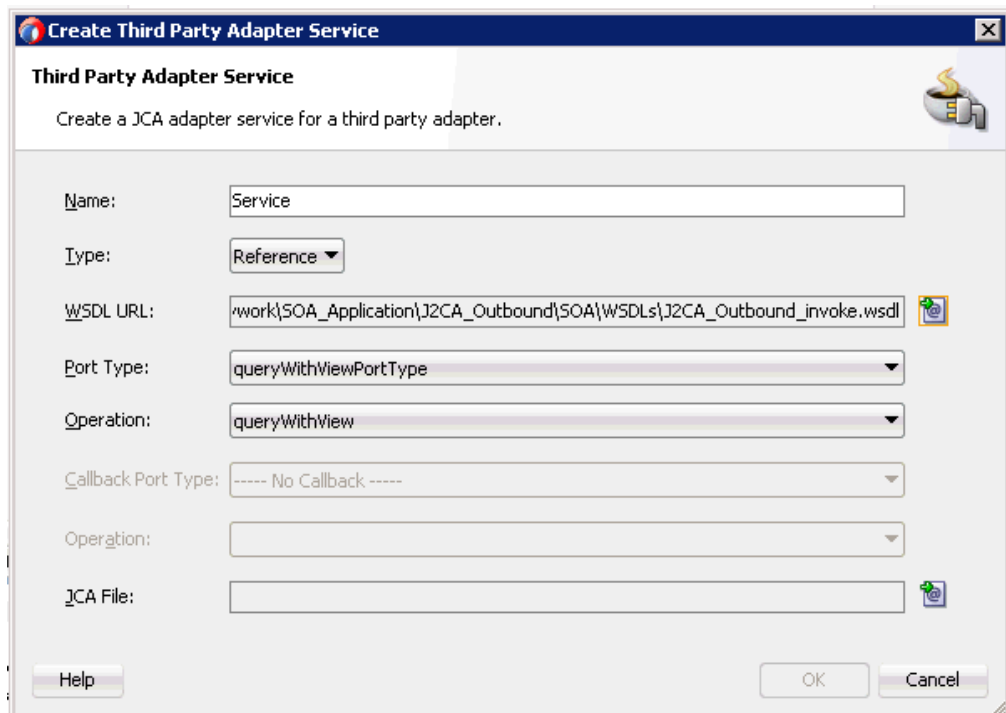


6. 「OK」をクリックします。

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 4-18 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。

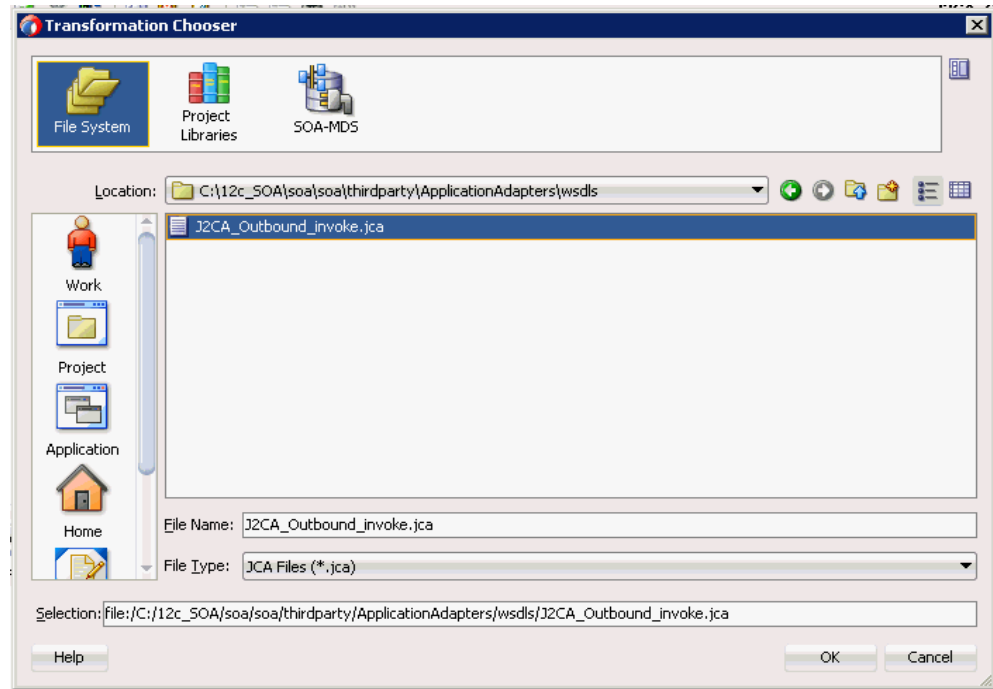
図 4-18 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



7. 「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 4-19 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 4-19 トランスフォーメーション・チューザ・ダイアログ



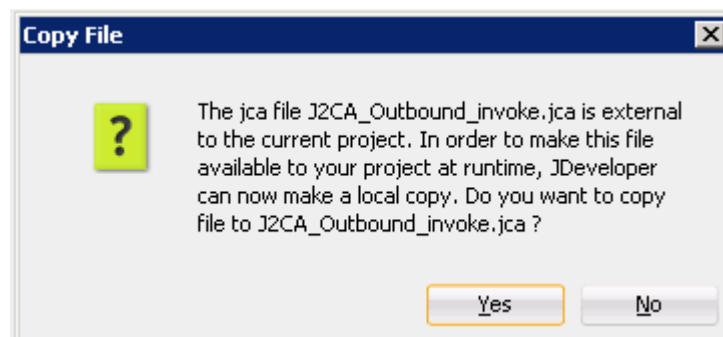
8. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。

<ADAPTER_HOME>\wsdl\

9. 「OK」をクリックします。

図 4-20 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 4-20 「ファイルのコピー」メッセージ

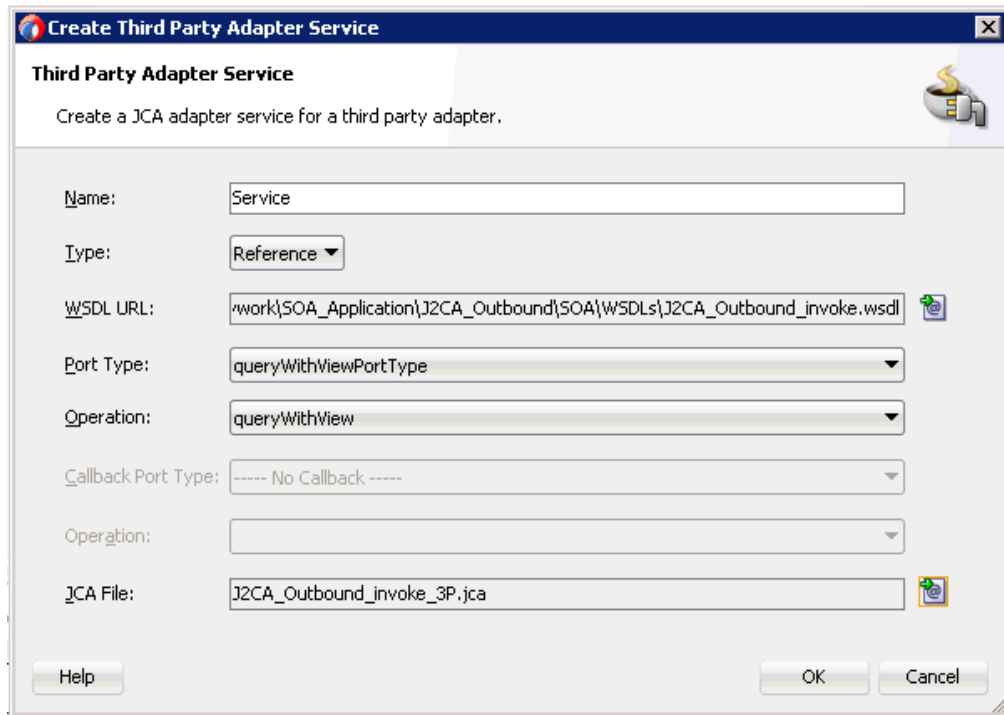


10. 「はい」をクリックします。

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 4-21 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 4-21 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



11. 「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。

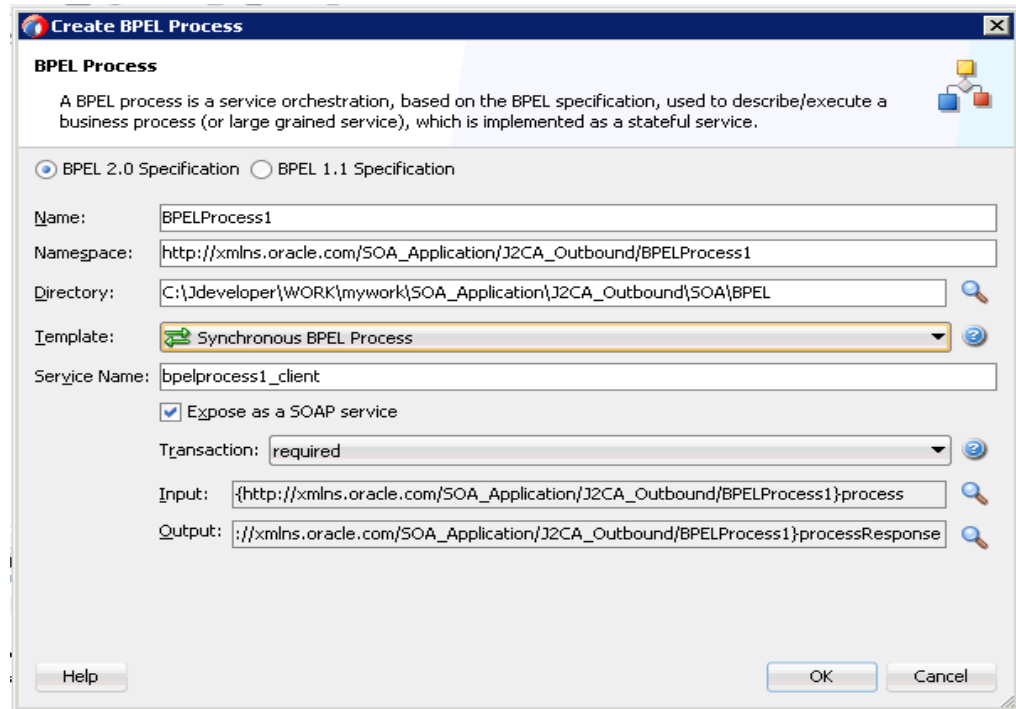
4.4.3.2 アウトバウンド BPEL プロセス・コンポーネントの構成

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「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。

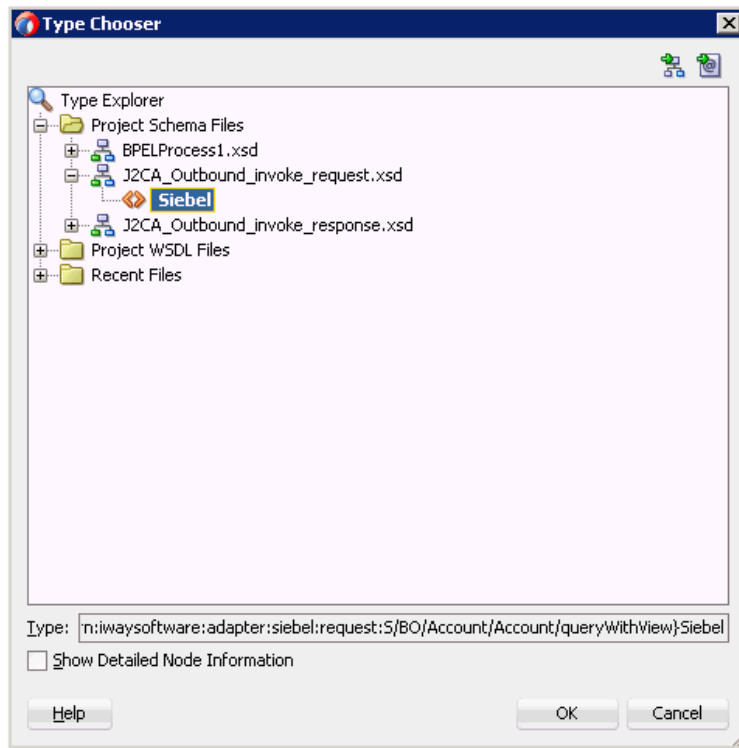
図 4-22 に示すように、「BPEL プロセスの作成」ダイアログが表示されます。

図 4-22 「BPEL プロセスの作成」 ダイアログ



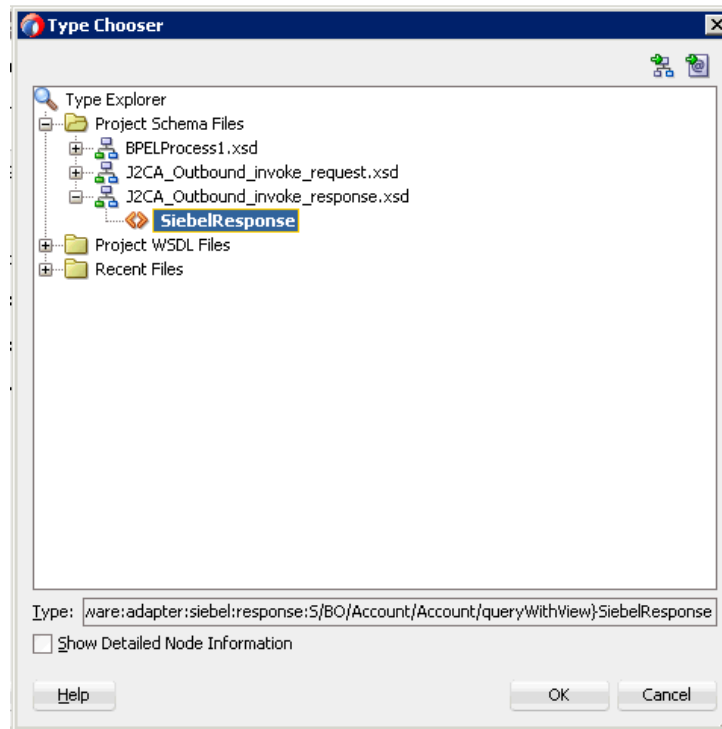
2. 「名前」フィールドに、新規のアウトバウンド BPEL プロセス・コンポーネントの名前を入力するか、デフォルト値のままにします。
「BPEL 2.0 仕様」オプションがデフォルトで選択されています。
3. 「テンプレート」リストで「同期 BPEL プロセス」を選択します。
4. 関連する XML リクエスト・スキーマ・ファイルの選択用の「入力」フィールドの右の「参照」アイコンをクリックします。
図 4-23 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 4-23 「タイプ・チューザ」 ダイアログ



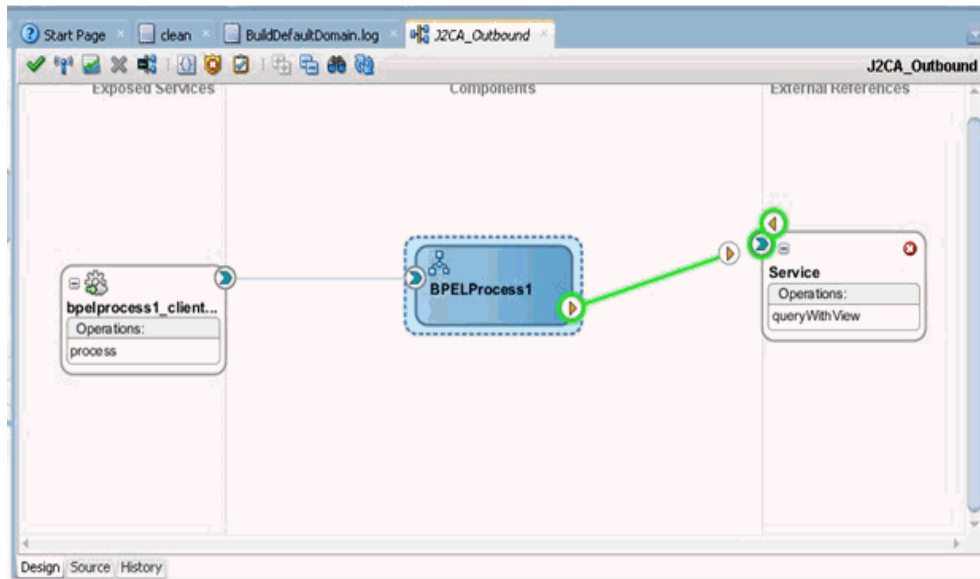
5. 「プロジェクトのスキーマ・ファイル」、「J2CA_Outbound_invoke_request.xsd」の順に展開し、「Siebel」を選択します。
6. 「OK」をクリックします。
「BPEL プロセスの作成」ダイアログに戻ります。
7. 関連する XML レスポンス・スキーマ・ファイルの選択用の「出力」フィールドの右の「参照」アイコンをクリックします。
図 4-24 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 4-24 「タイプ・チューザ」ダイアログ



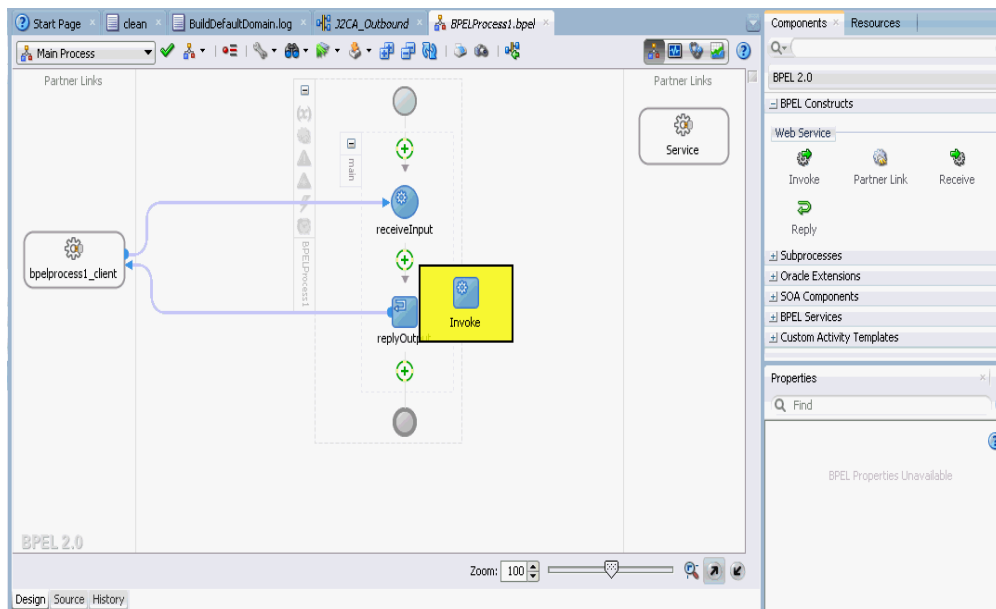
8. 「プロジェクトのスキーマ・ファイル」、「J2CA_Outbound_invoke_response.xsd」の順に展開し、「SiebelResponse」を選択します。
9. 「OK」をクリックします。
「BPEL プロセスの作成」ダイアログに戻ります。
10. 「OK」をクリックします。
11. 図 4-25 に示すように、アウトバウンド BPEL プロセス・コンポーネントとサード・パーティ・アダプタ・サービス・コンポーネント間の接続を作成します。

図 4-25 接続の作成



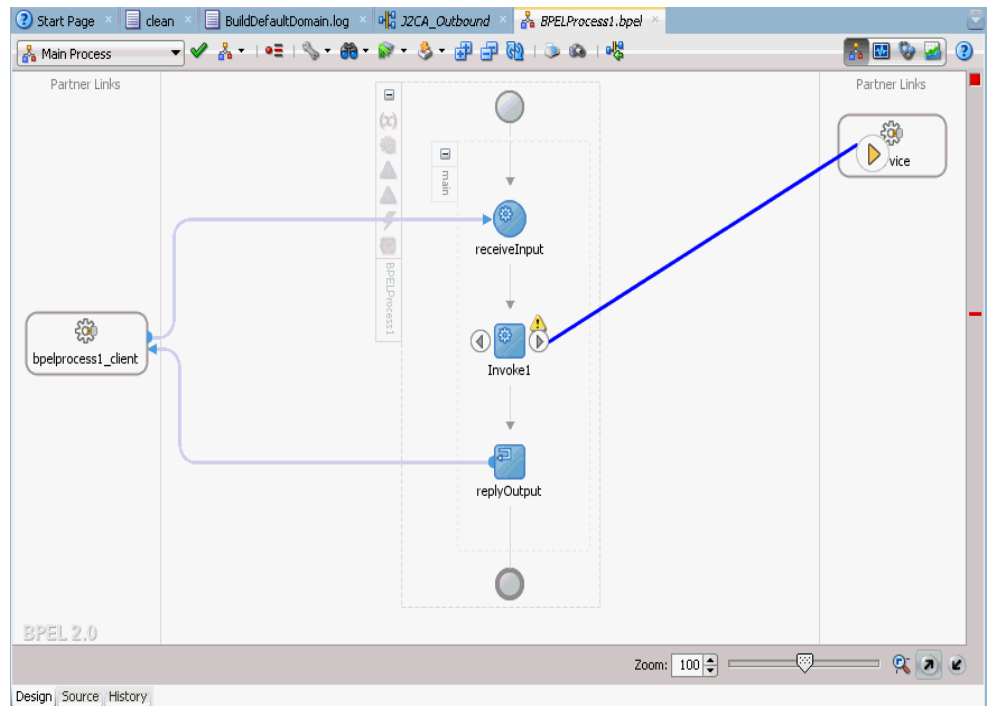
12. 「コンポーネント」 ペインで、アウトバウンド BPEL プロセス・コンポーネントをダブルクリックします。
13. 図 4-26 に示すように、「BPEL コンストラクト」の下の **Invoke** アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップし、**receiveInput** アクティビティ・コンポーネントと **replyOutput** アクティビティ・コンポーネントの間に置きます。

図 4-26 Invoke アクティビティ・コンポーネント



14. 図 4-27 に示すように、新規の Invoke アクティビティ・コンポーネント Service と サード・パーティ・アダプタ・サービス・コンポーネント (Service) の間に接続を作成します。

図 4-27 接続の作成



Invoke の編集ダイアログが表示されます。

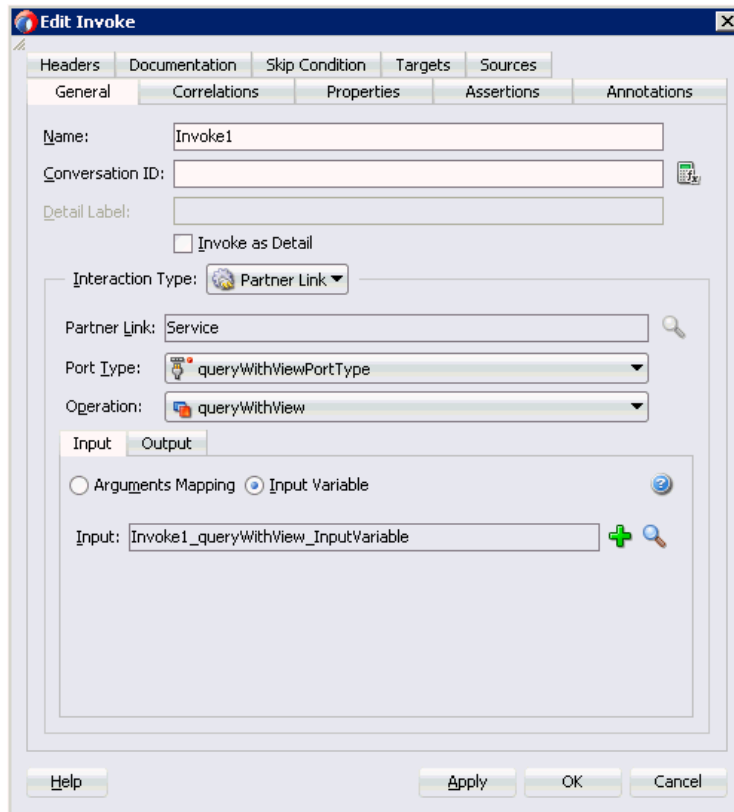
15. 新規入力変数を構成するために「入力」フィールドの右にある**プラス記号**アイコンをクリックします。

「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。

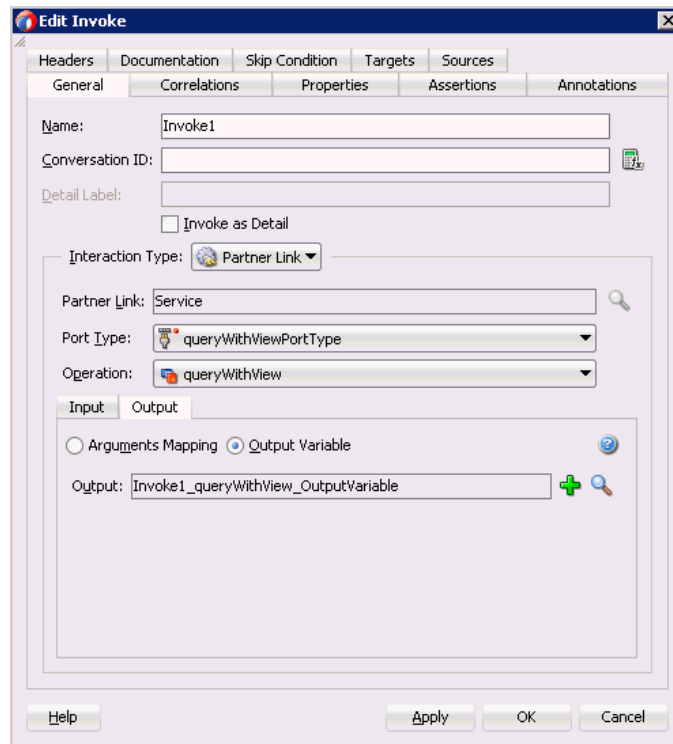
図 4-28 に示すように、Invoke の編集ダイアログに戻ります。

図 4-28 Invoke の編集ダイアログ



17. 「出力」タブを選択し、新規出力変数を構成するために「出力」フィールドの右にある**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 4-29 に示すように、Invoke の編集ダイアログに戻ります。

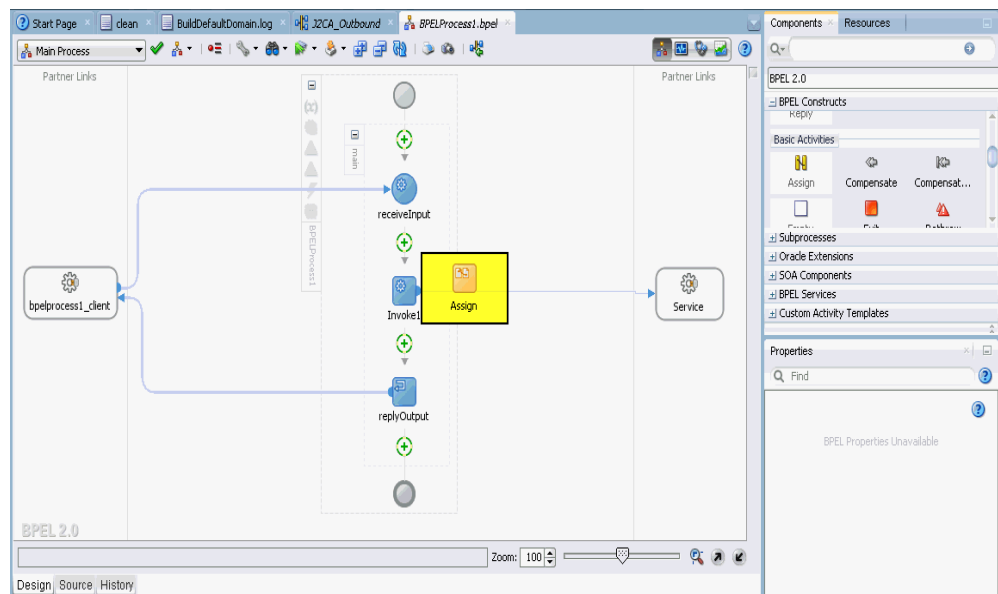
図 4-29 Invoke の編集ダイアログ



19. 「適用」をクリックし、「OK」をクリックします。

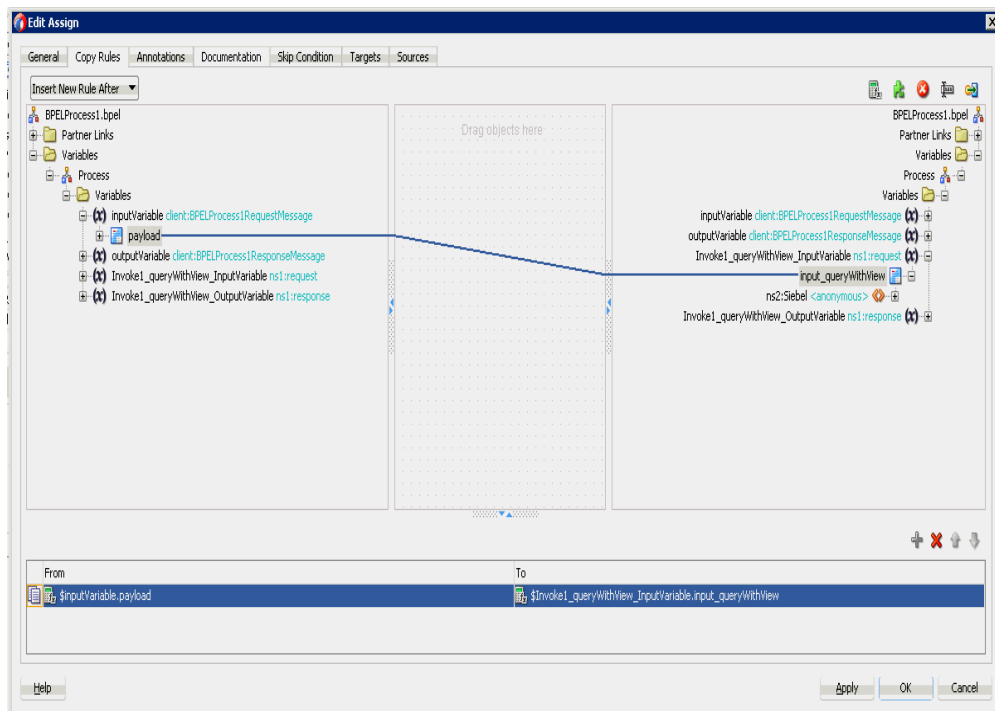
20. 図 4-30 に示すように、「BPEL コンストラクト」コンポーネントの下の **Assign** アクティビティを「コンポーネント」ペインにドラッグ・アンド・ドロップし、Receive アクティビティ・コンポーネント (receiveInput) と Invoke アクティビティ・コンポーネント (Invoke1) の間に置きます。

図 4-30 Assign アクティビティ・コンポーネント



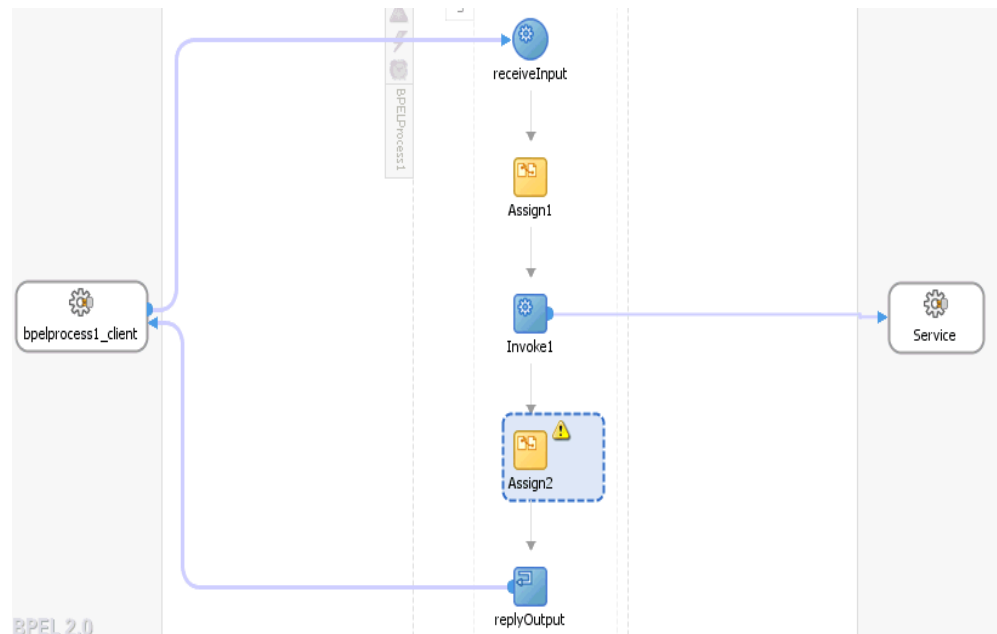
21. 新規の Assign アクティビティ・コンポーネント (**Assign1**) をダブルクリックします。
Assign の編集ダイアログが表示されます。
22. 左ペインの「変数」の下の「**InputVariable**」を展開し、「**payload**」を選択します。
23. 右ペインの「変数」の下の「**Invoke1_queryWithView_InputVariable**」を展開し、「**input_queryWithView**」を選択します。
24. **payload** 変数を **input_queryWithView** 変数にドラッグしてマップします。
図 4-31 に示すように、マップされた変数が強調表示された領域に表示されます。

図 4-31 Assign の編集ダイアログ



25. 「適用」をクリックし、「OK」をクリックします。
26. **Assign** 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、**Invoke** アクティビティ (**Invoke1**) と **Reply** アクティビティ (**replyOutput**) の間に置きます。
27. 図 4-32 に示すように、新規の Assign 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 4-32 新規の Assign アクティビティ・コンポーネント

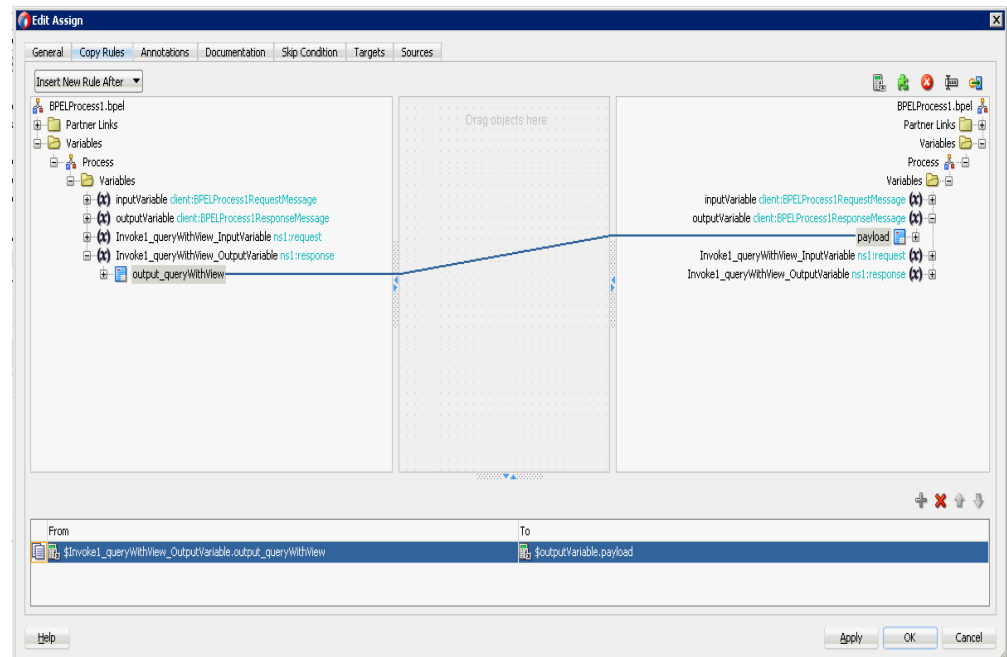


Assign の編集ダイアログが表示されます。

28. 左ペインの「変数」の下の「Invoke1_queryWithView_OutputVariable」を展開し、「output_queryWithView」を選択します。
29. 右ペインの「変数」の下の「outputVariable」を展開し、「payload」を選択します。
30. output_queryWithView 変数を payload 変数にドラッグしてマップします。

図 4-33 に示すように、マップされた変数が強調表示された領域に表示されます。

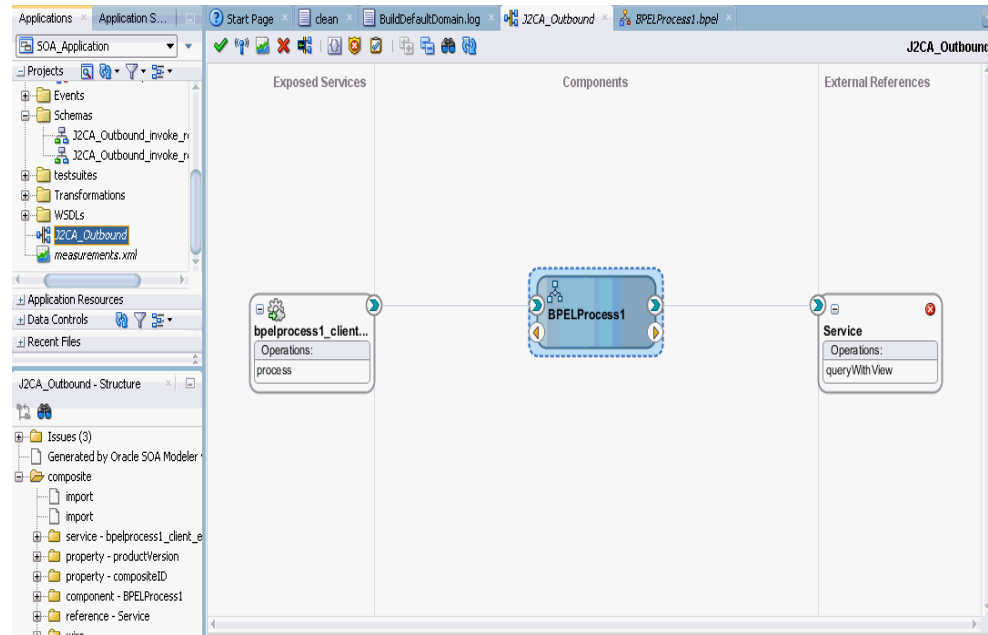
図 4-33 Assign の編集ダイアログ



31. 「適用」をクリックし、「OK」をクリックします。

図 4-34 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 4-34 アクティビティ・コンポーネント・ペイン



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

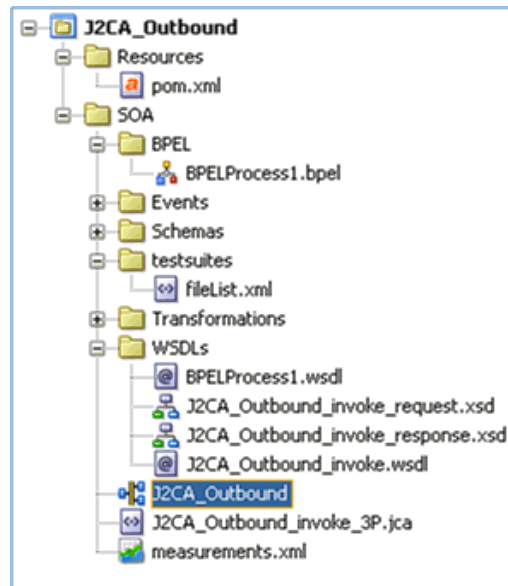
これで、BPEL アウトバウンド・プロセスをデプロイする準備ができました。

4.4.3.3 12c のデプロイメントに関する既知の問題の調整

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

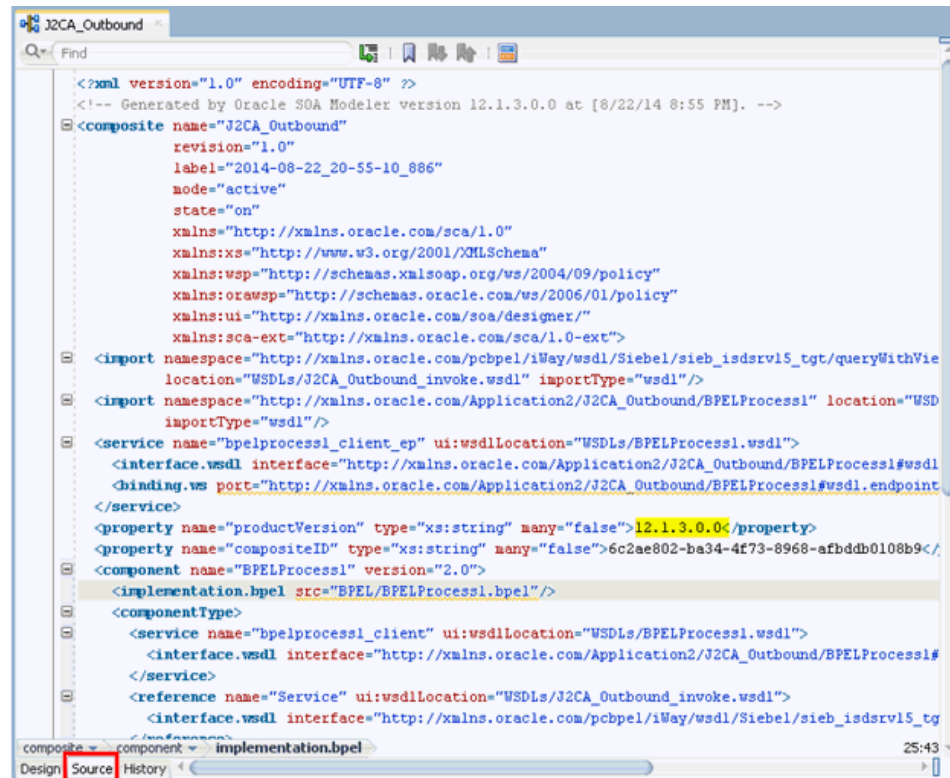
1. 図 4-35 に示すように、作成したプロセスの「J2CA_Outbound」(作成した BPEL プロセス)をダブルクリックします。

図 4-35 「J2CA_Outbound」ノード



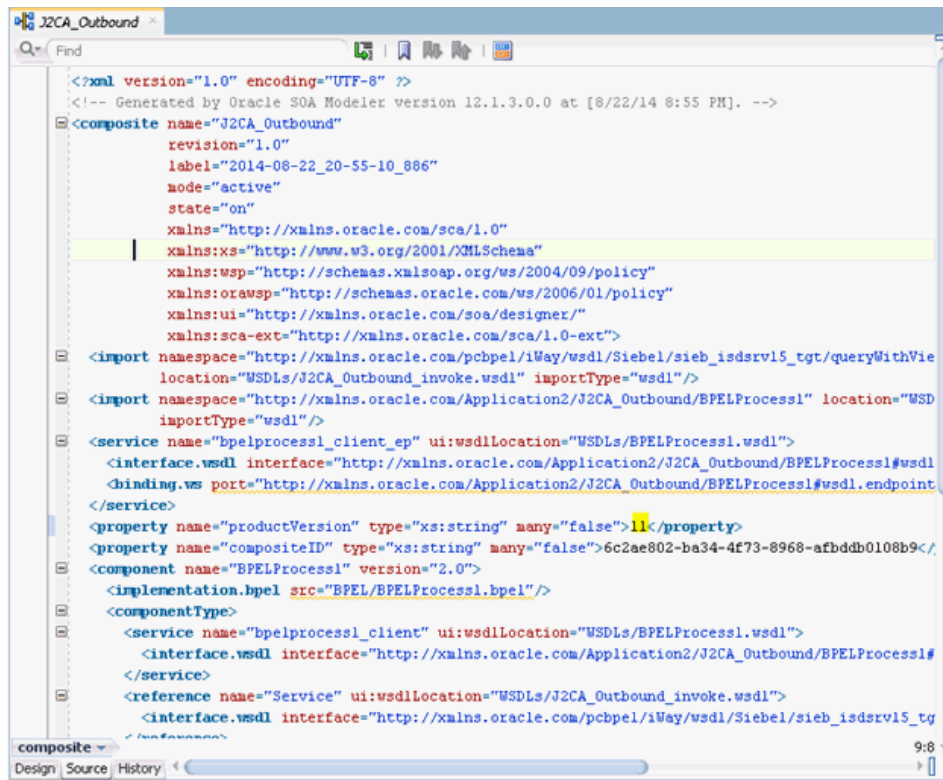
2. 図 4-36 に示すように、開いたプロセスの下の「ソース」タブをクリックします。

図 4-36 「ソース」タブ



3. 図 4-37 に示すように、productVersion プロパティの値を 12.1.3.0.0 から 11 に変更します。

図 4-37 プロパティ値



4. 変更内容を保存し、プロジェクトのデプロイに進みます。

4.4.4 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 4-38 に示すように、左ペインで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 4-38 「J2CA_Outbound」オプション

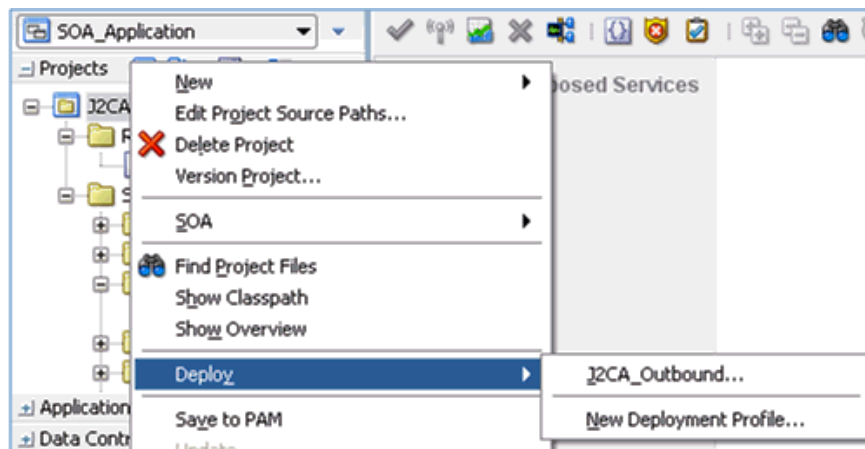
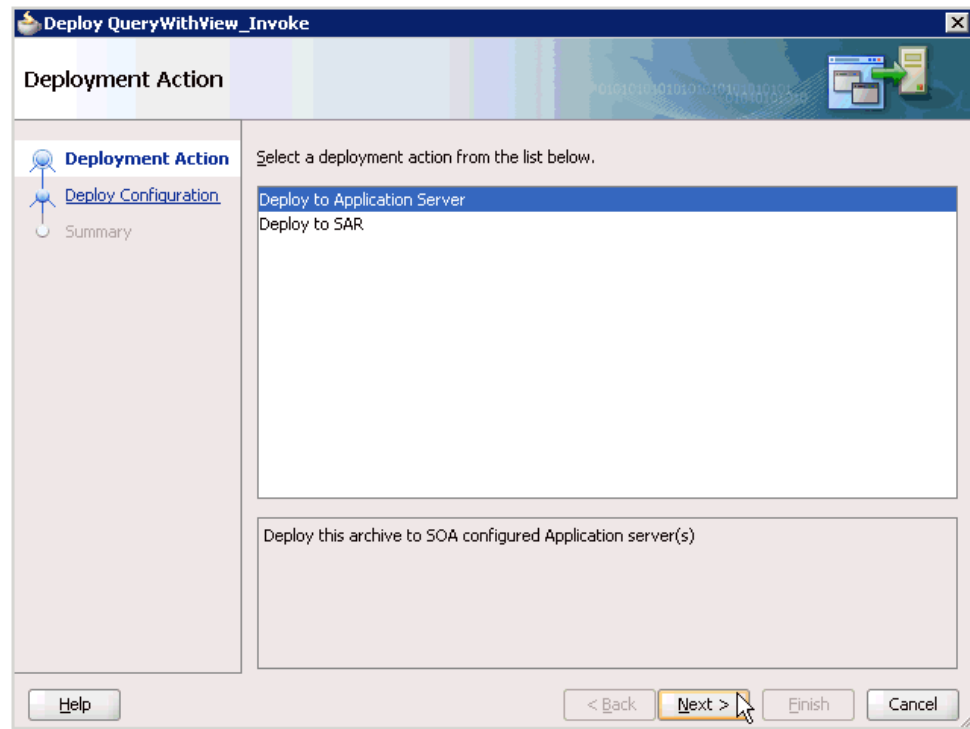


図 4-39 に示すように、「デプロイメント・アクション」ページが表示されます。

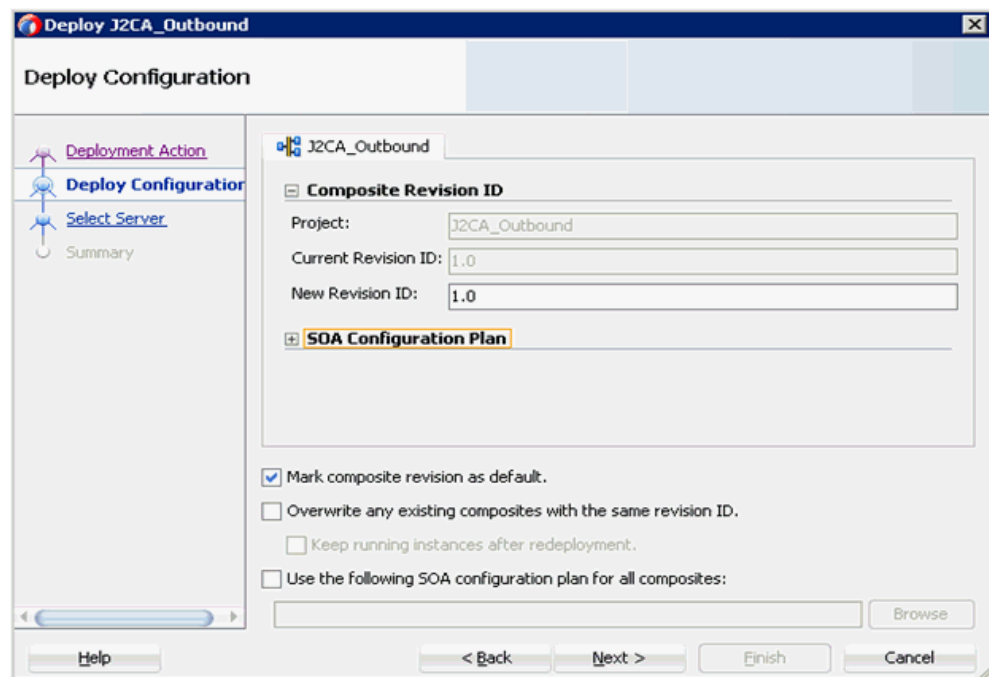
図 4-39 「デプロイメント・アクション」 ページ



2. 「アプリケーション・サーバーへのデプロイ」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 4-40 に示すように、「構成のデプロイ」ページが表示されます。

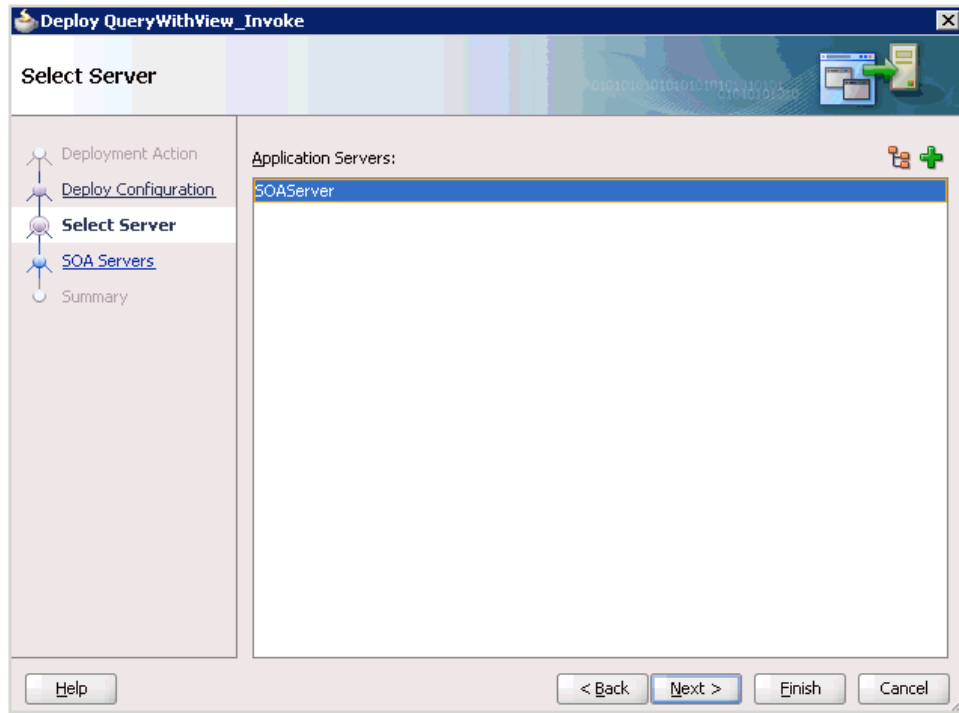
図 4-40 「構成のデプロイ」 ページ



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 4-41 に示すように、「サーバーの選択」ページが表示されます。

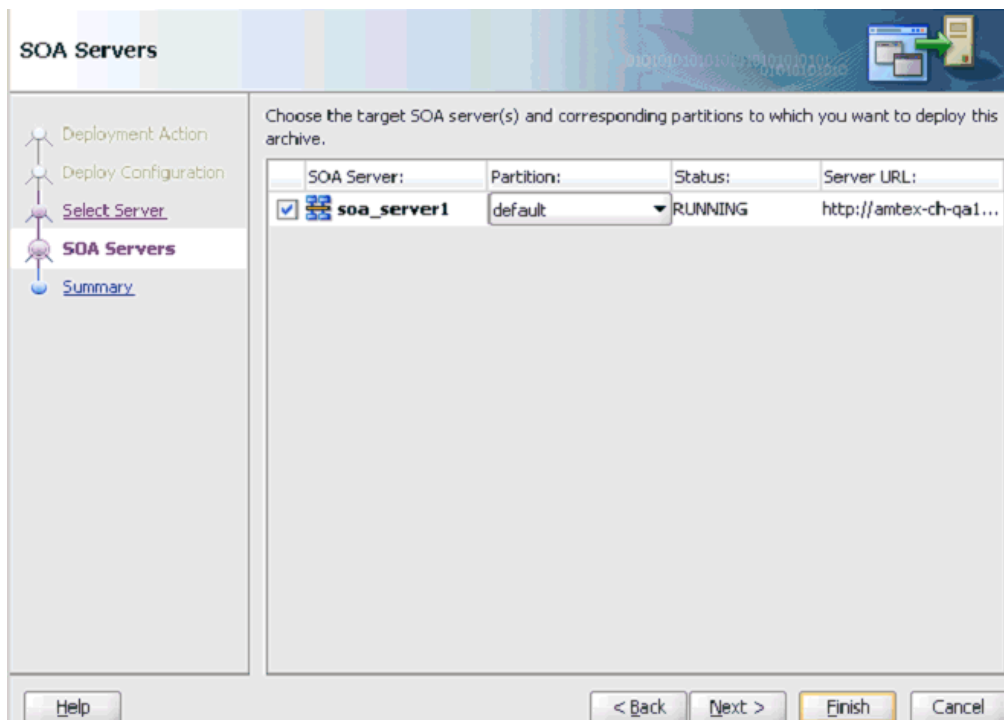
図 4-41 「サーバーの選択」ページ



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 4-42 に示すように、「SOA サーバー」ページが表示されます。

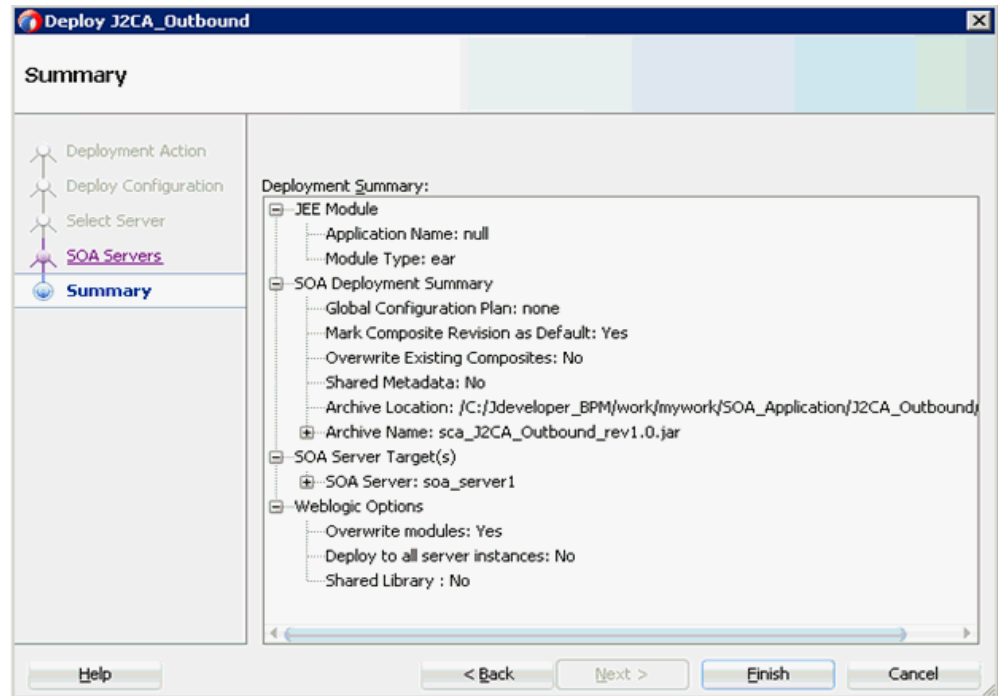
図 4-42 「SOA サーバー」ページ



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 4-43 に示すように、「サマリー」ページが表示されます。

図 4-43 「サマリー」ページ



7. プロジェクトの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 4-44 に示すように、プロセスが正常にデプロイされます。

図 4-44 デプロイメントの成功を伝えるメッセージ



4.4.5 Oracle Enterprise Manager Console での入力 XML ドキュメントの起動

Oracle Enterprise Manager Console 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 Console にログオンします。

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。

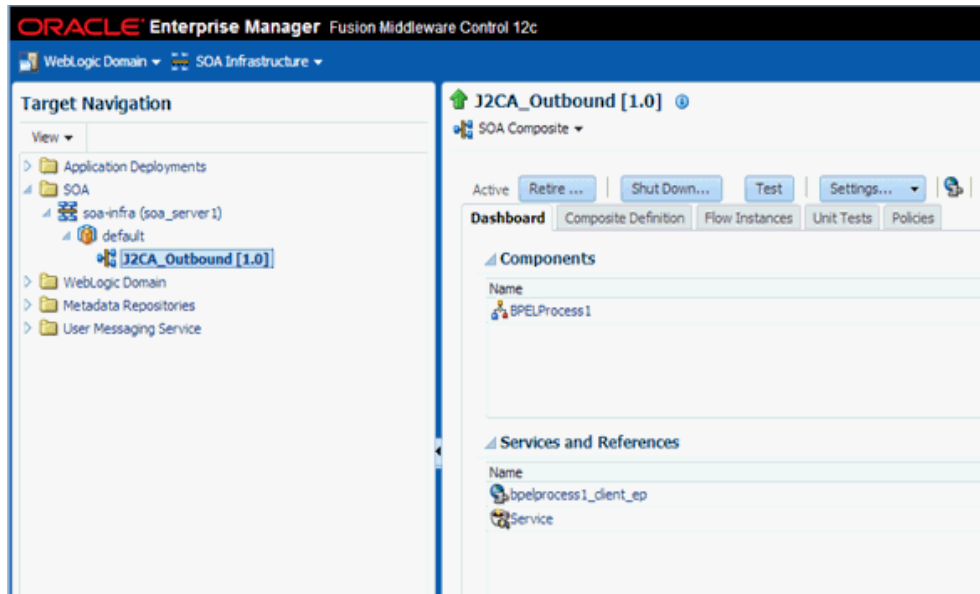
2. 左ペインの「ターゲット・ナビゲーション」をクリックし、「SOA」を展開し、「soa-infra (soa_server1)」を選択します。

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。

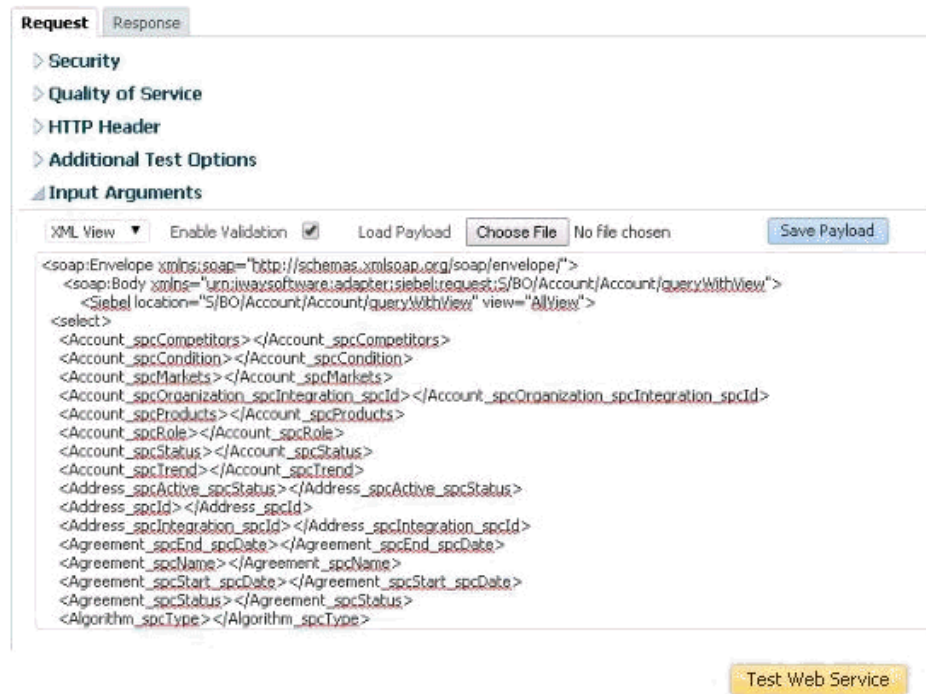
2. 「SOA」を展開して「soa-infra (soa_server1)」を選択し、「デフォルト」をクリックします。
3. [図 4-45](#) に示すように、使用可能なプロジェクト (J2CA_Outbound など) を選択して「テスト」をクリックします。

図 4-45 「テスト」ボタン



4. 「リクエスト」タブをクリックします。
5. [図 4-46](#) に示すように、リストから XML ビューを選択します。

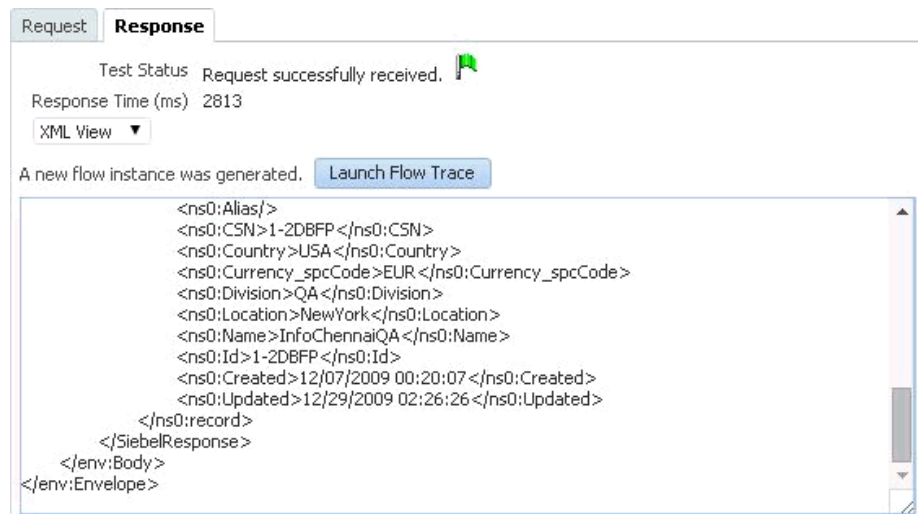
図 4-46 「入力引数」 リスト



6. 「入力引数」領域に適切な入力 XML ドキュメントを指定して「Web サービスのテスト」をクリックします。

図 4-47 に示すように、Oracle Enterprise Manager Console に出力レスポンスが表示されます。

図 4-47 受信された出力レスポンス



4.4.6 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 Console 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。

Oracle Enterprise Manager Console を使用してメディアエータ・データ・フローをテストできます。メディアエータ・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。メディアエータ・アウトバウンド・プロセスの作成の詳細は、第 5 章「Oracle SOA Suite でのメディアエータ・サービス・コンポーネントとの統合」を参照してください。

4.5 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、Siebel イベントの統合について説明します。

このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_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 Oracle JDeveloper が使用されています。

この項では、次のトピックについて説明します。

- 4.5.1 項「イベント統合のための WSDL の生成」
- 4.5.2 項「SOA 用の空のコンポジットの作成」
- 4.5.3 項「BPEL インバウンド・プロセスの定義」
- 4.5.4 項「BPEL インバウンド・プロセスのデプロイ」
- 4.5.5 項「Siebel でのイベントのトリガー」

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対応する WSDL ファイルを生成する必要があります。詳細は、4-34 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。

4.5.1 イベント統合のための WSDL の生成

インバウンド J2CA サービスごとに別個のチャンネルを作成し、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対話用の WSDL を生成する際、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。

この項では、次のトピックについて説明します。

- 4.5.1.1 項「チャンネルの作成」
- 4.5.1.2 項「統合オブジェクト・ノードの作成」
- 4.5.1.3 項「イベント通知のための WSDL の生成」

4.5.1.1 チャンネルの作成

インバウンド J2CA サービスごとに別個のチャンネルを作成し、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対話用の WSDL を生成する際、そのチャンネルを選択する必要があります。

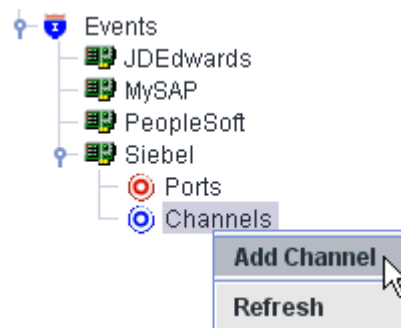
注意： 2つ以上のイベントで同じチャンネルが共有されている場合、正しい BPEL プロセスにイベント・メッセージが配信されないことがあります。

チャンネルを作成するには次を行います。

1. 左ペインで「イベント」をクリックします。
2. 「Siebel」ノードを展開します。

図 4-48 に示すように、左ペインに「ポート」ノードと「チャンネル」ノードが表示されます。

図 4-48 「Siebel」ノード



3. 「チャンネル」を右クリックして「チャンネルの追加」を選択します。

図 4-49 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 4-49 「チャンネルの追加」ダイアログ

The screenshot shows a standard Windows-style dialog box titled "Add Channel". It has a close button in the top right corner. The dialog is organized into several sections: "Name:" with a text input field containing "SiebelEvent"; "Description:" with a larger empty text area; "Protocol:" with a dropdown menu currently set to "HTTP Listener"; and two list boxes, "Available Port(s)" and "Selected Port(s)", with four arrow buttons (>>, >, <, <<) between them for moving items. At the bottom, there are "Next" and "Cancel" buttons, with a mouse cursor hovering over the "Next" button.

次のステップを実行します。

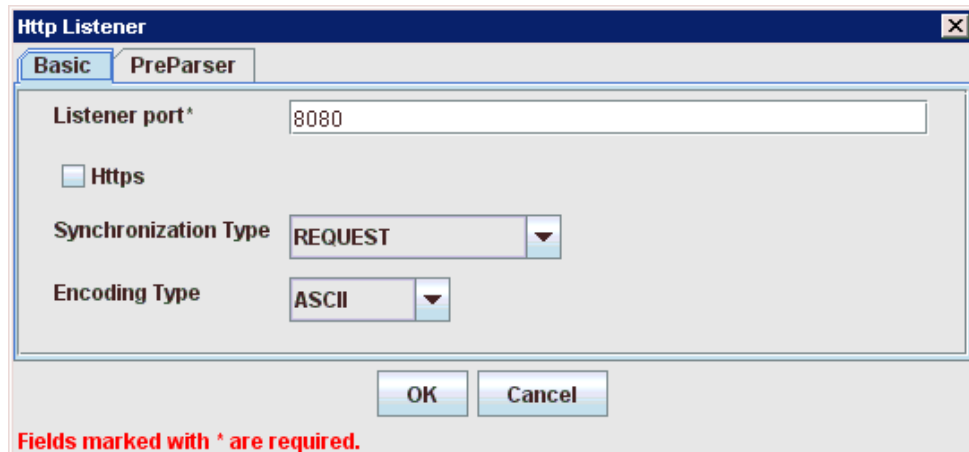
- a. チャンネルの名前 (SiebelEvent など) を入力します。
- b. 簡単な説明を入力します。
- c. 「プロトコル」リストから「HTTP リスナー」、MQ Series リスナー、またはファイル・リスナーを選択します。

デモ用に、この手順では、例として HTTP リスナーを使用します。

4. 「次へ」をクリックします。

図 4-50 に示すように、標準ダイアログが表示されます。

図 4-50 標準ダイアログ

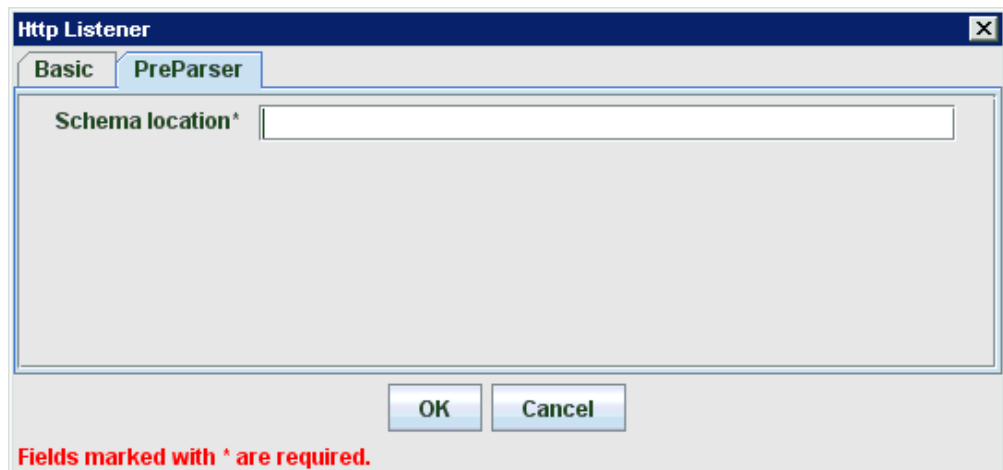


5. 次の表で指定されているようにシステム情報を入力します。

パラメータ	説明
リスナー・ポート	Siebel イベント・データをリスニングするポート。
Https	セキュアな HTTP 接続の場合は Https チェック・ボックスを選択します。このオプションは現在サポートされていません。
同期タイプ	リストから REQUEST_RESPONSE (推奨オプション) を選択します。
エンコーディング・タイプ	使用するエンコーディング・タイプをリストから選択します。デフォルトでは、「ASCII」が選択されています。

6. 図 4-51 に示すように、プリパーサー・タブをクリックします。

図 4-51 プリパーサー・タブ



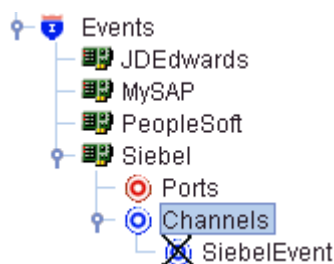
7. アプリケーション・エクスプローラの「スキーマのエクスポート」オプションを使用して統合オブジェクト・ノード用に生成されたスキーマ・ファイルの場所を指定します。

注意: 実行時、Oracle Application Adapter for Siebel は、プリパーサー・タブで指定されたスキーマを使用して Siebel に公開されたドキュメントにネームスペースを追加します。プリパーサー・タブのスキーマの場所フィールドが空の場合、Siebel に公開されたドキュメントにネームスペースが含まれないため、BPEL プロセスとメディアエータ・プロセスが適切に機能しません。

8. 「OK」をクリックします。

図 4-52 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。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。

図 4-52 新規チャンネル・ノード



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。テストおよびデバッグ目的で起動する場合はランタイム前に停止してください。

統合オブジェクト・ノードを作成する必要があります。

4.5.1.2 統合オブジェクト・ノードの作成

1. アプリケーション・エクスプローラを起動します。
2. 図 4-53 に示すように、「アダプタ」ノードを展開します。

図 4-53 「Siebel」ノードの下の切断されている Siebel ターゲット・ノード「Siebel」



次のステップを実行します。

- a. 「Siebel」ノードを展開します。
定義されている Siebel ターゲットがアダプタ・ノードの下に表示されます。
 - b. 「Siebel」ノードの下のターゲット名 (siebel など) をクリックします。
「接続」ダイアログに、入力した値が表示されます。
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、「接続」を選択します。

図 4-54 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 4-54 接続されているノード



5. 「統合オブジェクト」ノードを展開して「Sample Account」を選択します。
6. 「Sample Account」ノードを右クリックして IO ノードの追加を選択します。
図 4-55 に示すように、IO ノードの追加ダイアログが表示されます。

図 4-55 IO ノードの追加ダイアログ

7. 「ノード名」フィールドにノード名 (Sample_Account など) を入力し、スキーマの場所フィールドに Sample Account の XSD ファイルへのパスを入力します。

以下の点に注意してください。

- **Siebel 7.5 以降の場合** : XSD スキーマを Siebel Tools から直接生成します。アプリケーション・エクスプローラで Web サービスを作成する際、XSD スキーマを使用します。Siebel Tools を使用して XSD スキーマを生成した後、これを使用して IO ノードおよび Web サービスを作成します。
 - **Siebel 7.0 の場合** : Siebel Tools から XSD スキーマを直接生成できません。XDR スキーマのみ作成できます。Web サービスを作成する前に、まずアプリケーション・エクスプローラを使用して XDR スキーマから XSD スキーマを生成する必要があります。
8. XSD スキーマがすでに生成されている場合、「XSD スキーマ」を選択します。Siebel で生成された XDR スキーマを使用する場合、「XSD スキーマ」オプションを選択しないでください。
 9. 「プロトコル」リストからプロトコル (HTTP、ファイルまたは MQ Series) を選択します。
 10. 「続行」をクリックします。

図 4-56 に示すように、新規統合オブジェクト・ノードが追加されます。

図 4-56 統合オブジェクト・ノード



注意: 統合オブジェクト・ノードおよびチャネルの作成後、Oracle WebLogic Server を再起動する必要があります。

4.5.1.3 イベント通知のための WSDL の生成

チャネルとこれに関連付ける統合オブジェクトを作成したら、アプリケーション・エクスプローラを使用してイベント用の WSDL を生成する必要があります。

アプリケーション・エクスプローラで「アダプタ」ノード下の Siebel ターゲットに接続されている必要があります。ターゲットの定義方法と接続方法の詳細は、2-5 ページ「Siebel 用の接続 (ターゲット) の確立」を参照してください。

Siebel ターゲットに接続したら、次のようにしてイベント用の WSDL を生成します。

1. 図 4-57 に示すように、統合オブジェクト・ノード (Sample_Account など) を右クリックし、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 4-57 アプリケーション・エクスプローラで選択されているインバウンド JCA サービスの作成 (イベント) オプション

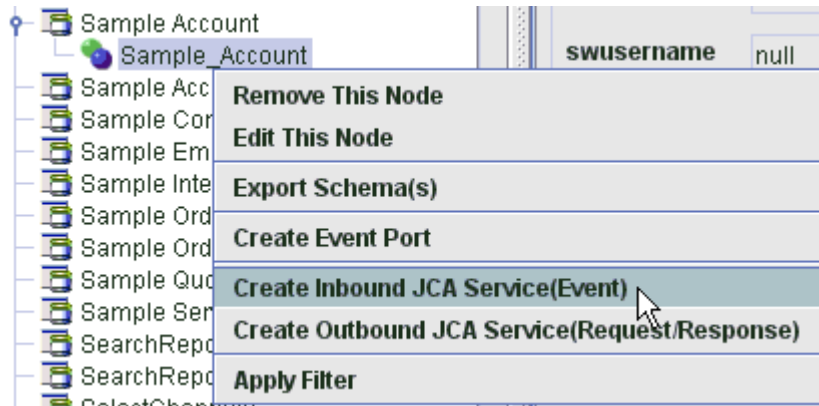


図 4-58 に示すように、「WSDL のエクスポート」ダイアログが表示されます。

図 4-58 「WSDL のエクスポート」ダイアログ

注意： スキーマ検証オプション (ルート) は Oracle Application Adapter for Siebel には適用されません。

次のステップを実行します。

- a. 「名前」フィールドに、WSDL ファイルの名前を指定します。

.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、イベント用に生成された WSDL ファイルの名前の末尾には「_receive」が付きます。

- b. 「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 (SiebelEvent など)。

重要： インバウンド・サービスごと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。

2. 「OK」をクリックします。

4.5.2 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、次のステップを実行します。

1. 新規の SOA アプリケーションを作成します。
2. 新規の SOA アプリケーションの名前を入力し、「次へ」をクリックします。
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクトの名前を入力し、「次へ」をクリックします。

「SOA」ページが表示されます。

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

詳細は、4-9 ページ 4.4.2 項「SOA 用の空のコンポジットの作成」を参照してください。

4.5.3 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。

- 4.5.3.1 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」
- 4.5.3.2 項「インバウンド BPEL プロセス・コンポーネントの作成」

4.5.3.1 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 4-59 に示すように、「サービス・アダプタ」ペインから「サード・パーティ・アダプタ」コンポーネントを「公開されたサービス」ペインにドラッグ・アンド・ドロップします。

図 4-59 「サード・パーティ・アダプタ」コンポーネント

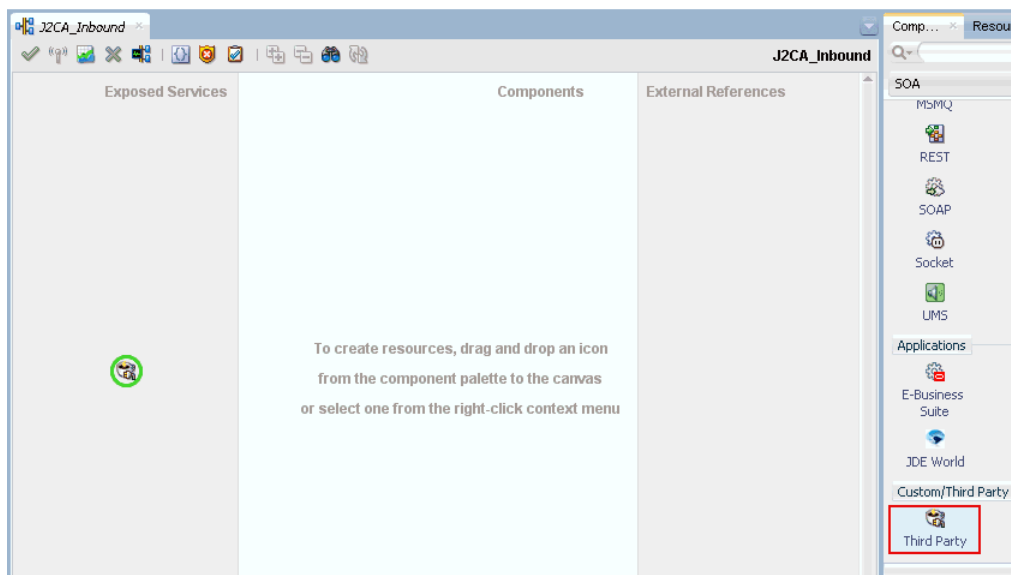
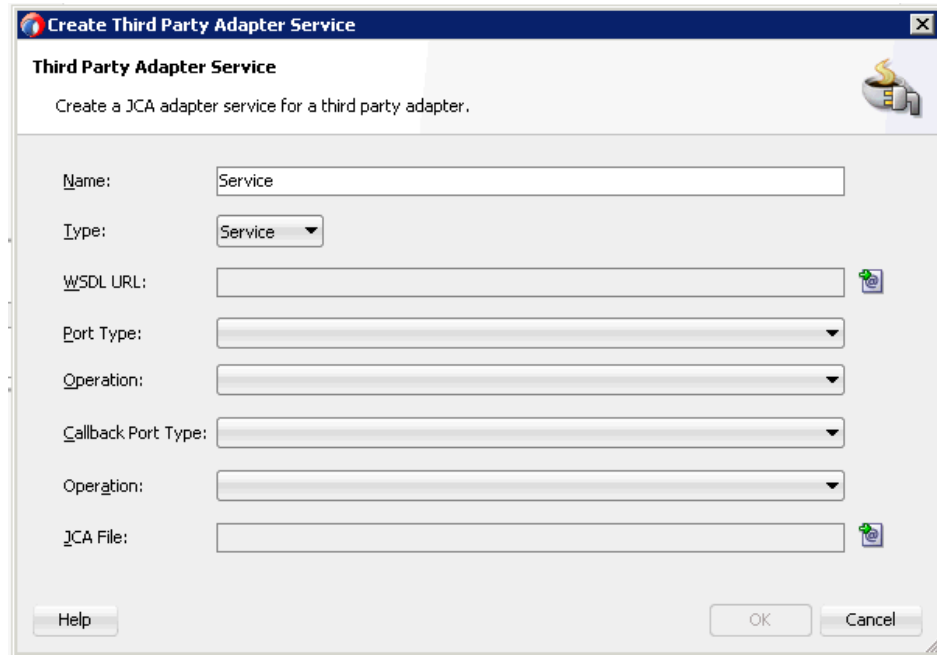


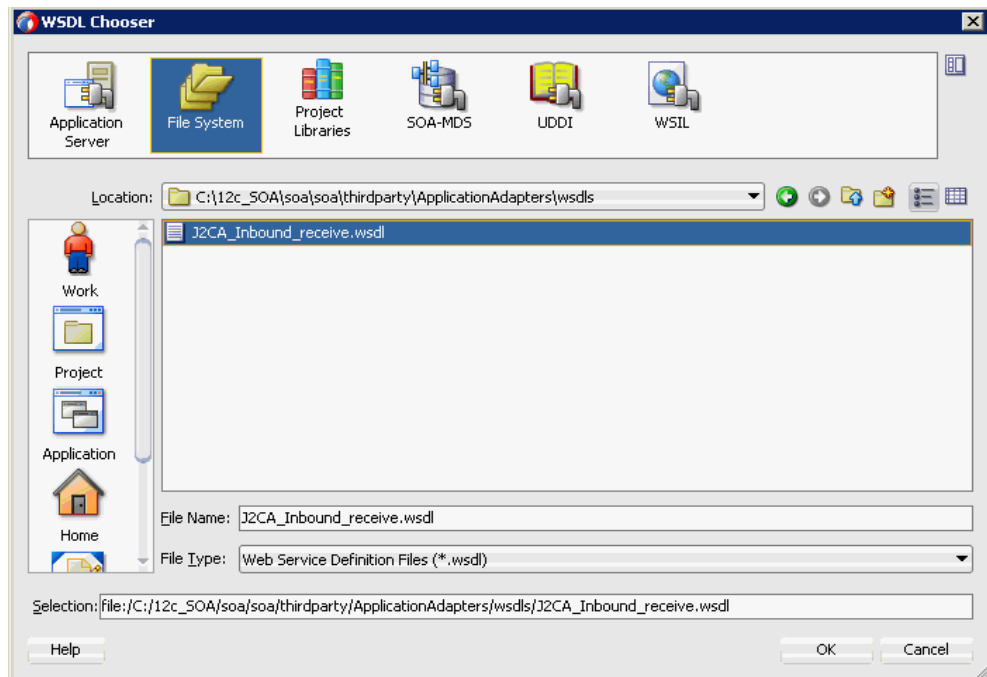
図 4-60 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

図 4-60 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



2. 「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)。
3. 「WSDL URL」フィールドの右の既存の WSDL を検索アイコンをクリックします。
 図 4-61 に示すように、「WSDL チューザ」ダイアログが表示されます。

図 4-61 「WSDL チューザ」ダイアログ

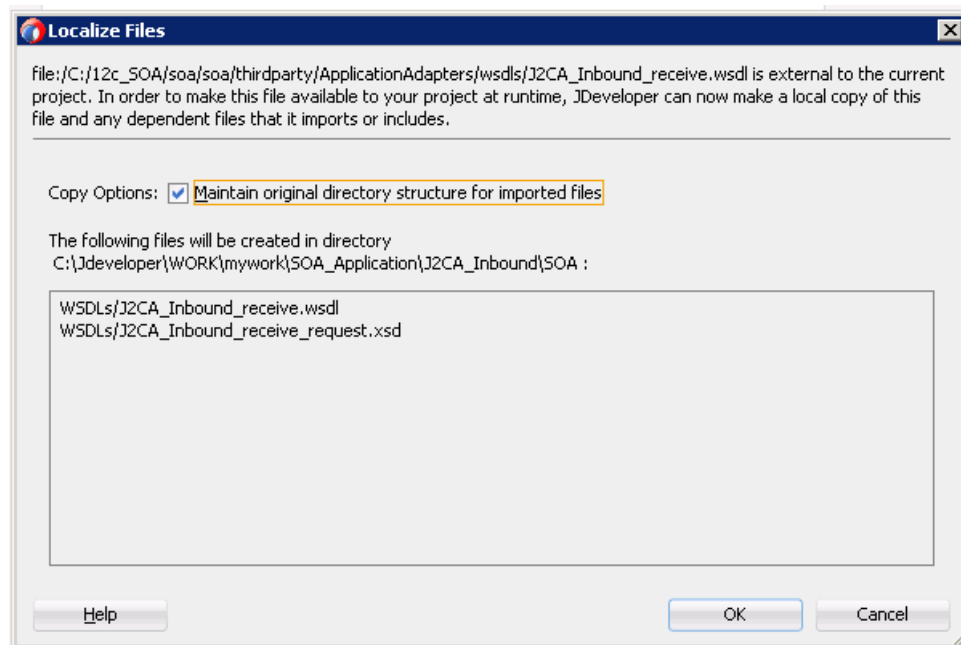


4. 次のディレクトリまでブラウズし、インバウンド WSDL ファイルを選択します。
 <ADAPTER_HOME>\wsdls

5. 「OK」をクリックします。

図 4-62 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 4-62 「ファイルのローカライズ」ダイアログ



6. 「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。「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。

7. 「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。

「XQuery 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。

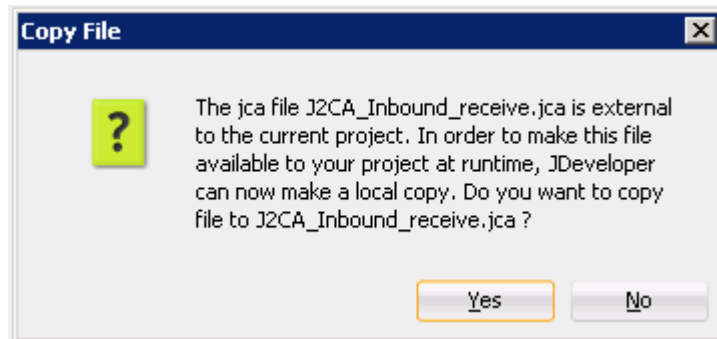
8. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。

<ADAPTER_HOME>\wsdl

9. 「OK」をクリックします。

図 4-63 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 4-63 「ファイルのコピー」 確認メッセージ

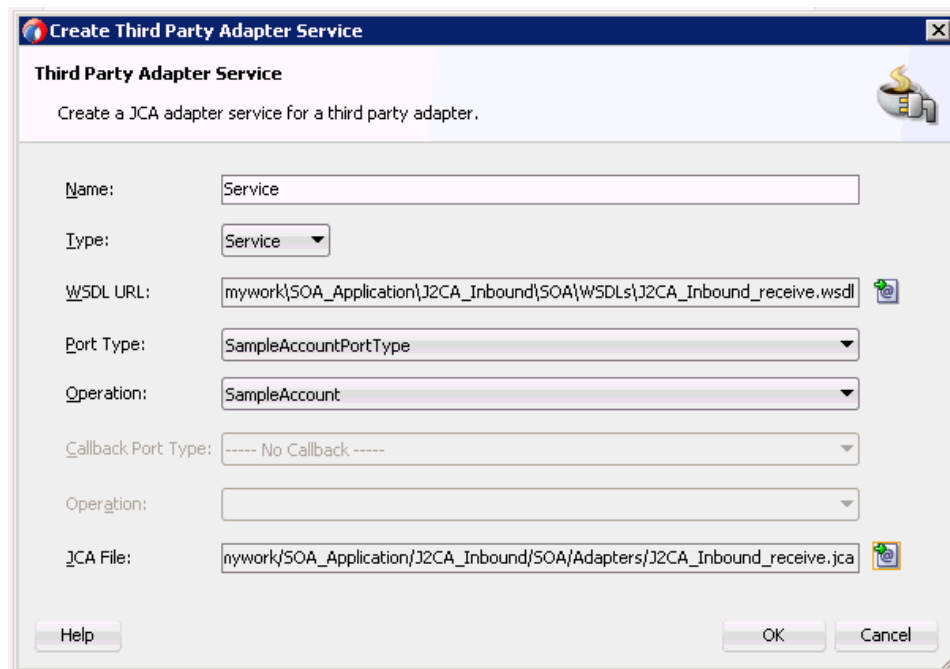


10. 「はい」 をクリックします。

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 4-64 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 4-64 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



11. 「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、「公開されたサービス」ペインに表示されます。

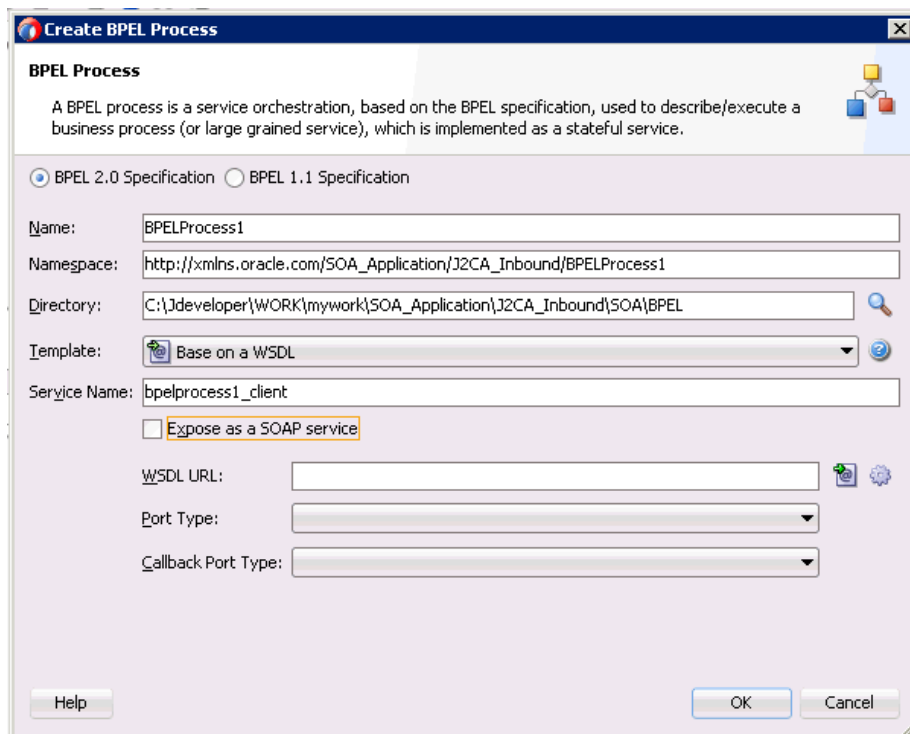
これで、インバウンド BPEL プロセス・コンポーネントを構成する準備ができました。

4.5.3.2 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。

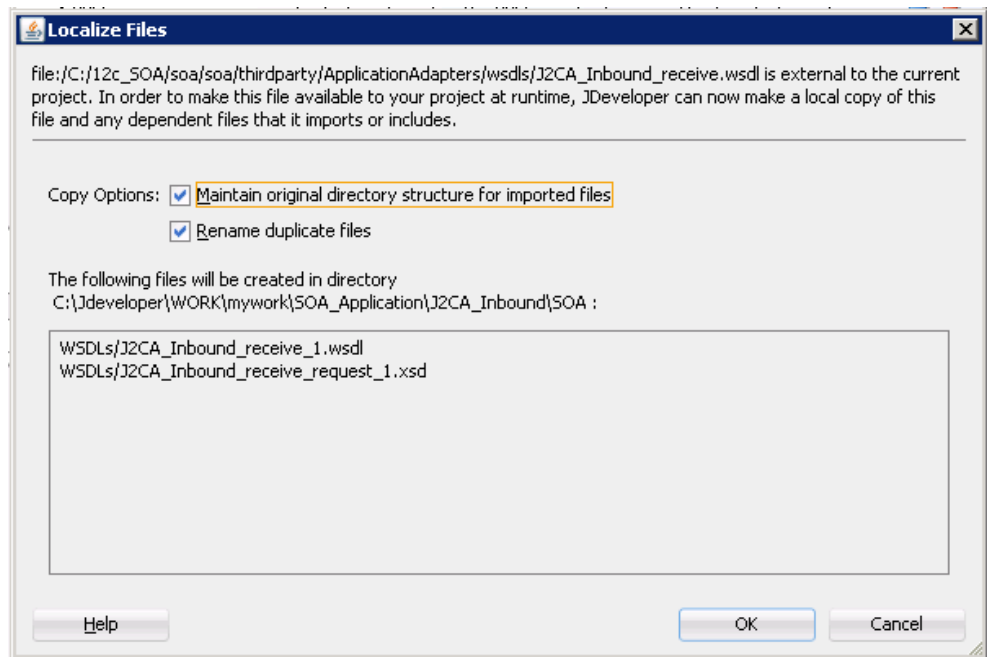
1. 「サービス・コンポーネント」ペインから「**BPEL プロセス**」コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップします。
☒ 4-65 に示すように、「BPEL プロセスの作成」ダイアログが表示されます。

図 4-65 「BPEL プロセスの作成」ダイアログ



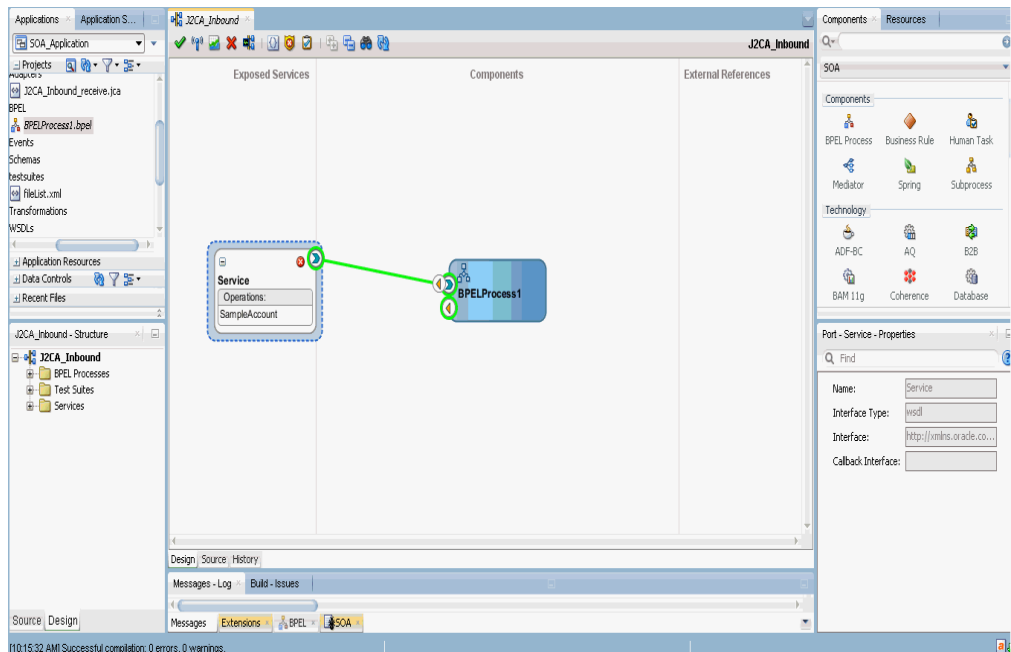
2. 「名前」フィールドに、新規のインバウンド BPEL プロセス・コンポーネントの名前を入力するか、デフォルト値のままにします。
「BPEL 2.0 仕様」オプションがデフォルトで選択されています。
3. 「テンプレート」リストで「**WSDL に基づく**」を選択します。
4. 「**SOAP サービスとして公開**」チェック・ボックスの選択を解除します。
5. 「WSDL URL」フィールドの右の**既存の WSDL を検索**アイコンをクリックします。
「WSDL チューザ」ダイアログが表示されます。
6. 次のディレクトリからインバウンド WSDL ファイルを選択します。
`<ADAPTER_HOME>\wsdl1s`
7. 「**OK**」をクリックします。
☒ 4-66 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 4-66 「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。

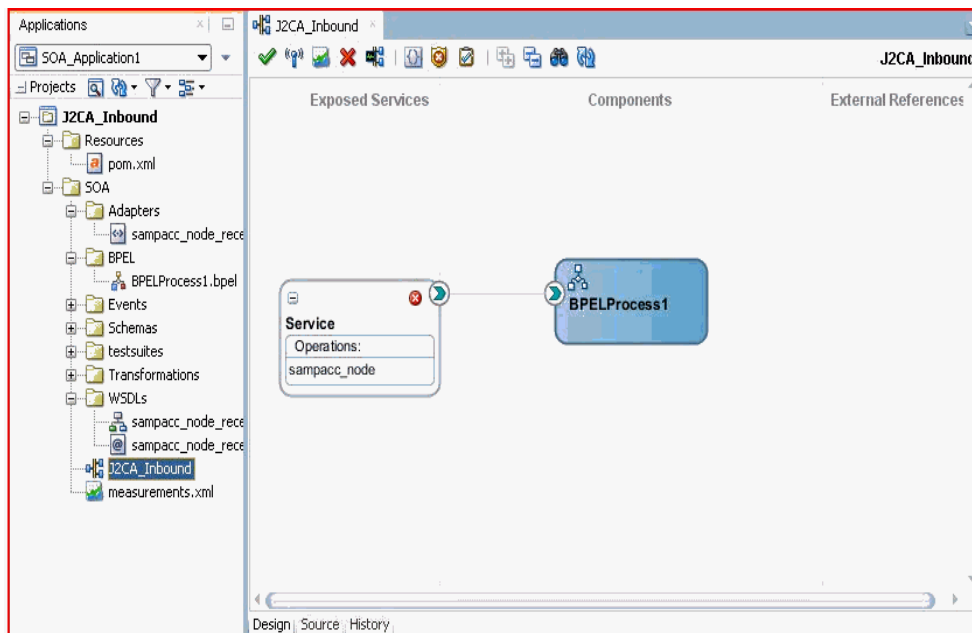
図 4-67 接続の作成



11. 図 4-67 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・コンポーネントの間の接続を作成します。

12. 左ペインの「J2CA_Outbound」をダブルクリックします。

図 4-68 「すべて保存」アイコン



13. 図 4-68 に示すように、メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規インバウンド BPEL プロセス・コンポーネントを保存します。

これで、BPEL インバウンド・プロセスをデプロイする準備ができました。

4.5.3.3 12c のデプロイメントに関する既知の問題の調整

12c のデプロイメントに関する既知の問題の調整方法の詳細は、4-26 ページ [4.4.3.3 項「12c のデプロイメントに関する既知の問題の調整」](#) を参照してください。

4.5.4 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. 左ペインでプロジェクト名を右クリックし、「デプロイ」を選択して、「J2CA_Inbound」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・サーバーへのデプロイ」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。

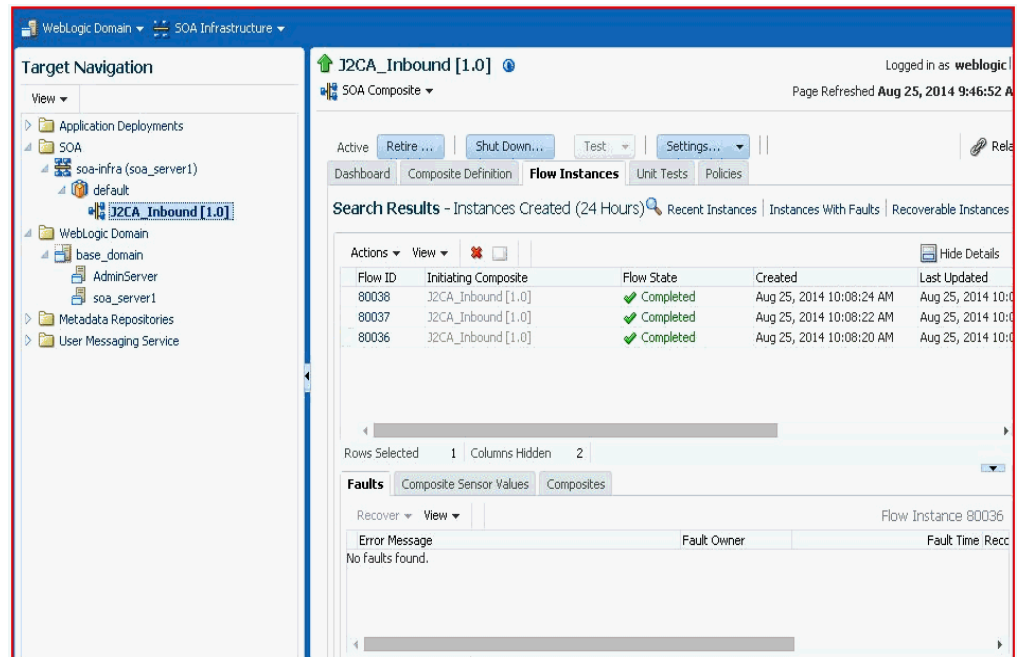
7. プロジェクトの使用可能なすべてのデプロイメント情報をレビューおよび確認し、「終了」をクリックします。

プロセスが正常にデプロイされます。

詳細は、4-28 ページ 4.4.4 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。

図 4-69 に示すように、イベント・メッセージが Siebel を介してトリガーされると、Oracle Enterprise Manager Console 側でインスタンスが正常に受信されます。

図 4-69 受信されたインスタンス



4.5.5 Siebel でのイベントのトリガー

この項では、Siebel でのイベントのトリガー方法と Oracle Application Adapter for Siebel を使用したイベントの統合の確認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。

- 4.5.5.1 項「イベントのランタイム統合をテストする Siebel イベントのトリガー」
- 4.5.5.2 項「Siebel 7.8 でイベントをトリガーしてイベントのランタイム統合をテストする」
- 4.5.5.3 項「Siebel 8.0 でイベントをトリガーしてイベントのランタイム統合をテストする」
- 4.5.5.4 項「結果の確認」

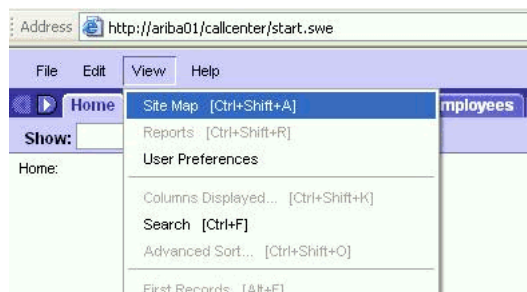
4.5.5.1 イベントのランタイム統合をテストする Siebel イベントのトリガー

Siebel でイベントをトリガーするには次を行います。

1. 図 4-70 に示すように、ブラウザに次の URL を入力して Siebel Call Center を起動します。

`http://host name/callcenter/start.swe`

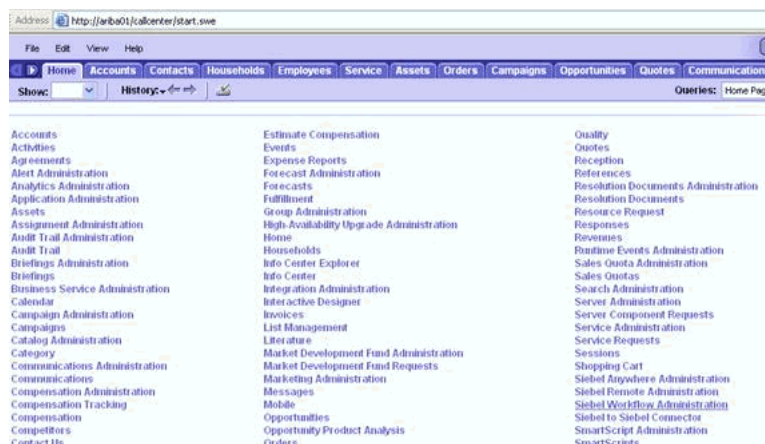
図 4-70 Siebel Call Center の「表示」メニューで選択されたサイト・マップ・オプション



2. 「表示」をクリックし、リストからサイト・マップを選択します。

図 4-71 に示すように、サイト・マップ・ビューが表示されます。

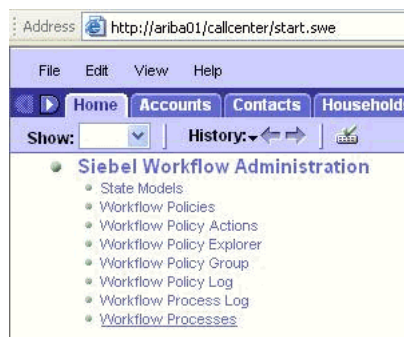
図 4-71 サイト・マップ・ビュー



3. Siebel ワークフロー管理をクリックします。

図 4-72 に示すように、Siebel ワークフロー管理ページが表示されます。

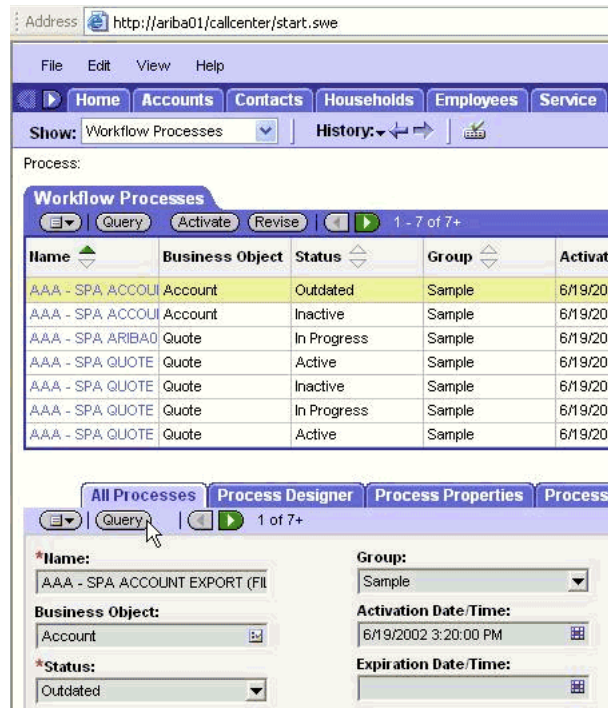
図 4-72 Siebel ワークフロー管理ページ



4. ワークフロー・プロセスをクリックします。

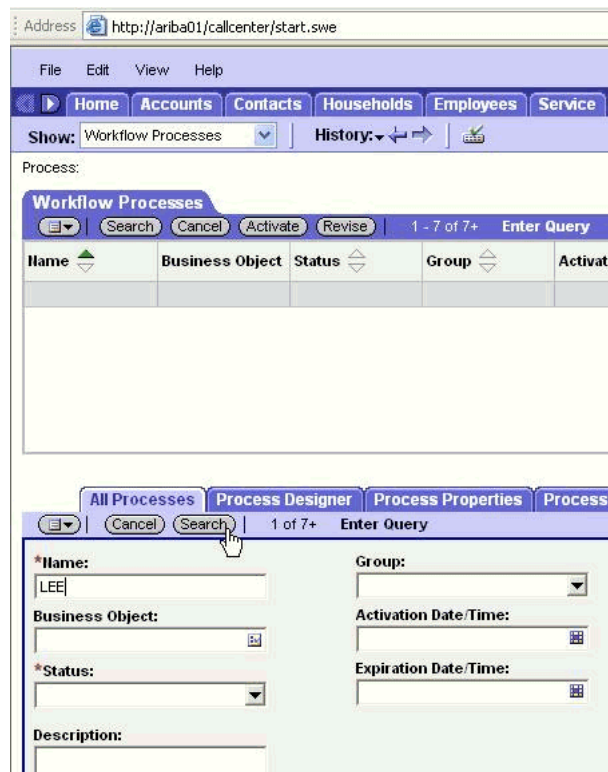
図 4-73 に示すように、ワークフロー・プロセス・ページが表示されます。

図 4-73 ワークフロー・プロセス・ページ



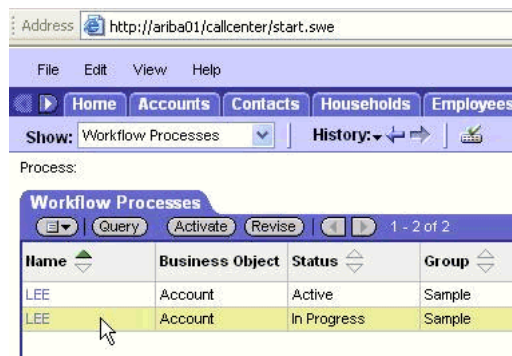
5. 図 4-74 に示すように、Siebel イベントのトリガーに必要なワークフローを検索するため「問合せ」をクリックします。

図 4-74 ワークフロー・プロセス・ページの「検索」ボタン



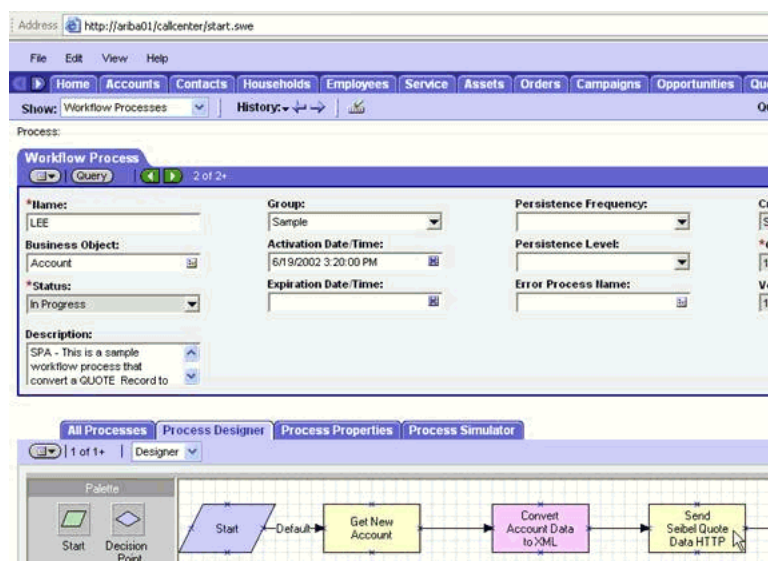
- 図 4-75 に示すように、Siebel ワークフローの名前を入力し、「検索」をクリックします。

図 4-75 ワークフロー・プロセス・ページ



- 図 4-76 に示すように、ワークフローを選択します。

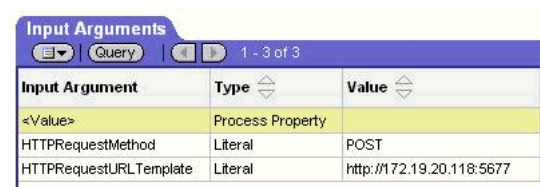
図 4-76 プロセス・デザイナー・タブ



- プロセス・デザイナー・タブをクリックし、「Send Siebel Quote Data HTTP」ワークフロー要素をダブルクリックします。

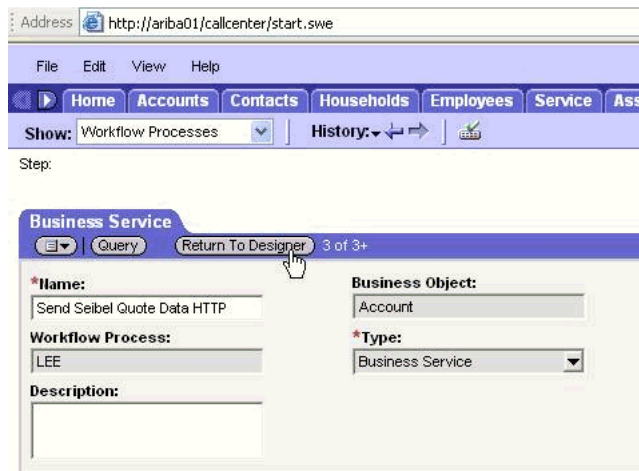
図 4-77 に示すように、「入力引数」タブが表示されます。

図 4-77 「入力引数」タブ



- HTTPRequestURLTemplate 入力引数の IP アドレスとポートを入力します。
- 図 4-78 に示すように、デザイナーに戻るをクリックします。

図 4-78 「ビジネス・サービス」タブのデザイナーに戻るボタン



11. 図 4-79 に示すように、プロセス・シミュレータ・タブをクリックします。

図 4-79 プロセス・シミュレータ・タブ

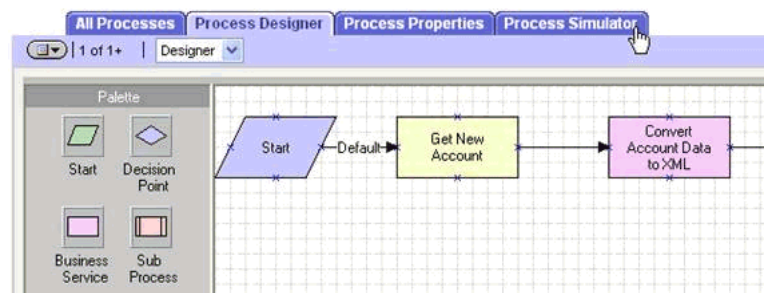
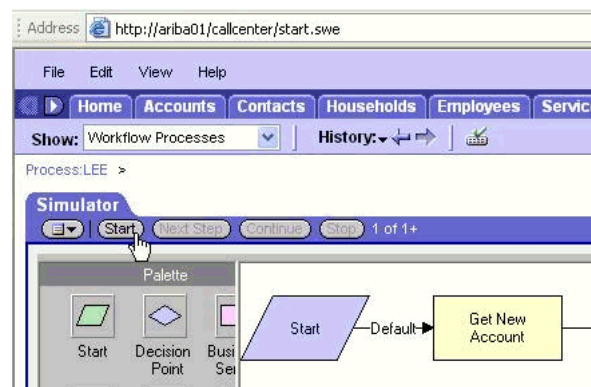


図 4-80 に示すように、「シミュレータ」タブが表示されます。

図 4-80 「シミュレータ」タブ



12. 「開始」、「続行」をクリックして Siebel イベントのトリガー・プロセスを完了します。

4.5.5.2 Siebel 7.8 でイベントをトリガーしてイベントのランタイム統合をテストする

Siebel 7.8 でイベントをトリガーするには次を行います。

1. 次のパラメータを使用して Siebel Tools 7.8 にログインします。
Username = sadmin
Password = sadmin
2. 図 4-81 に示すように、「接続先」リストから「サーバー」を選択して「OK」をクリックします。

図 4-81 Siebel Tools 7.8 のログイン・ペイン

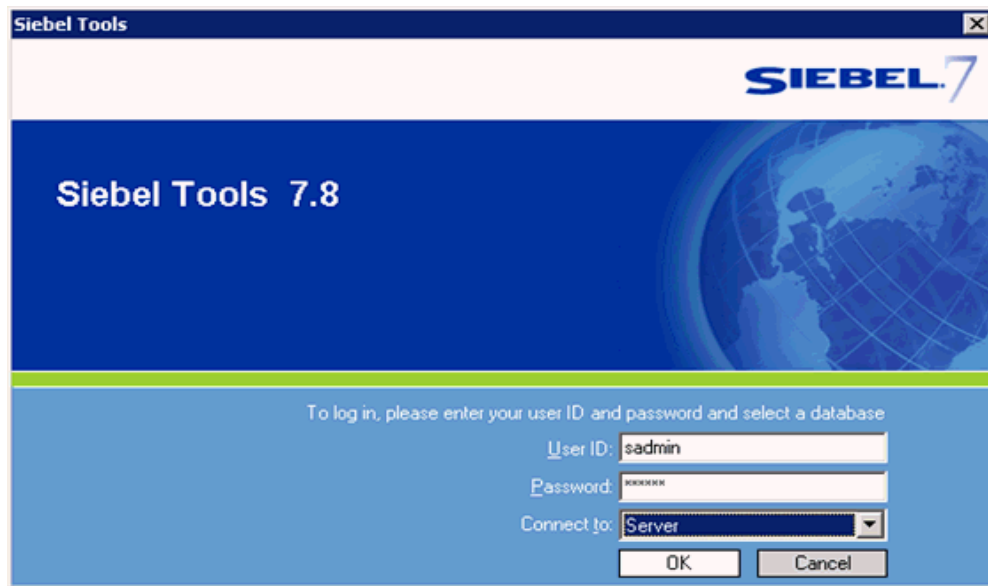


図 4-82 に示すように、Siebel Tools 7.8 にログインします。

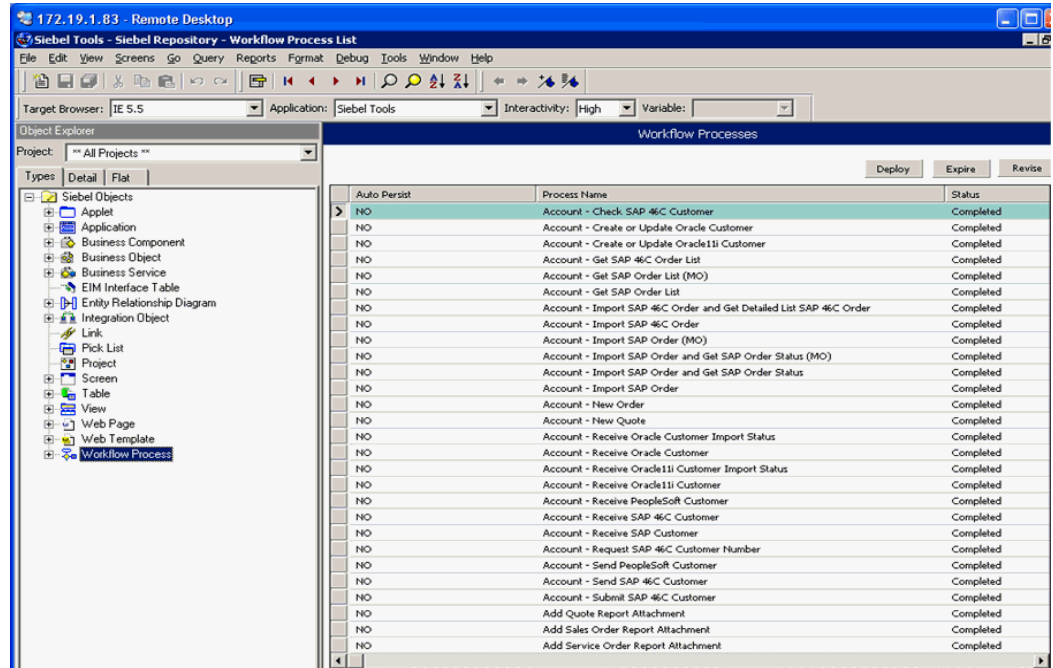
図 4-82 Siebel Tools 7.8 の「起動」ペイン



3. 左ペインのワークフロー・プロセスをクリックします。

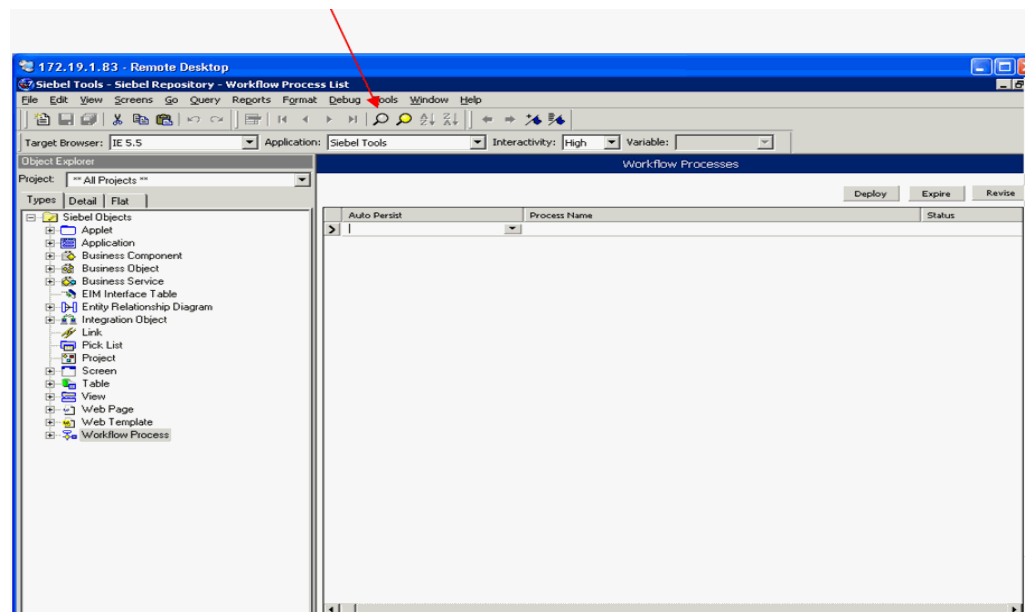
図 4-83 に示すように、ワークフロー・プロセス・ペインが表示されます。

図 4-83 ワークフロー・プロセス・ペイン



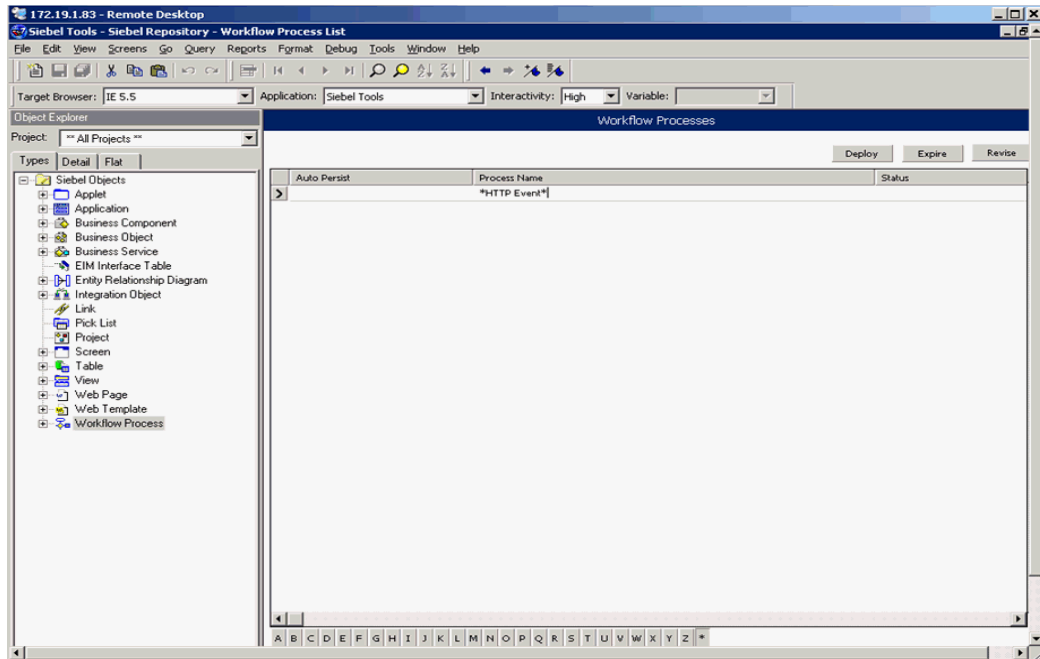
4. 図 4-84 に示すように、白く光っている「新規問合せ」拡大鏡ツール・アイコンをクリックします。

図 4-84 「新規問合せ」拡大鏡ツール・アイコン



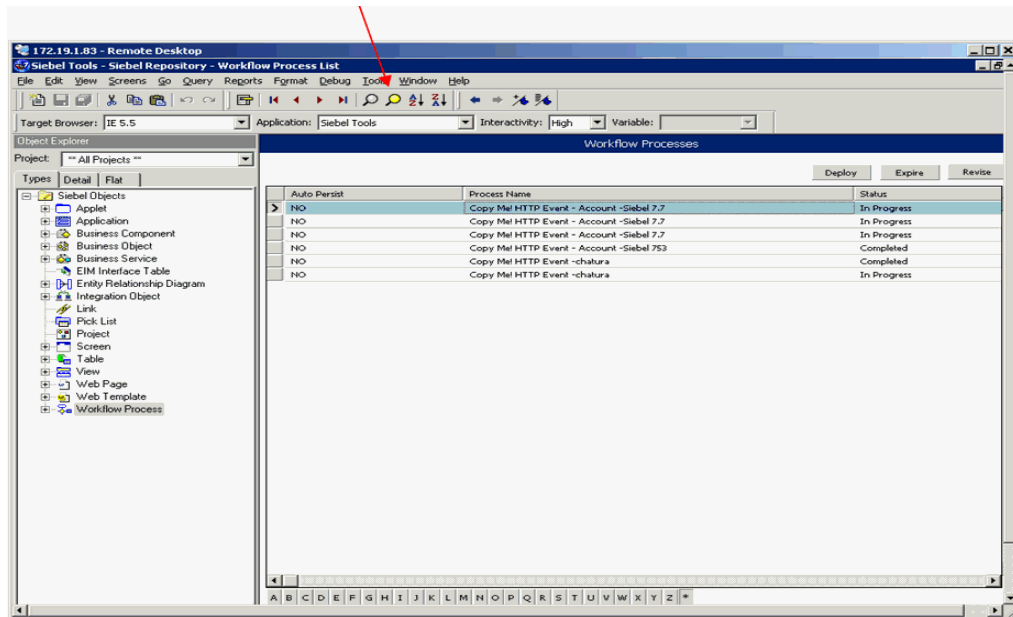
5. 図 4-85 に示すように、「プロセス名」フィールドに名前「*HTTP Event*」を入力します。

図 4-85 「プロセス名」 フィールド



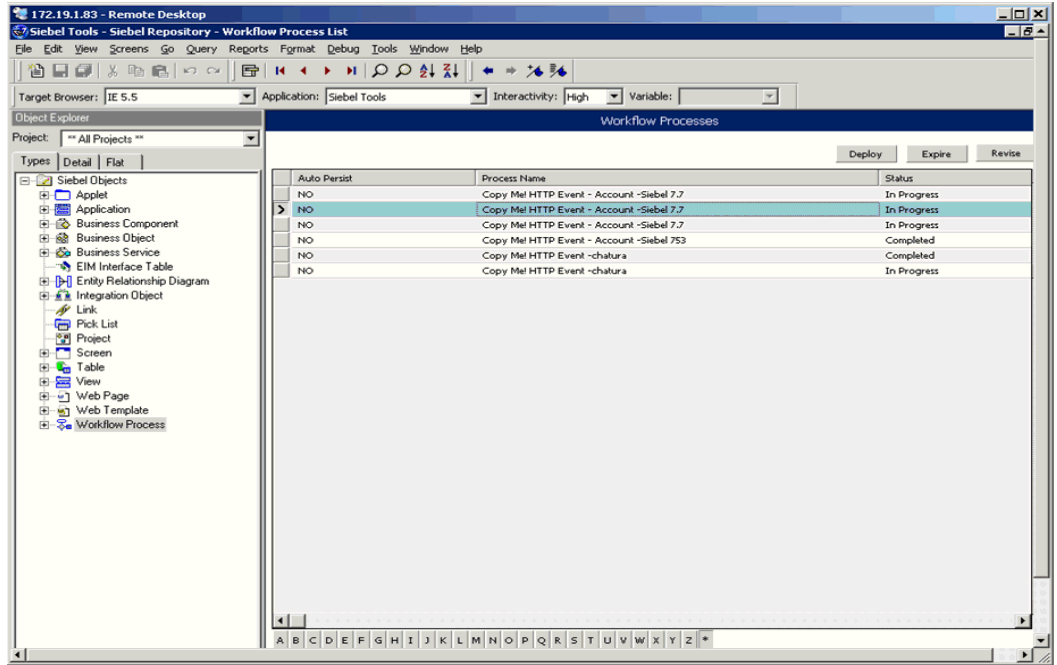
6. 図 4-86 に示すように、黄色く光っている拡大鏡ツール・アイコンをクリックします。

図 4-86 黄色い拡大鏡ツール・アイコン



7. 次の行をクリックします。「Copy Me!HTTP Event - Account -Siebel 7.7」 (図 4-87 を参照)。

図 4-87 Copy Me!HTTP Event - Account -Siebel 7.7



8. 図 4-88 に示すように、選択したプロセスの隣の矢印を右クリックし、ワークフロー・プロセスの編集を選択します。

図 4-88 ワークフロー・プロセスの編集オプション

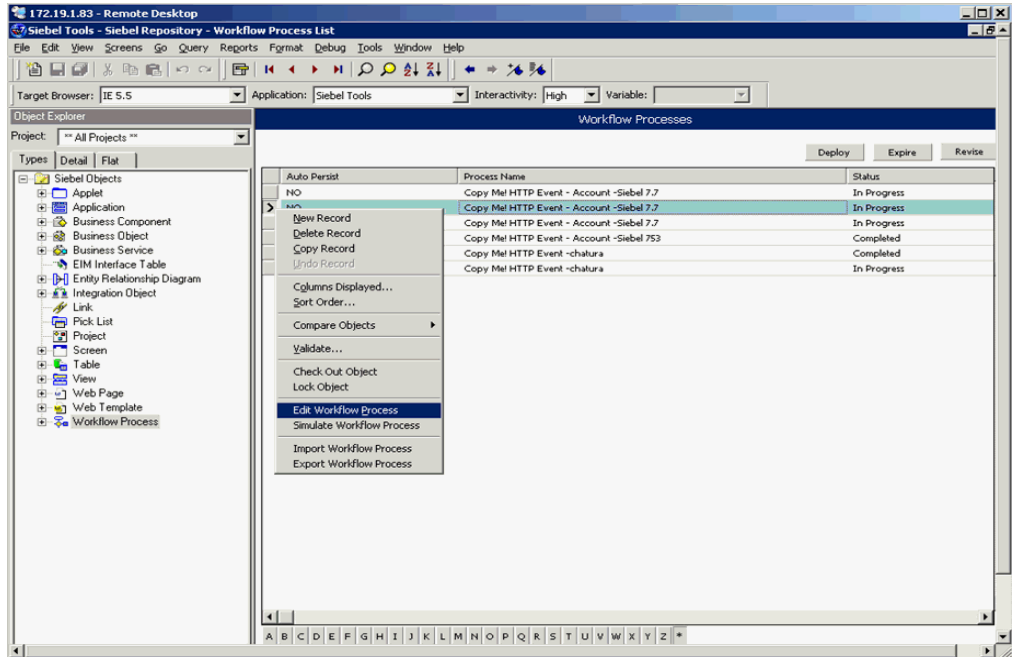
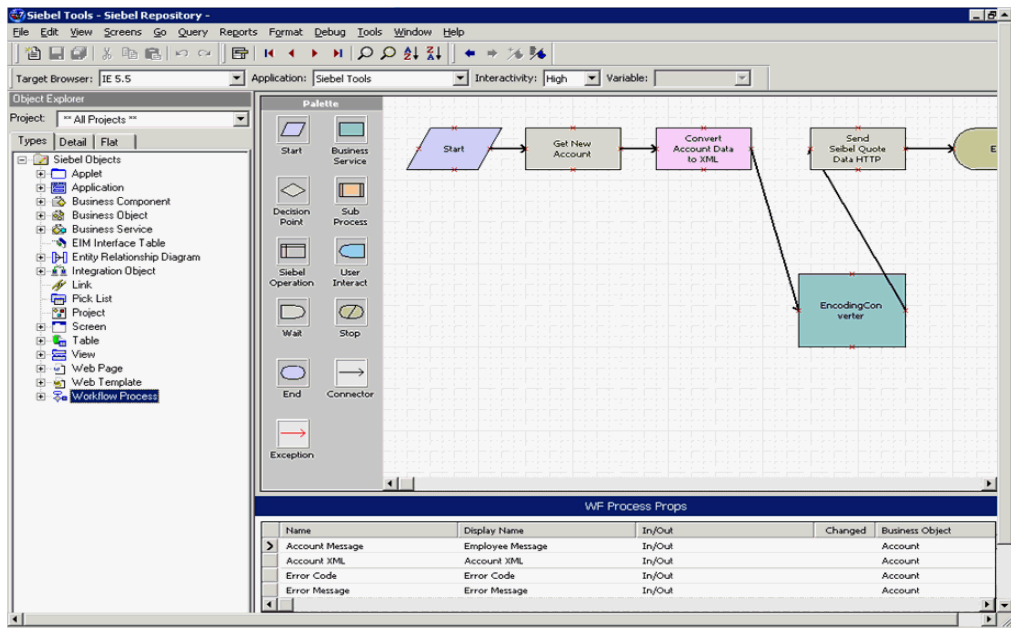


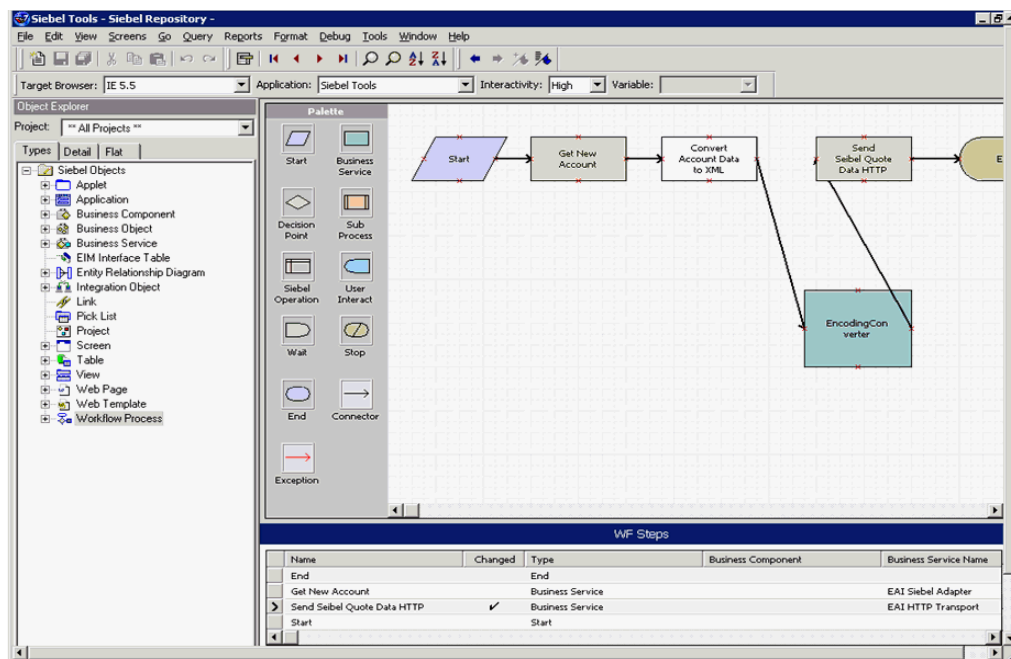
図 4-89 に示すように、右ペインにダイアグラムが表示されます。

図 4-89 ワークフロー・プロセス・ダイアグラム



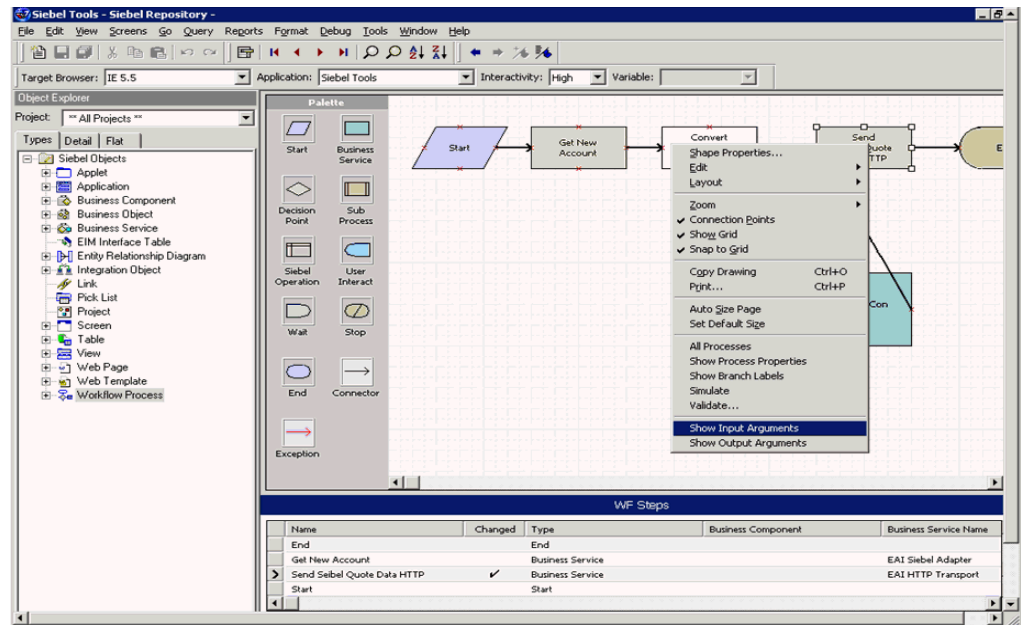
9. 図 4-90 に示すように、「Send Siebel Quote Data HTTP」というタイトルのダイアグラム・ボックスをクリックします。

図 4-90 Send Siebel Quote Data HTTP



10. 図 4-91 に示すように、「Send Siebel Quote Data HTTP」を右クリックして「入力引数」を選択します。

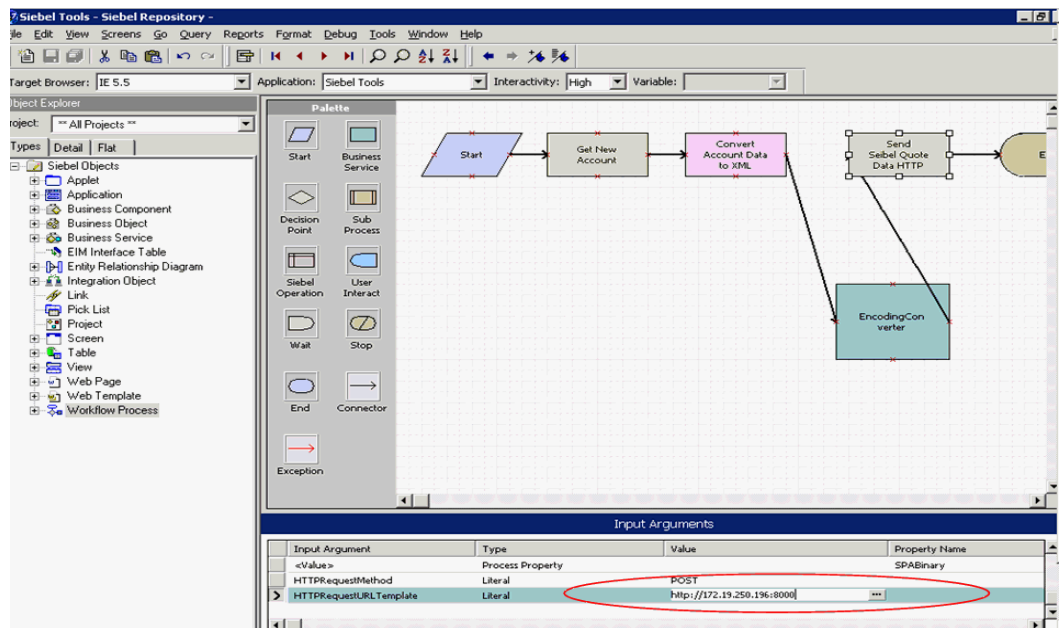
図 4-91 「入力引数」オプション



11. 図 4-92 に示すように、下のペインで次の URL を使用して「HTTPRequestURLTemplate」に値を入力します。

http://machineIP: portno

図 4-92 HTTPRequestURLTemplate の値



12. 図 4-93 に示すように、ダイアグラムを右クリックして「シミュレート」を選択します。

図 4-93 「シミュレート」オプション

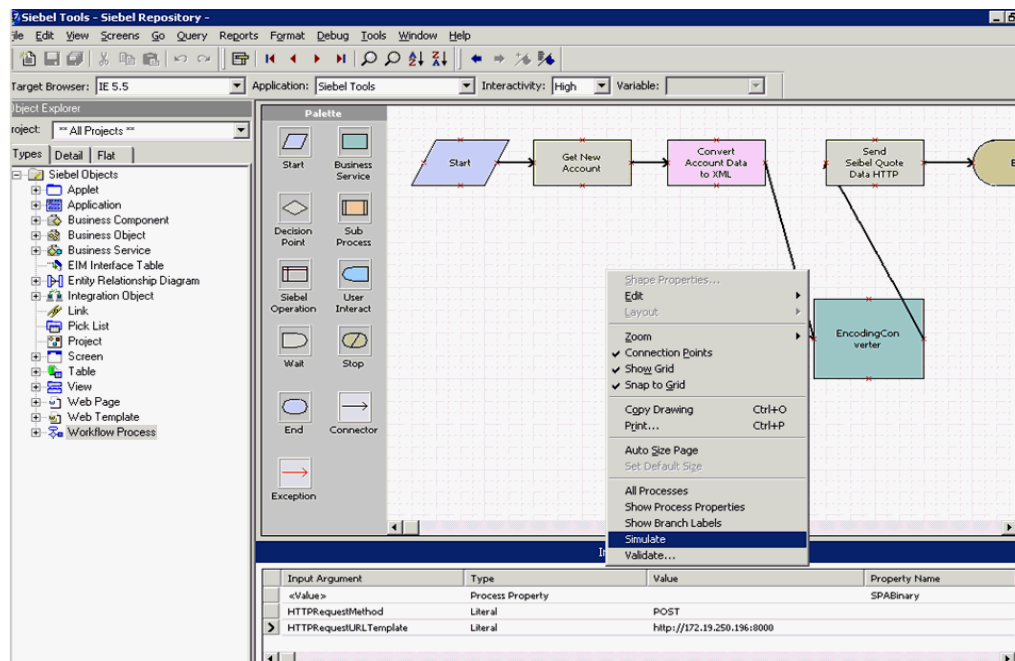
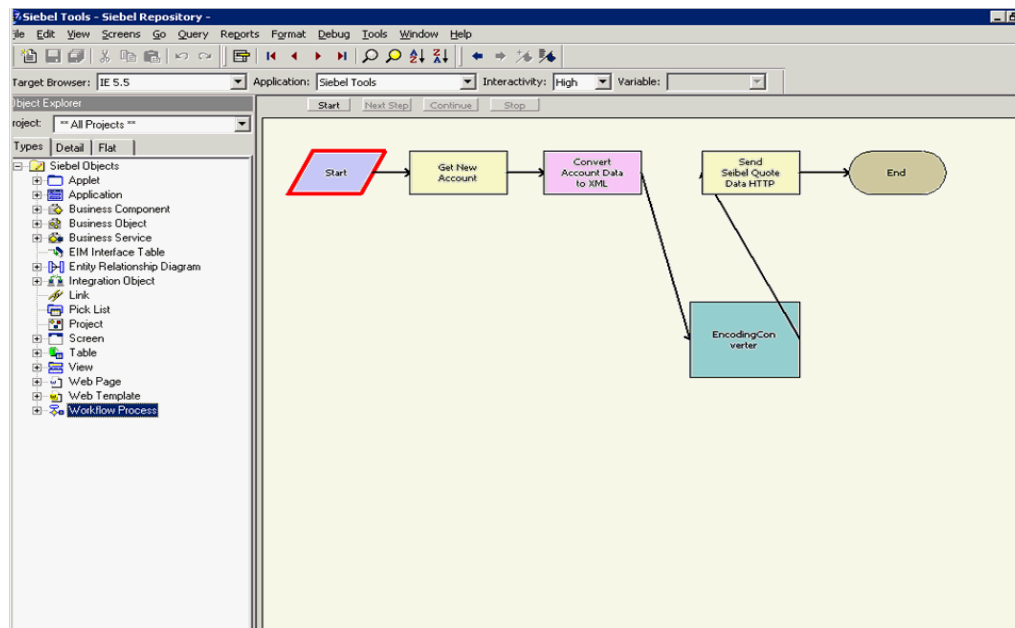


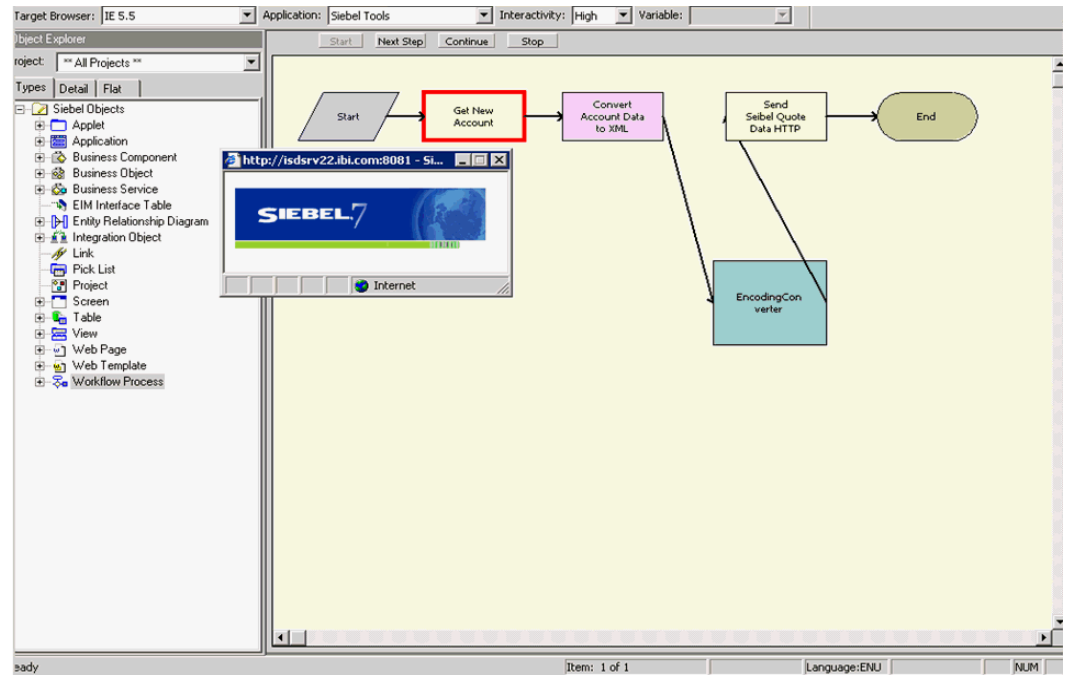
図 4-94 に示すように、リポジトリ・ダイアグラムが表示されます。

図 4-94 リポジトリ・ダイアグラム



13. 図 4-95 に示すように、「開始」をクリックし、表示される Siebel 7 ウィンドウを最小化します。

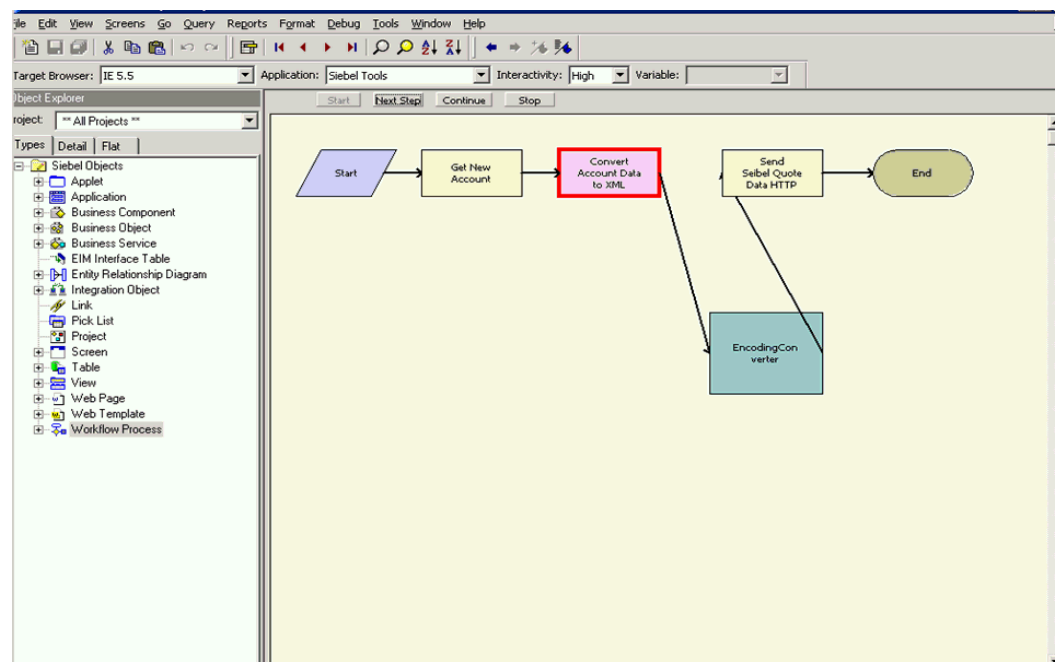
図 4-95 Siebel 7 ウィンドウ



14. 次のステップをクリックします。図 4-96 に示すように、「Convert Account Data to XML」が強調表示されます。

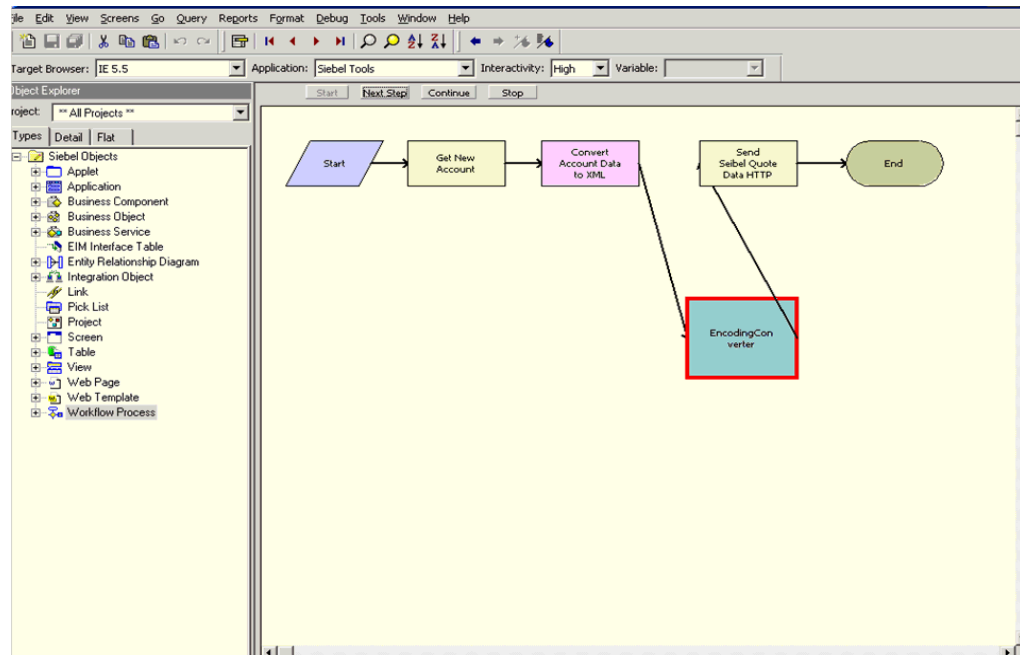
注意： それぞれのステップにおける各ダイアグラム図が赤色の外枠によって強調表示されます。

図 4-96 Convert Data to XML



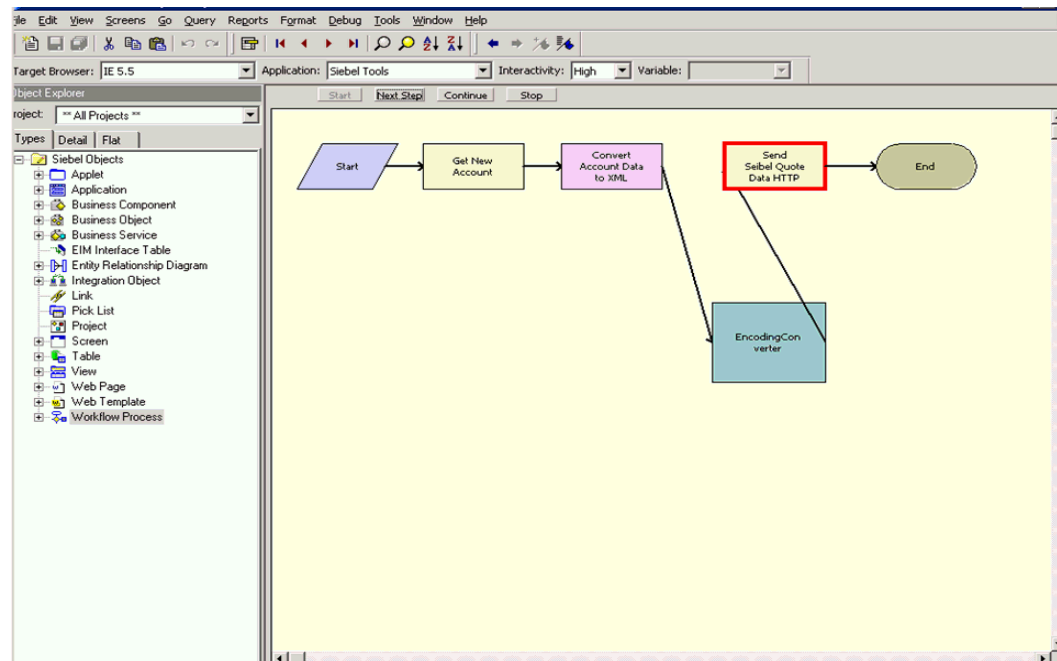
15. 次のステップをクリックします。図 4-97 に示すように、「Encoding Converter」が強調表示されます。

図 4-97 *Encoding Converter*



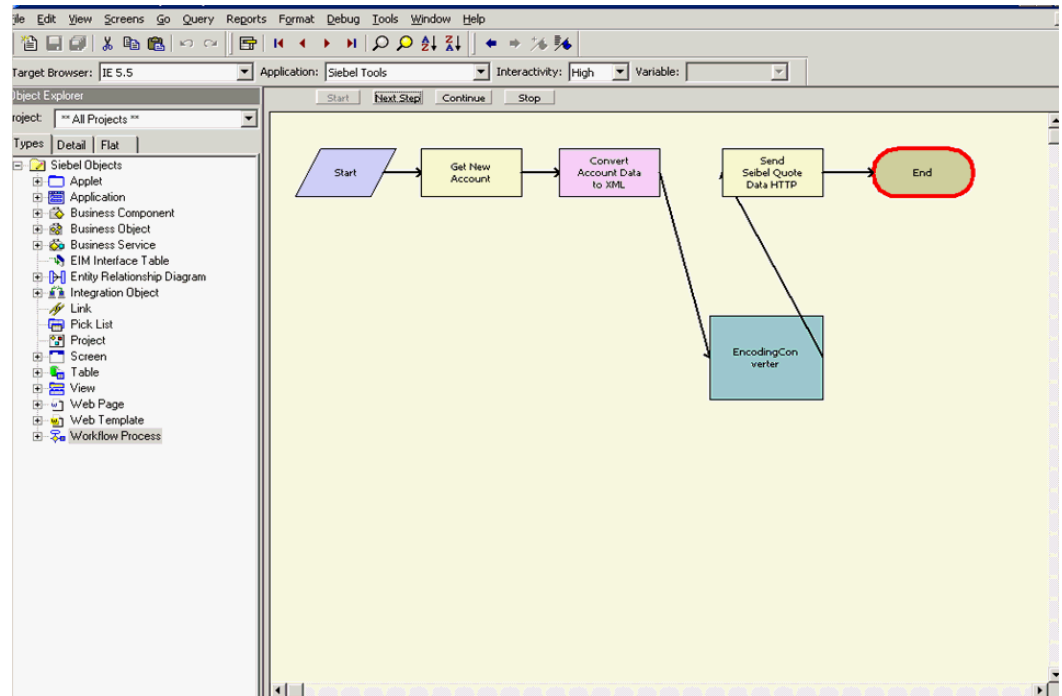
16. 次のステップをクリックします。図 4-98 に示すように、「Send Siebel Quote Data HTTP」が強調表示されます。

図 4-98 *Send Siebel Quote Data HTTP*



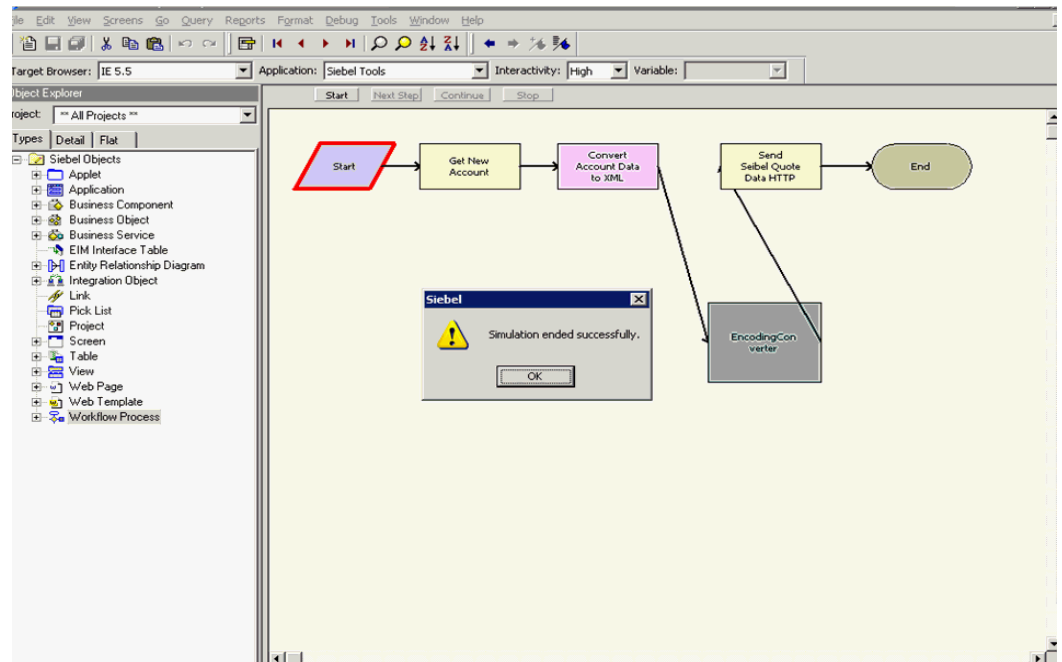
17. 次のステップをクリックします。図 4-99 に示すように、「終了」が強調表示されます。

図 4-99 終了



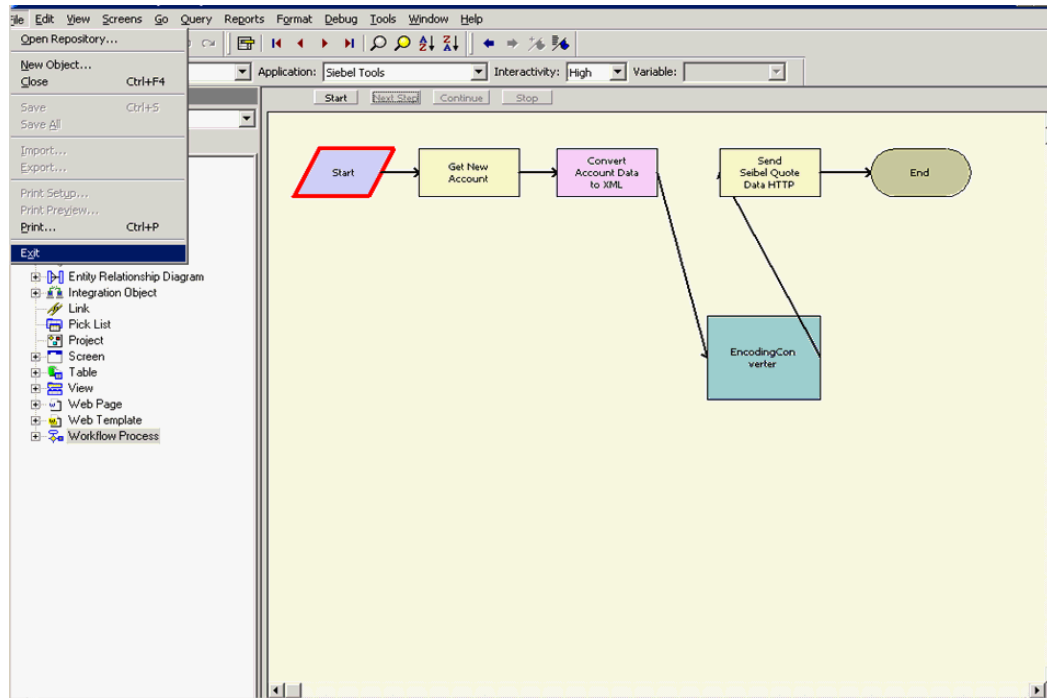
- 次のステップをクリックします。図 4-100 に示すように、トリガーが正常に終了したことを確認する成功を伝えるメッセージが表示されます。

図 4-100 成功を伝えるメッセージ



- 「OK」をクリックします。
- 図 4-101 に示すように、「ファイル」メニューを右クリックして「終了」を選択します。

図 4-101 「終了」オプション



4.5.5.3 Siebel 8.0 でイベントをトリガーしてイベントのランタイム統合をテストする

Siebel 8.0 でイベントをトリガーするには次を行います。

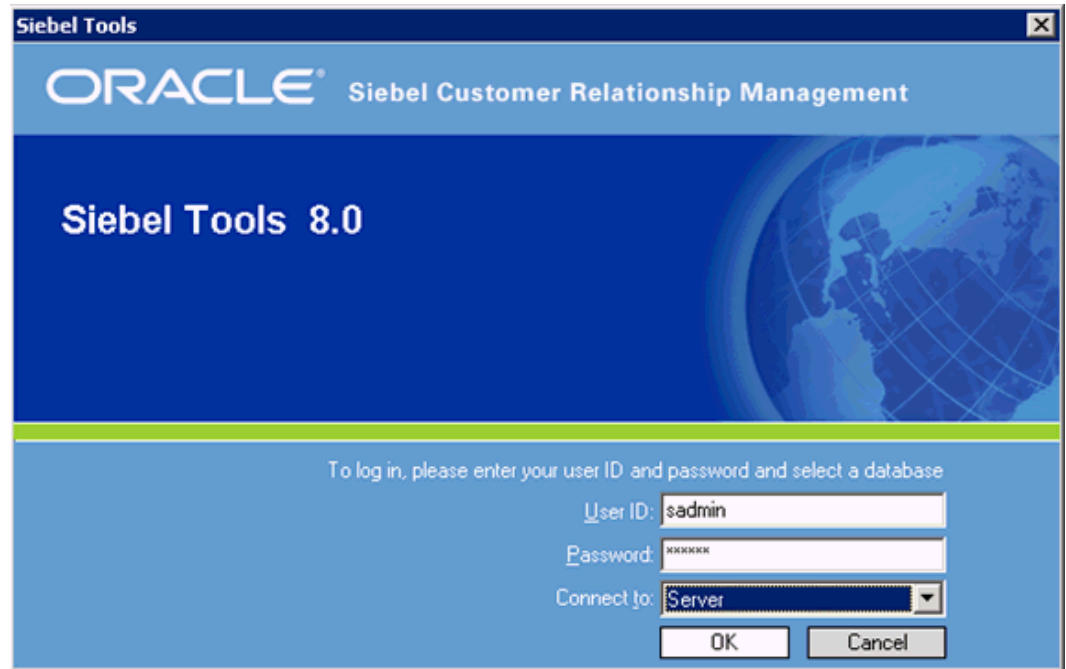
1. 次のパラメータを使用して Siebel Tools 8.0 にログインします。

Username = sadmin

Password = sadmin

2. 図 4-102 に示すように、「接続先」リストから「サーバー」を選択して「OK」をクリックします。

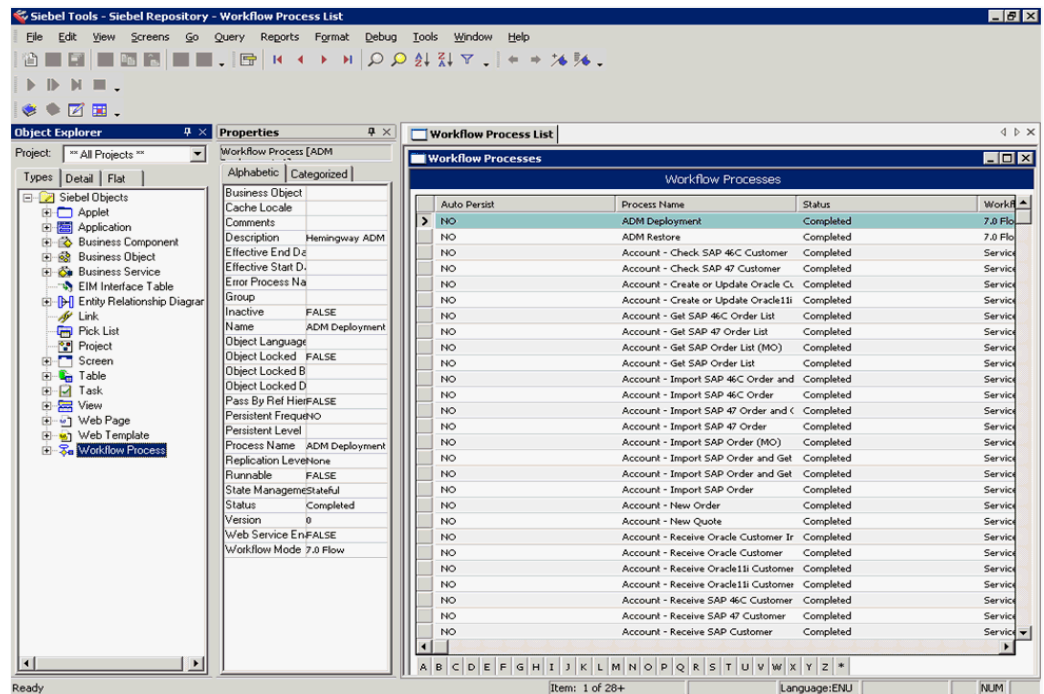
図 4-102 Siebel Tools 8.0 のログイン・ペイン



3. 左ペインのワークフロー・プロセスをクリックします。

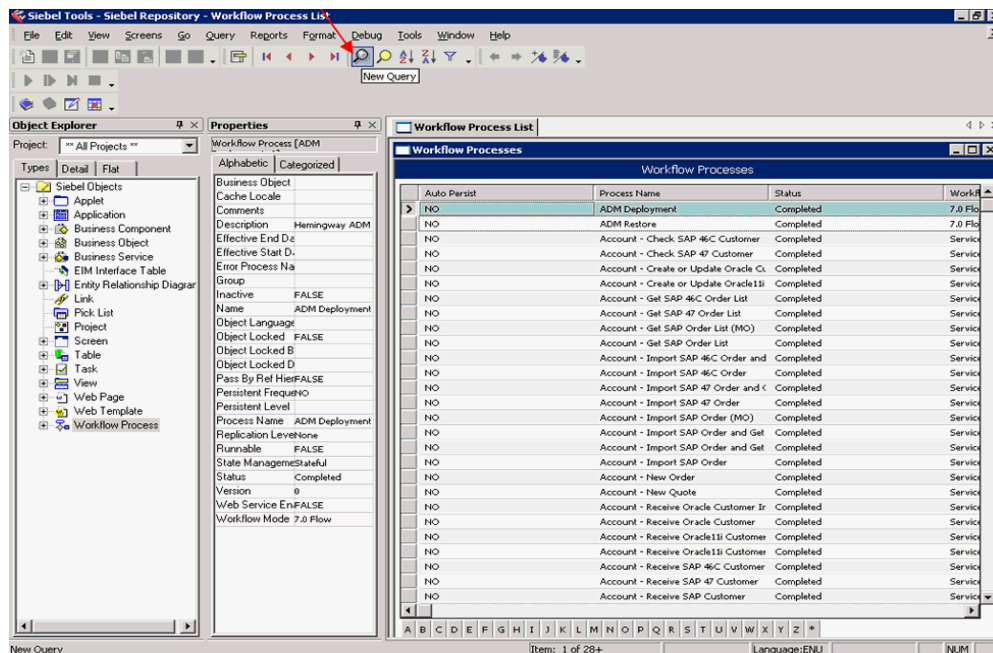
図 4-103 に示すように、右ペインにワークフロー・プロセス・リストが表示されます。

図 4-103 ワークフロー・プロセス・リスト



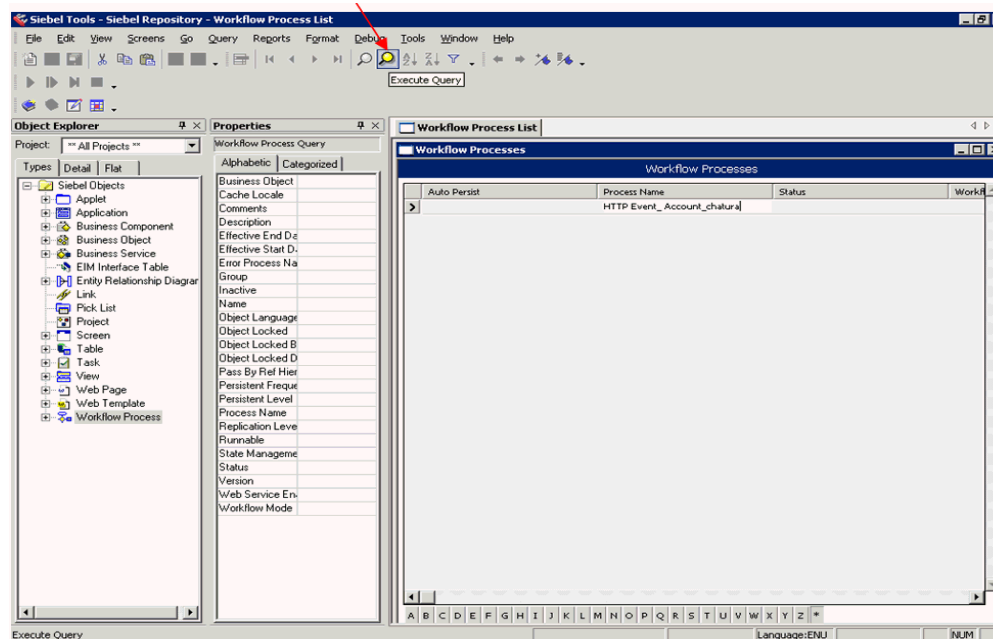
4. 図 4-104 に示すように、白く光っている「新規問合せ」拡大鏡ツール・アイコンをクリックします。

図 4-104 「新規問合せ」拡大鏡ツール・アイコン



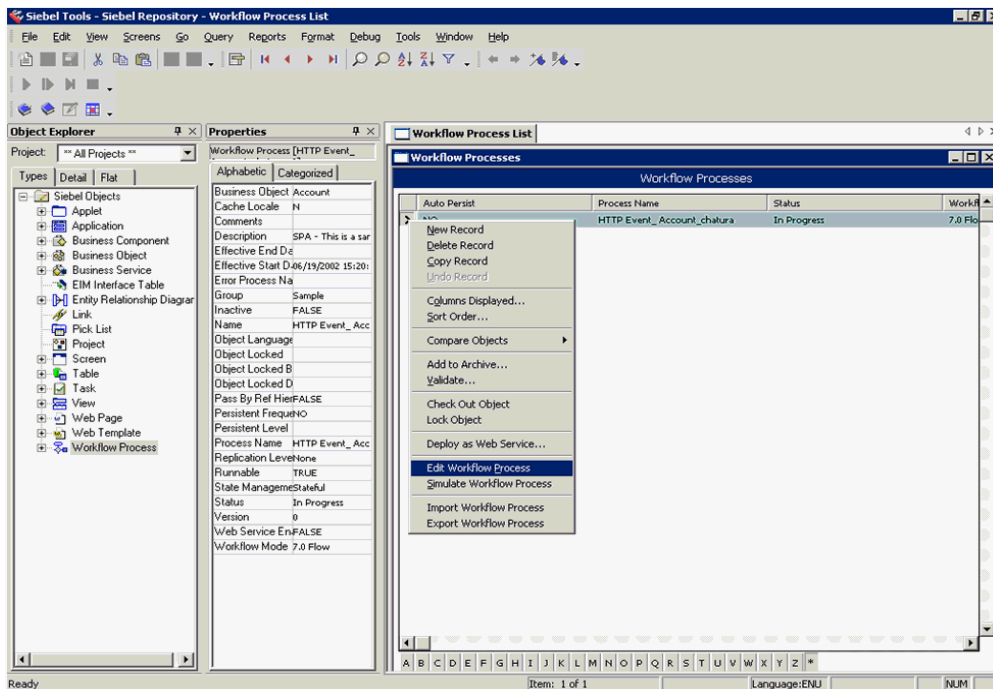
5. 図 4-105 に示すように、プロセス名「HTTP Event_Account_chatura」を入力し、黄色く光っている問合せ実行拡大鏡ツール・アイコンをクリックして問合せを実行します。

図 4-105 問合せ実行アイコン



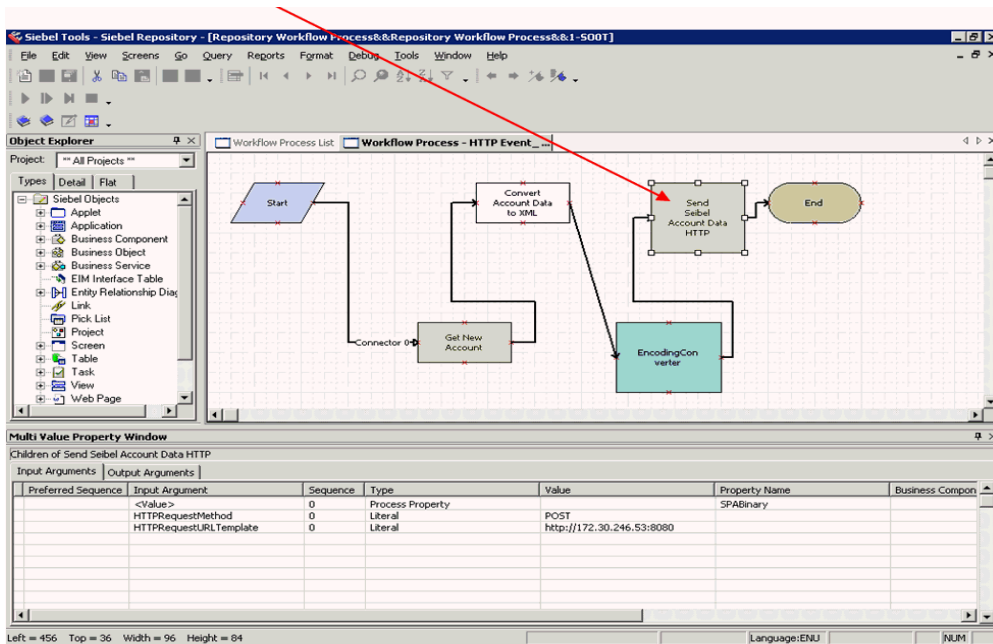
6. 図 4-106 に示すように、選択したプロセスの隣の矢印を右クリックし、ワークフロー・プロセスの編集を選択します。

図 4-106 ワークフロー・プロセスの編集オプション



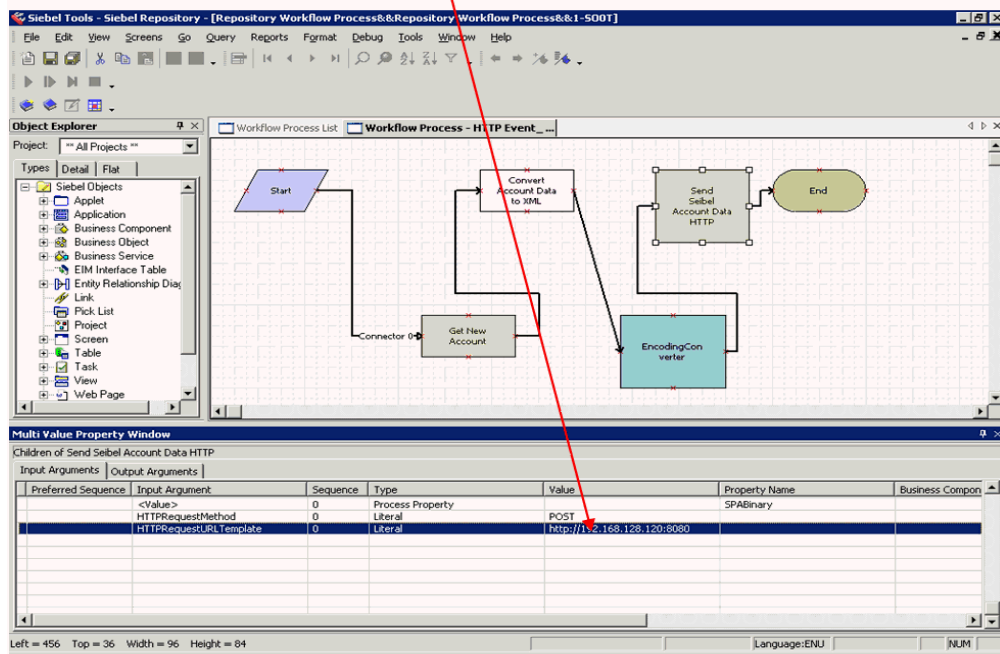
7. 図 4-107 に示すように、「Send Siebel Account Data HTTP」ボックスをクリックします。

図 4-107 「Send Siebel Account Data HTTP」ボックス



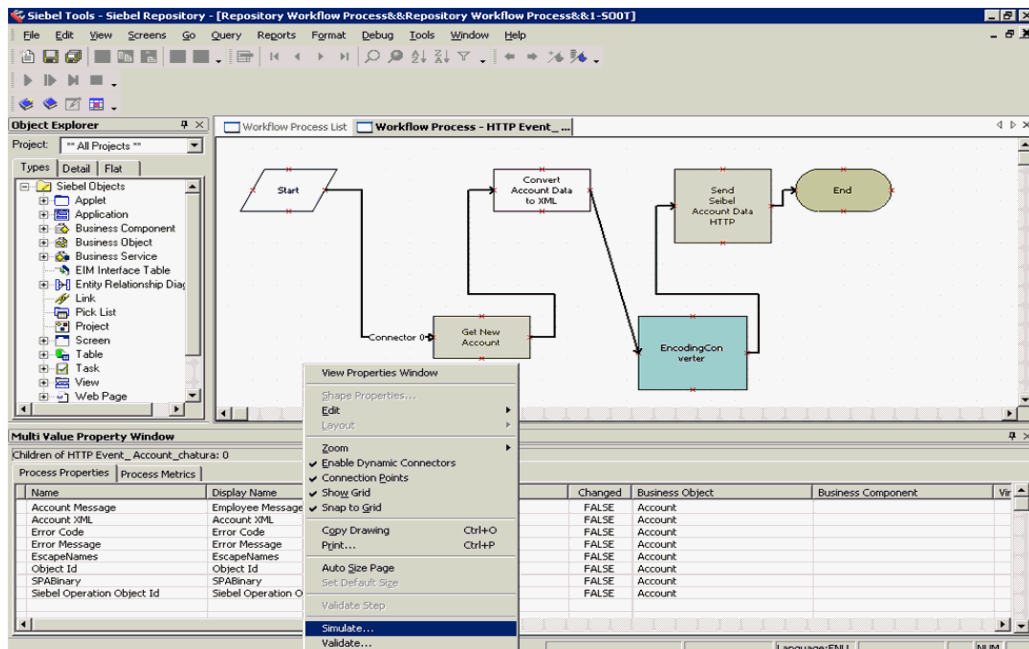
8. 図 4-108 に示すように、下にある複数値プロパティ・ウィンドウで、「HTTPRequestURLTemplate」の値に「http://machineIP:portno」を入力し、値を保存します。

図 4-108 HTTPRequestURLTemplate の値



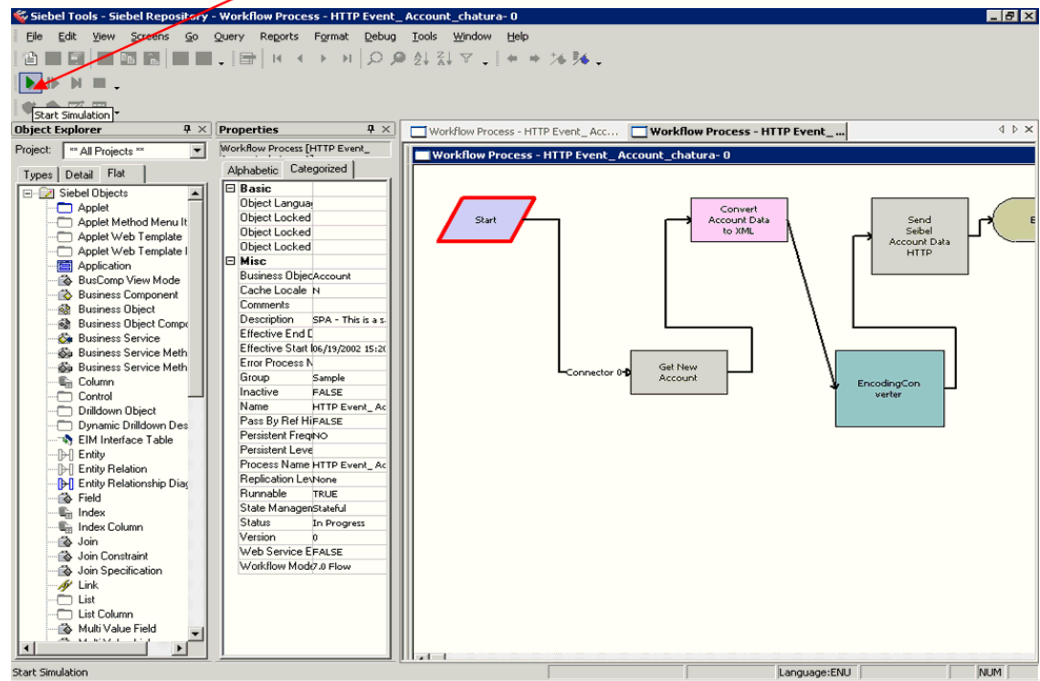
- 図 4-109 に示すように、ダイアグラムを右クリックして「シミュレート」を選択します。

図 4-109 シミュレート



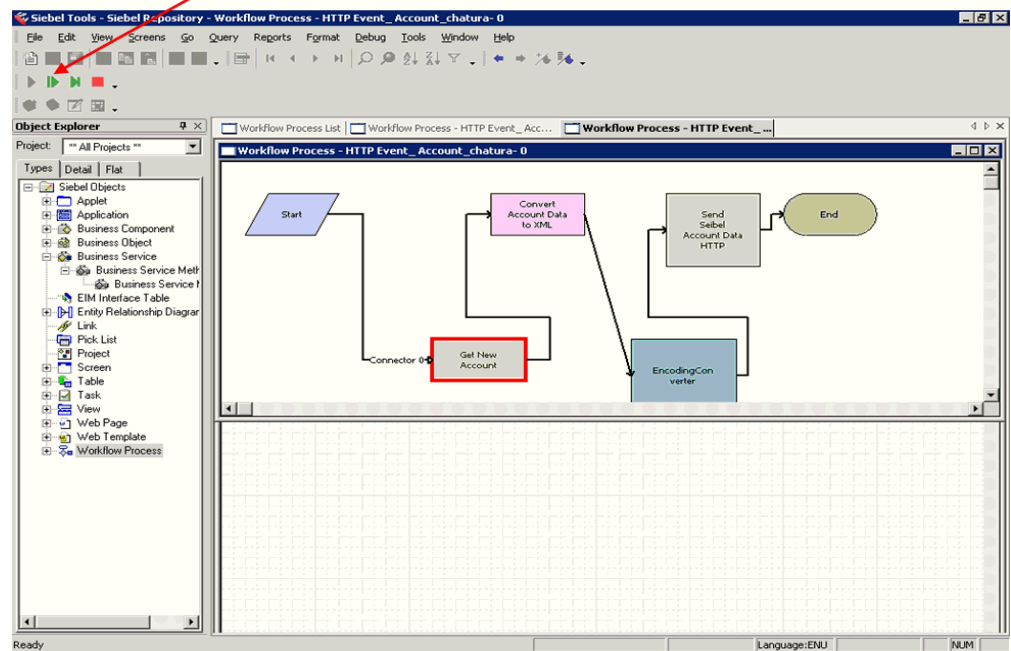
- 図 4-110 に示すように、「シミュレーションの開始」アイコンをクリックします。

図 4-110 シミュレーションの開始



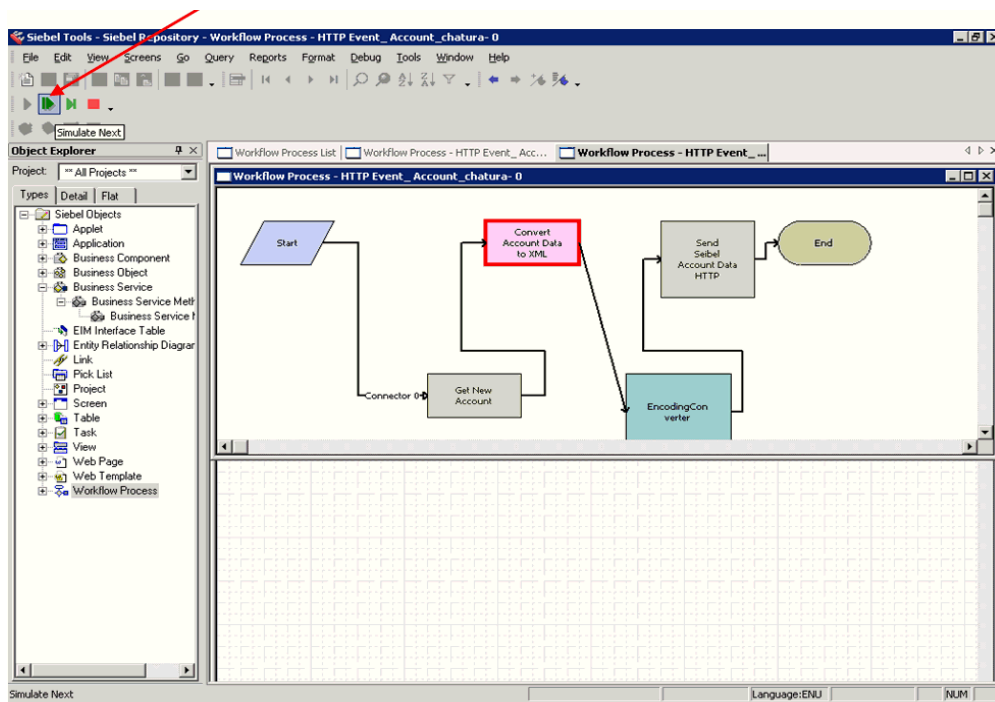
11. 「次をシミュレート」アイコンをクリックします。図 4-111 に示すように、「Get New Account」ボックスが強調表示されます。

図 4-111 次をシミュレート



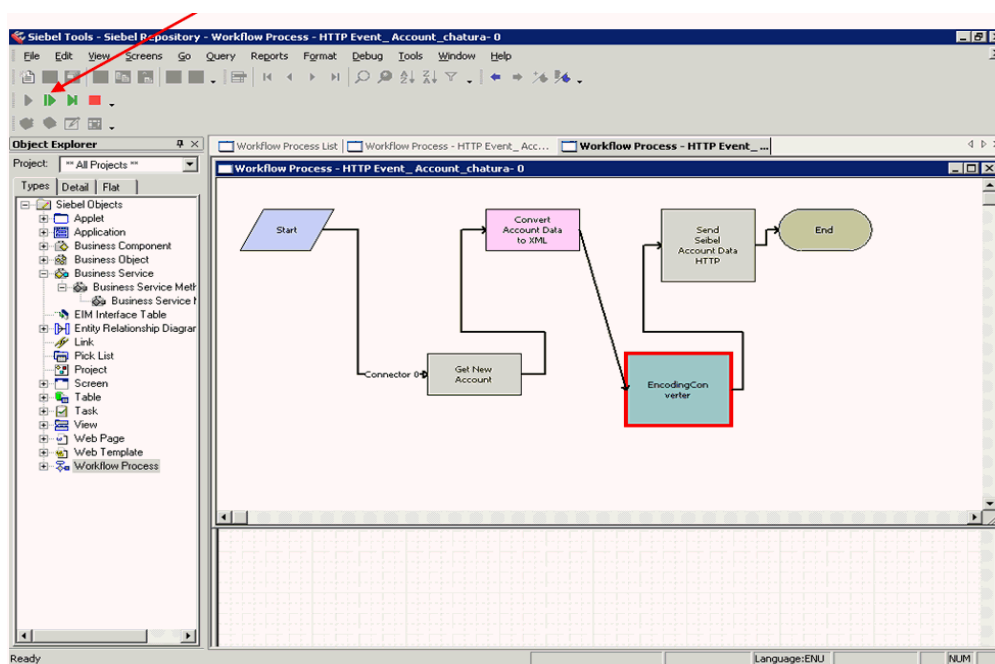
12. 「次をシミュレート」アイコンをクリックします。図 4-112 に示すように、「Convert Account Data to XML」ボックスが強調表示されます。

図 4-112 Convert Account Data to XML



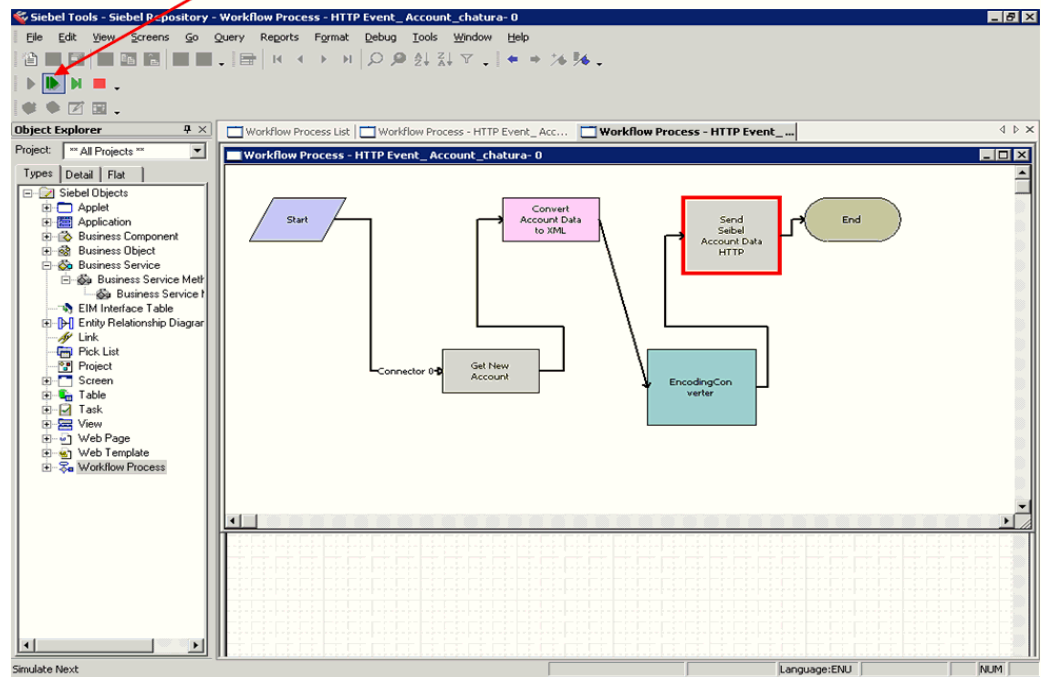
13. 「次をシミュレート」アイコンをクリックします。図 4-113 に示すように、「Encoding Converter」ボックスが強調表示されます。

図 4-113 Encoding Converter



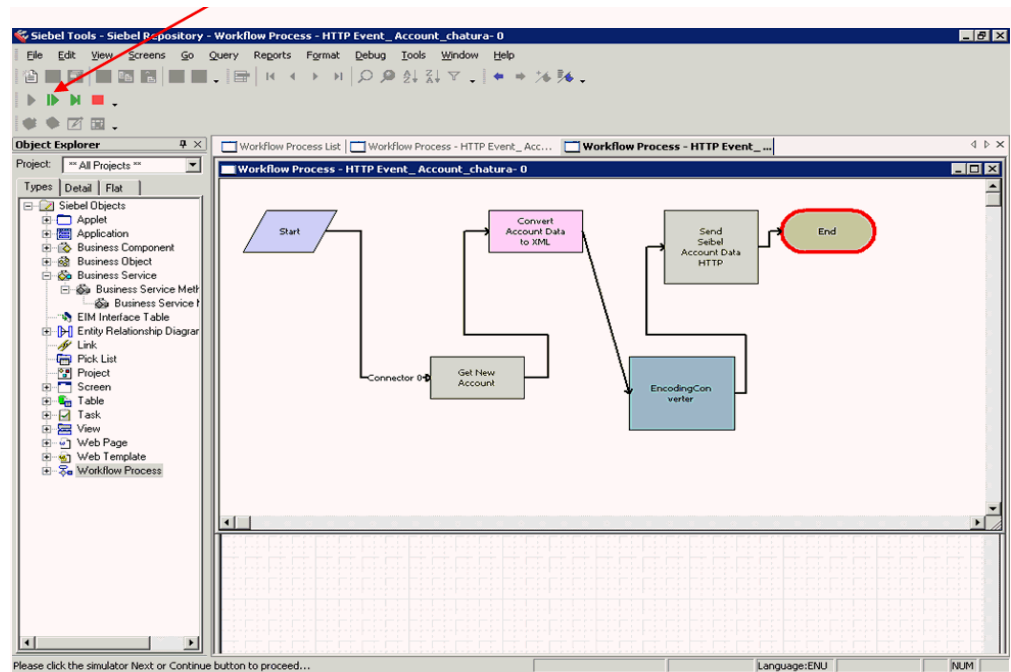
14. 「次をシミュレート」アイコンをクリックします。図 4-114 に示すように、「Send Siebel Account Data HTTP」ボックスが強調表示されます。

図 4-114 Send Siebel Account Data HTTP



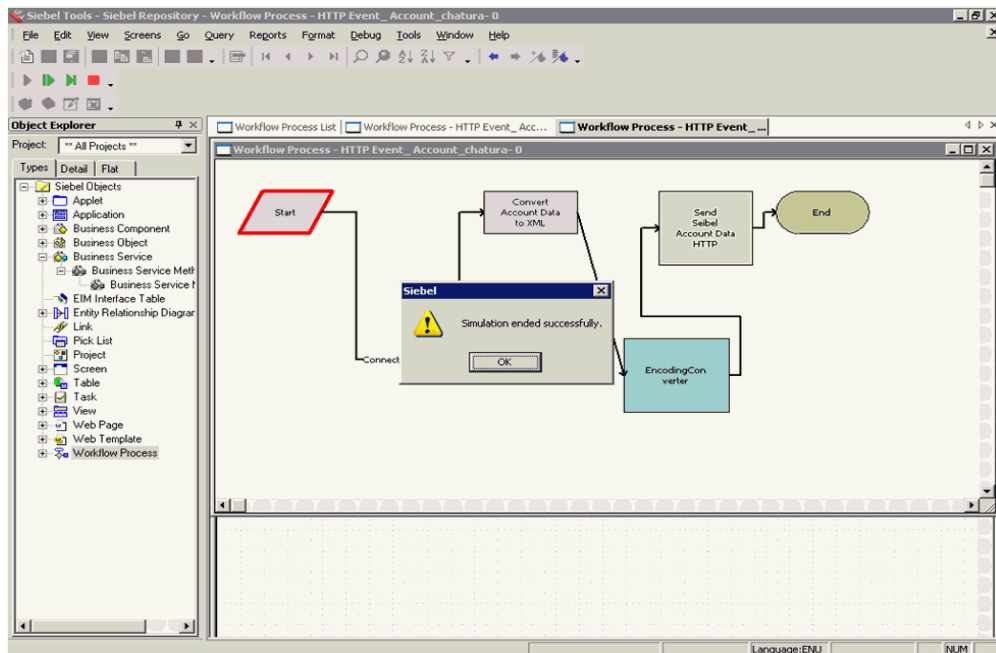
15. 「次をシミュレート」アイコンをクリックします。図 4-115 に示すように、「終了」が強調表示されます。

図 4-115 次をシミュレート



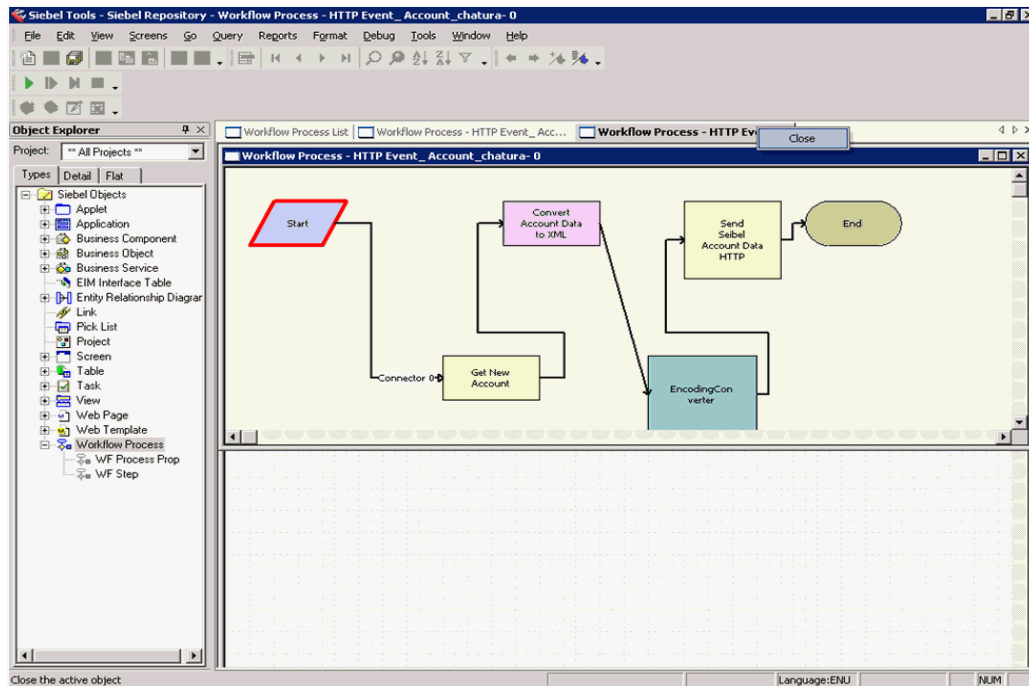
16. 図 4-116 に示すように、次のステップをクリックし、Siebel の成功を伝えるメッセージが表示されたら「OK」をクリックします。

図 4-116 Siebel の成功を伝えるメッセージ



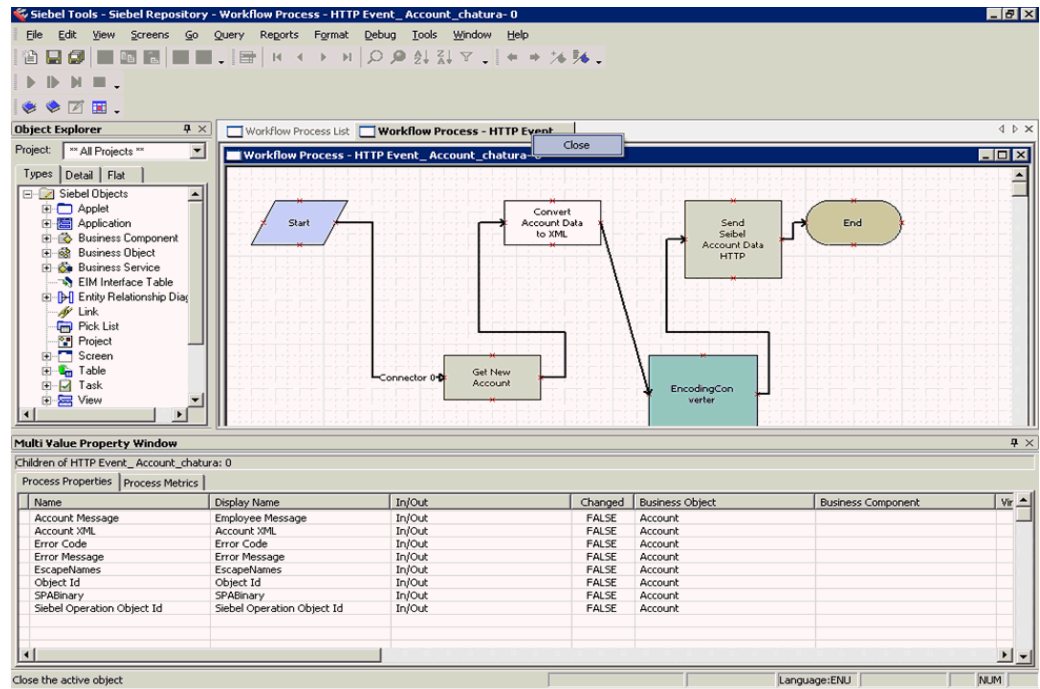
17. 図 4-117 に示すように、3 番目のワークフロー・プロセス・タブを右クリックして「閉じる」を選択します。

図 4-117 ワークフロー・プロセス・タブ



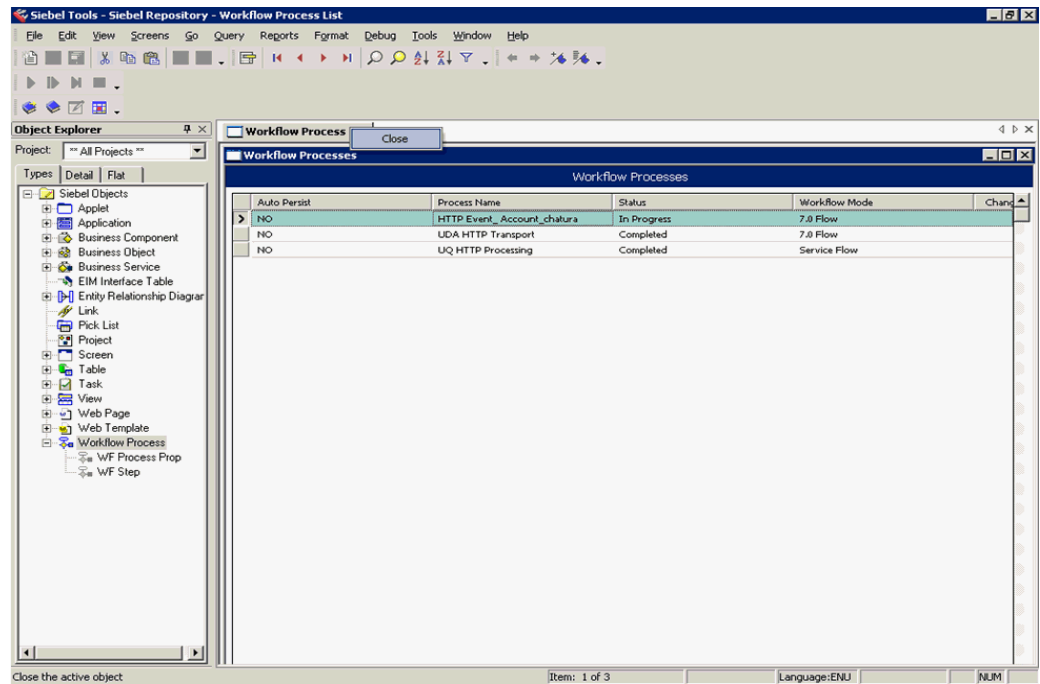
18. 図 4-118 に示すように、2 番目のワークフロー・プロセス・タブを右クリックして「閉じる」を選択します。

図 4-118 ワークフロー・プロセス・タブ



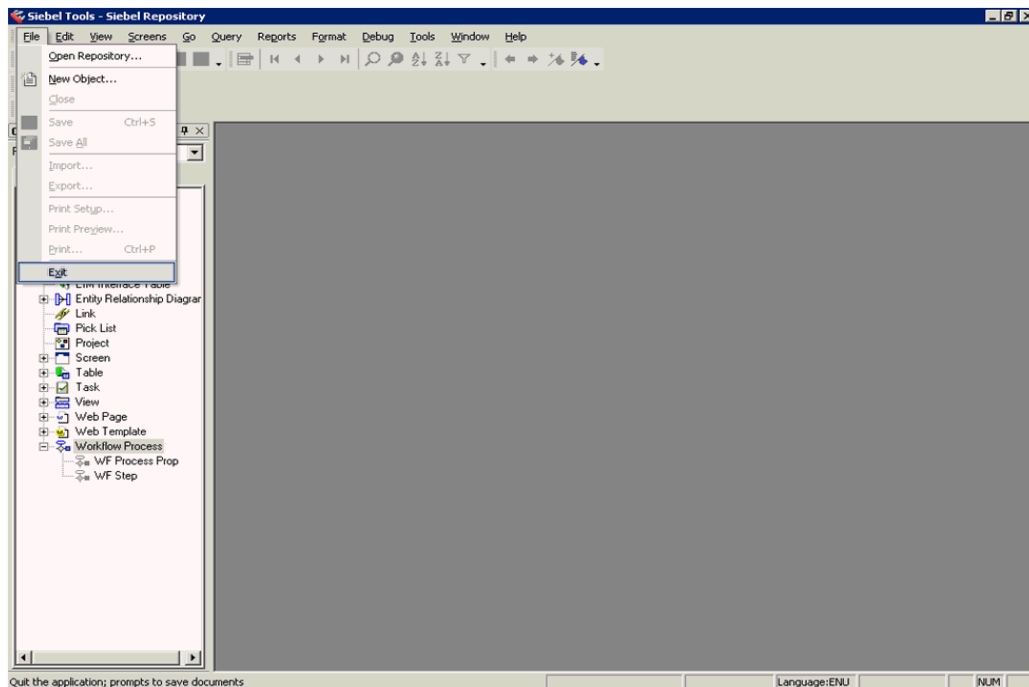
19. 図 4-119 に示すように、残っているワークフロー・プロセス・タブを右クリックして「閉じる」を選択します。

図 4-119 ワークフロー・プロセス・タブ



20. 図 4-120 に示すように、「ファイル」メニューの「終了」をクリックしてツールを閉じます。

図 4-120 終了

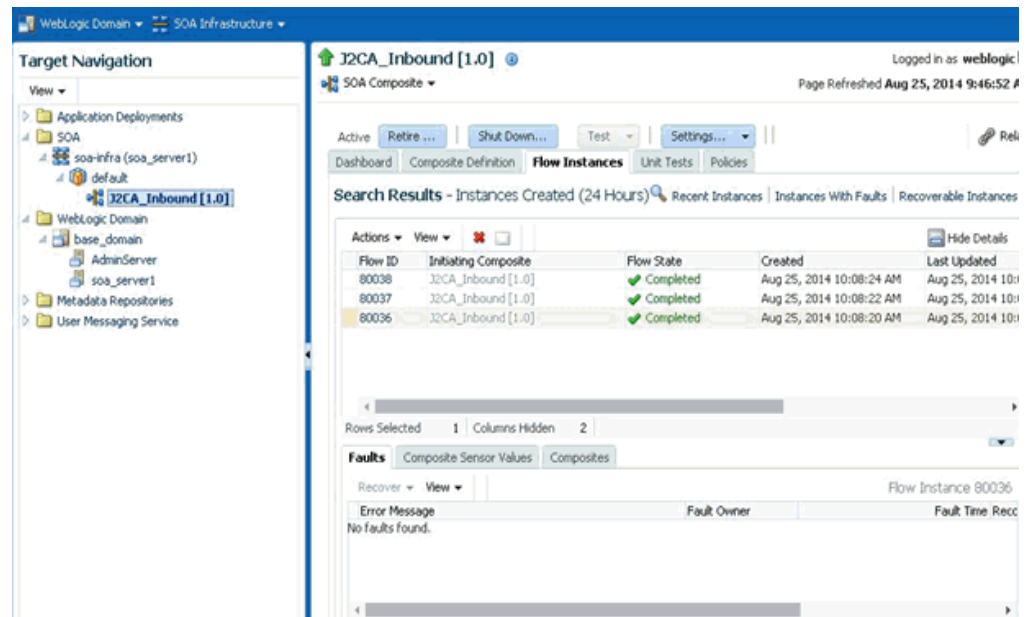


4.5.5.4 結果の確認

結果を確認するには次を行います。

1. 次の URL を使用して、Oracle Enterprise Manager Console にログインします。
`http://localhost:7001/em`
2. 「SOA」をクリックして「soa-infra (soa_server1)」、「デフォルト」を選択し、「J2CA_Inbound」をクリックします。
3. 「フロー・インスタンス」をクリックします。
☒ 4-121 に示すように、インスタンスが受信されます。

図 4-121 「フロー・インスタンス」タブ



4.6 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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPEL\BSE\Outbound_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)

この項では、次のトピックについて説明します。

- [4.6.1 項「Web サービスを使用するリクエスト/レスポンス型サービス用の WSDL ファイルの生成」](#)
- [4.6.2 項「SOA 用の空のコンポジットの作成」](#)
- [4.6.3 項「BPEL アウトバウンド・プロセスの定義」](#)

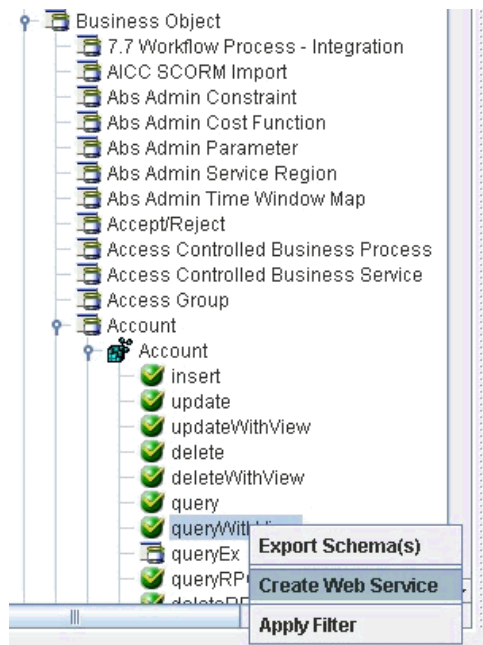
BPEL プロセスを設計する前に、アプリケーション・エクスプローラを使用して対応する WSDL ファイルを生成する必要があります。詳細は、[4.6.1 項「Web サービスを使用するリクエスト/レスポンス型サービス用の WSDL ファイルの生成」](#)を参照してください。

4.6.1 Web サービスを使用するリクエスト/レスポンス型サービス用の WSDL ファイルの生成

Web サービスを使用するリクエスト/レスポンス型サービス用の WSDL ファイルを生成するには次を行います。

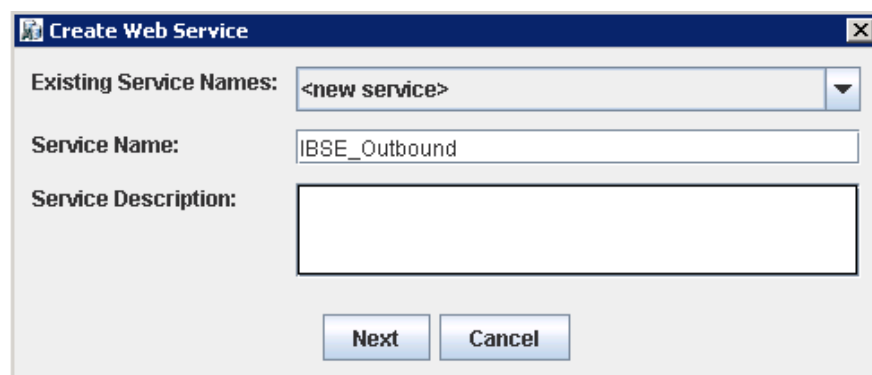
1. アプリケーション・エクスプローラを起動し、定義済の Siebel ターゲット (BSE 構成) に接続します。
ターゲットの定義および Siebel への接続の詳細は、[2.4.1 項「Siebel へのターゲットの定義」](#)を参照してください。
2. 接続先の Siebel ターゲットを展開します。
3. 「ビジネス・オブジェクト」、「アカウント」、「アカウント」の順に展開します。
4. [図 4-122](#) に示すように、「queryWithView」を右クリックし、メニューから「Web サービスの作成」を選択します。

図 4-122 「queryWithView」ノード



[図 4-123](#) 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 4-123 「Web サービスの作成」ダイアログ



5. サービスの名前を入力し、「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. WSDL を wsdl フォルダに保存して「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できます。

4.6.2 SOA 用の空のコンポジットの作成

SOA 用の空のコンポジットを作成するには次を行います。

1. 新規の SOA アプリケーションを作成します。
2. SOA アプリケーションの名前を入力し、「次へ」をクリックします。
「プロジェクトの名前付け」ページが表示されます。
3. プロジェクトの名前を入力し、「次へ」をクリックします。
「SOA」ページが表示されます。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。
詳細は、4-9 ページ [4.4.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

4.6.3 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・コンポーネントの構成方法について説明します。

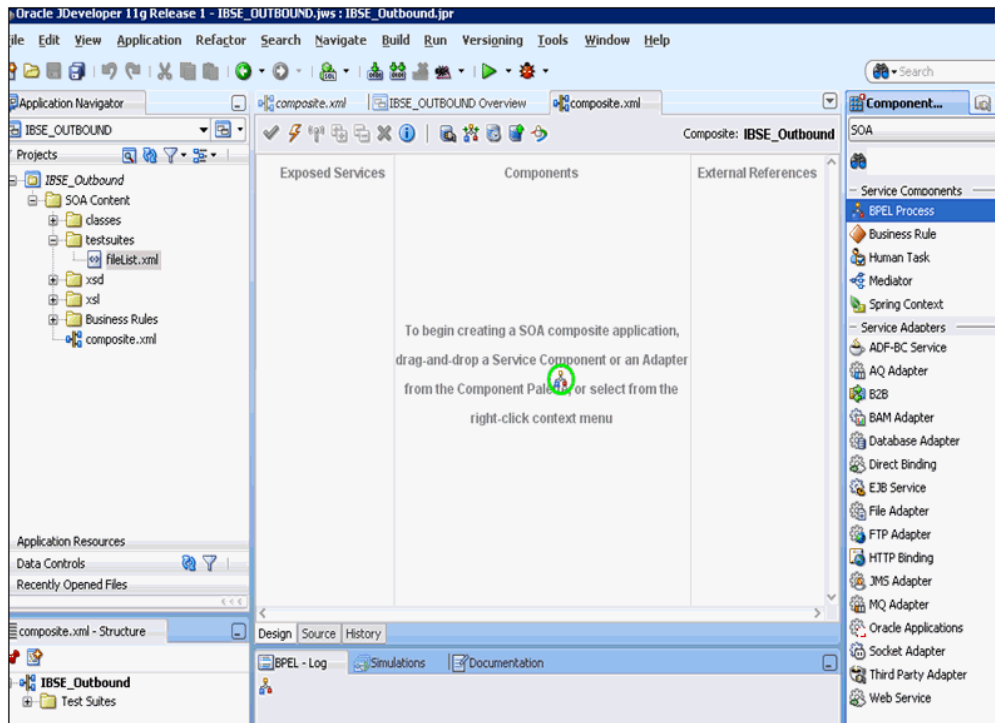
この項では、次のトピックについて説明します。

- [4.6.3.1 項「パートナ・リンクの作成」](#)
- [4.6.3.2 項「BPEL アクティビティの作成および作成済のパートナ・リンクとのマッピング」](#)

BPEL アウトバウンド・プロセスを定義するには次を行います。

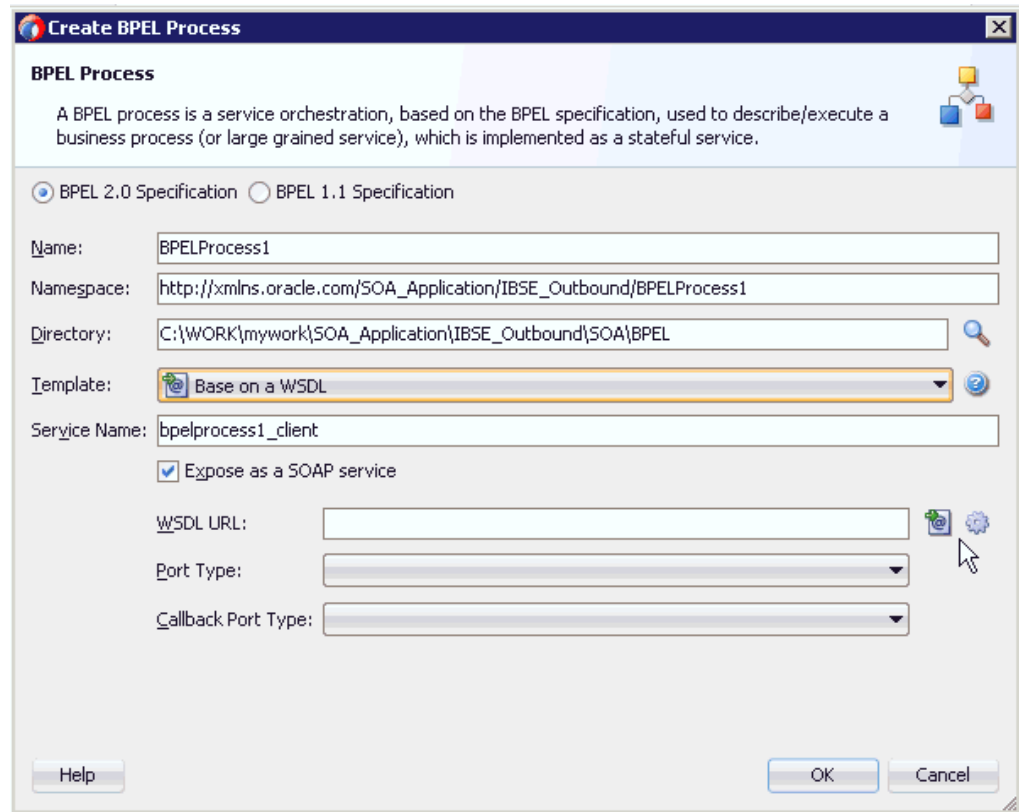
1. [図 4-124](#) に示すように、「サービス・コンポーネント」ペインから「BPEL プロセス」コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップします。

図 4-124 BPEL プロセス・コンポーネント



2. 「名前」フィールドに、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、デフォルト値のままにします。
「BPEL 2.0 仕様」オプションがデフォルトで選択されています。
3. 「テンプレート」ドロップダウン・リストで「**WSDL に基づく**」を選択します。
4. 図 4-125 に示すように、「WSDL URL」フィールドの右の既存の WSDL を検索アイコンをクリックします。

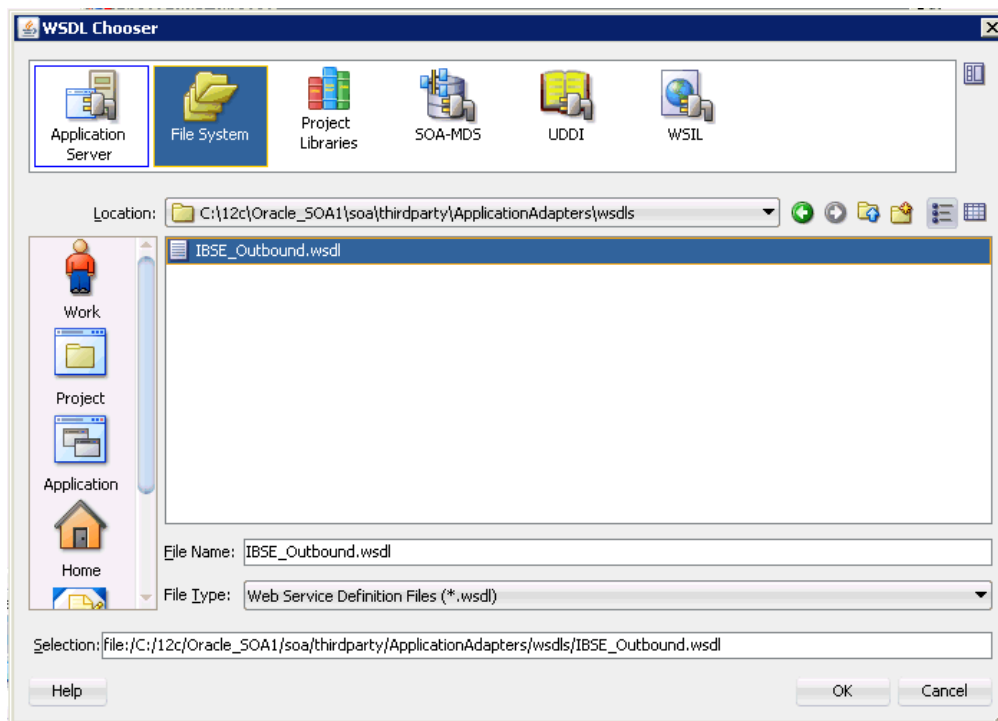
図 4-125 既存の WSDL を検索アイコン



「WSDL チューザ」ダイアログが表示されます。

5. 図 4-126 に示すように、アプリケーション・エクスプローラから WSDL がエクスポートされる場所に移動し、WSDL を選択して「OK」をクリックします。

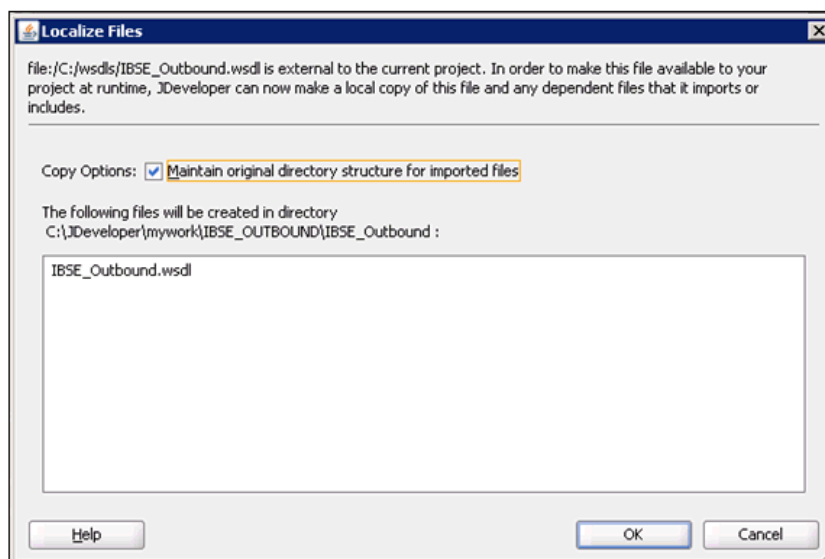
図 4-126 「WSDL チューザ」 ダイアログ



「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 4-127 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 4-127 「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 4-128 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 4-128 「BPEL プロセス」 ペイン

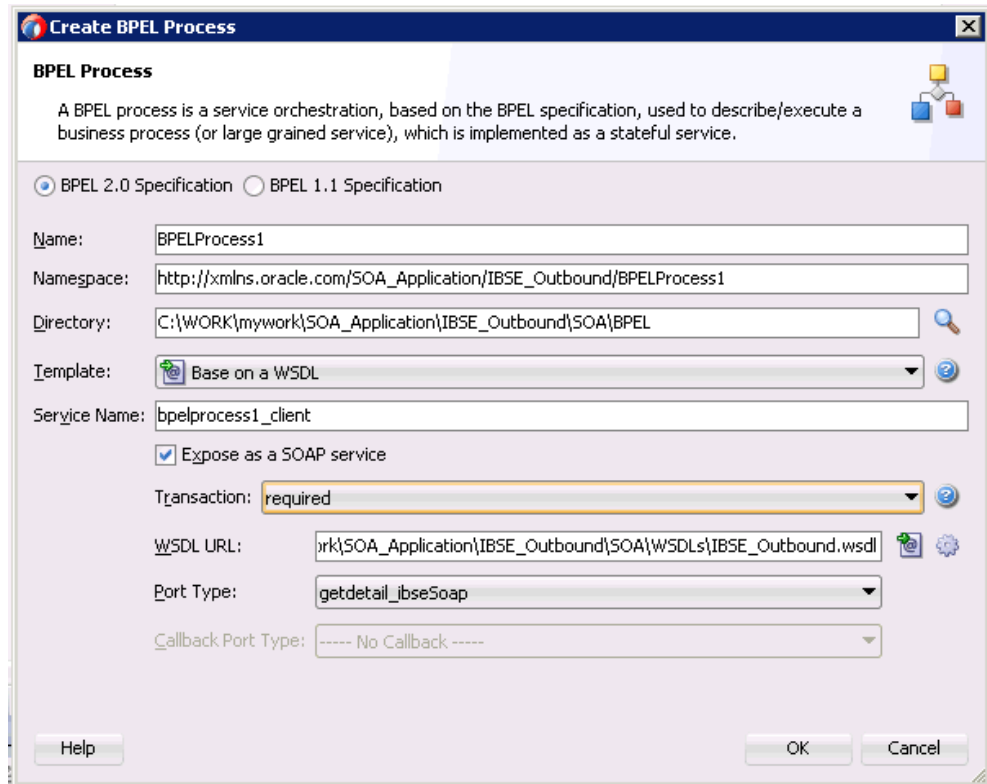
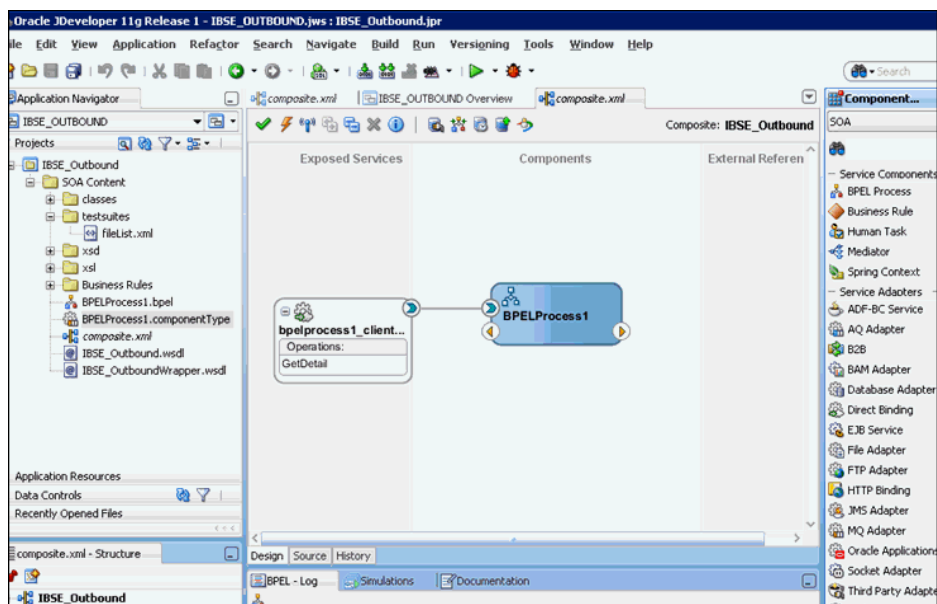


図 4-129 に示すように、BPEL プロセス・コンポーネントが作成されて表示されます。

図 4-129 BPEL プロセス・コンポーネント



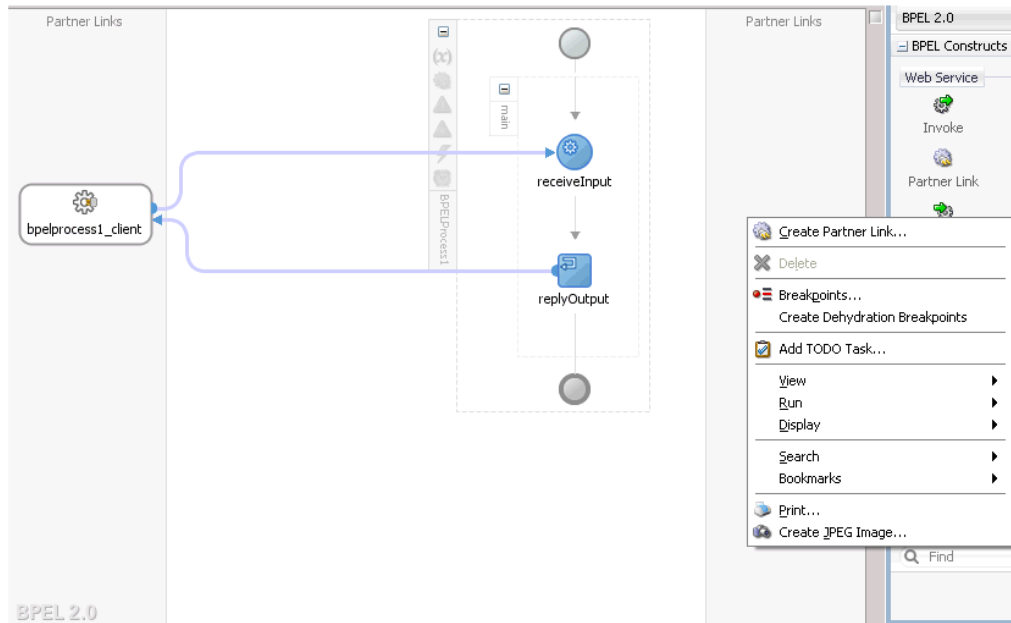
4.6.3.1 パートナ・リンクの作成

この項では、パートナ・リンクの作成方法について説明します。

パートナ・リンクを作成するには次を行います。

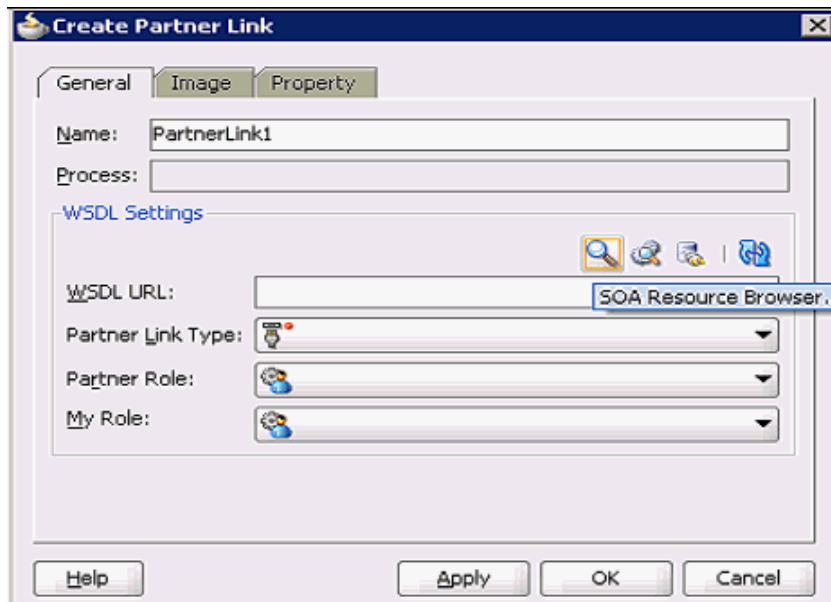
1. 「コンポーネント」ペインで、アウトバウンド BPEL プロセス・コンポーネントをダブルクリックします。
2. 図 4-130 に示すように、「パートナ・リンク」ペインを右クリックして「パートナ・リンクの作成」を選択します。

図 4-130 パートナ・リンクの作成



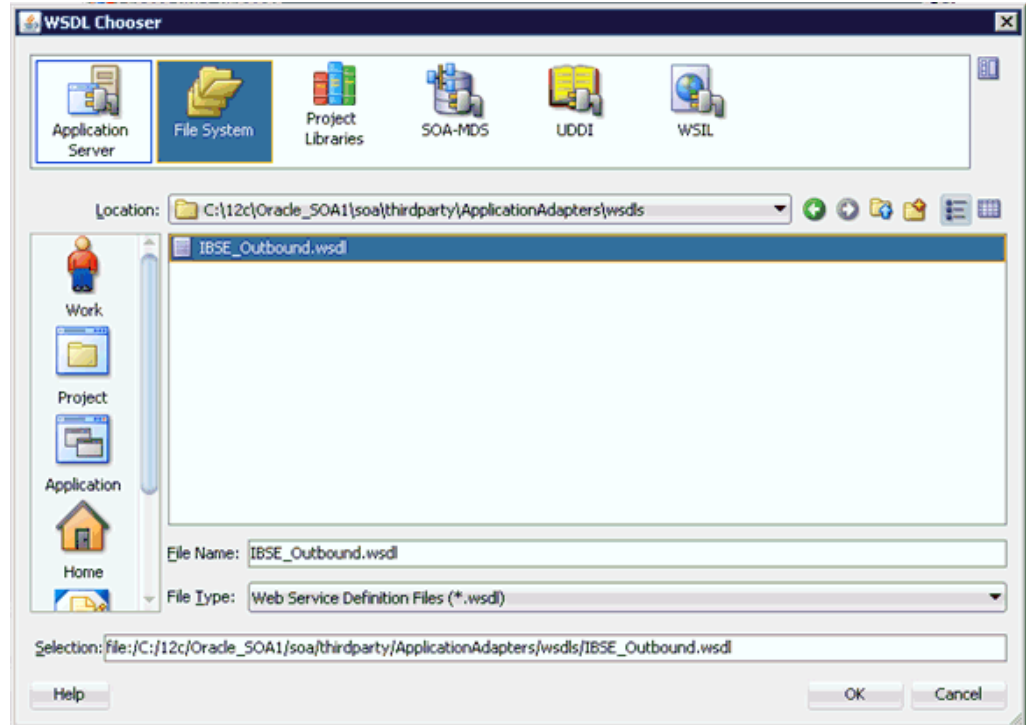
3. 図 4-131 に示すように、表示される「パートナ・リンクの作成」ウィンドウで、適切な名前を入力して SOA リソース・ブラウザ・ツールをクリックします。

図 4-131 SOA リソース・ブラウザ・ツール



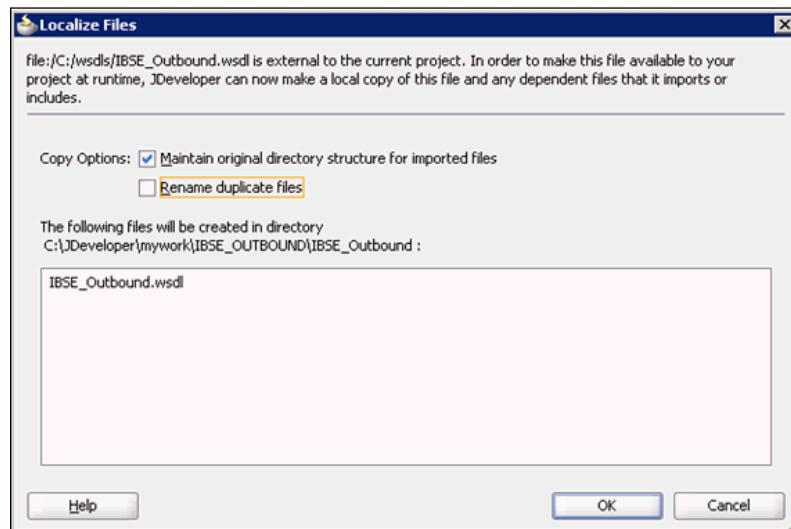
4. 図 4-132 に示すように、表示される「WSDL チューザ」ダイアログで、アプリケーション・エクスプローラから WSDL がエクスポートされる場所へ移動し、WSDL を選択して「OK」をクリックします。

図 4-132 「WSDL チューザ」ダイアログ



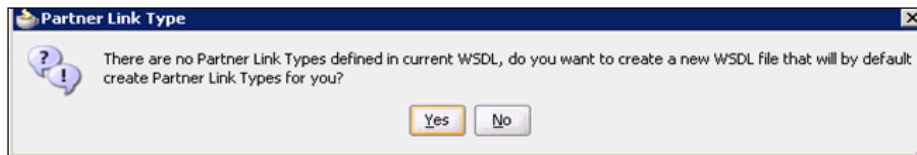
5. 図 4-133 に示すように、表示される「ファイルのローカライズ」ウィンドウで、「重複ファイルの名前変更」チェック・ボックスの選択を解除して「OK」をクリックします。

図 4-133 「ファイルのローカライズ」ウィンドウ



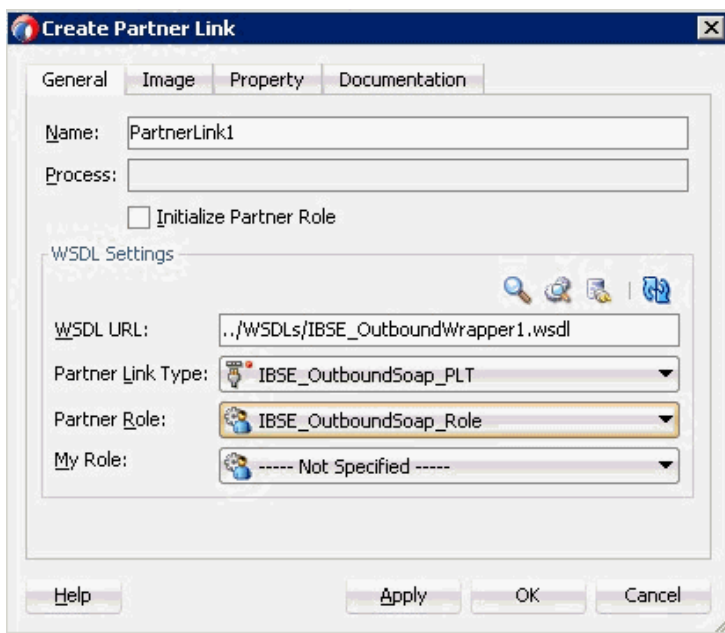
6. 図 4-134 に示すように、表示される「パートナ・リンク・タイプ」ウィンドウで「はい」をクリックします。

図 4-134 パートナ・リンク・タイプ



7. 表示される「パートナ・リンクの作成」ウィンドウで、「パートナ・ロール」ドロップダウン・リストを展開して使用可能なパートナ・ロールを選択します。
8. 図 4-135 に示すように、「適用」をクリックし、「OK」をクリックします。

図 4-135 パートナ・リンクの作成



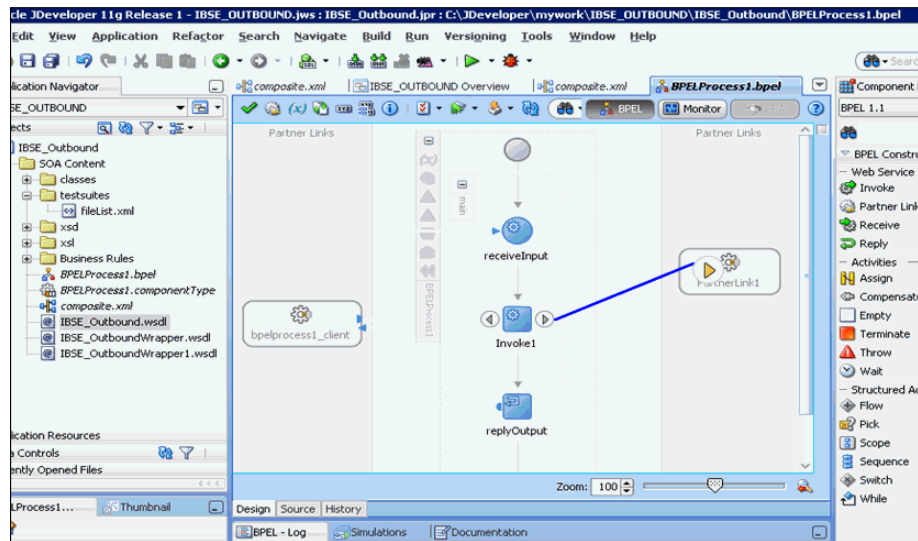
4.6.3.2 BPEL アクティビティの作成および作成済のパートナ・リンクとのマッピング

この項では、BPEL アクティビティの作成方法および作成済のパートナ・リンクとのマッピング方法について説明します。

BPEL アクティビティを作成し、作成済のパートナ・リンクとマッピングするには次を行います。

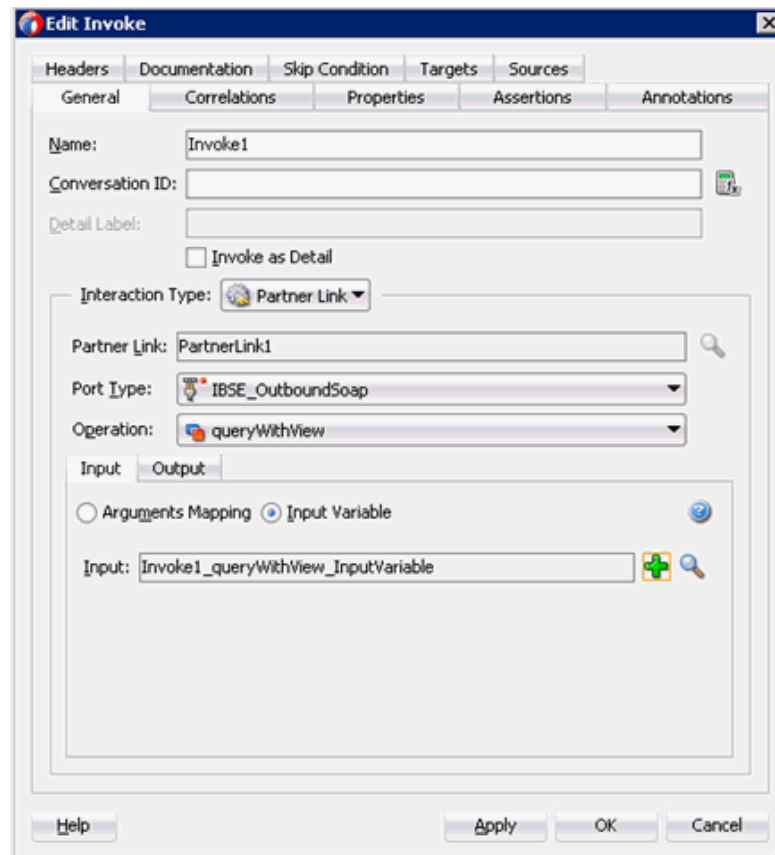
1. 「BPEL コンストラクト」から **Invoke** アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップします。それを **receiveInput** アクティビティ・コンポーネントと **replyOutput** アクティビティ・コンポーネントの間に置きます。
2. 図 4-136 に示すように、新規の **Invoke** アクティビティ・コンポーネント (Invoke1) とパートナ・リンク・コンポーネント (Partner link1) の間に接続を作成します。

図 4-136 パートナ・リンク・コンポーネント



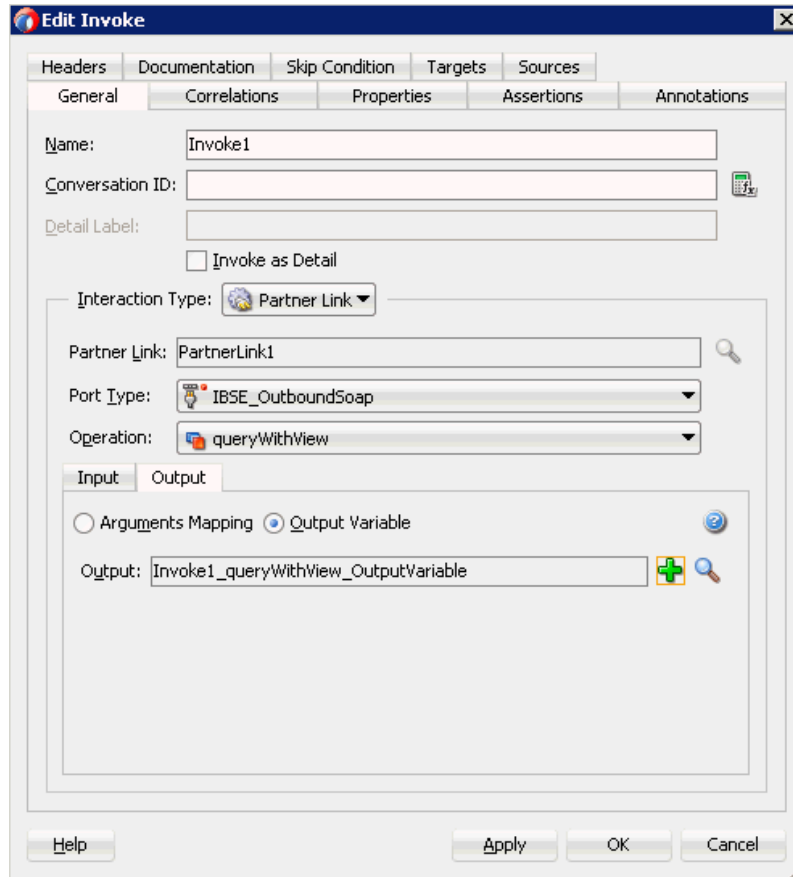
3. 表示される Invoke の編集ウィンドウで、新規入力変数を構成するために「入力」(+)アイコンをクリックします。
4. 新規の入力変数に示されたデフォルト値を受け入れ、「OK」をクリックします。
5. 図 4-137 に示すように、新規出力変数を構成するために「出力」(+)アイコンをクリックします。

図 4-137 Invoke の編集ウィンドウ



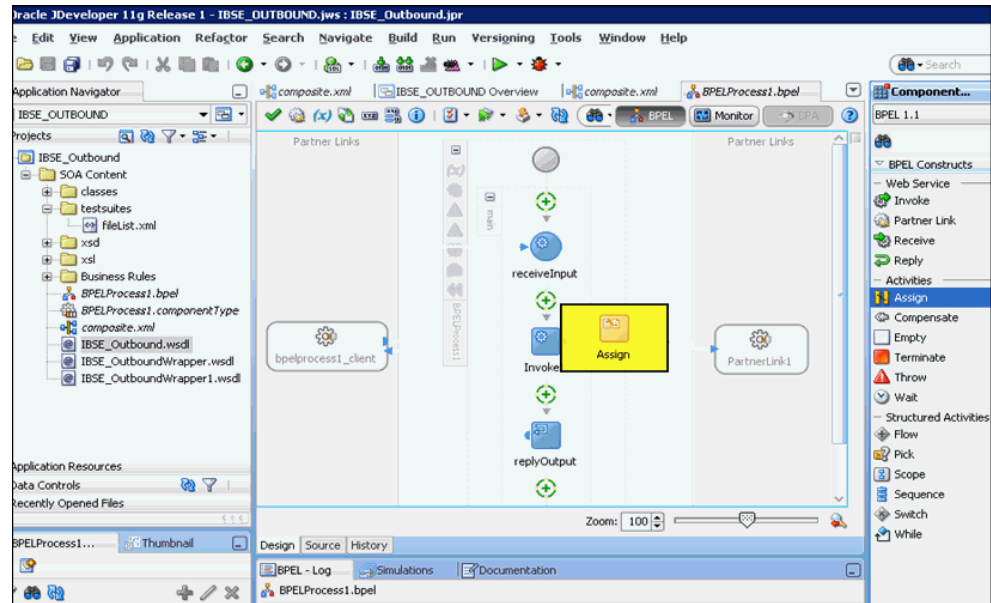
6. 新規の出力変数に示されたデフォルト値を受け入れ、「OK」をクリックします。
7. [図 4-138](#) に示すように、「適用」をクリックし、「OK」をクリックします。

図 4-138 Invoke の編集ウィンドウ



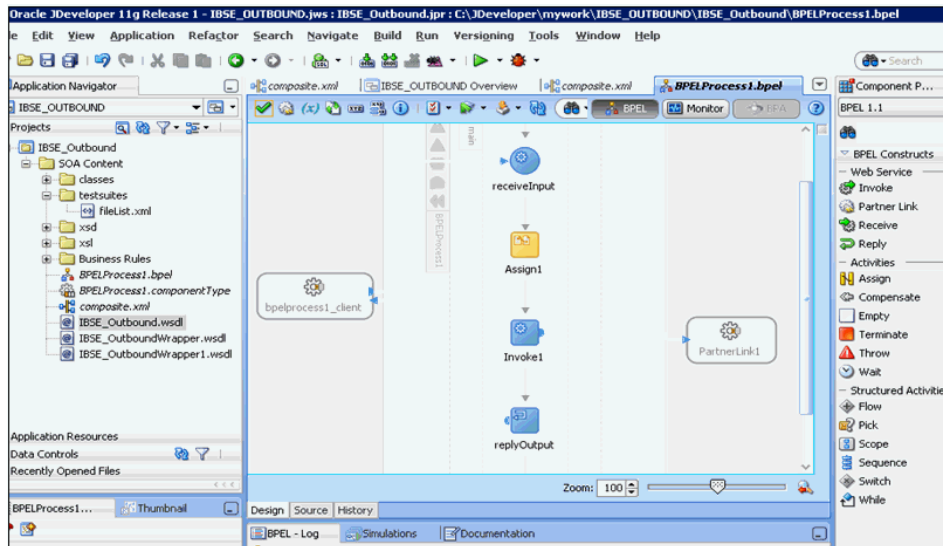
- 「BPEL コンストラクト」から **Assign** アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップします。図 4-139 に示すように、それを **Receive** アクティビティ・コンポーネント (receiveInput) と **Invoke** アクティビティ・コンポーネント (Invoke1) の間に置きます。

図 4-139 Assign アクティビティ・コンポーネント



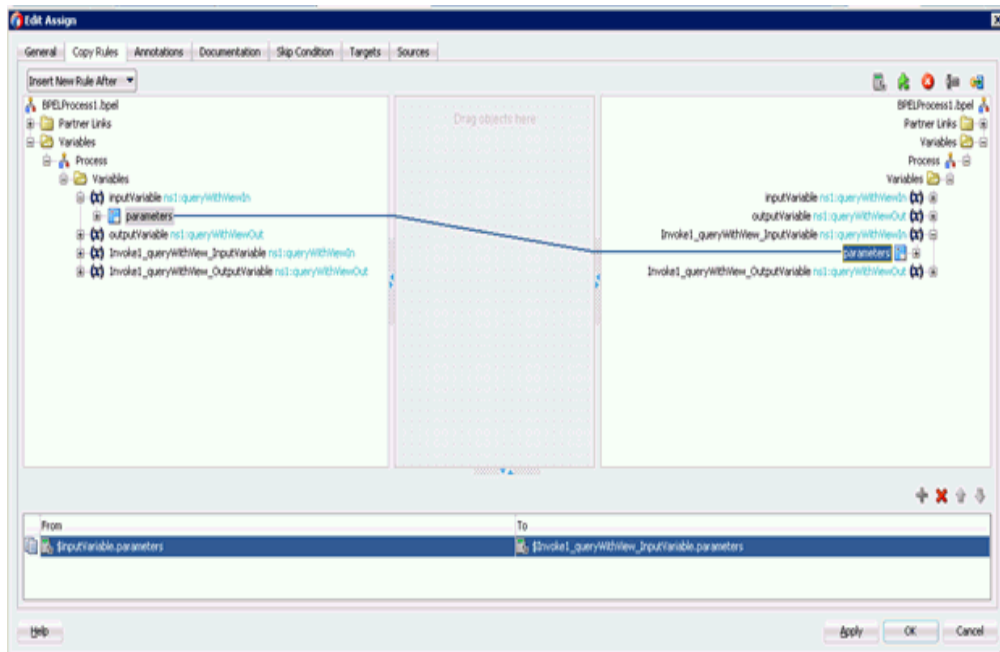
- 図 4-140 に示すように、新規の **Assign** アクティビティ・コンポーネント (Assign1) をダブルクリックします。

図 4-140 Assign アクティビティ・コンポーネント



10. 左ペインの「変数」の下の「InputVariable」を展開し、「パラメータ」を選択します。
11. 右ペインの「変数」の下の「Invoke1_queryWithView_InputVariable」を展開し、「パラメータ」を選択します。
12. 図 4-141 に示すように、InputVariable のパラメータを Invoke1_queryWithView_InputVariable のパラメータにドラッグしてマップします。

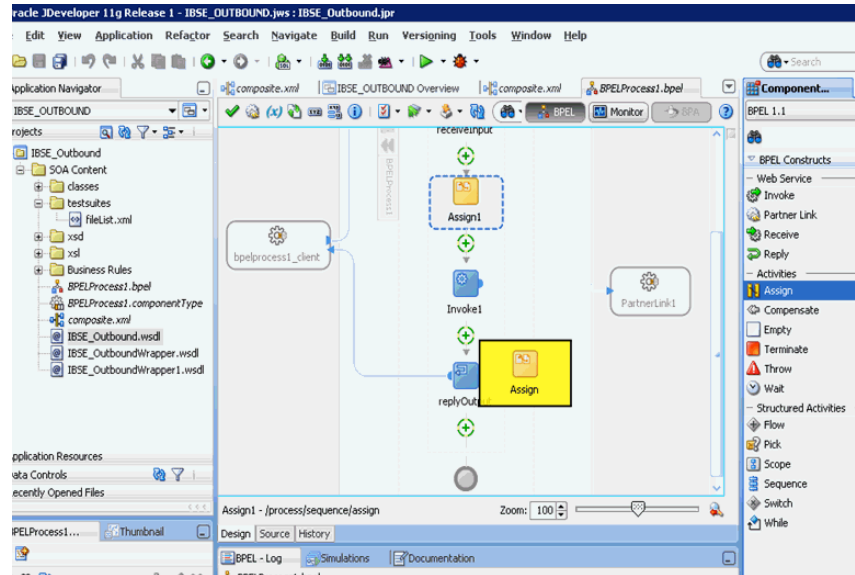
図 4-141 InputVariable のパラメータ



13. 「適用」をクリックし、「OK」をクリックします。

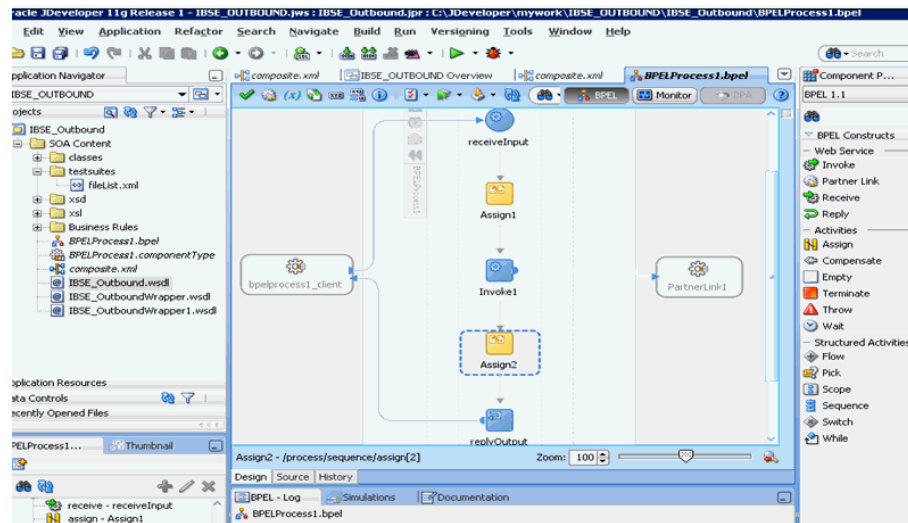
14. 図 4-142 に示すように、**Assign** アクティビティ・コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップし、**Invoke** アクティビティ (Invoke1) と **Reply** アクティビティ (replyOutput) の間に置きます。

図 4-142 Assign アクティビティ・コンポーネント



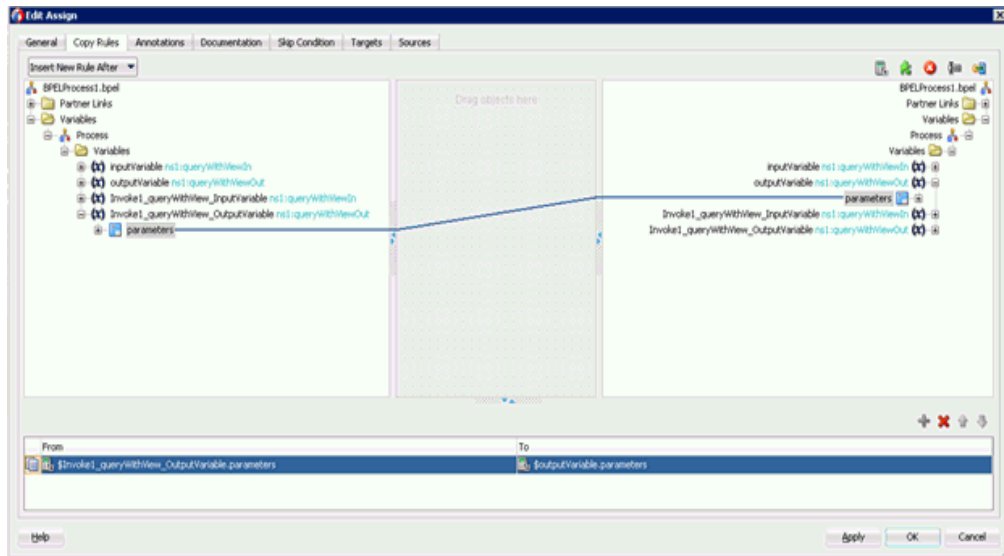
15. 図 4-143 に示すように、新規の **Assign** アクティビティ・コンポーネント (Assign2) をダブルクリックします。

図 4-143 新規の Assign アクティビティ・コンポーネント



16. 左ペインの「変数」の下の「**Invoke1_queryWithView_OutputVariable**」を展開し、「**パラメータ**」を選択します。
17. 右ペインの「変数」の下の「**outputVariable**」を展開し、「**パラメータ**」を選択します。
18. 図 4-144 に示すように、**Invoke1_queryWithView_OutputVariable** のパラメータを **outputVariable** のパラメータにドラッグしてマップします。

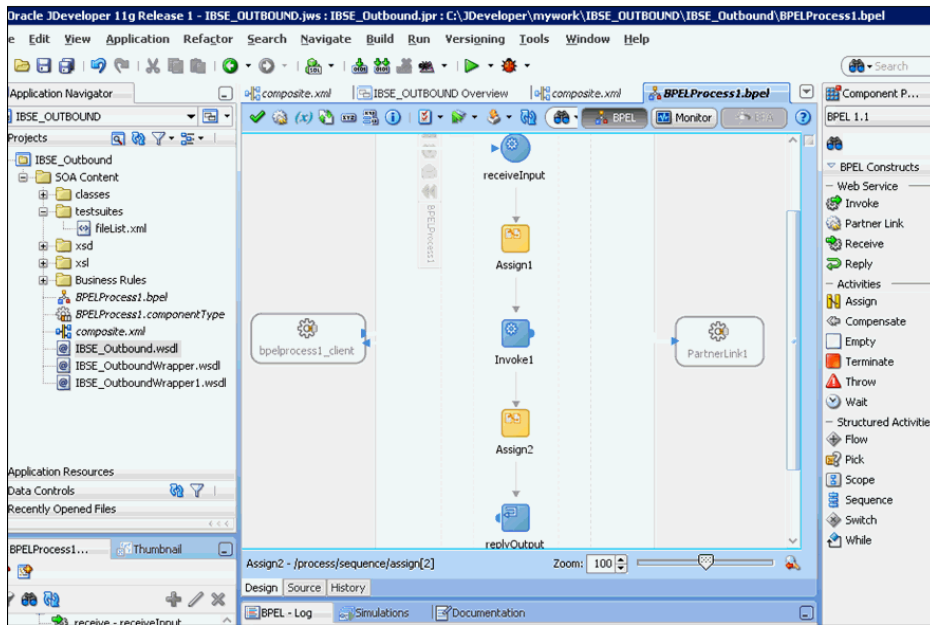
図 4-144 outputVariable のパラメータ



19. 「適用」をクリックし、「OK」をクリックします。

図 4-145 に示すように、「コンポーネント」ペインに戻ります。

図 4-145 「コンポーネント」ペイン



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

これで、BPEL アウトバウンド・プロセスをデプロイする準備ができました。4-28 ページ 4.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」の記載と同じ手順に従います。

デプロイすると、4-31 ページ 4.4.5 項「Oracle Enterprise Manager Console での入力 XML ドキュメントの起動」に定義されているように、入力 XML を起動できるようになります。

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

この章では、Oracle SOA Suite でのメディエータ・サービス・コンポーネントとの統合について説明します。次のトピックについて説明します。

- 5.1 項「新規のアプリケーション・サーバー接続の構成」
- 5.2 項「メディエータ・アウトバウンド・プロセスの構成 (J2CA 構成)」
- 5.3 項「メディエータ・インバウンド・プロセスの構成 (J2CA 構成)」
- 5.4 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

次は、インストールおよび構成に関する要件です。

- Oracle Application Adapter for Siebel が Oracle WebLogic Server にインストールされている必要があります。
- Siebel 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。
- OracleAS テクノロジ・アダプタがデプロイされ、適切に構成されている必要があります。

この章の例では、Siebel とのサービスおよびイベント統合を示すのに必要な構成手順について説明しています。これを使用する前に、次のことについて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 Siebel を構成する方法。詳細は、第 2 章「Oracle Application Server Adapter for Siebel 構成」を参照してください。
- Oracle JDeveloper の構成方法。詳細は、第 4 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。
- Siebel ワークフローの使用法。Siebel の設計要件の詳細は、付録 A「Siebel ワークフローの使用」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 Siebel をメディエータと使用することにより、カスタム・コードの記述を必要とせずに、シームレスにエンタープライズ・ソフトウェアを統合できます。カスタム・コード・ソリューションではなく機能的なモデリングなので、ソフトウェア再利用が可能になり、ソフトウェアのライフサイクルで発生する複雑さや管理上の問題を削減します。この統合モデルは、高レベルの統合ロジックと低レベルのプラットフォーム・サービスの 2 つのコンポーネントで構成されています。

Oracle WebLogic Server とのアダプタの統合、メディアータは次の 2 つのステップからなるプロセスです。

1. **設計時:** 第 2 章「[Oracle Application Server Adapter for Siebel 構成](#)」の説明に従って、アプリケーション・エクスペローラでサービスおよびイベント用に Oracle Application Adapter for Siebel を構成します。統合ロジックは iStudio でモデル化されています。メタデータはリポジトリに格納されます。
2. **実行時:** 基に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・インストラクションとして処理し、特定のアプリケーション間の通信を可能にします。

5.1 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、4-2 ページ [4.3 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。

5.2 メディアータ・アウトバウンド・プロセスの構成 (J2CA 構成)

この項では、Oracle JDeveloper でメディアータ・プロジェクトを使用して、Siebel システム用にメディアータ・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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

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

この項では、次のトピックについて説明します。

- [5.2.1 項「SOA 用の空のコンポジットの作成」](#)
- [5.2.2 項「メディアータ・アウトバウンド・プロセスの定義」](#)
- [5.2.3 項「メディアータ・アウトバウンド・プロセスのデプロイ」](#)
- [5.2.4 項「Oracle Enterprise Manager Console での入力 XML ドキュメントの起動」](#)

前提条件

メディアータ・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。詳細は、4-8 ページ [4.4.1 項「リクエスト/レスポンス・サービス用の WSDL の生成」](#) を参照してください。

5.2.1 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、次のステップを実行します

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

詳細は、4-9 ページ [4.4.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

5.2.2 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。

- 5.2.2.1 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」
- 5.2.2.2 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」
- 5.2.2.3 項「ルーティング・ルールの構成」
- 5.2.2.4 項「12c のデプロイメントに関する既知の問題の調整」

5.2.2.1 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「サード・パーティ・アダプタ」コンポーネントを「外部参照」ペインにドラッグ・アンド・ドロップします。
2. サード・パーティ・アダプタ・サービスの名前を入力します。
3. 「タイプ」ドロップダウン・リストで「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 を検索」アイコンをクリックします。
5. 次のディレクトリでアウトバウンド WSDL ファイルを特定し、選択します。

```
<ADAPTER_HOME>\wsdl
```

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。

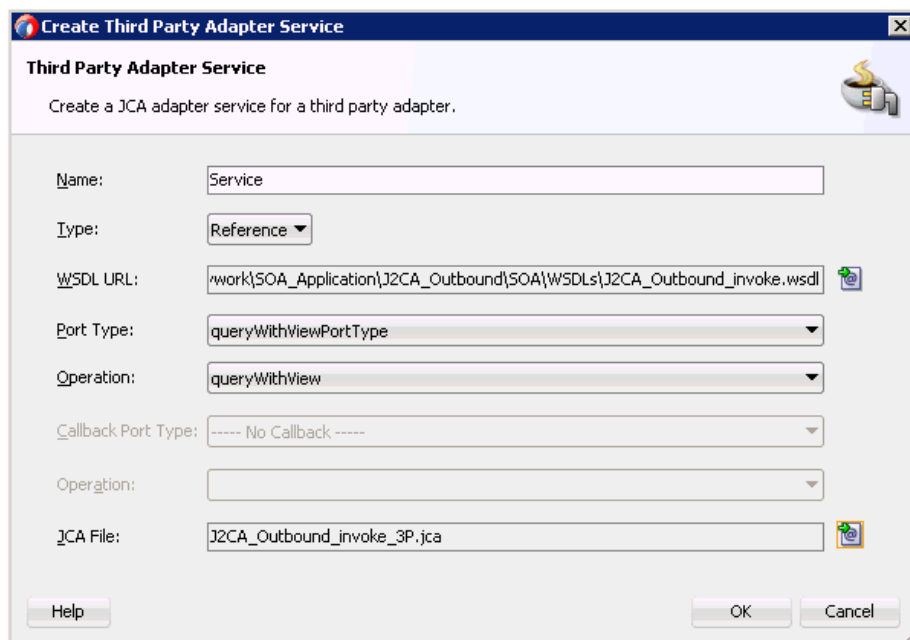
8. 「JCA ファイル」フィールドの右の JCA ファイルの検索アイコンをクリックします。
9. 次のディレクトリまでブラウズし、JCA プロパティ・ファイルを選択します。

```
<ADAPTER_HOME>\wsdl
```

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 5-1 「サード・パーティ・アダプタ・サービスの作成」ダイアログ



12. 「OK」をクリックします。

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

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

詳細は、5-3 ページ [5.2.2.1 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。

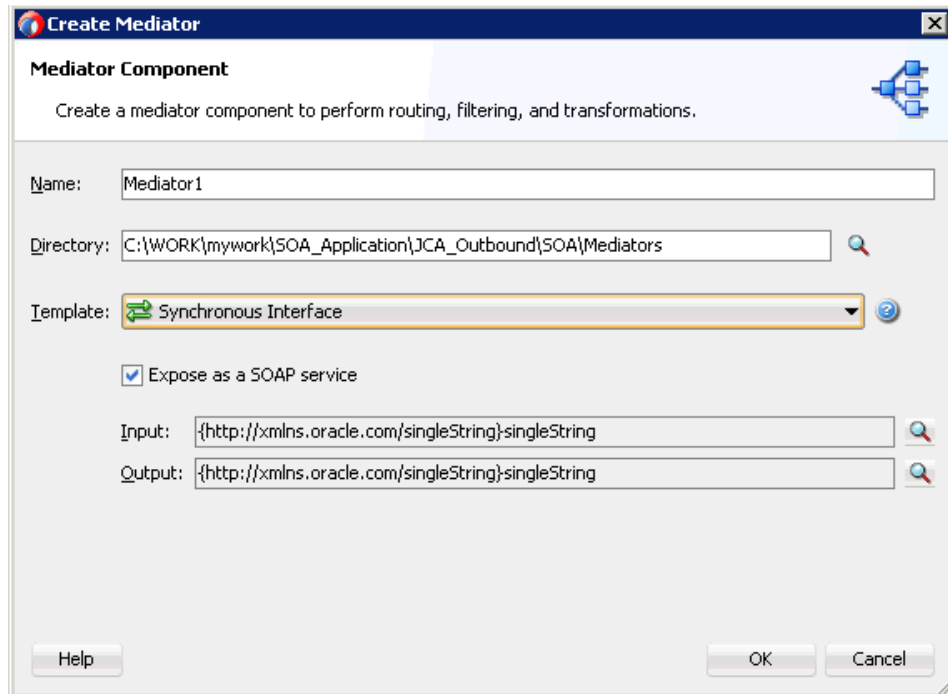
5.2.2.2 アウトバウンド・メディエータ・プロセス・コンポーネントの構成

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

図 5-2 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。

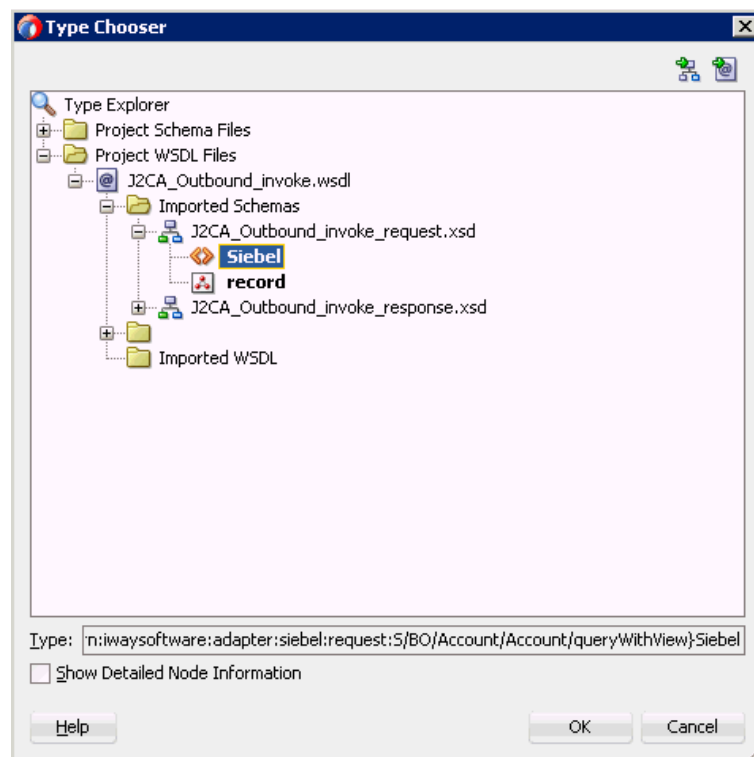
図 5-2 「メディエータの作成」ダイアログ



2. 「名前」フィールドに、新規のアウトバウンド・メディエータ・プロセス・コンポーネントの名前を入力するか、デフォルト値のままにします。
3. 「テンプレート」ドロップダウン・リストで、「同期インタフェース」を選択します。
4. 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。

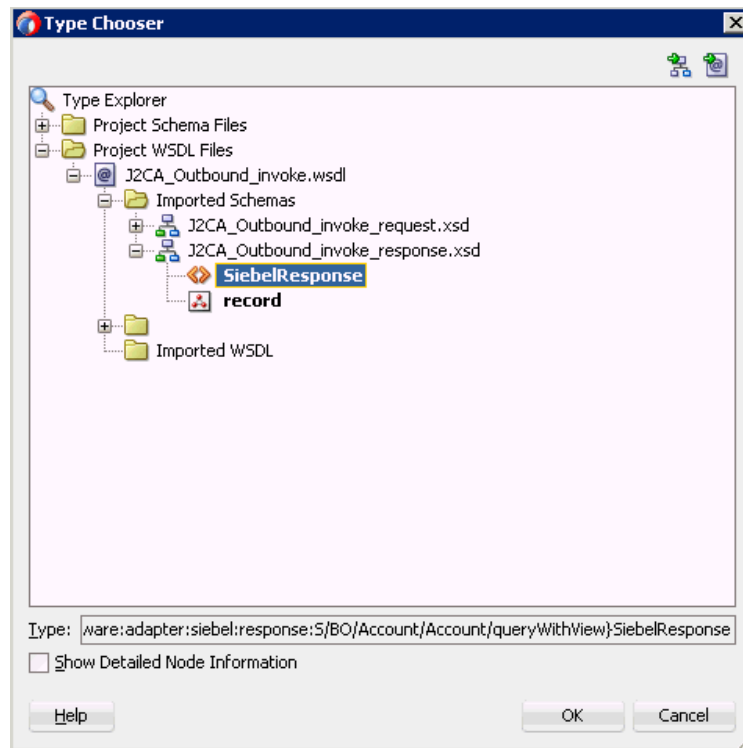
図 5-3 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 5-3 「タイプ・チューザ」 ダイアログ



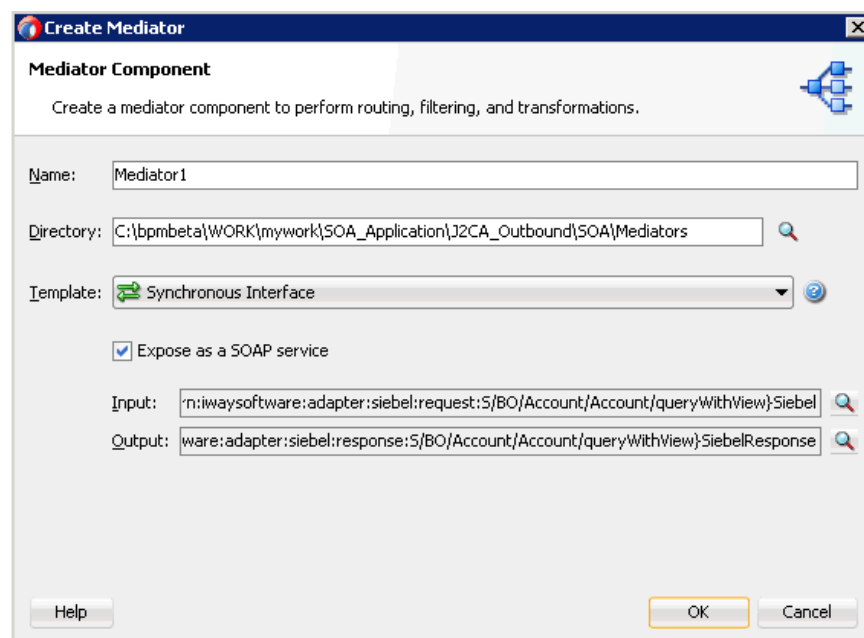
5. 「プロジェクトの WSDL ファイル」、「J2CA_Outbound_invoke.wsdl」、「インポートしたスキーマ」、「J2CA_Outbound_invoke_request.xsd」の順に展開し、「Siebel」を選択します。
6. 「OK」をクリックします。
「メディエータの作成」ダイアログに戻ります。
7. 関連する XML レスポンス・スキーマ・ファイルの選択用の「出力」フィールドの右の「参照」アイコンをクリックします。
図 5-4 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 5-4 「タイプ・チューザ」 ダイアログ



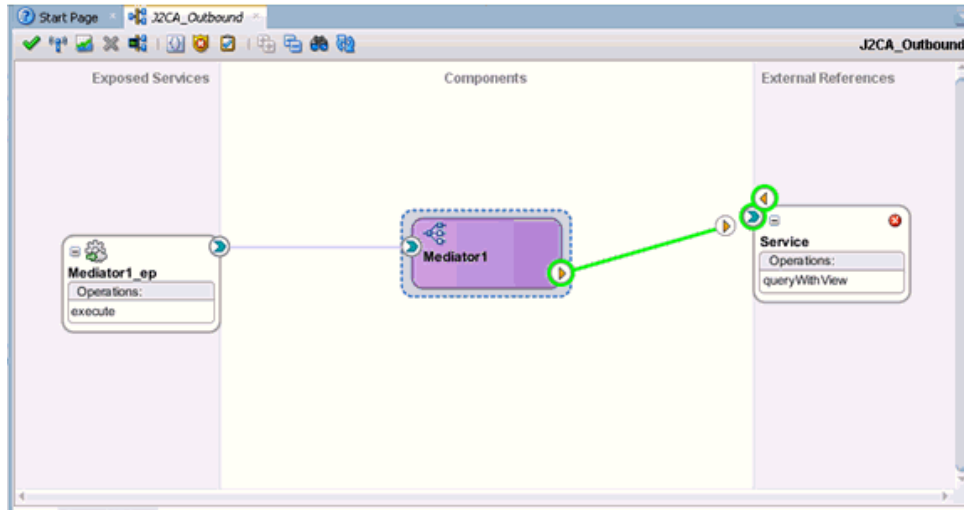
8. 「プロジェクトの WSDL ファイル」、「J2CA_Outbound_invoke.wsdl」、「インポートしたスキーマ」、「J2CA_Outbound_invoke_response.xsd」の順に展開し、「SiebelResponse」を選択します。
9. 「OK」をクリックします。
☒ 5-5 に示すように、「メディエータの作成」ダイアログに戻ります。

図 5-5 「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 5-6 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 5-6 接続の作成



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。

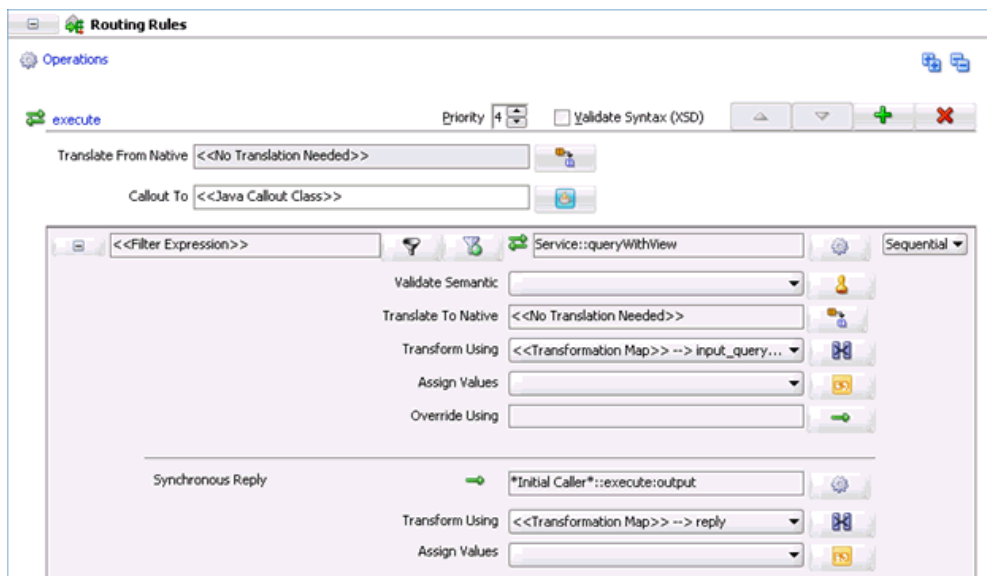
5.2.2.3 ルーティング・ルールの構成

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 5-7 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 5-7 「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。

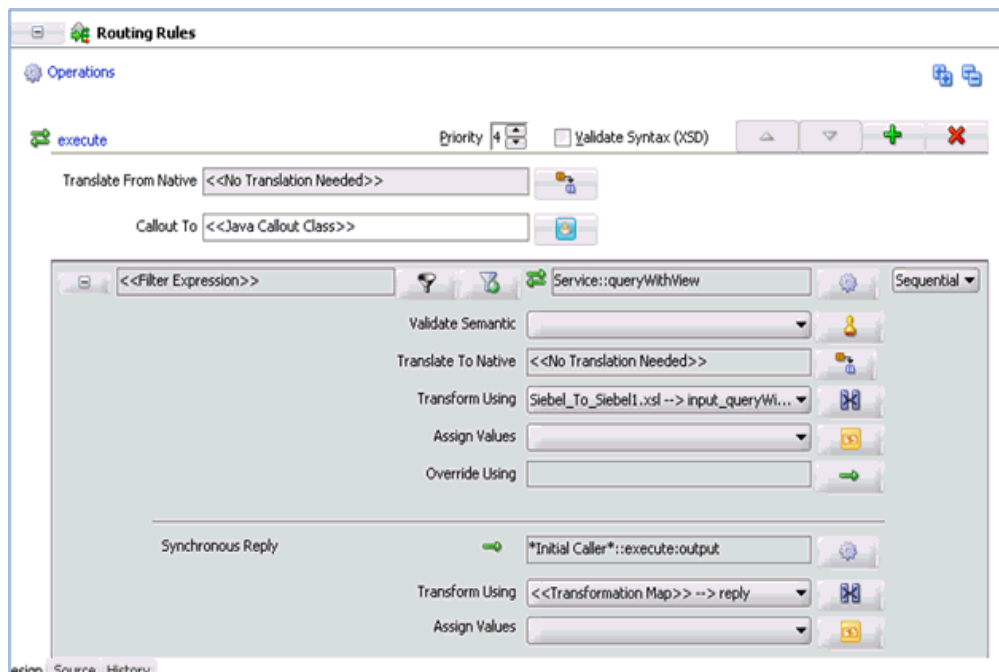
図 5-8 に示すように、「リクエスト・トランスフォーメーション・マップ」ダイアログが表示されます。

図 5-8 「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」として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:Siebel」ソース要素を「ns0:Siebel」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。
8. 図 5-9 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。

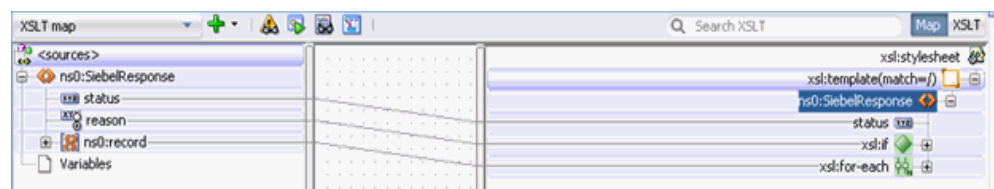
図 5-9 「ルーティング・ルール」ダイアログ



9. 「同期リプライ」領域で、「次を使用して変換」フィールドの右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。
「変換の作成」ページが表示されます。
 11. 「タイプ」が「XSLT」として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:SiebelResponse」ソース要素を「ns0:SiebelResponse」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。
 14. デフォルト値を維持し、「OK」をクリックします。
図 5-10 に示すように、マッピングが完成します。

図 5-10 完成したマッピング



15. メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規アウトバウンド・メディエータ・プロセス・コンポーネントを保存します。

5.2.2.4 12c のデプロイメントに関する既知の問題の調整

12c のデプロイメントに関する既知の問題の調整方法の詳細は、4-26 ページ [4.4.3.3 項「12c のデプロイメントに関する既知の問題の調整」](#) を参照してください。

5.2.3 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. 左ペインで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・サーバーへのデプロイ」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。

詳細は、4-28 ページ [4.4.4 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。

5.2.4 Oracle Enterprise Manager Console での入力 XML ドキュメントの起動

詳細は、4-31 ページ [4.4.5 項「Oracle Enterprise Manager Console での入力 XML ドキュメントの起動」](#) を参照してください。

5.3 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、Siebel システム用にメディエータ・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。

このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\Mediator\J2CA\Inbound_Project
```

この項では、次のトピックについて説明します。

- [5.3.1 項「SOA 用の空のコンポジットの作成」](#)
- [5.3.2 項「メディエータ・インバウンド・プロセスの定義」](#)

前提条件

メディエータ・インバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して各 WSDL ファイルを生成する必要があります。詳細は、4-34 ページ [4.5.1 項「イベント統合のための WSDL の生成」](#) を参照してください。

5.3.1 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、次のステップを実行します

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

詳細は、4-9 ページ [4.4.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

5.3.2 メディエータ・インバウンド・プロセスの定義

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

- [5.3.2.1 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」](#)
- [5.3.2.2 項「ファイル・アダプタを使用したインバウンド・メディエータ・プロセス・コンポーネントの構成」](#)

- 5.3.2.3 項「ルーティング・ルール」の構成
- 5.3.2.4 項「12c のデプロイメントに関する既知の問題の調整」

5.3.2.1 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「サード・パーティ・アダプタ」 コンポーネントを「公開されたサービス」 ペインにドラッグ・アンド・ドロップします。
「サード・パーティ・アダプタ・サービスの作成」 ダイアログが表示されます。
2. サード・パーティ・アダプタ・サービスの名前を入力します。
3. 「タイプ」 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 を検索** 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。
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する受信 / リクエスト・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。
「XQuery 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。
「公開されたサービス」 ペインにサード・パーティ・アダプタ・サービス・コンポーネントが作成されます。
これで、インバウンド・メディアエータ・プロセス・コンポーネントを構成する準備ができました。
詳細は、4-42 ページ [4.5.3.1 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。

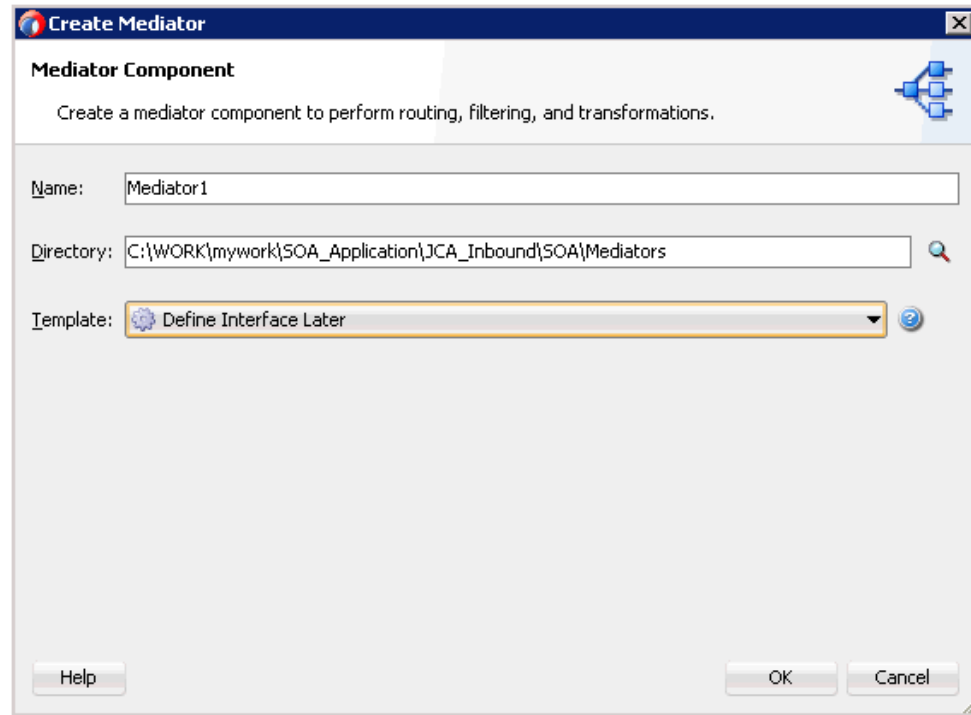
5.3.2.2 ファイル・アダプタを使用したインバウンド・メディエータ・プロセス・コンポーネントの構成

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

図 5-11 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。

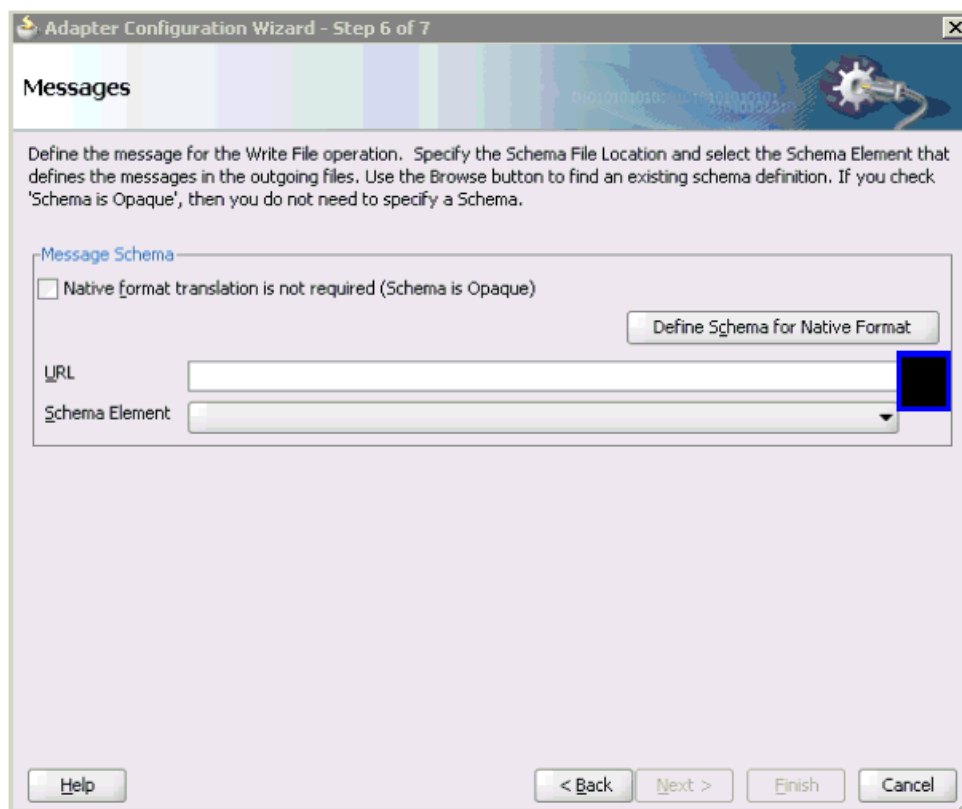
図 5-11 「メディエータの作成」 ダイアログ



2. 「名前」 フィールドに、新規のインバウンド・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・リストで、「**インタフェースを後で定義**」を選択します。
4. 「**OK**」をクリックします。
「コンポーネント」 ペインに新規メディエータ・プロセス・コンポーネントが追加されます。
5. 「テクノロジー・アダプタ」 ペインから**ファイル・コンポーネント**を「外部参照」 ペインにドラッグ・アンド・ドロップします。
「ファイル・アダプタ構成ウィザード」が表示されます。
6. 新規の「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。
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。

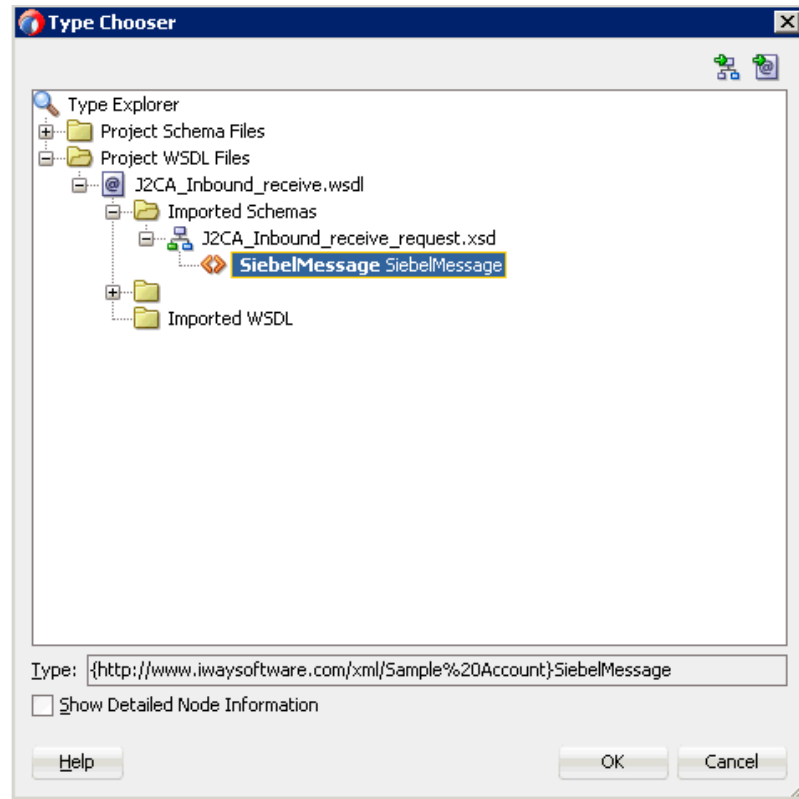
- 「操作」ページが表示されます。
9. 「次へ」をクリックします。
 10. 「操作タイプ」オプションのリストから「**Write File**」を選択し、「操作名」(「Write」など)を指定します。
 11. 「次へ」をクリックします。
「ファイル構成」ページが表示されます。
 12. ファイル・システムで出力ファイルが書き込まれる場所を指定します。
 13. ファイル・ネーミング規則フィールドで、出力ファイルの名前を指定します。
 14. 「次へ」をクリックします。
図 5-12 に示すように、「メッセージ」ページが表示されます。

図 5-12 「メッセージ」ページ



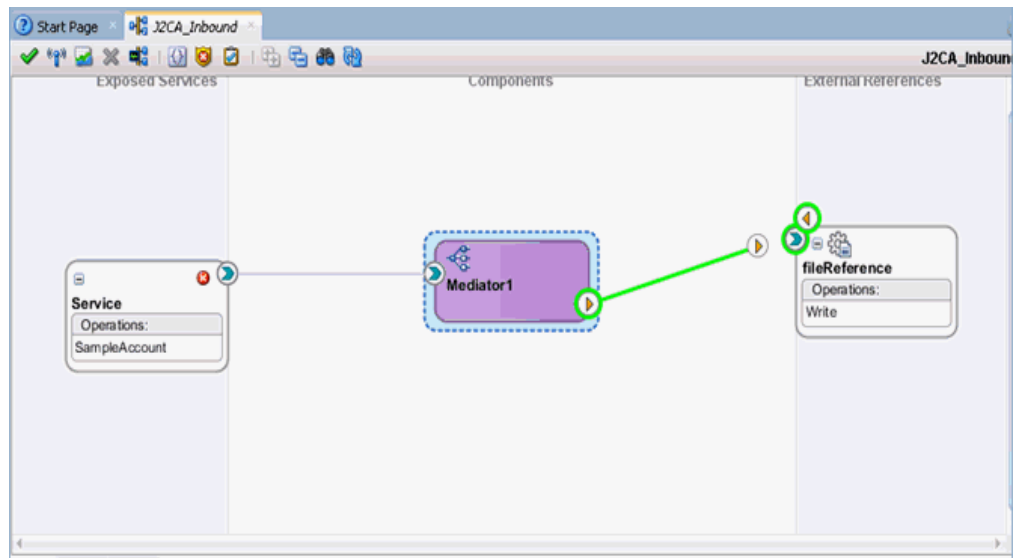
15. 「URL」フィールドの右の「参照」をクリックします。
図 5-13 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 5-13 「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」、「インポートしたスキーマ」、「J2CA_Inbound_receive_request.xsd」の順に展開し、「SiebelMessage SiebelMessage」を選択します。
17. 「OK」をクリックします。
「メッセージ」ページに戻ります。
18. 「次へ」をクリックします。
「終了」ページが表示されます。
19. 「終了」をクリックします。
20. インバウンド・メディエータ・プロセス・コンポーネントとサード・パーティ・アダプタ・サービス・コンポーネント間の接続を作成します。
21. 図 5-14 に示すように、インバウンド・メディエータ・プロセス・コンポーネントとファイル・アダプタ・コンポーネント間の接続を作成します。

図 5-14 接続の作成



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。

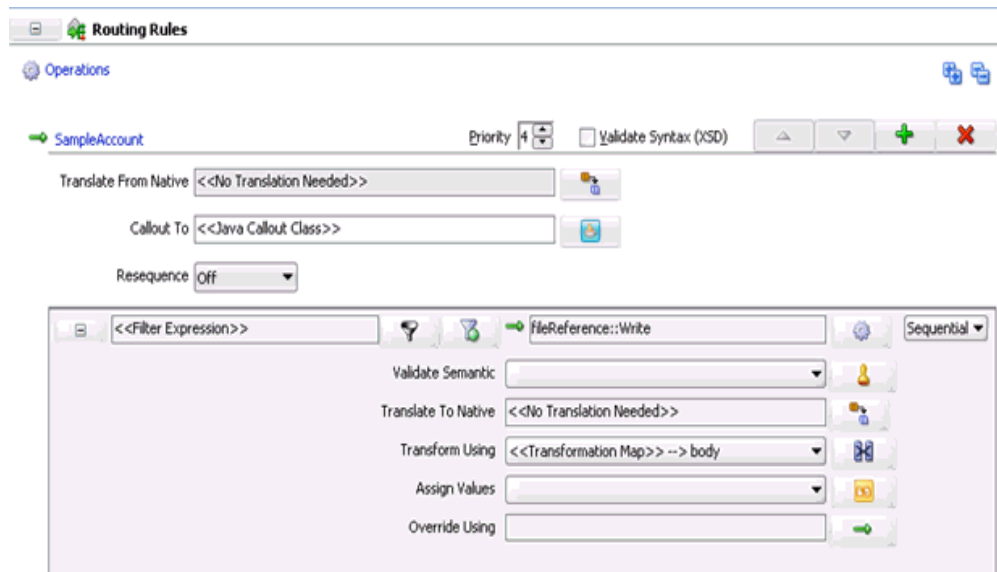
5.3.2.3 ルーティング・ルールの構成

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 5-15 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 5-15 「ルーティング・ルール」 ダイアログ



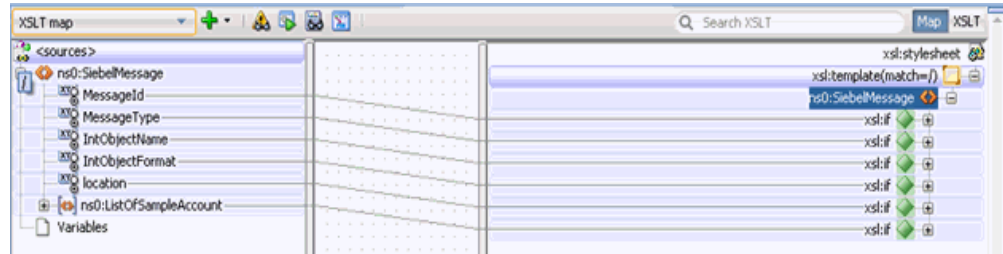
2. <<フィルタ式>> 領域で、「次を使用して変換」フィールドの右のアイコンをクリックします。

「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、選択した「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。

図 5-16 に示すように、マッピング・ページが表示されます。

図 5-16 マッピング・ページ



5. 「OK」をクリックします。
6. 「ns0:SiebelMessage」ソース要素を「ns0:SiebelMessage」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。
マッピングが完成します。
8. メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規インバウンド・メディエータ・プロセス・コンポーネントを保存します。

5.3.2.4 12c のデプロイメントに関する既知の問題の調整

12c のデプロイメントに関する既知の問題の調整方法の詳細は、4-26 ページ 4.4.3.3 項「12c のデプロイメントに関する既知の問題の調整」を参照してください。

これで、メディエータ・インバウンド・プロセスをデプロイする準備ができました。4-48 ページ 4.5.4 項「BPEL インバウンド・プロセスのデプロイ」の記載と同じ手順に従います。

Siebel を介してイベント・メッセージがトリガーされると、出力 XML が、ファイル・アダプタ・コンポーネントで指定された場所で受信されます。Siebel でのイベントのトリガーの詳細は、4-64 ページ 4.5.5.3 項「Siebel 8.0 でイベントをトリガーしてイベントのランタイム統合をテストする」を参照してください。

5.4 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、Siebel システム用にメディエータ・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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\Mediator\BSE\Outbound
Project
```

この項では、次のトピックについて説明します。

- 5.4.1 項「SOA 用の空のコンポジットの作成」
- 5.4.2 項「メディエータ・アウトバウンド・プロセスの定義」

前提条件

メディエータ・アウトバウンド・プロセスを設計する前に、アプリケーション・エクスプローラを使用して各 WSDL ファイルを生成する必要があります。詳細は、4-76 ページ [4.6.1 項「Web サービスを使用するリクエスト / レスポンス型 サービス用の WSDL ファイルの生成」](#) を参照してください。

5.4.1 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、次のステップを実行します

1. 新規の SOA アプリケーションを作成します。
2. SOA アプリケーションの名前 (IBSE_OUTBOUND など) を入力し、「次へ」をクリックします。
3. プロジェクトの名前 (IBSE_Outbound など) を入力し、「次へ」をクリックします。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

詳細は、4-9 ページ [4.4.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

5.4.2 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・プロセスの定義方法について説明します。内容は、次のとおりです。

- [5.4.2.1 項「SOAP サービスの構成」](#)
- [5.4.2.2 項「メディエータ・コンポーネントの作成」](#)
- [5.4.2.3 項「ルーティング・ルールの構成」](#)

5.4.2.1 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・アダプタ」ペインから「SOAP」ノードを「外部参照」ペインにドラッグ・アンド・ドロップします。
2. SOAP サービスに適切な名前を入力し、「WSDL URL」フィールドの右の**既存の WSDL を検索**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」タブを選択し、アプリケーション・エクスプローラから WSDL がエクスポートされる場所へ移動して WSDL を選択し、「OK」をクリックします。
4. 「Web サービスの作成」ウィンドウで、「OK」をクリックします。
5. 表示された「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。

Web サービスが作成され、表示されます。

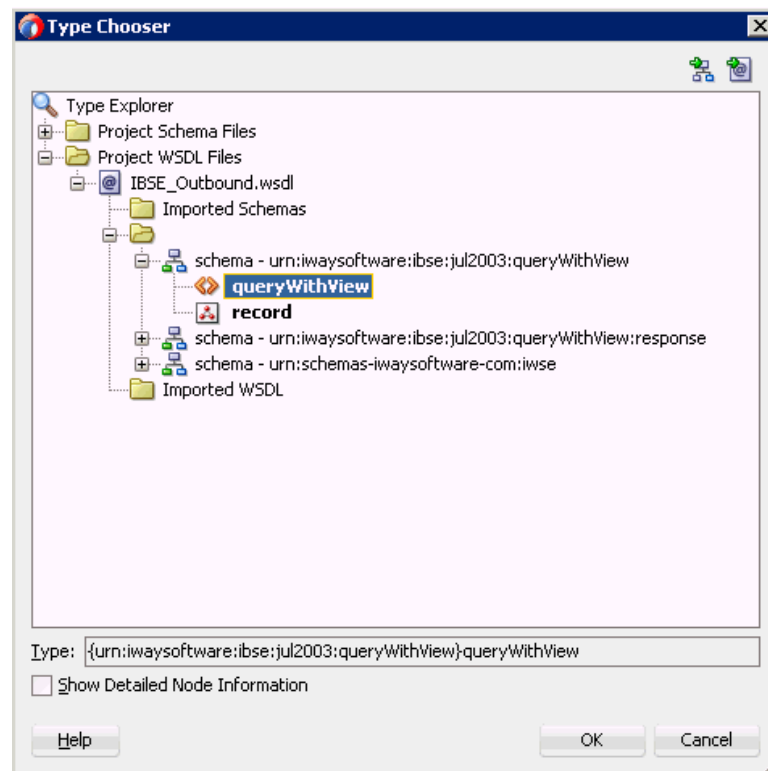
5.4.2.2 メディエータ・コンポーネントの作成

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. 「コンポーネント」ペインから**メディエータ・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。

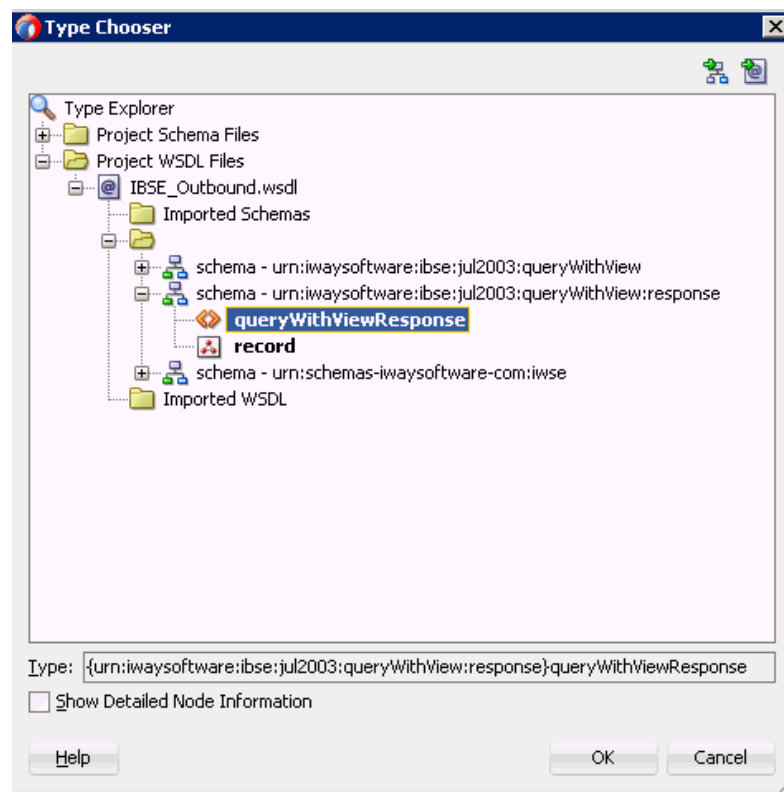
2. 「名前」フィールドに、新規のアウトバウンド・メディアータ・プロセス・コンポーネントの名前を入力します。
3. 「テンプレート」ドロップダウン・リストで、「同期インタフェース」を選択します。
4. 関連する XML リクエスト・スキーマ・ファイルの選択用の「入力」フィールドの右の「参照」アイコンをクリックします。
5. 図 5-17 に示すように、「タイプ・チューザ」で「プロジェクトの WSDL ファイル」を展開し、「IBSE_Outbound.wsdl」を選択して、「queryWithView」をクリックします。

図 5-17 「タイプ・チューザ」ダイアログ



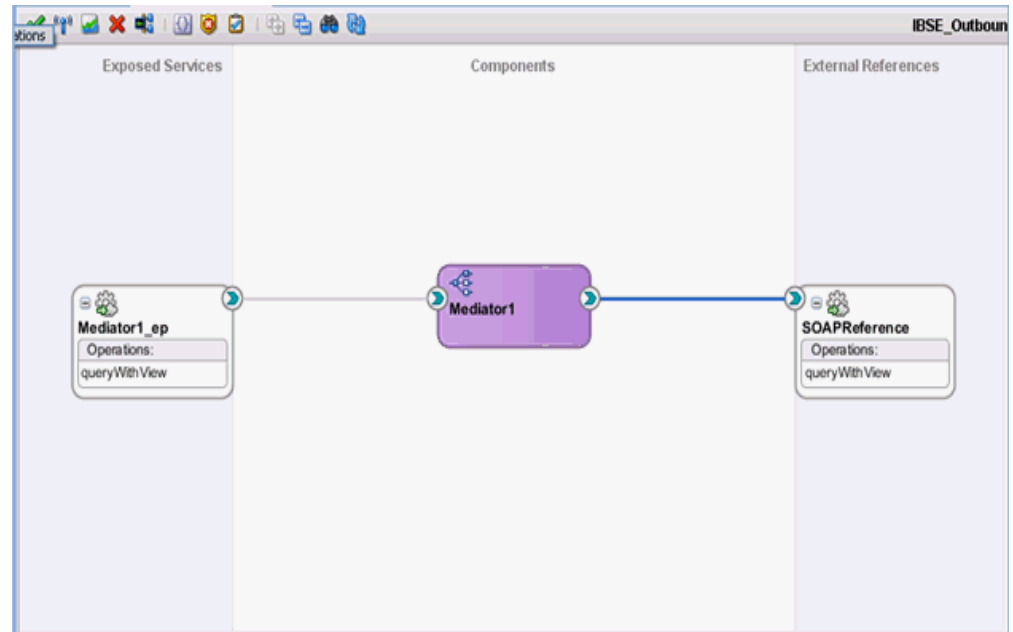
6. 「OK」をクリックします。
7. 関連する XML レスポンス・スキーマ・ファイルの選択用の「出力」フィールドの右の「参照」アイコンをクリックします。
8. 図 5-18 に示すように、「タイプ・チューザ」ダイアログで「プロジェクトの WSDL ファイル」を展開し、「IBSE_Outbound.wsdl」を選択して、「queryWithViewResponse」をクリックします。

図 5-18 「タイプ・チューザ」 ダイアログ



9. 「OK」をクリックします。
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 5-19 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 5-19 接続の作成

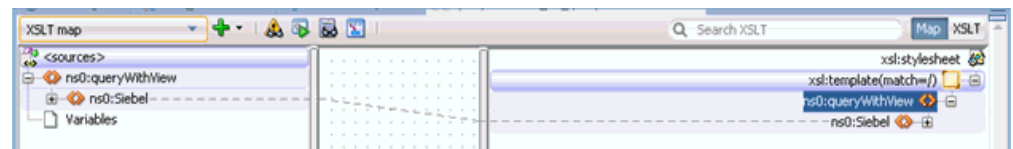


5.4.2.3 ルーティング・ルール構成

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. 表示された「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・ボックスで選択した「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 5-20 に示すように、「ns0:queryWithView」ソース要素を「ns0:queryWithView」ターゲット要素にマップします。

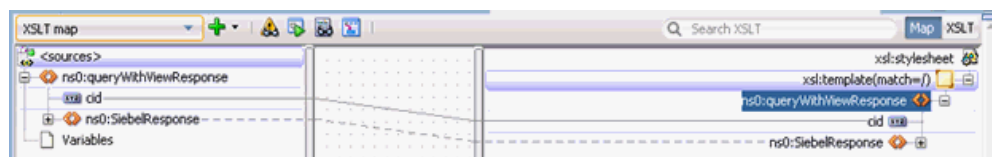
図 5-20 「GetDetail_To_GetDetail.xsl」タブ



6. 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。
7. 「同期リプライ」フィールドの右のアイコンをクリックします。
8. 表示された「リプライ・トランスフォーメーション・マップ」ウィンドウで「追加」(+)アイコンをクリックし、「トランスフォーメーション・マップの作成」ダイアログ・ボックスで選択した「タイプ」が「XSLT」であることを確認し、「OK」をクリックします。

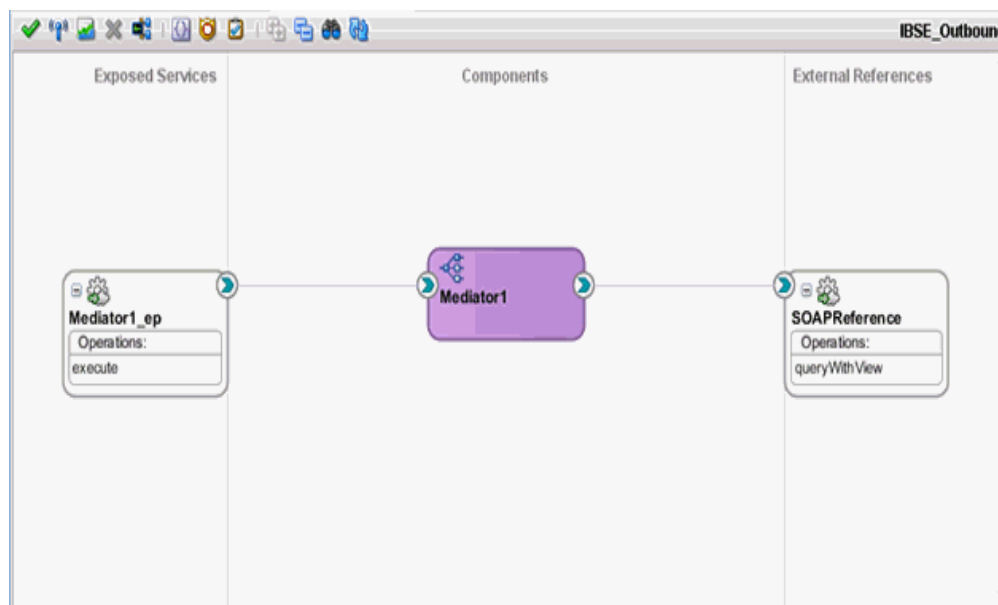
9. 図 5-21 に示すように、「ns0:queryWithViewResponse」ソース要素を「ns0:queryWithViewResponse」ターゲット要素にマップします。

図 5-21 ソース要素とターゲット要素



10. 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。
11. 左のペインで「composite.xml」をダブルクリックします。
12. 図 5-22 に示すように、メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規アウトバウンド・メディエータ・コンポーネントを保存します。

図 5-22 「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。5-10 ページ 5.2.3 項「メディエータ・アウトバウンド・プロセスのデプロイ」の記載と同じ手順に従います。

デプロイすると、5-11 ページ 5.2.4 項「Oracle Enterprise Manager Console での入力 XML ドキュメントの起動」に定義されているように、入力 XML を起動できるようになります。

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

Oracle Application Adapter for Siebel では、Oracle Business Process Management (BPM) がシームレスに統合されており、Web サービスの簡易な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。

この章の内容は次のとおりです。

- 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「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)」
- 6.5 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)」
- 6.6 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)」

6.1 概要

Oracle BPM を統合するには、Oracle Application Adapter for Siebel 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。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) で設計時に生成される WSDL ファイルとして公開される必要があります。詳細は、2-27 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・サービスに適した 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する必要があります。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。

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 Siebel 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。

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、4-2 ページ 4.3 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。

6.4 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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\BPM\J2CA\Siebel_Sample_J2CA_BPM_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 が使用されています。

この項では、次のトピックについて説明します。

- [6.4.1 項「BPM 用の空のコンポジットの作成」](#)
- [6.4.2 項「BPM アウトバウンド・プロセスの定義」](#)
- [6.4.3 項「12c のデプロイメントに関する既知の問題の調整」](#)
- [6.4.4 項「BPM アウトバウンド・プロセスのデプロイ」](#)
- [6.4.5 項「Oracle Enterprise Manager Console 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対応する WSDL ファイルを生成する必要があります。詳細は、4-8 ページ [4.4.1 項「リクエスト/レスポンス・サービス用の WSDL の生成」](#) を参照してください。

6.4.1 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 アプリケーションの名前を入力し、「[次へ](#)」をクリックします。
「プロジェクトの名前付け」ページが表示されます。

3. 「プロジェクトの機能」でプロジェクトの名前を入力し、「BPM」を選択して「次へ」をクリックします。
「SOA」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。

6.4.2 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。

- 6.4.2.1 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」
- 6.4.2.2 項「アウトバウンド BPM プロセス・コンポーネントの構成」
- 6.4.2.3 項「書込み操作のファイル・アダプタの作成」

6.4.2.1 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読み込みます。
2. カスタム/サード・パーティ・ペインから「サード・パーティ・アダプタ」コンポーネントを「外部参照」ペインにドラッグ・アンド・ドロップします。
「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。
3. サード・パーティ・アダプタ・サービスの名前を入力します。
4. 「タイプ」リストで「参照」が選択されていることを確認します(デフォルト)。
5. 「WSDL URL」フィールドの右の既存の WSDL を検索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。
6. 次のディレクトリでアウトバウンド WSDL ファイルを特定し、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は、4-11 ページ [4.4.3.1 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。

6.4.2.2 アウトバウンド BPM プロセス・コンポーネントの構成

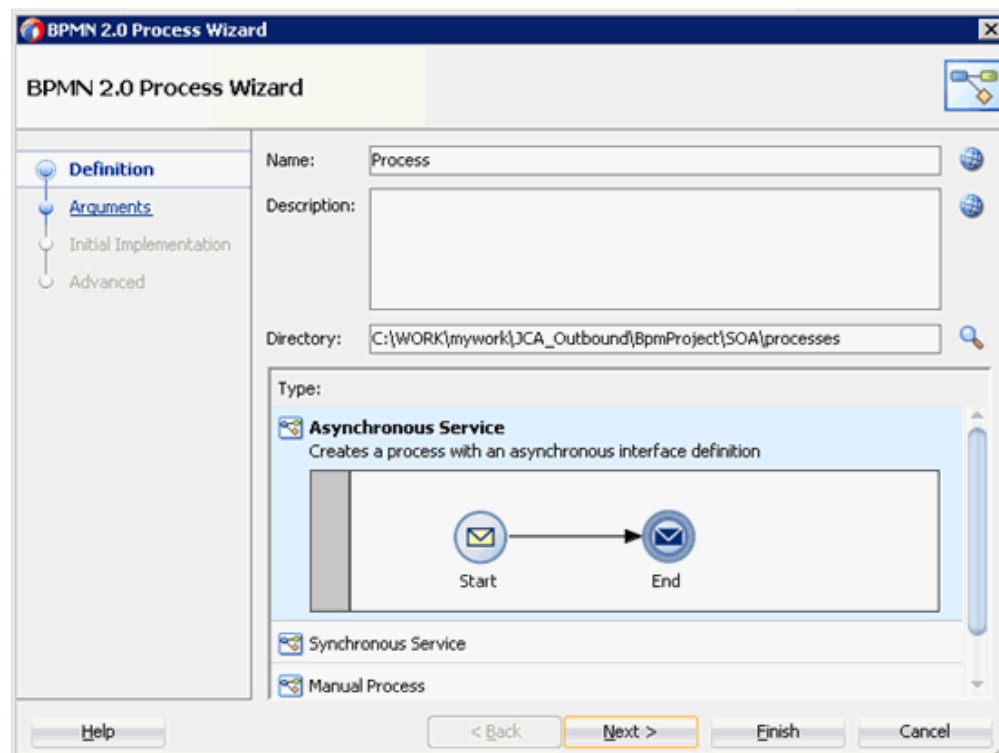
この項では、アウトバウンド BPM プロセス・コンポーネントの構成方法について説明します。

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「BPMN 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 6-1 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。

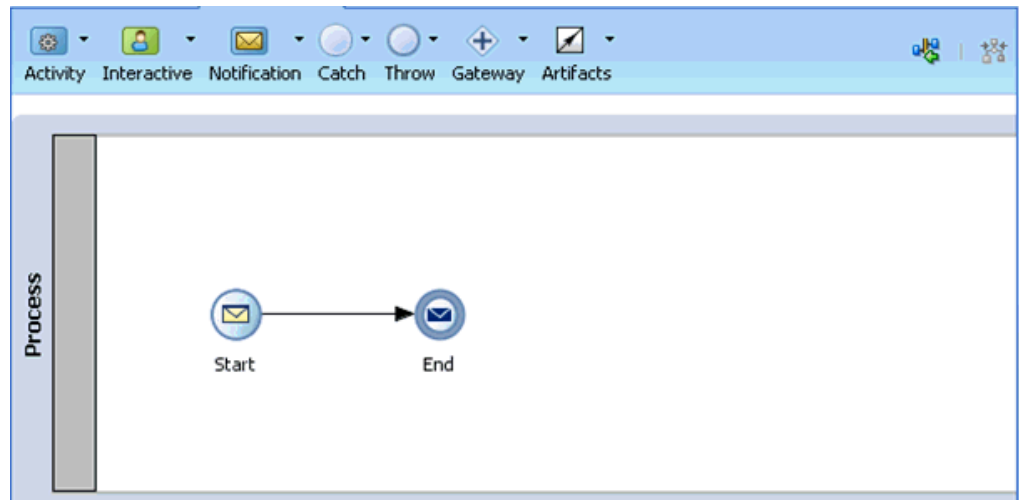
図 6-1 「BPMN プロセスの作成」ダイアログ



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。

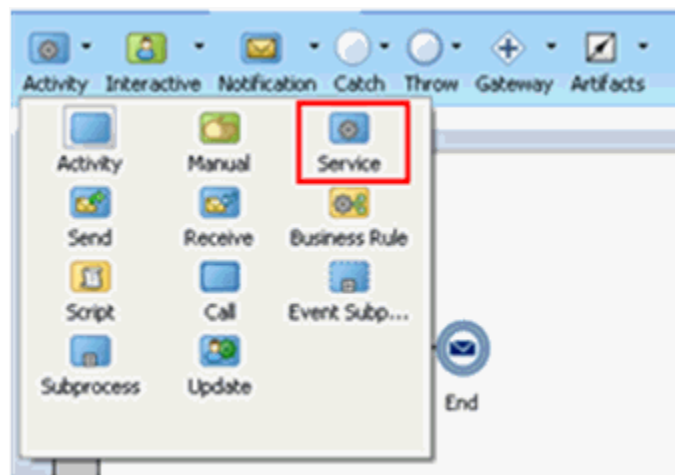
図 6-2 に示すように、BPMN プロセスが表示されます。

図 6-2 BPMN プロセス



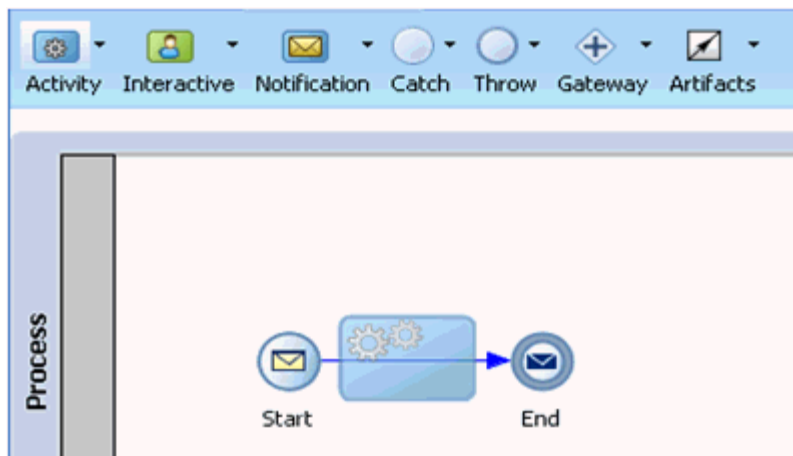
3. 図 6-3 に示すように、「アクティビティ」ドロップダウン・メニューをクリックして「サービス」を選択します。

図 6-3 「アクティビティ」ドロップダウン・メニュー



4. 図 6-4 に示すように、「開始」アイコンをドロップします。

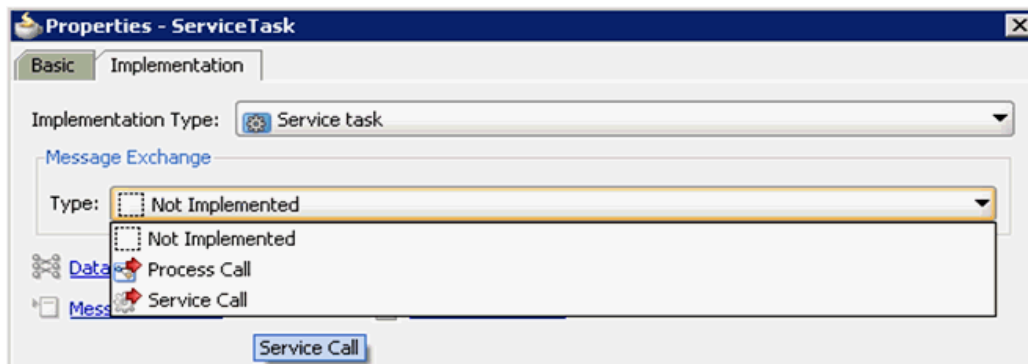
図 6-4 「アクティビティ」アイコン



「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 6-5 に示すように、「メッセージ交換」の「タイプ」リストから「サービス・コール」を選択します。

図 6-5 サービス・コール



7. 図 6-6 に示すように、「サービス」アイコンをクリックします。

図 6-6 「参照」アイコン

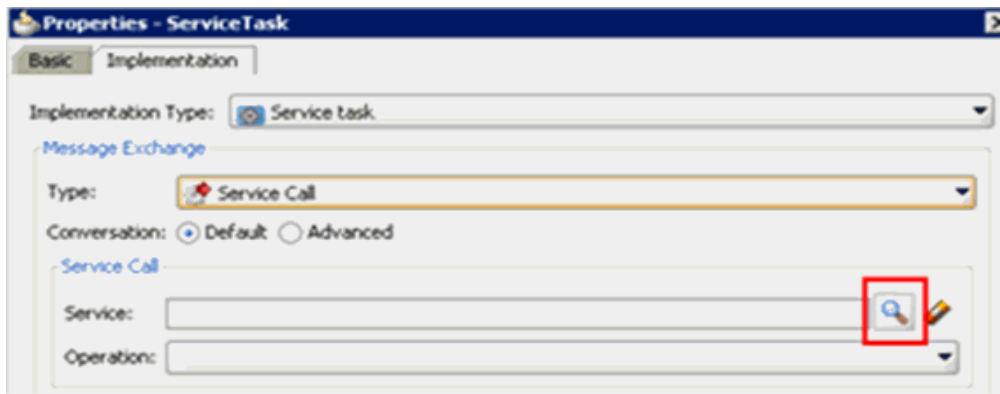
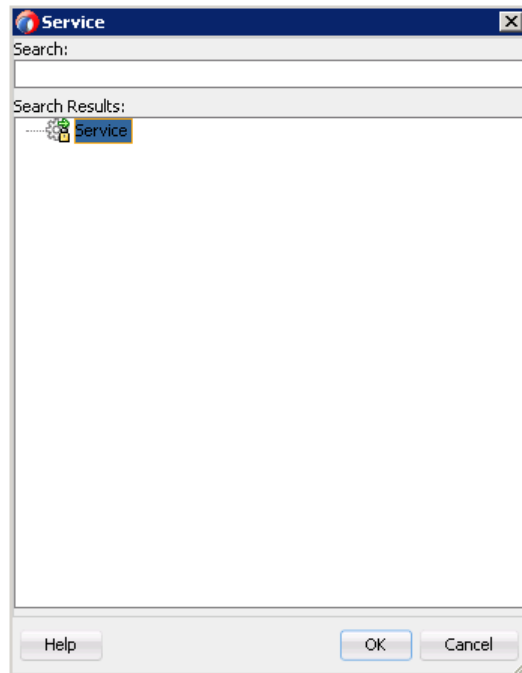


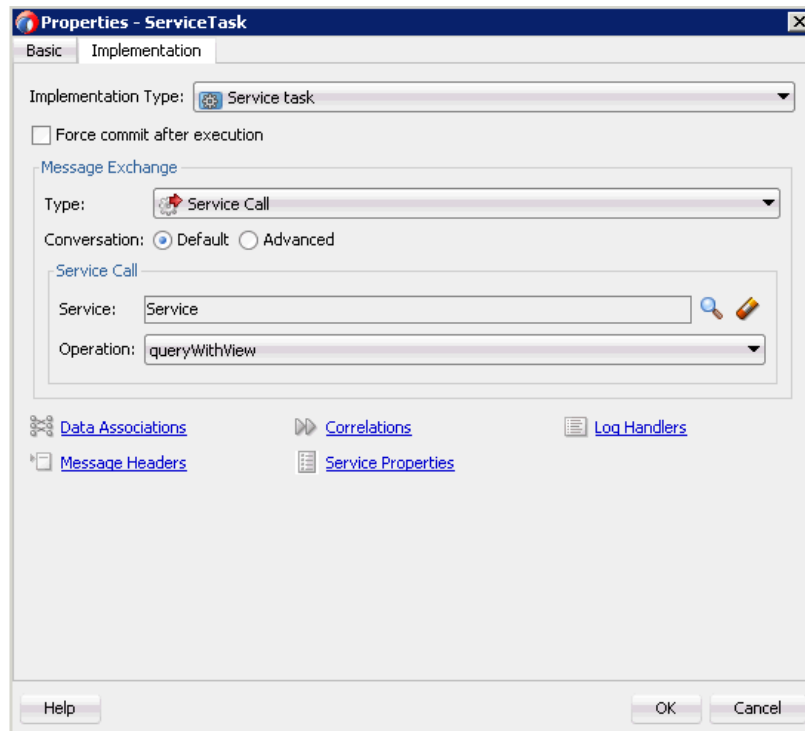
図 6-7 に示すように、「サービス」ダイアログが表示されます。

図 6-7 「サービス」 ダイアログ



- 作成されたサード・パーティ・サービスを選択して「OK」をクリックします。
図 6-8 に示すように、「プロパティ - ServiceTask」ダイアログに戻ります。

図 6-8 「プロパティ - ServiceTask」ダイアログ



- 「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 6-9 に示すように、左ペインの「プロセス」の下の「データ・オブジェクト」ノードを右クリックし、「新規」を選択します。

図 6-9 「新規」オプション

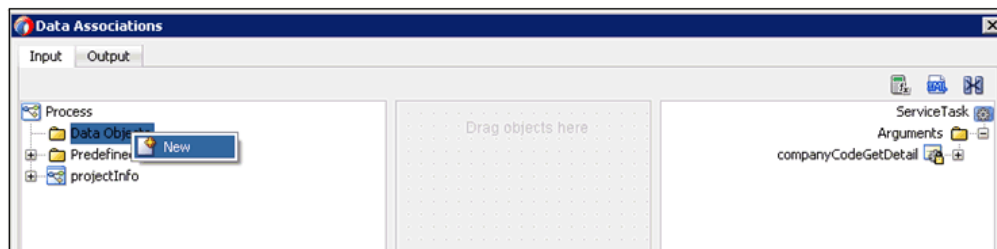
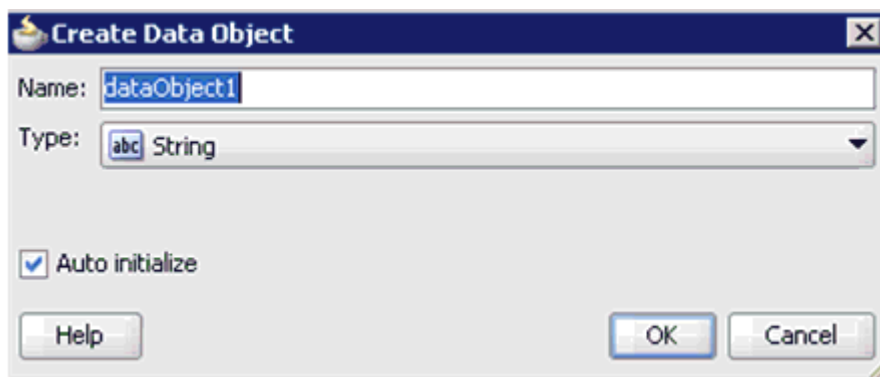


図 6-10 に示すように、「データ・オブジェクトの作成」ダイアログが表示されます。

図 6-10 「データ・オブジェクトの作成」ダイアログ



11. 図 6-11 に示すように、「名前」フィールドに名前を入力し（「Request」など）、「タイプ」フィールドのドロップダウン・ボタンをクリックし、リストから「参照」を選択します。

図 6-11 「データ・オブジェクトの作成」ダイアログ

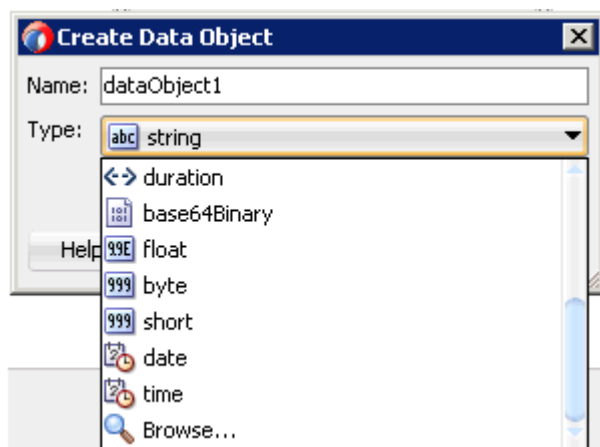
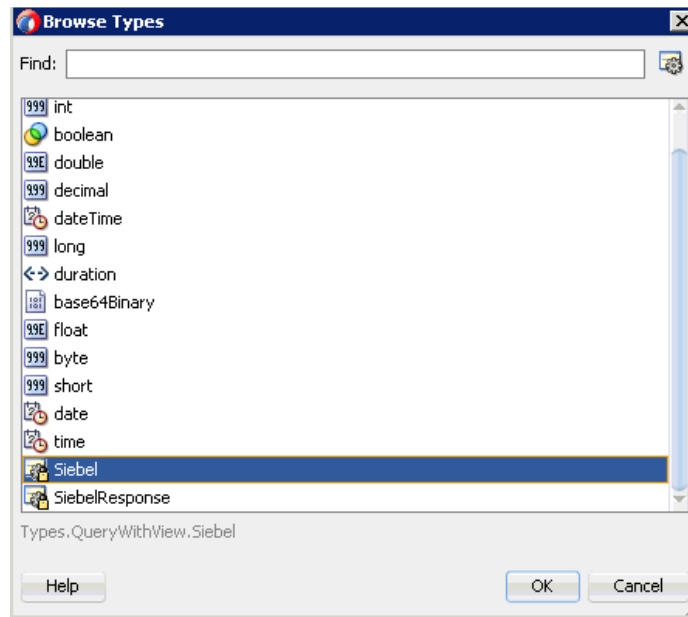


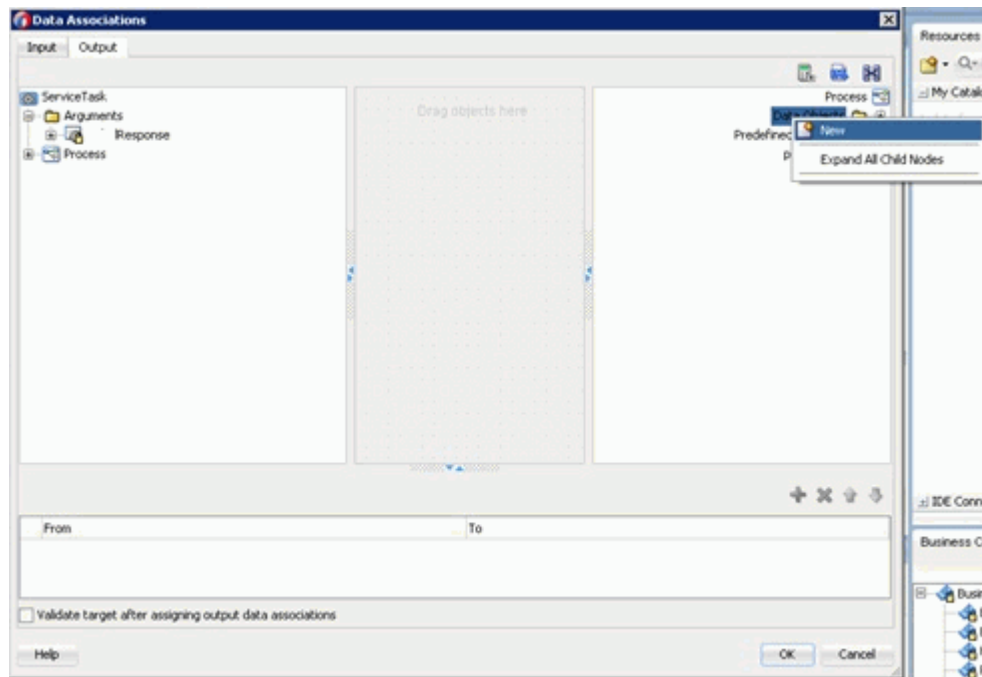
図 6-12 に示すように、「参照タイプ」ダイアログが表示されます。

図 6-12 「参照タイプ」 ダイアログ



12. 1つ目のコンポーネント (Siebel など) を選択し、「OK」をクリックします。
「データ・オブジェクトの作成」ダイアログに戻ります。
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」ノードの下に表示されます。
14. 図 6-13 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 6-13 「データ・アソシエーション」 ダイアログ

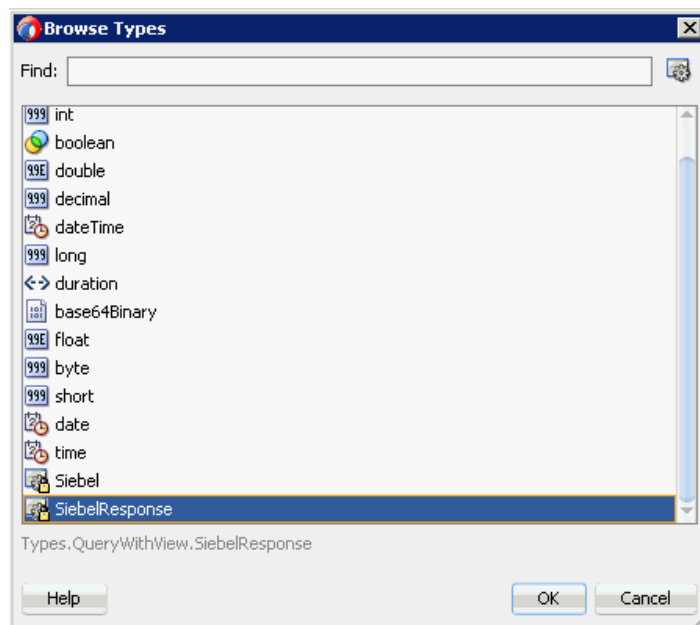


「データ・オブジェクトの作成」ダイアログが表示されます。

15. 「名前」フィールドに名前を入力し(「Response」など)、「タイプ」フィールドのドロップダウン・ボタンをクリックし、リストから「参照」を選択します。

図 6-14 に示すように、「参照タイプ」ダイアログが表示されます。

図 6-14 「参照タイプ」ダイアログ



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

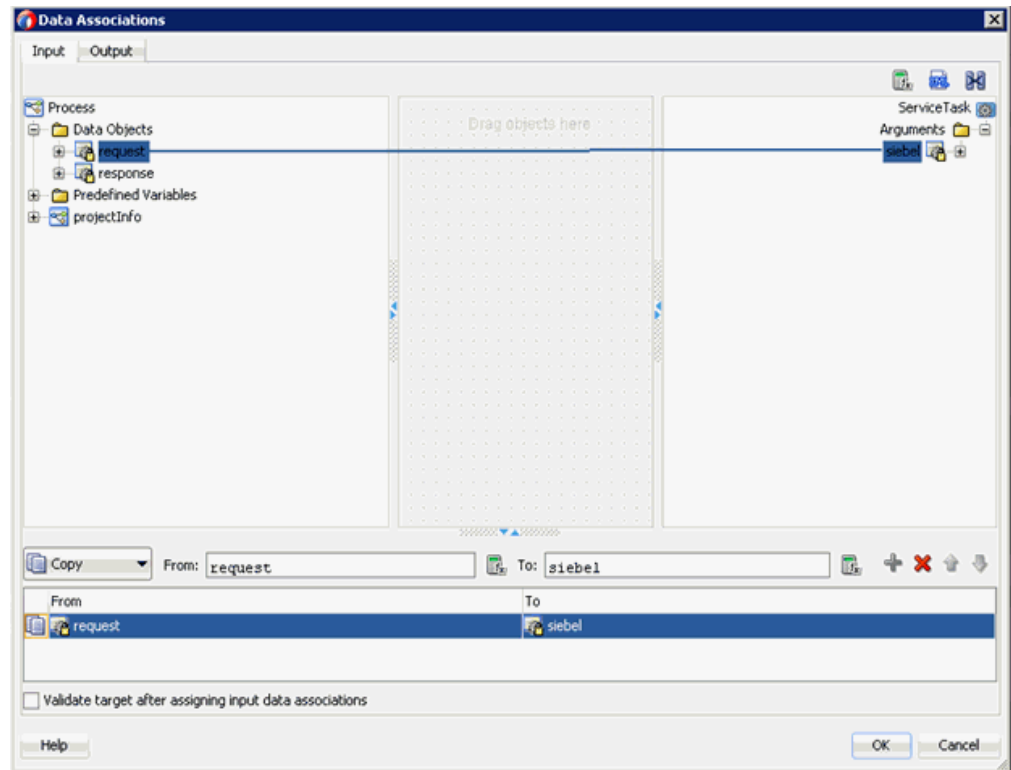
「データ・オブジェクトの作成」ダイアログに戻ります。

17. 「OK」をクリックします。

作成されたデータ・オブジェクト (「Response」 など) が 「データ・アソシエーション」 ノードの下に表示されます。

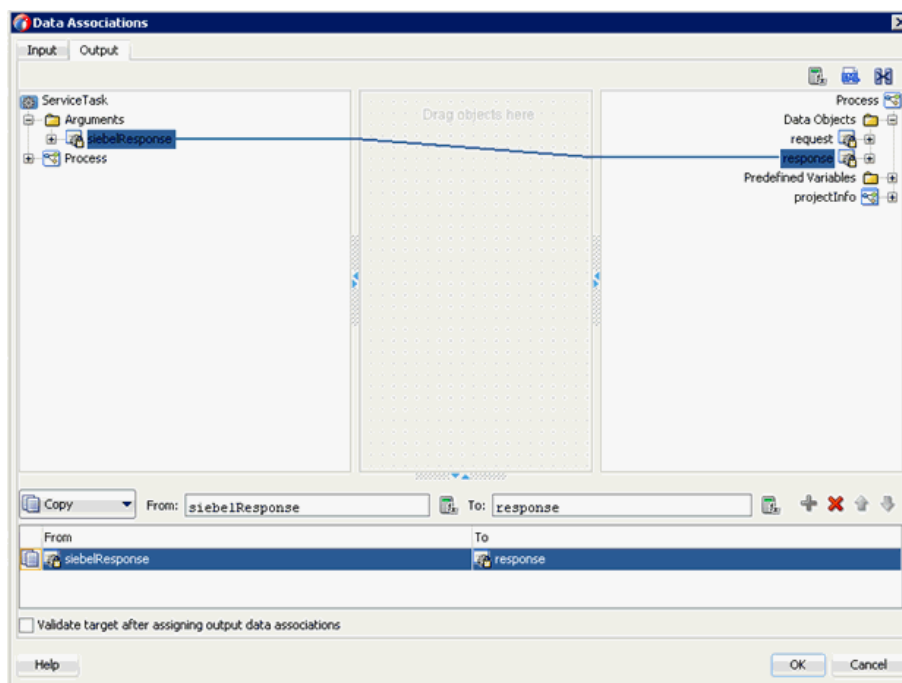
18. 図 6-15 に示すように、「入力」タブの左ペインにある「データ・オブジェクト」ノードの下の **Request** データ・オブジェクトを選択し、右ペインの「引数」ノードの下の「Siebel」にドラッグして接続します。

図 6-15 Request データ・オブジェクト



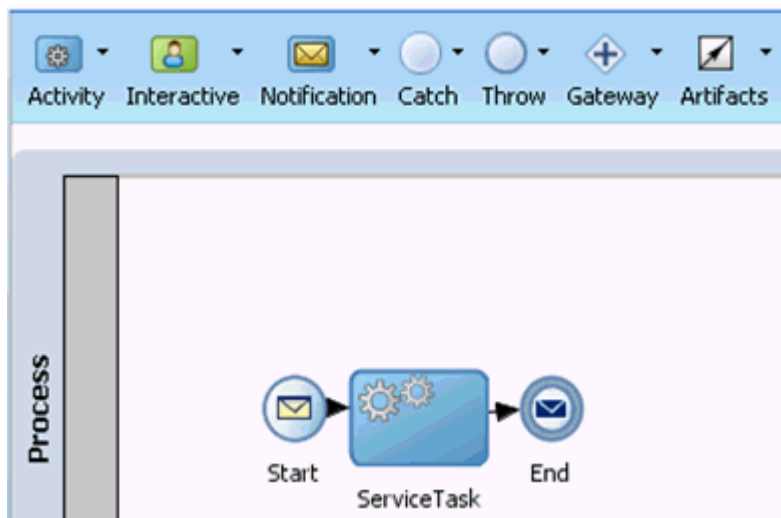
19. 図 6-16 に示すように、「出力」タブをクリックし、左ペインの「引数」ノードの下の「SiebelResponse」を選択し、「データ・オブジェクト」ノードの下の Response データ・オブジェクトにドラッグして接続します。

図 6-16 Response データ・オブジェクト



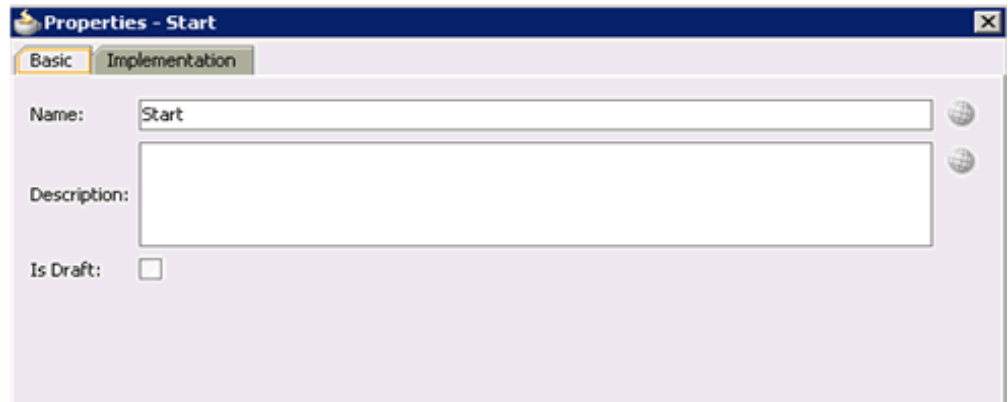
20. 「OK」をクリックします。
「プロパティ - ServiceTask」ダイアログに戻ります。
21. 「OK」をクリックします。
図 6-17 に示すように、「開始」が作成されます。

図 6-17 サービス・タスク



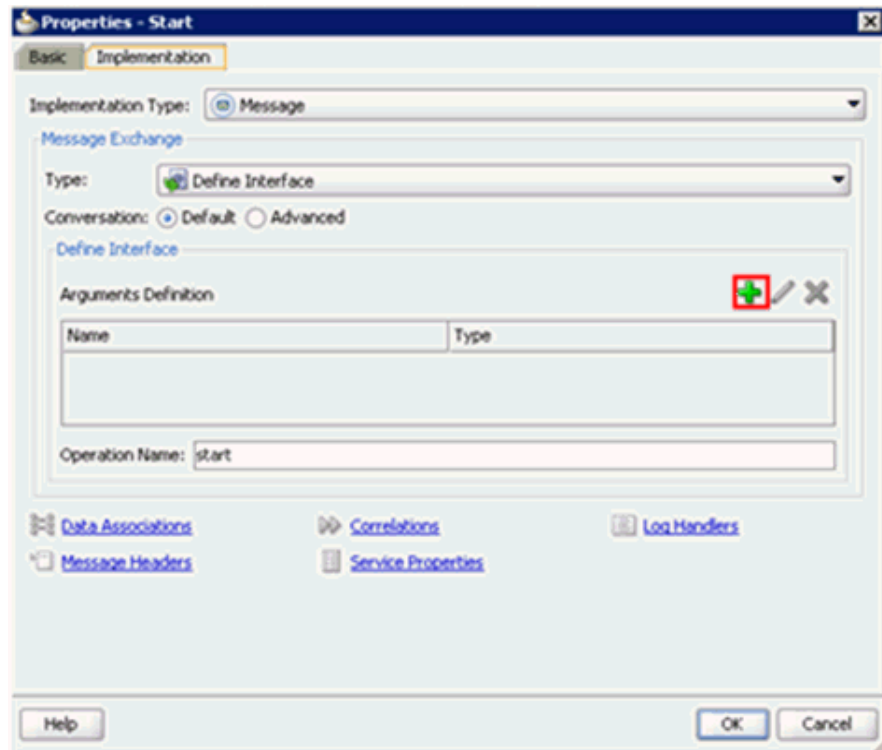
22. プロセスを保存し、「開始」イベント・コンポーネントをダブルクリックします。
図 6-18 に示すように、「プロパティ - 開始」ダイアログが表示されます。

図 6-18 「プロパティ - 開始」 ダイアログ



23. 図 6-19 に示すように、「実装」タブをクリックします。

図 6-19 「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 6-20 に示すように、「名前」フィールドに名前を入力し (デフォルトでは「argument1」)、 「タイプ」フィールドのドロップダウン・ボタンをクリックし、リストから「参照」を選択します。

図 6-20 「引数の作成」 ダイアログ

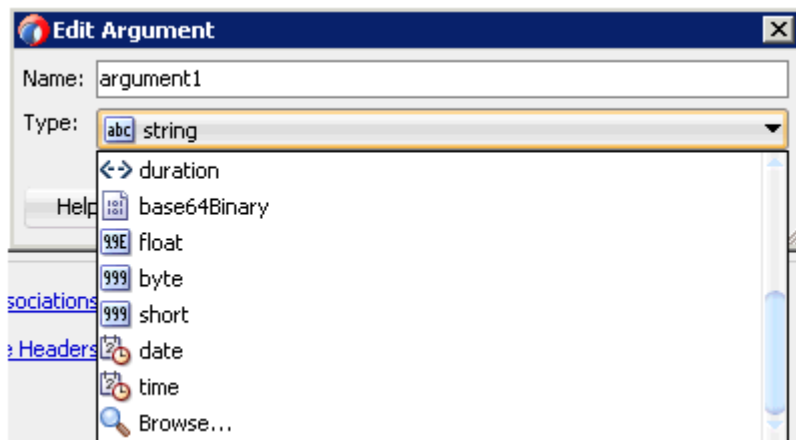
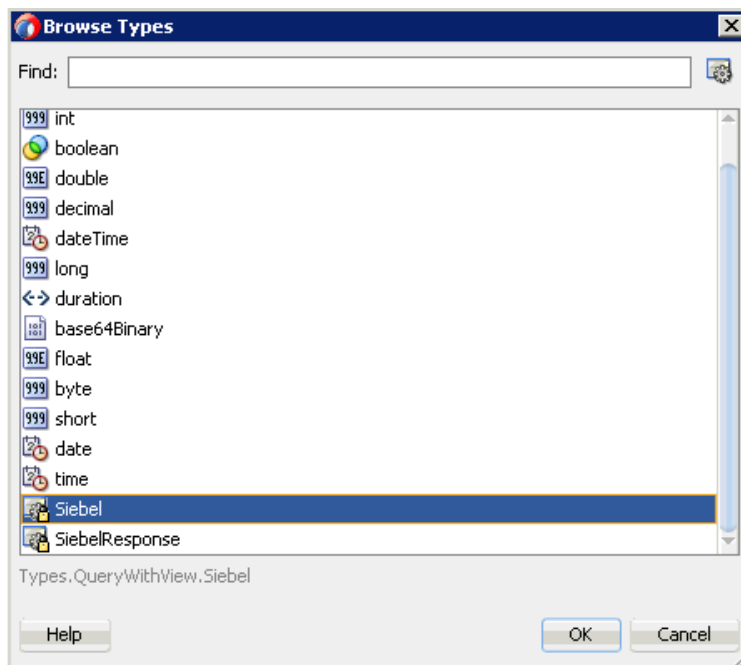


図 6-21 に示すように、「参照タイプ」ダイアログが表示されます。

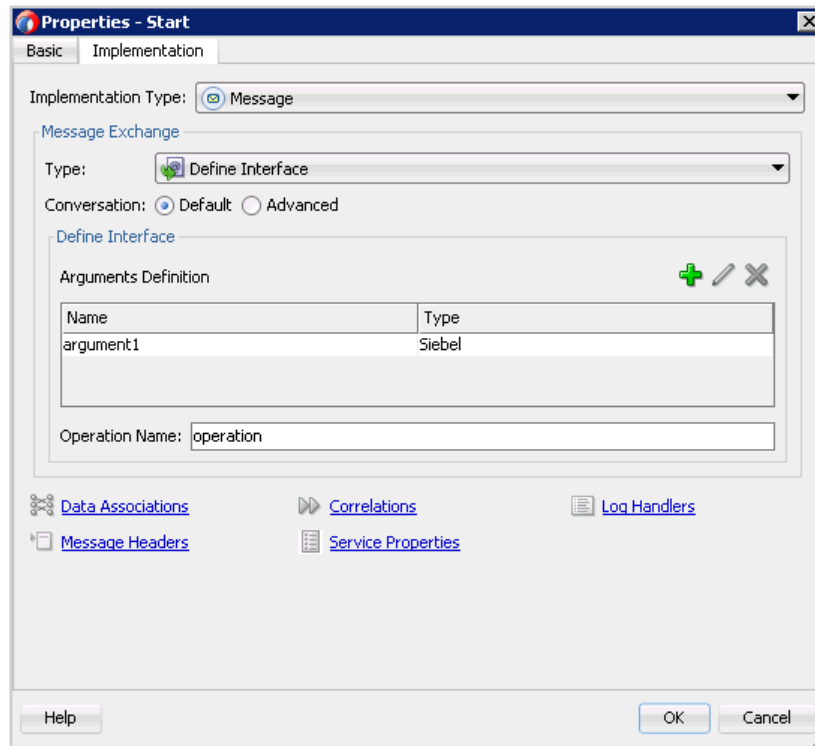
図 6-21 「参照タイプ」ダイアログ



26. 1つ目のコンポーネント (Siebel など) を選択し、「OK」をクリックします。
「引数の作成」ダイアログに戻ります。
27. 「OK」をクリックします。
「プロパティ - 開始」ダイアログに戻ります。
28. 図 6-22 に示すように、「操作名」フィールドで **start**(デフォルト) を **operation** に変更します。

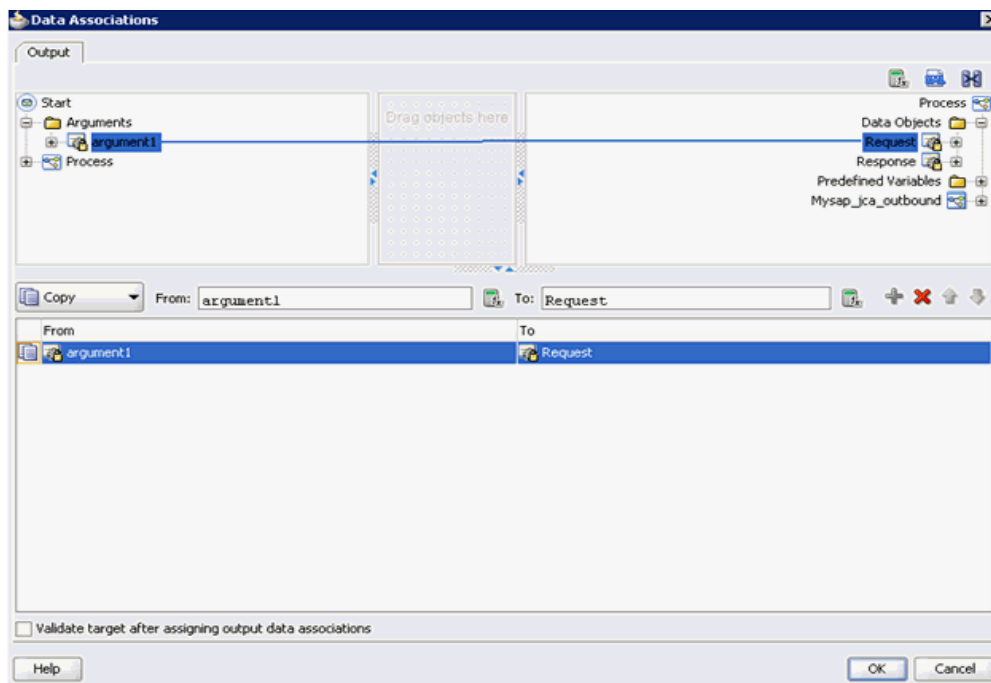
注意: この変更は古い BPM ペイロードを操作するために必要です。

図 6-22 「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 **Request** データ・オブジェクトにドラッグして接続します。
31. 図 6-23 に示すように、「OK」をクリックします。

図 6-23 「OK」 ボタン

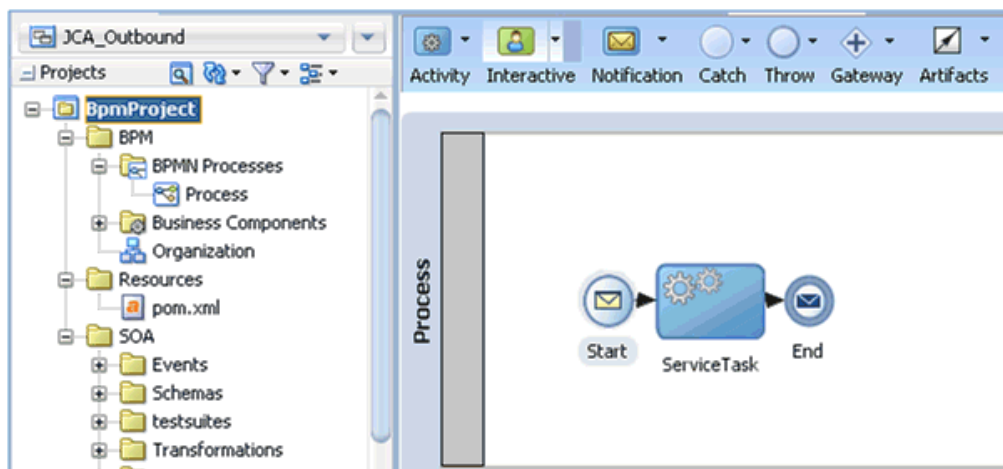


「プロパティ - 開始」 ダイアログに戻ります。

32. 「OK」 をクリックします。

図 6-24 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 6-24 「プロセス」 ワークスペース領域



33. 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読み込みます。
34. メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規アウトバウンド BPM プロセス・コンポーネントを保存します。
- これで書込み操作のファイル・アダプタを作成する準備ができました。

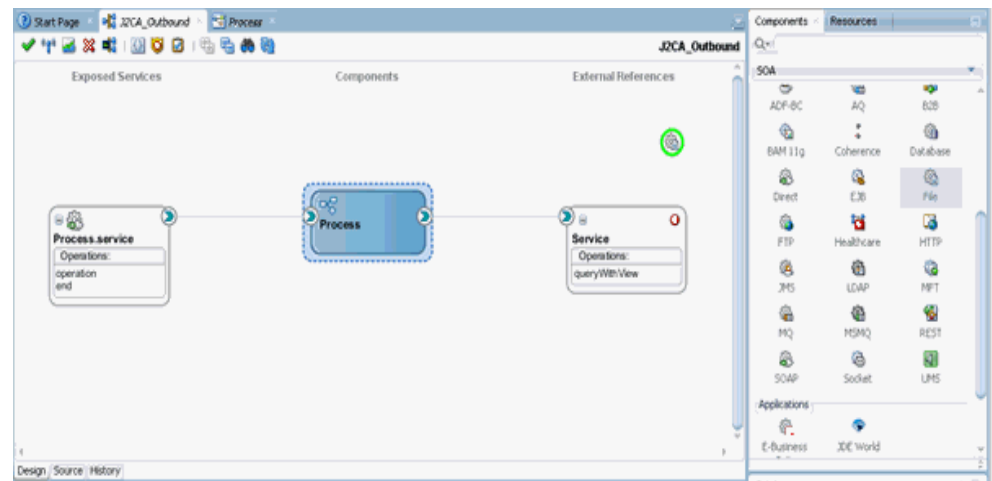
6.4.2.3 書込み操作のファイル・アダプタの作成

この項では、書込み操作のファイル・アダプタの作成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 6-25 に示すように、「テクノロジ・アダプタ」ペインから「ファイル・アダプタ」コンポーネントを「外部参照」ペインにドラッグ・アンド・ドロップします。

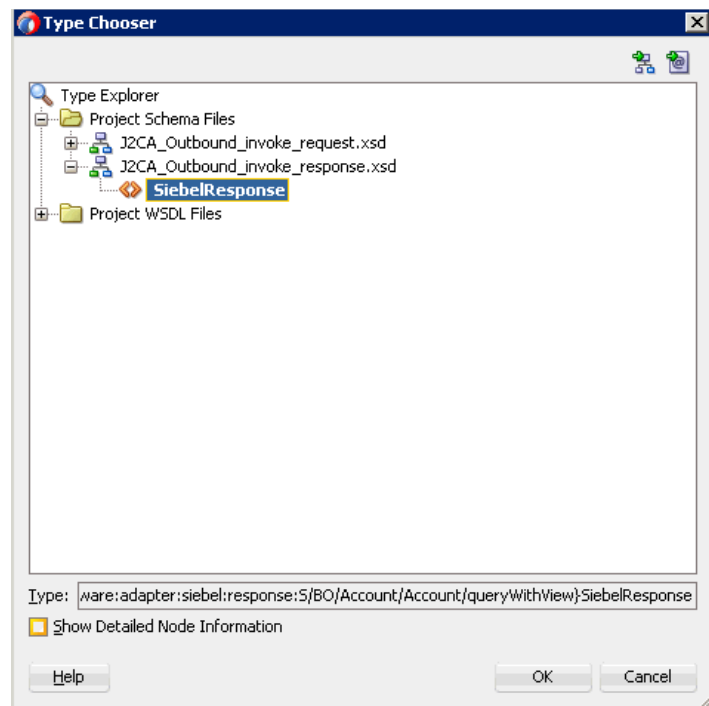
図 6-25 「ファイル・アダプタ」コンポーネント



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. ファイル・システムで出力ファイルが書き込まれる場所を指定します。
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 6-26 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 6-26 「タイプ・チューザ」 ダイアログ



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. 使用可能なスキーマを選択します (SiebelResponse など)。

15. 「OK」をクリックします。

「メッセージ」ページに戻ります。

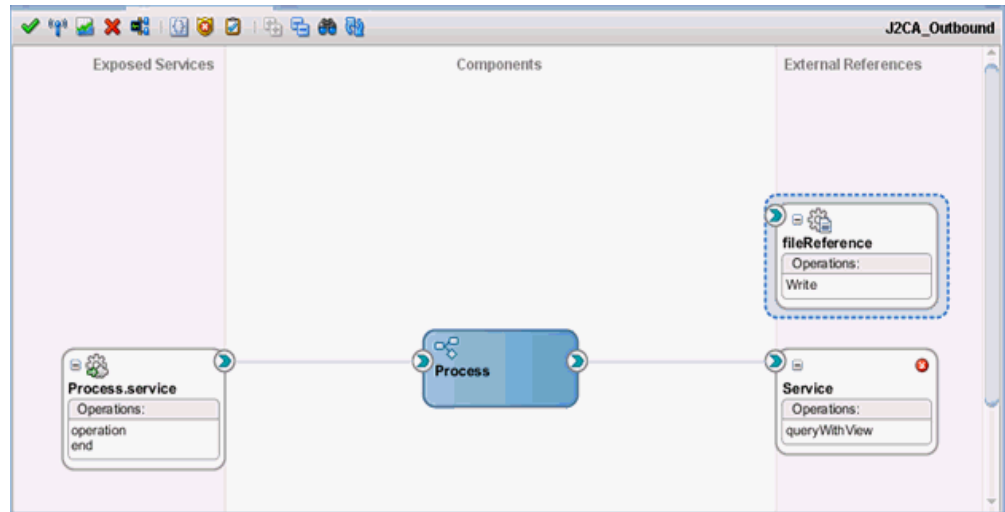
16. 「次へ」をクリックします。

「終了」ページが表示されます。

17. 「終了」をクリックします。

図 6-27 に示すように、「外部参照」ペインにファイル・アダプタ・サービスが作成されます。

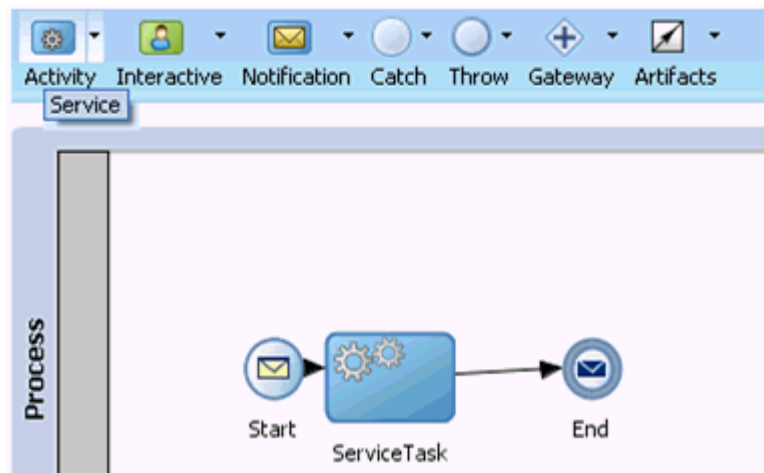
図 6-27 ファイル・アダプタ・サービス



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 6-28 に示すように、BPMN プロセスが表示されます。

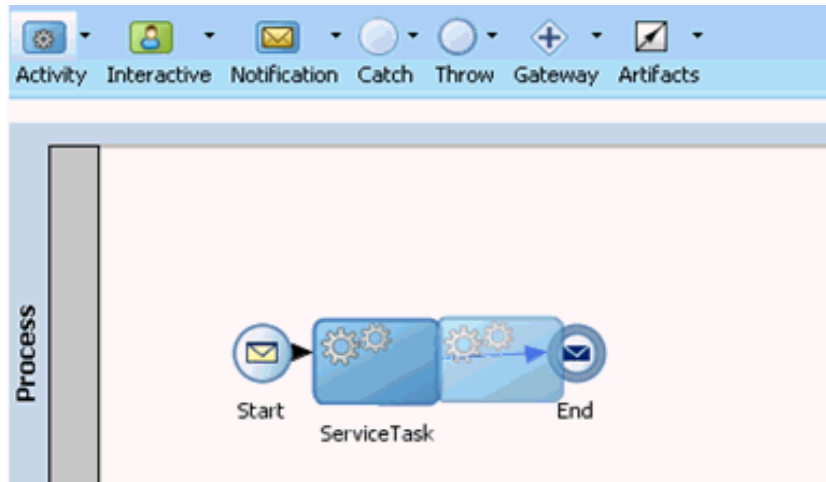
図 6-28 BPMN プロセス



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 6-29 に示すように、「サービス・タスク」アイコンをドロップします。

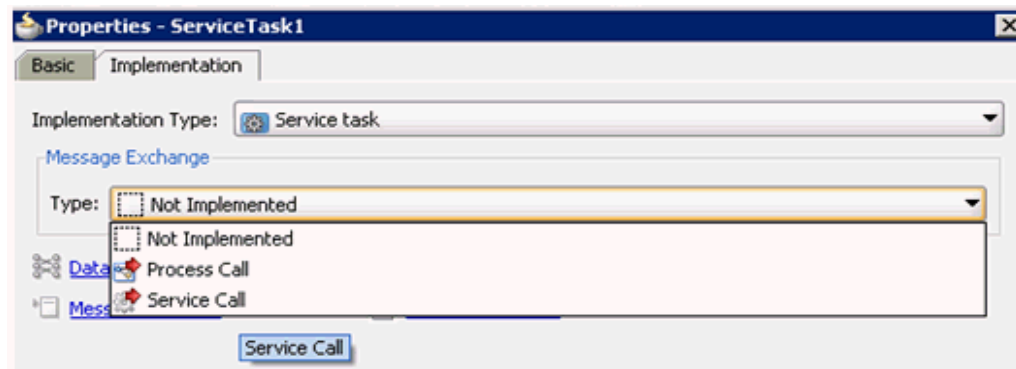
図 6-29 「アクティビティ」アイコン



「プロパティ - ServiceTask1」ダイアログが表示されます。

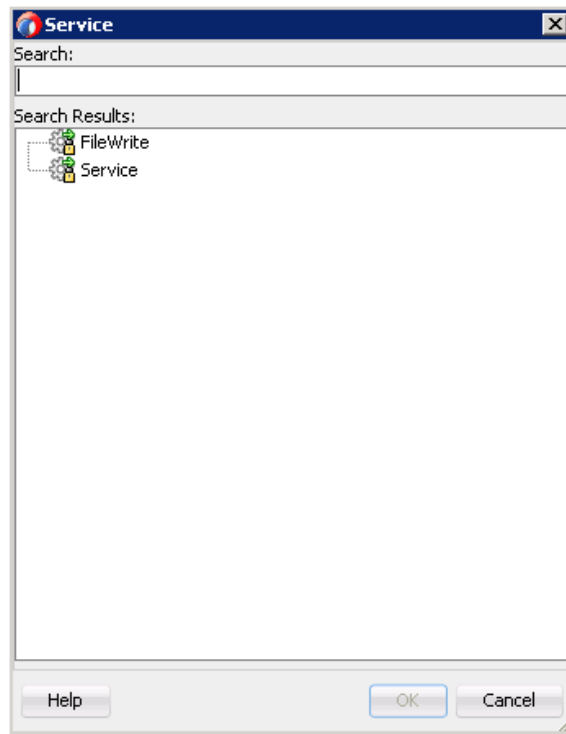
21. 「実装」タブをクリックします。
22. 図 6-30 に示すように、「メッセージ交換」セクションの「タイプ」ドロップダウン・リストから「サービス・コール」を選択します。

図 6-30 サービス・コール



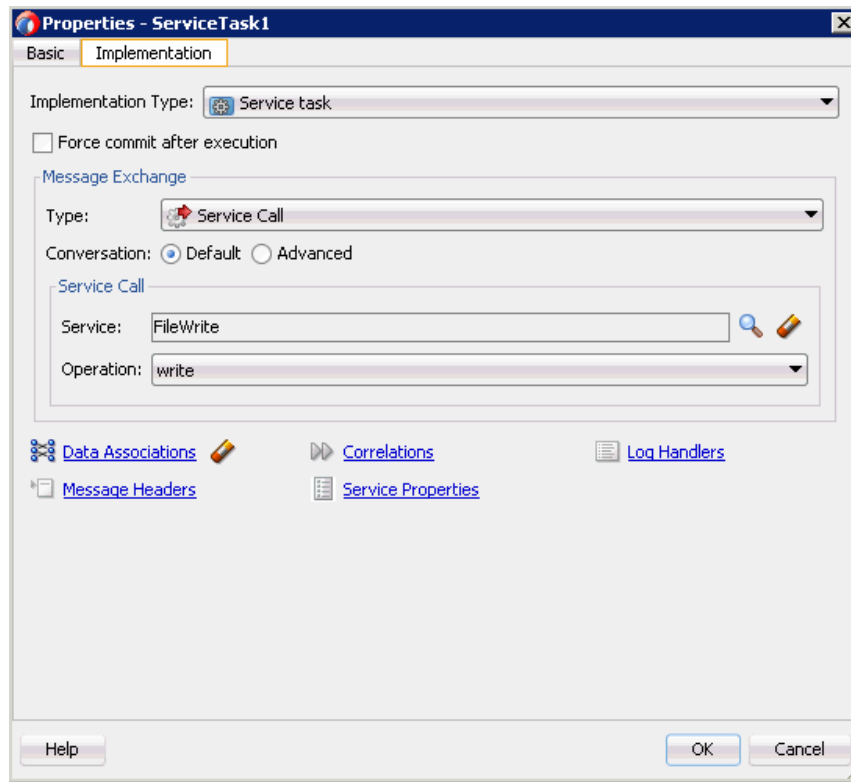
23. 「サービス」フィールドの右の「参照」アイコンをクリックします。
図 6-31 に示すように、「サービス」ダイアログが表示されます。

図 6-31 「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 6-32 に示すように、「プロパティ - ServiceTask1」ダイアログに戻ります。

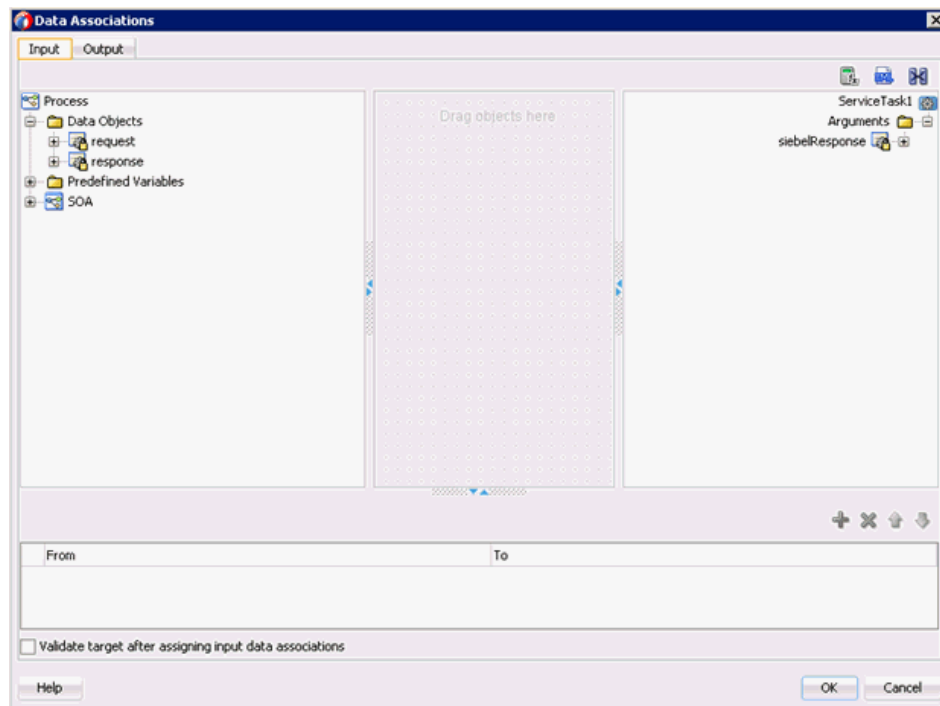
図 6-32 「プロパティ - ServiceTask1」 ダイアログ



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。

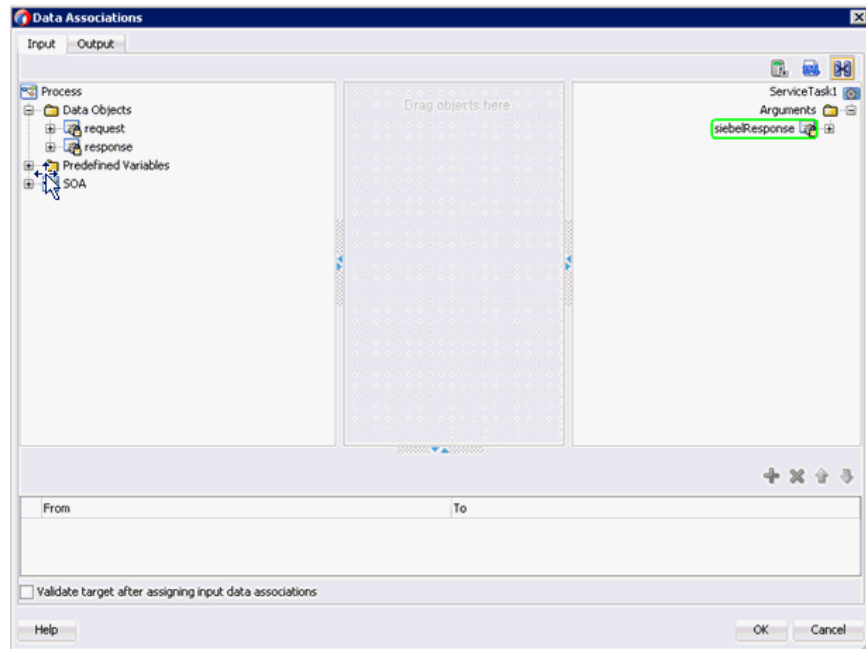
図 6-33 に示すように、「データ・アソシエーション」ダイアログが表示されます。

図 6-33 「データ・アソシエーション」ダイアログ



26. 「入力」アイコンをクリックします。
27. 図 6-34 に示すように、「XSL 変換」アイコンを「SiebelResponse」ノードにドラッグ・アンド・ドロップします。

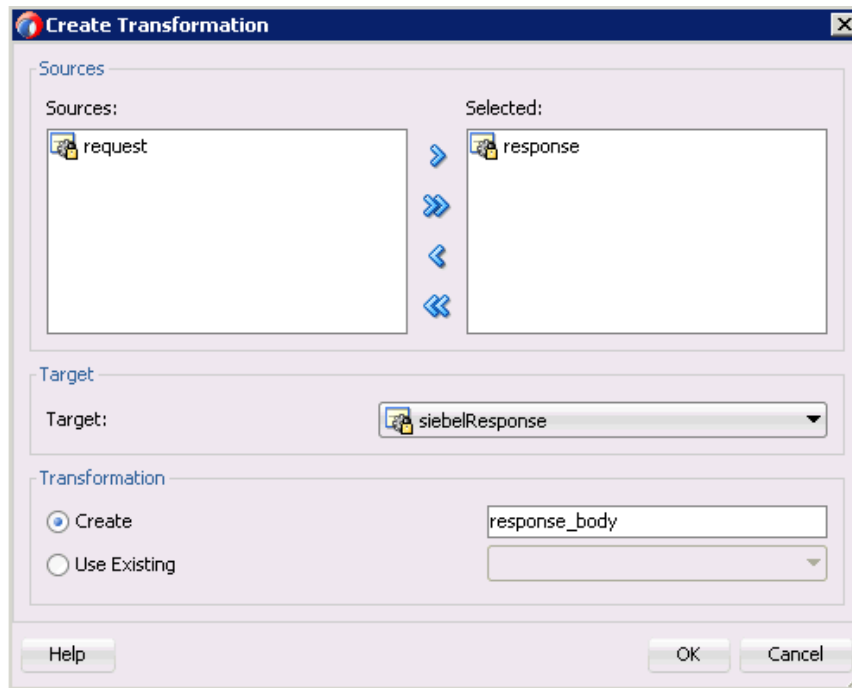
図 6-34 「CompanyCodeSiebelResponse」ノード



「変換の作成」ダイアログが表示されます。

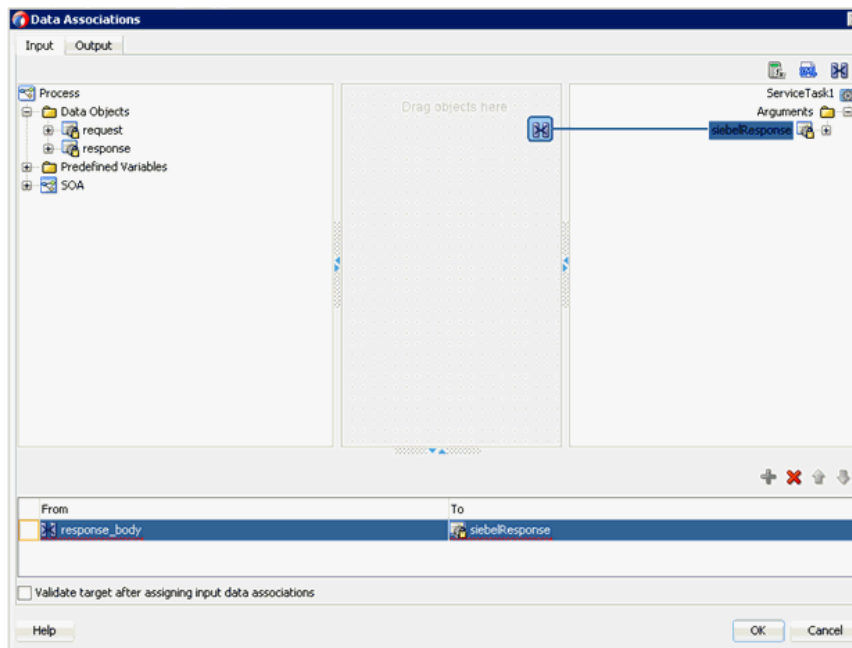
28. 「ソース」セクションの「Response」を選択し、右矢印記号をクリックします。
図 6-35 に示すように、Response オブジェクトが「選択済」要素領域に追加されます。

図 6-35 Response オブジェクト



29. 「OK」をクリックして、「ターゲット」ドロップダウン・リストで選択されているデフォルト値および「作成」フィールドのデフォルトの名前を受け入れます。
- 図 6-36 に示すように、XSL 変換が作成され、「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。

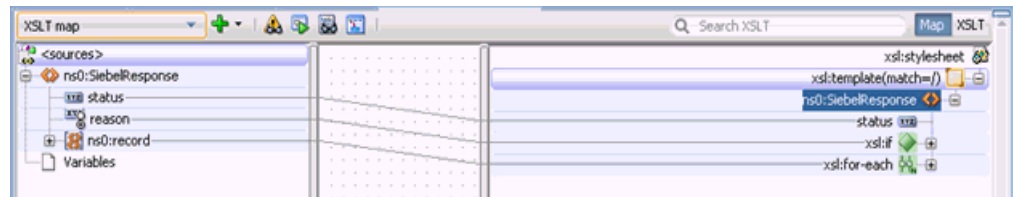
図 6-36 「データ・アソシエーション」ダイアログ



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 6-37 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 6-37 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・バーの「すべて保存」アイコンをクリックし、構成された新規アウトバウンド BPM プロセス・コンポーネントを保存します。
これで、アウトバウンド BPM プロセスをデプロイする準備ができました。

6.4.3 12c のデプロイメントに関する既知の問題の調整

12c のデプロイメントに関する既知の問題の調整方法の詳細は、4-26 ページ [4.4.3.3 項「12c のデプロイメントに関する既知の問題の調整」](#) を参照してください。

6.4.4 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. 左ペインで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・サーバーへのデプロイ」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。

- プロジェクトの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。

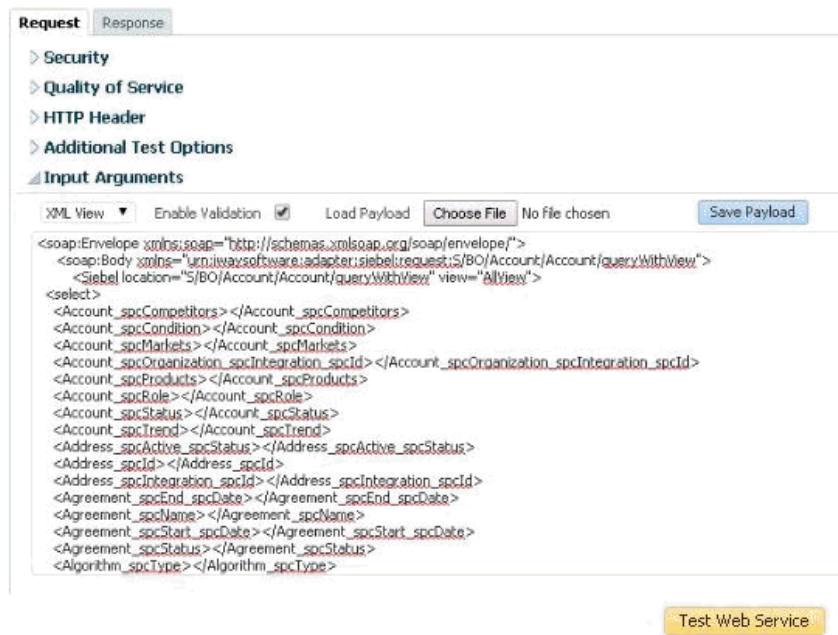
詳細は、4-28 ページ 4.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」を参照してください。

6.4.5 Oracle Enterprise Manager Console での入力 XML ドキュメントの起動

Oracle Enterprise Manager Console 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

- Oracle Enterprise Manager Console にログオンします。
- 左ペインでドメイン、「SOA」フォルダの順に展開します。
- 使用可能なプロジェクト (J2CA_Outbound など) を選択します。
- 「テスト」をクリックします。
- 「リクエスト」タブをクリックします。

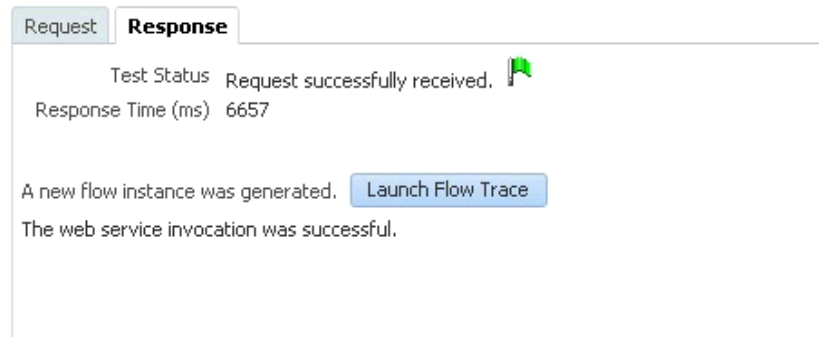
図 6-38 「リクエスト」タブ



- 図 6-38 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 6-39 に示すように、Oracle Enterprise Manager Console 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 6-39 受信されたレスポンス



7. ファイル・システムの定義済出力ディレクトリにナビゲートし、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。

6.5 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 Siebel を Siebel 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。

このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストレーションの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_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。

この項では、次のトピックについて説明します。

- [6.5.1 項「BPM 用の空のコンポジットの作成」](#)
- [6.5.2 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対応する WSDL ファイルを生成する必要があります。詳細は、4-34 ページ [4.5.1 項「イベント統合のための WSDL の生成」](#) を参照してください。

6.5.1 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、4-2 ページ [4.3 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。

6.5.2 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。

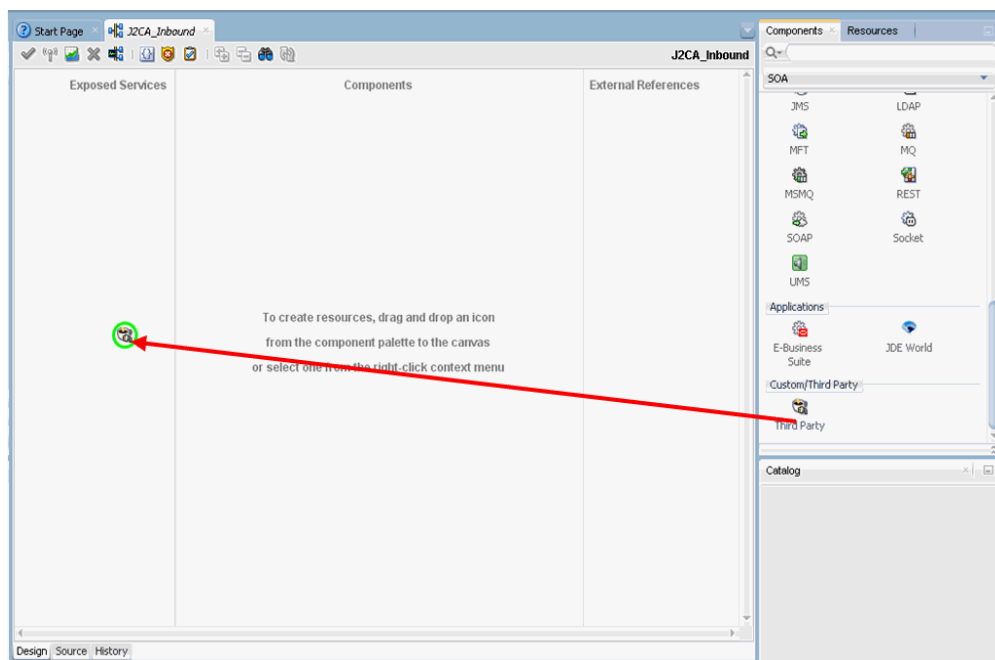
- 6.5.2.1 項「サード・パーティ・アダプタ・サービス・コンポーネントの構成」
- 6.5.2.2 項「インバウンド BPM プロセス・コンポーネントの構成」
- 6.5.2.3 項「書込み操作のファイル・アダプタの作成」
- 6.5.2.4 項「12c のデプロイメントに関する既知の問題の調整」

6.5.2.1 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読み込みます。
2. 図 6-40 に示すように、カスタム/サード・パーティ・ペインから「サード・パーティ・アダプタ」コンポーネントを公開された参照 ペインにドラッグ・アンド・ドロップします。

図 6-40 「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 を検索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。
6. 「ファイルシステム」を選択し、次のディレクトリまでブラウズしてインバウンド WSDL ファイルを選択します。

```
<ADAPTER_HOME>\wsdl1s
```

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。
「XQuery トランスフォーメーション・チューザ」ダイアログが表示されます。
10. 「ファイルシステム」を選択し、次のディレクトリまでブラウズして JCA プロパティ・ファイルを選択します。
`<ADAPTER_HOME>\wsdl`
11. 「OK」をクリックします。
「ファイルのコピー」メッセージ表示されます。
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。
「公開された参照」ペインにサード・パーティ・アダプタ・サービス・コンポーネント (matmas) が作成されます。
これで、インバウンド BPM プロセス・コンポーネントを構成する準備ができました。
詳細は、4-42 ページ [4.5.3.1 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。

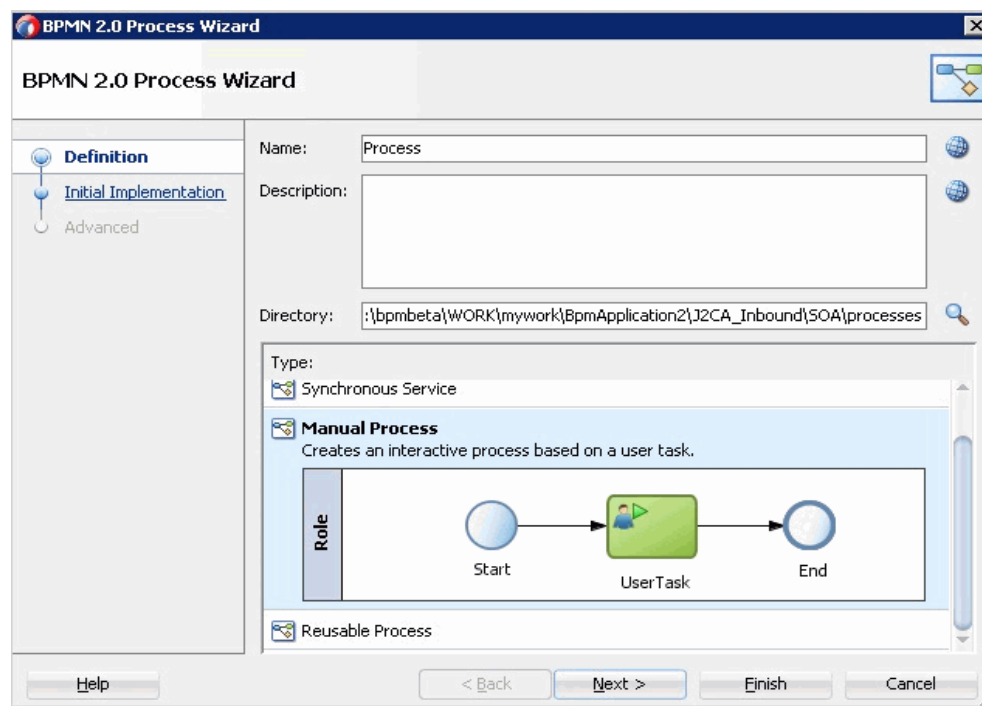
6.5.2.2 インバウンド BPM プロセス・コンポーネントの構成

この項では、インバウンド BPM プロセス・コンポーネントの構成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。

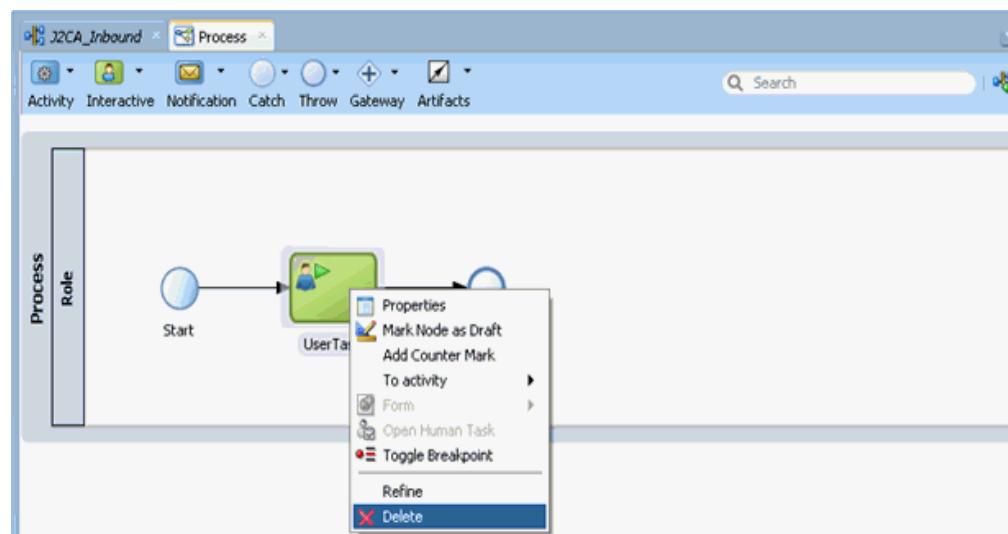
1. 「コンポーネント」ペインから「BPMN プロセス」コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップします。
[図 6-41](#) に示すように、「BPMN プロセスの作成」ダイアログが表示されます。

図 6-41 「BPMN プロセスの作成」ダイアログ



2. 「タイプ」セクションで「手動プロセス」を選択します。
3. 「終了」をクリックします。
 6-42 に示すように、BPMN プロセスが表示されます。

図 6-42 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」の「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 6-43 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 6-43 「参照」アイコン

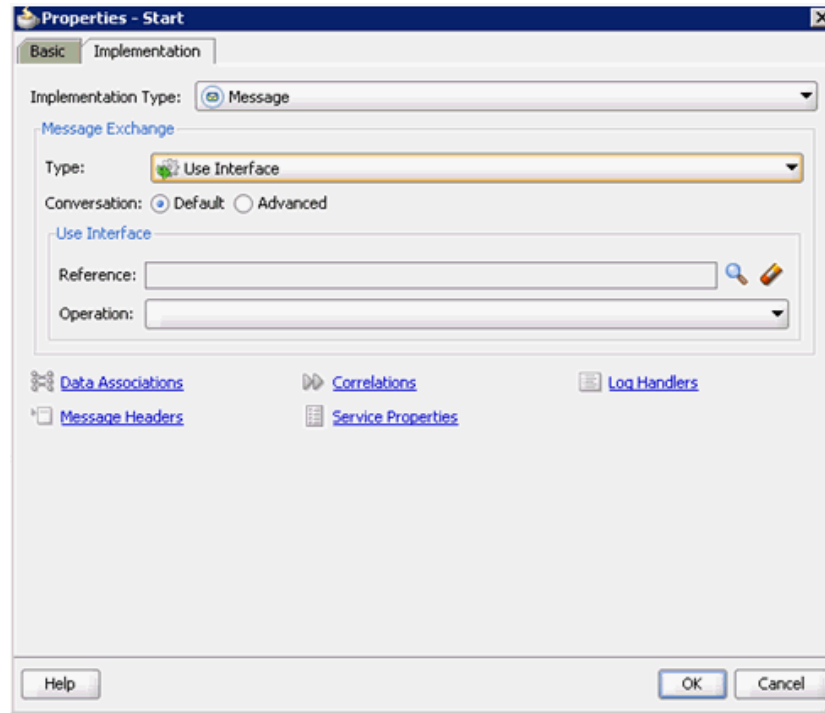
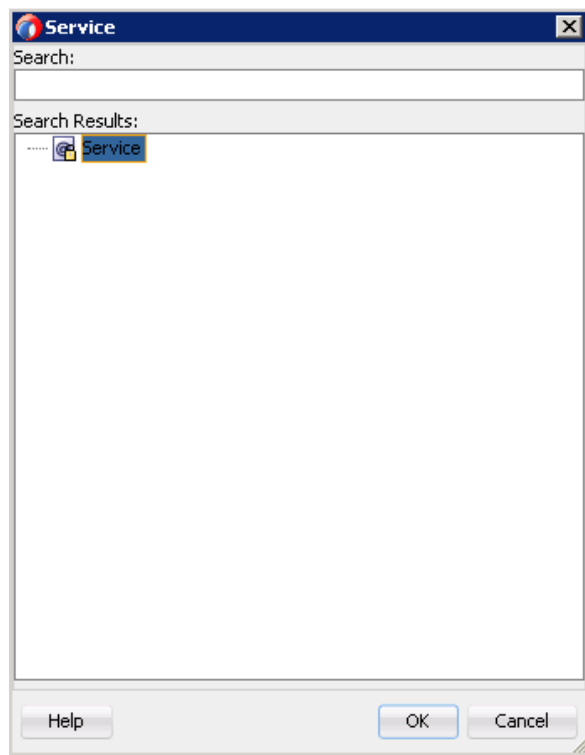


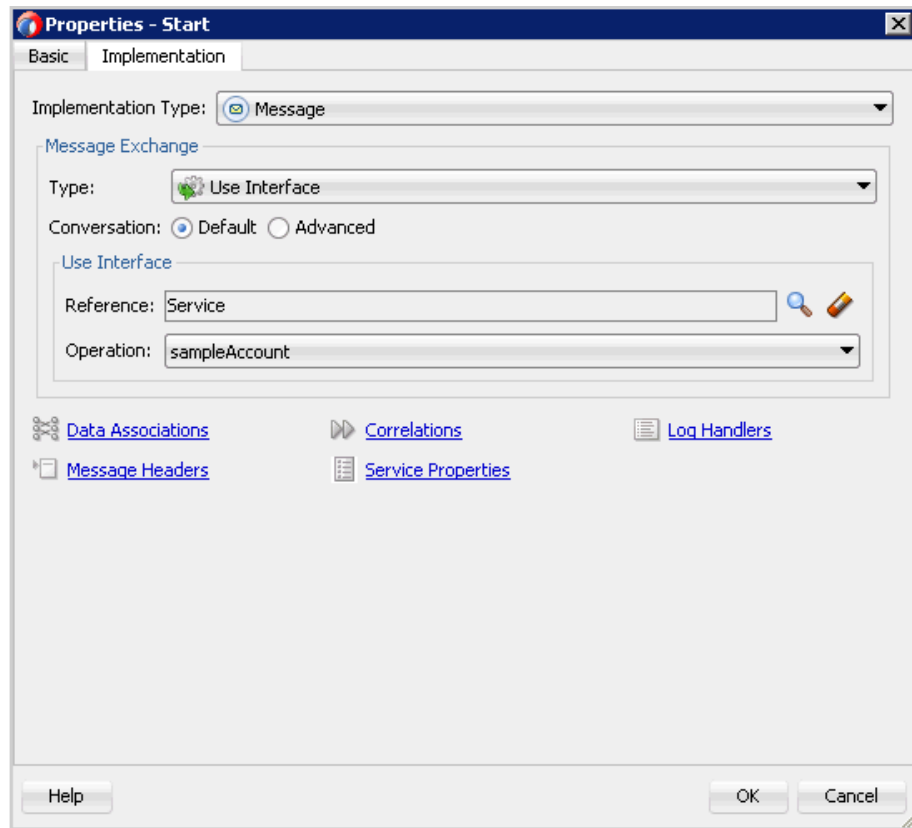
図 6-44 に示すように、「サービス」ダイアログが表示されます。

図 6-44 「サービス」 ダイアログ



10. 作成されたサード・パーティ・サービスを選択して「OK」をクリックします。
図 6-45 に示すように、「プロパティ - 開始」ダイアログに戻ります。

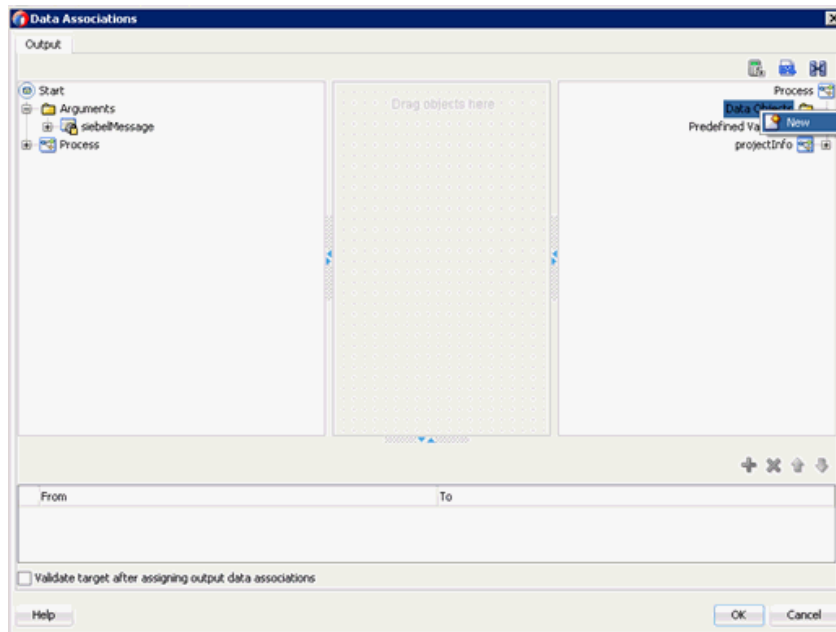
図 6-45 「プロパティ - 開始」 ダイアログ



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 6-46 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 6-46 「データ・アソシエーション」ダイアログ



12. 右ペインの「データ・オブジェクト」ノードを右クリックし、「新規」を選択します。
「データ・オブジェクトの作成」ダイアログが表示されます。
13. 図 6-47 に示すように、「名前」フィールドに名前を入力し、「タイプ」フィールドのドロップダウン・ボタンをクリックし、リストから「参照」を選択します。

図 6-47 「データ・オブジェクトの作成」ダイアログ

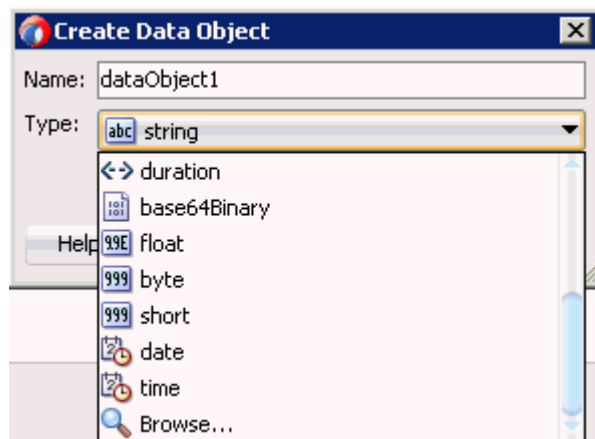
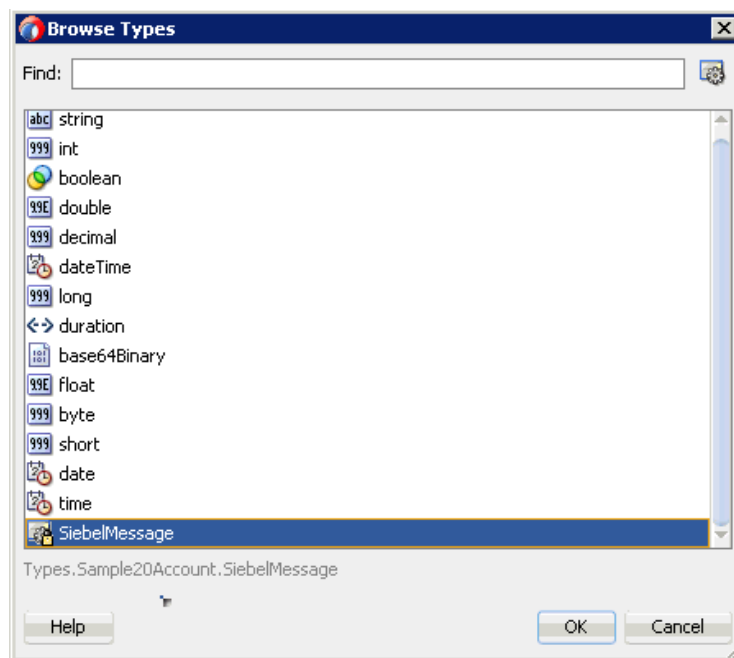


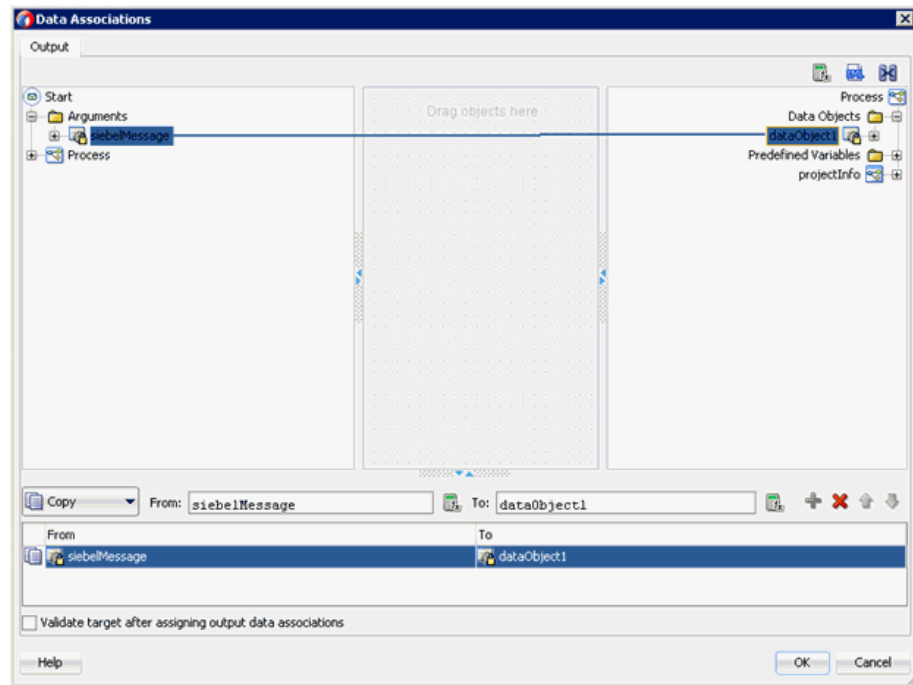
図 6-48 に示すように、「参照タイプ」ダイアログが表示されます。

図 6-48 「参照タイプ」ダイアログ



14. コンポーネントを選択し、「OK」をクリックします。
「データ・オブジェクトの作成」ダイアログに戻ります。
15. 「OK」をクリックします。
図 6-49 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」ノードの下に表示されます。

図 6-49 「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「**siebelMessage**」引数を選択し、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. 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読み込みます。
20. メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規インバウンド BPM プロセス・コンポーネントを保存します。
これで書き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。

6.5.2.3 書き込み操作のファイル・アダプタの作成

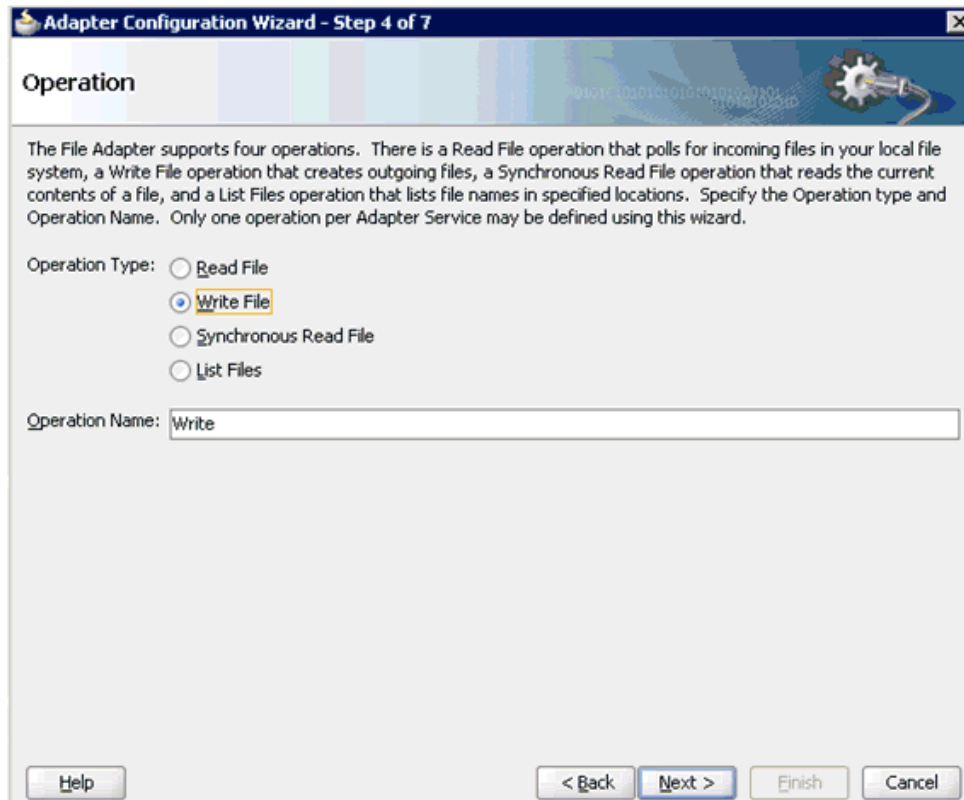
この項では、書き込み操作のファイル・アダプタの作成方法について説明します。

書き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. 「テクノロジー・アダプタ」ペインから「**ファイル・アダプタ**」コンポーネントを「外部参照」ペインにドラッグ・アンド・ドロップします。
アダプタ構成ウィザードが表示されます。
2. 「名前」フィールドに新規のファイル・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。

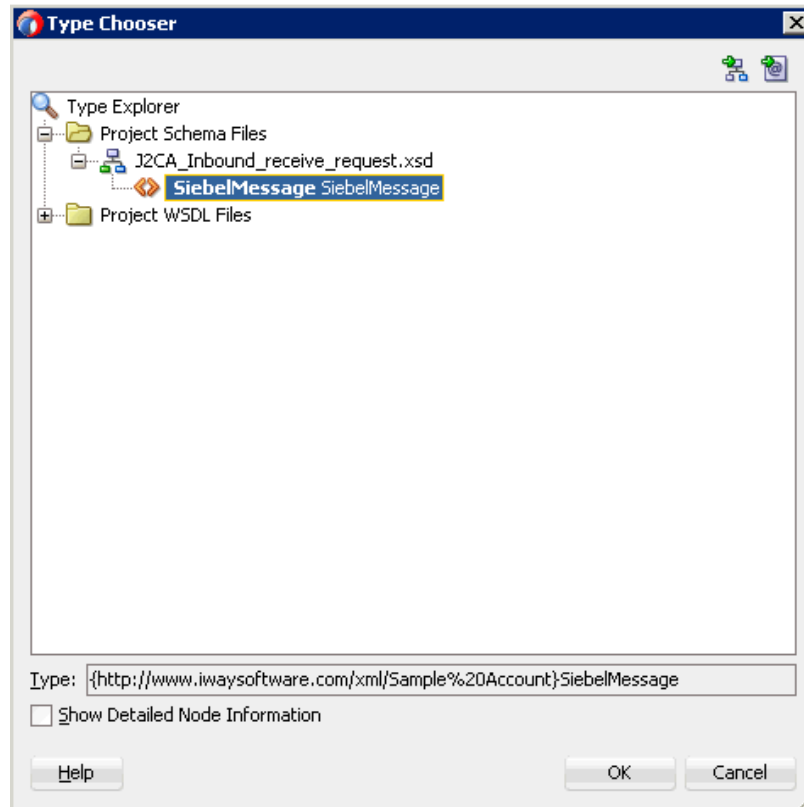
4. 「次へ」をクリックします。
「ファイル・サーバーの接続」ページが表示されます。
5. 「次へ」をクリックします。
図 6-50 に示すように、「操作」ページが表示されます。

図 6-50 「操作」ページ



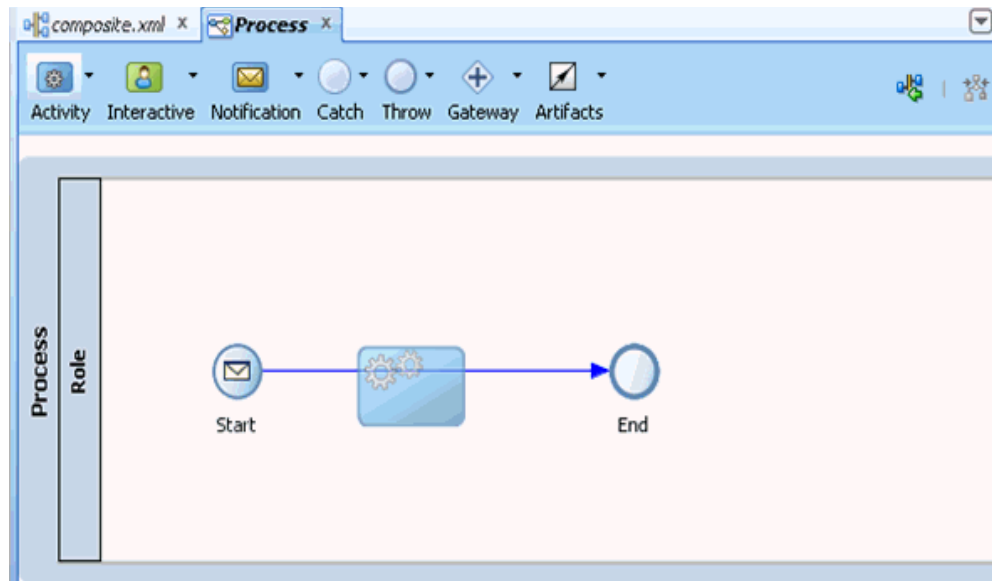
6. 「操作タイプ」オプションのリストから「Write File」を選択し、「操作名」(「Write」など)を指定します。
7. 「次へ」をクリックします。
「ファイル構成」ページが表示されます。
8. ファイル・システムで出力ファイルが書き込まれる場所を指定します。
9. 「ファイル・ネーミング規則」フィールドで、出力ファイルの名前を指定します。
10. 「次へ」をクリックします。
「メッセージ」ページが表示されます。
11. 「URL」フィールドの右の「参照」をクリックします。
図 6-51 に示すように、「タイプ・チューザ」ダイアログが表示されます。

図 6-51 「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。
「外部参照」ペインにファイル・アダプタ・サービスが作成されます。
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 6-52 に示すように、「開始」アイコンをドロップします。

図 6-52 「アクティビティ」アイコン

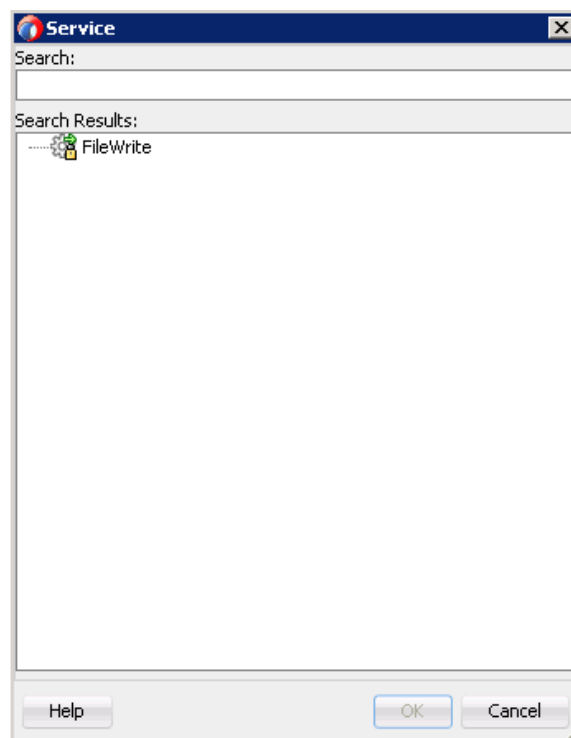


「プロパティ - ServiceTask」ダイアログが表示されます。

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」の「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。

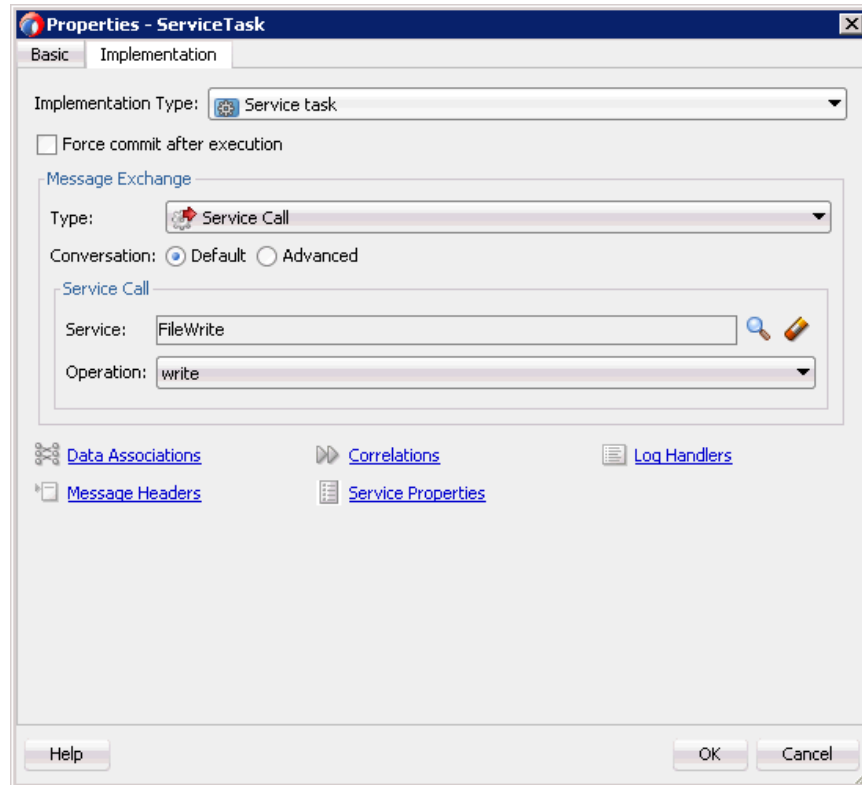
図 6-53 に示すように、「タイプ」ダイアログが表示されます。

図 6-53 「タイプ」ダイアログ



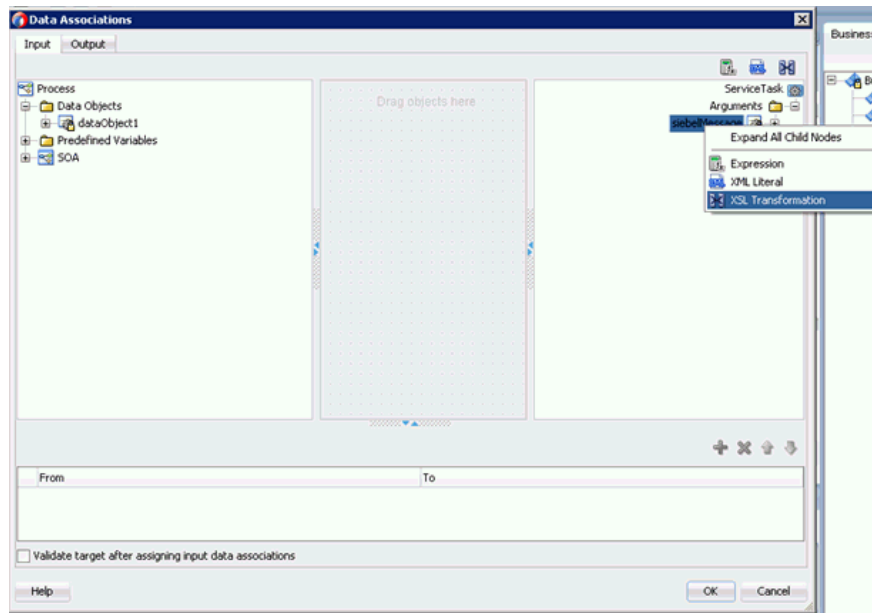
24. 作成された書き込み操作のサービスを選択し、「OK」をクリックします。
図 6-54 に示すように、「プロパティ - ServiceTask」ダイアログに戻ります。

図 6-54 「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 6-55 に示すように、右ペインの「siebelMessage」引数を右クリックして「XSL 変換」を選択します。

図 6-55 XSL 変換



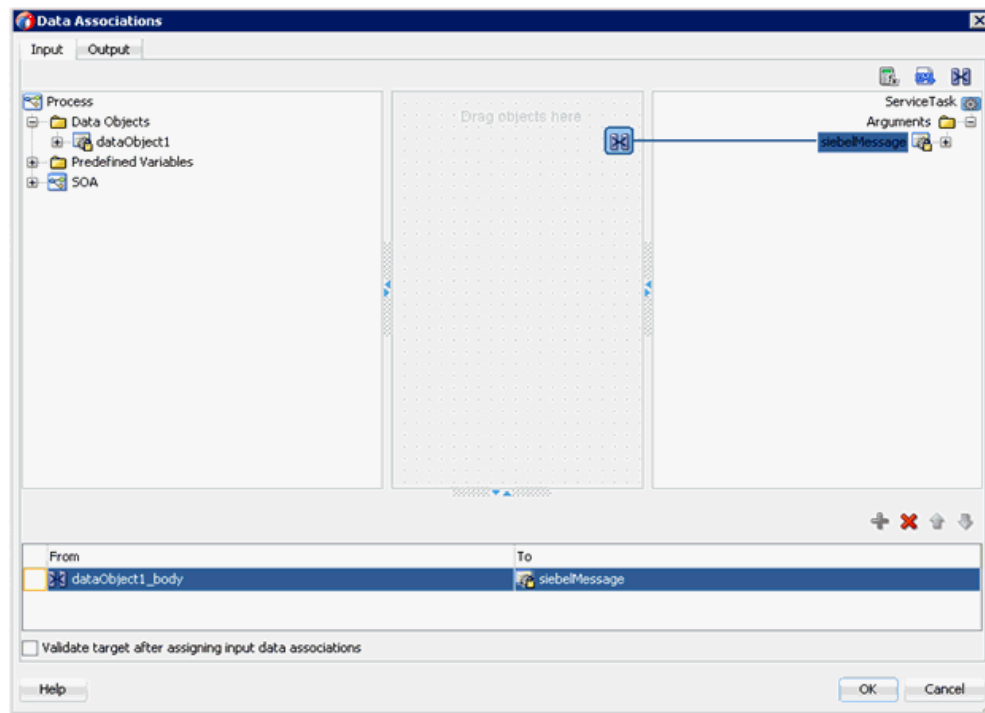
「変換の作成」ダイアログが表示されます。

27. 「ソース」領域に追加します。

28. 「OK」をクリックします。

図 6-56 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 6-56 「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。

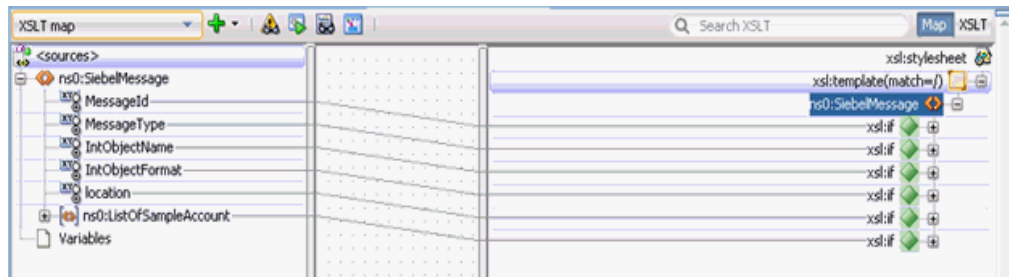
31. ソース要素とターゲット要素を自動マップします。

「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 6-57 に示すように、トランスフォーメーションが完成します。

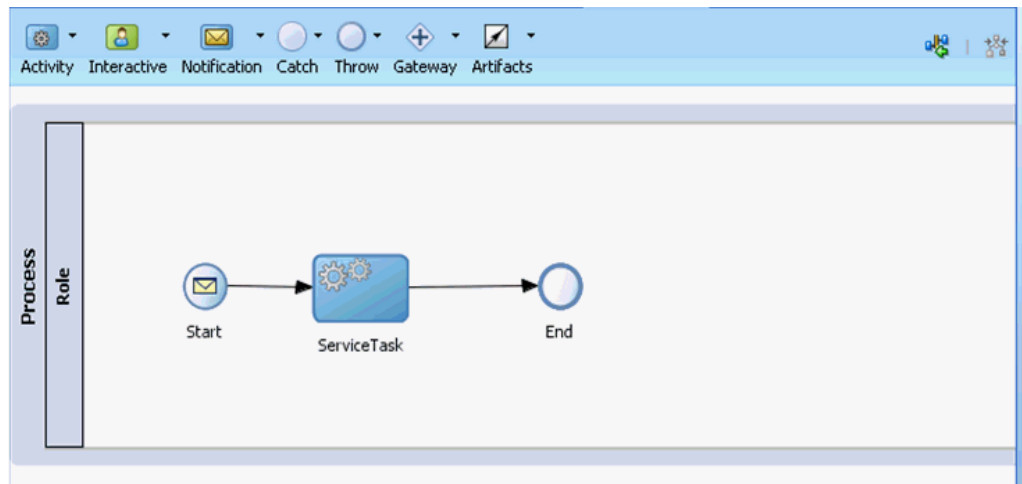
図 6-57 完成したトランスフォーメーション



33. トランスフォーメーションを保存します。

34. 図 6-58 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 6-58 「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・バーの「すべて保存」アイコンをクリックし、構成された新規インバウンド BPM プロセス・コンポーネントを保存します。

6.5.2.4 12c のデプロイメントに関する既知の問題の調整

12c のデプロイメントに関する既知の問題の調整方法の詳細は、4-26 ページ [4.4.3.3 項「12c のデプロイメントに関する既知の問題の調整」](#) を参照してください。

これで、インバウンド BPM プロセスをデプロイする準備ができました。4-48 ページ [4.5.4 項「BPEL インバウンド・プロセスのデプロイ」](#) の記載と同じ手順に従います。Siebel でイベントをトリガーする方法の詳細は、4-64 ページ [4.5.5.3 項「Siebel 8.0 でイベントをトリガーしてイベントのランタイム統合をテストする」](#) を参照してください。

6.6 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、Siebel 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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_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)

この項では、次のトピックについて説明します。

- [6.6.1 項「BPM 用の空のコンポジットの作成」](#)
- [6.6.2 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対応する WSDL ファイルを生成する必要があります。詳細は、4-76 ページ [4.6.1 項「Web サービスを使用するリクエスト/レスポンス型サービス用の WSDL ファイルの生成」](#) を参照してください。

6.6.1 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、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. BPM アプリケーションの名前を入力し、「次へ」をクリックします。
3. 「プロジェクト名」フィールドに名前を入力し、「次へ」をクリックします。
4. 「コンポジット・テンプレート」リストで「空のコンポジット」を選択し、「終了」をクリックします。

詳細は、4-41 ページ [4.5.2 項「SOA 用の空のコンポジットの作成」](#) を参照してください。

6.6.2 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・プロセスの定義方法について説明します。内容は次のとおりです。

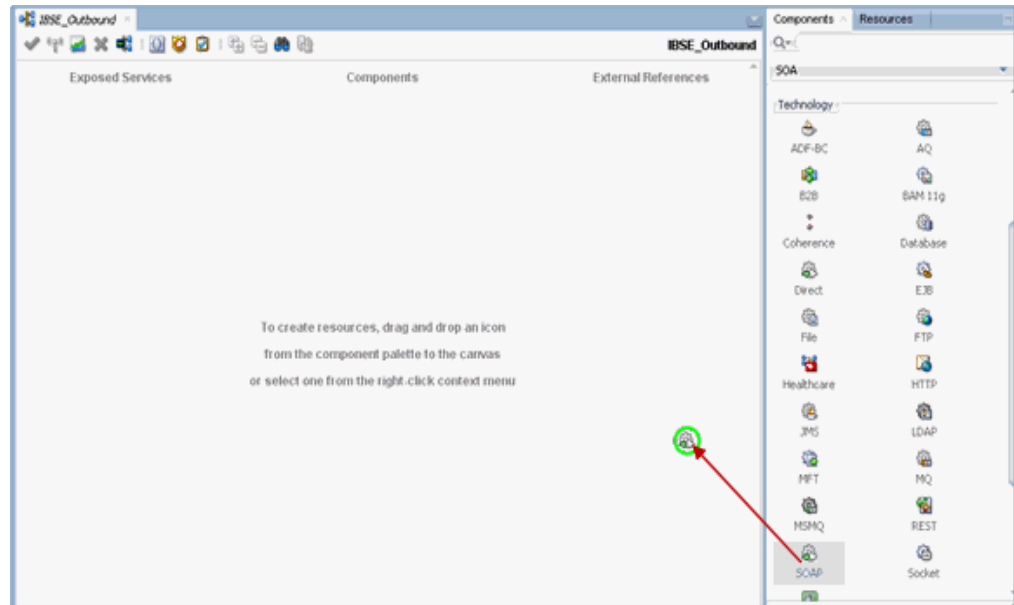
- [6.6.2.1 項「Web サービス・コンポーネントの構成」](#)
- [6.6.2.2 項「BPM プロセス・コンポーネントの構成」](#)
- [6.6.2.3 項「書込み操作のファイル・アダプタの作成」](#)

6.6.2.1 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読み込みます。
2. 図 6-59 に示すように、「テクノロジー・アダプタ」ペインから「Web サービス」ノードを「外部参照」ペインにドラッグ・アンド・ドロップします。

図 6-59 「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 を検索**アイコンをクリックします。
4. 表示される「WSDL チューザ」ウィンドウで、アプリケーション・エクスプローラから WSDL がエクスポートされる場所へ移動し、その WSDL を選択します。
5. 「OK」をクリックします。
6. 図 6-60 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 6-60 「Web サービス」 ペイン

Create Web Service

SOAP
Create a web service for services external to the SOA composite.

Name: SOAPReference

Type: Reference

WSDL URL: .2c_SOA\soa\soa\thirdparty\ApplicationAdapters\wsdls\IBSE_Outbound.wsdl

Port Type: IBSE_OutboundSoap

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

copy wsdl and its dependent artifacts into the project.

Transaction Participation: WSDLDriven

Version: DEFAULT

Help OK Cancel

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。

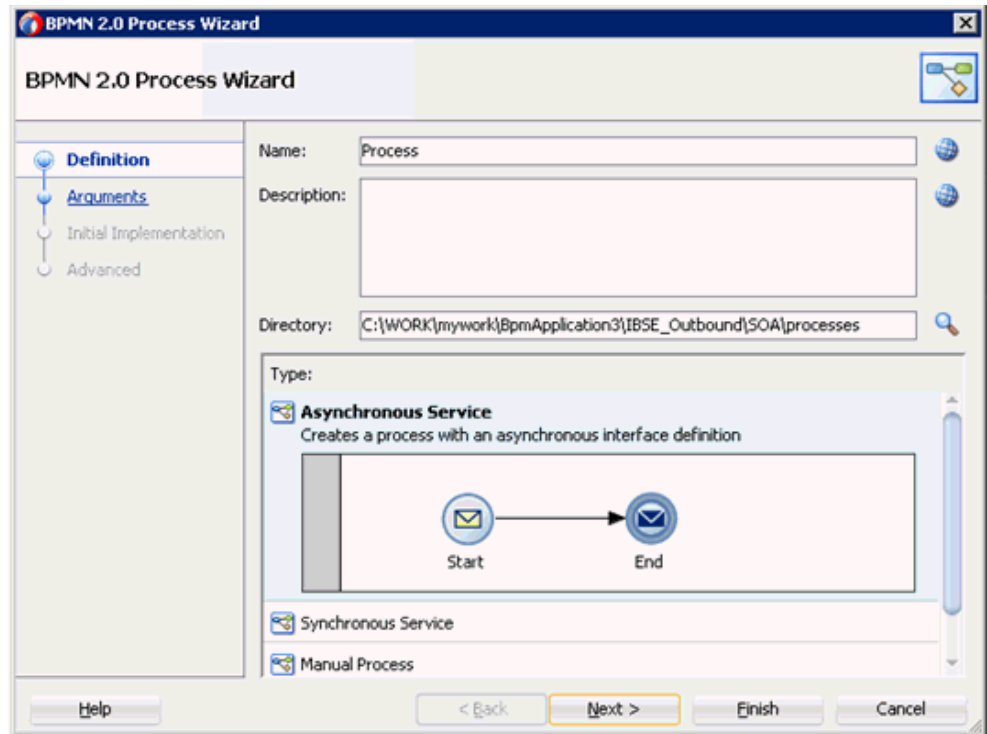
6.6.2.2 BPM プロセス・コンポーネントの構成

この項では、アウトバウンド BPM プロセス・コンポーネントの構成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。

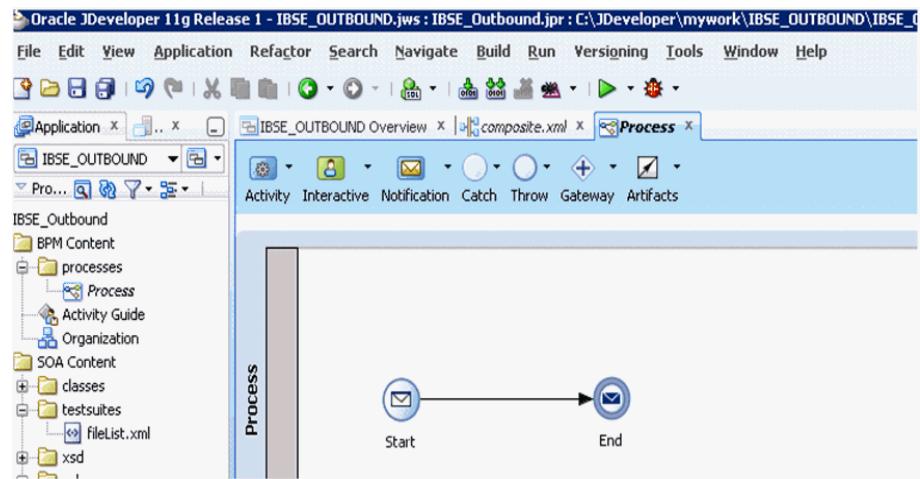
1. 「コンポーネント」ペインから「BPMN プロセス」コンポーネントを「コンポーネント」ペインにドラッグ・アンド・ドロップします。
2. 図 6-61 に示すように、「タイプ」領域（「非同期サービス」）で選択されたデフォルトのオプションを受け入れ、「終了」をクリックします。

図 6-61 「タイプ」領域



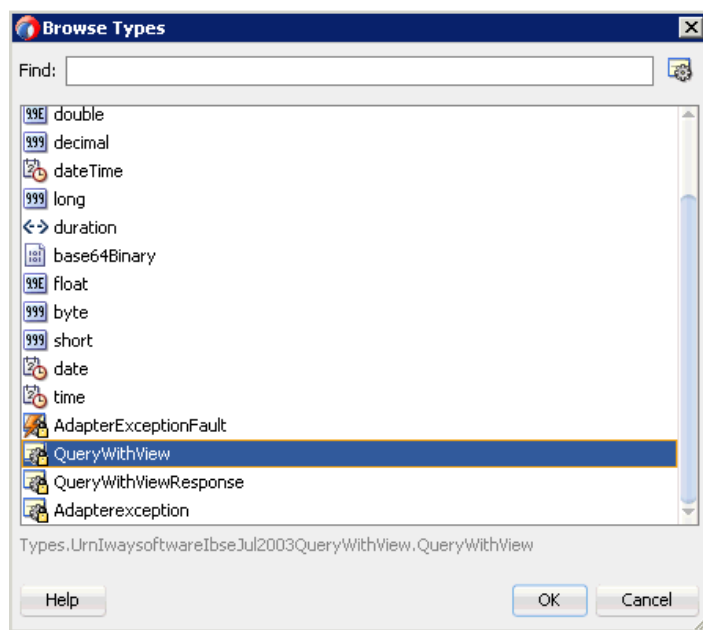
3. 図 6-62 に示すように、「開始イベント」コンポーネントをダブルクリックします。

図 6-62 「開始イベント」コンポーネント



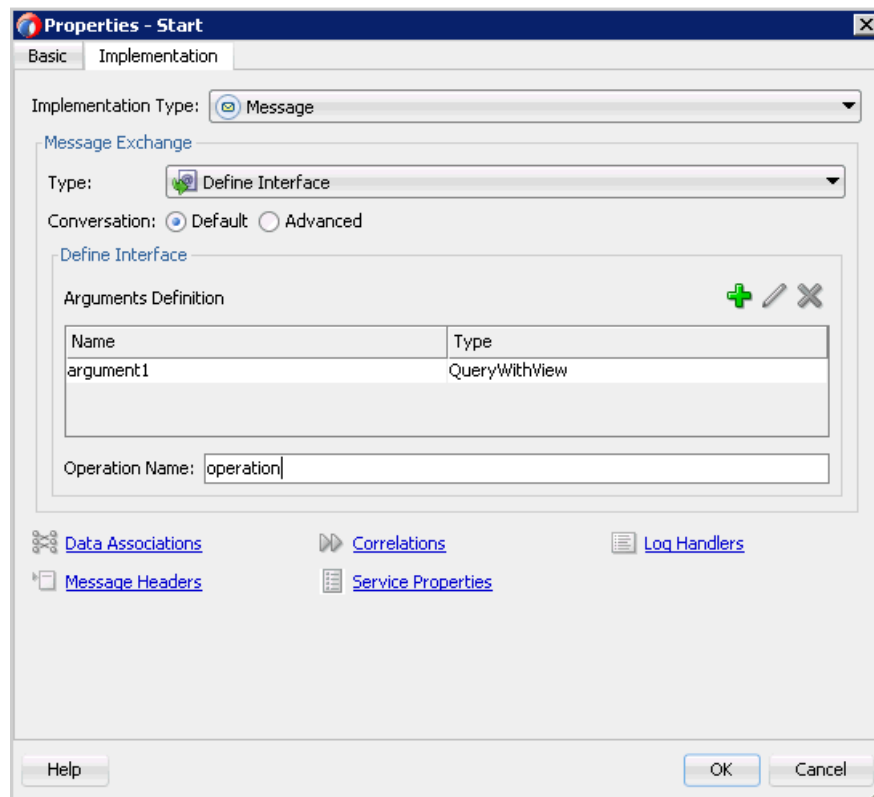
4. 表示される「プロパティ - 開始」ウィンドウで「実装」タブをクリックします。
5. 「引数の定義」フィールドの右の「プラス」(+) アイコンをクリックします。
「引数の編集」ウィンドウが表示されます。
6. 「名前」フィールドに名前を入力し、「タイプ」ドロップダウン・リストをクリックして「参照」を選択します。
7. 図 6-63 に示すように、リクエスト・コンポーネント (「queryWithView」など) を選択し、「OK」をクリックします。

図 6-63 リクエスト・コンポーネント



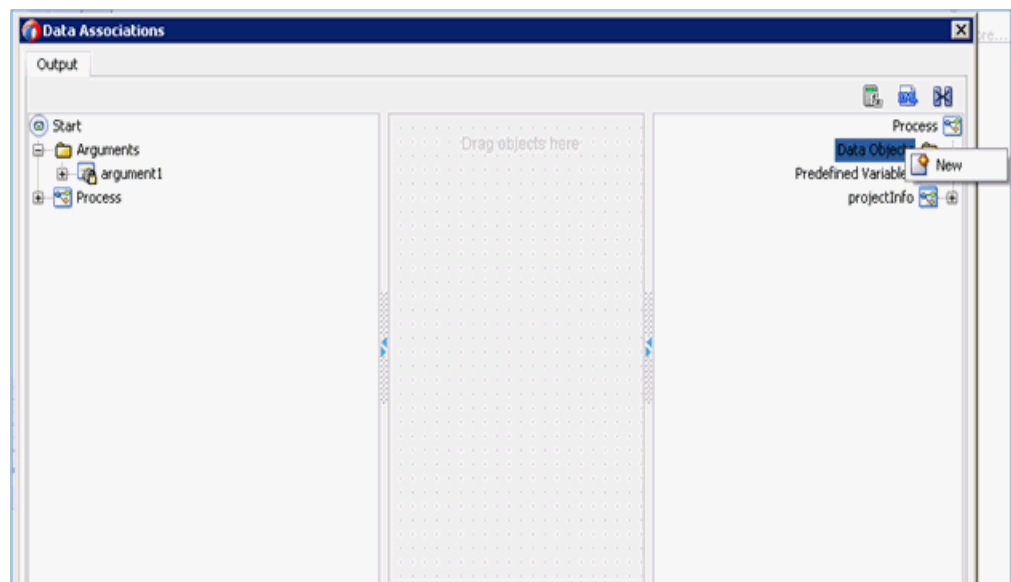
8. 表示される「引数の編集」ウィンドウで「OK」をクリックします。
「プロパティ - 開始」ウィンドウが表示されます。
9. 「操作名」フィールドでデフォルトのエントリを **start** から **operation** に変更します。
10. 図 6-64 に示すように、「データ・アソシエーション」ハイパーリンクをクリックします。

図 6-64 「プロパティ - 開始」 ウィンドウ



11. 図 6-65 に示すように、右ペインの「プロセス」の下の「データ・オブジェクト」ノードを右クリックし、「新規」を選択します。

図 6-65 「データ・オブジェクト」ノード

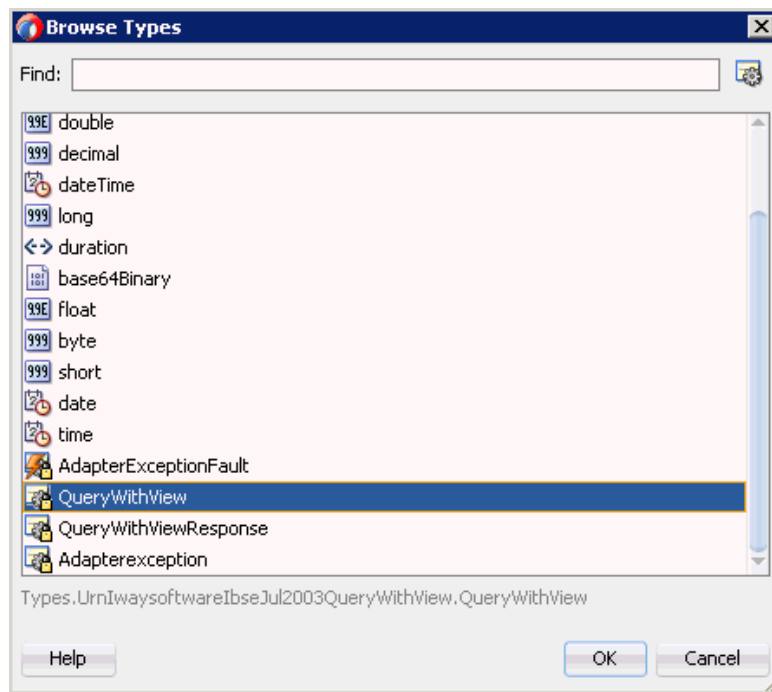


「データ・オブジェクトの作成」ウィンドウが表示されます。

12. 「名前」フィールドに名前を入力し、「タイプ」ドロップダウン・リストをクリックして「参照」を選択します。

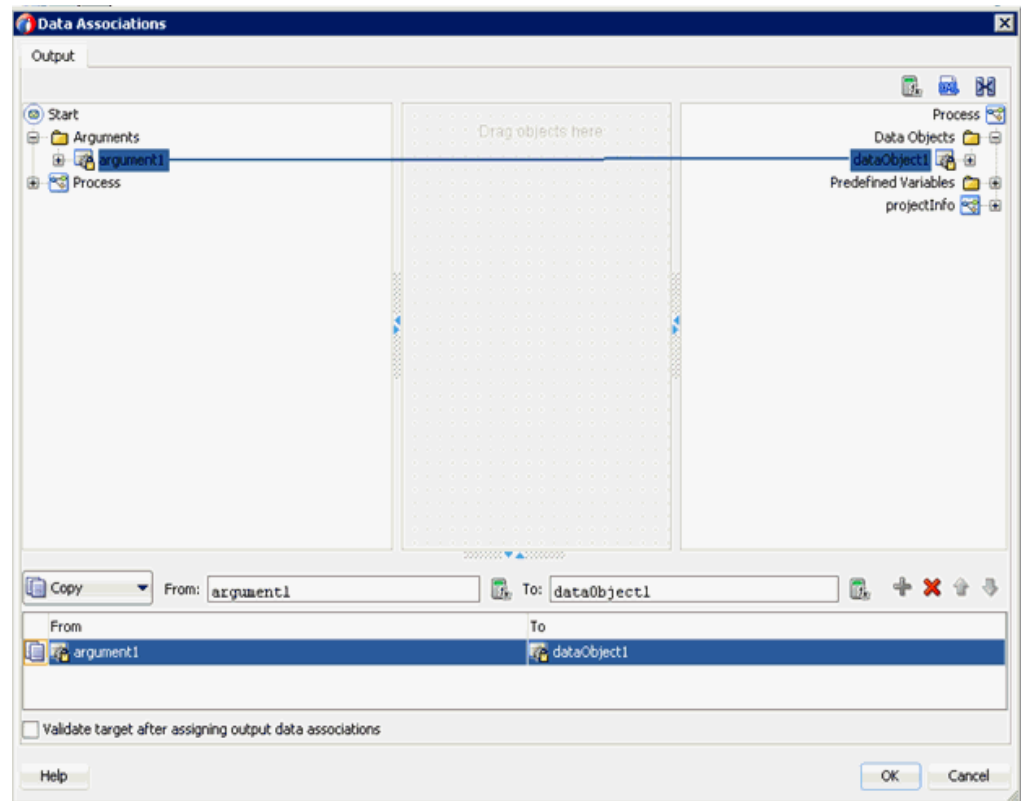
13. 図 6-66 に示すように、リクエスト・コンポーネント (「queryWithView」など) を選択し、「OK」をクリックします。

図 6-66 リクエスト・コンポーネント



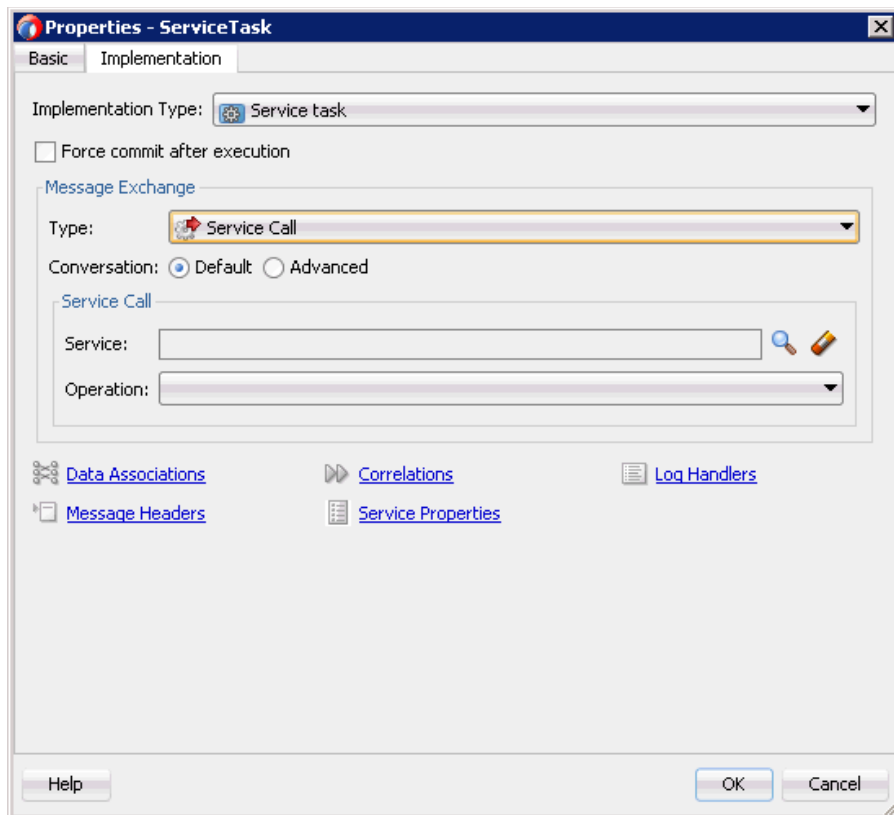
14. 「データ・オブジェクトの作成」 ウィンドウで、「OK」をクリックします。
「データ・アソシエーション」 ウィンドウが表示されます。
15. 左ペインの「引数」 ノードの下の「argument1」を選択し、右ペインの「データ・オブジェクト」の下の「dataObject1」にドラッグして接続します。
16. 図 6-67 に示すように、「OK」をクリックします。

図 6-67 データ・アソシエーション



17. 表示される「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」イベント・コンポーネントと「終了」イベント・コンポーネントの間の伝送路に「サービス」アイコンをドロップします。
20. 表示される「プロパティ - ServiceTask」ウィンドウで「実装」タブをクリックします。
21. 「メッセージ交換」の「タイプ」リストから「サービス・コール」を選択します。
22. 図 6-68 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。

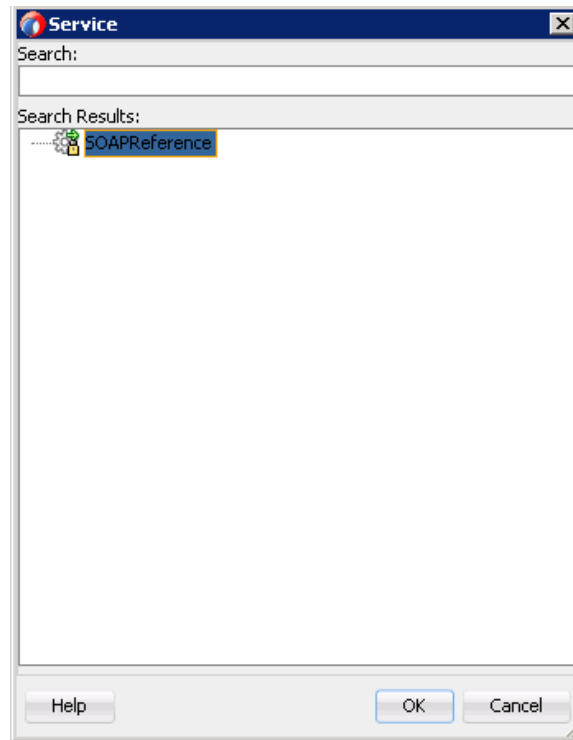
図 6-68 「参照」アイコン



「サービス」ウィンドウが表示されます。

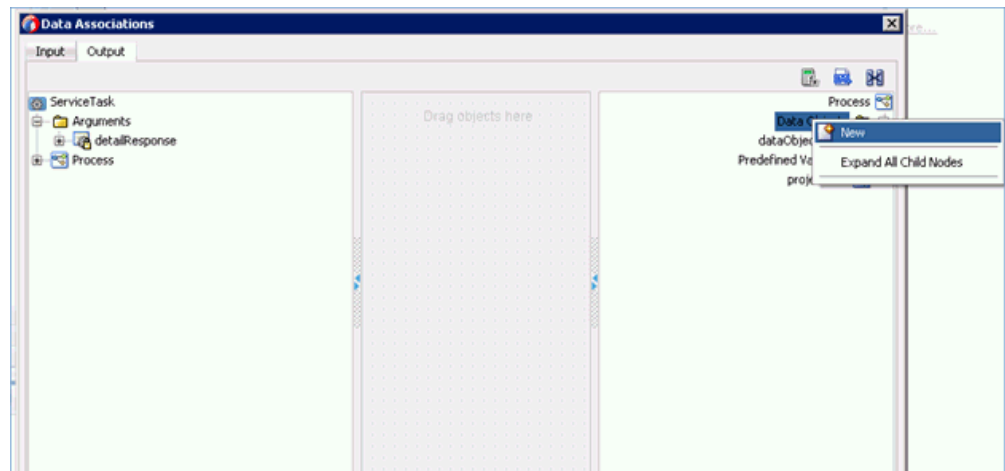
23. 図 6-69 に示すように、作成された Web サービスを選択して「OK」をクリックします。

図 6-69 作成された Web サービス



24. 表示される「プロパティ - ServiceTask」ウィンドウで「データ・アソシエーション」ハイパーリンクをクリックします。
「データ・アソシエーション」ウィンドウが表示されます。
25. 図 6-70 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。

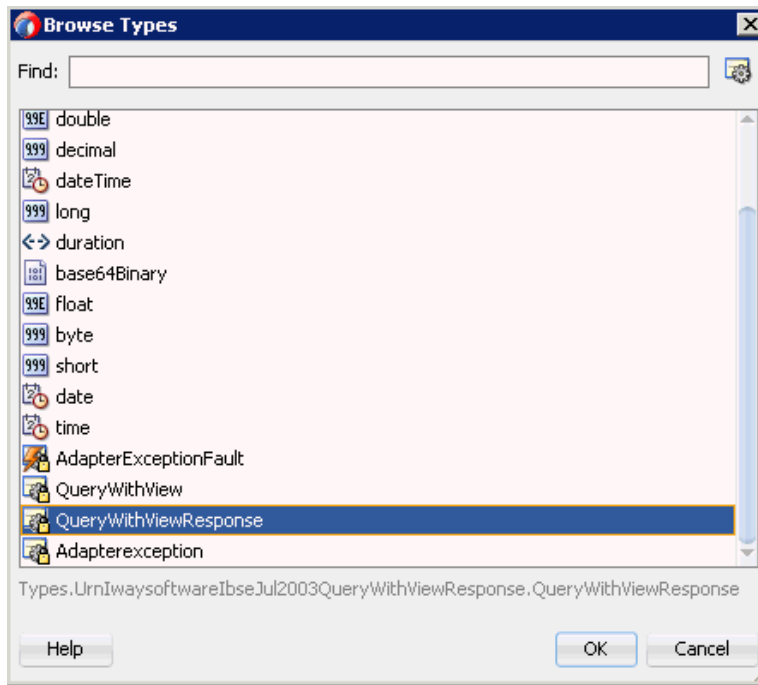
図 6-70 「データ・オブジェクト」ノード



「データ・オブジェクトの作成」ウィンドウが表示されます。

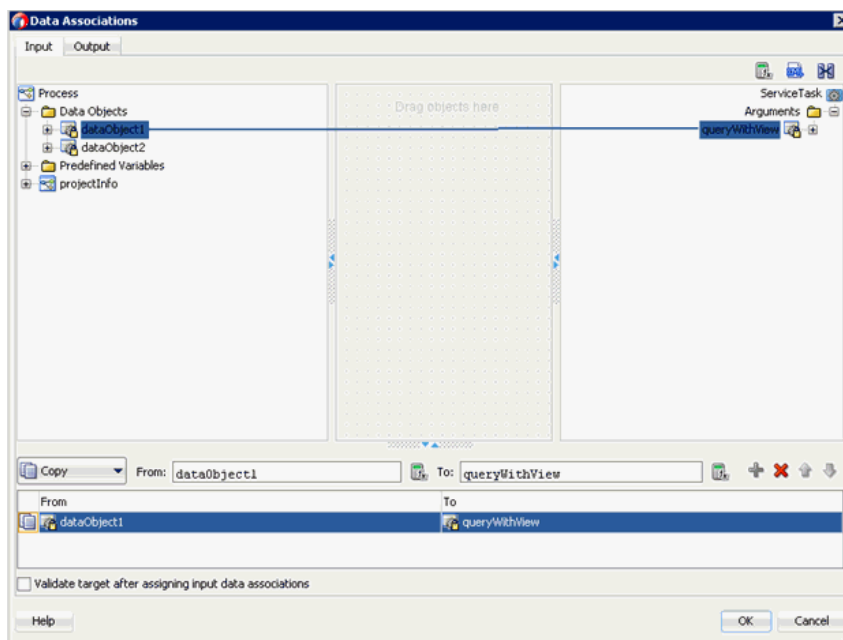
26. 「名前」フィールドに名前を入力し、「タイプ」ドロップダウン・リストをクリックして「参照」を選択します。
27. 図 6-71 に示すように、レスポンス・コンポーネント（「QueryWithViewResponse」など）を選択し、「OK」をクリックします。

図 6-71 レスポンス・コンポーネント (QueryWithViewResponse)



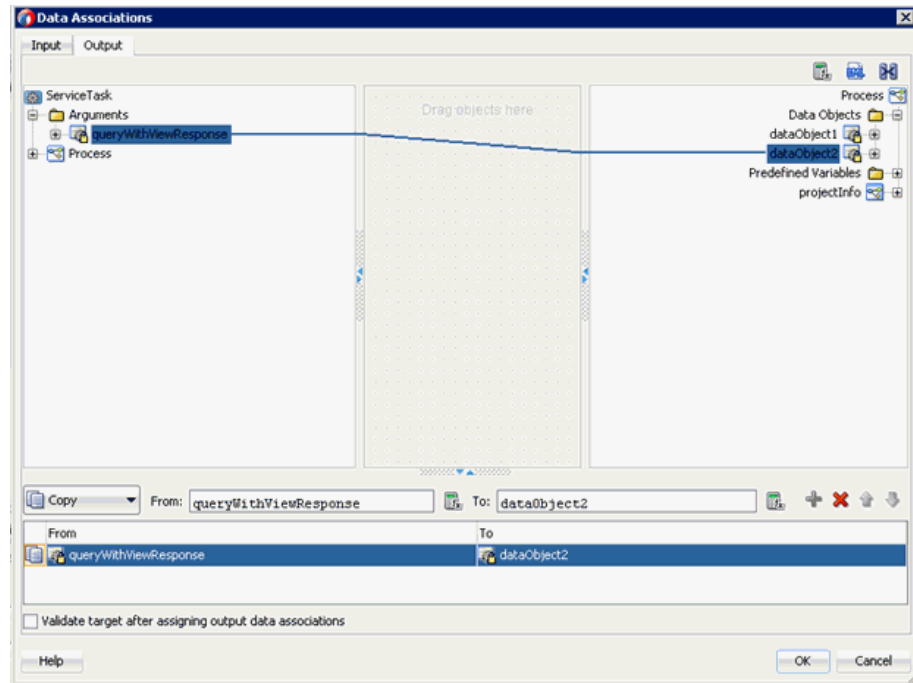
28. 「データ・オブジェクトの作成」 ウィンドウで、「OK」をクリックします。
「データ・アソシエーション」 ウィンドウが表示されます。
29. 図 6-72 に示すように、「入力」タブの左ペインにある「データ・オブジェクト」ノードの下の「dataObject1」を選択し、右ペインの「引数」ノードの下の「queryWithView」ノードにドラッグして接続します。

図 6-72 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「queryWithViewResponse」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 6-73 に示すように、「OK」をクリックします。

図 6-73 「出力」タブ



32. 表示される「プロパティ - ServiceTask」ウィンドウで「OK」をクリックします。
33. メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規アウトバウンド BPM プロセス・コンポーネントを保存します。
34. 左のペインで「composite.xml」ノードをダブルクリックします。

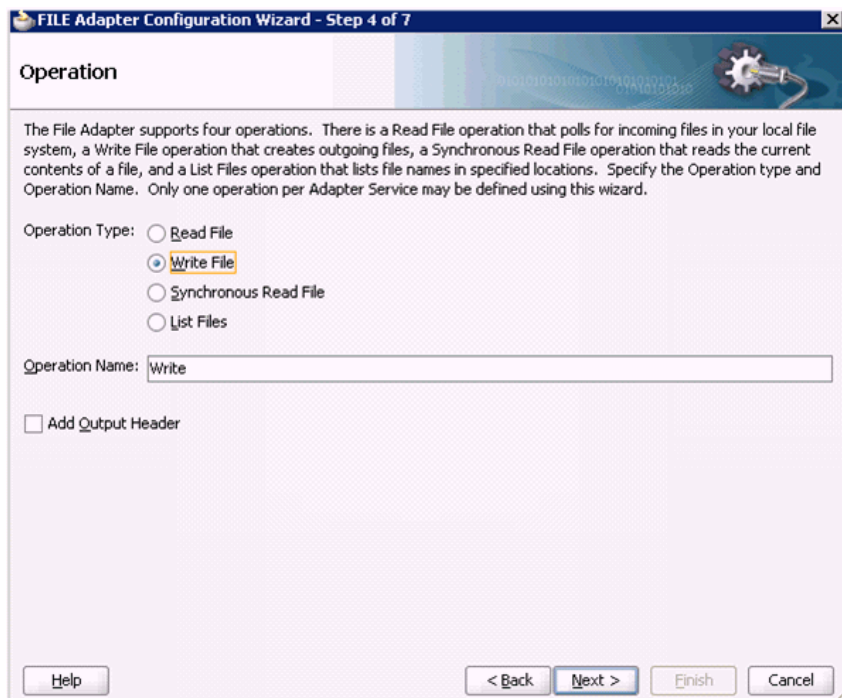
6.6.2.3 書込み操作のファイル・アダプタの作成

この項では、書込み操作のファイル・アダプタの作成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. 「テクノロジー・アダプタ」ペインから「ファイル・アダプタ」コンポーネントを「外部参照」ペインにドラッグ・アンド・ドロップし、「ファイル・アダプタ」に名前を入力します。
2. 表示される「アダプタ・インタフェース」ペインで、「操作およびスキーマから定義 (後で指定)」オプションが選択されていることを確認し、「次へ」をクリックします。
3. 「次へ」をクリックします。
4. 図 6-74 に示すように、表示される「操作」ペインで、「操作タイプ」オプションのリストから「ファイルの書込み」を選択し、「次へ」をクリックします。

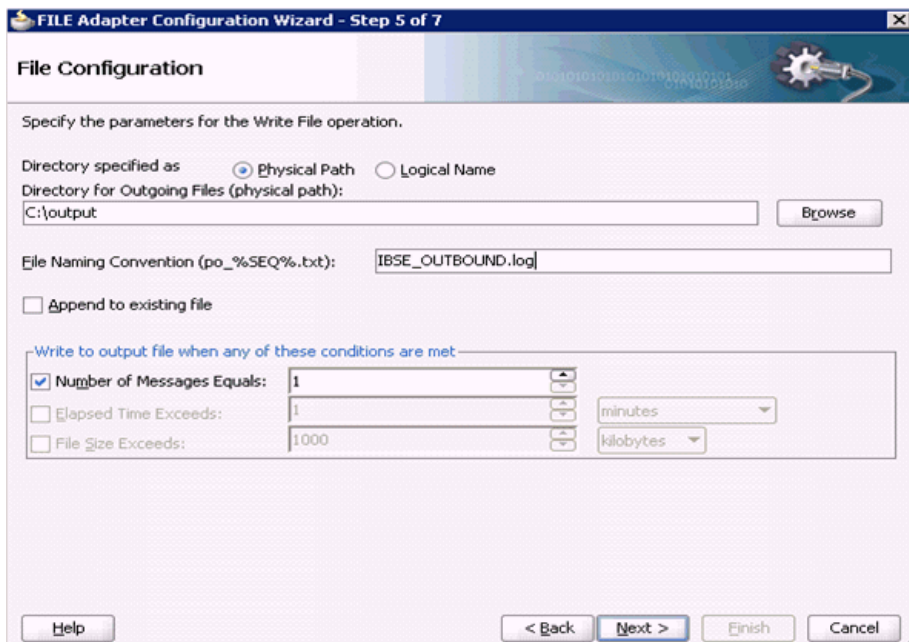
図 6-74 「操作」 ペイン



「ファイル構成」 ペインが表示されます。

5. 「発信ファイルのディレクトリ (物理パス)」フィールドに、出力ファイルが書き込まれるファイル・システム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 6-75 に示すように、「次へ」をクリックします。

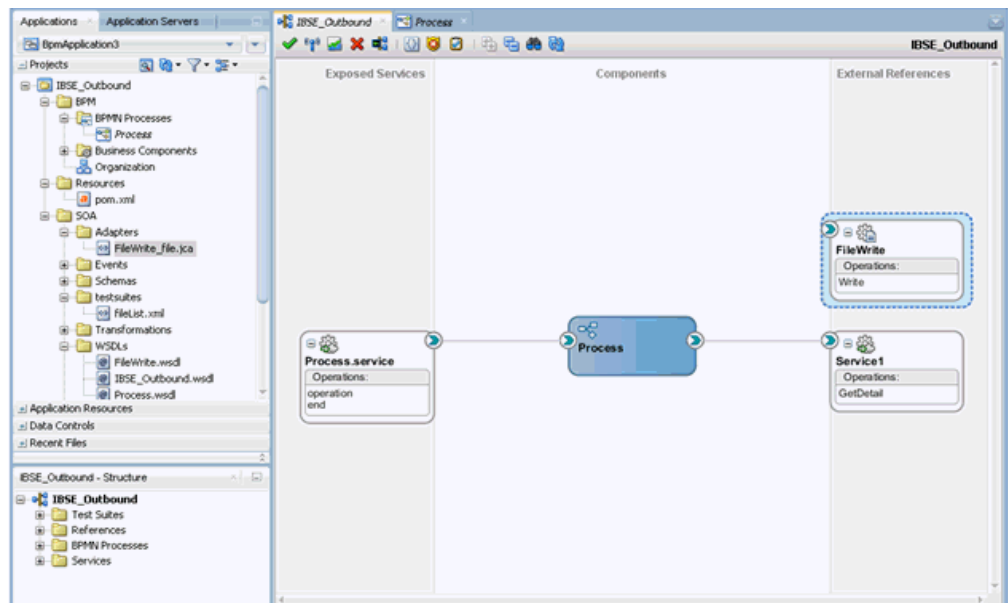
図 6-75 「ファイル構成」 ペイン



「メッセージ」ペインが表示されます。

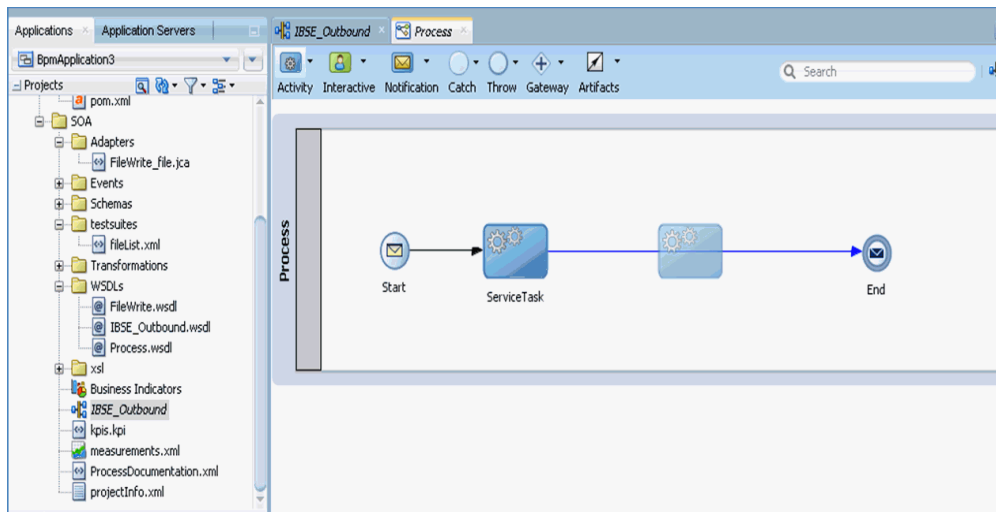
8. 「URL」フィールドの右の「参照」をクリックします。
9. 表示される「タイプ・チューザ」ウィンドウで、「プロジェクトの WSDL ファイル」、「IBSE_Outbound.wsdl」、「インライン・スキーマ」の順に展開し、「SiebelResponse」を選択します。
10. 「OK」をクリックします。
11. 「メッセージ」ペインで「次へ」をクリックします。
12. 表示される「終了」ペインで「終了」をクリックします。
13. 図 6-76 に示すように、「BPMN プロセス」コンポーネントをダブルクリックします。

図 6-76 「Composite.xml」タブ



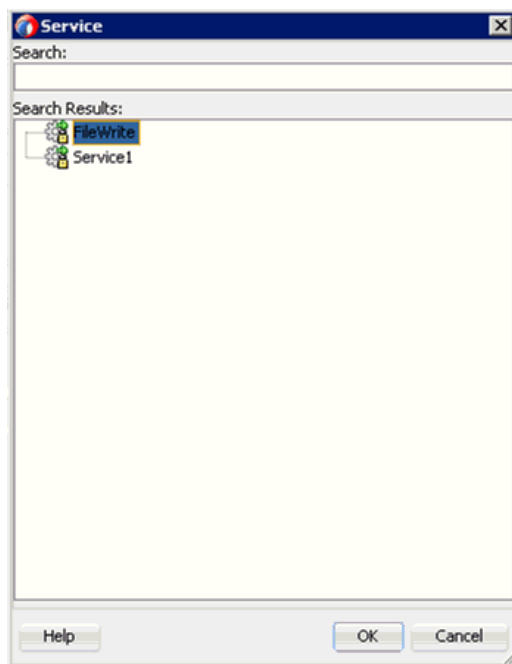
14. 「アクティビティ」アイコンをクリックします。
15. 図 6-77 に示すように、「サービス・タスク」イベント・コンポーネントと「終了」イベント・コンポーネントの間の伝送路に「アクティビティ」アイコンをドロップします。

図 6-77 「アクティビティ」アイコン



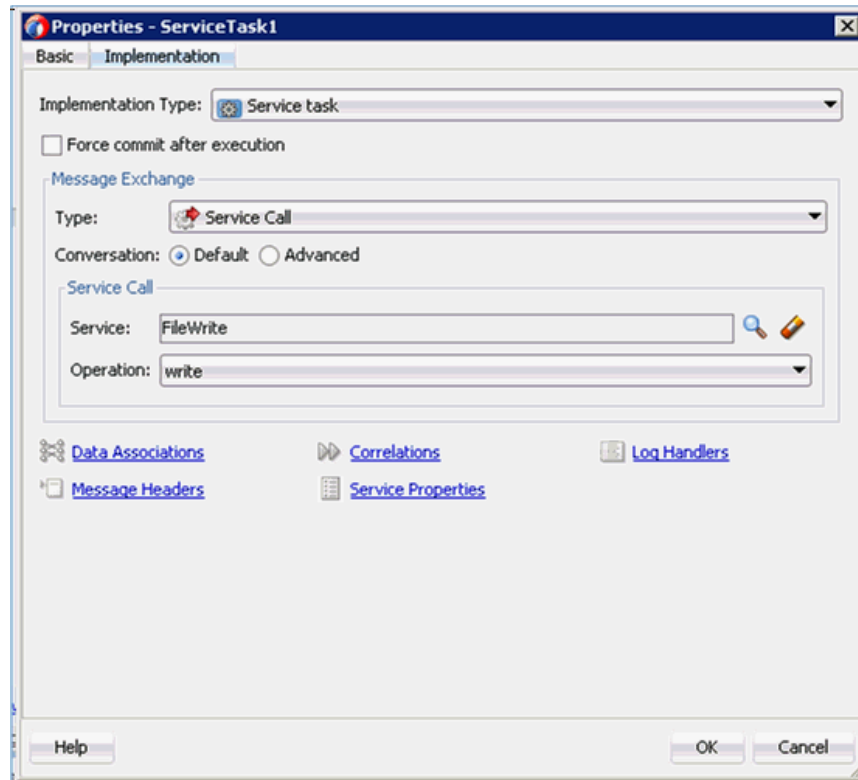
16. 表示される「プロパティ - ServiceTask1」ウィンドウで「実装」タブをクリックします。
17. 「メッセージ交換」セクションの「タイプ」ドロップダウン・リストから「サービス・コール」を選択します。
18. 「サービス」フィールドの右の「参照」アイコンをクリックします。
19. 図 6-78 に示すように、作成された書込み操作のサービスを選択し、「OK」をクリックします。

図 6-78 「サービス」ウィンドウ



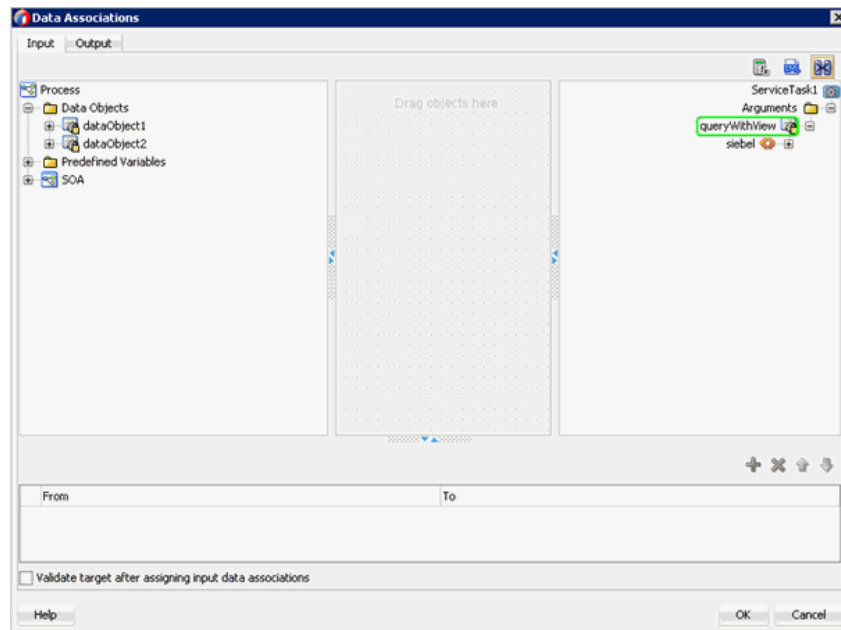
20. 図 6-79 に示すように、「プロパティ - ServiceTask1」ウィンドウで、「データ・アソシエーション」ハイパーリンクをクリックします。

図 6-79 データ・アソシエーション



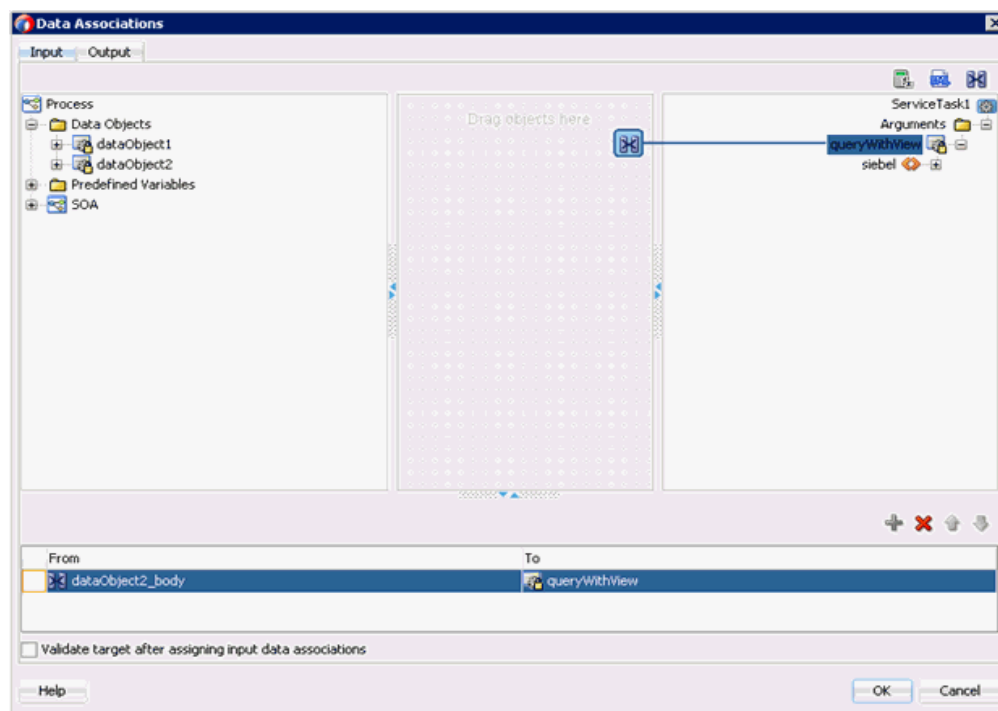
21. 「入力」タブで、右上のパインの「XSL 変換」アイコンをクリックします。
22. 図 6-80 に示すように、「XSL 変換」アイコンを「queryWithViewResponse」ノードにドラッグ・アンド・ドロップします。

図 6-80 「queryWithView」ノード



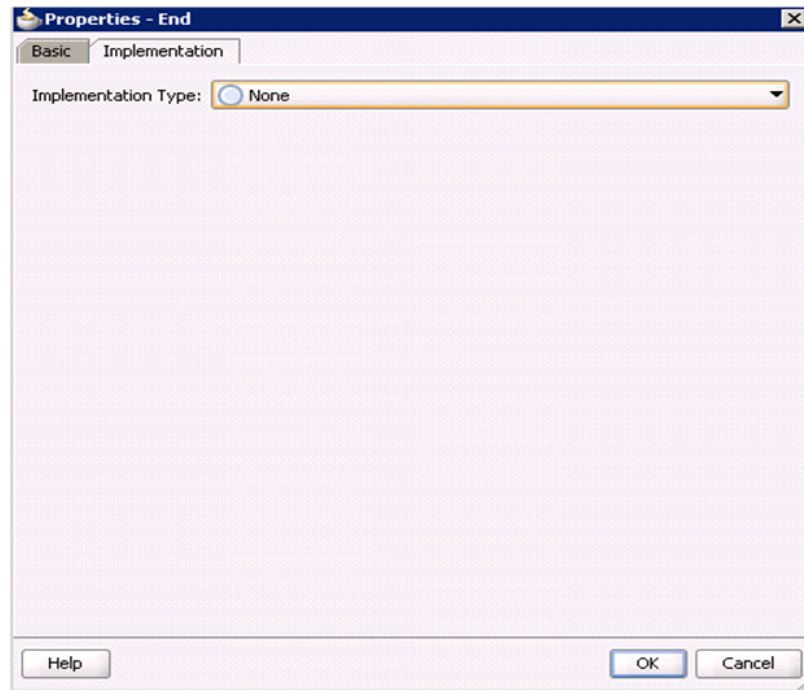
23. 表示される「変換の作成」セクションの「dataObject2」を選択して右矢印記号をクリックします。
24. 「OK」をクリックして、「ターゲット」ドロップダウン・リストで選択されているデフォルト値および「作成」フィールドのデフォルトの名前を受け入れます。
25. 図 6-81 に示すように、「データ・アソシエーション」ウィンドウで「OK」をクリックします。

図 6-81 「データ・アソシエーション」ウィンドウ



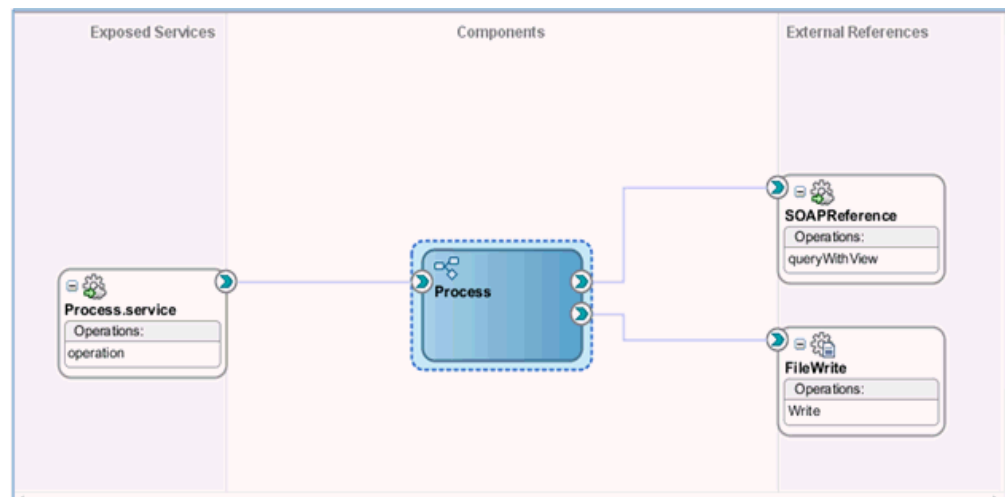
26. 「プロパティ - ServiceTask1」ウィンドウで「OK」をクリックします。
27. 「response_body.xml」タブで、「ns0:queryWithViewResponse」ソース要素を「ns0:queryWithViewResponse」ターゲット要素にマップします。
28. 表示された「自動マップ・プリファレンス」ウィンドウでデフォルト値を維持し、「OK」をクリックします。
29. 「プロセス」ワークスペース領域に戻り、「終了」イベント・コンポーネントをダブルクリックします。
30. 表示される「プロパティ - 終了」ウィンドウで「実装」タブをクリックします。
31. 「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 6-82 に示すように、「OK」をクリックします。

図 6-82 「実装」タブ



33. 図 6-83 に示すように、メニュー・バーの「すべて保存」アイコンをクリックし、構成された新規アウトバウンド BPM コンポーネントを保存します。

図 6-83 「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。6-25 ページ 6.4.4 項「BPM アウトバウンド・プロセスのデプロイ」の記載と同じ手順に従います。

デプロイすると、6-26 ページ 6.4.5 項「Oracle Enterprise Manager Console での入力 XML ドキュメントの起動」に定義されているように、入力 XML を起動できるようになります。

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 Siebel は Oracle Service Bus (OSB) とシームレスに統合され、Web サービスの簡易な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、次のとおりです。

- 7.1 項「Application Adapter と Oracle Service Bus の統合の概要」
- 7.2 項「sbconsole を使用したアウトバウンド処理の構成 (J2CA 構成)」
- 7.3 項「sbconsole を使用したインバウンド処理の構成 (J2CA 構成)」
- 7.4 項「sbconsole を使用したアウトバウンド処理の構成 (BSE 構成)」
- 7.5 項「Oracle Service Bus を使用した JMS プロキシ・サービスの構成 (J2CA 構成)」
- 7.6 項「Oracle Service Bus を使用した HTTP プロキシ・サービスの構成 (J2CA 構成)」

7.1 Application Adapter と Oracle Service Bus の統合の概要

Oracle Service Bus (OSB) を統合するには、Oracle Application Adapter for Siebel が OSB と同じ Oracle WebLogic Server にデプロイされている必要があります。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) で設計時に生成される WSDL ファイルとして公開される必要があります。

7.2 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 構成の場合の sbconsole を使用したアウトバウンド処理の構成方法について説明します。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB\J2CA\Siebel_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。

- 7.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」
- 7.2.2 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」
- 7.2.3 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」
- 7.2.4 項「WSDL ベースのビジネス・サービスの構成」
- 7.2.5 項「ファイル・タイプ・ビジネス・サービスの構成」
- 7.2.6 項「プロキシ・サービスでのパイプラインの構成」

7.2.1 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、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
```

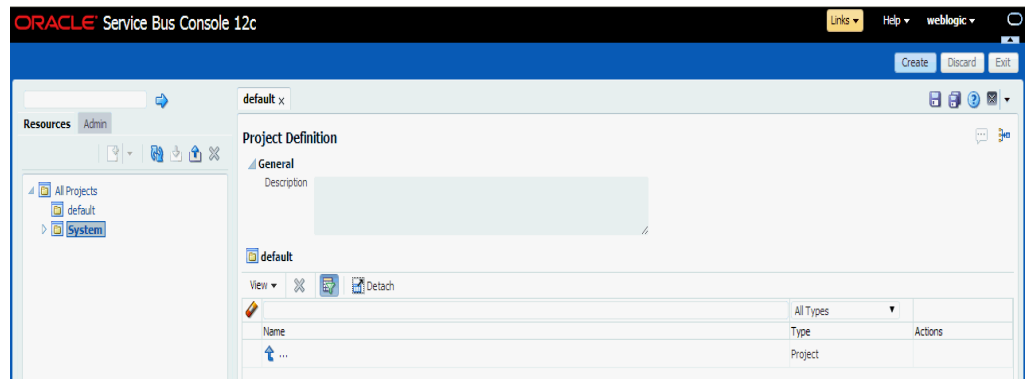
ここで、*hostname* は Oracle WebLogic Server が稼働しているマシンの名前、*port* は使用しているドメイン用のポートです。

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。

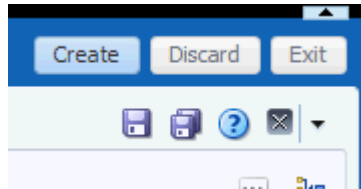
図 7-1 に示すように、Oracle Service Bus コンソールのホーム・ページが表示されます。

図 7-1 Oracle Service Bus コンソールのホーム・ページ



4. 図 7-2 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 7-2 Oracle Service Bus セッション



5. 図 7-3 に示すように、「すべてのプロジェクト」を選択し、左ペインの下矢印をクリックし、「プロジェクト」を選択します。

図 7-3 「すべてのプロジェクト」フォルダ

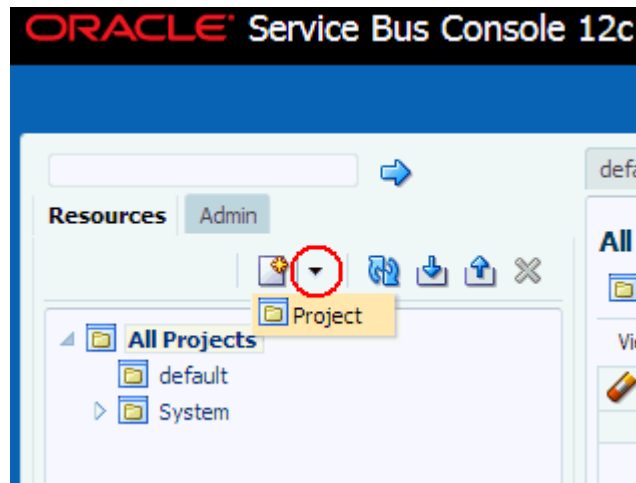
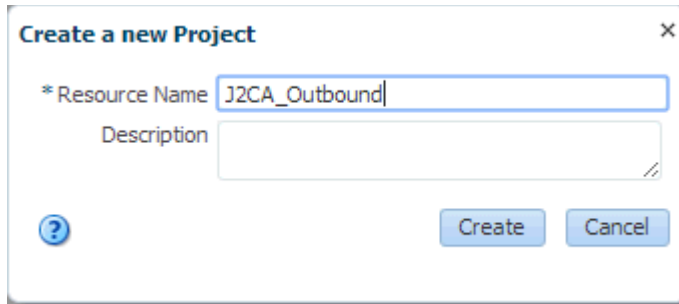


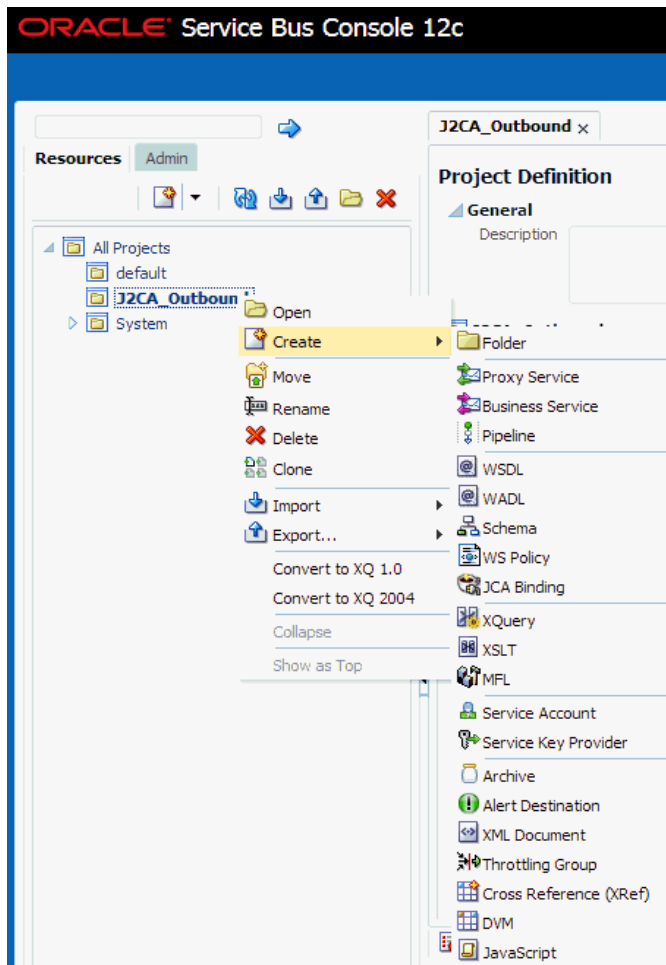
図 7-4 に示すように、「新規プロジェクトの作成」ウィンドウが表示されます。

図 7-4 「新規プロジェクトの作成」 ウィンドウ



6. 「リソース名」フィールドに新規プロジェクトの有効な名前 (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 7-5 に示すように、新しく作成したプロジェクトを右クリックして、「作成」を選択し、「フォルダ」をクリックします。

図 7-5 「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および8を繰り返し、「Proxy Service」および「Wsdls」という名前のフォルダを作成します。

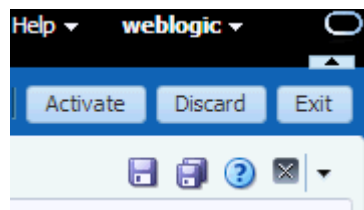
図7-6に示すように、「ビジネス・サービスフォルダが左ペインのプロジェクト・ノードの下に表示されます。

図7-6 プロジェクト・ノード



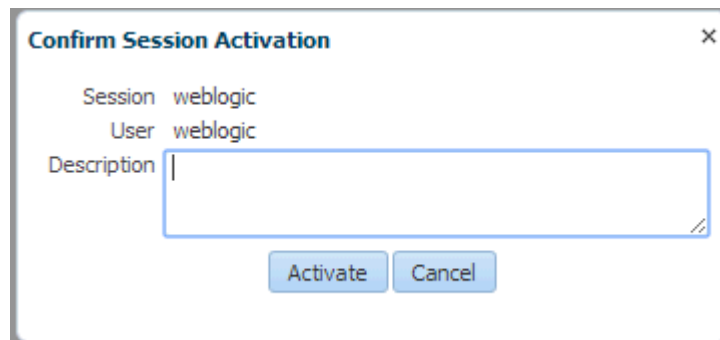
10. 図7-7に示すように、Oracle Service Bus セッションの右ペインで「アクティブ化」をクリックします。

図7-7 「アクティブ化」ボタン



11. 図7-8に示すように、セッションのアクティブ化の確認ページで、「アクティブ化」をクリックして変更を保存します。

図7-8 セッションのアクティブ化の確認ウィンドウ



7.2.2 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) Console(プロジェクトまたはフォルダ)に公開する前に、OSB ユーザーは次のステップを実行する必要があります。

1. コマンド・プロンプト・ウィンドウを開きます。
2. 次のディレクトリに移動します。
`<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。

7.2.3 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、Siebel ターゲットに接続します。
 詳細は、2-1 ページ第 2 章「Oracle Application Server Adapter for Siebel 構成」を参照してください。
2. 接続先の Siebel ターゲットを展開します。
3. 「ビジネス・オブジェクト」、「アカウント」、「アカウント」の順に展開します。
4. 「queryWithView」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 7-9 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 7-9 「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表示されます。デフォルトを受け入れるか、任意の名前を指定します。
6. 「OSB」オプションを選択します。
7. 「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。

このロケーションは、Oracle Service Bus のプロジェクト名と、オプションで 1 つ以上のフォルダ名で構成されます。プロジェクト名および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。

8. 「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。
9. 「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。
10. 「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。
11. 「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。
12. 「OK」をクリックします。

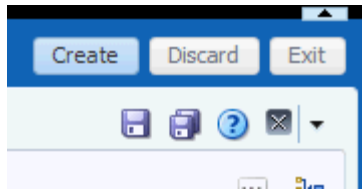
WSDL が「WSDL のエクスポート」ダイアログで指定されている場所に公開され、Oracle Service Bus でビジネス・サービスまたはプロキシ・サービスとともに使用できるようになります。

7.2.4 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。

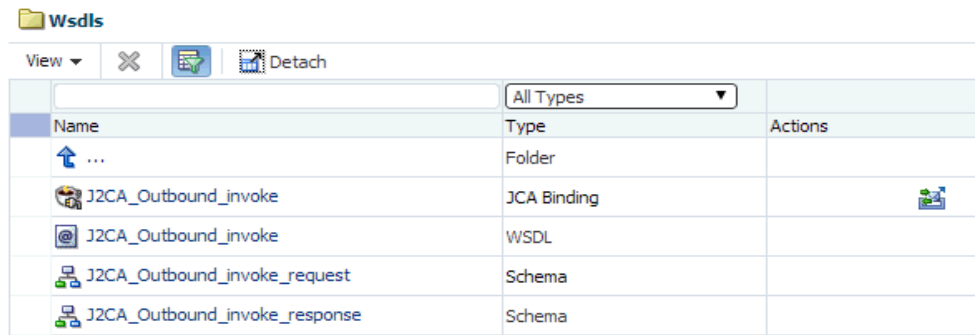
1. 図 7-10 に示すように、Oracle Service Bus セッションの右ペインで、Oracle Service Bus コンソールを開き、「作成」をクリックします。

図 7-10 「作成」 ボタン



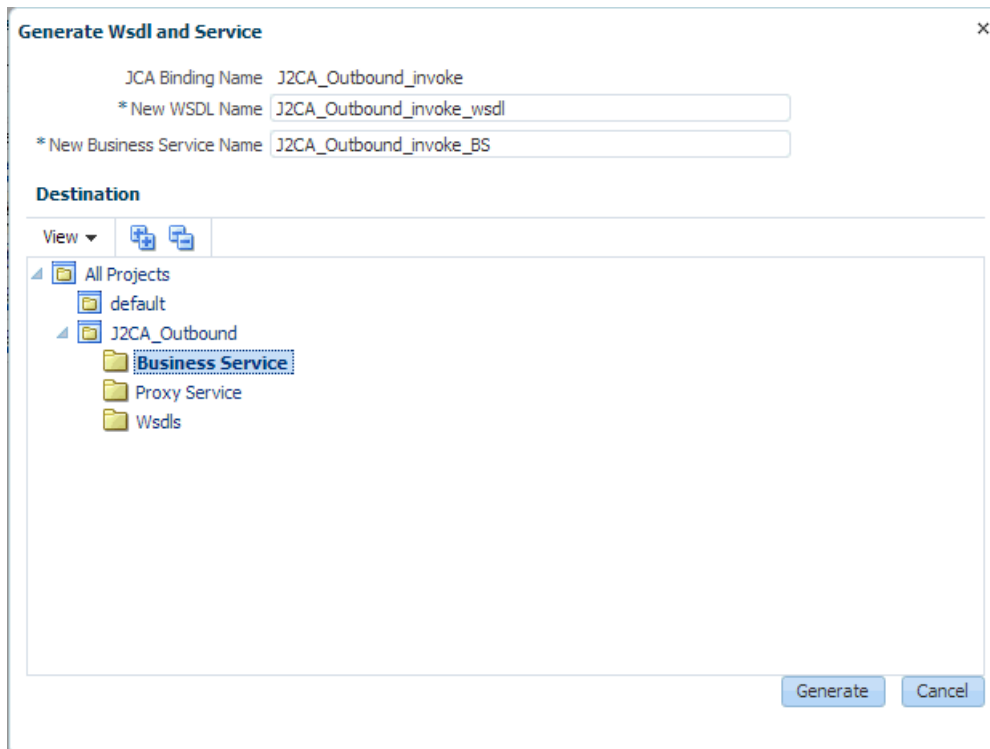
2. 図 7-11 に示すように、左ペインで作成済みの WSDL フォルダ (Wsdls など) をダブルクリックして、エクスポートした WSDL が右ペインに表示されることを確認します。

図 7-11 Wsdls フォルダ



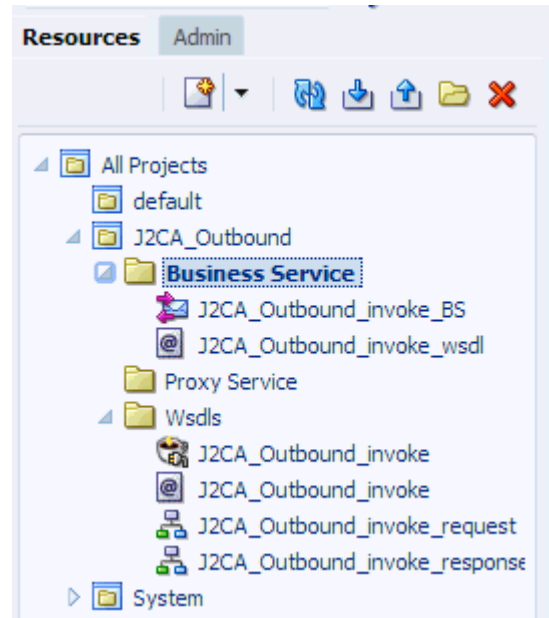
3. 「アクション」列で、JCA バインドに対応するアイコンをクリックします。
 図 7-12 に示すように、「WSDL とサービスの生成」ウィンドウが表示されます。

図 7-12 「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・サービスの名前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. 図 7-13 に示すように、プロジェクト・フォルダの下で「ビジネス・サービス」を展開し、生成した WSDL とビジネス・サービスが表示されていることを確認します。

図 7-13 「ビジネス・サービス」フォルダ

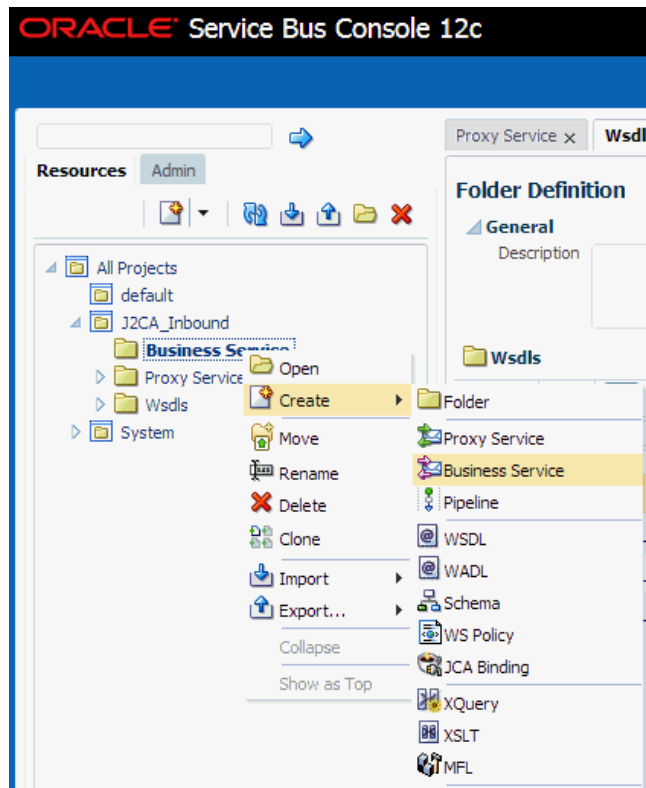


7.2.5 ファイル・タイプ・ビジネス・サービスの構成

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

1. 図 7-14 に示すように、左ペインで作成済みの「ビジネス・サービス」フォルダを右クリックして、「作成」を選択し、「ビジネス・サービス」をクリックします。

図 7-14 「ビジネス・サービス」フォルダ



「ビジネス・サービス」ウィンドウが表示されます。

2. 図 7-15 に示すように、「リソース名」フィールドにビジネス・サービスの名前を入力し、サービス定義の下にある「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 7-15 サービス定義

Create Business Service

Create Type Transport

Create Service

*Resource Name File_Out

Description

Service Definition

WSDL Based Service

Name

Path

Port/Binding

Transport file

Back Next Create Cancel

- 「サービス・タイプ」セクションで「メッセージ・サービス」を選択します。デフォルトでは、「リクエスト・タイプ」に設定されます。図 7-16 に示すように、「次へ」をクリックします。

図 7-16 「サービス・タイプ」の構成ページ

Create Business Service

Create Type Transport

Service Type

WSDL Based Service

Any SOAP Service

Any XML Service

Messaging Service

Request Type XML

Schema Name

Path

Element/Type

Response Type None

Back Next Create Cancel

- 「エンドポイント URI」フィールドで、ファイル・システム上の宛先フォルダへのパスを入力します。

5. 図 7-17 に示すように、「作成」をクリックします。

図 7-17 「トランスポート」 ページ

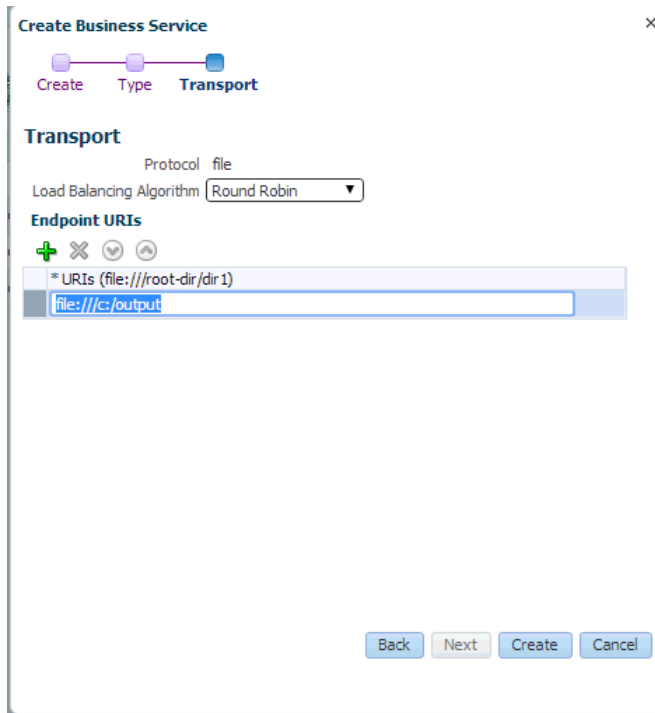
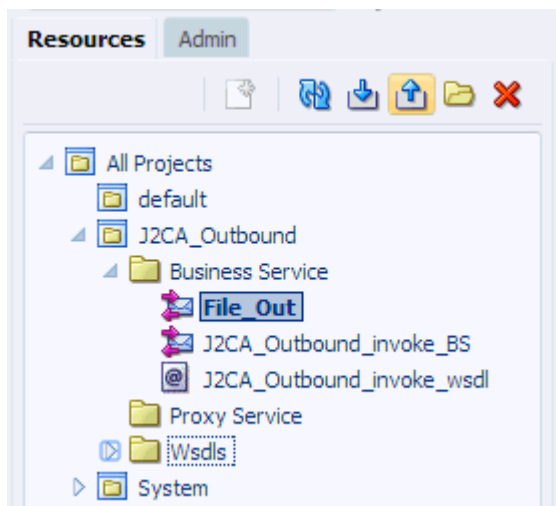


図 7-18 に示すように、「ビジネス・サービス」の下にビジネス・サービス **File_Out** が作成されて表示されます。

図 7-18 File_Out ビジネス・サービス



6. 図 7-19 に示すように、**File_Out** をダブルクリックして、左ペインのトランスポートの詳細をクリックし、受信される出力ファイルの接頭辞および接尾辞を入力します。

図 7-19 トランスポートの詳細

Business Service Definition

Configuration | SLA Alert Rules

General
Transport
Transport Detail
Message Handling
Performance

Transport Detail

Prefix

Suffix

Request encoding

7. 図 7-20 に示すように、右隅の「保存」または「すべて保存」アイコンをクリックします。

図 7-20 「保存」アイコン

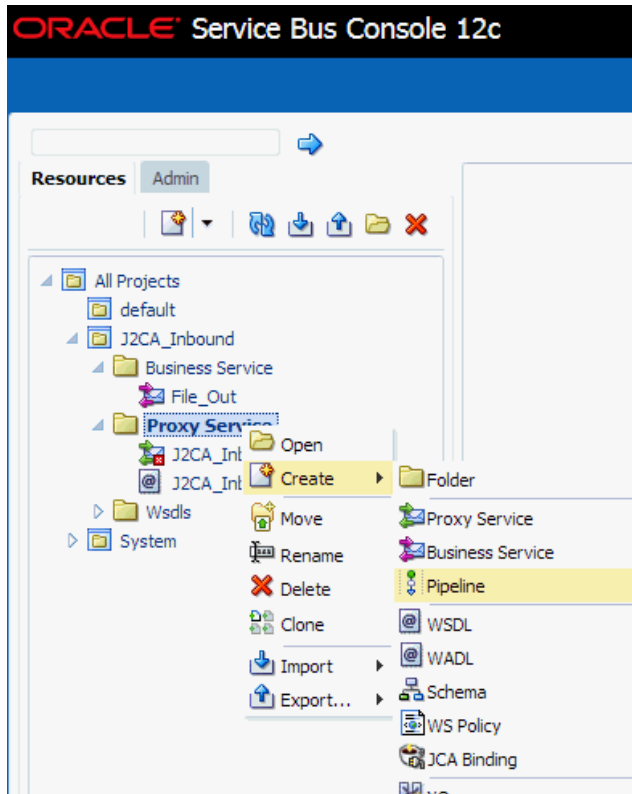


7.2.6 プロキシ・サービスでのパイプラインの構成

パイプラインを構成するには、次のステップを実行します

1. 図 7-21 に示すように、「プロキシ・サービス」フォルダを右クリックして、「作成」を選択し、「パイプライン」をクリックします。

図 7-21 「パイプライン」 オプション



「パイプライン」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**が選択されます。図 7-22 に示すように、プロキシ・サービス名を変更する場合は、プロキシ・サービス名を変更して、「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 7-22 「パイプライン」ウィンドウ

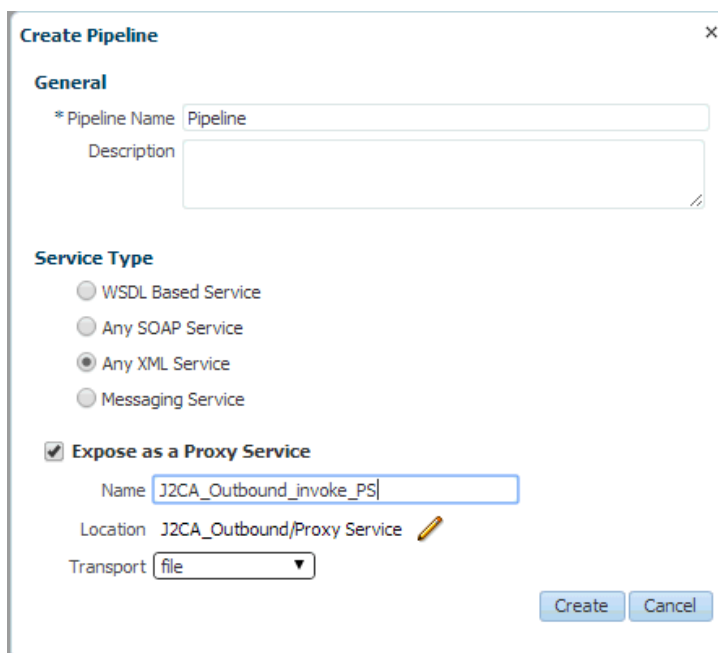
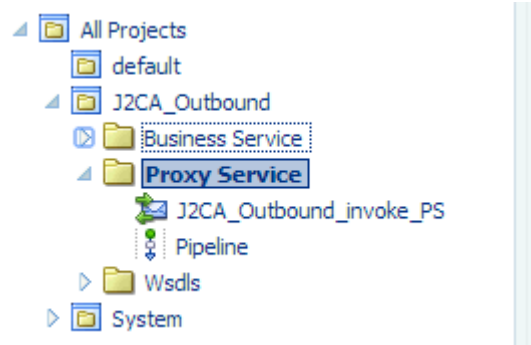


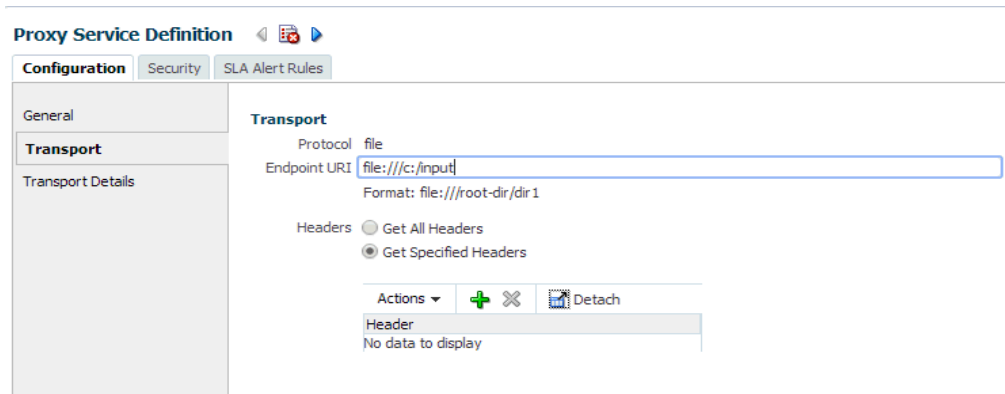
図 7-23 に示すように、作成したパイプラインとプロキシ・サービスが「プロキシ・サービス」の下に表示されます。

図 7-23 「パイプライン」ノード



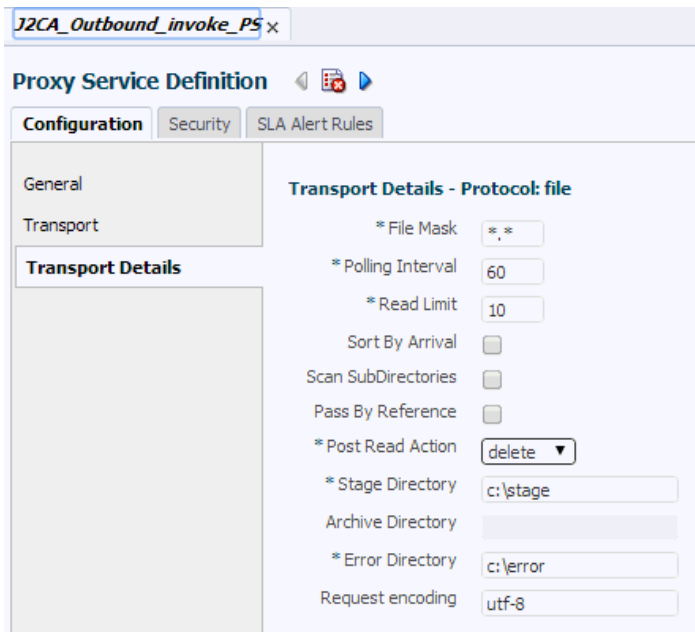
- 作成したプロキシ・サービスをダブルクリックし、左ペインで「トランスポート」をクリックします。図 7-24 に示すように、「エンドポイント URI」フィールドに入力場所を入力します。

図 7-24 トランスポート



- 図 7-25 に示すように、左ペインで「トランスポートの詳細」をクリックして、「ステージ・ディレクトリ」フィールドに場所を入力します。

図 7-25 トランスポートの詳細



5. 図 7-26 に示すように、右隅の「すべて保存」アイコンをクリックします。

図 7-26 「すべて保存」アイコン



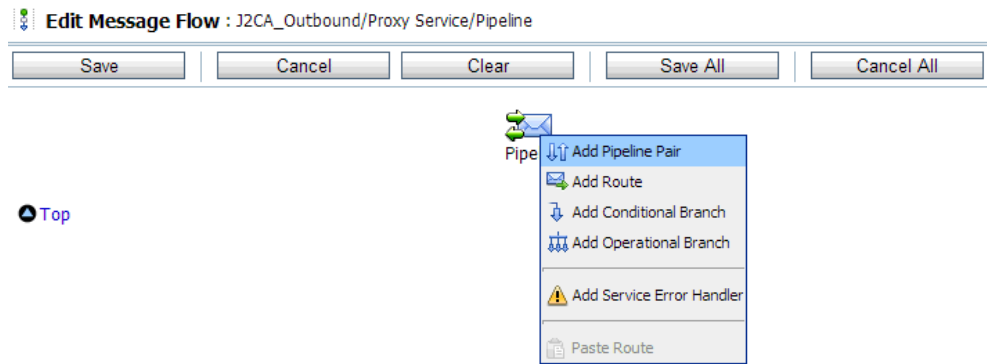
6. 図 7-27 に示すように、「パイプライン」ノードをダブルクリックして、右ペインの「メッセージ・フローを開く」をクリックし、メッセージ・フローを開きます。

図 7-27 「メッセージ・フロー」アイコン



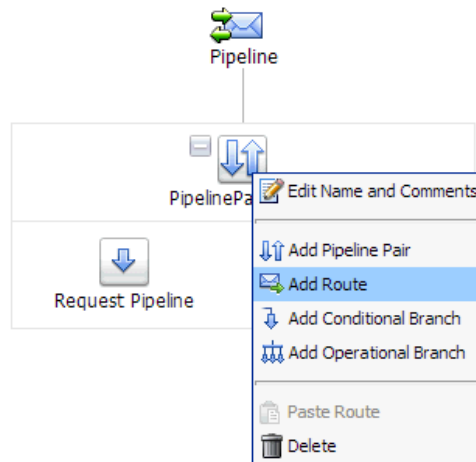
7. 図 7-28 に示すように、プロキシ・サービス・アイコンをクリックして、メニューから「パイプライン・ペアの追加」を選択します。

図 7-28 「パイプライン・ペアの追加」オプション



8. 図 7-29 に示すように、**PipelinePairNode1** アイコンをクリックして、メニューから「ルートへの追加」を選択します。

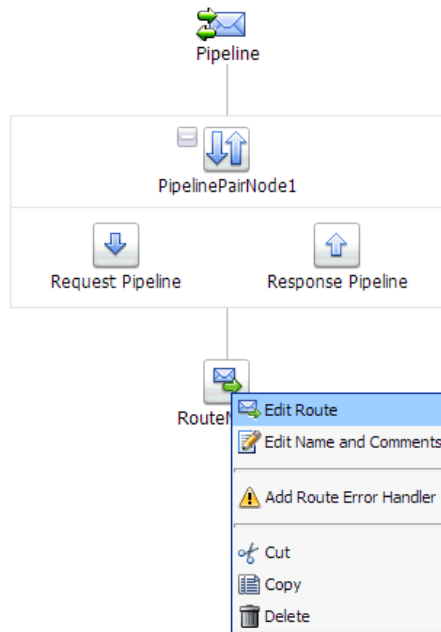
図 7-29 「ルートへの追加」オプション



RouteNode1 アイコンの下に追加されます。

9. 図 7-30 に示すように、**RouteNode1** アイコンをクリックして、メニューから「ルートへの編集」を選択します。

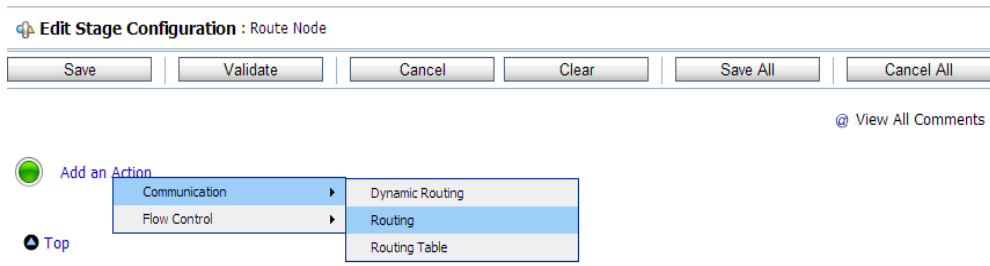
図 7-30 「ルート編集」オプション



「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 7-31 に示すように、「アクションの追加」をクリックして、「通信」を選択し、「ルーティング」をクリックします。

図 7-31 「ステージ構成の編集」ワークスペース領域



11. 図 7-32 に示すように、<サービス>をクリックします。

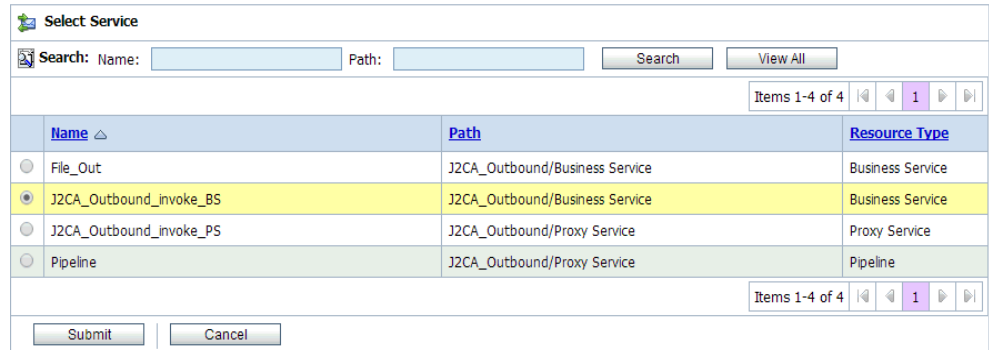
図 7-32 アクション



「サービスの選択」ダイアログが表示されます。

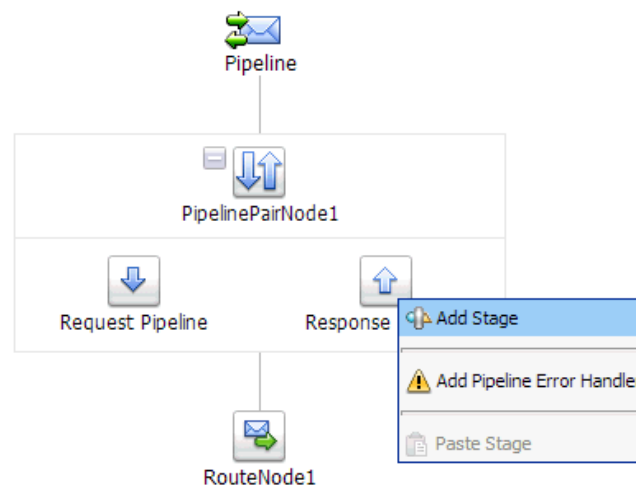
12. 図 7-33 に示すように、Siebel 用に構成されている WSDL タイプのビジネス・サービスを選択し、「発行」をクリックします。

図 7-33 「サービスの選択」ダイアログ



13. リストから操作属性として Siebel ビジネス・オブジェクトの名前 (queryWithView など) を選択し、「保存」をクリックします。
14. 図 7-34 に示すように、「レスポンス・パイプライン」アイコンをクリックして、メニューから「ステージの追加」を選択します。

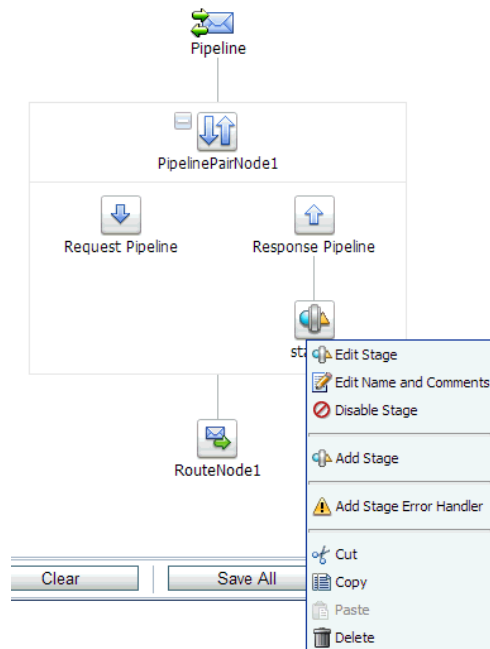
図 7-34 「レスポンス・パイプライン」アイコン



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 7-35 に示すように、「ステージ 1」アイコンをクリックして、メニューから「ステージの編集」を選択します。

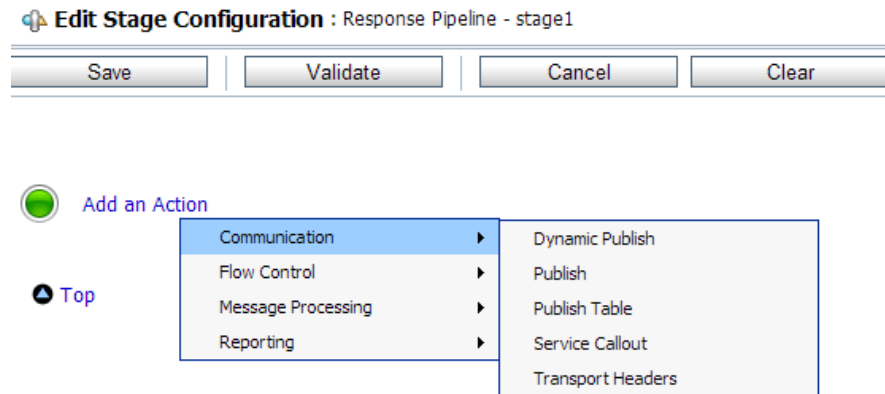
図 7-35 「ステージの編集」オプション



「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 7-36 に示すように、「アクションの追加」をクリックして、「通信」を選択し、「パブリッシュ」をクリックします。

図 7-36 「ステージ構成の編集」ワークスペース領域



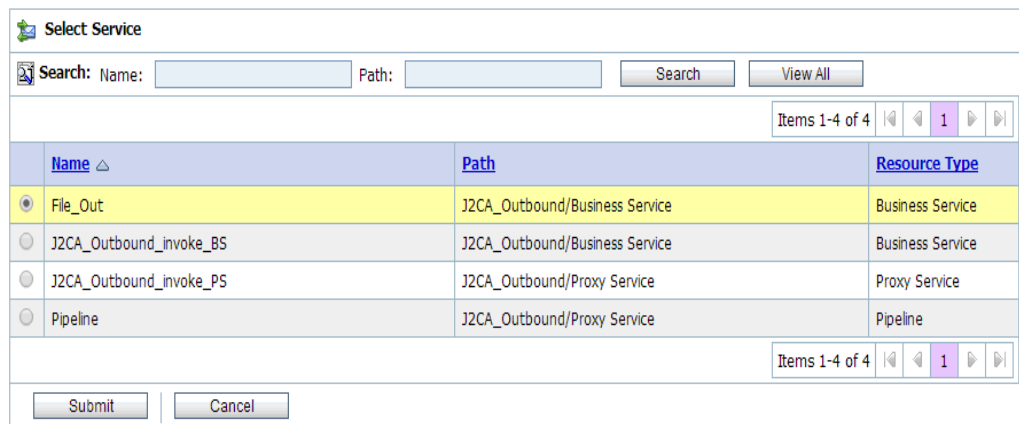
17. 図 7-37 に示すように、<サービス>をクリックします。

図 7-37 < サービス > アクション



18. 図 7-38 に示すように、「サービスの選択」ダイアログで、ファイル・タイプ・ビジネス・サービスを選択して、「発行」をクリックします。

図 7-38 「サービスの選択」ダイアログ

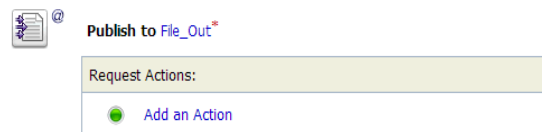


19. 図 7-39 に示すように、「すべて保存」をクリックします。

図 7-39 「すべて保存」ボタン

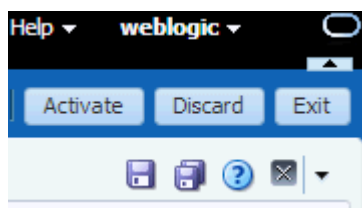


@ View



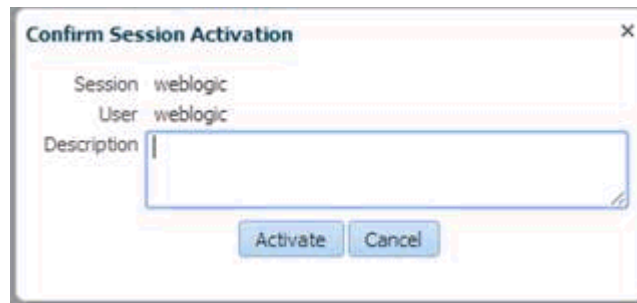
20. 図 7-40 に示すように、Oracle Service Bus セッションの右ペインで「アクティブ化」をクリックします。

図 7-40 「アクティブ化」ボタン



21. 図 7-41 に示すように、「アクティブ化」をクリックして、変更を保存します。

図 7-41 セッションのアクティブ化の確認



22. 入力 XML ファイルをコピーし、構成した入力フォルダ (C:\input など) に貼り付けます。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。

7.3 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 構成の場合の sbconsole を使用したインバウンド処理の構成方法について説明します。

このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストレーションの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB\J2CA\Siebel_Sample_J2CA_OSB_Inbound_Project
```

この項では、次のトピックについて説明します。

- 7.3.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」
- 7.3.2 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」
- 7.3.3 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」
- 7.3.4 項「WSDL ベースのプロキシ・サービスの構成」
- 7.3.5 項「ファイル・タイプ・ビジネス・サービスの構成」
- 7.3.6 項「パイプラインの構成」

7.3.1 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、7-2 ページ 7.2.1 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。

7.3.2 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・パスの設定の詳細は、7-6 ページ 7.2.2 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。

7.3.3 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、Siebel ターゲットに接続します。
詳細は、2-1 ページ第 2 章「Oracle Application Server Adapter for Siebel 構成」を参照してください。
2. Siebel チャンネルを作成します。
詳細は、4-35 ページ 4.5.1.1 項「チャンネルの作成」を参照してください。
3. 統合オブジェクト・ノードを作成します。詳細は、4-38 ページ 4.5.1.2 項「統合オブジェクト・ノードの作成」を参照してください。
4. 作成した統合ノードを右クリックして、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 7-42 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 7-42 「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表示されます。デフォルトを受け入れるか、任意の名前を指定します。
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」検証オプションのチェック・ボックスが 3 つ表示されます。複数の検証オプションを選択できます。

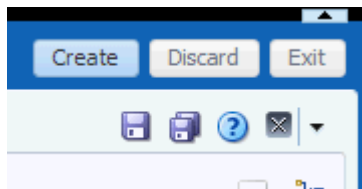
8. 「OSB」 オプションを選択します。
9. 「ロケーション」 フィールドで、WSDL ドキュメントの公開先となる Oracle Service Bus 内のフォルダ名を入力します。
このロケーションは、Oracle Service Bus のプロジェクト名と、オプションで1つ以上のフォルダ名で構成されます。プロジェクト名およびフォルダ名はいずれもスラッシュ文字の「/」で区切る必要があります。
10. 「ホスト」 フィールドで、Oracle Service Bus がインストールされているマシンの名前を入力します。
11. 「ポート」 フィールドで、Oracle Service Bus に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。

7.3.4 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。

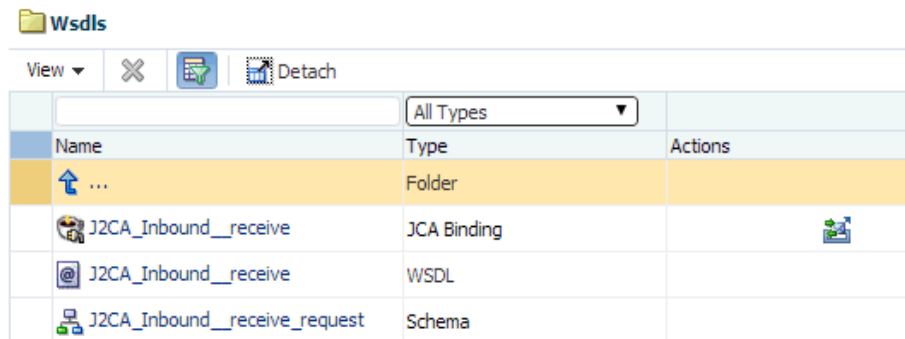
1. [図 7-43](#) に示すように、Oracle Service Bus セッションの右ペインで、Oracle Service Bus コンソールを開き、「作成」 をクリックします。

図 7-43 「作成」 ボタン



2. [図 7-44](#) に示すように、左ペインで作成済みの WSDL フォルダ (Wsdls など) をダブルクリックして、エクスポートした WSDL が右ペインに表示されることを確認します。

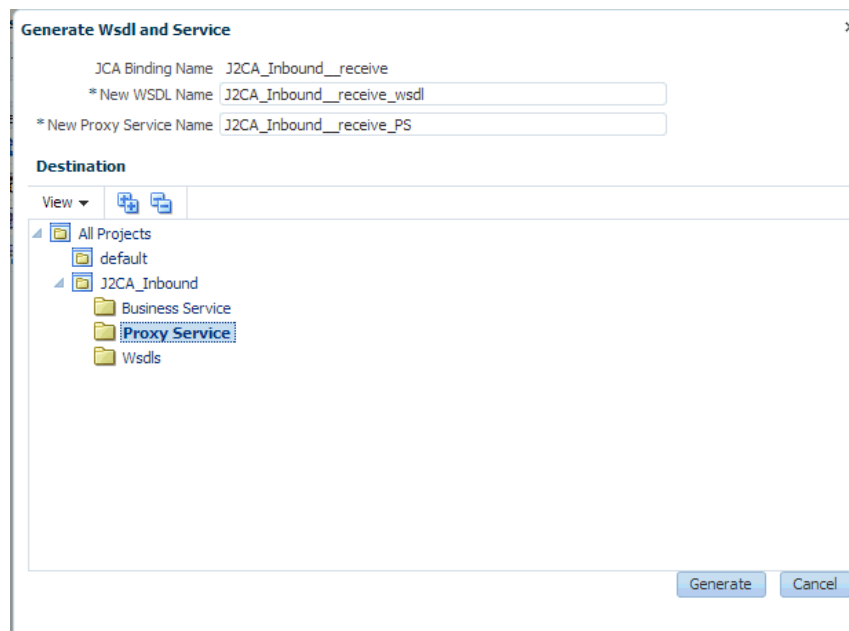
図 7-44 エクスポートされた WSDL



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。

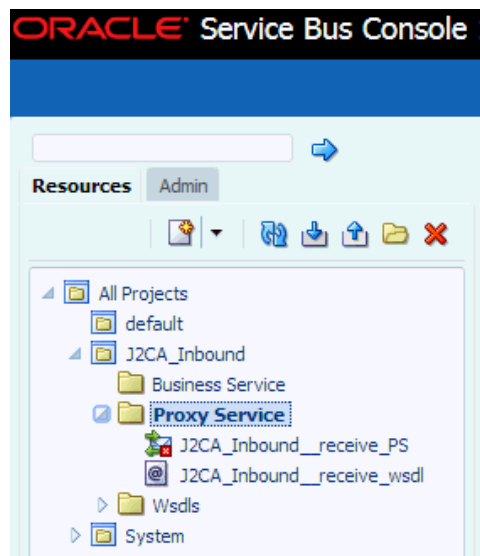
図 7-45 に示すように、「WSDL とサービスの生成」ページが表示されます。

図 7-45 「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・サービスの名前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 7-46 に示すように、プロジェクト・エクスプローラの下で「プロキシ・サービス」を展開し、生成した WSDL とプロキシ・サービスが表示されていることを確認します。

図 7-46 生成された WSDL

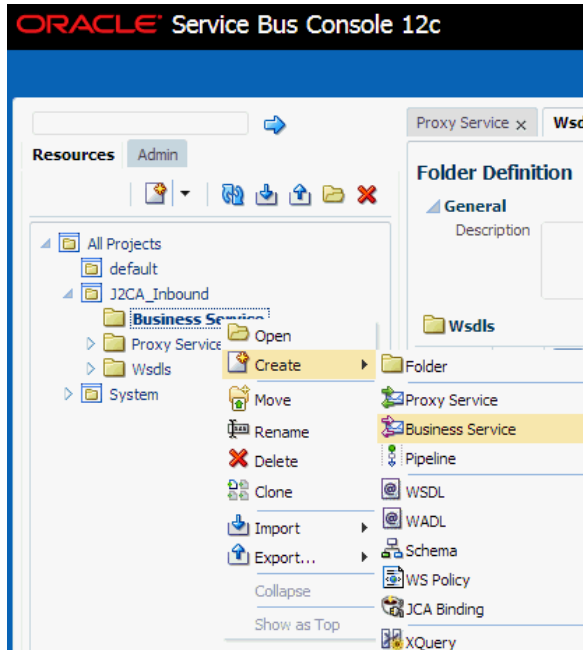


7.3.5 ファイル・タイプ・ビジネス・サービスの構成

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

1. 図 7-47 に示すように、左ペインで作成済みの「ビジネス・サービス」フォルダを右クリックして、「作成」を選択し、「ビジネス・サービス」をクリックします。

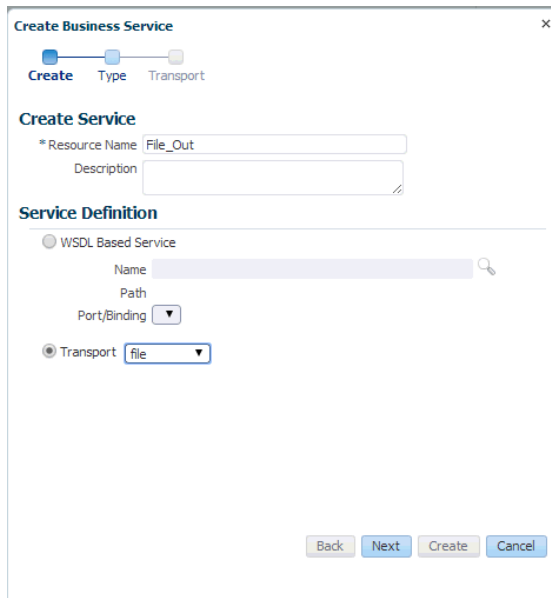
図 7-47 「ビジネス・サービス」フォルダ



「ビジネス・サービス」ウィンドウが表示されます。

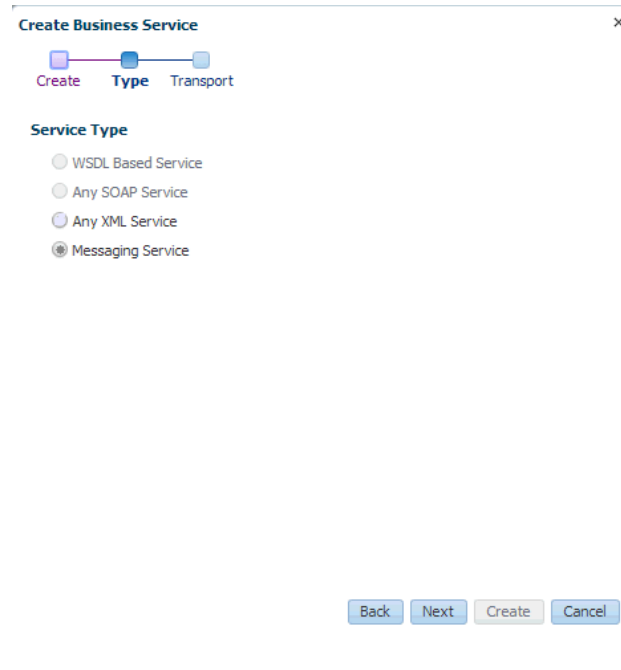
2. 図 7-48 に示すように、「リソース名」フィールドにビジネス・サービスの名前を入力し、サービス定義領域の「トランスポート」ドロップダウン・リストから「ファイル」オプションを選択します。

図 7-48 「ビジネス・サービス」ウィンドウ



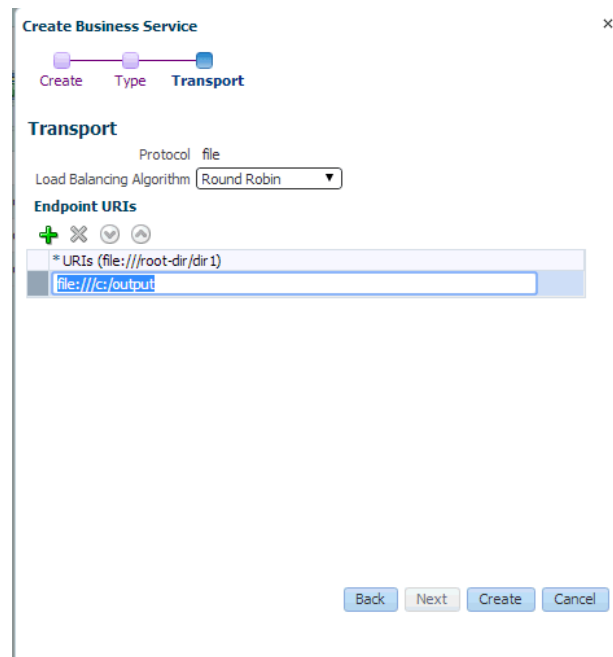
3. 「次へ」をクリックします。
4. 図 7-49 に示すように、サービス・タイプとして「メッセージ・サービス」を選択します。

図 7-49 「サービス・タイプ」領域



5. 「次へ」をクリックします。
図 7-50 に示すように、「トランスポート」ページが表示されます。

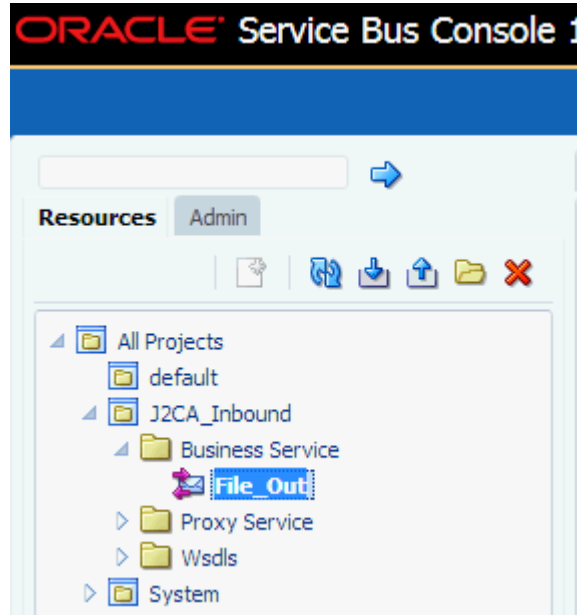
図 7-50 「トランスポート」ページ



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。

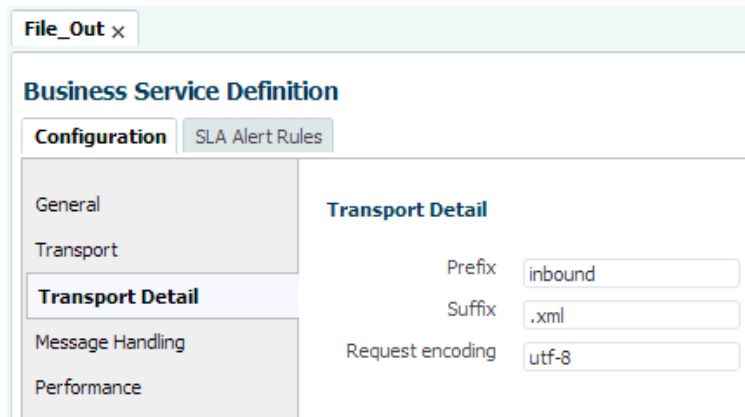
図 7-51 に示すように、「ビジネス・サービス」が作成されて表示されます。

図 7-51 File_Out ビジネス・サービス



7. 図 7-52 に示すように、「File_Out」をダブルクリックして、左ペインのトランスポートの詳細をクリックし、受信される出力ファイルの接頭辞および接尾辞を入力します。

図 7-52 トランスポートの詳細ページ



8. 図 7-53 に示すように、右隅の「保存」アイコンをクリックします。

図 7-53 「保存」アイコン

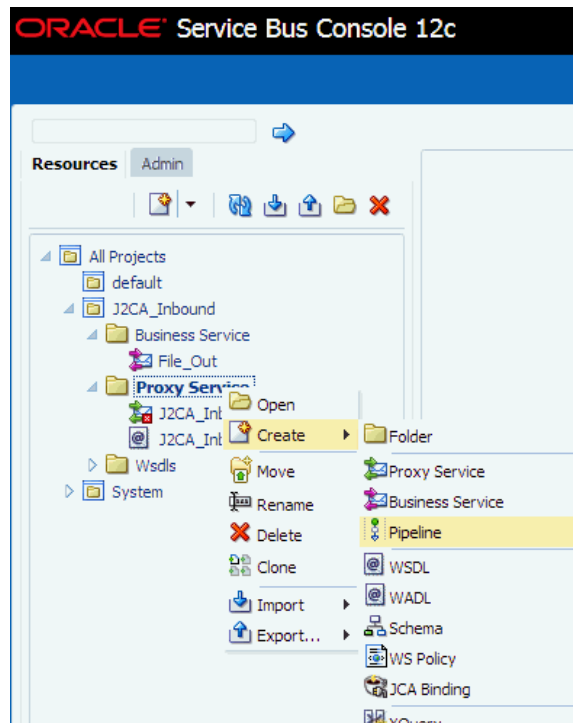


7.3.6 パイプラインの構成

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 7-54 に示すように、作成したプロキシ・サービスを右クリックして、「作成」を選択し、「パイプライン」をクリックします。

図 7-54 「パイプライン」オプション



「パイプライン」ウィンドウが表示されます。

2. 図 7-55 に示すように、「パイプライン名」で WSDL ベースのサービスを選択します。

図 7-55 「パイプライン」 ウィンドウ

3. 図 7-56 に示すように、「検索」アイコンをクリックし、表示された検索と選択: 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 7-56 検索と選択: WSDL リソース・ウィンドウ

Name	Path	Namespace
J2CA_Inbound__receive_wsdl	J2CA_Inbound/Proxy Service	http://xmlns.oracle...
J2CA_Inbound__receive	J2CA_Inbound/Wsdls	http://xmlns.oracle...

OK Cancel

「パイプライン」ウィンドウが表示されます。

4. 図 7-57 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、「作成」をクリックします。

図 7-57 「パイプライン」 ウィンドウ

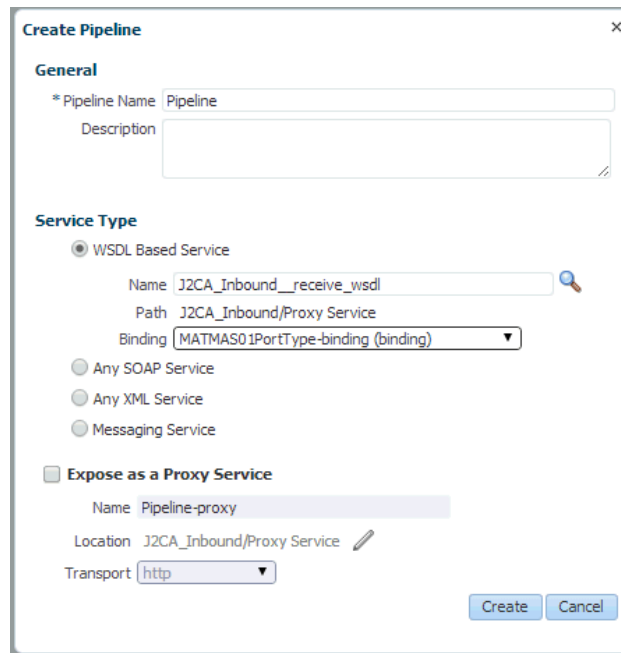
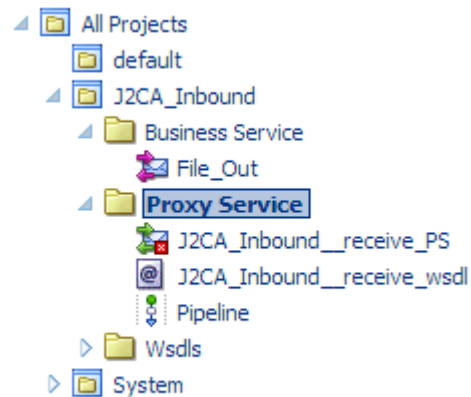


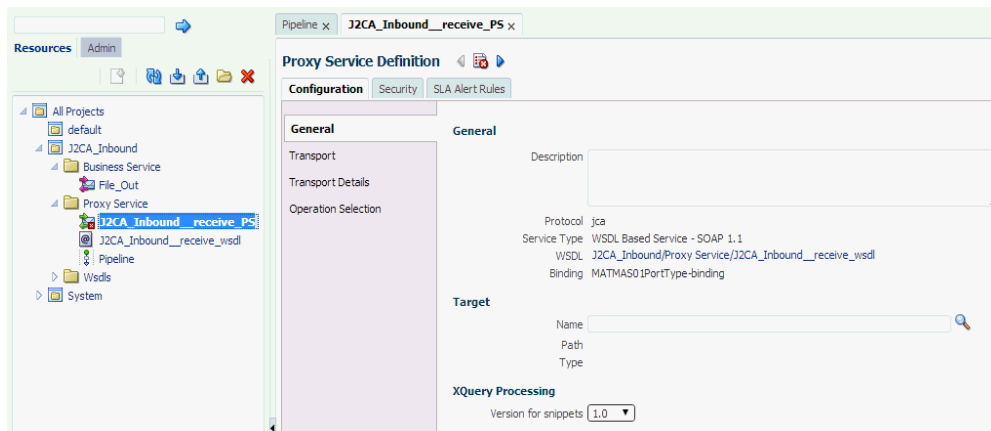
図 7-58 に示すように、「プロキシ・サービス」の下にパイプラインが作成されて表示されます。

図 7-58 プロキシ・サービスのパイプライン



5. 図 7-59 に示すように、左ペインにある「プロキシ・サービス」の下で **J2CA_Inbound_receive_PS** ノードをダブルクリックし、右ペインにある「ターゲット」領域の「検索」アイコンをクリックします。

図 7-59 プロキシ・サービスの定義ウィンドウ

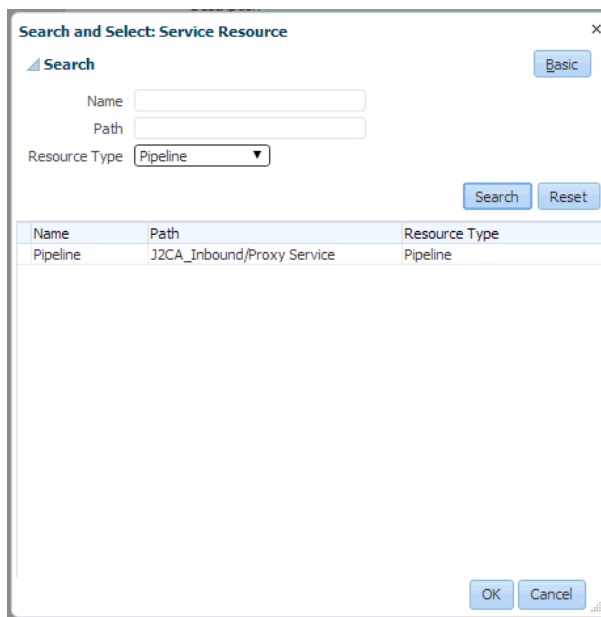


検索と選択: サービス・リソース・ウィンドウが表示されます。

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」を選択してから、「検索」ボタンをクリックします。

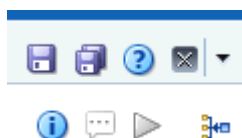
図 7-60 に示すように、パイプラインが表示されます。

図 7-60 検索と選択: サービス・リソース・ウィンドウ



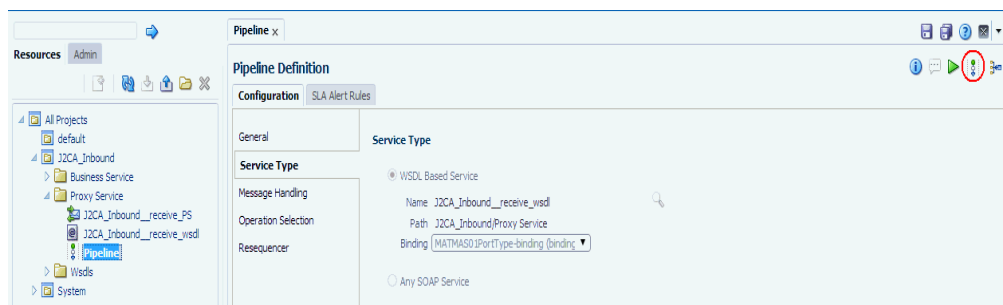
7. パイプラインを選択して、「OK」をクリックします。
8. 図 7-61 に示すように、右隅の「保存」アイコンをクリックします。

図 7-61 「保存」アイコン



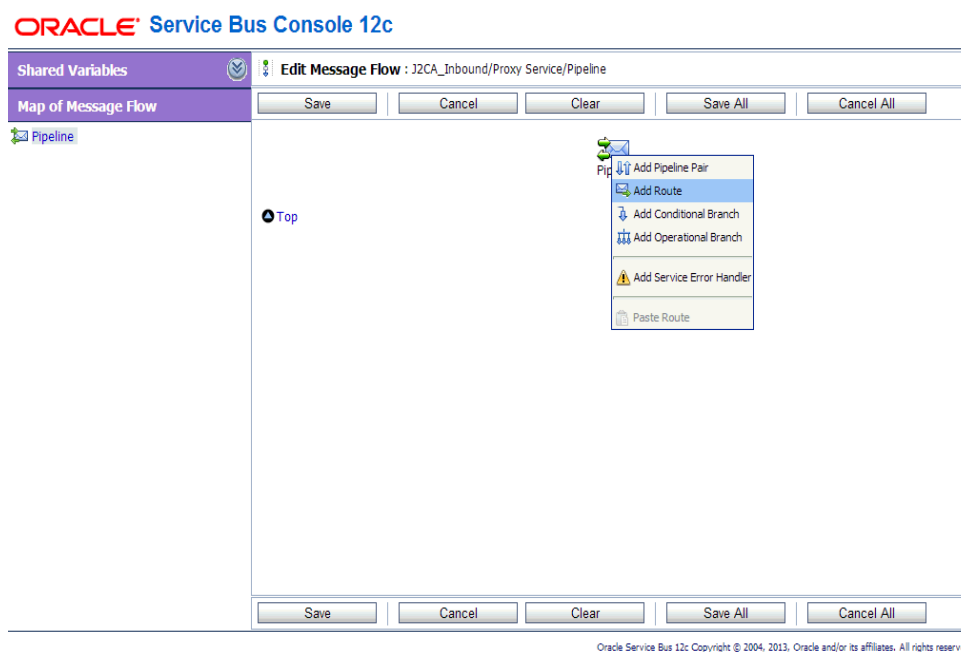
9. 図 7-62 に示すように、左ペインにある「プロキシ・サービス」フォルダの下で「パイプライン」をダブルクリックし、右ペインにある下向きのアイコンをクリックして、メッセージ・フローを開きます。

図 7-62 メッセージ・フロー



10. 図 7-63 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て、メニューから「ルートの追加」を選択します。

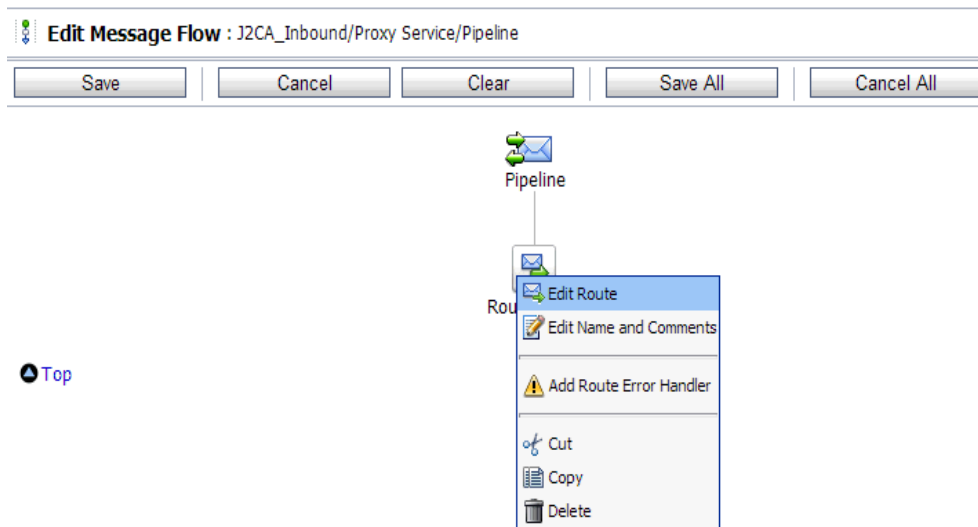
図 7-63 「ルートの追加」オプション



RouteNode1 アイコンが追加されました。

11. 図 7-64 に示すように、RouteNode1 アイコンをクリックして、メニューから「ルートの編集」を選択します。

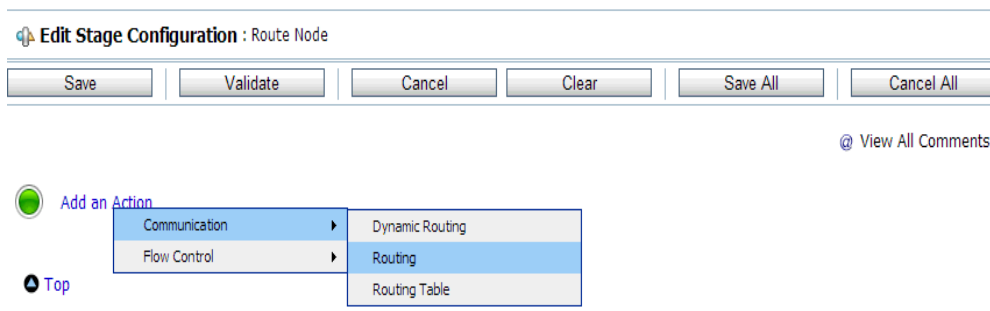
図 7-64 「ルート編集」オプション



「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。

12. 図 7-65 に示すように、「アクションの追加」をクリックして、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 7-65 「ステージ構成の編集」ワークスペース



13. 図 7-66 に示すように、<サービス>をクリックします。

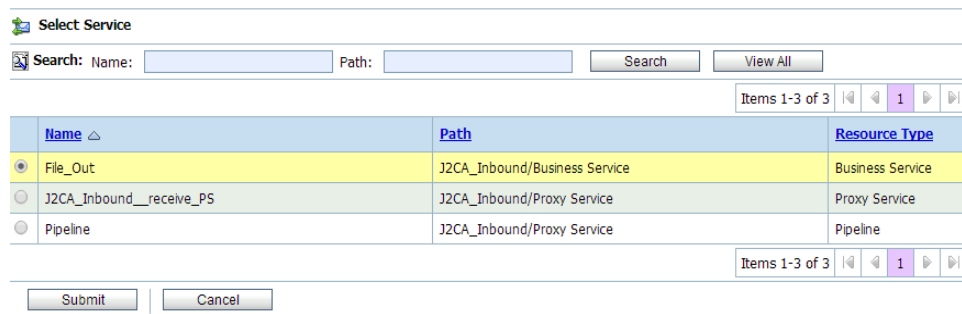
図 7-66 サービス・ルート・アクション



「サービスの選択」ダイアログが表示されます。

14. 図 7-67 に示すように、「File_Out」ビジネス・サービスを選択して、「発行」をクリックします。

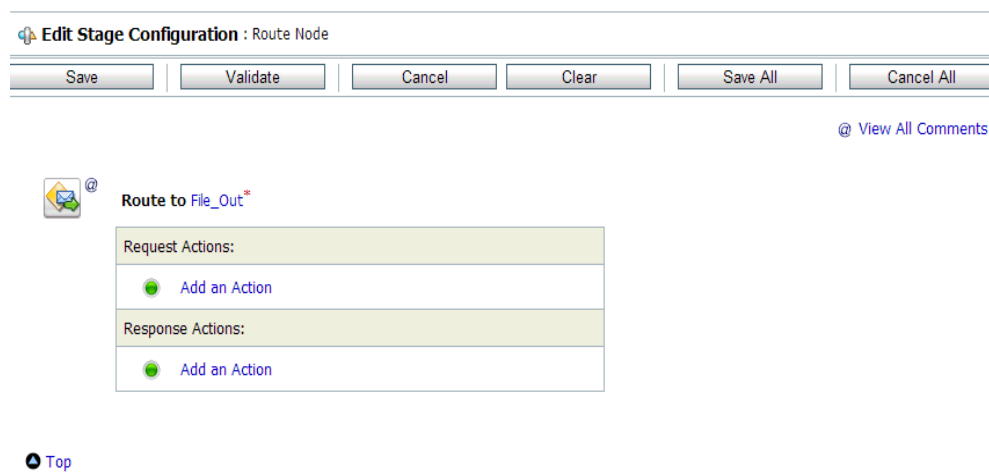
図 7-67 「サービスの選択」ダイアログ



「ステージ構成の編集」ワークスペース領域に戻ります。

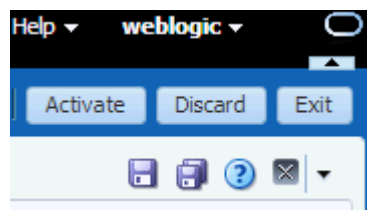
15. 図 7-68 に示すように、「すべて保存」をクリックします。

図 7-68 「ステージ構成の編集」ワークスペース領域



16. 図 7-69 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。

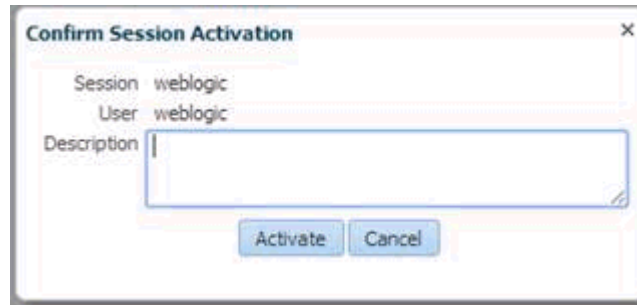
図 7-69 「アクティブ化」ボタン



セッションのアクティブ化の確認ウィンドウが表示されます。

17. 図 7-70 に示すように、「アクティブ化」をクリックして、変更を保存します。

図 7-70 セッションのアクティブ化の確認ウィンドウ



18. Siebel システムからのイベントをトリガーし、構成された出力場所で出力が受信されたかを確認します。

イベントのトリガーの詳細は、4-49 ページ 4.5.5 項「Siebel でのイベントのトリガー」を参照してください。

7.4 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 構成の場合の sbconsole を使用したアウトバウンド処理の構成方法について説明します。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB\BSE\Siebel_Sample_BSE_OSB_Outbound_Project
```

この項では、次のトピックについて説明します。

- 7.4.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」
- 7.4.2 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」
- 7.4.3 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」
- 7.4.4 項「ファイル・タイプ・ビジネス・サービスの構成」
- 7.4.5 項「WSDL ベースのビジネス・サービスの構成」
- 7.4.6 項「プロキシ・サービスでのパイプラインの構成」

7.4.1 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、7-2 ページ 7.2.1 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。

7.4.2 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

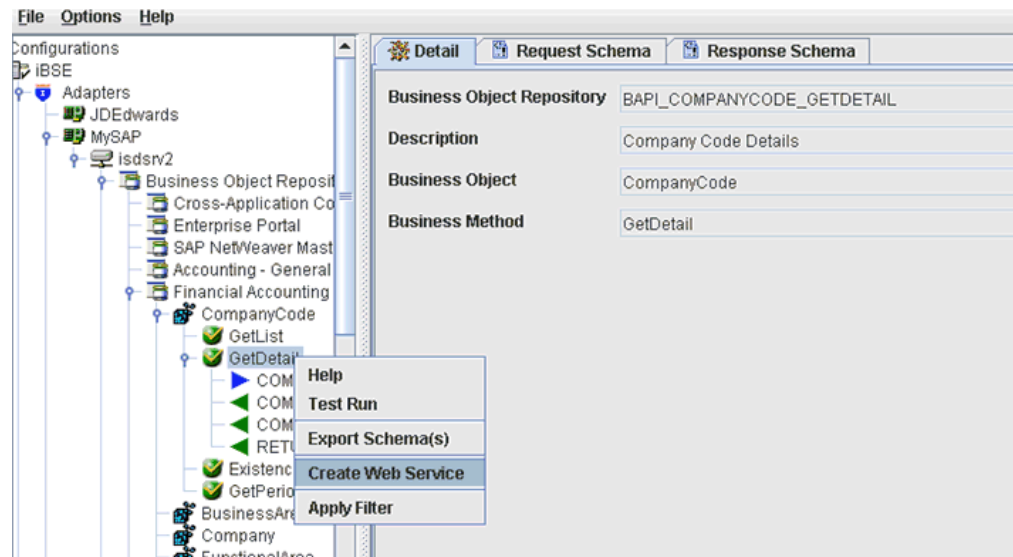
Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定の詳細は、7-6 ページ 7.2.2 項「Oracle Service Bus 統合のためのアプリケーション・エクスプローラのクラス・パスの設定」を参照してください。

7.4.3 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、Siebel ターゲットに接続します。

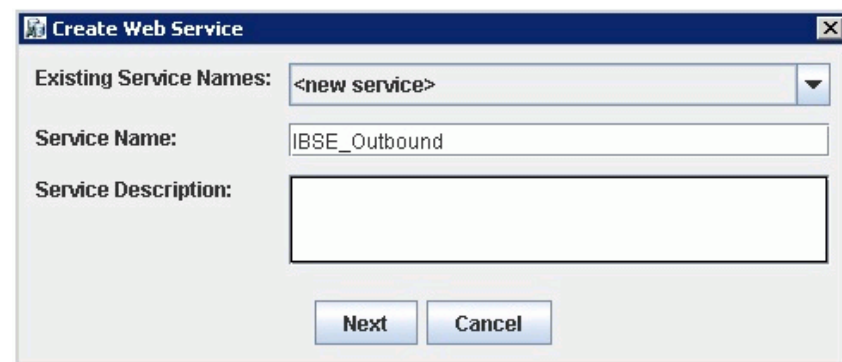
図 7-71 「Web サービスの作成」オプション



2. 「ビジネス・オブジェクト・リポジトリ」ノード、財務会計ノード、CompanyCode ビジネス・オブジェクトの順に展開します。
3. 図 7-71 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 7-72 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 7-72 「Web サービスの作成」ダイアログ



4. サービスの名前を入力し、「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 7-73 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 7-73 「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表示されます。デフォルトを受け入れるか、任意の名前を指定します。
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。
このロケーションは、Oracle Service Bus のプロジェクト名と、オプションで1つ以上のフォルダ名で構成されます。プロジェクト名および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。

7.4.4 ファイル・タイプ・ビジネス・サービスの構成

ファイル・タイプ・ビジネス・サービスの構成の詳細は、7-9 ページ 7.2.5 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。

7.4.5 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. 左ペインの「ビジネス・サービス」フォルダを右クリックして、「ビジネス・サービス」を選択します。

図 7-74 に示すように、「ビジネス・サービス」ウィンドウが表示されます。

図 7-74 「ビジネス・サービス」 ウィンドウ

Create Business Service

Create Type Transport

Create Service

* Resource Name: BSE_Outbound_BS
Description: []

Service Definition

WSDL Based Service
Name: []
Path: []
Port/Binding: []
 Transport: http

Back Next Create Cancel

- 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ます。

図 7-75 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 7-75 検索と選択 : WSDL リソース・ウィンドウ

Search and Select: WSDL Resource

Name: []
Path: []
Namespace: []

Search Reset

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

OK Cancel

- 「検索」ボタンをクリックして、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」ウィンドウに戻ります。

4. 「次へ」をクリックします。
5. 図 7-76 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 7-76 「ビジネス・サービス」ウィンドウ

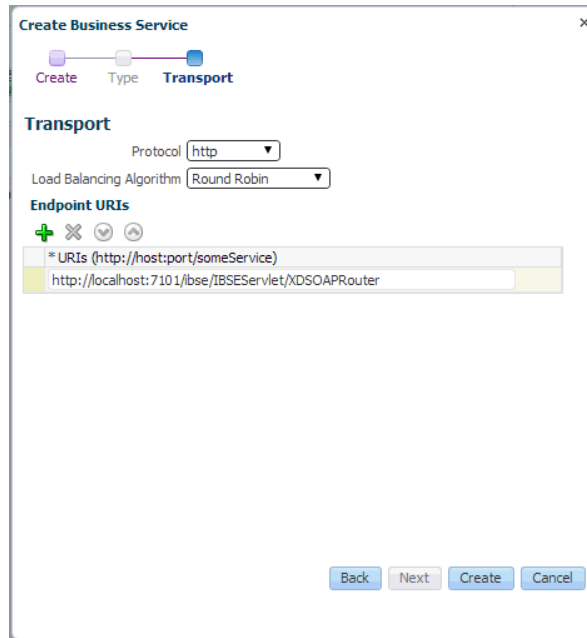
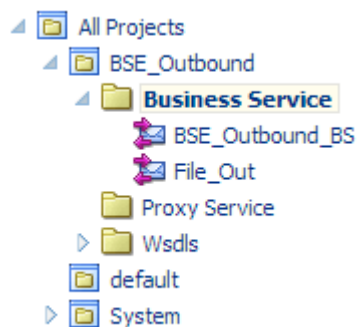


図 7-77 に示すように、作成した WSDL ベースのビジネス・サービスが「ビジネス・サービス」フォルダの下に表示されます。

図 7-77 WSDL ベースのビジネス・サービス

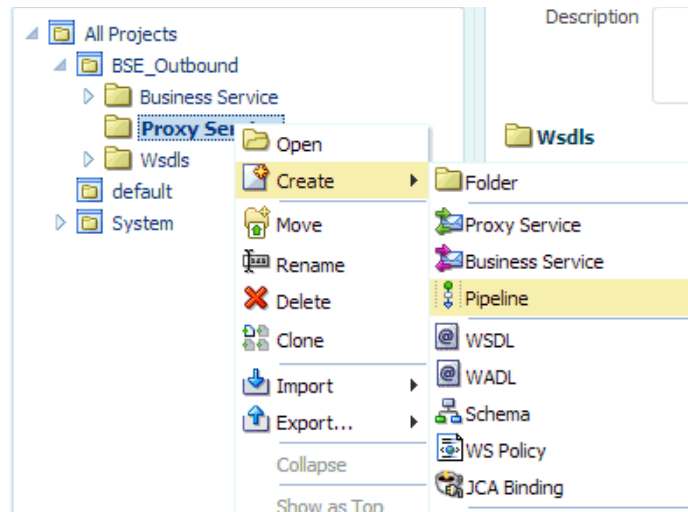


7.4.6 プロキシ・サービスでのパイプラインの構成

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

1. 図 7-78 に示すように、「プロキシ・サービス」フォルダを右クリックして、「作成」を選択し、「パイプライン」をクリックします。

図 7-78 「パイプライン」オプション



「パイプライン」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**が選択されます。図 7-79 に示すように、プロキシ・サービス名を変更する場合は、プロキシ・サービス名を変更して、「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 7-79 「パイプライン」ウィンドウ

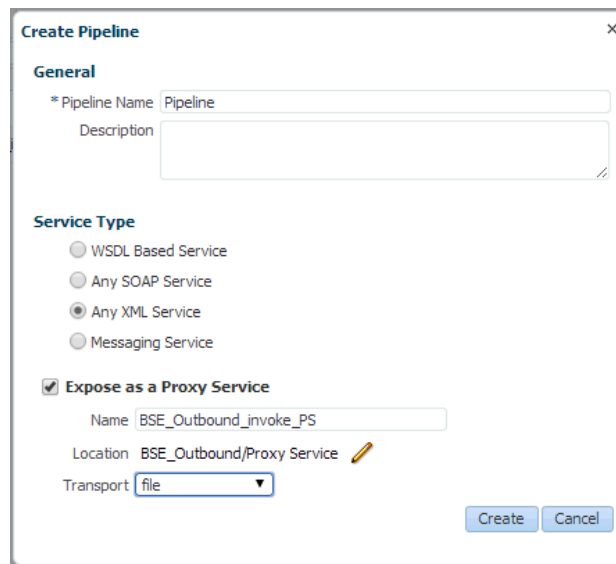
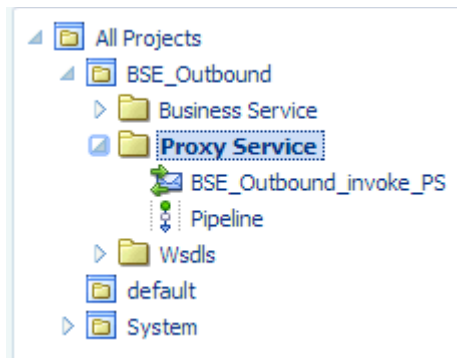


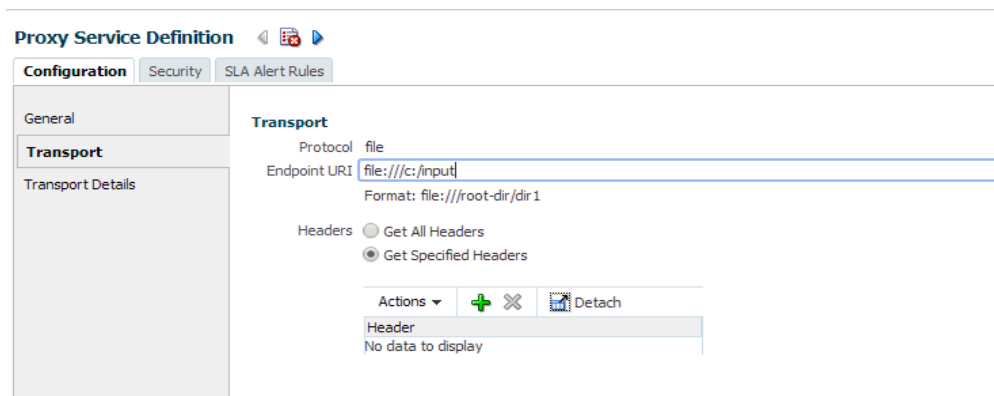
図 7-80 に示すように、作成したパイプラインとプロキシ・サービスが「プロキシ・サービス」の下に表示されます。

図 7-80 「パイプライン」ノード



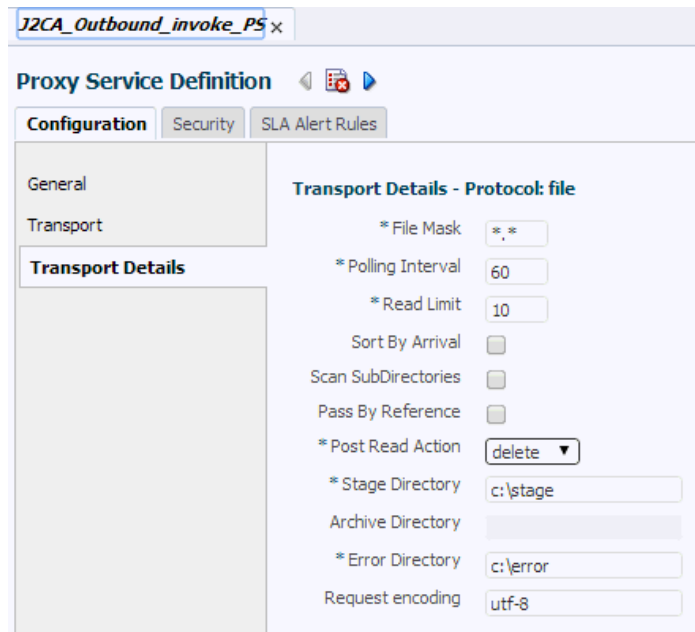
3. 作成したプロキシ・サービスをダブルクリックし、左ペインで「トランスポート」をクリックします。図 7-81 に示すように、「エンドポイント URI」フィールドに入力場所を入力します。

図 7-81 トランスポート



4. 図 7-82 に示すように、左ペインで「トランスポートの詳細」をクリックして、「ステージ・ディレクトリ」フィールドに場所を入力します。

図 7-82 トランスポートの詳細



5. 図 7-83 に示すように、右隅の「すべて保存」アイコンをクリックします。

図 7-83 「すべて保存」アイコン



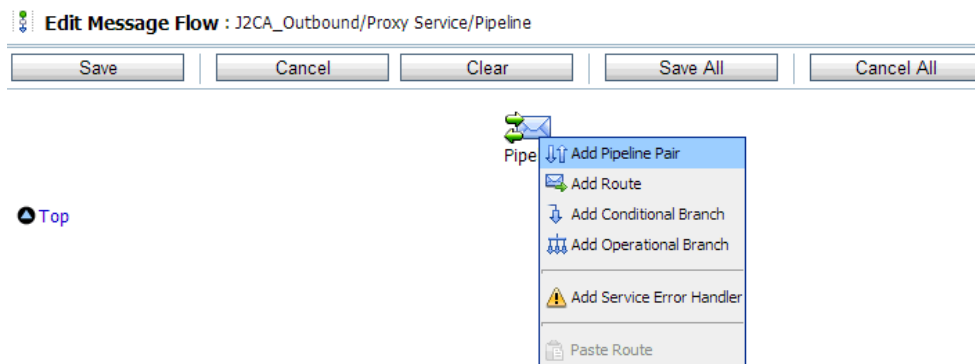
6. 図 7-84 に示すように、「パイプライン」ノードをダブルクリックして、右ペインの「メッセージ・フローを開く」をクリックし、メッセージ・フローを開きます。

図 7-84 「メッセージ・フロー」アイコン



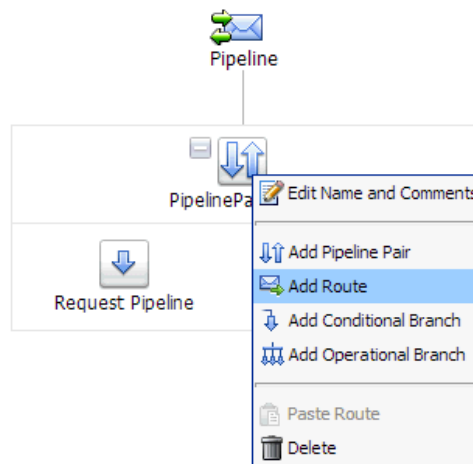
7. 図 7-85 に示すように、プロキシ・サービス・アイコンをクリックして、メニューから「パイプライン・ペアの追加」を選択します。

図 7-85 「パイプライン・ペアの追加」オプション



8. 図 7-86 に示すように、**PipelinePairNode1** アイコンをクリックして、メニューから「ルートの追加」を選択します。

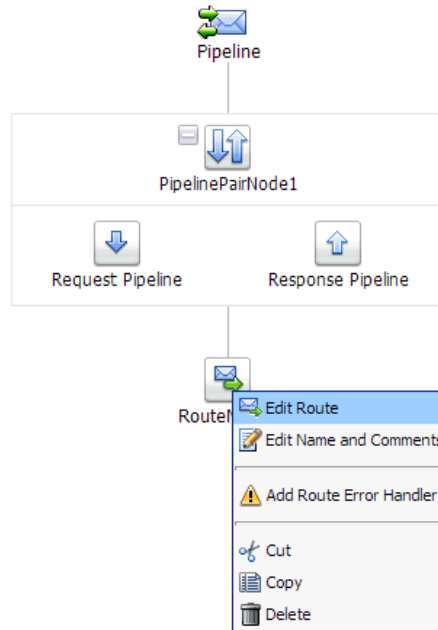
図 7-86 「ルートの追加」オプション



RouteNode1 アイコンの下に追加されます。

9. 図 7-87 に示すように、**RouteNode1** アイコンをクリックして、メニューから「ルートの編集」を選択します。

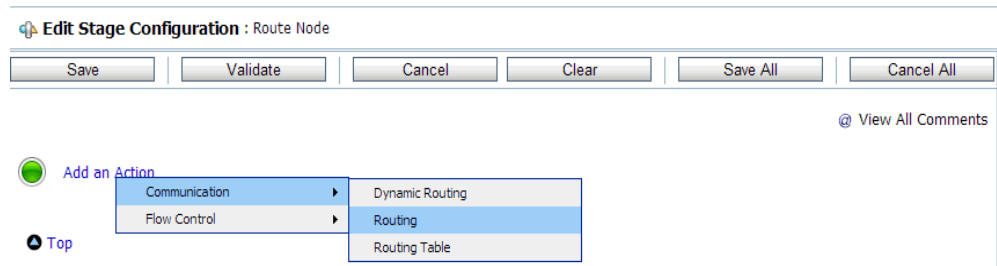
図 7-87 「ルート編集」オプション



「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 7-88 に示すように、「アクションの追加」をクリックして、「通信」を選択し、「ルーティング」をクリックします。

図 7-88 「ステージ構成の編集」ワークスペース領域



11. 図 7-89 に示すように、<サービス>をクリックします。

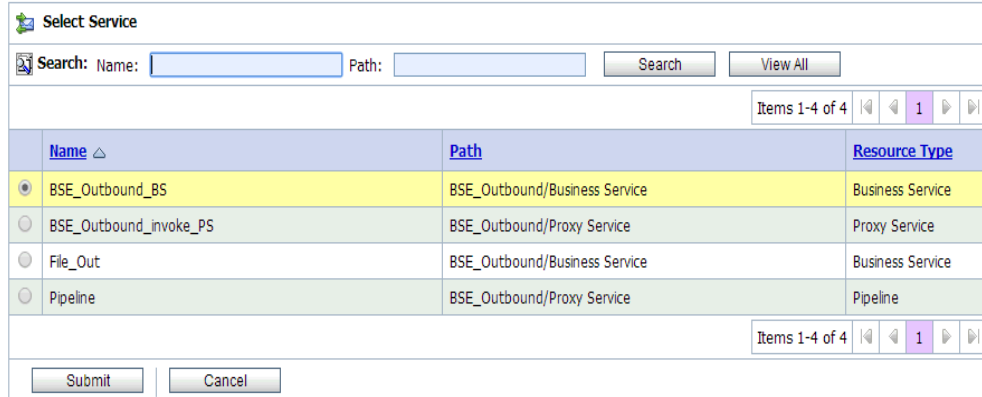
図 7-89 アクション



「サービスの選択」ダイアログが表示されます。

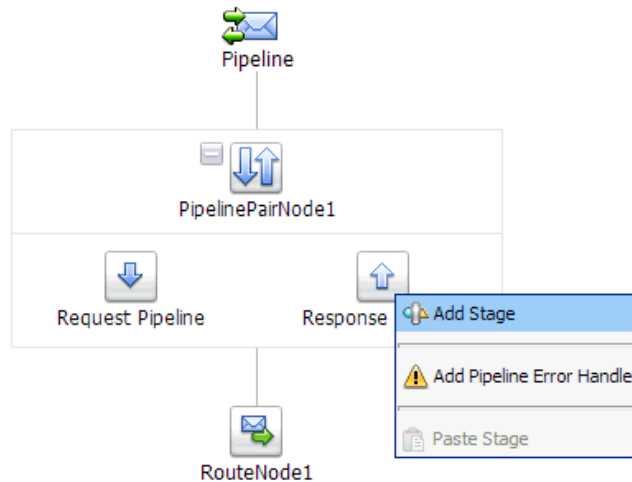
12. 図 7-90 に示すように、Siebel 用に構成されている WSDL タイプのビジネス・サービスを選択し、「発行」をクリックします。

図 7-90 「サービスの選択」ダイアログ



13. リストから操作属性として Siebel ビジネス・オブジェクトの名前 (queryWithView など) を選択し、「保存」をクリックします。
14. 図 7-91 に示すように、「レスポンス・パイプライン」アイコンをクリックして、メニューから「ステージの追加」を選択します。

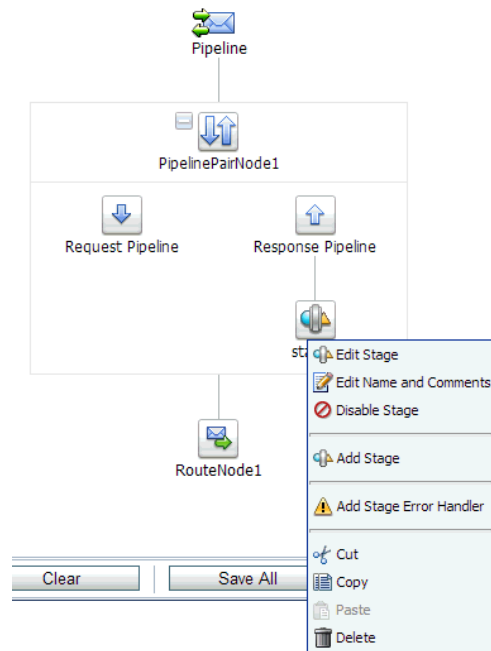
図 7-91 「レスポンス・パイプライン」アイコン



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 7-92 に示すように、「ステージ 1」アイコンをクリックして、メニューから「ステージの編集」を選択します。

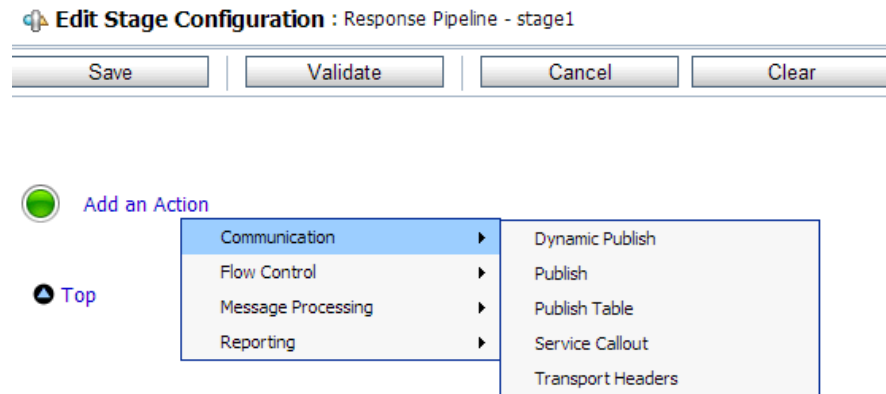
図 7-92 「ステージの編集」オプション



「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。

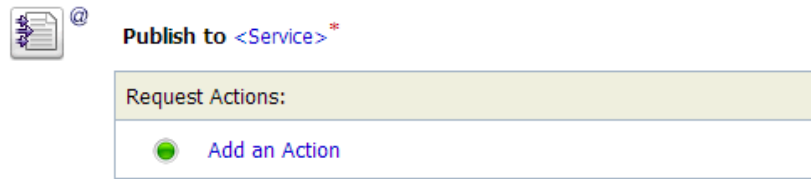
16. 図 7-93 に示すように、「アクションの追加」をクリックして、「通信」を選択し、「パブリッシュ」をクリックします。

図 7-93 「ステージ構成の編集」ワークスペース領域



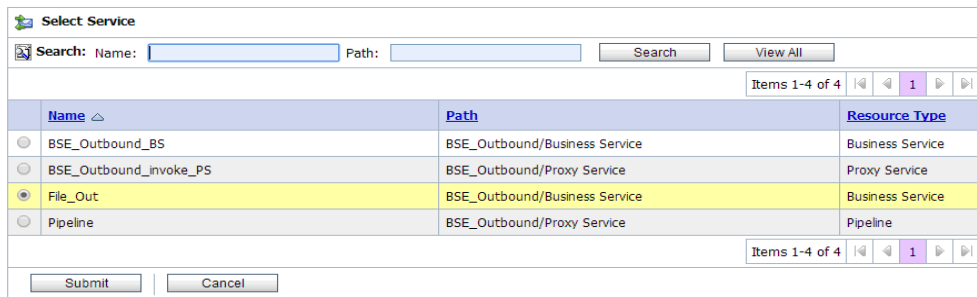
17. 図 7-94 に示すように、<サービス>をクリックします。

図 7-94 < サービス > アクション



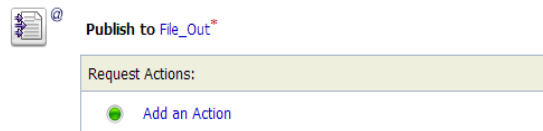
18. 図 7-95 に示すように、「サービスの選択」ダイアログで、ファイル・タイプ・ビジネス・サービスを選択して、「発行」をクリックします。

図 7-95 「サービスの選択」ダイアログ



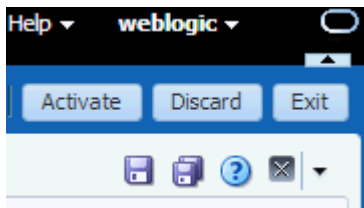
19. 図 7-96 に示すように、「すべて保存」をクリックします。

図 7-96 「すべて保存」ボタン



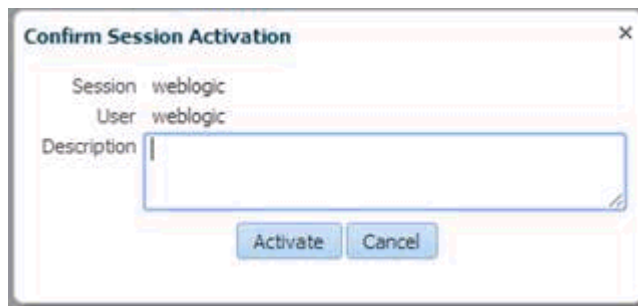
20. 図 7-97 に示すように、Oracle Service Bus セッションの右ペインで「アクティブ化」をクリックします。

図 7-97 「アクティブ化」ボタン



21. 図 7-98 に示すように、「アクティブ化」をクリックして、変更を保存します。

図 7-98 セッションのアクティブ化の確認



22. 入力 XML ファイルをコピーし、構成した入力フォルダ (C:\input など) に貼り付けます。

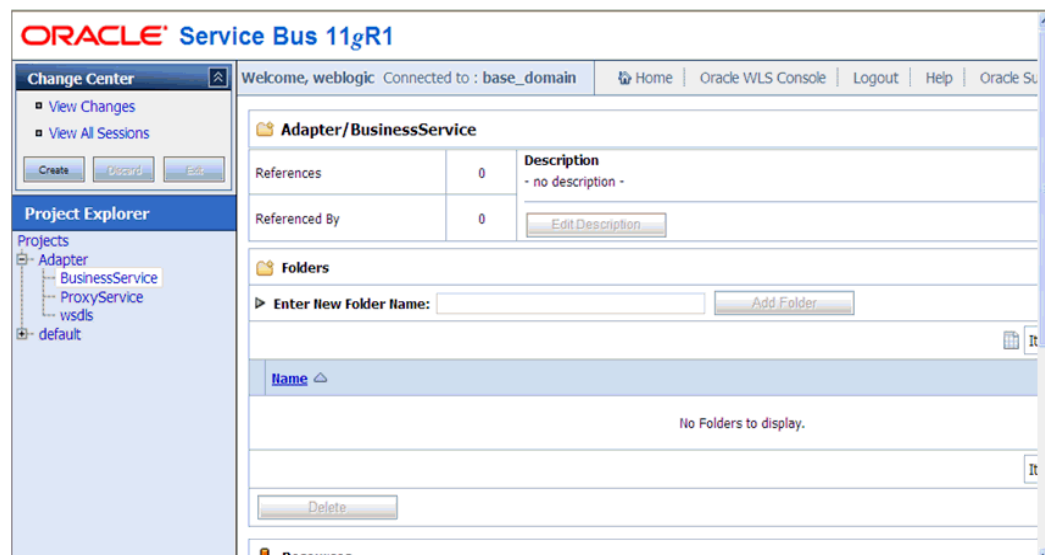
出力が構成済の出力ロケーション (C:\output など) で受信されます。

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

この項では、J2CA 構成の場合の Oracle Service Bus を使用した JMS プロキシ・サービスの構成方法について説明します。

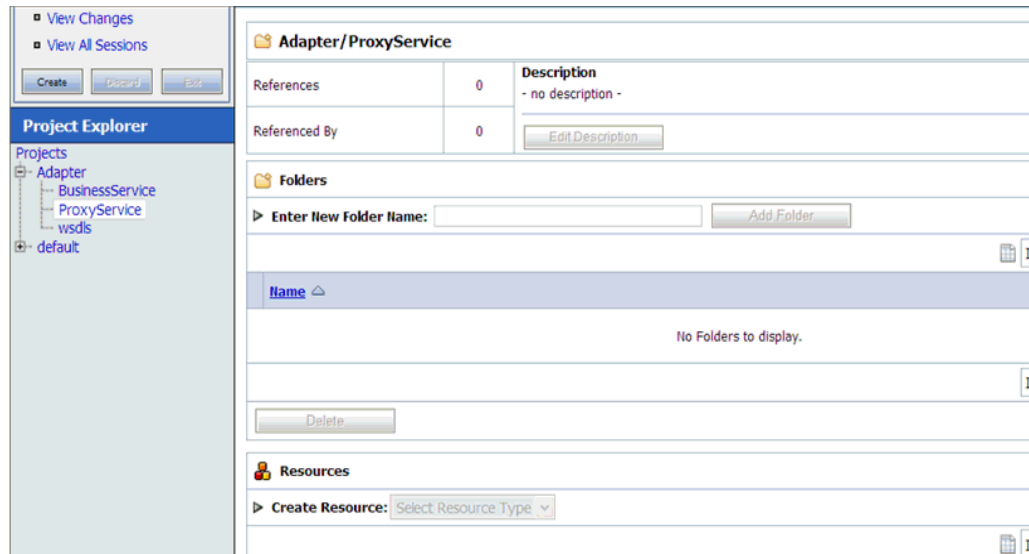
1. Oracle Service Bus を起動し、必要なプロジェクト・フォルダを作成します。
詳細は、7.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」を参照してください。
2. WSDL を生成して、アプリケーション・エクスプローラから作成済みのプロジェクト・フォルダに公開します。公開した WSDL を使用して、ビジネス・サービスを作成します。
詳細は、7.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL の公開」を参照してください。
3. 図 7-99 に示すように、Service Bus コンソール・ページを表示します。

図 7-99 Service Bus コンソール



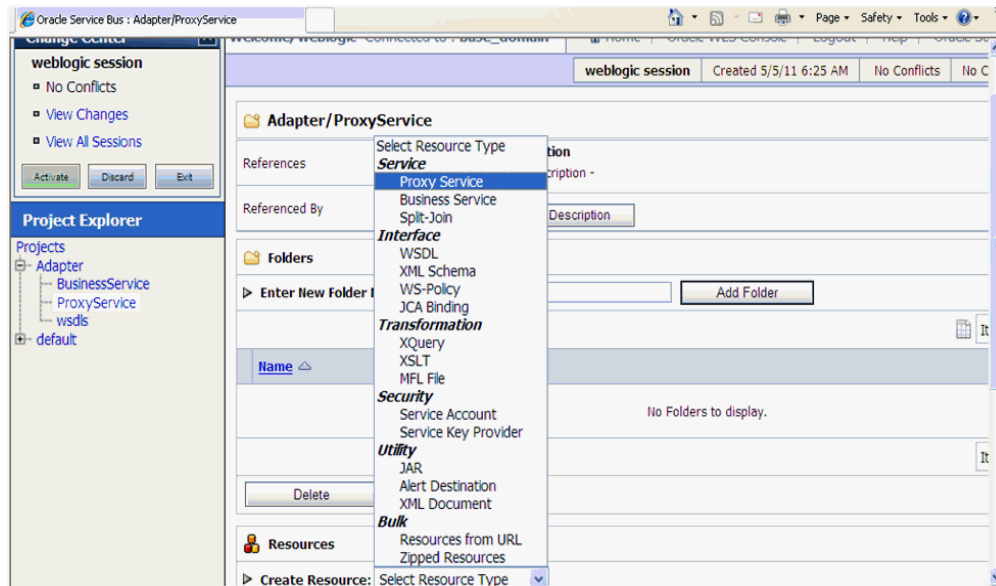
4. 図 7-100 に示すように、左ペインで ProxyService プロジェクト・フォルダを選択して、「作成」をクリックします。

図 7-100 プロキシ・サービス



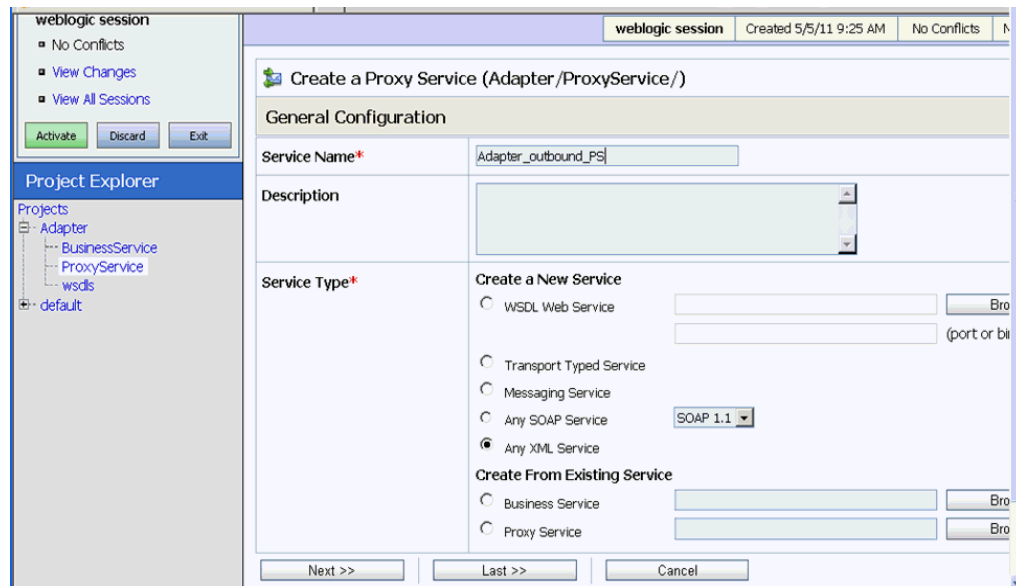
5. 図 7-101 に示すように、右ペインで「リソースの作成」リストから「プロキシ・サービス」を選択します。

図 7-101 「リソースの作成」メニュー



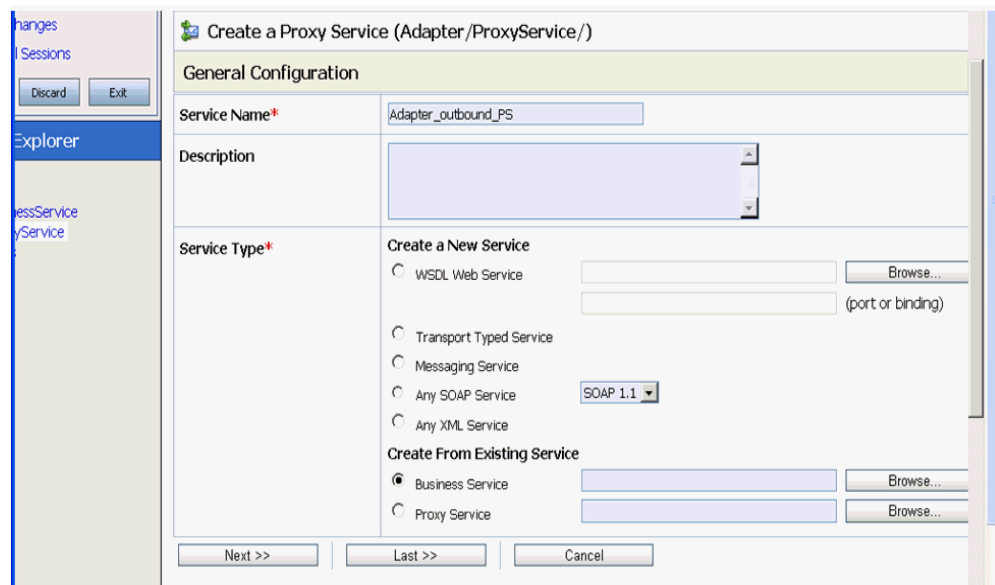
6. 図 7-102 に示すように、「サービス名」フィールドに適切な名前を入力します。

図 7-102 サービス名



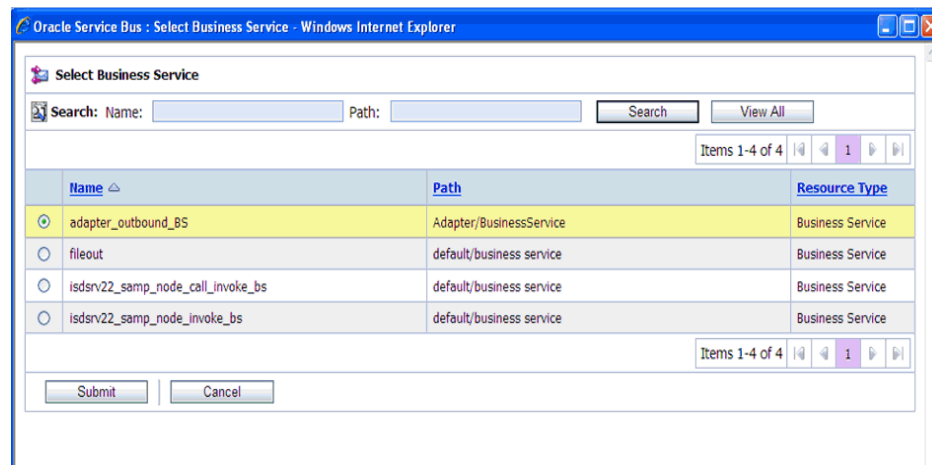
7. 図 7-103 に示すように、「サービス・タイプ」セクションの「既存のサービスから作成」の下にある「ビジネス・サービス」ラジオ・ボタンを選択し、「参照」をクリックします。

図 7-103 ビジネス・サービス



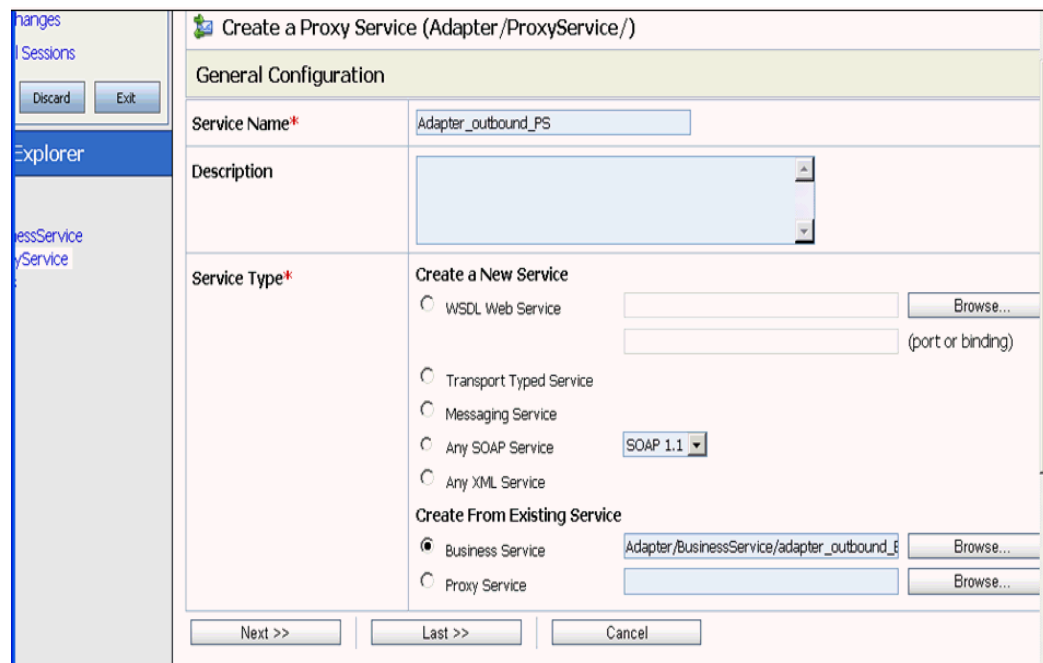
8. 図 7-104 に示すように、既存のビジネス・サービスを選択して、「発行」をクリックします。

図 7-104 既存のビジネス・サービス



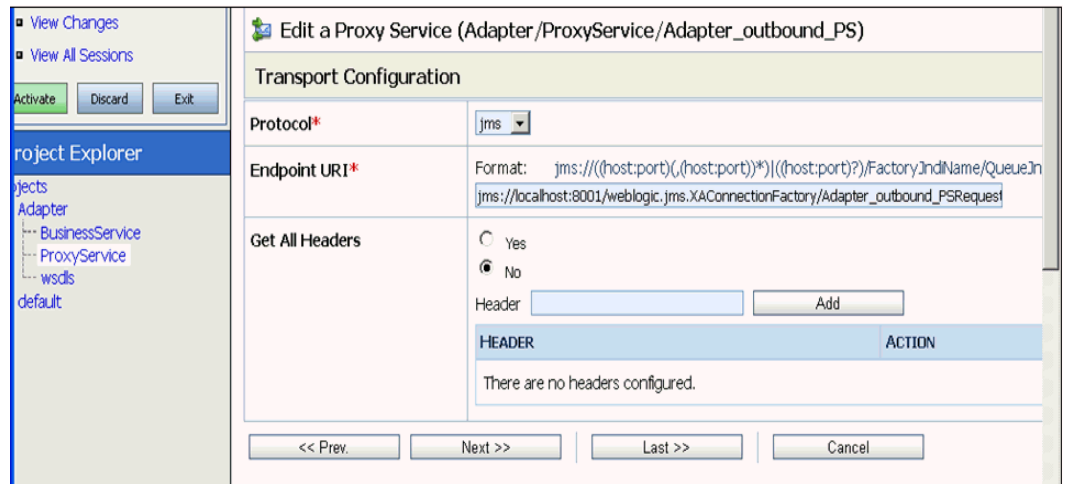
9. 図 7-105 に示すように、「次へ」をクリックします。

図 7-105 次



10. 図 7-106 に示すように、「プロトコル」リストから「jms」を選択して、「次へ」をクリックします。

図 7-106 「プロトコル」 リスト



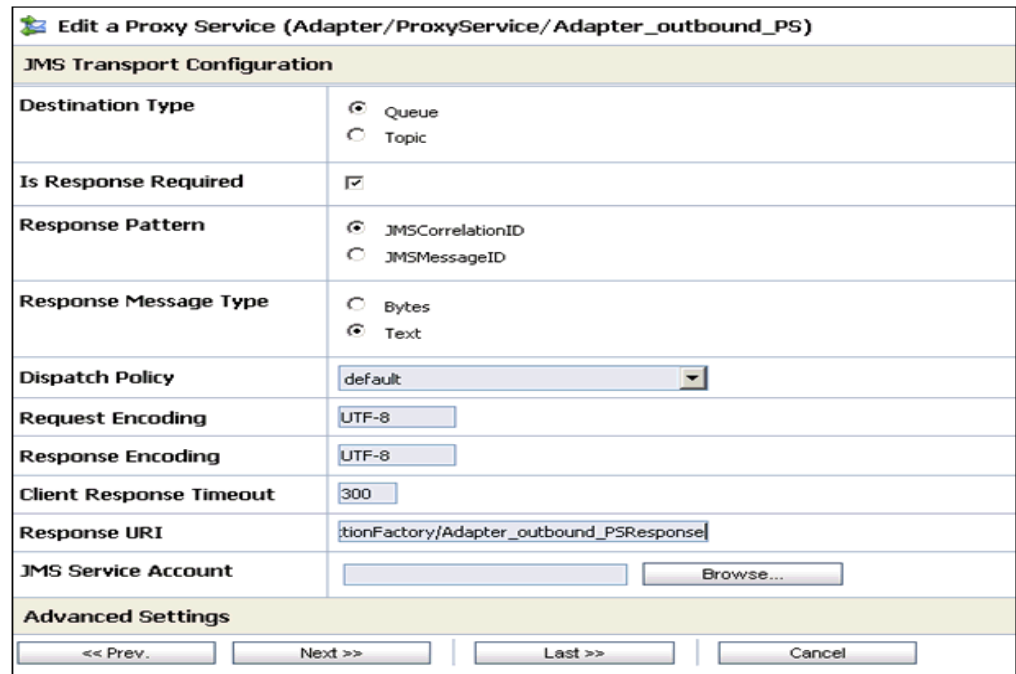
11. 図 7-107 に示すように、次のパラメータを入力します。

- 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・ボックスを選択します。
- 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するエンドポイント URI を入力し、Request を Response に変更します。

次に例を示します。

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

図 7-107 プロキシ・サービスの編集



12. 「次へ」をクリックします。

図 7-108 に示すように、「操作選択構成」 ペインが表示されます。

図 7-108 「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 7-109 に示すように、「トランザクションが必要」ボックスを選択して、「次へ」をクリックします。

図 7-109 メッセージ処理

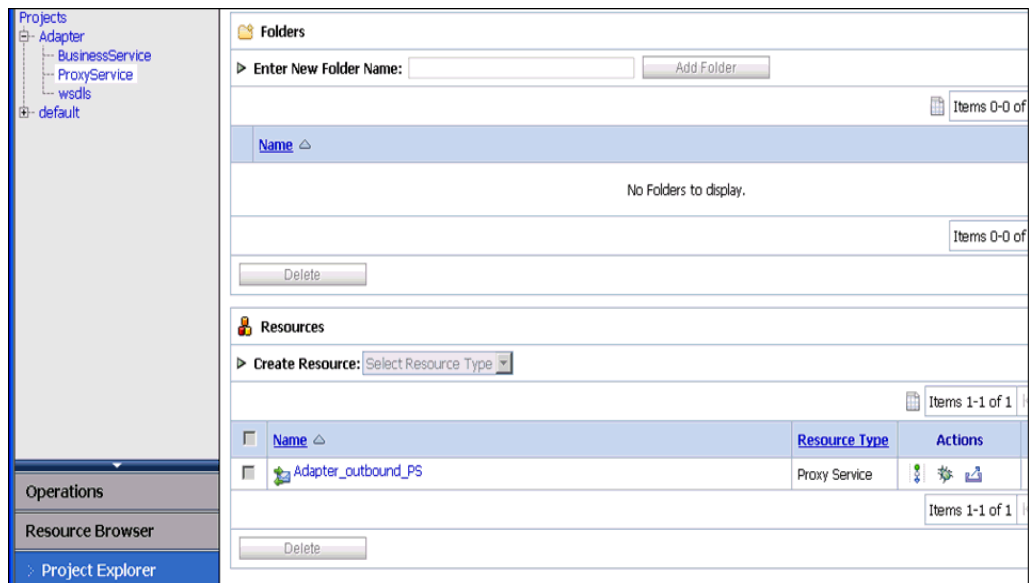
15. 図 7-110 に示すように、「保存」をクリックします。

図 7-110 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
<input data-bbox="462 760 602 783" type="button" value=" << Prev. "/> <input data-bbox="651 760 790 783" type="button" value=" Save "/> <input data-bbox="839 760 979 783" type="button" value=" Cancel "/>	

図 7-111 に示すように、作成したプロキシ・サービスが保存されました。

図 7-111 プロキシ・サービス

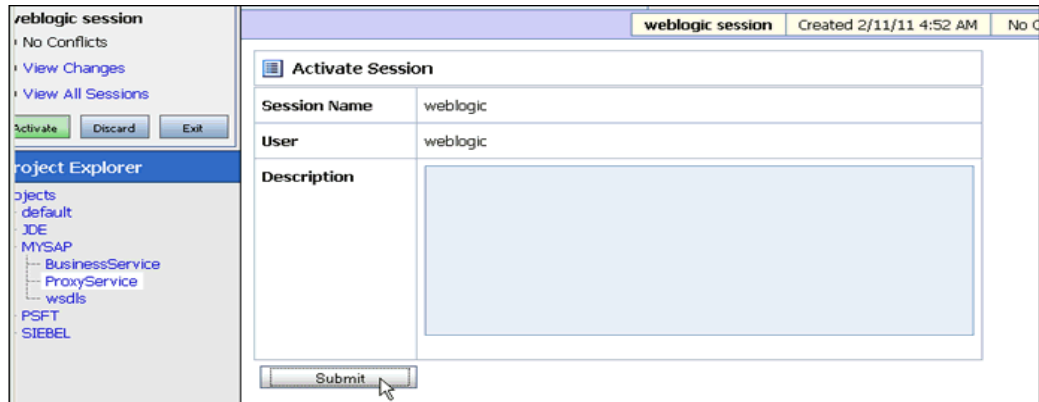


The screenshot shows the Oracle Service Bus console interface. On the left, the 'Project Explorer' pane displays a tree structure under 'Adapter' including 'BusinessService', 'ProxyService', 'wsdls', and 'default'. The main workspace is divided into 'Folders' and 'Resources' sections. The 'Resources' section shows a 'Create Resource' dropdown set to 'Select Resource Type'. Below this, a table lists the resources:

Name	Resource Type	Actions
Adapter_outbound_PS	Proxy Service	[Refresh] [Refresh] [Refresh]

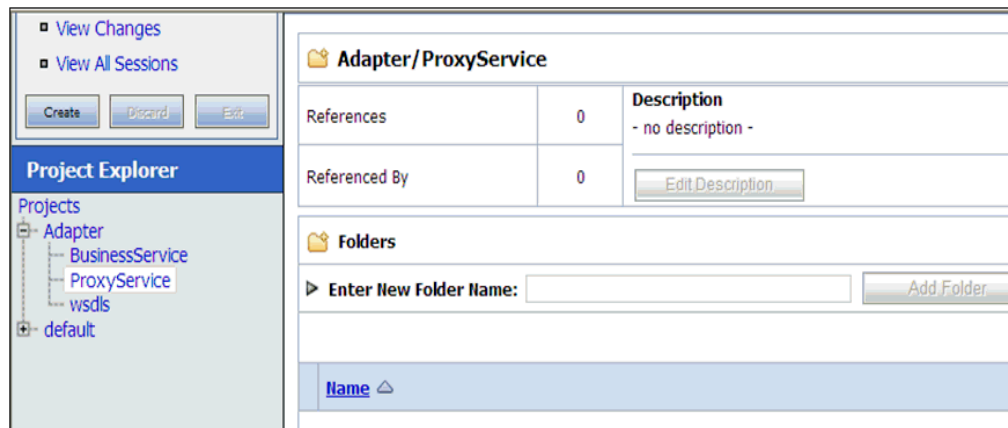
16. 図 7-112 に示すように、左ペインで「アクティブ化」をクリックしてから、「発行」をクリックします。

図 7-112 セッションのアクティブ化



17. 図 7-113 に示すように、左ペインの「プロジェクト」フォルダの下にある **ProxyService** をクリックします。

図 7-113 アダプタ



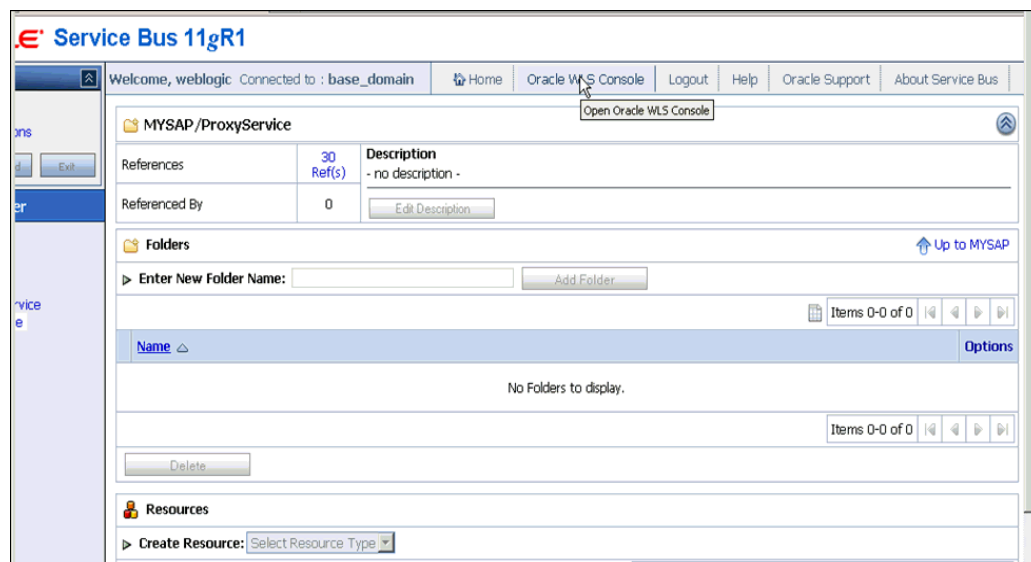
18. 図 7-114 に示すように、作成したプロキシ・サービスのために「テスト・コンソールの起動」アイコンをクリックします。

図 7-114 「テスト・コンソールの起動」アイコン



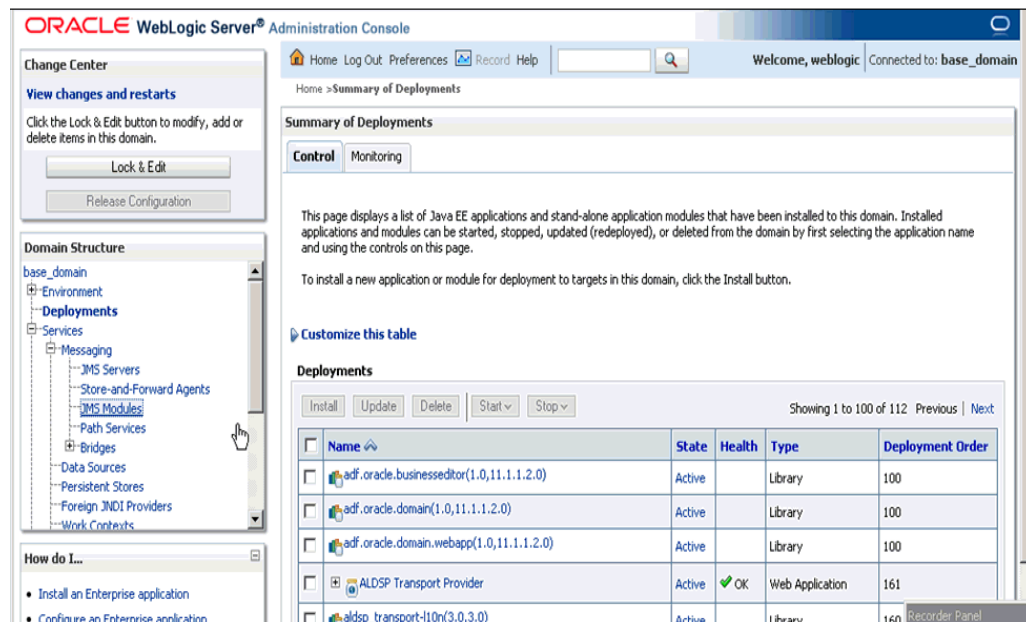
19. 「ペイロード」に入力値を入力して、「直接呼出し」ボックスの選択を解除し、「実行」をクリックします。
20. レスポンス・ドキュメントを確認してから、「閉じる」をクリックします。
21. 図 7-115 に示すように、「Oracle WLS コンソール」タブをクリックします。

図 7-115 ProxyService



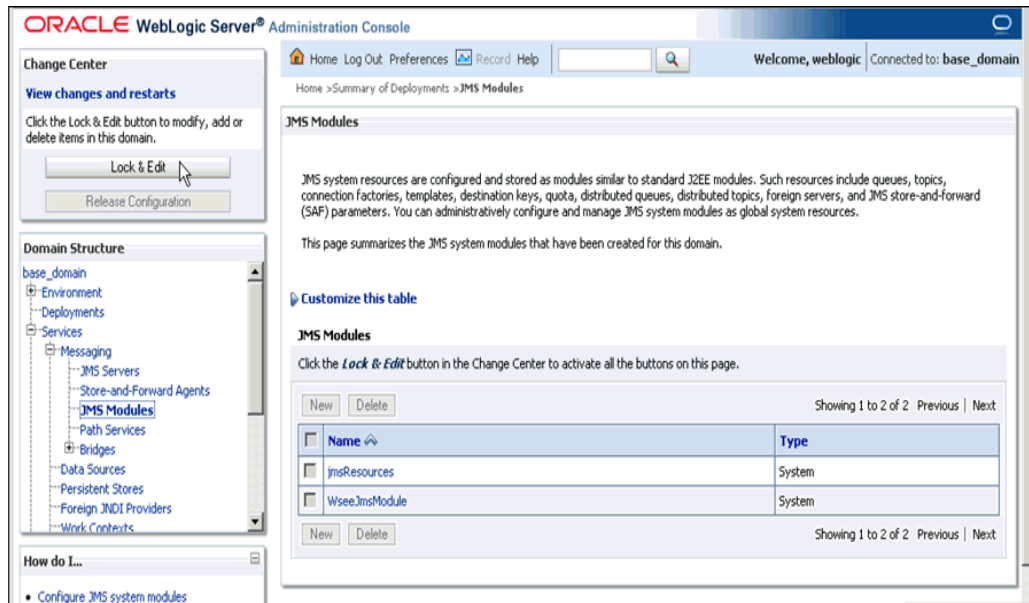
22. 図 7-116 に示すように、Oracle WLS コンソールで、「サービス」、「メッセージ機能」の順に展開し、「JMS モジュール」をクリックします。

図 7-116 Oracle WLS コンソール



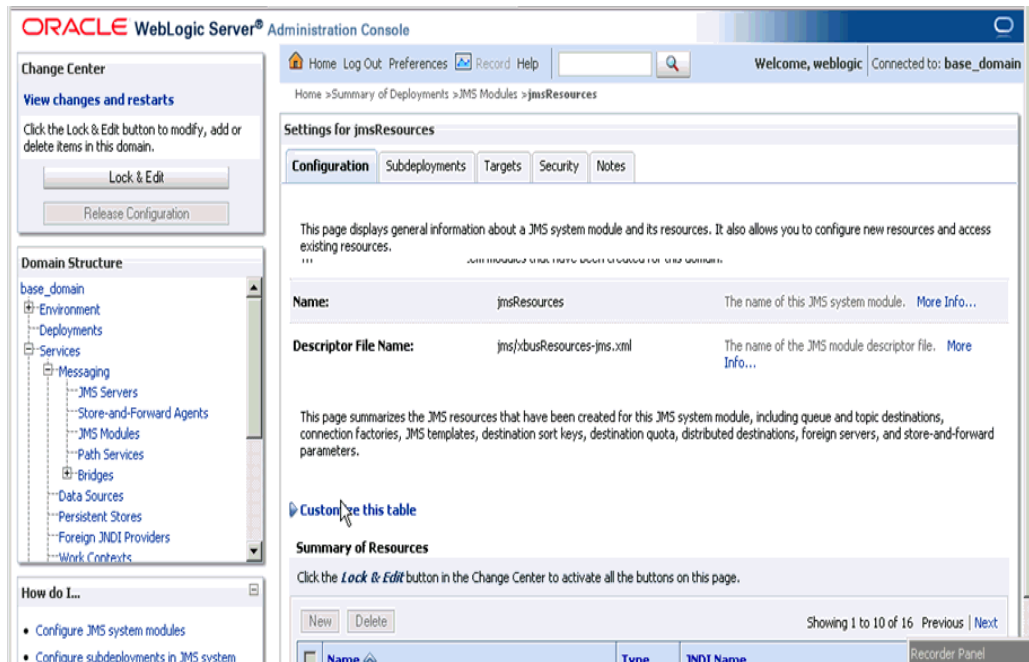
23. 図 7-117 に示すように、jmsResources をクリックします。

図 7-117 JMS モジュール



24. 図 7-118 に示すように、「ロックして編集」をクリックします。

図 7-118 構成設定



25. 図 7-119 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 7-119 Adapter_outbound_PSRequest

Name	Type	JNDI Name
Adapter_outbound_PSRequest-1531625329	Queue	Adapter_outbound_PSRequest
Adapter_outbound_PSResponse2015668417	Queue	Adapter_outbound_PSResponse
QueueIn	Queue	QueueIn
QueueIn.Quota	Quota	N/A
TemporaryTplmt	Template	N/A
weblogic.wsb.jms.transporttask.QueueConnectionFactory	Connection Factory	weblogic.wsb.jms.transporttask.QueueG
wli.reporting.jmsprovider.ConnectionFactory	Connection Factory	wli.reporting.jmsprovider.ConnectionFact
wli.reporting.jmsprovider.queue	Queue	wli.reporting.jmsprovider.queue
wli.reporting.jmsprovider_error.queue	Queue	wli.reporting.jmsprovider_error.queue
wli.reporting.purge.queue	Queue	wli.reporting.purge.queue

26. 図 7-120 に示すように、モニタリング・タブをクリックします。

図 7-120 モニタリング・タブ

ORACLE WebLogic Server® Administration Console

Home Log Out Preferences Record Help

Welcome, weblogic Connected to: base_domain

Home > jmsResources > Adapter_outbound_PSRequest-1531625329 > JMS
 Modules > jmsResources > placeholder > Adapter_outbound_PSRequest-1531625329 > JMS
 Modules > jmsResources > Adapter_outbound_PSRequest-1531625329

Settings for Adapter_outbound_PSRequest-1531625329

Configuration Monitoring Control Security Subdeployment Notes

General Thresholds Monitoring-Tab Overrides Logging Delivery Failure

Save

Use this page to define the general configuration parameters for this queue, such as selecting a destination key for sorting messages as they arrive on the queue.

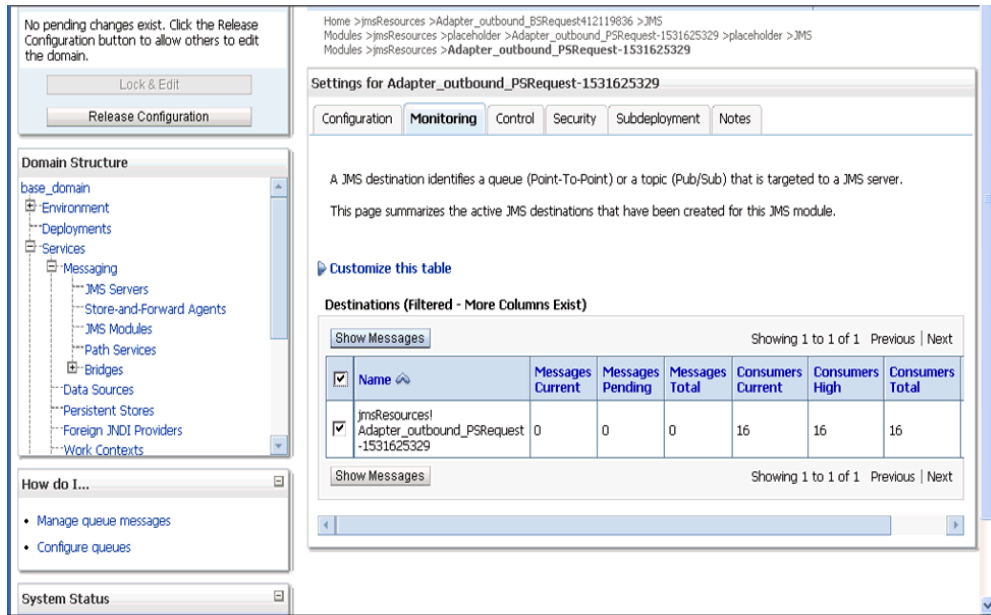
Name: Adapter_outbound_PSRequest-1531625329 The name of this JMS queue. More Info...

JNDI Name: Adapter_outbound_PSRequest The global JNDI name used to look up the destination within the JNDI namespace. More Info...

Template: None The JMS template from which the destination is derived. A template provides an efficient means of defining multiple

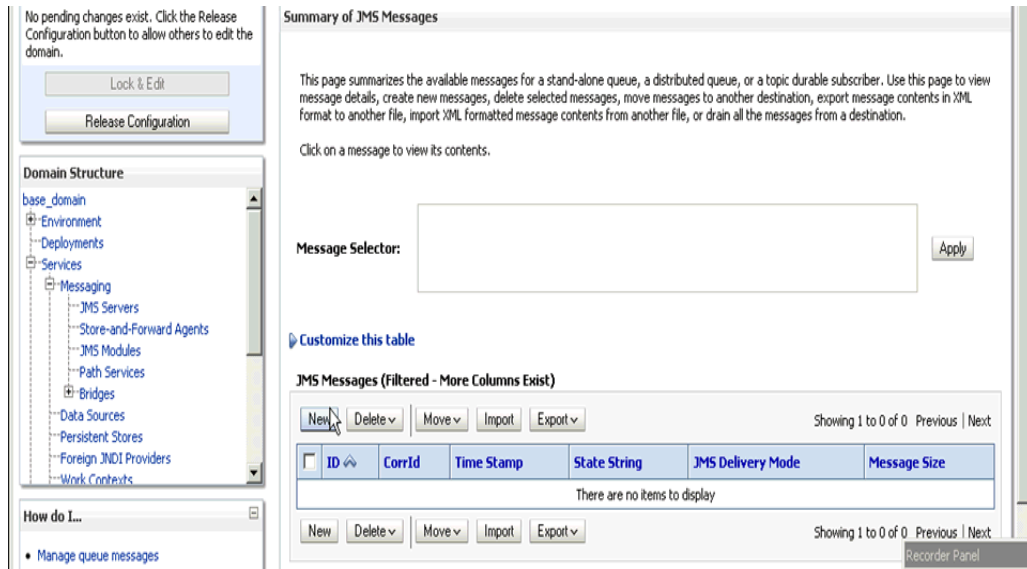
27. 図 7-121 に示すように、チェック・ボックスを選択して、「メッセージの表示」をクリックします。

図 7-121 アダプタの設定



28. 図 7-122 に示すように、「新規」をクリックします。

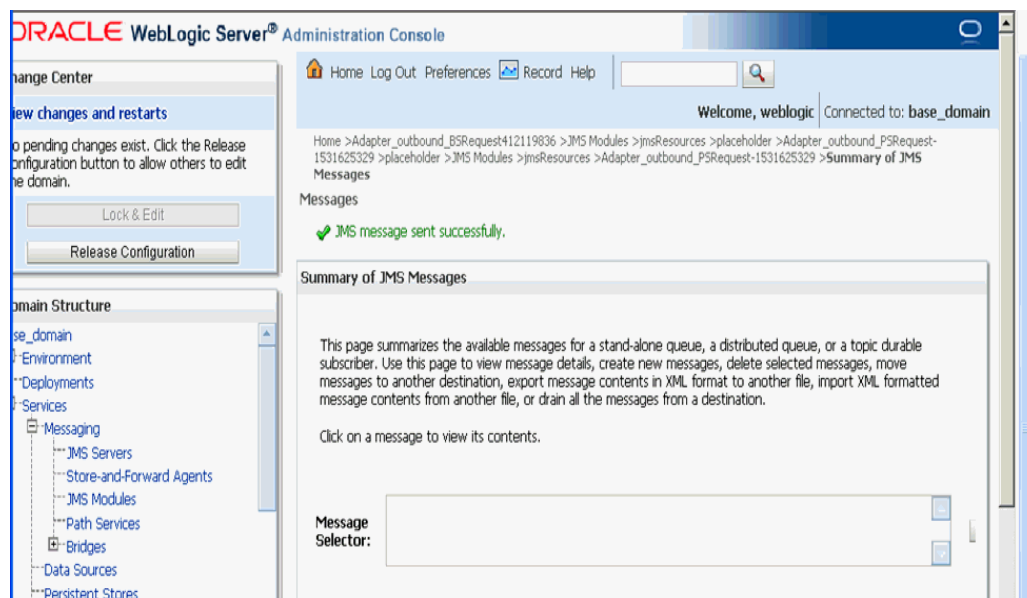
図 7-122 JMS メッセージ



29. 本文フィールドに入力ペイロードを入力して、「OK」をクリックします。

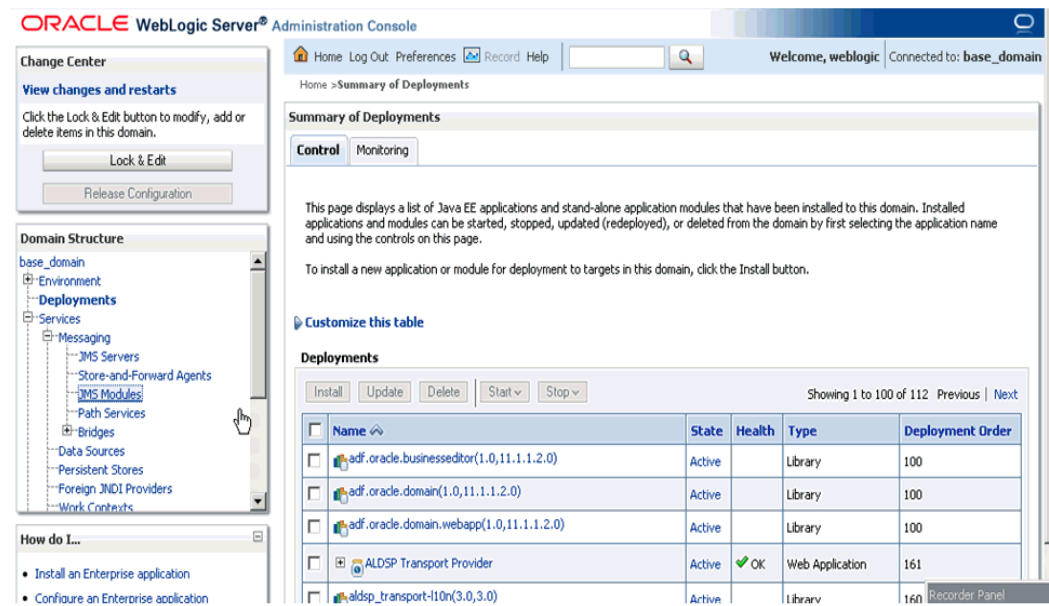
図 7-123 に示すように、成功を伝えるメッセージが表示されます。

図 7-123 JMS 成功メッセージ



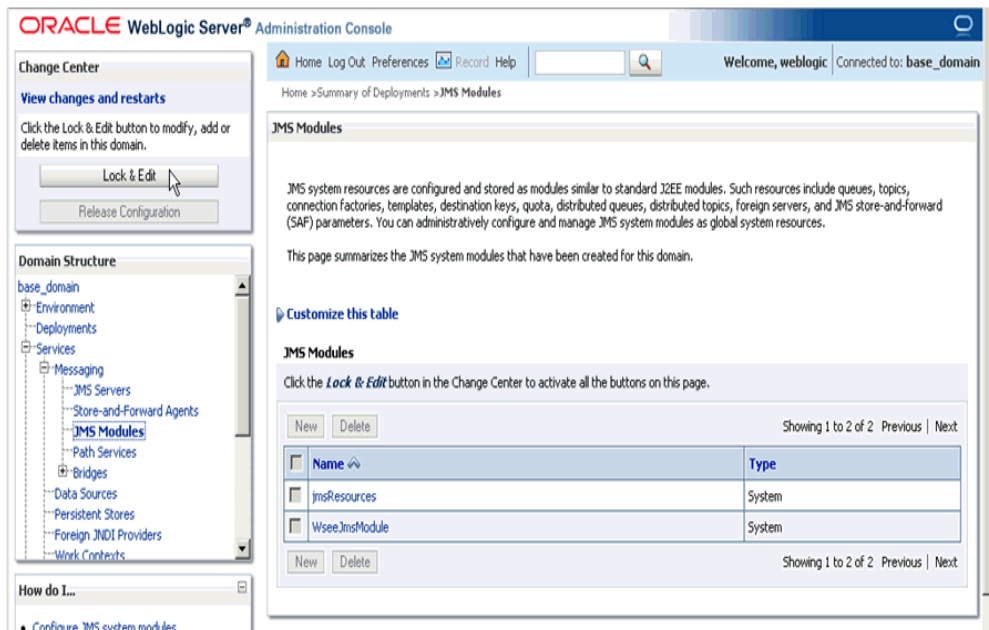
30. 図 7-124 に示すように、Oracle WLS コンソールで、「サービス」、「メッセージ機能」の順に展開し、「JMS モジュール」をクリックします。

図 7-124 JMS モジュール



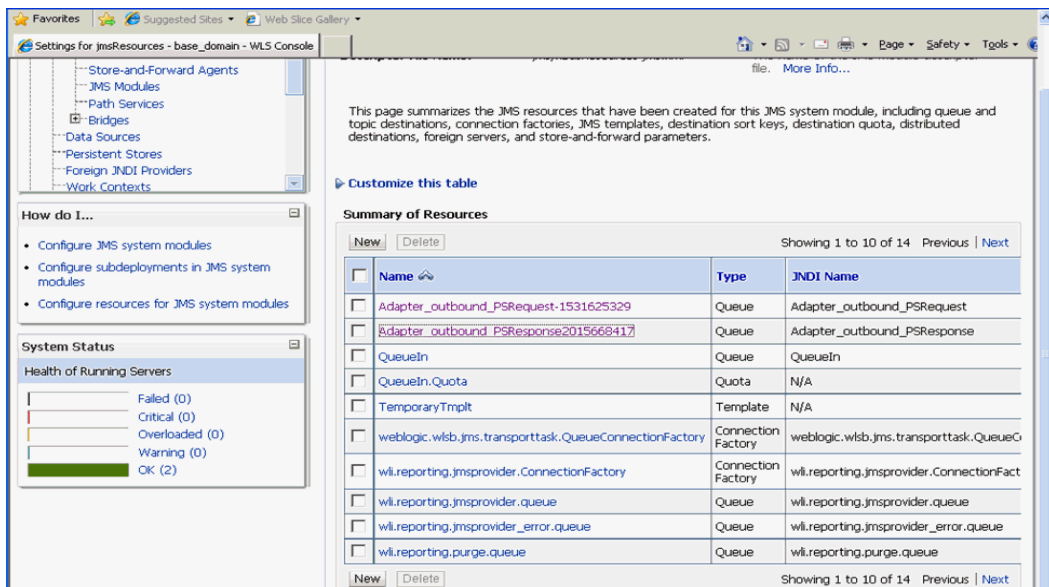
31. 図 7-125 に示すように、**jmsResources** をクリックします。

図 7-125 jmsResources



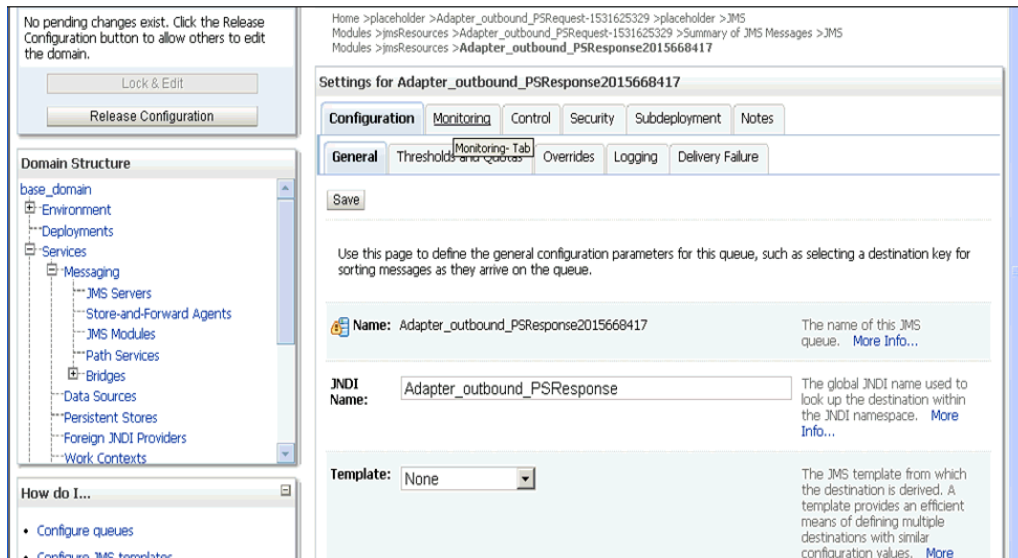
32. 図 7-126 に示すように、適切なレスポンス・リンク (Adapter_outbound_PSResponse など) をクリックします。

図 7-126 リソース



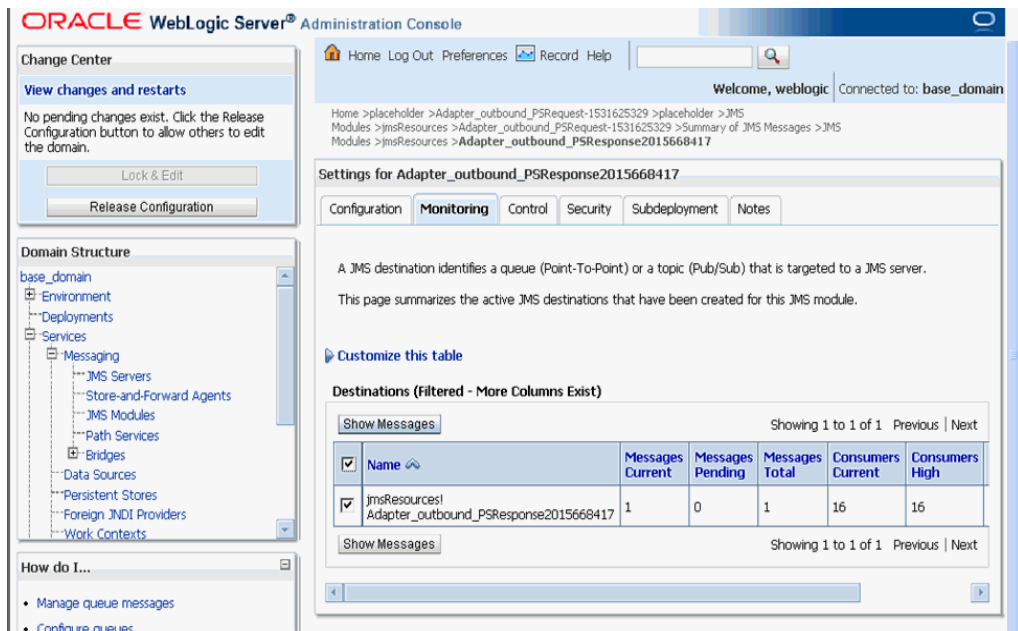
33. 図 7-127 に示すように、モニタリング・タブをクリックします。

図 7-127 モニタリング・タブ



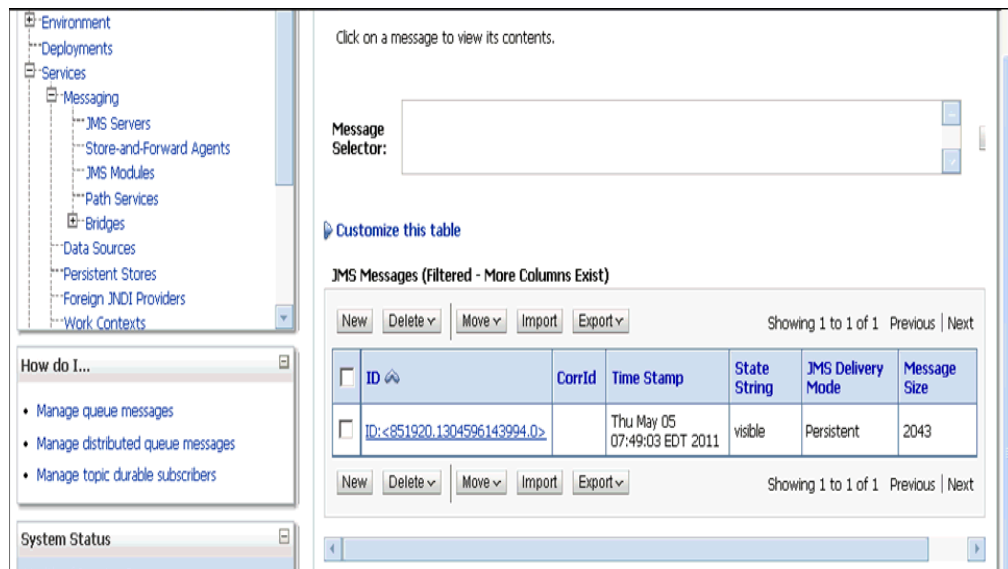
34. 図 7-128 に示すように、チェック・ボックスを選択して、「メッセージの表示」をクリックします。

図 7-128 宛先メッセージ



35. 図 7-129 に示すように、「ID」リンクをクリックします。

図 7-129 JMS メッセージ



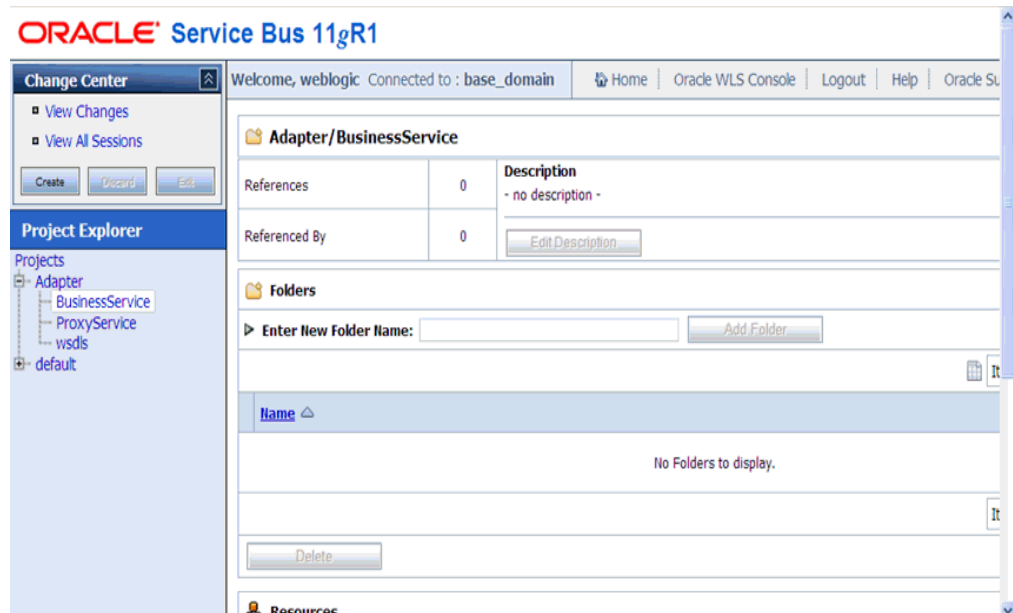
「テキスト」フィールドの下にレスポンス・ドキュメントが表示されます。

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

この項では、J2CA 構成の場合の Oracle Service Bus を使用した HTTP プロキシ・サービスの構成方法について説明します。

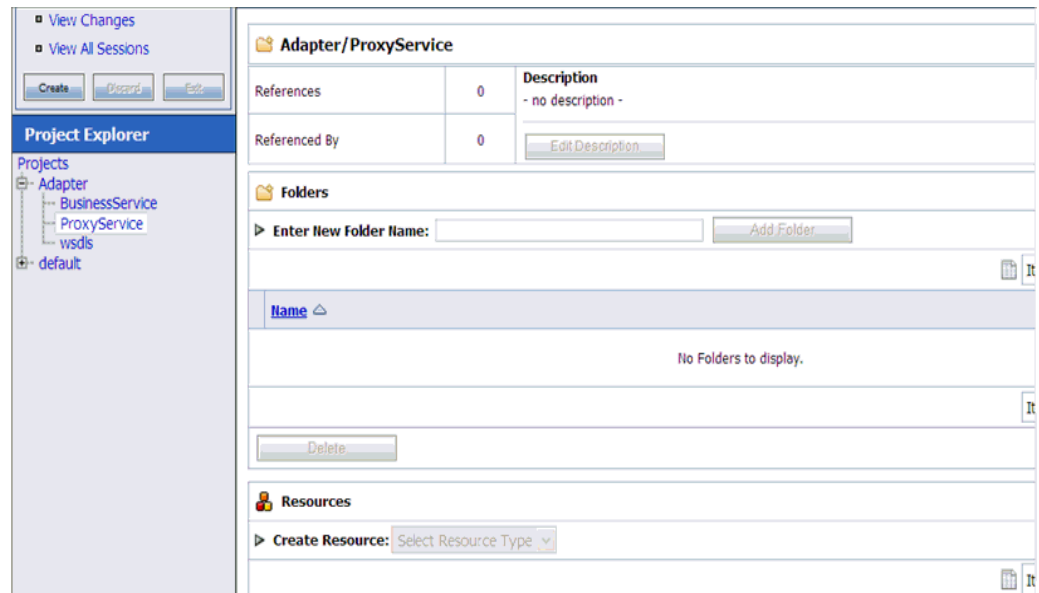
- Oracle Service Bus を起動し、必要なプロジェクト・フォルダを作成します。
詳細は、7.2.1 項「Oracle Service Bus の起動およびプロジェクト・フォルダの作成」を参照してください。
- WSDL を生成して、アプリケーション・エクスプローラから作成済みのプロジェクト・フォルダに公開し、公開した WSDL を使用して、ビジネス・サービスを作成します。
詳細は、7.2.3 項「アプリケーション・エクスプローラから Oracle Service Bus への WSDL の公開」を参照してください。
- 図 7-130 に示すように、Service Bus コンソール・ページを表示します。

図 7-130 Service Bus コンソール・ページ



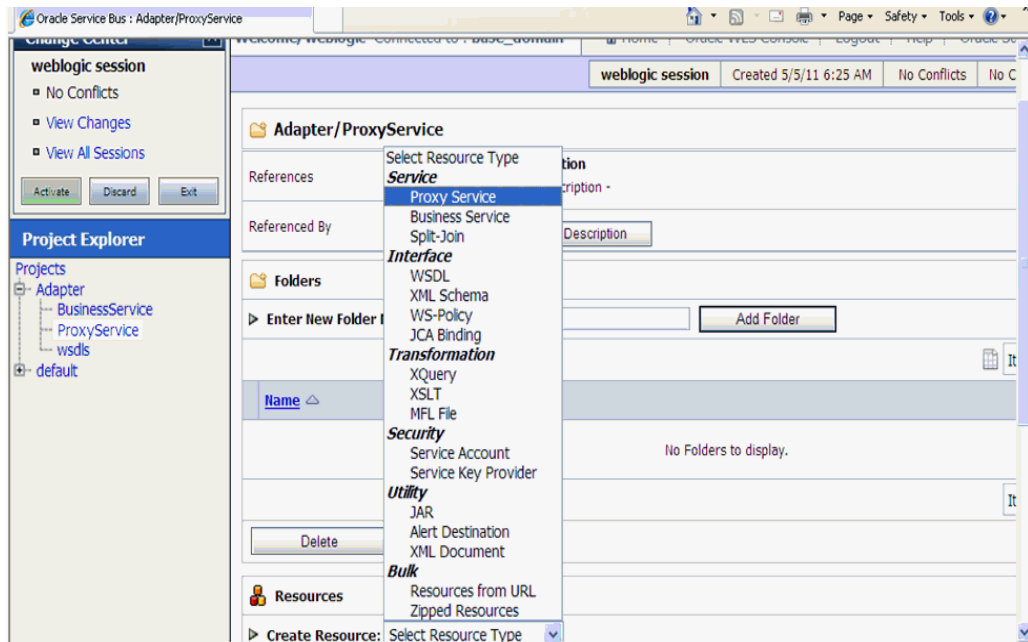
- 図 7-131 に示すように、プロジェクト・エクスプローラで **ProxyService** フォルダを選択して、「作成」をクリックします。

図 7-131 プロジェクト・エクスプローラ



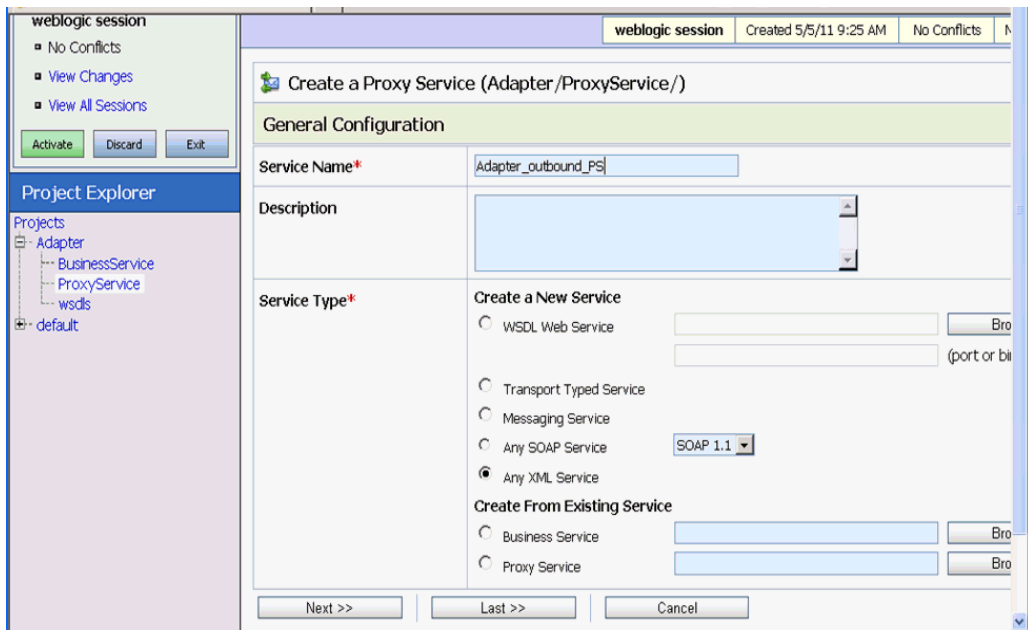
- 図 7-132 に示すように、右ペインの「リソースの作成」リストから「プロキシ・サービス」を選択します。

図 7-132 プロキシ・サービス



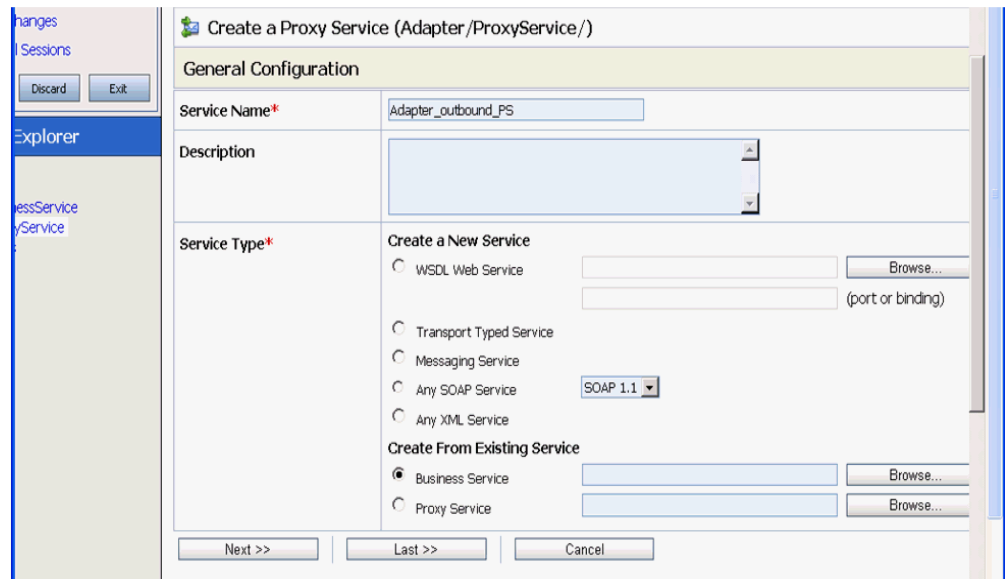
6. 図 7-133 に示すように、「サービス名」フィールドに、適切な名前を入力します。

図 7-133 サービス名



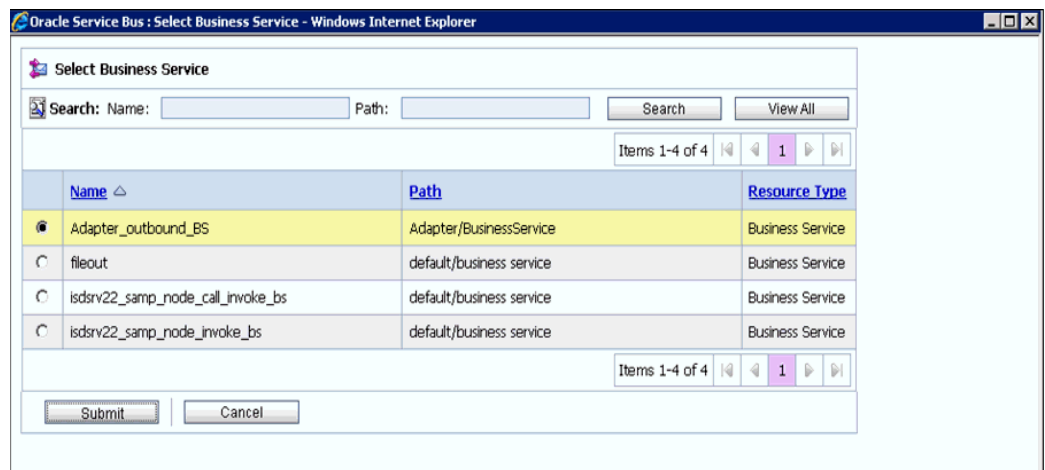
7. 図 7-134 に示すように、「サービス・タイプ」セクションの「既存のサービスから作成」の下にある「ビジネス・サービス」ラジオ・ボタンを選択し、「参照」をクリックします。

図 7-134 全般的な構成



8. 図 7-135 に示すように、既存のビジネス・サービスを選択して、「発行」をクリックします。

図 7-135 ビジネス・サービス



9. 図 7-136 に示すように、「次へ」をクリックします。

図 7-136 一般的な構成

The screenshot shows the 'Create a Proxy Service (Adapter/ProxyService/)' dialog box. The 'General Configuration' section is active. The 'Service Name' field contains 'Adapter_outbound_PS'. The 'Description' field is empty. Under 'Service Type', the 'Create From Existing Service' section is selected, with 'Business Service' chosen. The 'Endpoint URI' field contains 'Adapter/BusinessService/Adapter_outbc'. At the bottom, the 'Next >>' button is highlighted.

10. 図 7-137 に示すように、「プロトコル」リストから「http」を選択して、「次へ」をクリックします。

図 7-137 トランスポート構成

The screenshot shows the 'Create a Proxy Service (Adapter/ProxyService/Adapter_outbound_PS)' dialog box, now on the 'Transport Configuration' tab. The 'Protocol' dropdown is set to 'http'. The 'Endpoint URI' field contains '/Adapter/ProxyService/Adapter_outbound_PS'. The 'Get All Headers' section has 'No' selected. At the bottom, the 'Next >>' button is highlighted.

11. 図 7-138 に示すように、「次へ」をクリックします。

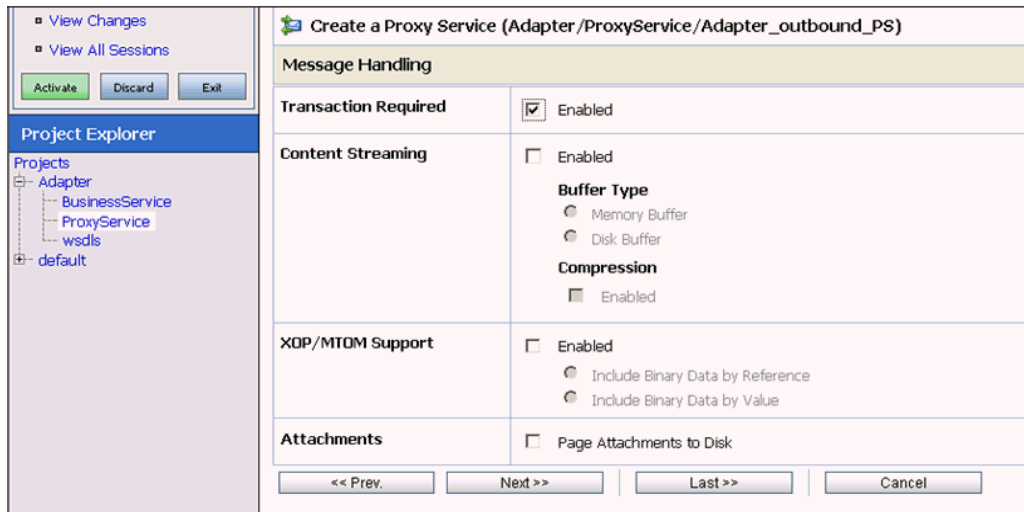
図 7-138 「HTTP トランスポート構成」

12. 図 7-139 に示すように、「次へ」をクリックします。

図 7-139 操作選択構成

13. 図 7-140 に示すように、「トランザクションが必要」チェック・ボックスを選択して、「次へ」をクリックします。

図 7-140 メッセージ処理



14. 図 7-141 に示すように、「保存」をクリックします。

図 7-141 保存

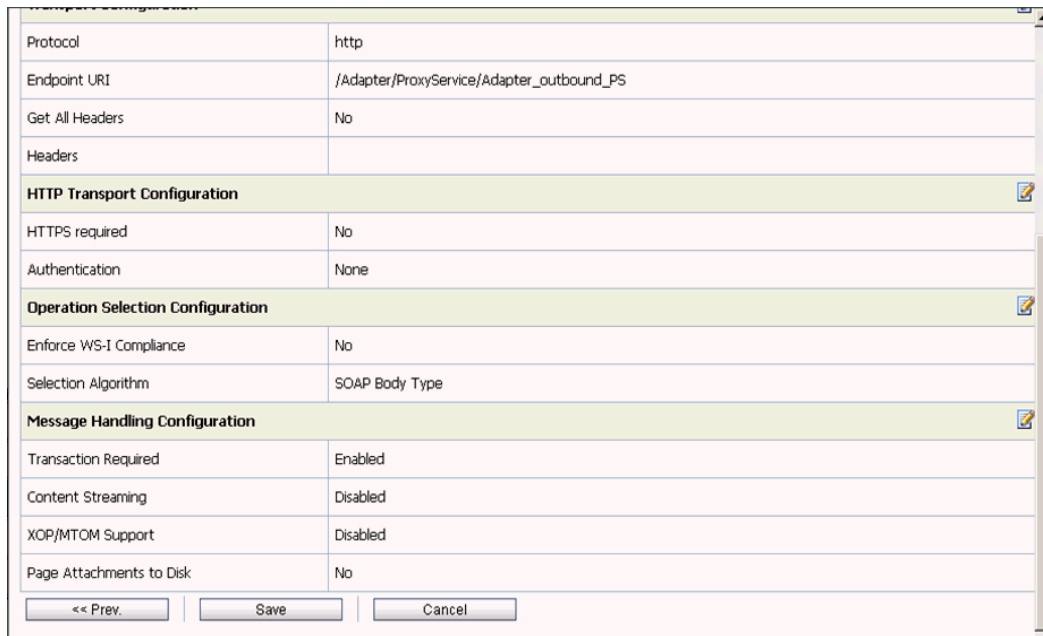
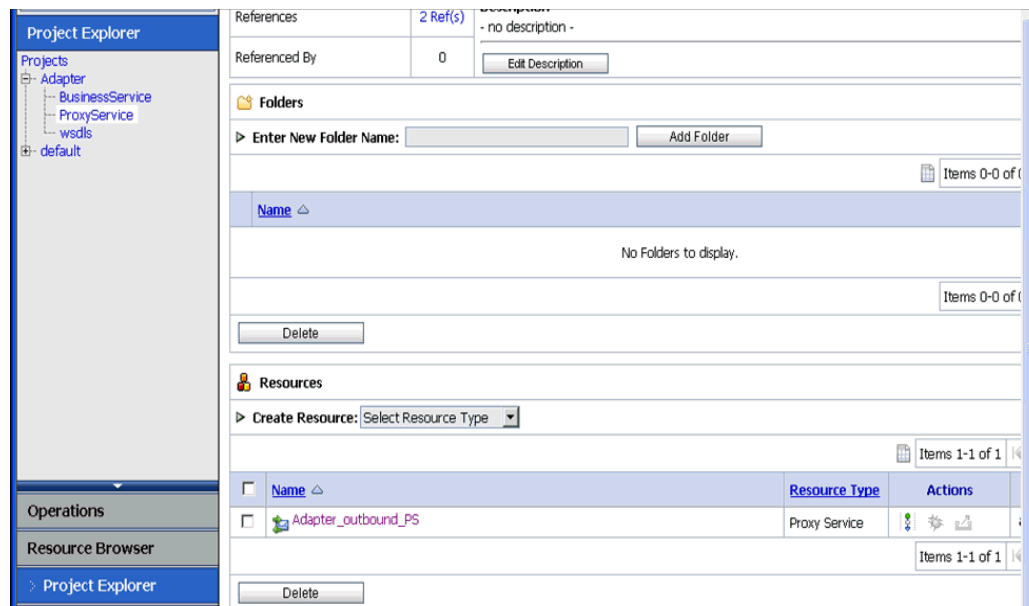


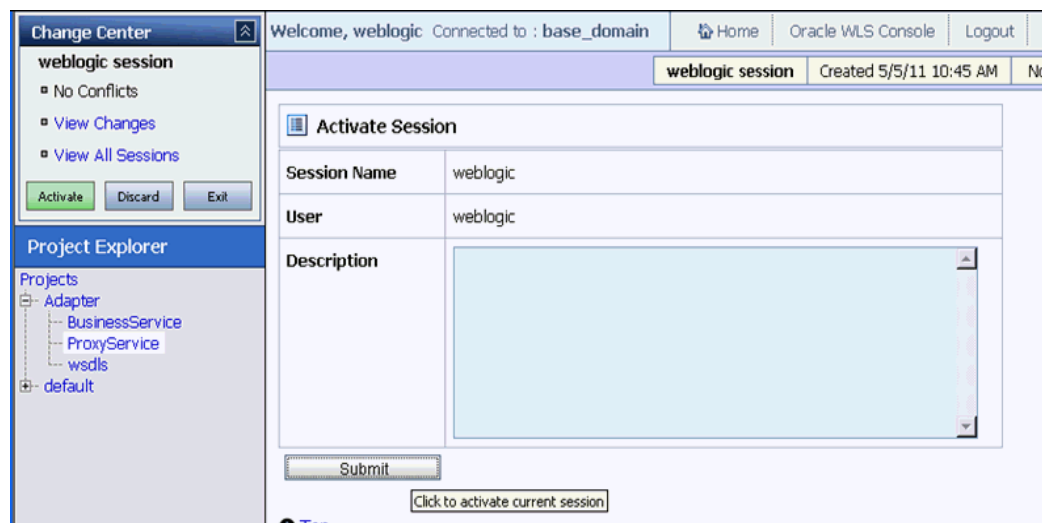
図 7-142 に示すように、作成したプロキシ・サービスが保存されました。

図 7-142 プロキシ・サービス



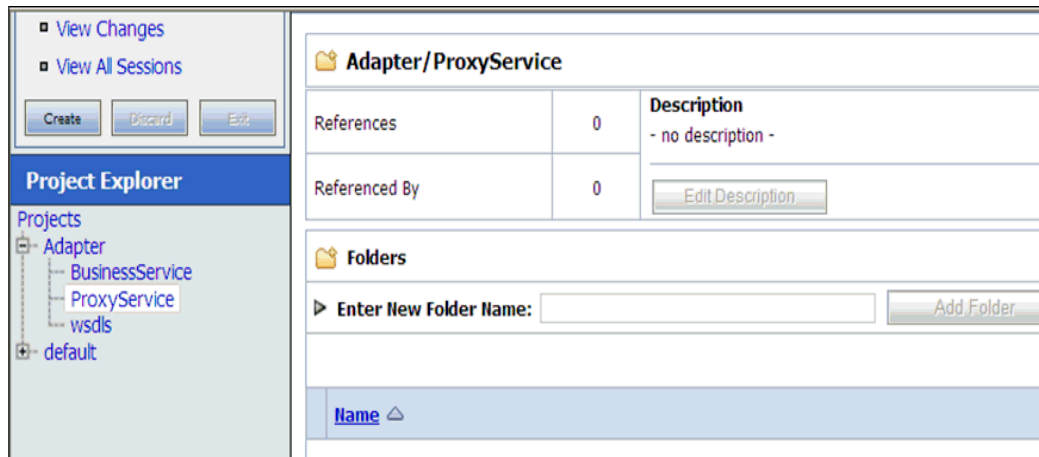
15. 図 7-143 に示すように、左ペインの「アクティブ化」をクリックしてから、右ペインの「発行」をクリックします。

図 7-143 セッションのアクティブ化



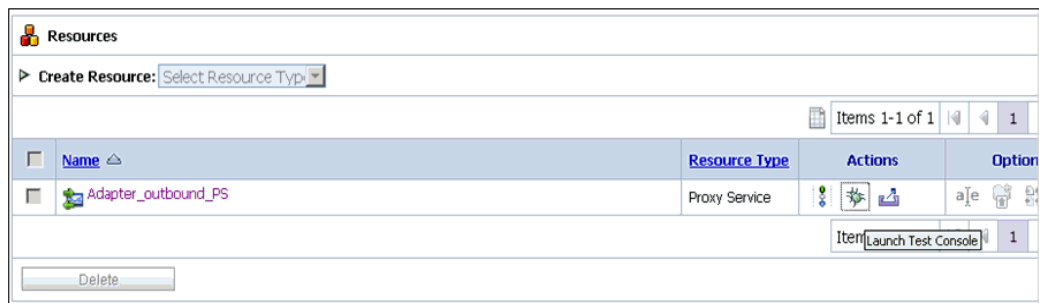
16. 図 7-144 に示すように、左ペインの「プロジェクト」フォルダの下にある **ProxyService** をクリックします。

図 7-144 ProxyService



17. 図 7-145 に示すように、作成したプロキシ・サービスのために「テスト・コンソールの起動」アイコンをクリックします。

図 7-145 テスト・コンソールの起動



18. 「直接呼出し」チェック・ボックスの選択を解除して、「ペイロード」に入力値を入力し、「実行」をクリックします。
19. レスポンス・ドキュメントを確認します。

JDeveloper を使用した Oracle Service Bus の アウトバウンド処理およびインバウンド処理の 構成

Oracle Application Adapter for Siebel では、Oracle JDeveloper がシームレスに統合されており、Web サービスの簡易な統合を可能にします。

この章の内容は次のとおりです。

- 8.1 項「JDeveloper を使用した OSB アウトバウンド処理の構成 (J2CA 構成)」
- 8.2 項「JDeveloper を使用した OSB インバウンド処理の構成 (J2CA 構成)」
- 8.3 項「JDeveloper を使用した OSB アウトバウンド処理の構成 (BSE 構成)」
- 8.4 項「JDeveloper を使用した JMS インバウンド処理の構成 (J2CA 構成)」
- 8.5 項「JDeveloper を使用した JMS アウトバウンド処理の構成 (J2CA 構成)」
- 8.6 項「JDeveloper を使用した HTTP アウトバウンド処理の構成 (J2CA 構成)」

8.1 JDeveloper を使用した OSB アウトバウンド処理の構成 (J2CA 構成)

この項では、J2CA 構成に Oracle JDeveloper を使用して、Siebel システムに OSB 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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストールの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB_
JDeveloper\J2CA\Siebel_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。

- 8.1.1 項「OSB 用の Service Bus アプリケーションの作成」
- 8.1.2 項「OSB アウトバウンド処理の定義」
- 8.1.3 項「OSB アウトバウンド処理のデプロイ」

前提条件

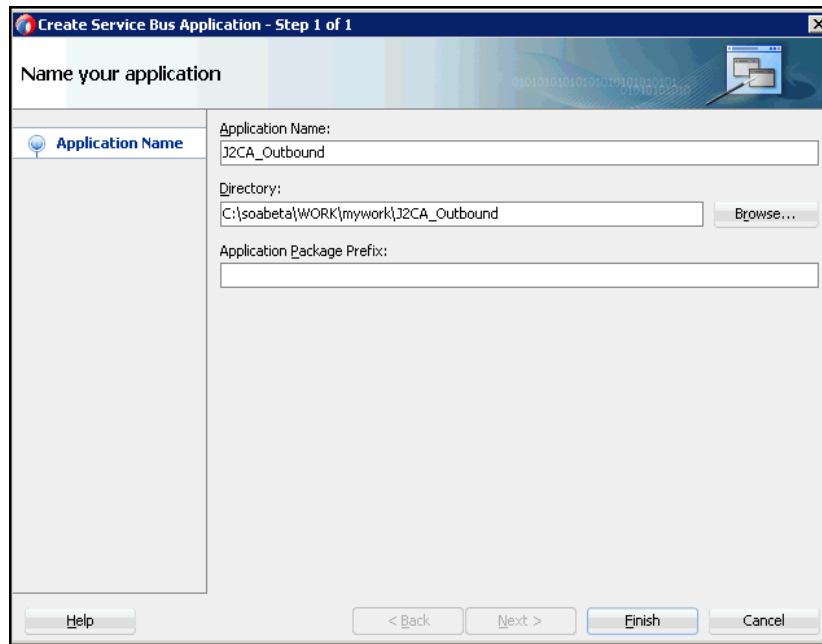
OSB アウトバウンド処理を設計する前に、アプリケーション・エクスプローラを使用して対応する WSDL ファイルを生成する必要があります。詳細は、4-8 ページ [4.4.1 項「リクエスト/レスポンス・サービス用の WSDL の生成」](#) を参照してください。

8.1.1 OSB 用の Service Bus アプリケーションの作成

OSB 用の Service Bus アプリケーションを作成するには、JDeveloper で次のステップを実行します。

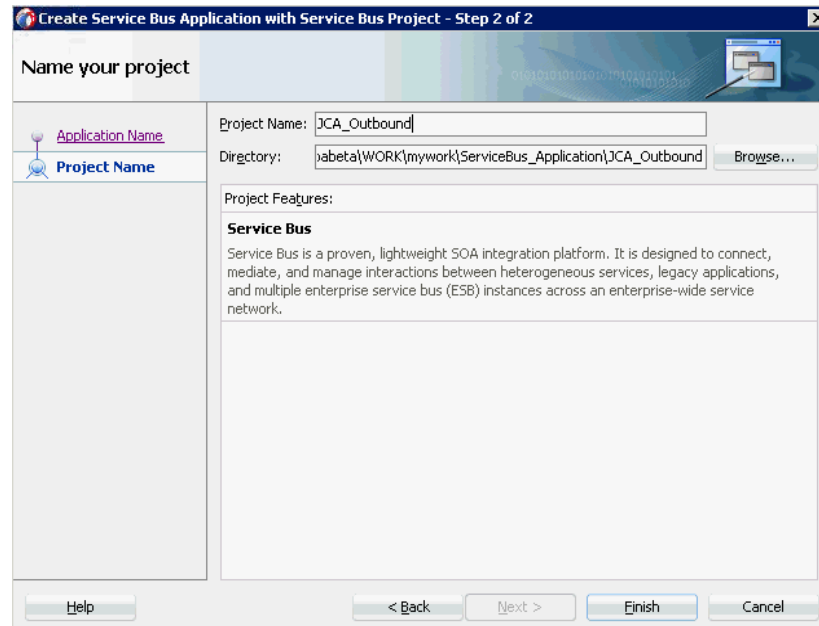
1. 新規の OSB アプリケーションを作成します。
2. 図 8-1 に示すように、OSB アプリケーションの名前を入力し (J2CA_Outbound など)、「終了」をクリックします。

図 8-1 「アプリケーションの名前付け」 ペイン



3. 図 8-2 に示すように、プロジェクトの名前 (JCA_Outbound など) を入力して「終了」をクリックします。

図 8-2 「プロジェクトの名前付け」 ペイン



8.1.2 OSB アウトバウンド処理の定義

この項では、OSB アウトバウンド処理の定義方法について説明します。内容は、次のとおりです。

- 8.1.2.1 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」
- 8.1.2.2 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」
- 8.1.2.3 項「パイプラインを備えたプロキシ・サービスの作成」
- 8.1.2.4 項「ルーティング・ルールの構成」

8.1.2.1 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

サード・パーティ・アダプタ・サービス・コンポーネントをビジネス・サービスとともに作成するには、次のステップを実行します。

1. 図 8-3 に示すように、Service Bus コンポーネント・ペインから「サード・パーティ・アダプタ」コンポーネントを外部サービス・ペインにドラッグ・アンド・ドロップします。

図 8-3 「サード・パーティ・アダプタ」コンポーネント

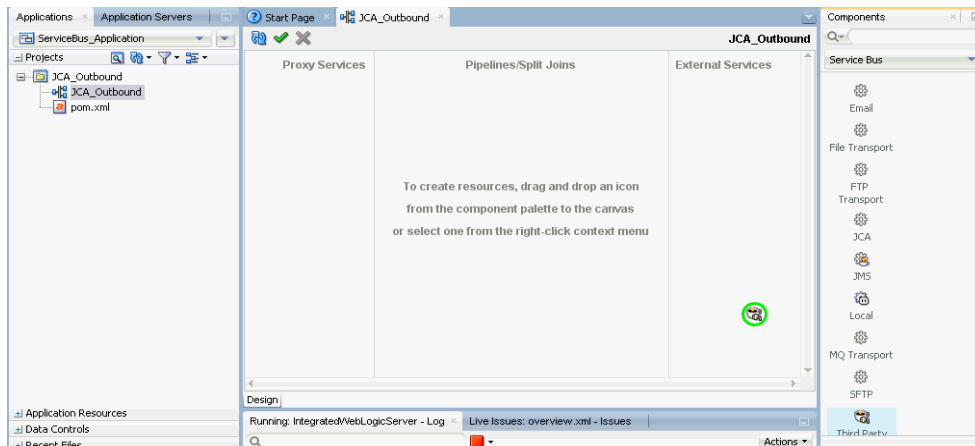
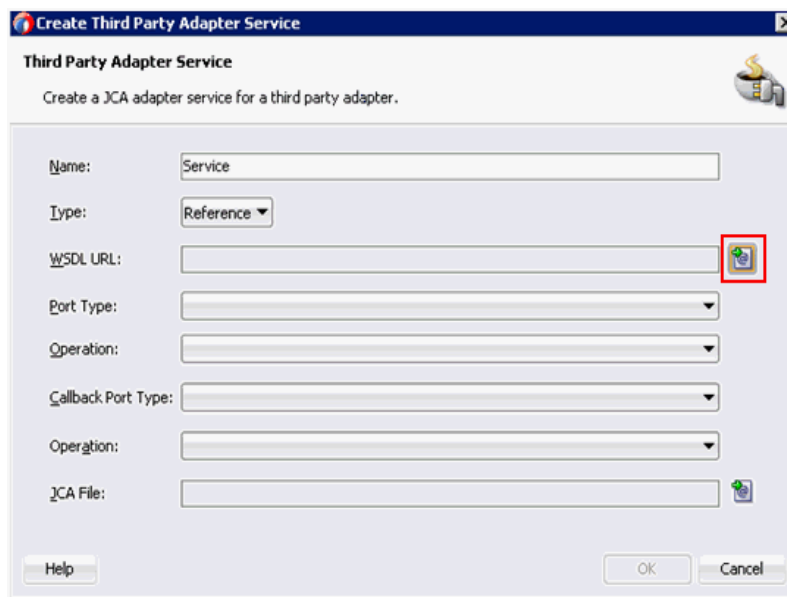


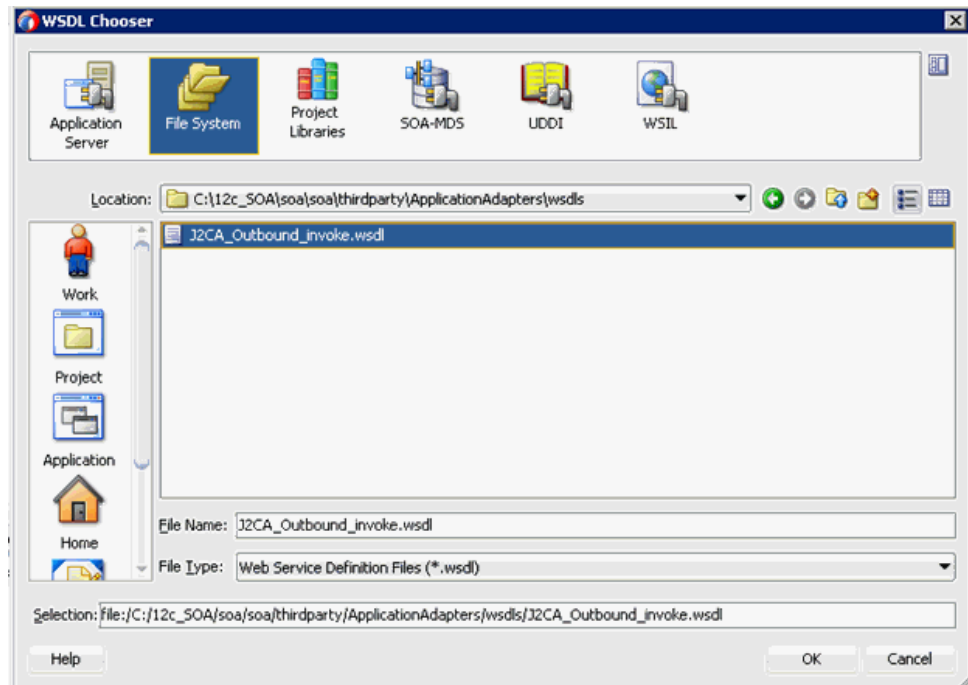
図 8-4 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログが表示されます。

図 8-4 「サード・パーティ・アダプタ・サービスの作成」ペイン



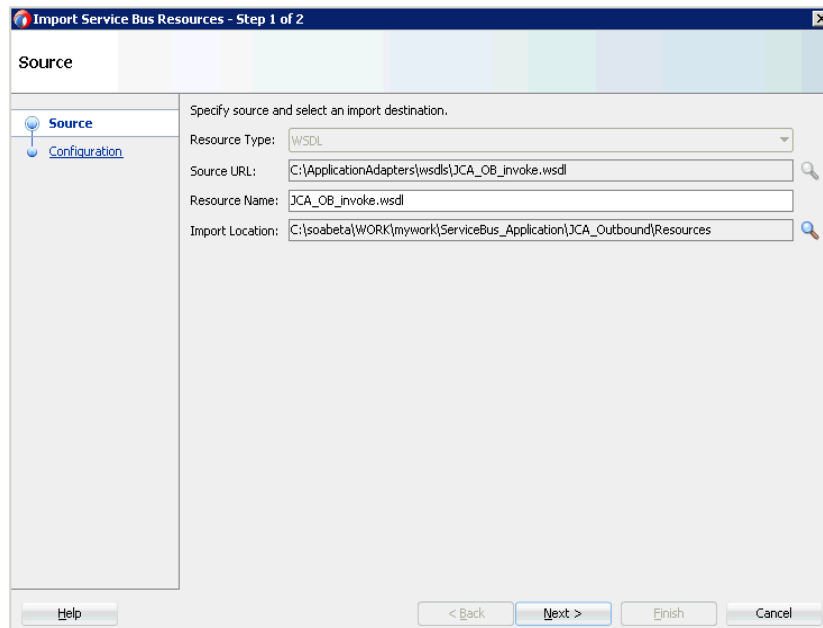
2. サード・パーティ・アダプタ・サービスの適切な名前を入力します。この名前はビジネス・サービスの名前として使用されます。
3. 「タイプ」ドロップダウン・リストで「参照」が選択されていることを確認します (デフォルト)。
4. 「WSDL URL」フィールドの右の既存の WSDL を検索アイコンをクリックします。
 図 8-5 に示すように、「WSDL チューザ」ダイアログが表示されます。

図 8-5 「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 ディレクトリのアウトバウンド WSDL ファイルを参照して選択します。
6. 「OK」をクリックします。
Service Bus リソースのインポート・ダイアログが表示されます。
7. 図 8-6 に示すように、「次へ」をクリックします。

図 8-6 「ソース」 ペイン



8. 「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。


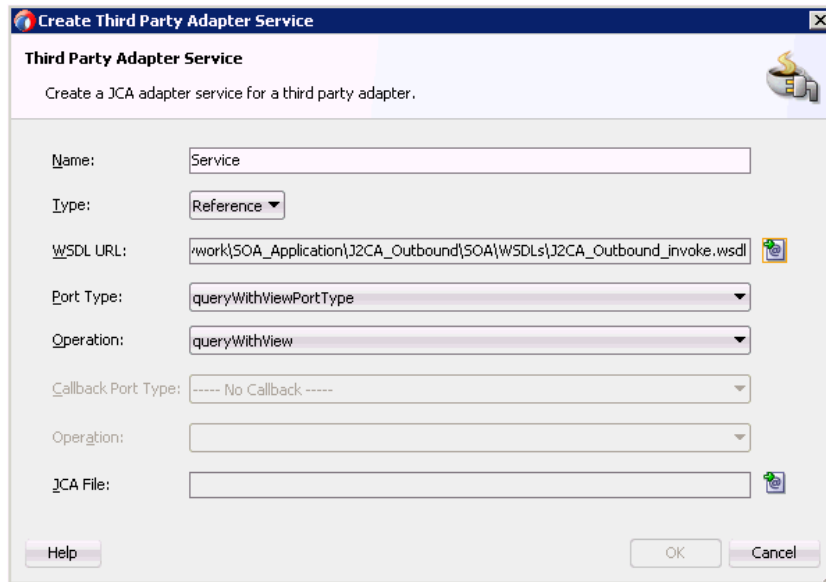
9.  8-7 に示すように、「JCA ファイル」フィールドの右の JCA ファイルの検索アイコンをクリックします。

図 8-7 JCA ファイルの検索アイコン



「XQuery 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. 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。「ファイルのコピー」メッセージ表示されます。
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。


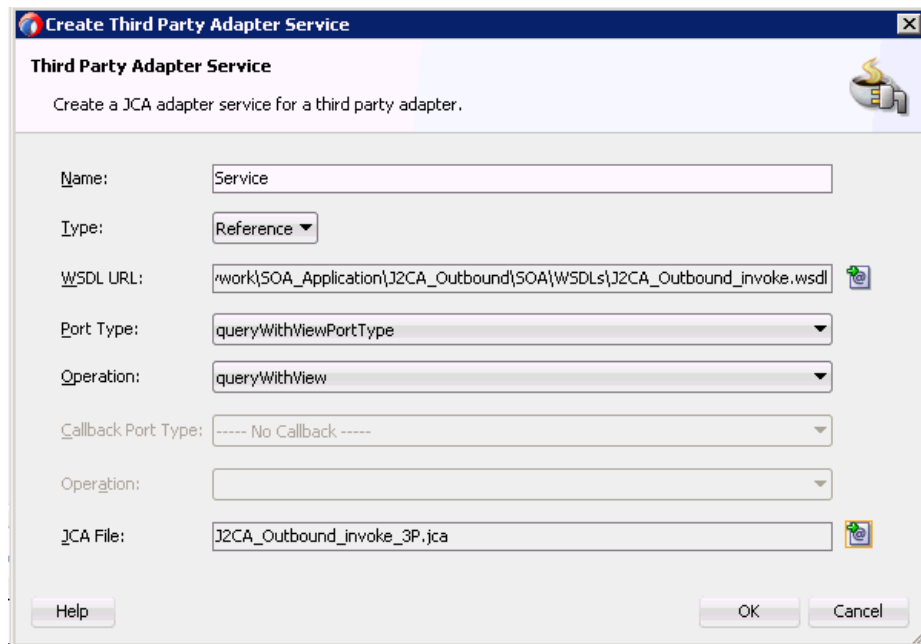
 8-8 に示すように、「サード・パーティ・アダプタ・サービスの作成」ダイアログに戻ります。

図 8-8 「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。

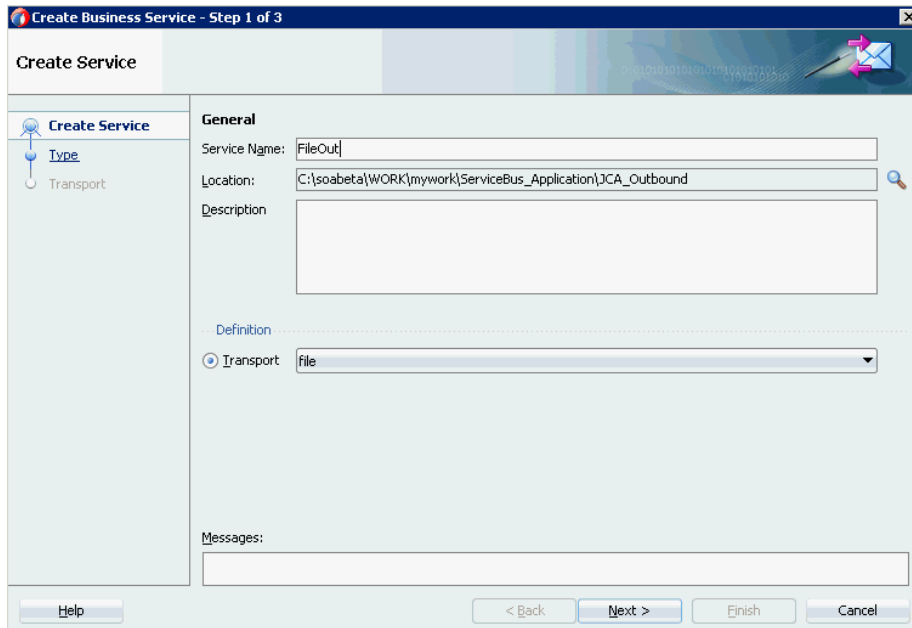
外部サービス・ペインにビジネス・サービス・コンポーネントが作成されます。

8.1.2.2 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、次のステップを実行します。

1. 「拡張」ペインからファイル・トランスポート・コンポーネントを外部サービス・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」ダイアログが表示されます。
2. 図 8-9 に示すように、「サービス名」フィールドに、任意のビジネス・サービスの名前 (FileOut など) を入力し、「次へ」をクリックします。

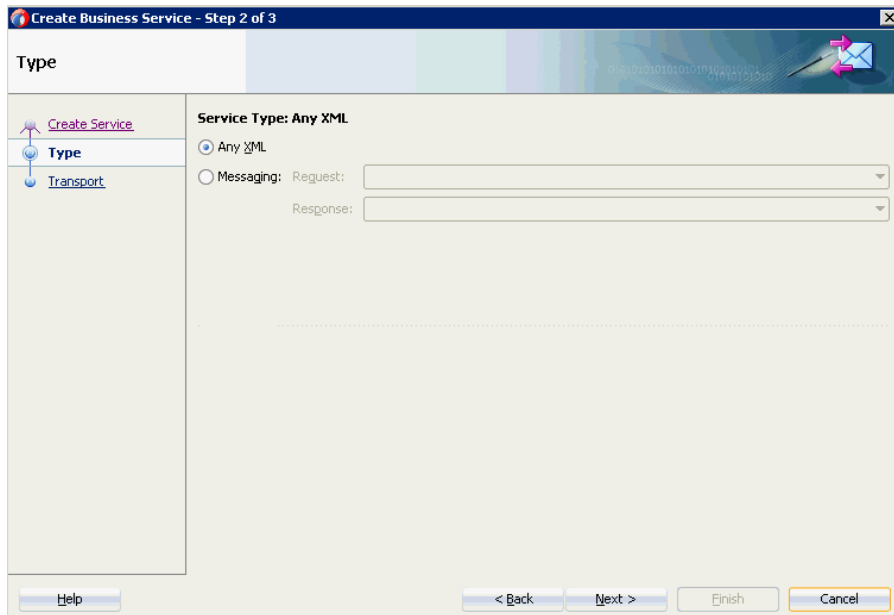
図 8-9 「サービスの作成」 ペイン



「タイプ」 ペインが表示されます。「任意の XML」 オプションがデフォルトで選択されています。

3. 図 8-10 に示すように、「次へ」をクリックします。

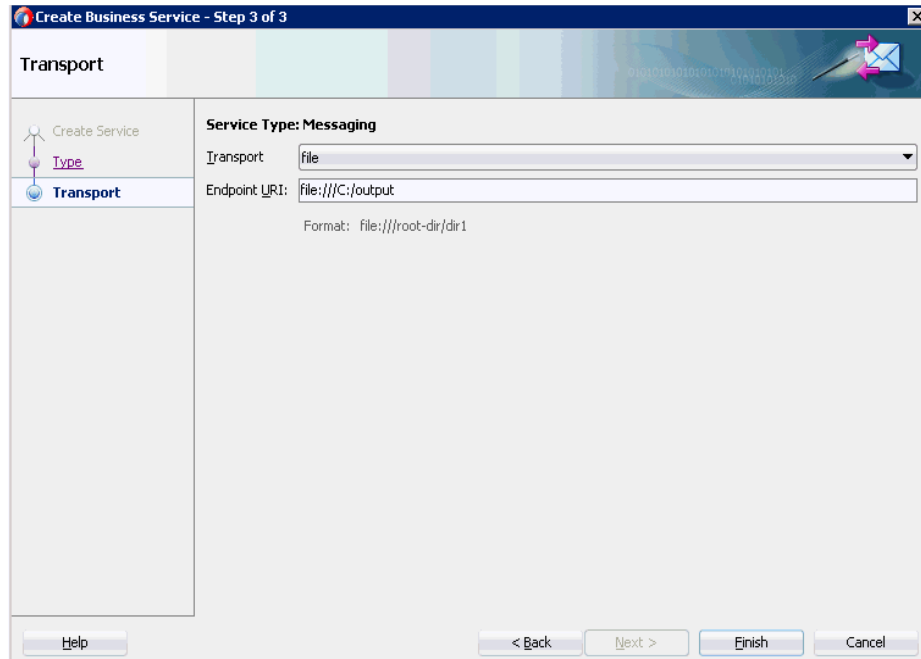
図 8-10 「タイプ」 ペイン



「トランスポート」 ペインが表示されます。

4. 図 8-11 に示すように、「エンドポイント URI」 フィールドに出力場所を入力し (「c:/output」 など)、「終了」をクリックします。

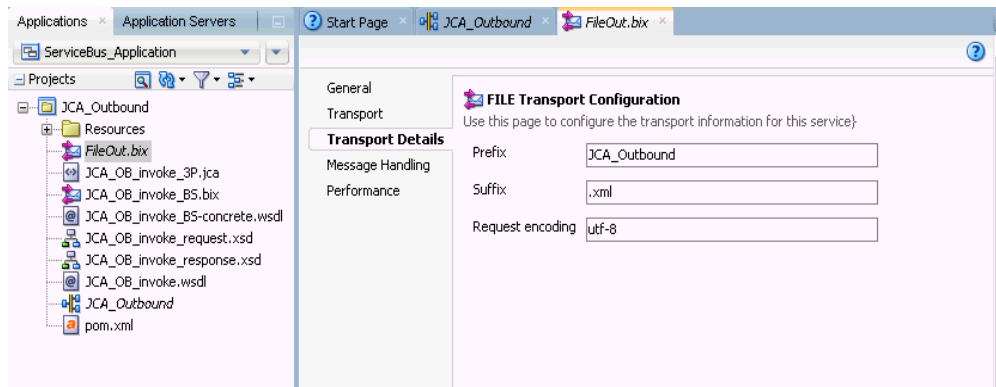
図 8-11 「トランスポート」 ペイン



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 8-12 に示すように、作成したビジネス・サービス「Fileout」をダブルクリックし、「トランスポートの詳細」タブの「接頭辞」フィールドに値を入力します。

図 8-12 トランスポートの詳細

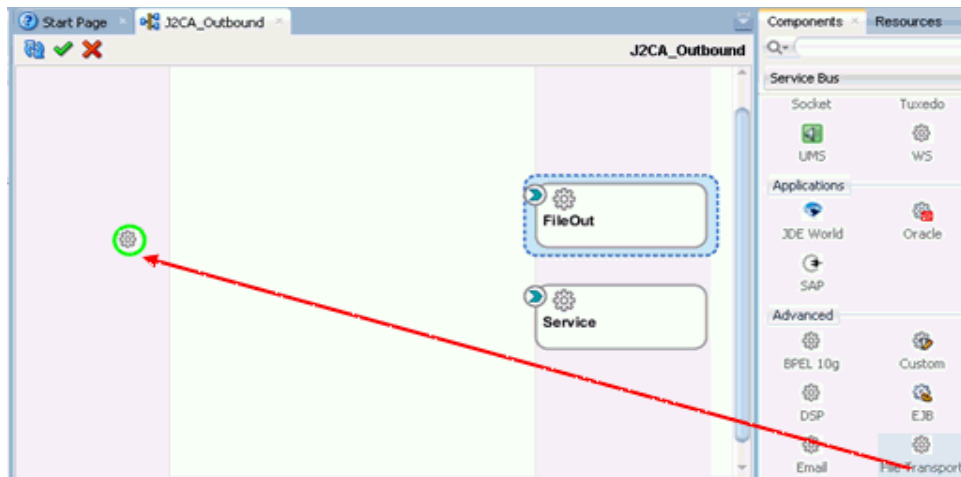


8.1.2.3 パイプラインを備えたプロキシ・サービスの作成

パイプラインを備えたプロキシ・サービスを作成するには、次のステップを実行します。

1. 図 8-13 に示すように、拡張コンポーネント・ペインからファイル・トランスポート・コンポーネントを「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。

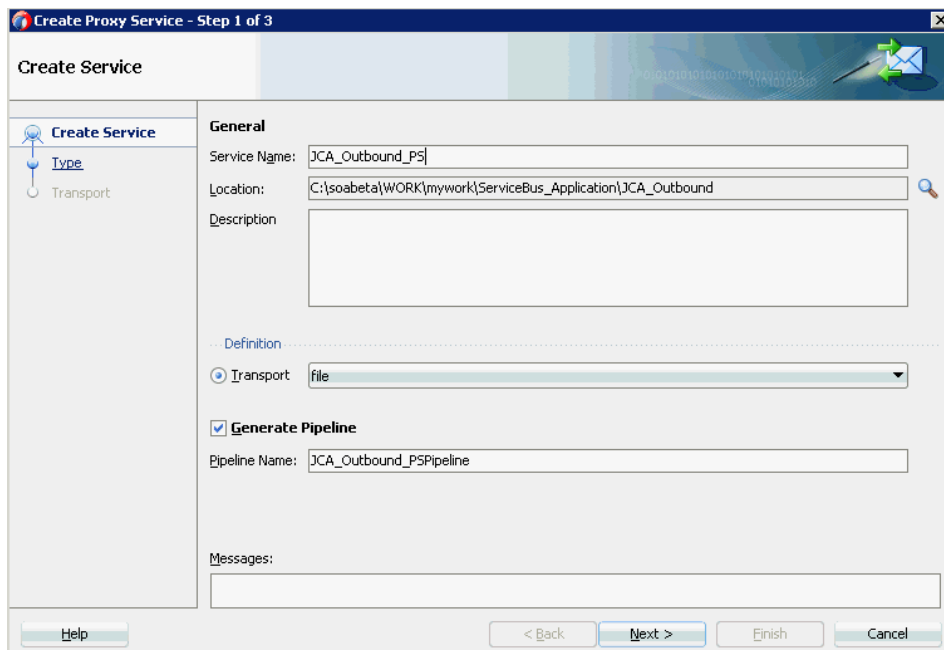
図 8-13 ファイル・トランスポート・コンポーネント



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、任意のプロキシ・サービスの名前 (JCA_Outbound_PS など) を入力します。「パイプラインを生成する」がデフォルトで選択されています。
3. 図 8-14 に示すように、「次へ」をクリックします。

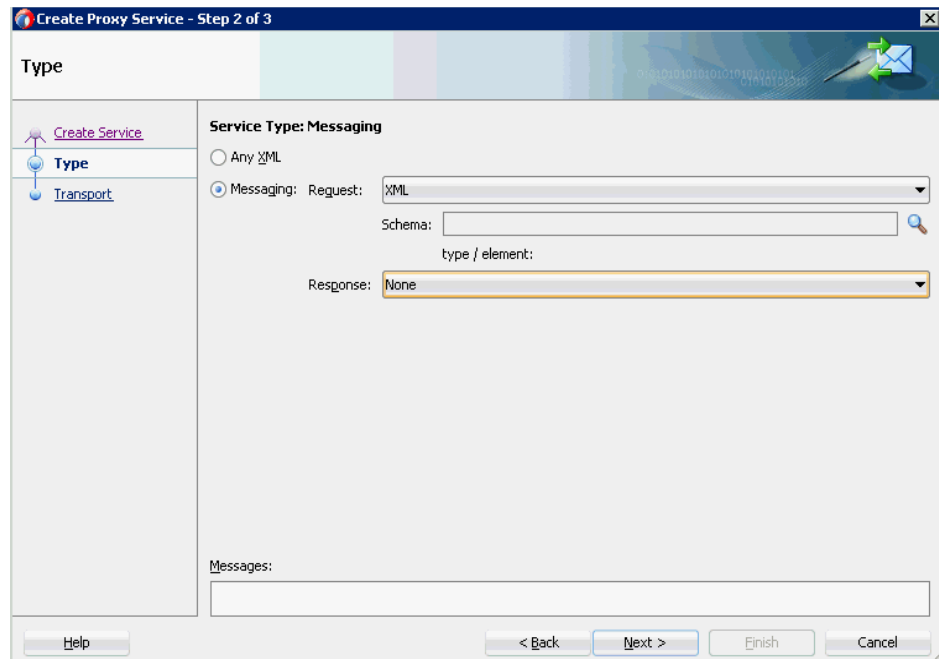
図 8-14 「サービスの作成」ペイン



「タイプ」ペインが表示されます。

4. 図 8-15 に示すように、「メッセージ機能」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「次へ」をクリックします。

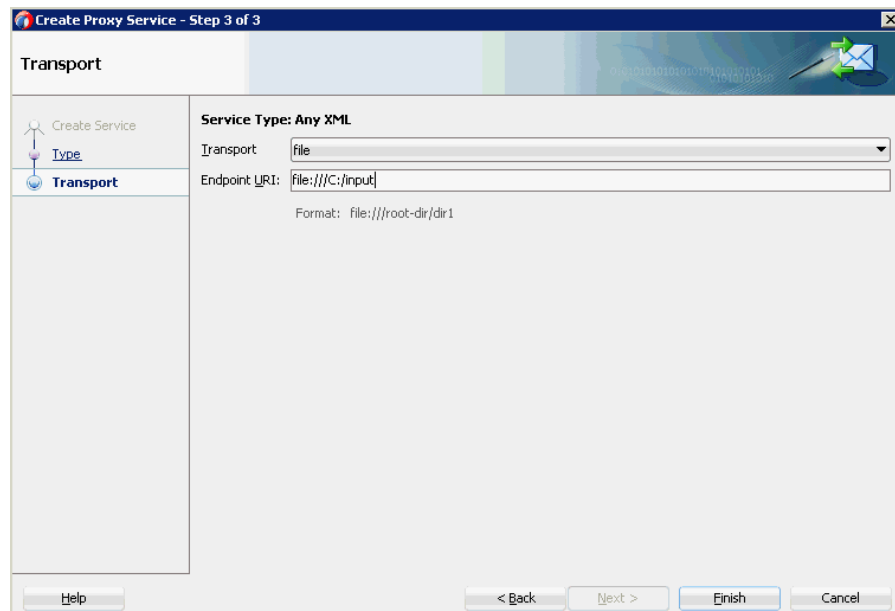
図 8-15 「タイプ」 ペイン



「トランスポート」ウィンドウが表示されます。

5. 図 8-16 に示すように、「エンドポイント URI」フィールドに入力場所を指定し（「c:/input」など）、「終了」をクリックします。

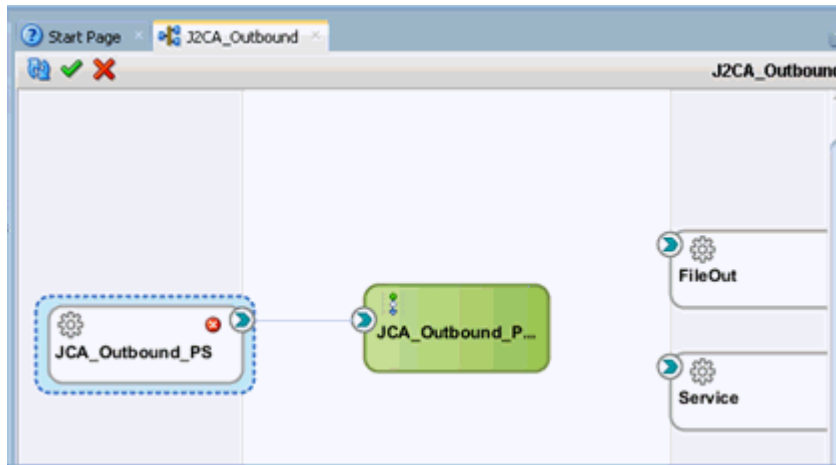
図 8-16 「トランスポート」ウィンドウ



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。

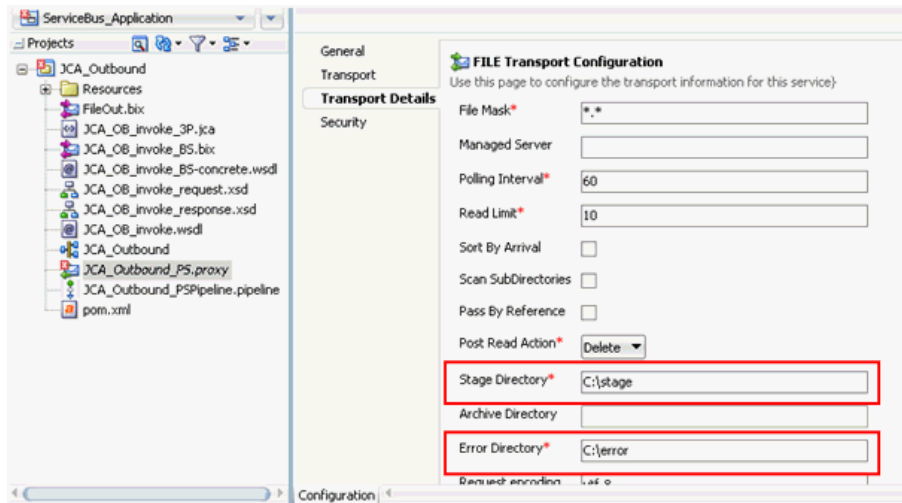
6. 図 8-17 に示すように、作成したプロキシ・サービス (JCA_Outbound_PS など) をダブルクリックします。

図 8-17 プロキシ・サービスの編集



7. 図 8-18 に示すように、表示される「プロキシ・サービスの構成」ページで「トランスポートの詳細」を選択し、「ステージ・ディレクトリ」と「エラー・ディレクトリ」に値を入力します。

図 8-18 FILE トランスポート構成



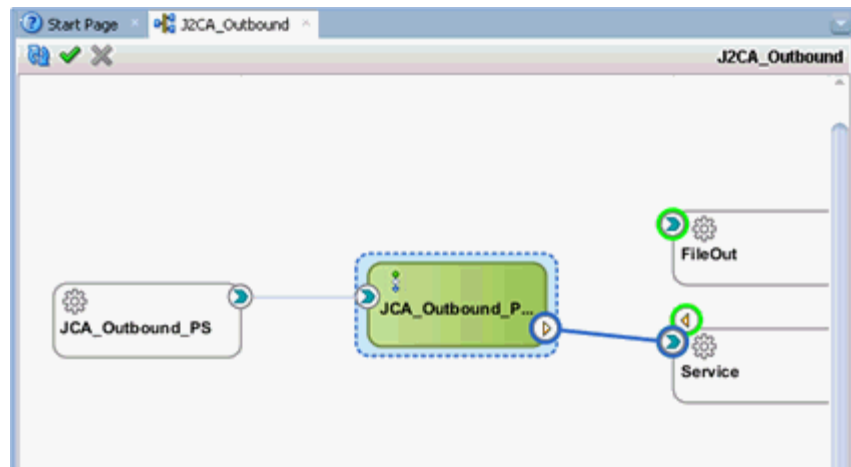
8. 「プロキシ・サービスの構成」ページを保存して閉じます。

8.1.2.4 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。

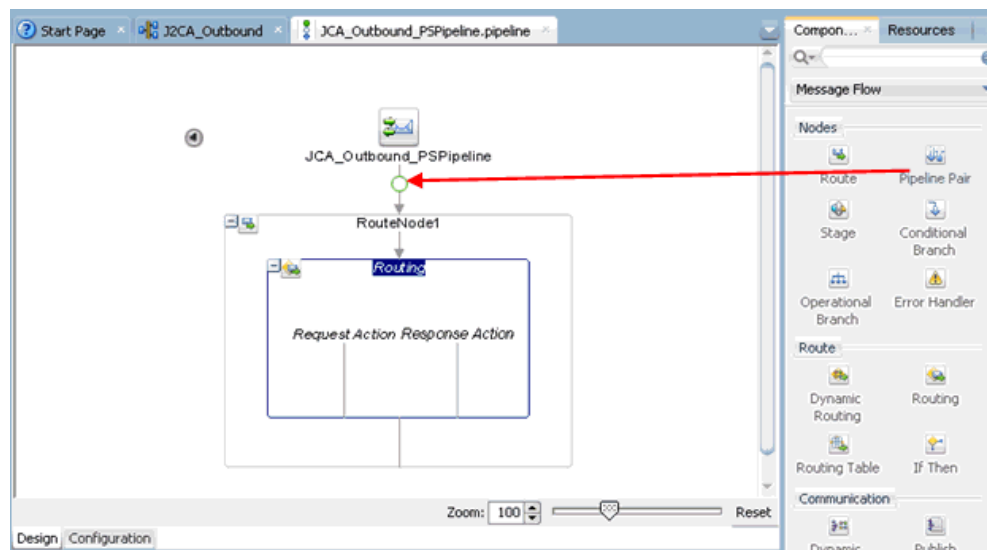
1. 図 8-19 に示すように、パイプラインをビジネス・サービス（「Service」など）に接続します。

図 8-19 ビジネス・サービス・パイプライン



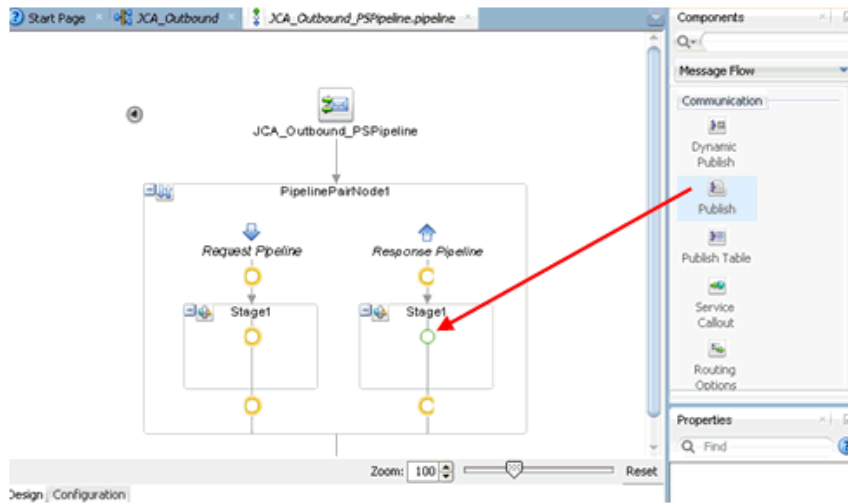
2. パイプライン/分割結合ペインでパイプライン(JCA_Outbound_PSPipeline など)をダブルクリックします。
パイプライン構成ページが表示されます。
3. 図 8-20 に示すように、「ノード」ペインからパイプライン・ペア・ノードをパイプライン (JCA_Outbound_PSPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 8-20 パイプライン・ペア・ノード



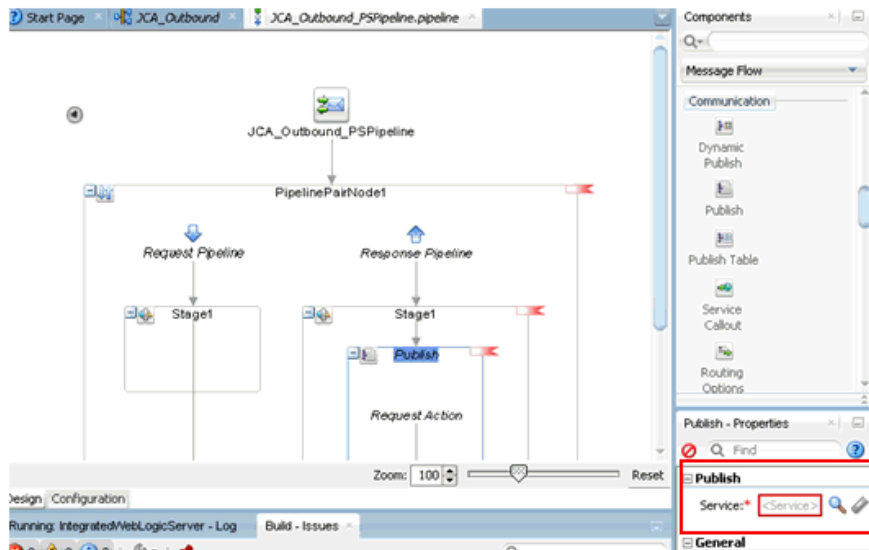
4. 図 8-21 に示すように、「通信」ペインから「公開」ノードを「レスポンス・パイプライン」の「Stage1」の下領域にドラッグ・アンド・ドロップします。

図 8-21 「公開」ノード



5. 図 8-22 に示すように、公開の右ペインにある「サービス」アイコンをクリックします。

図 8-22 「参照」アイコン



6. 図 8-23 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 8-23 リソース・チューザ

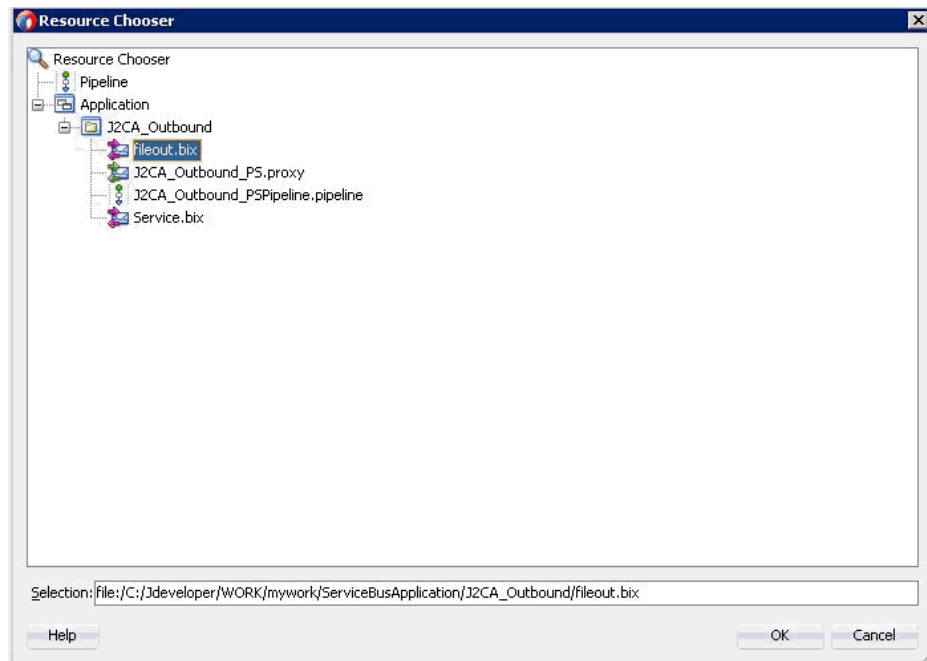
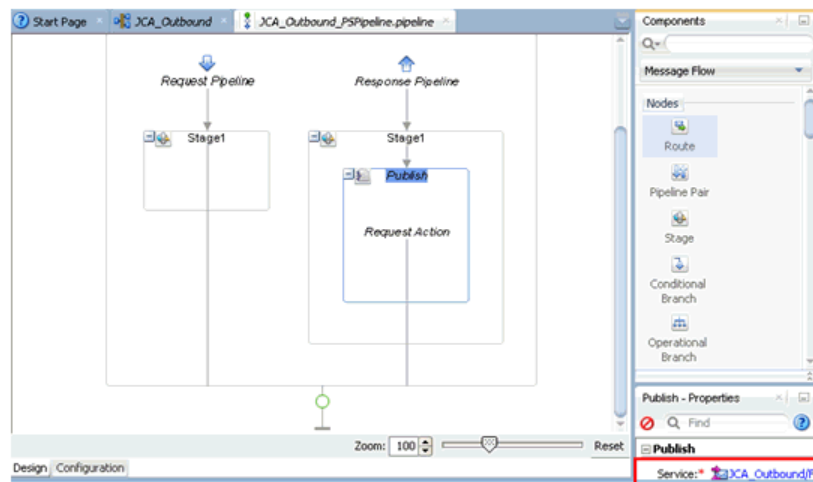


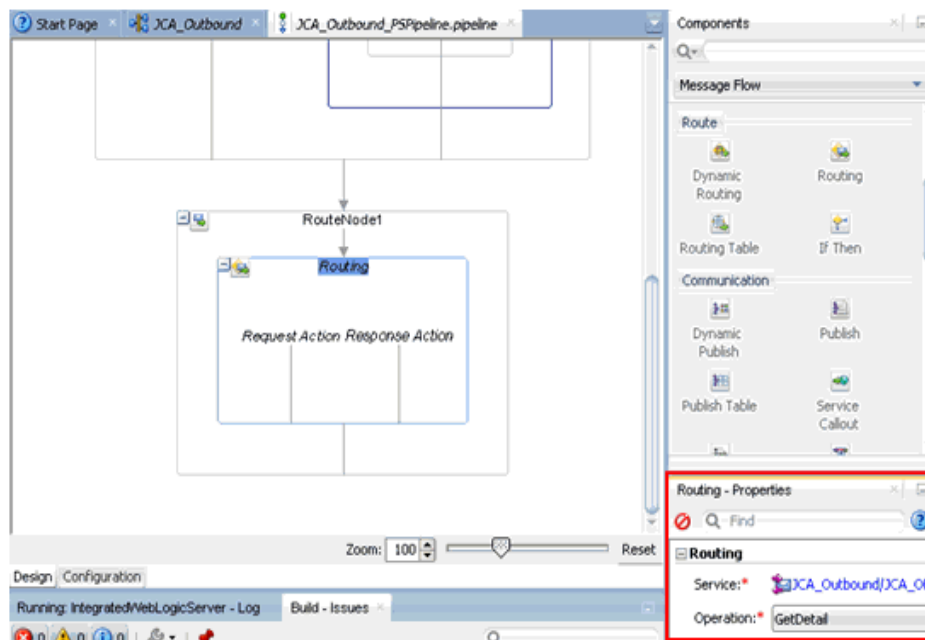
図 8-24 に示すように、右ペインで、選択したサービスが「公開」ペインで構成されます。

図 8-24 「公開」ペイン



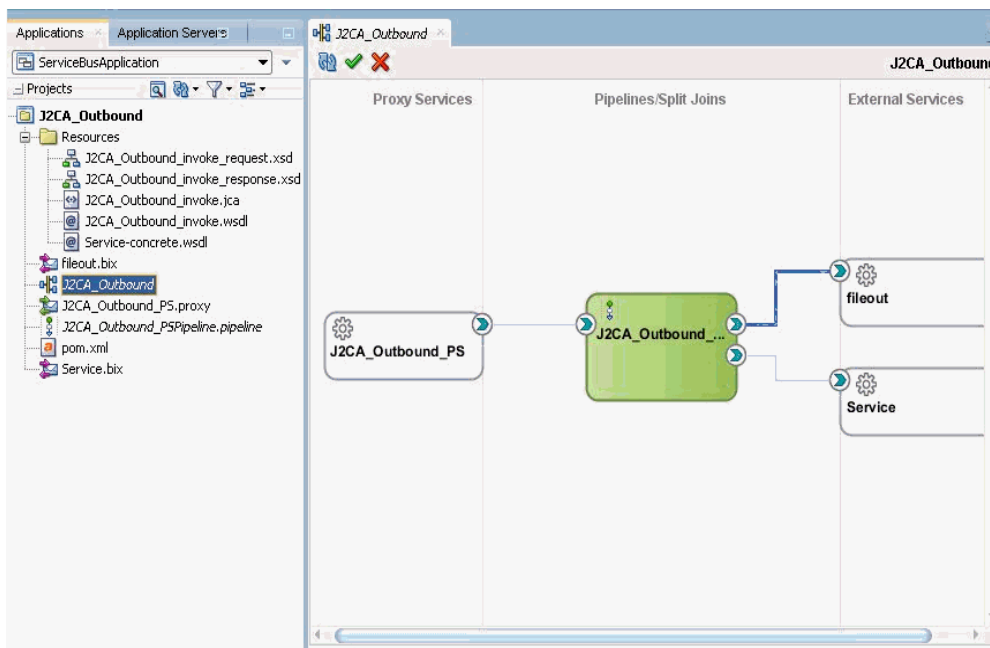
7. 図 8-25 に示すように、「ルーティング」が適切に選択されていることを確認します。

図 8-25 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 8-26 に示すように、overview.xml ファイル (JCA_Outbound など) をダブルクリックし、メニュー・バーの「すべて保存」をクリックして OSB プロセスを保存します。

図 8-26 「すべて保存」アイコン

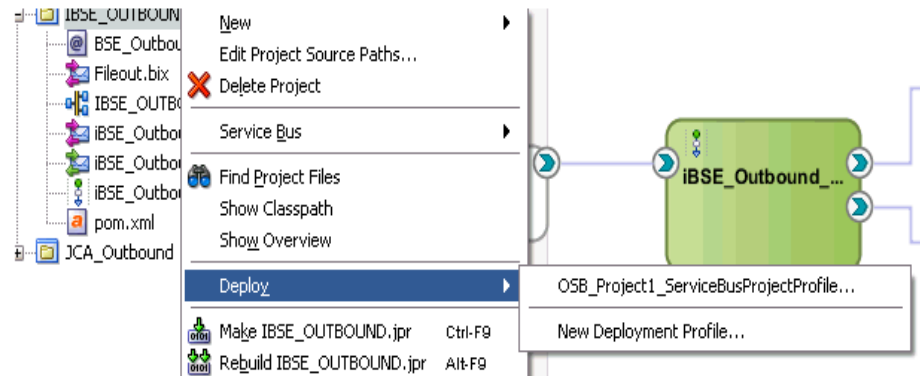


8.1.3 OSB アウトバウンド処理のデプロイ

OSB アウトバウンド処理をデプロイするには、次のステップを実行します。

1. 図 8-27 に示すように、OSB プロジェクト名を右クリックし、「デプロイ」を選択して、「OSB_Project1_ServiceBusProjectProfile...」を選択します。

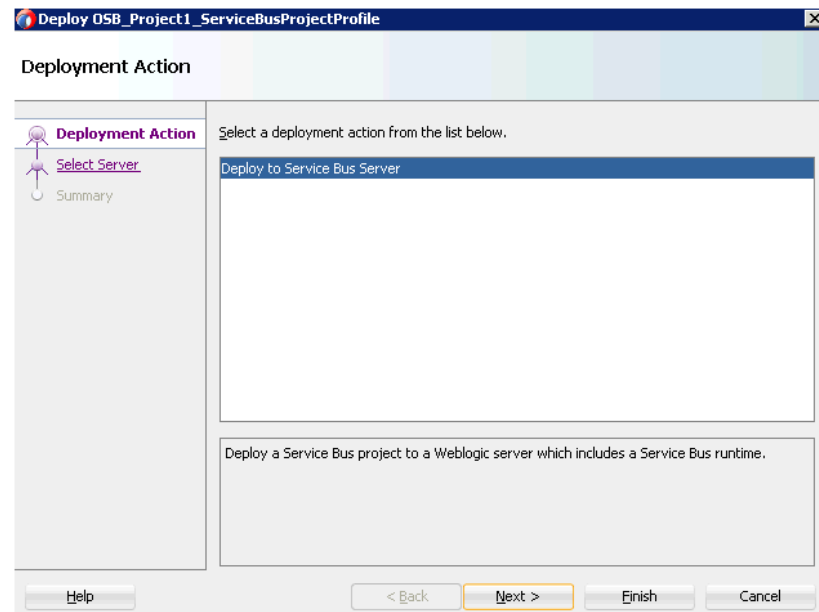
図 8-27 「デプロイ」オプション



「デプロイメント・アクション」ページが表示されます。

2. 図 8-28 に示すように、「次へ」をクリックします。

図 8-28 「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 8-29 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 8-29 「サーバーの選択」 ページ

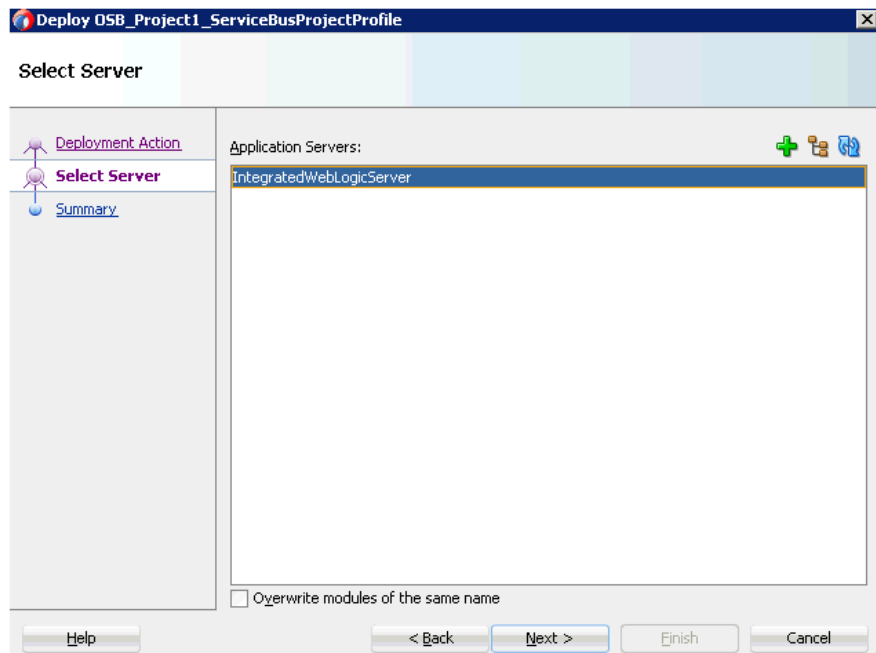
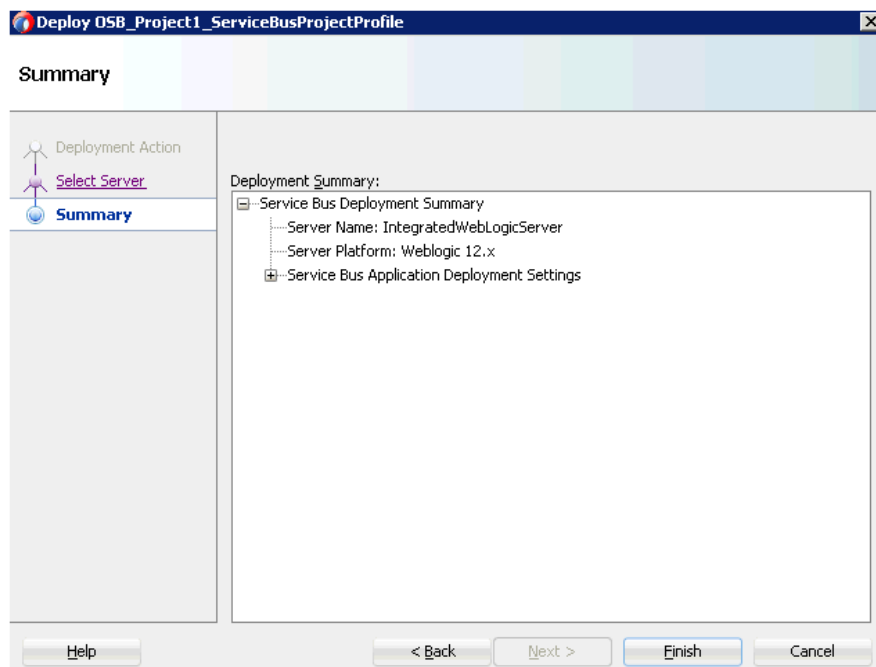


図 8-30 に示すように、「サマリー」 ページが表示されます。

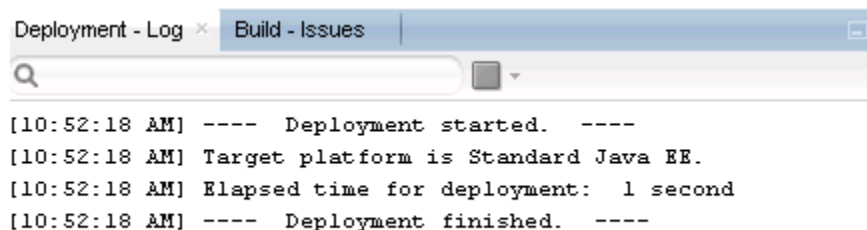
図 8-30 「サマリー」 ページ



- プロジェクトの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 8-31 に示すように、プロセスが正常にデプロイされます。

図 8-31 デプロイメントの成功を伝えるメッセージ



```

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. 入力 XML ファイルをコピーし、構成した入力フォルダ (C:\input など) に貼り付けます。

出力が構成済の出力場所 (C:\output など) で受信されます。

8.2 JDeveloper を使用した OSB インバウンド処理の構成 (J2CA 構成)

この項では、J2CA 構成に Oracle JDeveloper を使用して、Siebel システムに OSB インバウンド処理を構成する方法について説明します。

このインバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストレーションの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB_
JDeveloper\J2CA\Siebel_Sample_J2CA_OSB_Inbound_Project
```

この項では、次のトピックについて説明します。

- [8.2.1 項「OSB 用の Service Bus アプリケーションの作成」](#)
- [8.2.2 項「OSB インバウンド処理の定義」](#)
- [8.2.3 項「OSB インバウンド処理のデプロイ」](#)

前提条件

OSB インバウンド処理を設計する前に、アプリケーション・エクスプローラを使用して対応する WSDL ファイルを生成する必要があります。詳細は、4-34 ページ [4.5.1 項「イベント統合のための WSDL の生成」](#) を参照してください。

8.2.1 OSB 用の Service Bus アプリケーションの作成

OSB インバウンド処理を構成するには、OSB 用の Service Bus アプリケーションを作成する必要があります。詳細は、8-2 ページ [8.1.1 項「OSB 用の Service Bus アプリケーションの作成」](#) を参照してください。

8.2.2 OSB インバウンド処理の定義

この項では、OSB インバウンド処理の定義方法について説明します。内容は、次のとおりです。

- [8.2.2.1 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」](#)
- [8.2.2.2 項「パイプラインの作成」](#)
- [8.2.2.3 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」](#)

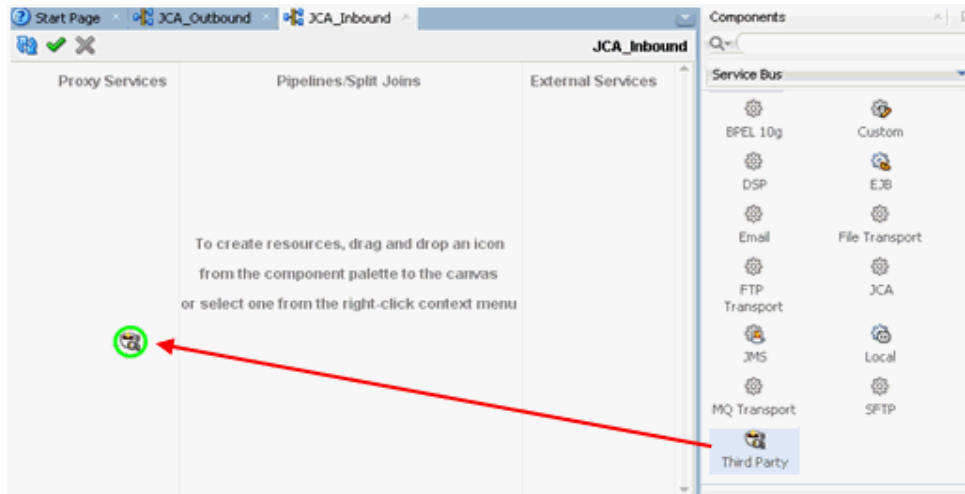
■ 8.2.2.4 項「ルーティング・ルール」の構成

8.2.2.1 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 8-32 に示すように、Service Bus コンポーネント・ペインから「サード・パーティ」アダプタ・コンポーネントを「プロキシ・サービス」にドラッグ・アンド・ドロップします。

図 8-32 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。

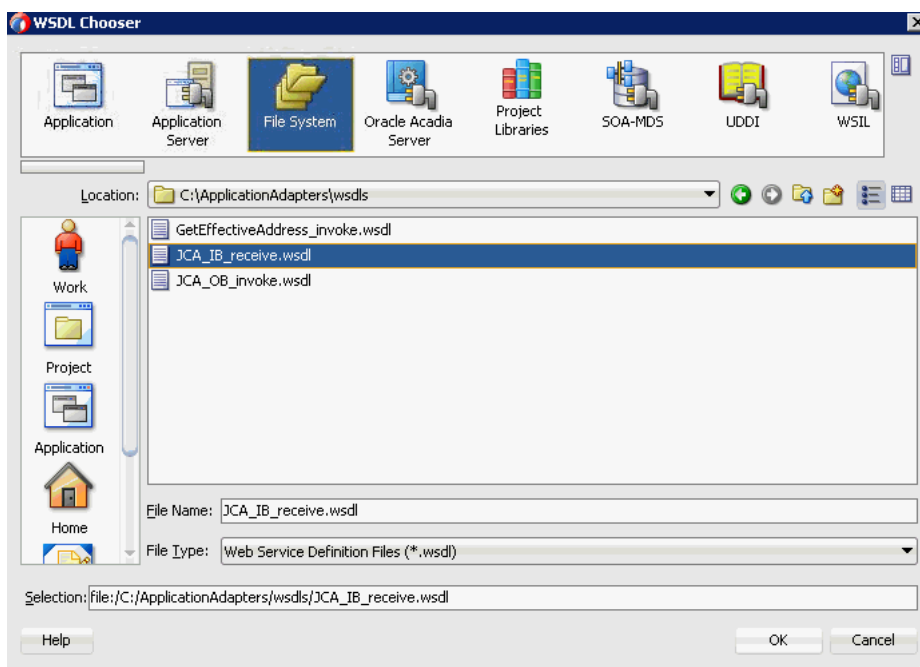
2. サード・パーティ・アダプタ・サービスの任意の名前を入力するか、デフォルト値のままにします。
3. 「タイプ」ドロップダウン・リストで「サービス」が選択されていることを確認します (デフォルト)。
4. 図 8-33 に示すように、「WSDL URL」フィールドの右の既存の WSDL を検索アイコンをクリックします。

図 8-33 「サード・パーティ・アダプタ・サービス」 ダイアログ



図 8-34 に示すように、「WSDL チューザ」ダイアログが表示されます。

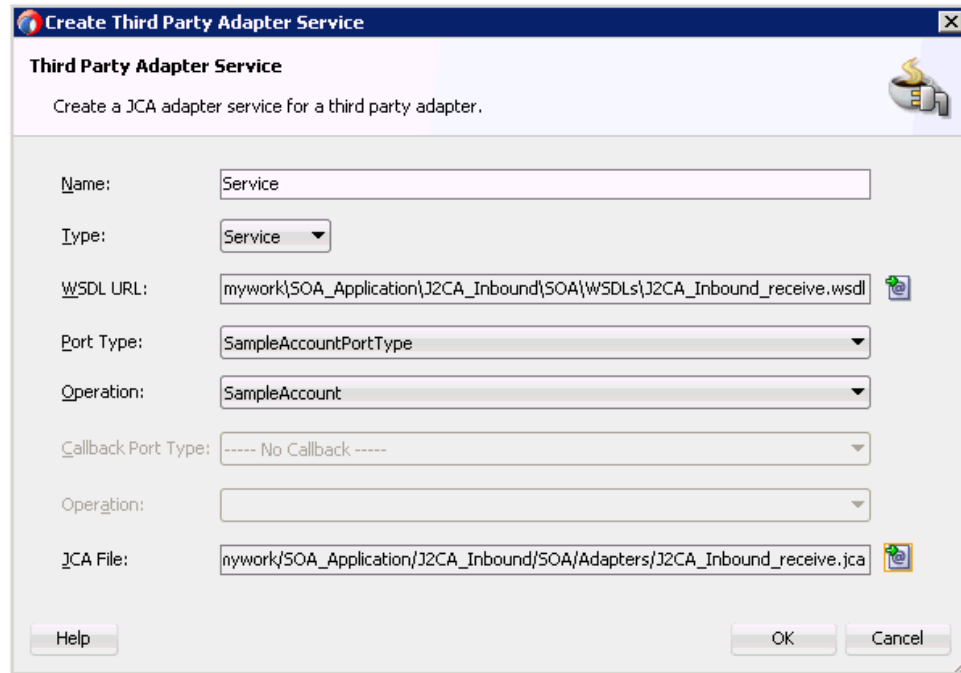
図 8-34 「WSDL チューザ」ダイアログ



5. 「ファイルシステム」フォルダを選択し、WSDL ディレクトリのインバウンド WSDL ファイルを参照して選択します。
6. 「OK」をクリックします。
Service Bus リソースのインポート・ダイアログが表示されます。
7. 「次へ」をクリックします。

8. 「構成」 ウィンドウで「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。
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。
「XQuery 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. 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。
「ファイルのコピー」 メッセージ表示されます。
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。
図 8-35 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 8-35 「サード・パーティ・アダプタ・サービスの作成」 ダイアログ



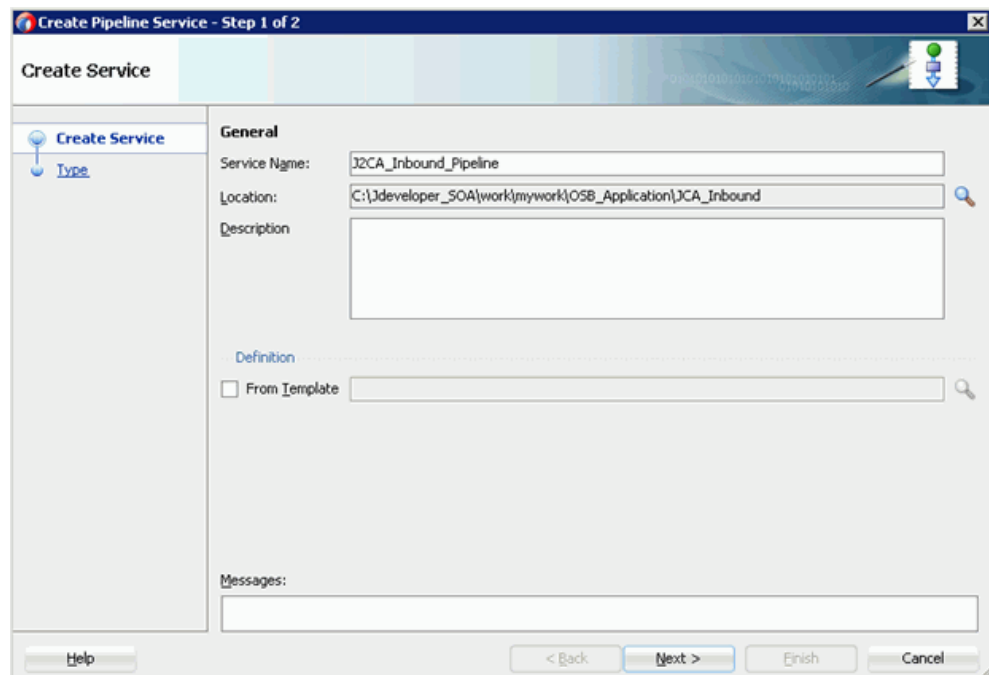
13. 「OK」をクリックします。
「プロキシ・サービス」 ペインにサード・パーティ・アダプタ・サービス・コンポーネントが作成されます。

8.2.2.2 パイプラインの作成

パイプラインを備えたインバウンド・プロキシ・サービスを生成するには、次のステップを実行します。

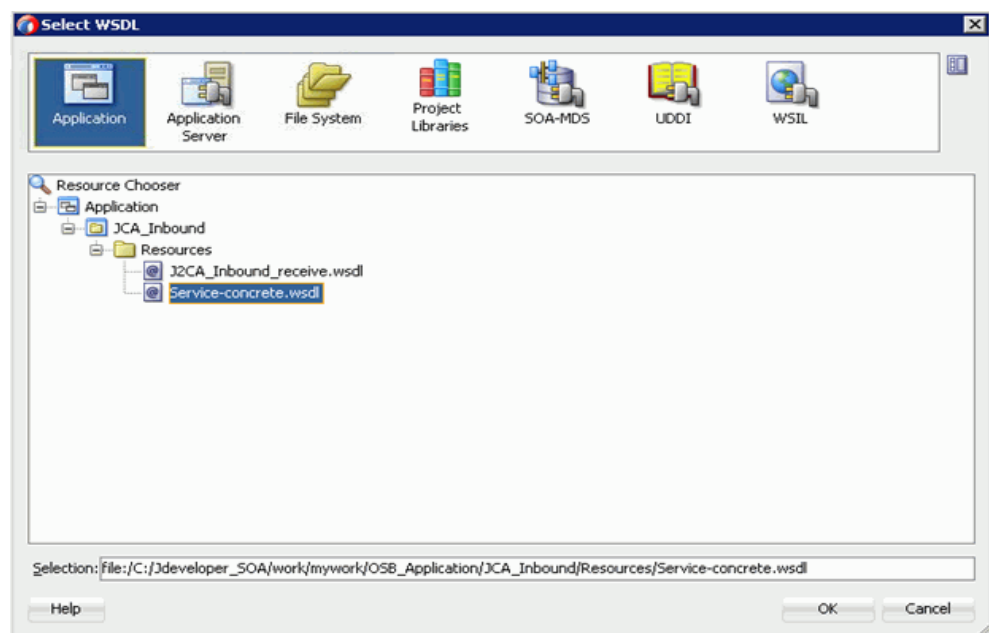
1. 「Service Bus」 の下の 「リソース」 をクリックします。
2. 「パイプラインペインにドラッグ・アンド・ドロップします。
3. 図 8-36 に示すように、パイプラインの名前を入力し、「次へ」をクリックします。

図 8-36 「サービスの作成」 ページ



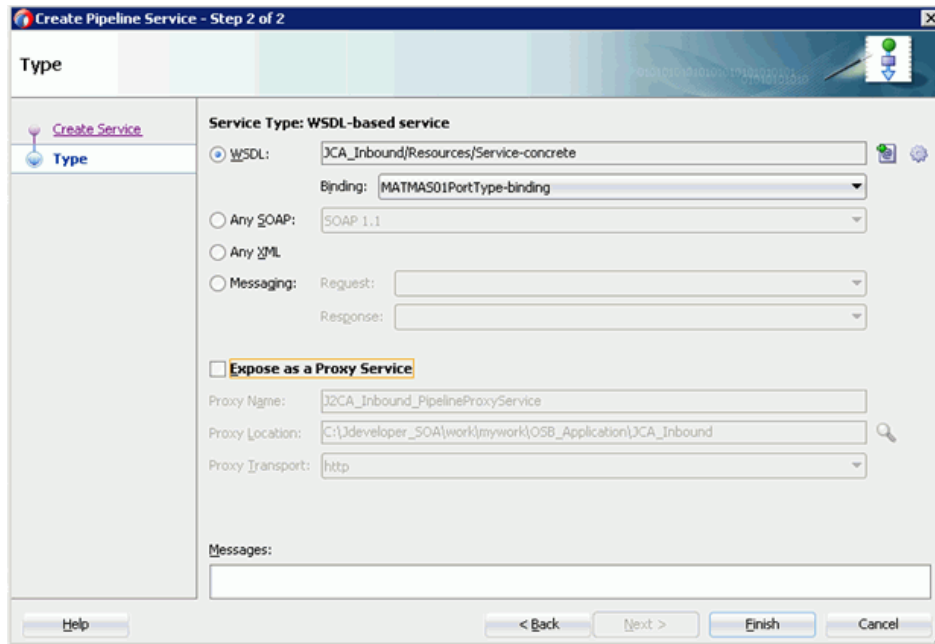
4. パイプライン・サービスの作成ウィンドウで、「WSDL」 オプションを選択して「WSDL URL」をクリックします。
5. 図 8-37 に示すように、「WSDL チューザ」ウィンドウで「アプリケーション」を選択し、適切な OSB プロジェクトの「service-concrete.wsdl」を選択して「OK」をクリックします。

図 8-37 「WSDL の選択」 ページ



6. 図 8-38 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、「終了」をクリックします。

図 8-38 「タイプ」 ページ



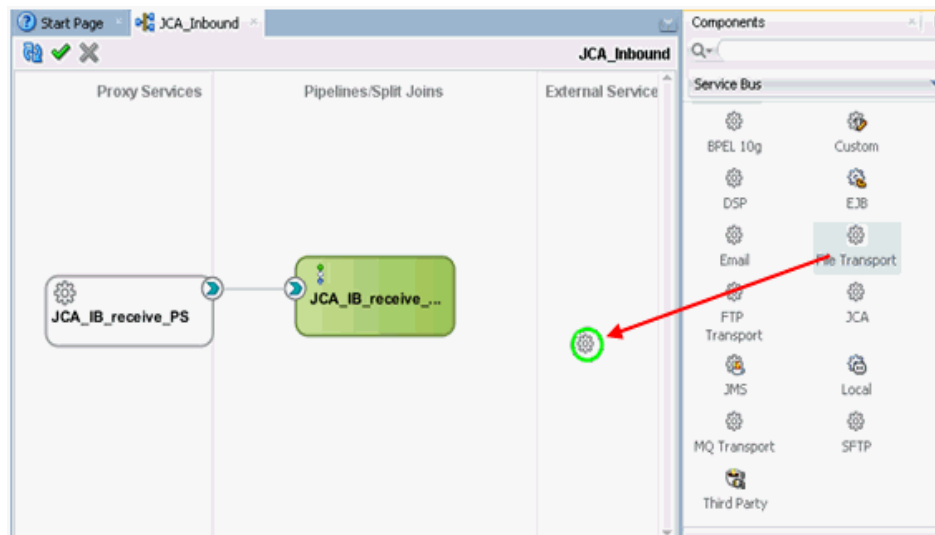
7. 「プロキシ・サービスペインにドラッグ・アンド・ドロップします。

8.2.2.3 ファイル・トランスポート・タイプ・ビジネス・サービスの構成

ファイル・トランスポート・タイプ・ビジネス・サービスを作成するには、次のステップを実行します。

1. 図 8-39 に示すように、「拡張」ペインからファイル・トランスポート・コンポーネントを外部サービス・ペインにドラッグ・アンド・ドロップします。

図 8-39 ファイル・トランスポート・ノード



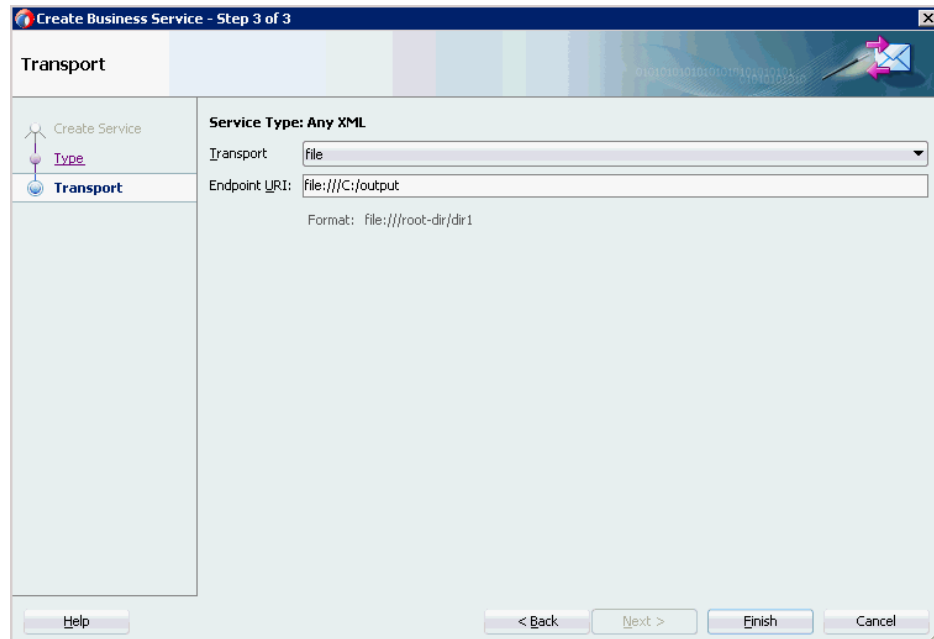
「ビジネス・サービス」ダイアログが表示されます。

2. 「サービス名」フィールドに、任意のビジネス・サービスの名前 (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」ウィンドウで、「任意の XML」オプションがデフォルトで選択されています。

3. 「次へ」をクリックします。
4. 図 8-40 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を入力し(「c:\output」など)、「終了」をクリックします。

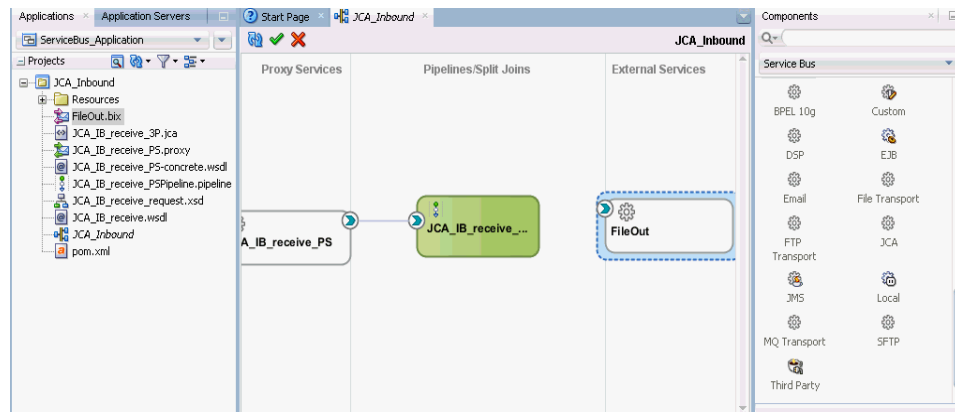
図 8-40 「トランスポート」 ペイン



FileOut ビジネス・サービスが作成されます。

5. 図 8-41 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。

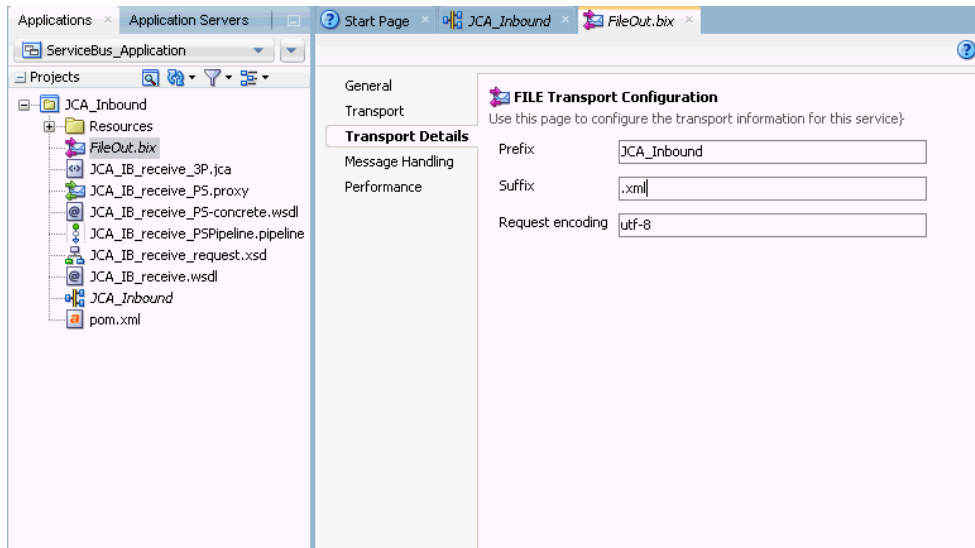
図 8-41 FileOut ビジネス・サービス



「構成」 ページが表示されます。

6. 図 8-42 に示すように、「トランスポートの詳細」タブに移動し、「接頭辞」フィールドに値を入力します。

図 8-42 FILE トランスポート構成



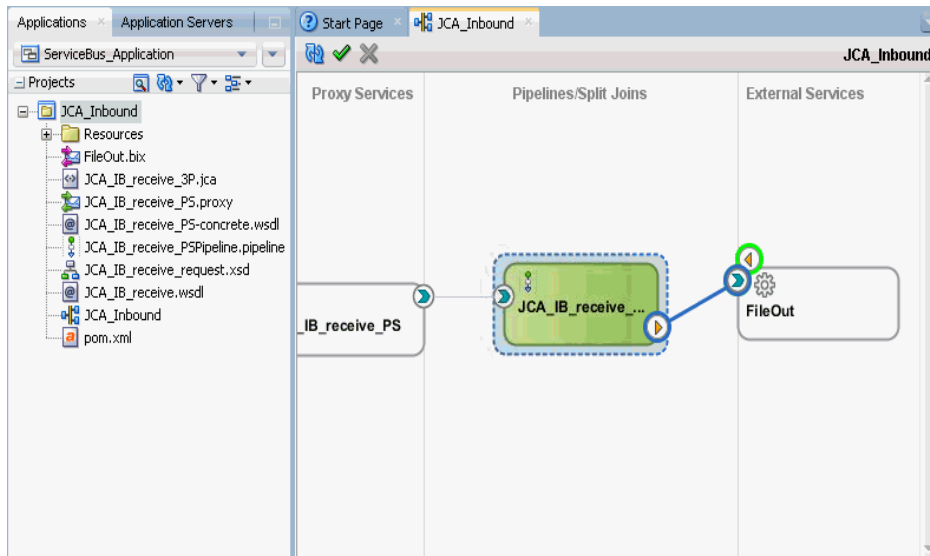
7. 「構成」 ページを保存して閉じます。

8.2.2.4 ルーティング・ルール構成

ルーティング・ルールを構成するには、次のステップを実行します。

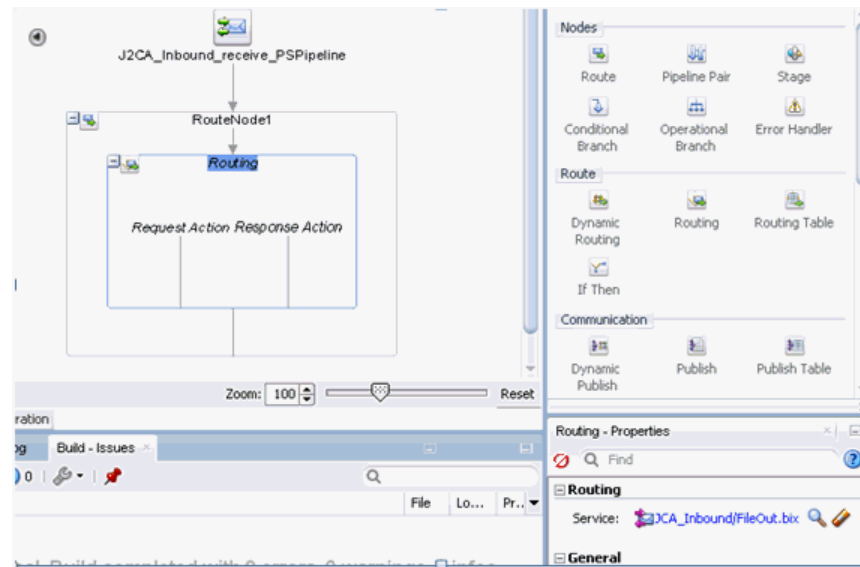
1. 図 8-43 に示すように、パイプライン (JCA_IB_receive_PSPipeline など) とファイル・タイプ・ビジネス・サービス (FileOut など) の間の接続を作成します。

図 8-43 プロキシと FileOut のマッピング



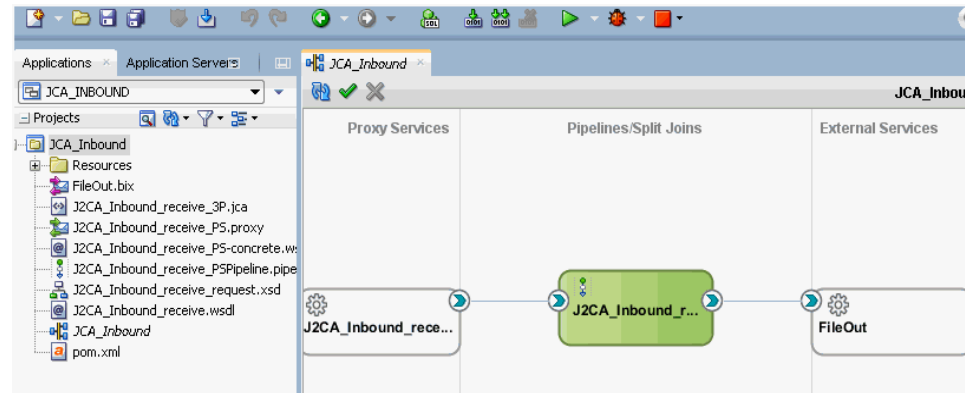
2. パイプライン (JCA_IB_receive_PSPipeline など) をダブルクリックします。
3. 図 8-44 に示すように、「ルーティング」 フィールドでファイル・タイプ・ビジネス・サービス (FileOut など) が適切に構成されていることを確認します。

図 8-44 「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 8-45 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 8-45 すべて保存



8.2.3 OSB インバウンド処理のデプロイ

作成した OSB インバウンド処理をデプロイするには、8-17 ページ 8.1.3 項「OSB アウトバウンド処理のデプロイ」のステップ 1 - 4 を参照してください。

OSB インバウンド処理が正常にデプロイされたら、Siebel システムからイベントをトリガーし、構成された出力場所 (C:\output など) で出力が受信されたかを確認します。

イベントのトリガーの詳細は、4-49 ページ 4.5.5 項「Siebel でのイベントのトリガー」を参照してください。

8.3 JDeveloper を使用した OSB アウトバウンド処理の構成 (BSE 構成)

この項では、BSE 構成に Oracle JDeveloper を使用して、Siebel システムに OSB 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。

このアウトバウンド・ユースケース・シナリオ用のサンプル・プロジェクトは、Application Adapters インストレーションの次のフォルダに提供されています。

```
<ADAPTER_HOME>\etc\sample\SIEBEL_Samples.zip\SIEBEL_Samples\OSB_JDeveloper\BSE\Siebel_Sample_BSE_OSB_Outbound_Project
```

この項では、次のトピックについて説明します。

- [8.3.1 項「OSB 用の Service Bus アプリケーションの作成」](#)
- [8.3.2 項「OSB アウトバウンド処理の定義」](#)
- [8.3.3 項「OSB アウトバウンド処理のデプロイ」](#)

前提条件

OSB アウトバウンド処理を設計する前に、アプリケーション・エクスプローラを使用して対応する WSDL ファイルを生成する必要があります。詳細は、4-76 ページ [4.6.1 項「Web サービスを使用するリクエスト/レスポンス型サービス用の WSDL ファイルの生成」](#) を参照してください。

8.3.1 OSB 用の Service Bus アプリケーションの作成

OSB アウトバウンド処理を構成するには、OSB 用の Service Bus アプリケーションを作成する必要があります。詳細は、8-2 ページ [8.1.1 項「OSB 用の Service Bus アプリケーションの作成」](#) を参照してください。

8.3.2 OSB アウトバウンド処理の定義

この項では、OSB アウトバウンド処理の定義方法について説明します。内容は、次のとおりです。

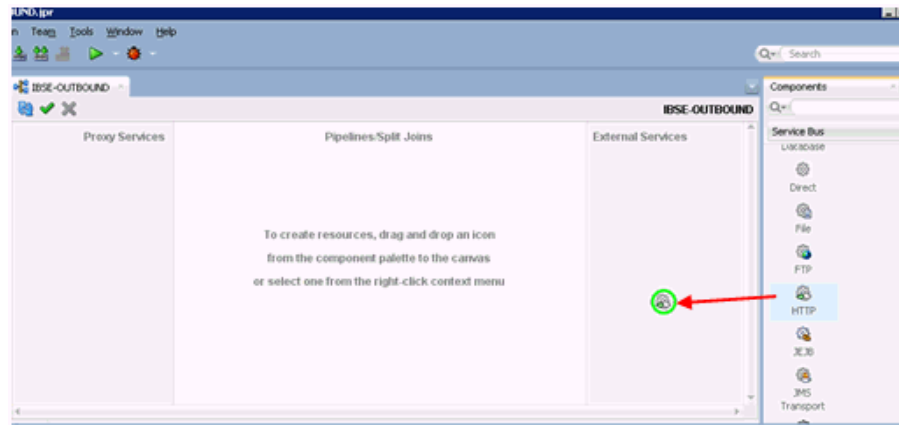
- [8.3.2.1 項「WSDL ベースのビジネス・サービスの構成」](#)
- [8.3.2.2 項「パイプラインを備えたプロキシ・サービスの作成」](#)
- [8.3.2.3 項「ファイル・トランスポート・タイプ・ビジネス・サービスの構成」](#)
- [8.3.2.4 項「ルーティング・ルールの構成」](#)

8.3.2.1 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、次のステップを実行します。

1. [図 8-46](#) に示すように、テクノロジ・コンポーネント・ペインから「HTTP」コンポーネントを外部サービス領域にドラッグ・アンド・ドロップします。

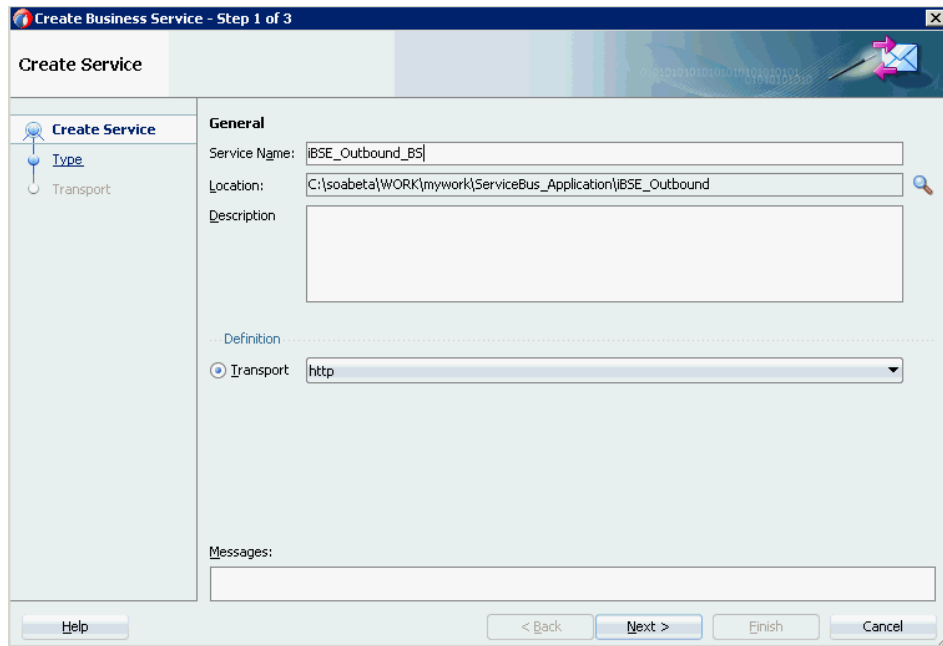
図 8-46 HTTP コンポーネント



「ビジネス・サービス」ウィンドウが表示されます。

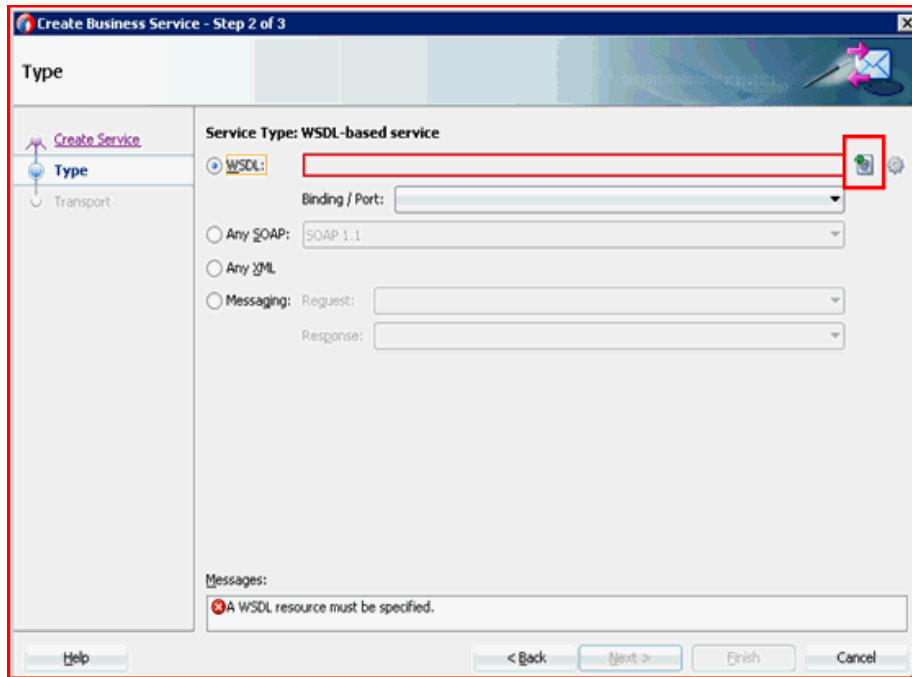
2. 図 8-47 に示すように、「サービス名」フィールドに、任意のビジネス・サービスの名前を入力し、「次へ」をクリックします。

図 8-47 ビジネス・サービス



3. 図 8-48 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。

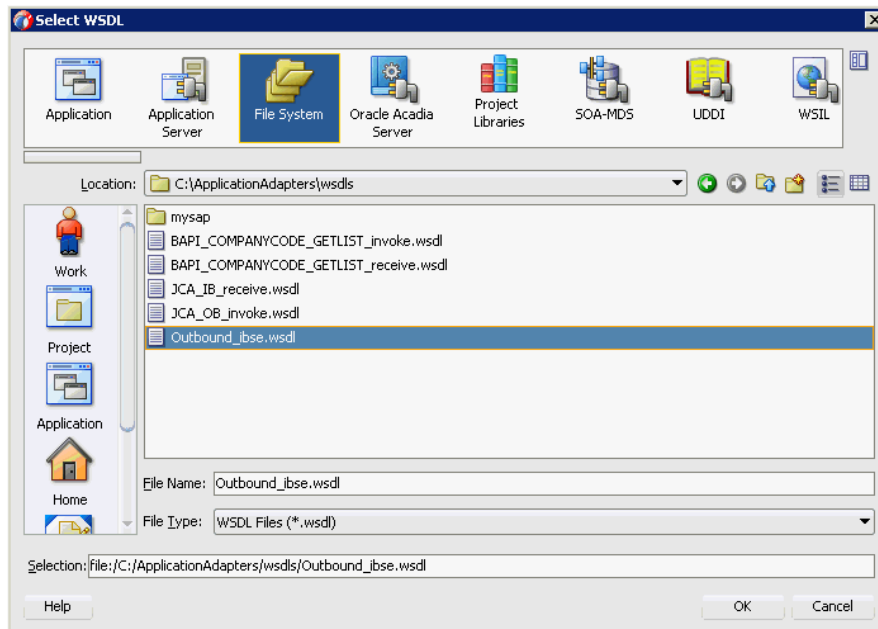
図 8-48 「タイプ」 ペイン



「WSDL の選択」 ウィンドウが表示されます。

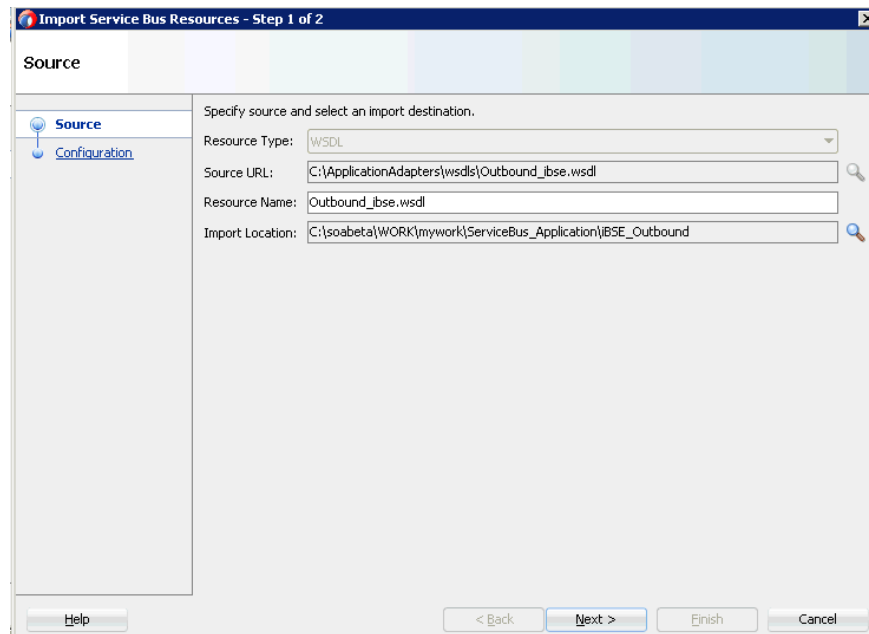
- 図 8-49 に示すように、「ファイルシステム」フォルダ・アイコンを選択し、iBSE WSDL ファイルを参照し、それを WSDL の場所から選択して「OK」をクリックします。

図 8-49 「WSDL の選択」 ウィンドウ



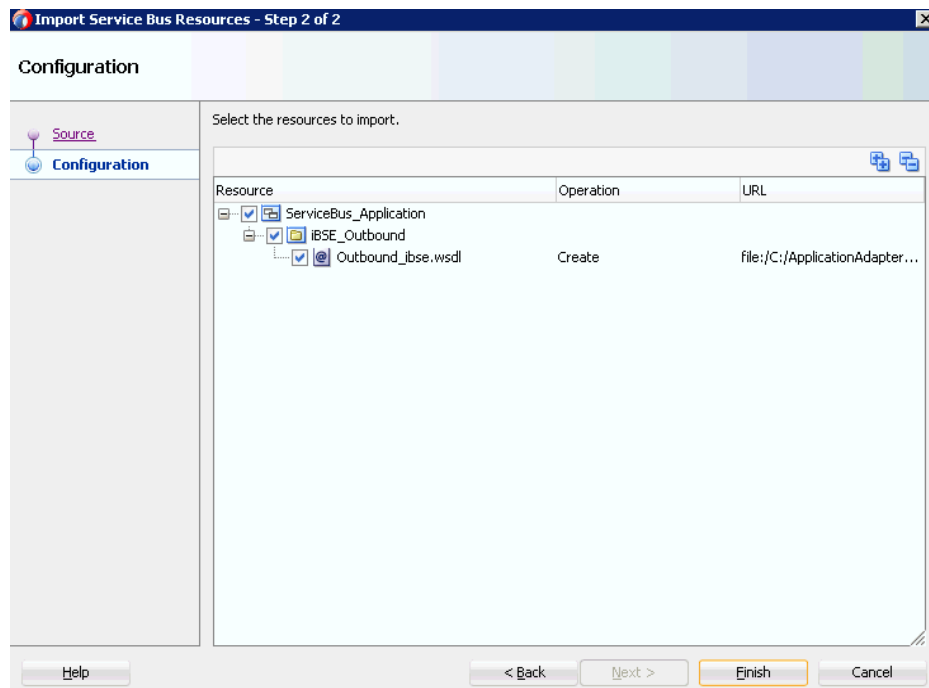
- 図 8-50 に示すように、表示される「ソース」ペインで「次へ」をクリックします。

図 8-50 「ソース」 ペイン



6. 図 8-51 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。

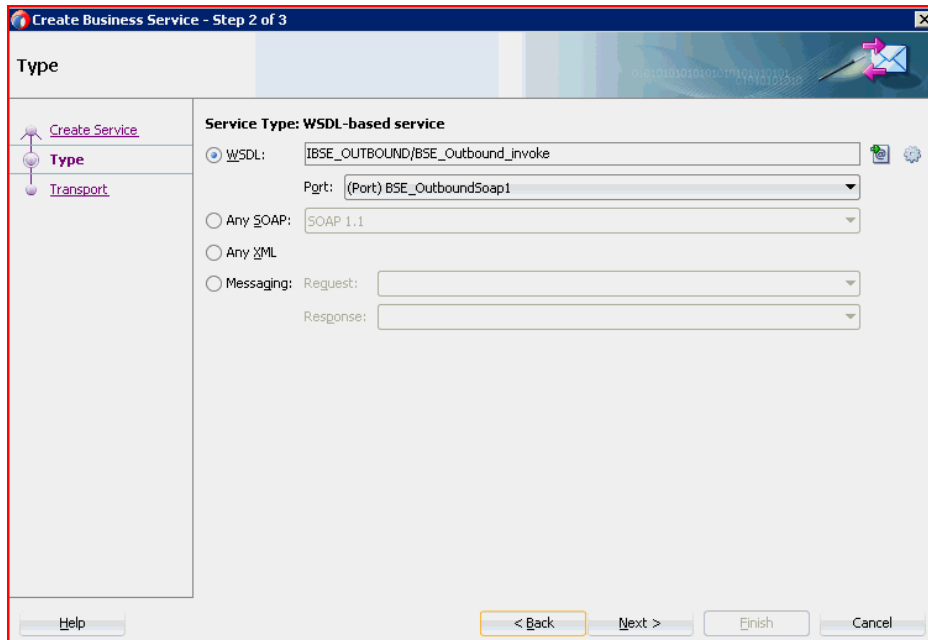
図 8-51 「構成」 ペイン



「ビジネス・サービス」ウィンドウに戻ります。

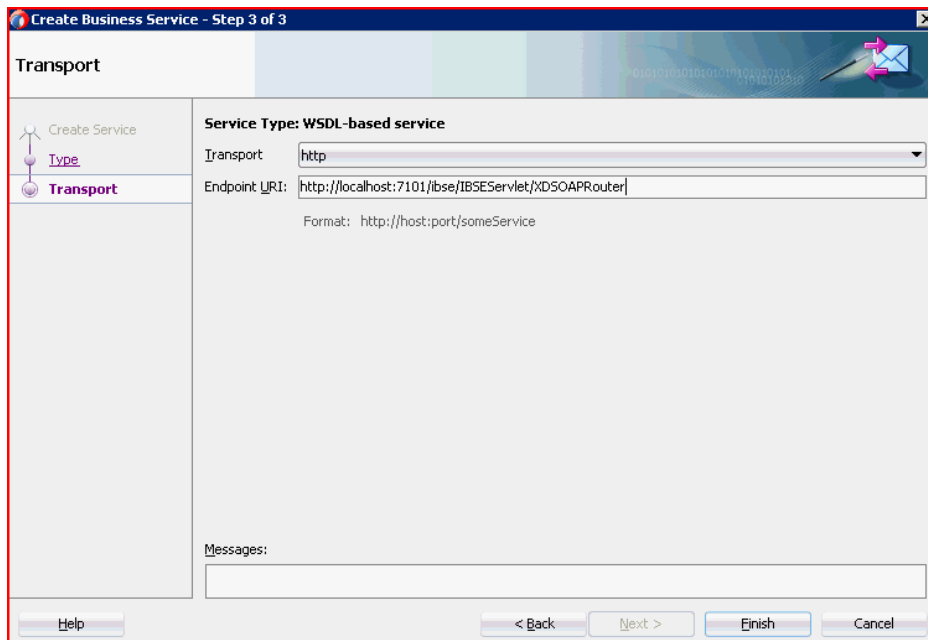
7. 図 8-52 に示すように、表示される「タイプ」ペインで「次へ」をクリックします。

図 8-52 「タイプ」 ペイン



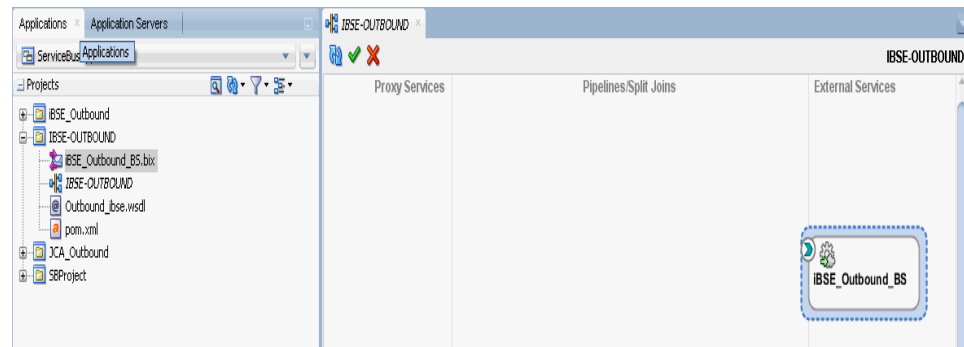
8. 図 8-53 に示すように、表示される「トランスポート」ウィンドウで、ホスト名とポート番号が異なる場合は「エンドポイント URI」フィールドを変更し、「終了」をクリックします。

図 8-53 「トランスポート」 ペイン



- 図 8-54 に示すように、ビジネス・サービスが作成され、外部サービス・ペインに表示されます。

図 8-54 外部サービス・ペイン

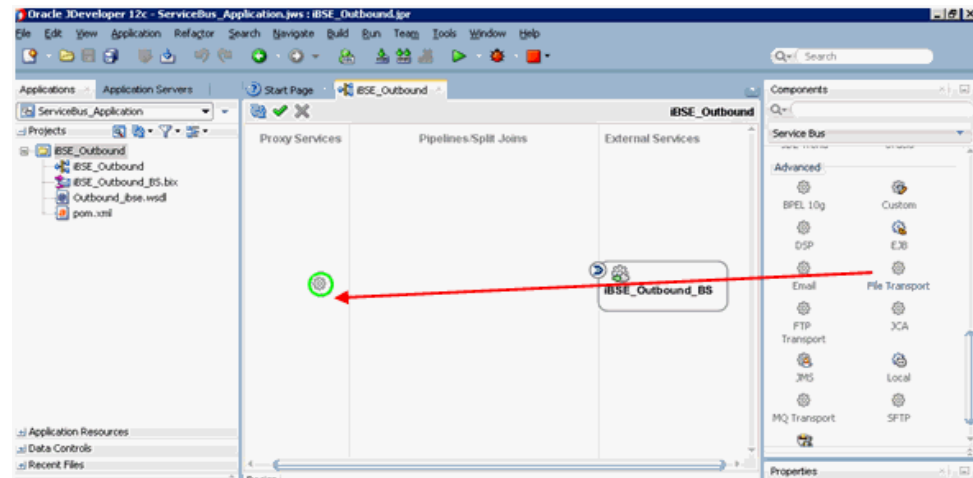


8.3.2.2 パイプラインを備えたプロキシ・サービスの作成

パイプラインを備えたプロキシ・サービスを作成するには、次のステップを実行します。

1. 図 8-55 に示すように、拡張コンポーネント・ペインからファイル・トランスポート・コンポーネントを「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。

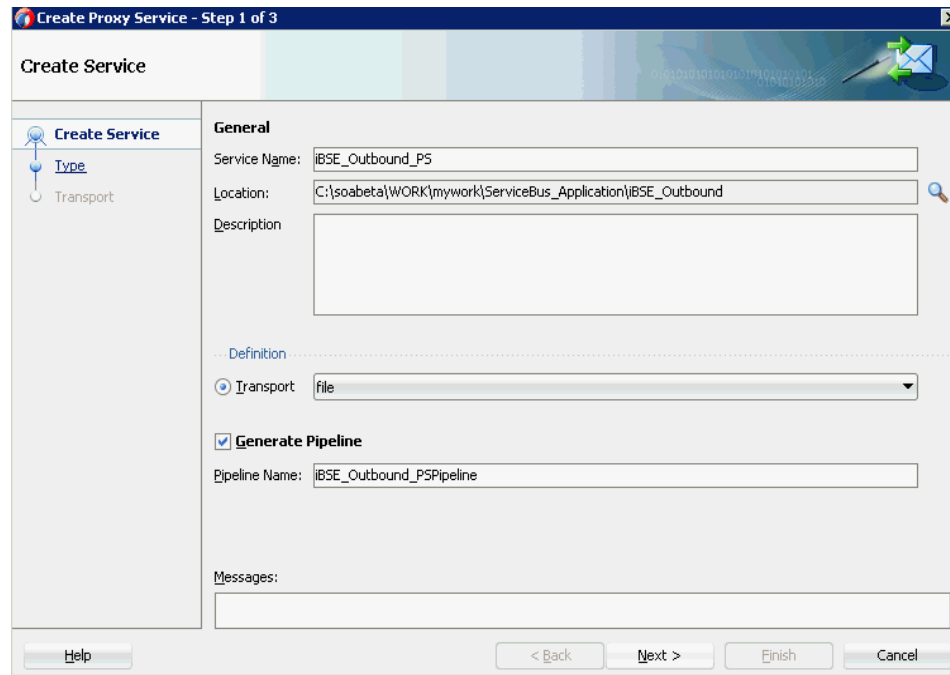
図 8-55 ファイル・トランスポート・コンポーネント



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、任意のプロキシ・サービスの名前 (JCA_Outbound_PS など) を入力します。「パイプラインを生成する」がデフォルトで選択されています。
3. 図 8-56 に示すように、「次へ」をクリックします。

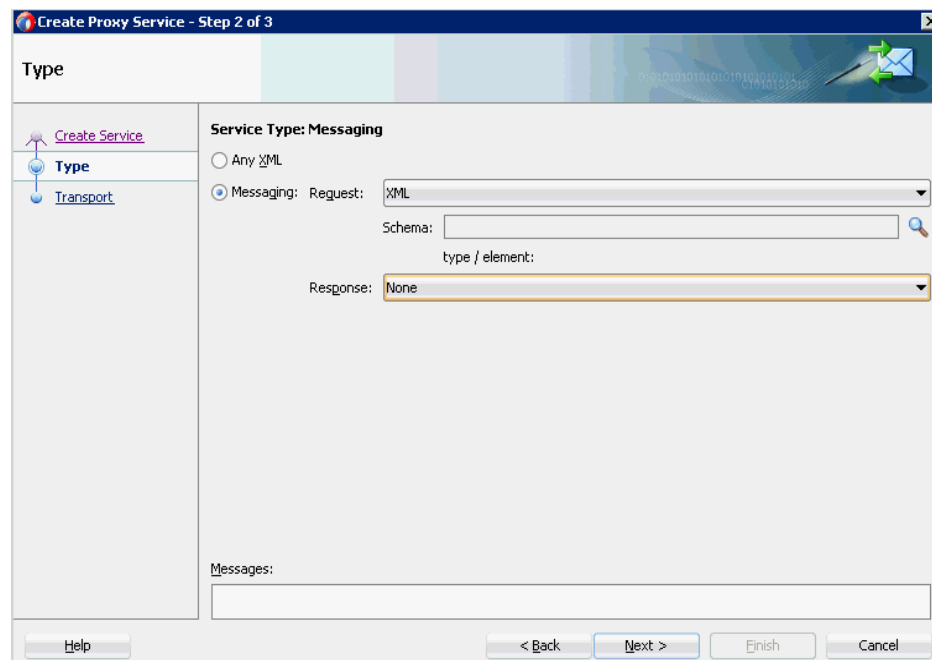
図 8-56 「サービスの作成」 ペイン



「タイプ」 ペインが表示されます。

4. 図 8-57 に示すように、「メッセージ機能」 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「次へ」をクリックします。

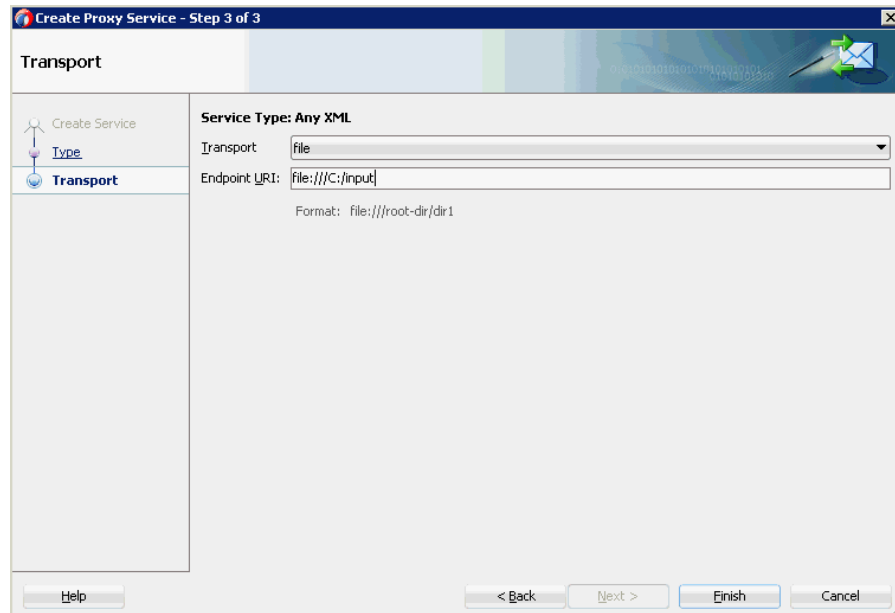
図 8-57 「タイプ」 ペイン



「トランスポート」 ウィンドウが表示されます。

5. 図 8-58 に示すように、「エンドポイント URI」 フィールドに入力場所を指定し（「c:/input」 など）、「終了」をクリックします。

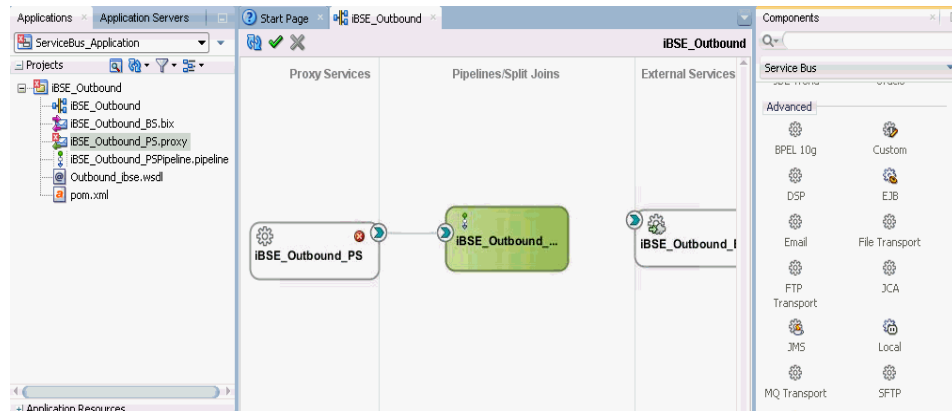
図 8-58 「トランスポート」 ウィンドウ



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。

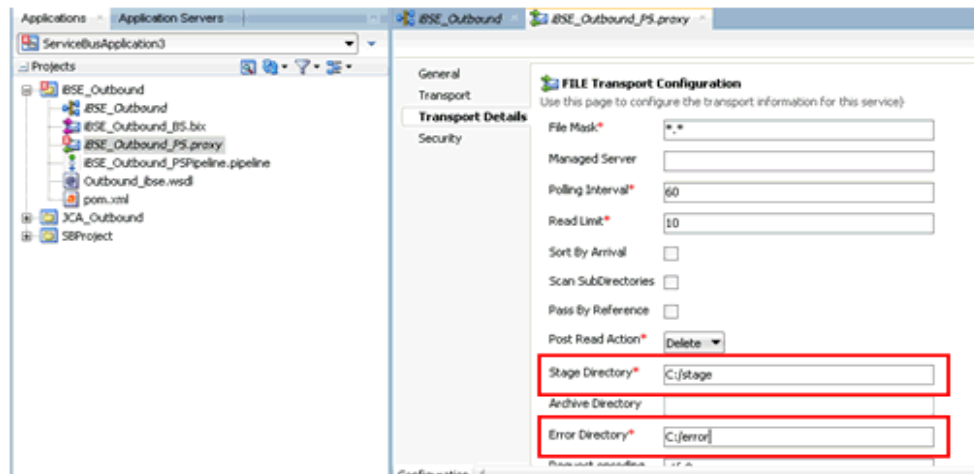
6. 図 8-59 に示すように、作成したプロキシ・サービス (iBSE_Outbound_PS など) をダブルクリックします。

図 8-59 プロキシ・サービスの編集



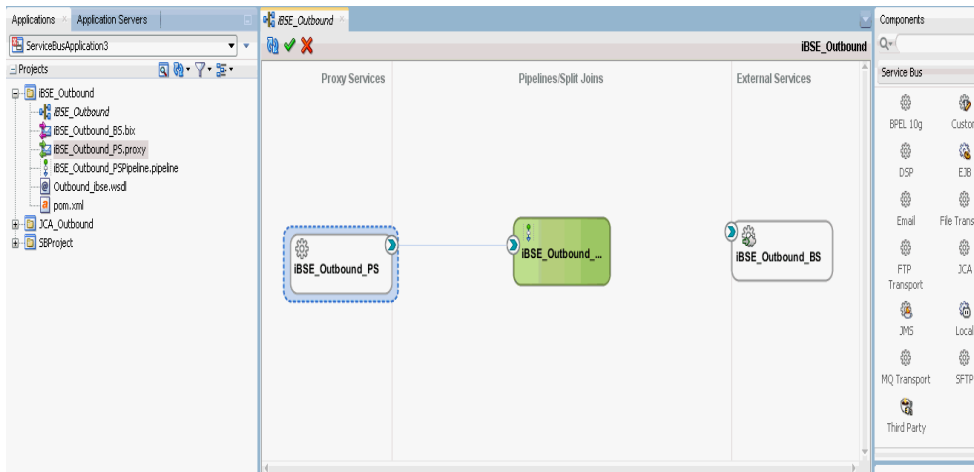
7. 図 8-60 に示すように、表示される「プロキシ・サービスの構成」ページで「トランスポートの詳細」を選択し、「ステージ・ディレクトリ」と「エラー・ディレクトリ」に値を入力します。

図 8-60 FILE トラns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 8-61 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 8-61 プロキシ・サービス

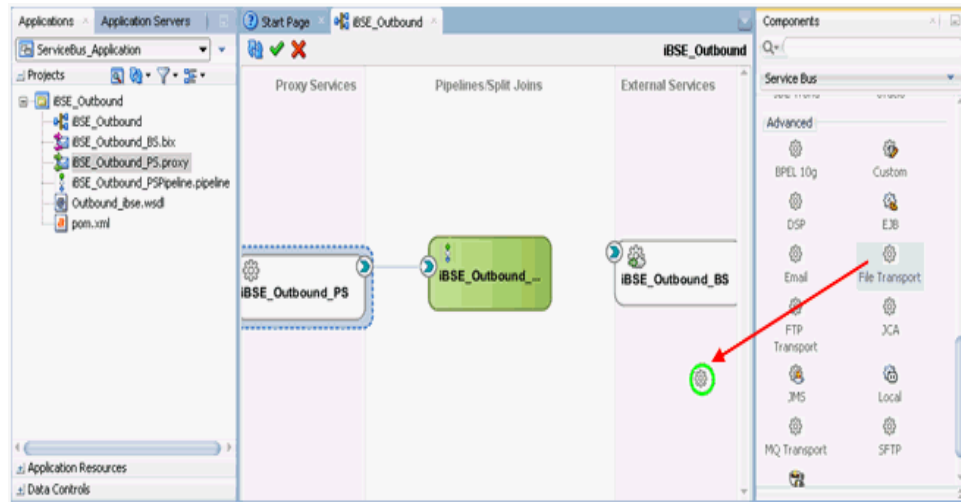


8.3.2.3 ファイル・トラnsポート・タイプ・ビジネス・サービスの構成

ファイル・トラnsポート・タイプ・ビジネス・サービスを作成するには、次のステップを実行します。

1. 図 8-62 に示すように、「拡張」 ペインからファイル・トラnsポート・コンポーネントを外部サービス・ペインにドラッグ・アンド・ドロップします。

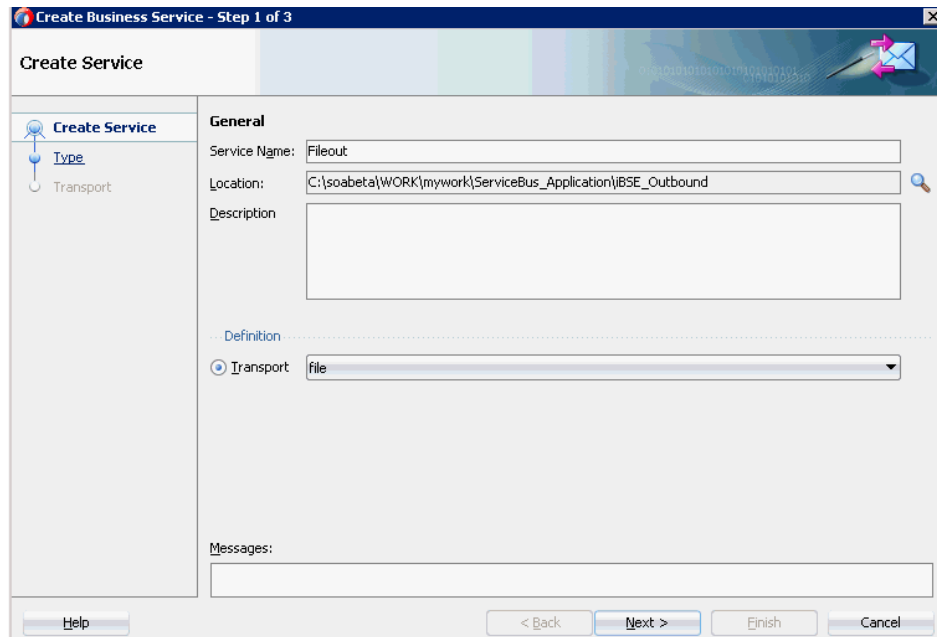
図 8-62 ファイル・トランスポート・コンポーネント



「ビジネス・サービス」ダイアログが表示されます。

2. 図 8-63 に示すように、「サービス名」フィールドに、任意のビジネス・サービスの名前 (FileOut など) を入力し、「次へ」をクリックします。

図 8-63 「サービスの作成」ペイン



「タイプ」ペインが表示されます。「任意の XML」オプションがデフォルトで選択されています。

3. 「次へ」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 8-64 に示すように、「エンドポイント URI」フィールドに出力場所を入力し (「c:/output」など)、「終了」をクリックします。

図 8-64 「トランスポート」 ペイン

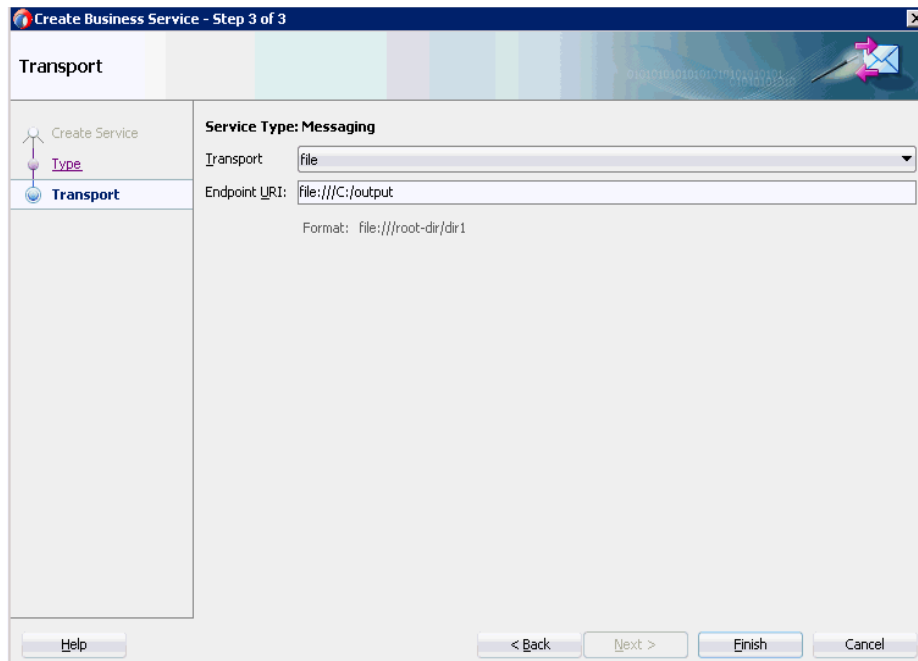
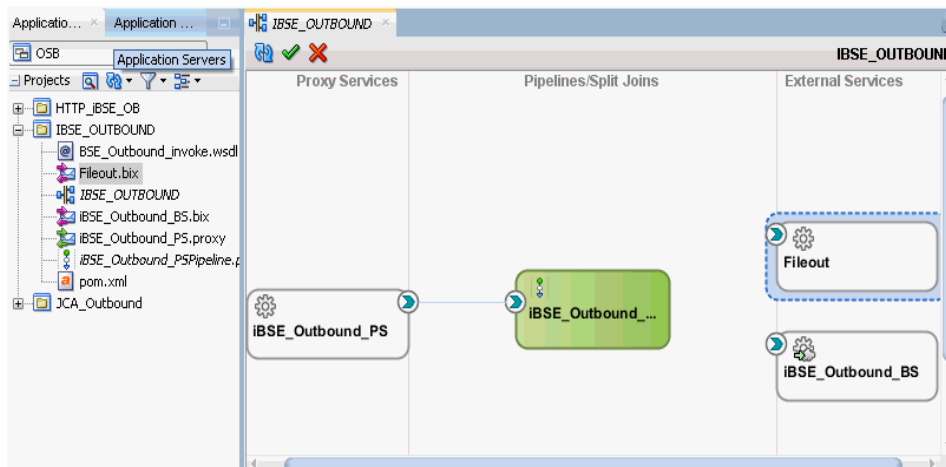


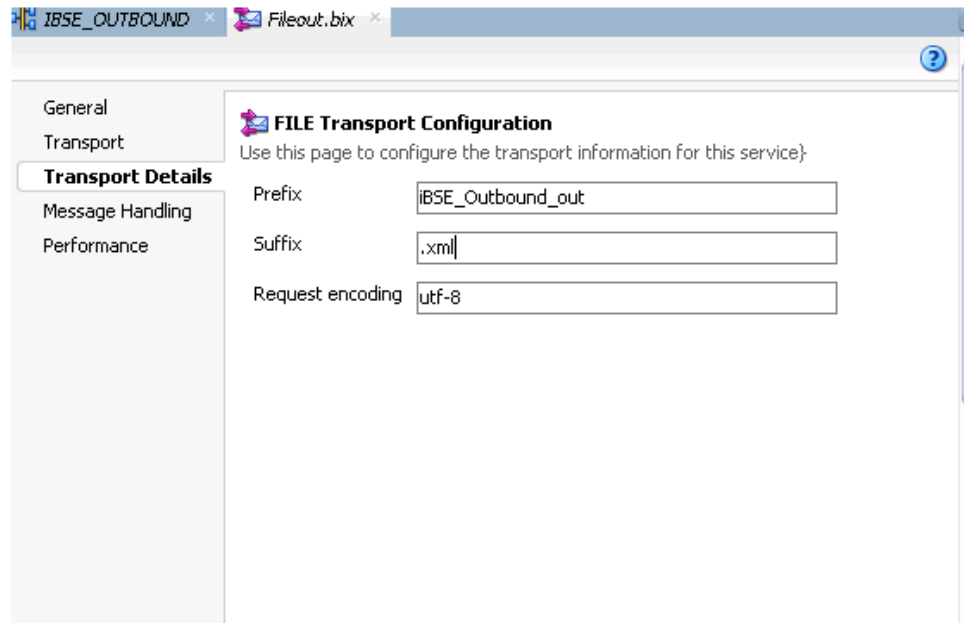
図 8-65 に示すように、ファイル・トランスポート・ビジネス・サービス Fileout が作成されて表示されます。

図 8-65 Fileout ビジネス・サービス



5. 図 8-66 に示すように、作成したビジネス・サービス「Fileout」をダブルクリックし、「トランスポートの詳細」タブの「接頭辞」フィールドに値を入力します。

図 8-66 トランスポートの詳細



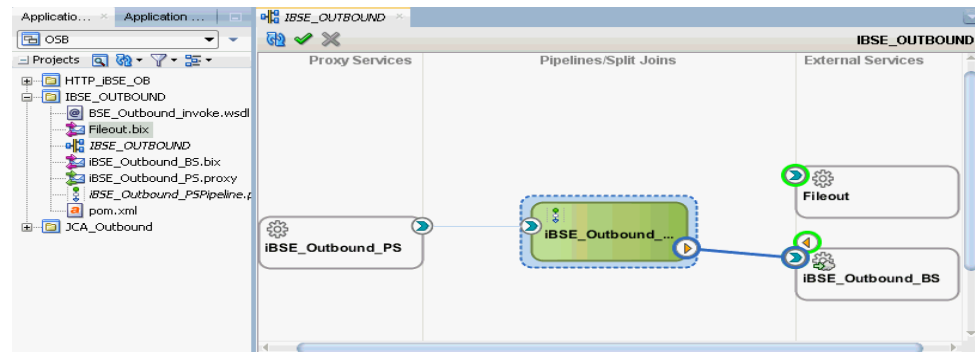
6. 「構成」 ページを保存して閉じ、overview.xml (iBSE_Outbound など) をダブルクリックします。

8.3.2.4 ルーティング・ルール構成

ルーティング・ルールを構成するには、次のステップを実行します。

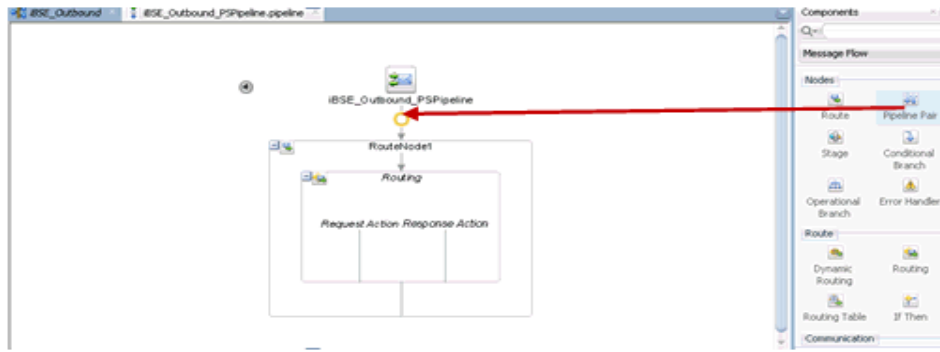
1. 図 8-67 に示すように、パイプライン・コンポーネント (iBSE_Outbound_PSPipeline など) と WSDL ベースのビジネス・サービス (iBSE_Outbound_BS など) の間の接続を作成します。

図 8-67 パイプライン・コンポーネント



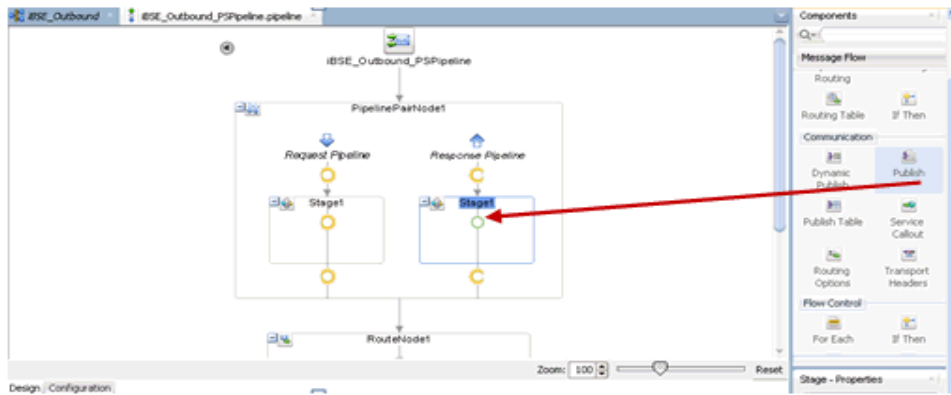
2. パイプライン/分割結合ペインで「パイプライン」コンポーネント (iBSE_Outbound_PSPipeline など) をダブルクリックします。
3. 図 8-68 に示すように、「ノード」ペインからパイプライン・ペア・ノードをパイプライン (iBSE_Outbound_PSPipeline など) と「RouteNode1」の間の領域にドラッグ・アンド・ドロップします。

図 8-68 パイプライン・ペア・ノード



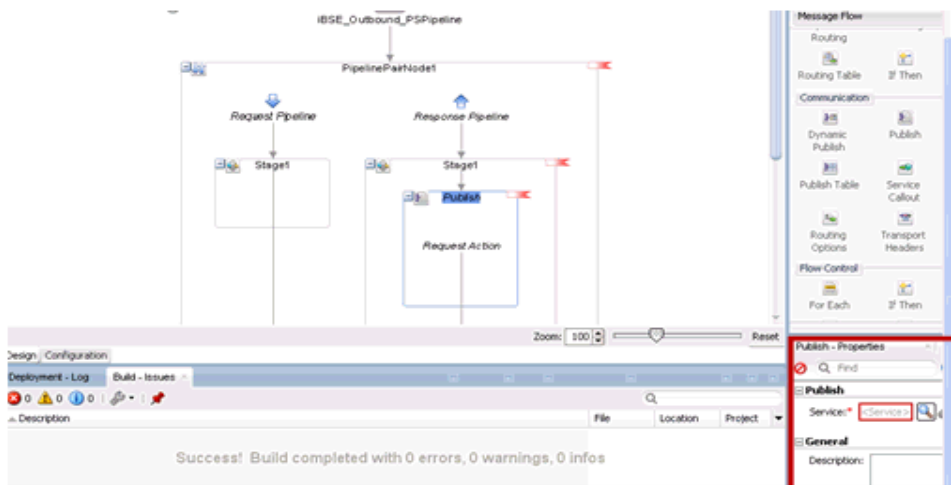
4. 図 8-69 に示すように、「通信」ペインから「公開」ノードを「レスポンス・パイプライン」の「Stage1」の下領域にドラッグ・アンド・ドロップします。

図 8-69 「公開」ノード



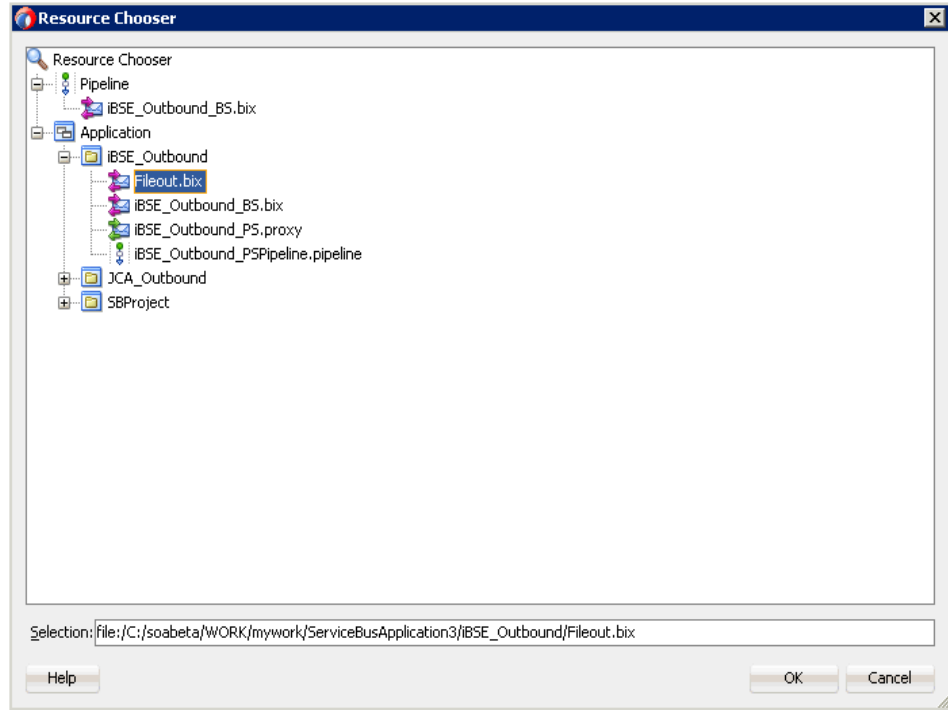
5. 図 8-70 に示すように、「公開」の右ペインにある「サービス」アイコンをクリックします。

図 8-70 「参照」アイコン



- 図 8-71 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。

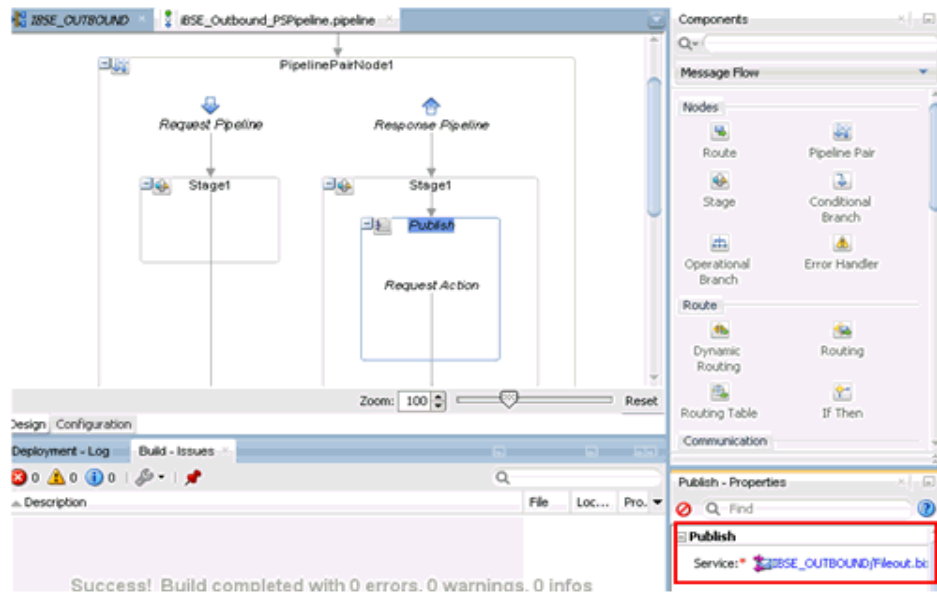
図 8-71 リソース・チューザ



パイプライン構成ページに戻ります。

- 図 8-72 に示すように、右ペインで、選択したサービスが「公開」ペインで構成されます。

図 8-72 「公開」ペイン



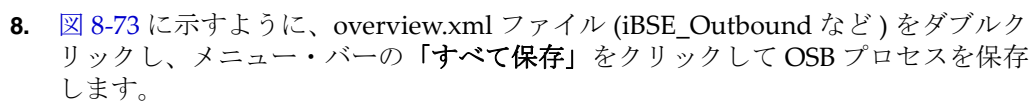
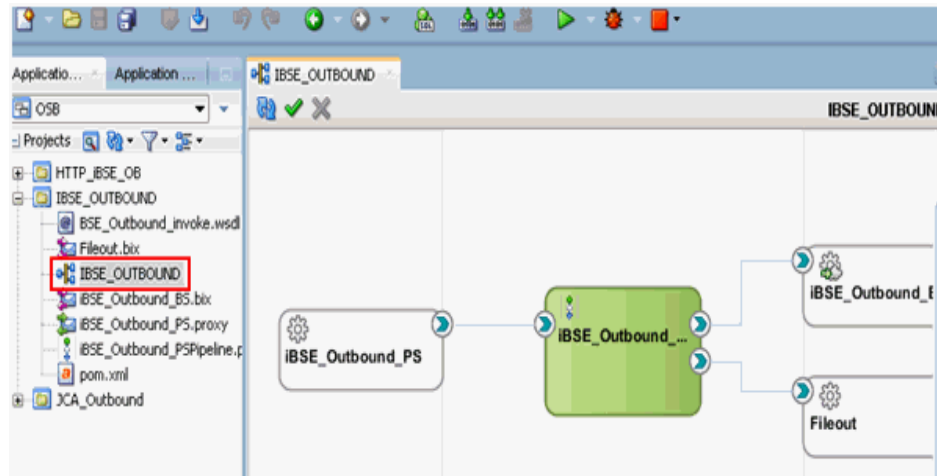
7. パイプライン構成ページを保存して閉じます。
8.  8-73 に示すように、overview.xml ファイル (iBSE_Outbound など) をダブルクリックし、メニュー・バーの「すべて保存」をクリックして OSB プロセスを保存します。

図 8-73 「すべて保存」アイコン



8.3.3 OSB アウトバウンド処理のデプロイ

作成した OSB アウトバウンド処理のデプロイおよび入力 XML ドキュメントの起動を行うには、[8.1.3 項「OSB アウトバウンド処理のデプロイ」](#)を参照してください。

8.4 JDeveloper を使用した JMS インバウンド処理の構成 (J2CA 構成)

この項では、J2CA 構成に Oracle JDeveloper を使用して、Siebel システムに JMS インバウンド処理を構成する方法について説明します。

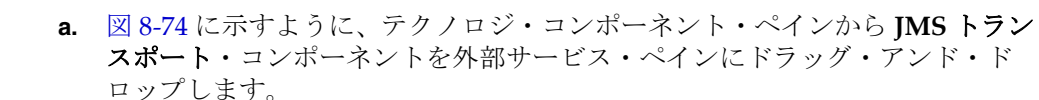
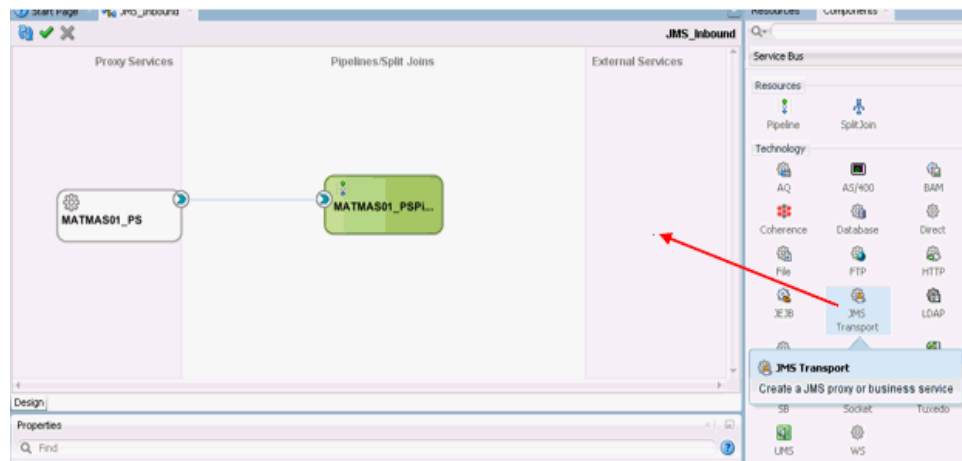
1. JMS プロセスを設計する前に、アプリケーション・エクスプローラを使用して対応する WSDL ファイルを生成する必要があります。詳細は、4-34 ページ [4.5.1 項「イベント統合のための WSDL の生成」](#)を参照してください。
2. Oracle JDeveloper を起動し、OSB 用の Service Bus アプリケーションを作成します。詳細は、8-2 ページ [8.1.1 項「OSB 用の Service Bus アプリケーションの作成」](#)を参照してください。
3. サード・パーティ・アダプタ・サービス・コンポーネントを作成します。詳細は、8-20 ページ [8.2.2.1 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。
4. JCA バインディング・ファイルからプロキシ・サービスとパイプラインを作成します。詳細は、8-22 ページ [8.2.2.2 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、次のステップを実行します。
 - a.  8-74 に示すように、テクノロジー・コンポーネント・ペインから **JMS トランSPORT・コンポーネント**を外部サービス・ペインにドラッグ・アンド・ドロップします。

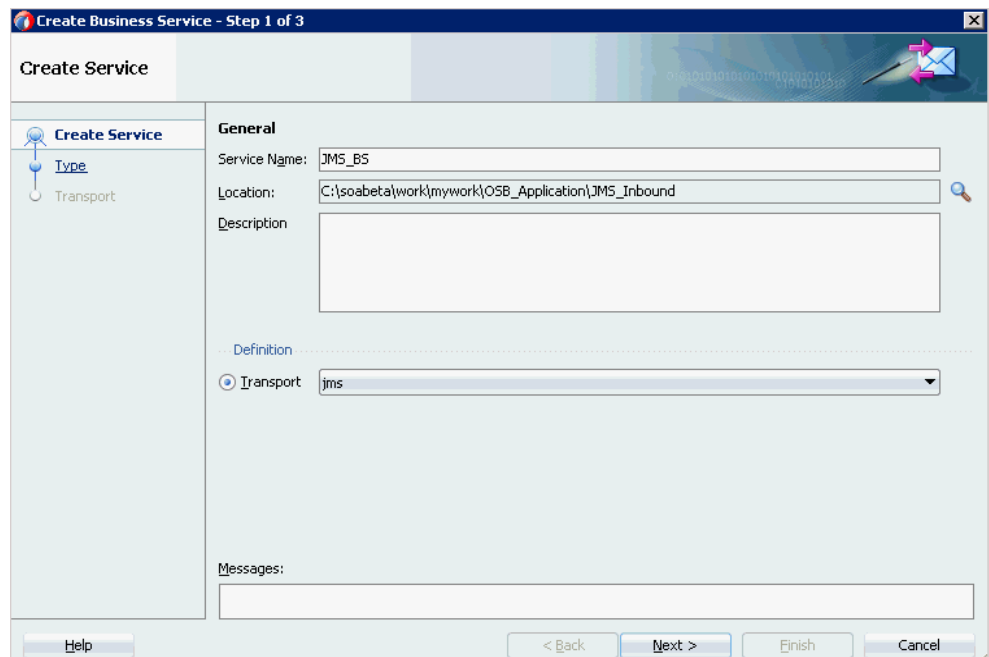
図 8-74 JMS トランスポート・コンポーネント



「ビジネス・サービス」ダイアログが表示されます。

- b. 図 8-75 に示すように、「サービス名」フィールドに、任意のビジネス・サービスの名前 (JMS_BS など) を入力し、「次へ」をクリックします。

図 8-75 「サービスの作成」ペイン

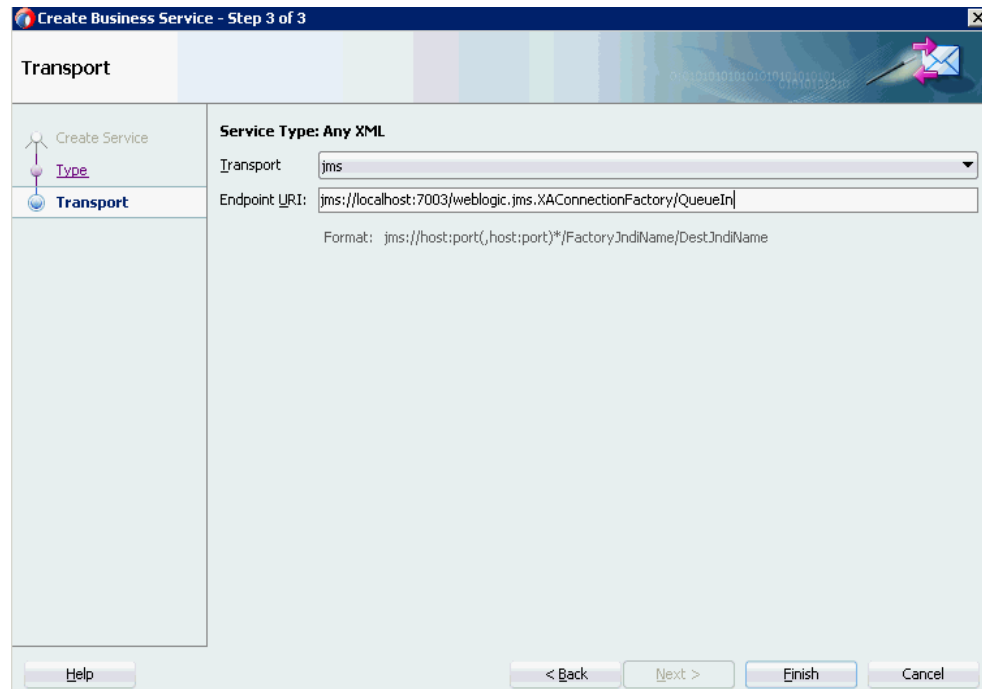


- c. 表示される「タイプ」ウィンドウで、「任意の XML」を選択して「次へ」をクリックします。

「トランスポート」ウィンドウが表示されます。

- d. 図 8-76 に示すように、「エンドポイント URI」フィールドで DestJndiName を QueueIn に置き換えることで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。

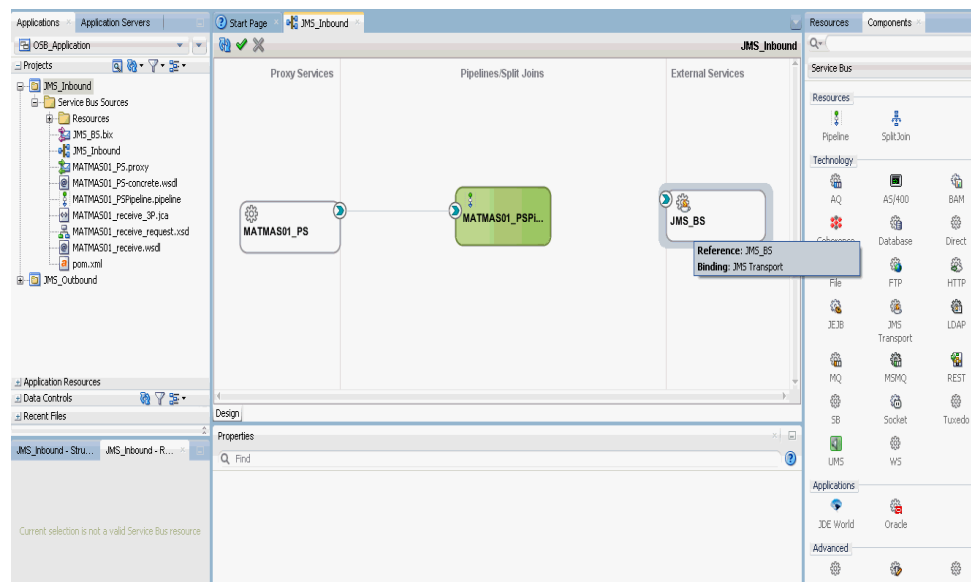
図 8-76 「トランスポート」 ウィンドウ



JMS ビジネス・サービスが作成されて表示されます。

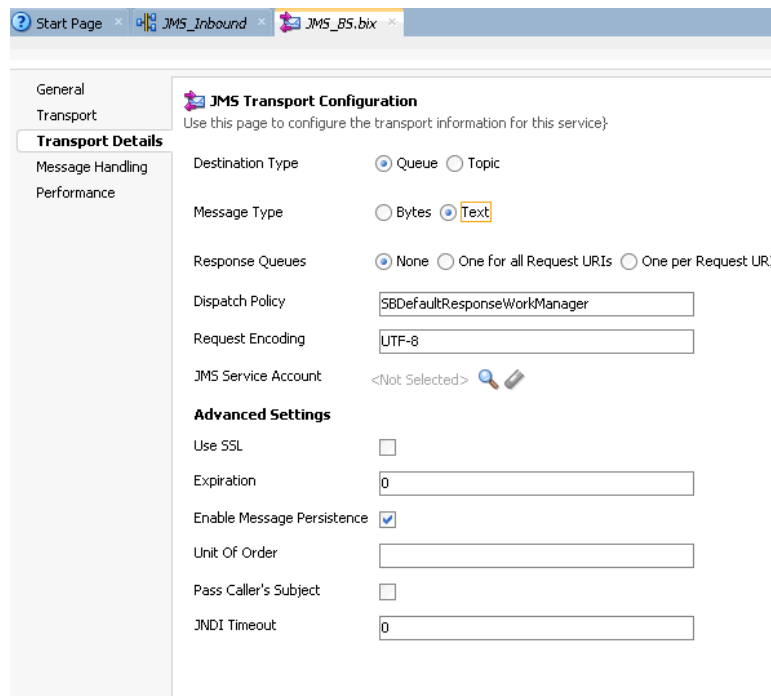
- e. 図 8-77 に示すように、「JMS_BS」をダブルクリックします。

図 8-77 JMS ビジネス・サービス



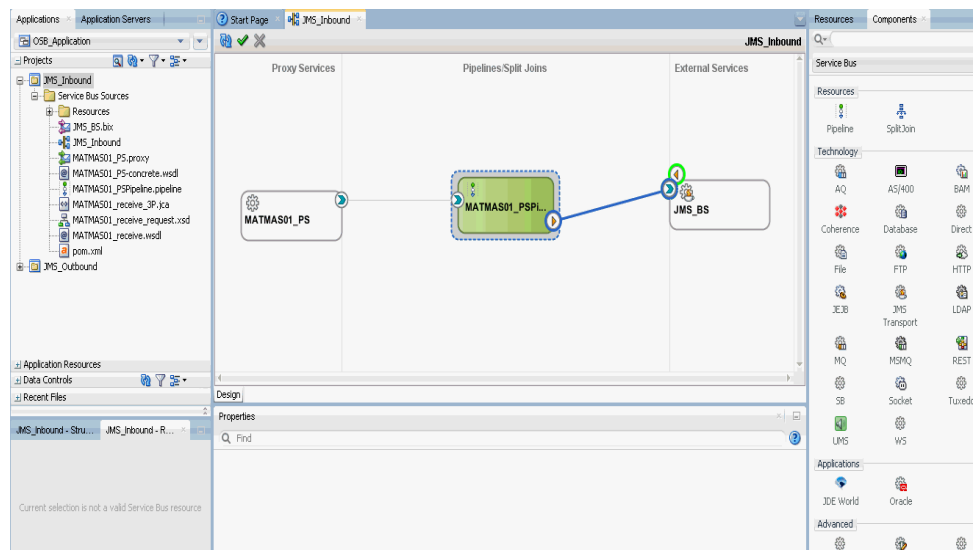
- f. 図 8-78 に示すように、表示されるビジネス・サービスの構成ページで、「トランスポートの詳細」タブに次のパラメータを入力します。

図 8-78 JMS トランスポート構成



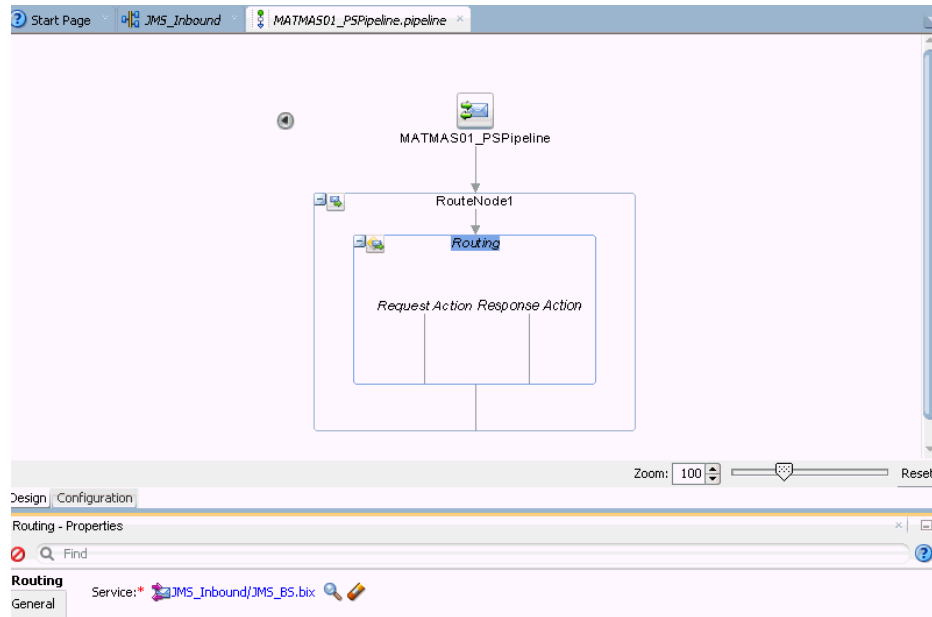
- g. 「接続先タイプ」セクションで「キュー」を選択します。
- h. 「メッセージ・タイプ」セクションで「テキスト」を選択します。
6. ビジネス・サービスの「構成」ページを保存して閉じます。
7. 図 8-79 に示すように、パイプライン (xxxx_PSPipeline など) と JMS ビジネス・サービス (JMS_BS など) の間の接続を作成します。

図 8-79 「構成」ページ



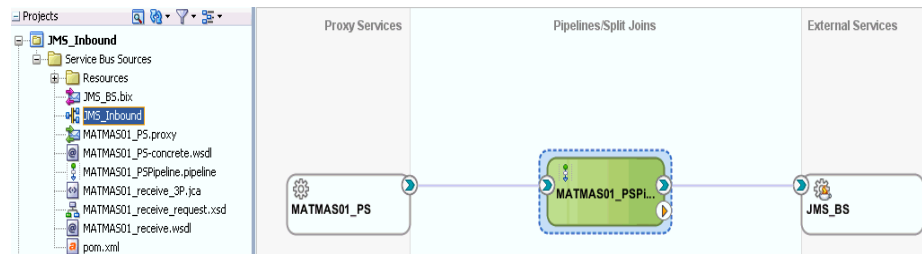
8. パイプラインをダブルクリックします。
 図 8-80 に示すように、パイプライン構成ページが表示されます。

図 8-80 パイプライン構成



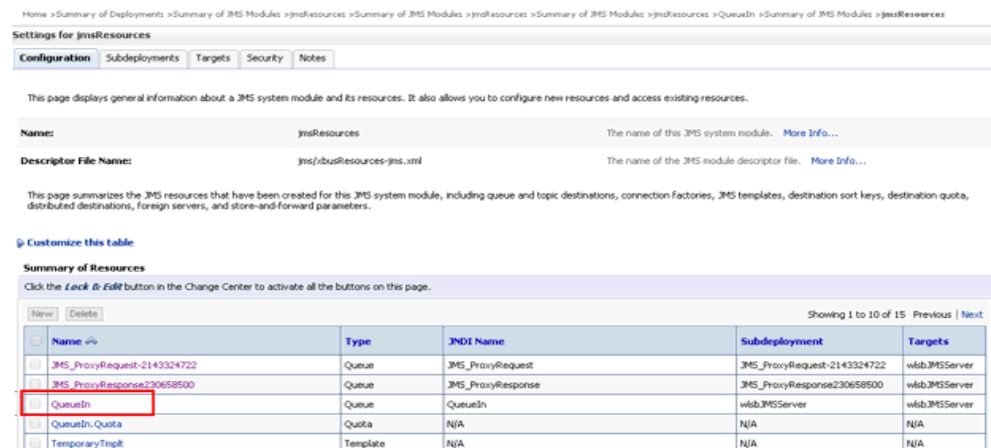
9. 詳細内容が適切に構成されていることを確認し、パイプライン構成ページを保存して閉じます。
コンポジット・エディタ・ウィンドウに戻ります。
10. 図 8-81 に示すように、メニュー・バーの「すべて保存」をクリックし、OSB JMS プロセスを保存します。

図 8-81 「すべて保存」アイコン



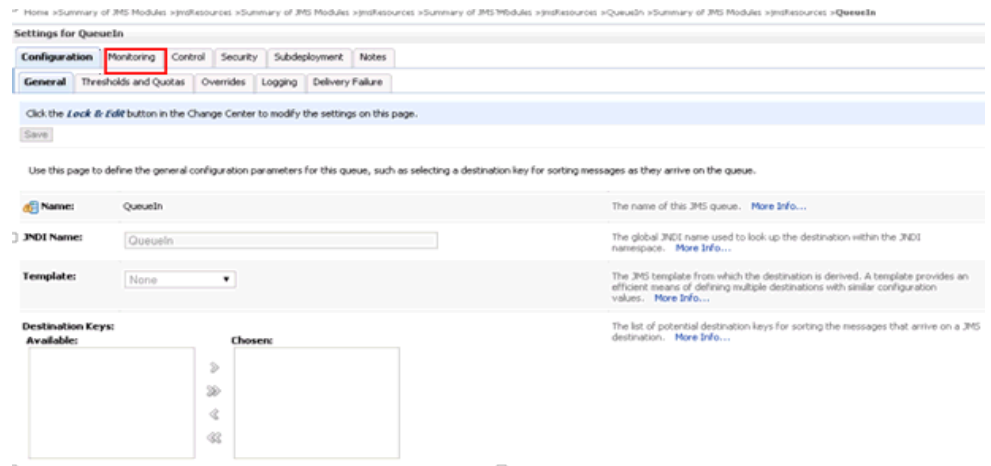
11. OSB JMS インバウンド処理をデプロイします。詳細は、8-27 ページ 8.2.3 項「OSB インバウンド処理のデプロイ」を参照してください。
12. 処理が正常にデプロイされたら、イベント・メッセージをトリガーします。
詳細は、4-49 ページ 4.5.5 項「Siebel でのイベントのトリガー」を参照してください。
13. Oracle WLS コンソールにログオンします。
14. Oracle WLS コンソールで、「サービス」を展開し、「メッセージ機能」をクリックし、「JMS モジュール」を選択して「jmsResources」をクリックします。
15. 図 8-82 に示すように、適切なレスポンス・リンク (QueueIn など) をクリックします。

図 8-82 QueueIn レスポンス・リンク



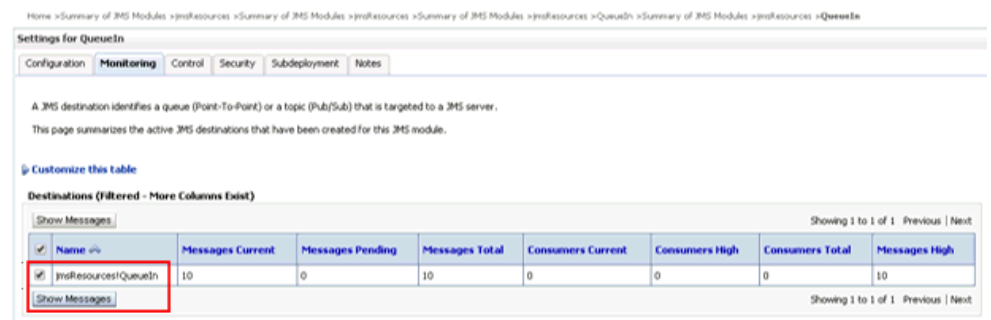
16. 図 8-83 に示すように、モニタリング・タブをクリックします。

図 8-83 モニタリング・タブ



17. 図 8-84 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 8-84 「メッセージの表示」ボタン



18. 適切な日付と時刻の ID リンクをクリックします。

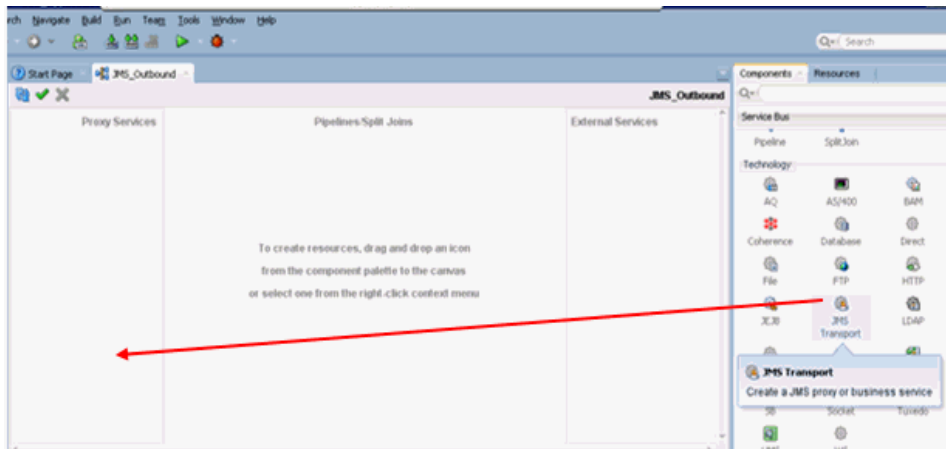
「テキスト」フィールドの下にレスポンス・ドキュメントが表示されます。

8.5 JDeveloper を使用した JMS アウトバウンド処理の構成 (J2CA 構成)

この項では、J2CA 構成に Oracle JDeveloper を使用して、Siebel システムに JMS 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。

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対応する WSDL ファイルを生成する必要があります。詳細は、4-8 ページ [4.4.1 項「リクエスト/レスポンス・サービス用の WSDL の生成」](#) を参照してください。
2. Oracle JDeveloper を起動し、OSB 用の Service Bus アプリケーションを作成します。詳細は、8-2 ページ [8.1.1 項「OSB 用の 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、8-3 ページ [8.1.2.1 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」](#) を参照してください。
4. JCA バインディング・ファイルから WSDL ベースのビジネス・サービスを作成します。詳細は、8-7 ページ [8.1.2.2 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. JMS プロキシ・サービスとパイプラインを作成し、次のステップを実行します。
 - a. [図 8-85](#) に示すように、テクノロジー・コンポーネント・ペインから **JMS トランスポート・コンポーネント** を「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。

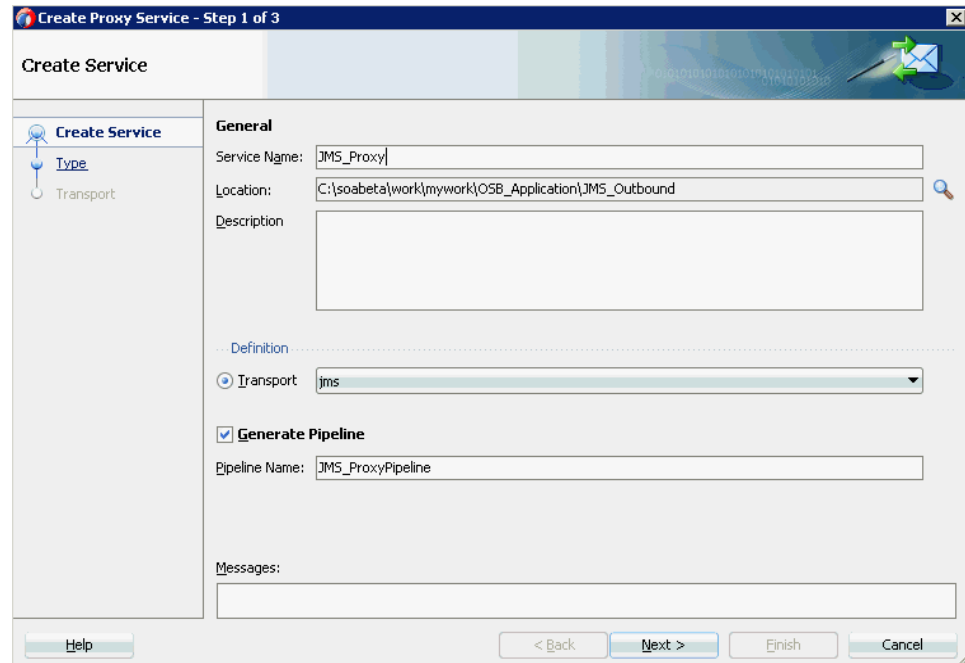
図 8-85 JMS トランスポート・コンポーネント



「ビジネス・サービス」ダイアログが表示されます。

- b. 「サービス名」フィールドに、任意のプロキシ・サービスの名前 (JMS_Proxy など) を入力します。「パイプラインを生成する」がデフォルトで選択されています。
- c. [図 8-86](#) に示すように、「次へ」をクリックします。

図 8-86 プロキシ・サービスの作成ペイン

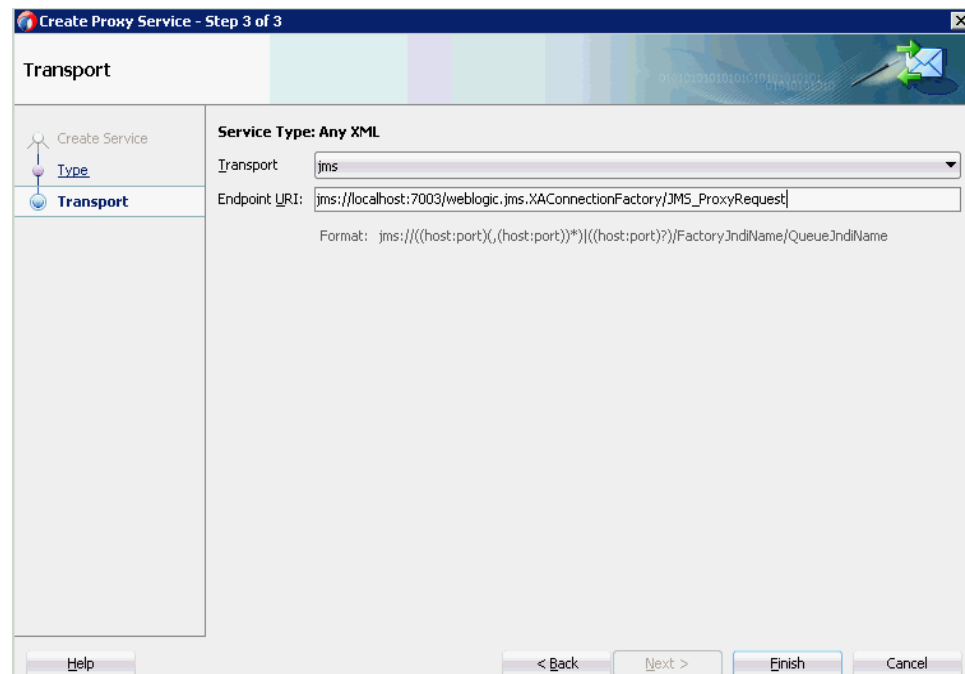


- d. 表示される「タイプ」ウィンドウで、「任意の XML」を選択して「次へ」をクリックします。

「トランスポート」ウィンドウが表示されます。

- e. 図 8-87 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 8-87 「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。


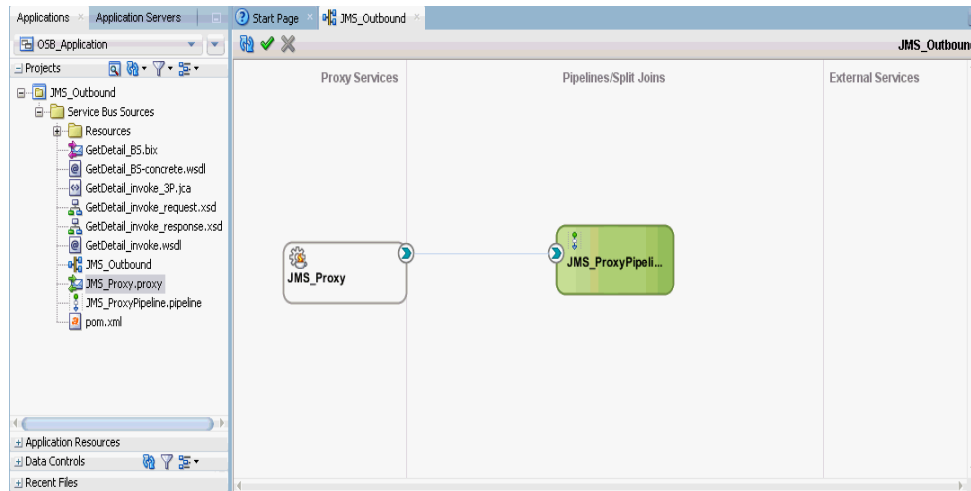
- f.  8-88 に示すように、作成したプロキシ・サービス (JMS_Proxy など) をダブルクリックします。

図 8-88 JMS プロキシ・サービス




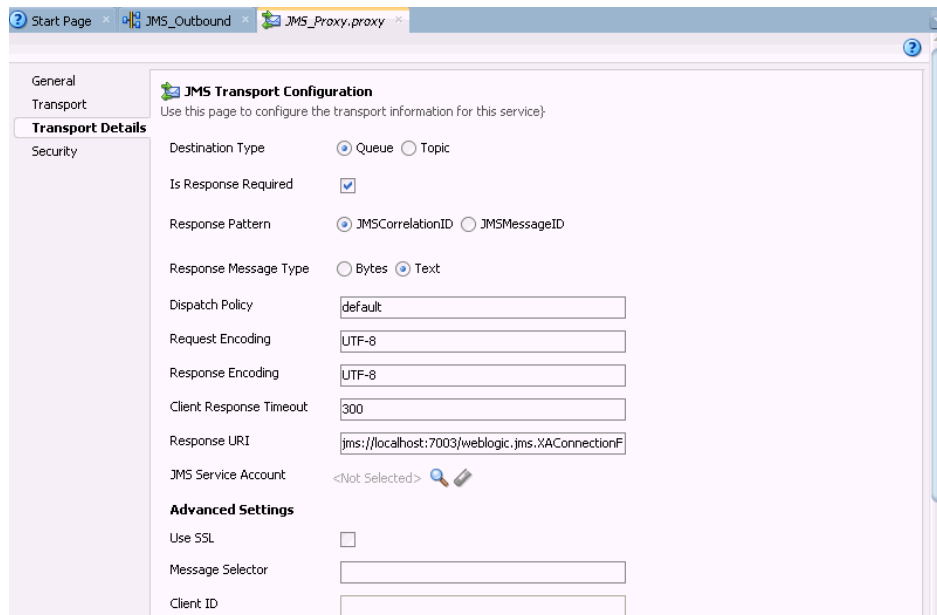
- g.  8-89 に示すように、表示されるプロキシ・サービスの構成ページで、「トランスポートの詳細」タブに次のパラメータを入力します。

図 8-89 JMS トランスポート構成

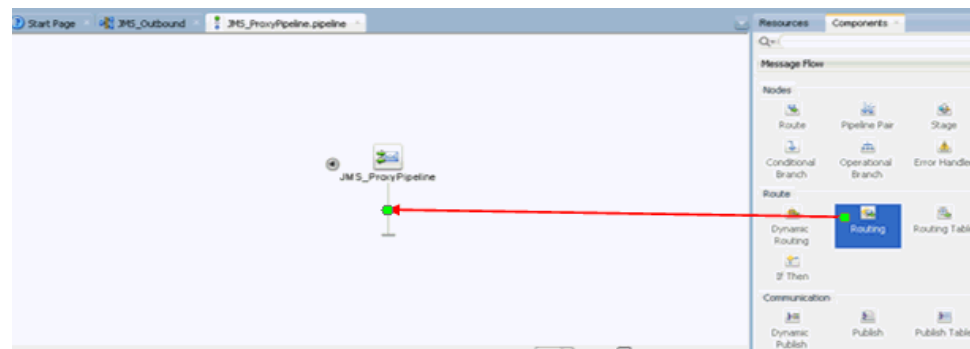


- h. 「接続先タイプ」セクションで「キュー」を選択します。
 i. 「レスポンスが必要」チェック・ボックスを選択します。
 j. 「レスポンス・メッセージ・タイプ」セクションで「テキスト」を選択します。
 k. 「レスポンス URI」フィールドに JMS トランスポート構成で使用されているエンドポイント URI を入力し、Request を Response に変更します。たとえば、


```
jms://localhost:7003/weblogic.jms.XAConnectionFactory/JMS_ProxyResponse
```

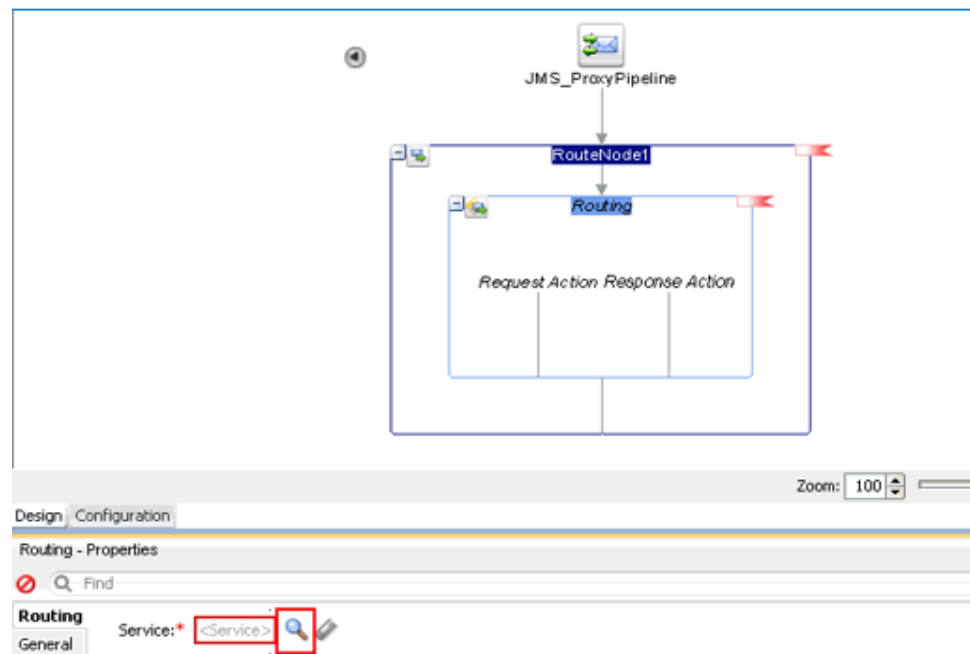
6. プロキシ・サービスの「構成」 ページを保存して閉じます。
7. ルーティング・ルールを構成し、次のステップを実行します。
 - a. パイプライン/分割結合ペインでパイプライン(JMS_ProxyPipelineなど)をダブルクリックします。
パイプライン構成ページが表示されます。
 - b. 図 8-90 に示すように、「ルート」セクションから「ルーティング」コンポーネントをパイプライン (JMS_ProxyPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 8-90 ルーティング・コンポーネント



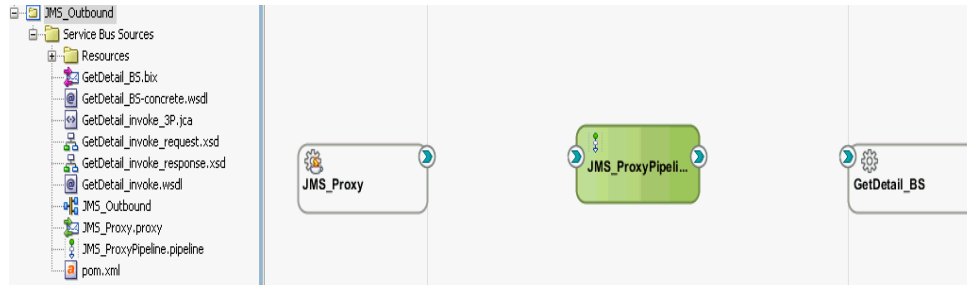
- c. 図 8-91 に示すように、パイプライン構成ページで、「ルーティング」を選択し、ルーティング・プロパティ・ペインの「サービス」フィールドの右の「参照」アイコンをクリックします。

図 8-91 サービスの参照



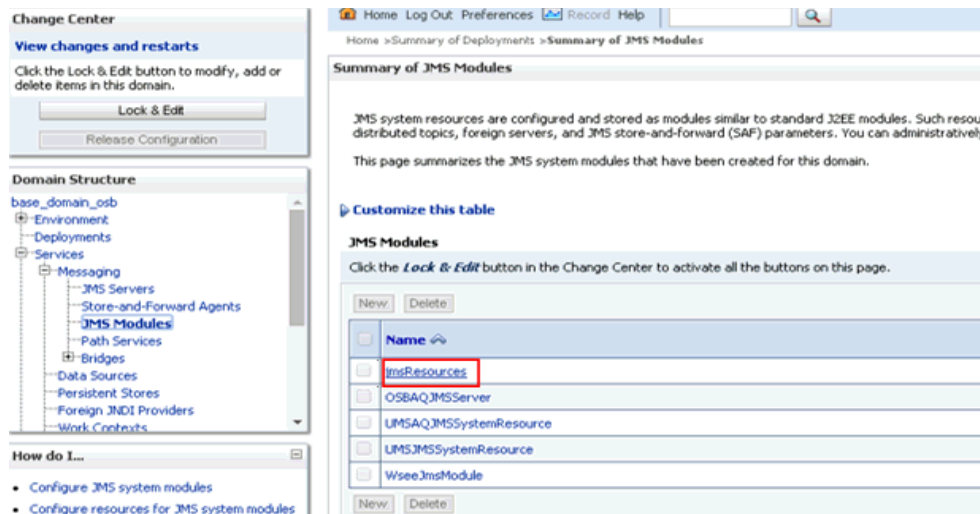
- d. 表示されるリソース・チューザ・ウィンドウで、WSDL ベースのビジネス・サービス (xxxxx_BS.bix など) を選択して「OK」をクリックします。
パイプライン構成ページに戻ります。
- e. パイプライン構成ページを保存して閉じます。
コンポジット・エディタ・ウィンドウに戻ります。
- f. 図 8-92 に示すように、メニュー・バーの「すべて保存」をクリックし、OSB JMS プロセスを保存します。

図 8-92 「トランスポート」ウィンドウ



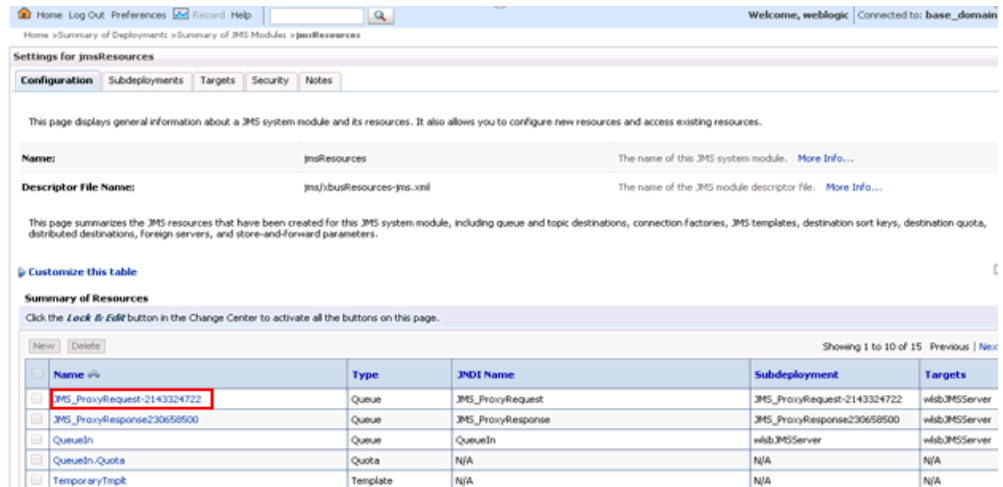
8. OSB JMS アウトバウンド処理をデプロイします。詳細は、8-17 ページ 8.1.3 項「OSB アウトバウンド処理のデプロイ」を参照してください。
9. 処理が正常にデプロイされたら、Oracle WLS コンソールにログオンします。
10. 図 8-93 に示すように、Oracle WLS コンソールで、「サービス」を展開し、「メッセージ機能」をクリックし、「JMS モジュール」を選択して「jmsResources」をクリックします。

図 8-93 JMS リソース



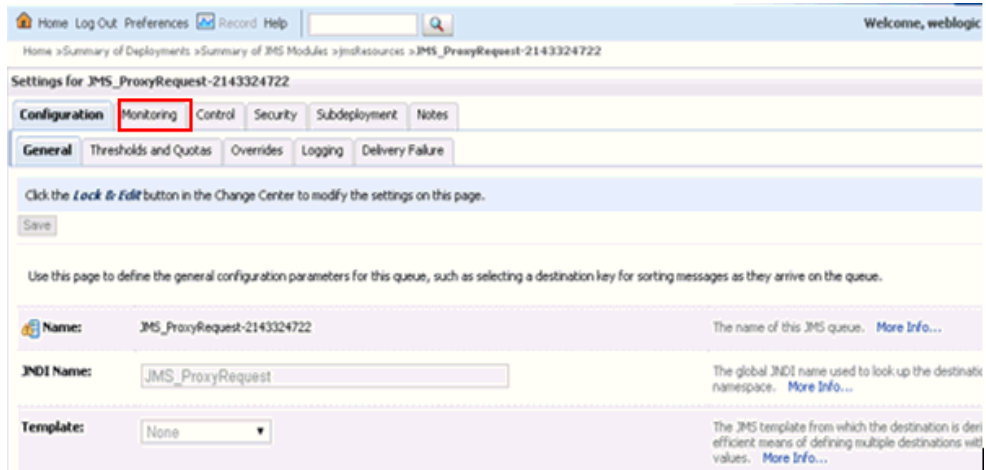
11. 図 8-94 に示すように、適切なリクエスト・リンク (JMS_ProxyRequest など) をクリックします。

図 8-94 JMS_ProxyRequest リンク



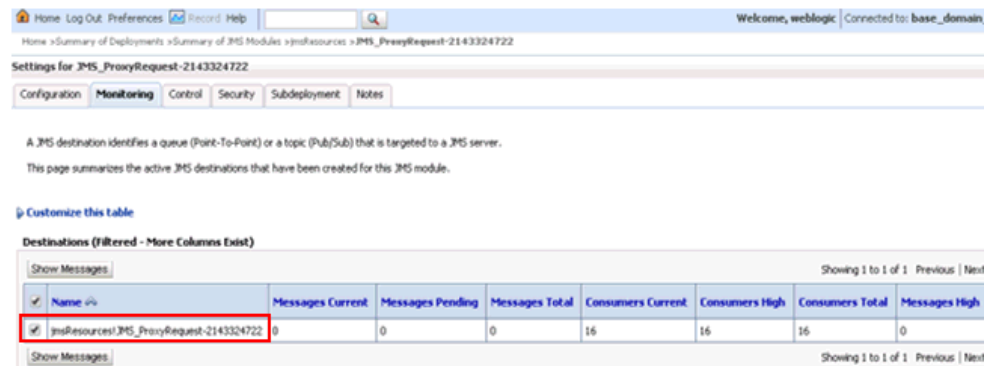
12. 図 8-95 に示すように、モニタリング・タブをクリックします。

図 8-95 モニタリング・タブ



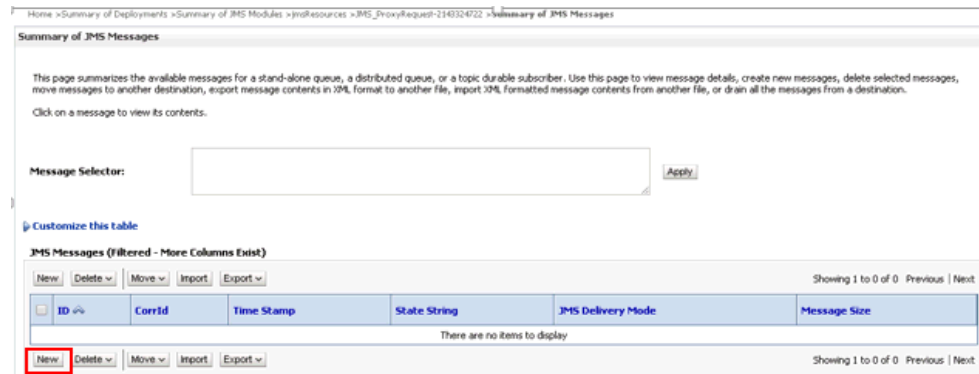
13. 図 8-96 に示すように、チェック・ボックスを選択して、「メッセージの表示」ボタンをクリックします。

図 8-96 「メッセージの表示」ボタン



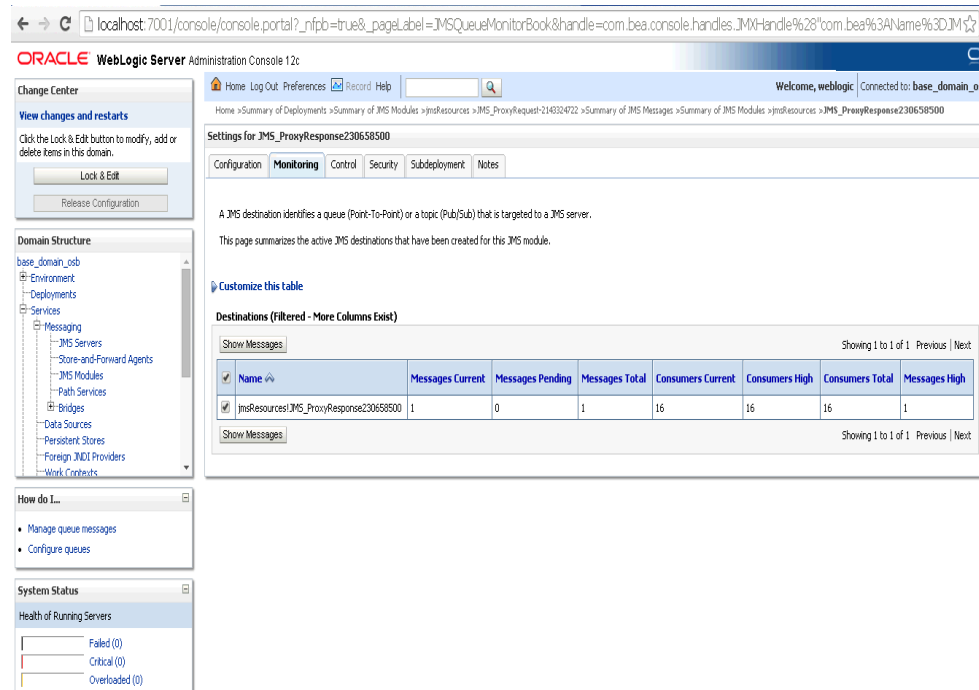
14. 図 8-97 に示すように、「新規」をクリックします。

図 8-97 JMS メッセージ



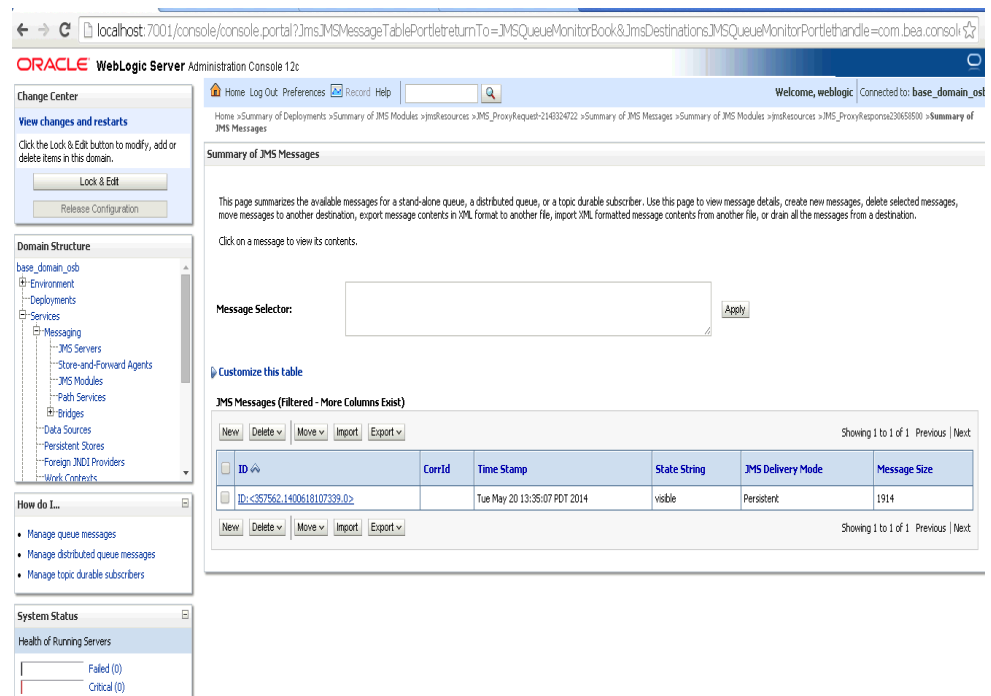
15. 本文フィールドに入力ペイロードを入力して、「OK」をクリックします。
16. Oracle WLS コンソールで、「サービス」を展開し、「メッセージ機能」をクリックし、「JMS モジュール」を選択して「jmsResources」をクリックします。
17. 適切なレスポンス・リンク (JMS_ProxyResponse など) をクリックします。
18. モニタリング・タブをクリックします。
19. 図 8-98 に示すように、チェック・ボックスを選択して、「メッセージの表示」をクリックします。

図 8-98 宛先メッセージ



20. 図 8-99 に示すように、適切な日付と時刻の ID リンクをクリックします。

図 8-99 「JMS メッセージのサマリー」 ウィンドウ



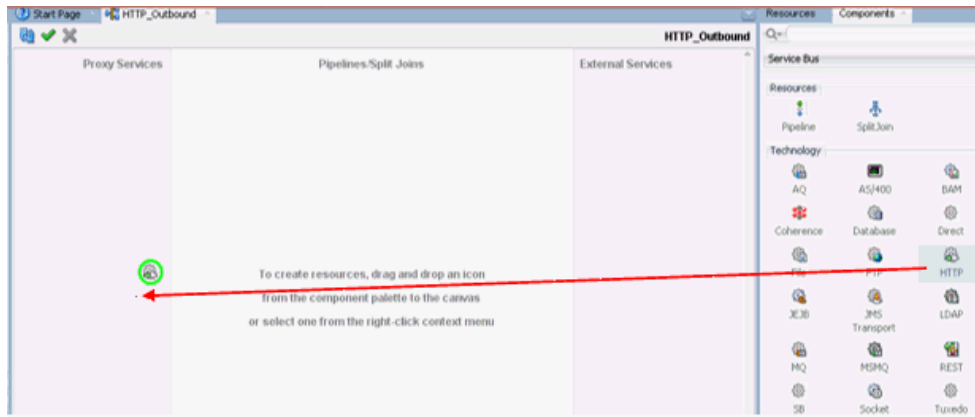
「テキスト」フィールドの下にレスポンス・ドキュメントが表示されます。

8.6 JDeveloper を使用した HTTP アウトバウンド処理の構成 (J2CA 構成)

この項では、J2CA 構成に Oracle JDeveloper を使用して、Siebel システムに HTTP 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。

1. HTTP アウトバウンド処理を設計する前に、アプリケーション・エクスプローラを使用して対応する WSDL ファイルを生成する必要があります。詳細は、4-8 ページ 4.4.1 項「リクエスト/レスポンス・サービス用の WSDL の生成」を参照してください。
2. Oracle JDeveloper を起動し、OSB 用の Service Bus アプリケーションを作成します。詳細は、8-2 ページ 8.1.1 項「OSB 用の 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、8-3 ページ 8.1.2.1 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」を参照してください。
4. HTTP プロキシ・サービスとパイプラインを作成し、次のステップを実行します。
 - a. 図 8-100 に示すように、テクノロジ・コンポーネント・ペインから「HTTP」コンポーネントを「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。

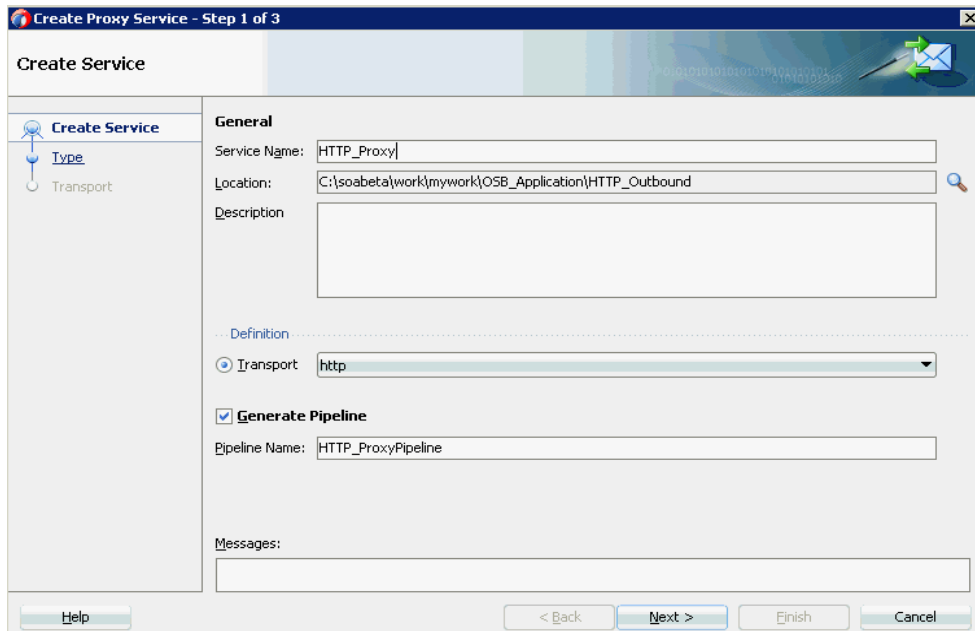
図 8-100 HTTP コンポーネント



プロキシ・サービスの作成ダイアログが表示されます。

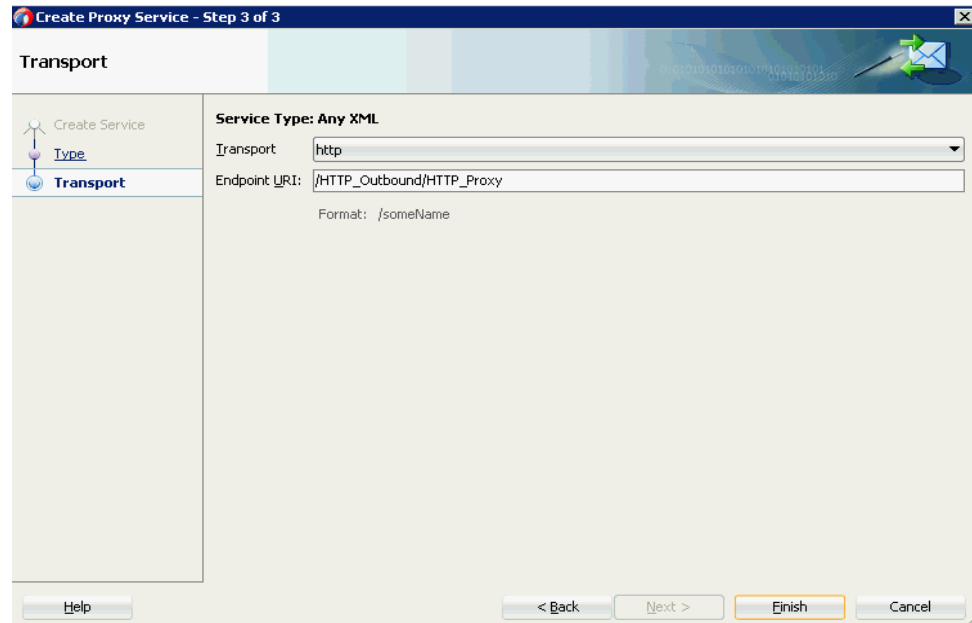
- b. 「サービス名」フィールドに、任意のプロキシ・サービスの名前 (HTTP_Proxy など) を入力します。「パイプラインを生成する」がデフォルトで選択されています。
- c. 図 8-101 に示すように、「次へ」をクリックします。

図 8-101 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」を選択して「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 8-102 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。

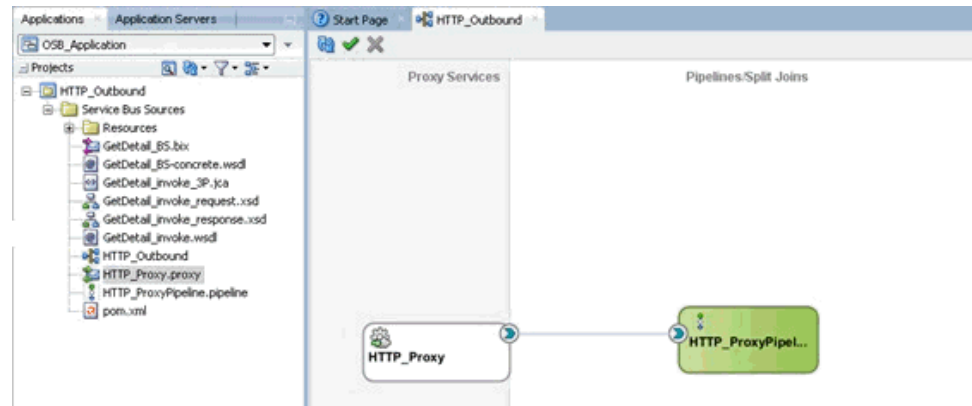
図 8-102 「トランスポート」 ウィンドウ



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 8-103](#) に示すように、パイプライン/分割結合ペインで、作成したパイプライン (HTTP_ProxyPipeline など) をダブルクリックします。

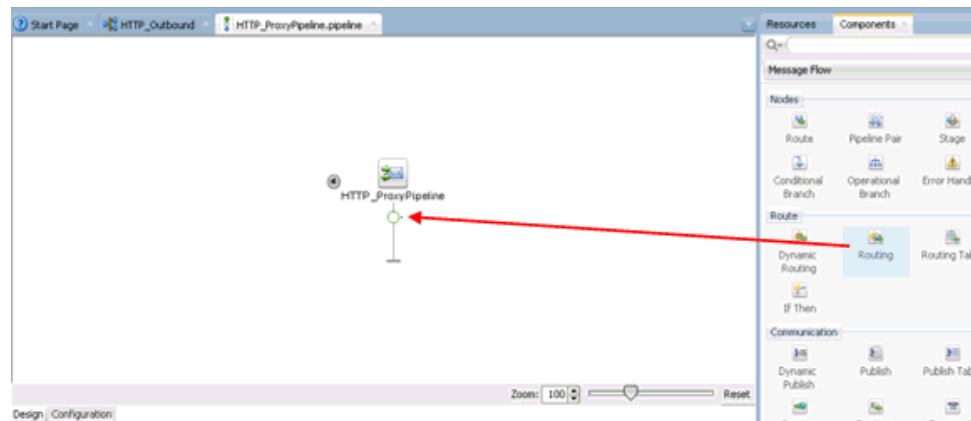
図 8-103 プロキシ・サービス



パイプライン構成ページが表示されます。

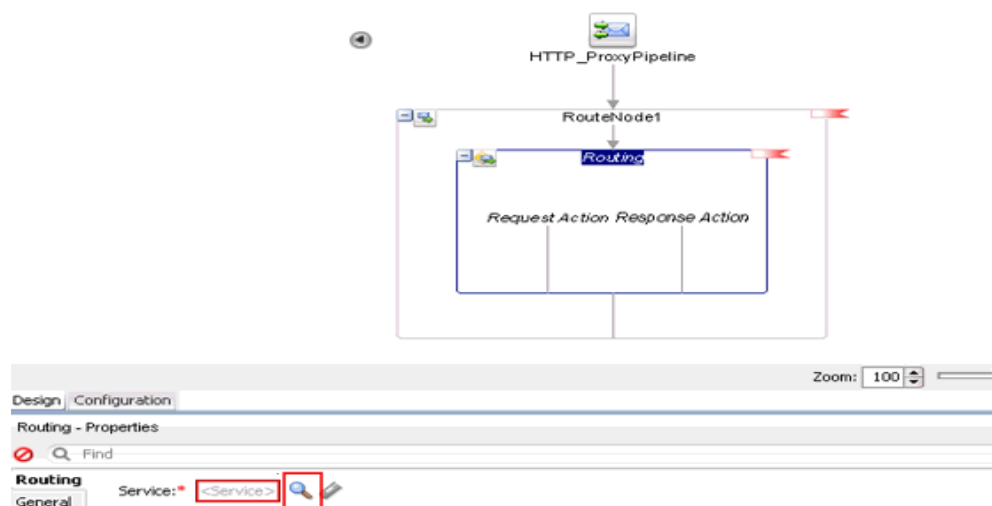
5. ルーティング・ルールを構成し、次のステップを実行します。
 - a. [図 8-104](#) に示すように、「ルート」セクションから「ルーティング」コンポーネントをパイプライン (HTTP_ProxyPipeline など) の下の領域にドラッグ・アンド・ドロップします。

図 8-104 ルーティング・コンポーネント



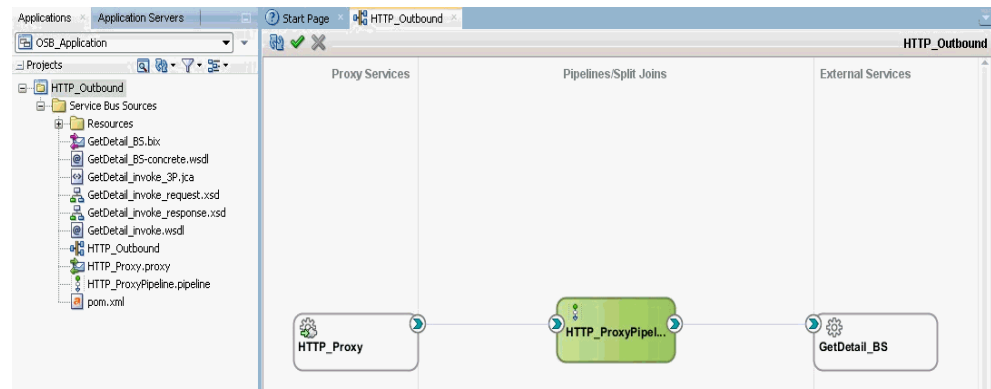
- b. 図 8-105 に示すように、パイプライン構成ページで、「ルーティング」を選択し、ルーティング-プロパティ・ペインの「サービス」フィールドの右の「参照」アイコンをクリックします。

図 8-105 サービスの参照



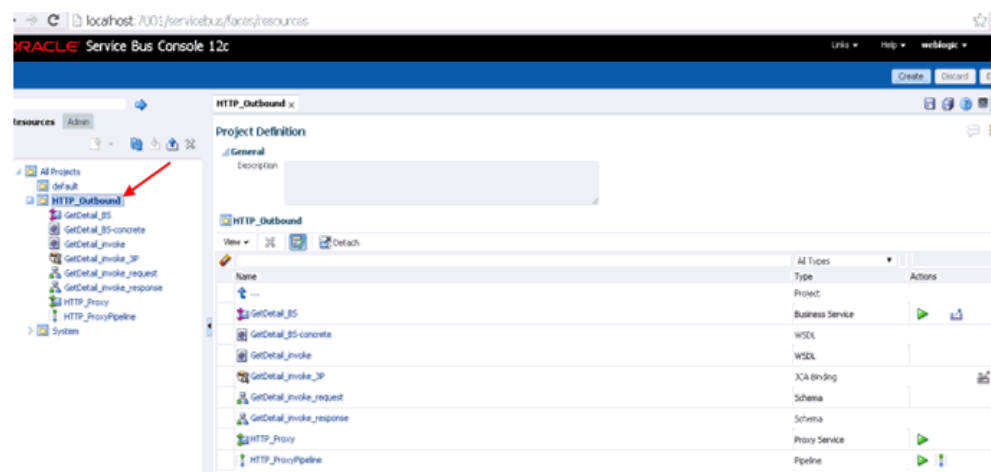
- c. 表示されるリソース・チューザ・ウィンドウで、WSDL ベースのビジネス・サービス (xxxxx_BS.bix など) を選択して「OK」をクリックします。
パイプライン構成ページに戻ります。
- d. パイプライン構成ページを保存して閉じます。
コンポジット・エディタ・ウィンドウに戻ります。
- e. 図 8-106 に示すように、メニュー・バーの「すべて保存」をクリックし、OSB HTTP プロセスを保存します。

図 8-106 「トランスポート」ウィンドウ



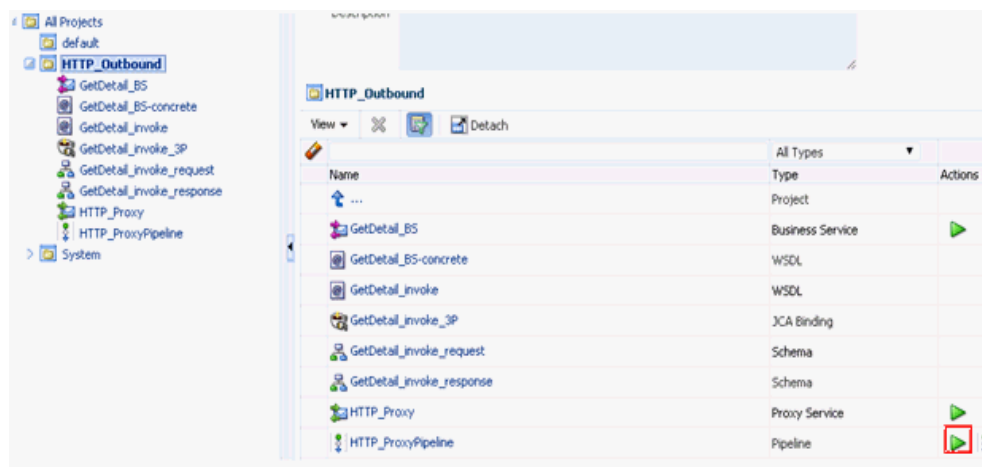
6. OSB HTTP アウトバウンド処理をデプロイします。詳細は、8-17 ページ 8.1.3 項「OSB アウトバウンド処理のデプロイ」を参照してください。
7. 処理が正常にデプロイされたら、Service Bus コンソールにログオンします。
8. 図 8-107 に示すように、Service Bus コンソールで、デプロイされた HTTP アウトバウンド・プロジェクト (HTTP_Outbound など) をクリックします。

図 8-107 Service Bus コンソール



9. 図 8-108 に示すように、作成したパイプライン用のテスト OSB コンソール・アイコンをクリックします。

図 8-108 テスト OSB コンソール・アイコン



10. 表示されるテスト OSB コンソール・ページで、入力 XML を入力し、「実行」ボタンをクリックします。

表示されるテスト OSB コンソール・ページで、レスポンスが受信されます。

この章では、Oracle Application Adapter for Siebel の新しい機能について説明します。この章の内容は以下のとおりです。

- 9.1 項「ロギング機能の構成」
- 9.2 項「診断機能の構成」
- 9.3 項「SOA デバッグ機能の構成」
- 9.4 項「例外フィルタ」
- 9.5 項「Oracle SOA Suite (BPEL、メディエータまたは BPM) 用の資格証明マッピング」
- 9.6 項「JDeveloper を使用した Oracle Service Bus (OSB) の資格証明マッピング」

9.1 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。内容は次のとおりです。

- 9.1.1 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」
- 9.1.2 項「Business Services Engine (BSE) のログ・ファイル管理の構成」

9.1.1 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.Siebel'  
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.Siebel' 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.Siebel' 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.Siebel'  
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.Siebel' 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。

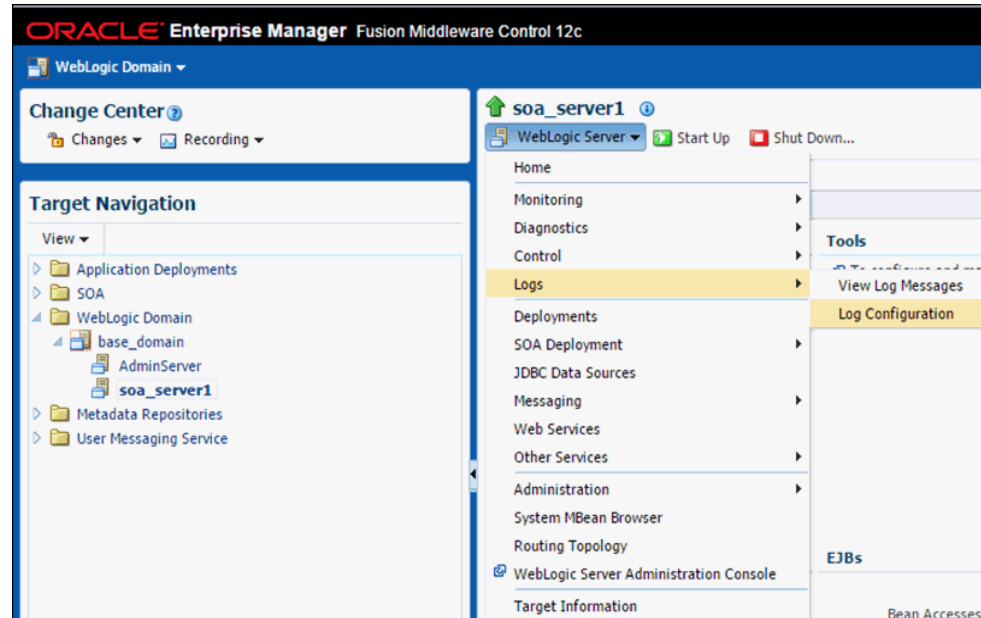
```
http://host name:port/em
```

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 9-1 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 9-1 「ログ構成」オプション



- 図 9-2 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 9-2 「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 when a 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 an EJB module is deployed. Log levels for runtime loggers are not persisted across component restarts.

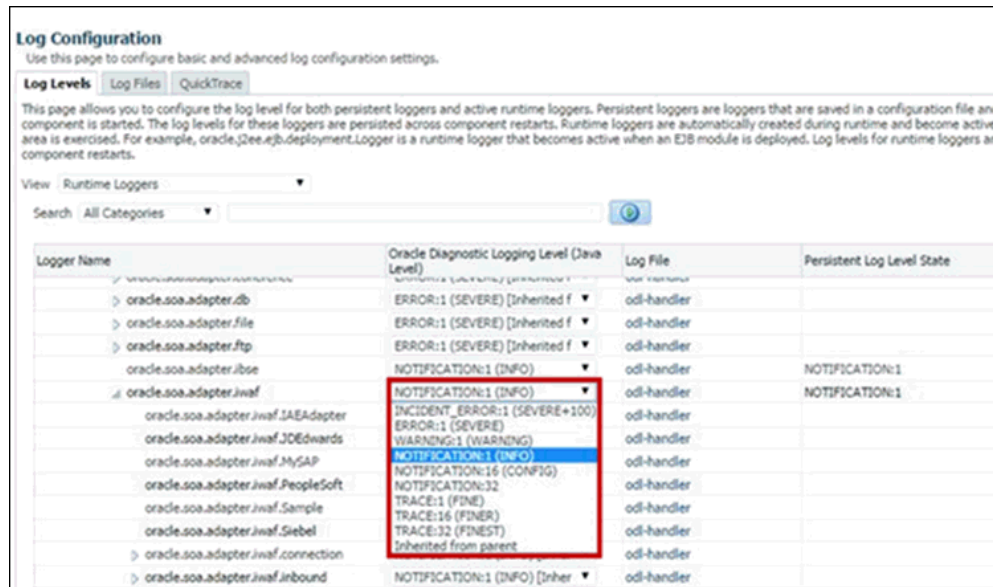
View: Runtime Loggers

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 ...]	odl-handler	
oracle.soa.adapter.file	ERROR:1 (SEVERE) [Inherited from ...]	odl-handler	
oracle.soa.adapter.ftp	ERROR:1 (SEVERE) [Inherited from ...]	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 ...]	odl-handler	
oracle.soa.adapter.iwaf.JDEdwards	NOTIFICATION:1 (INFO) [Inherited from ...]	odl-handler	
oracle.soa.adapter.iwaf.MySAP	NOTIFICATION:1 (INFO) [Inherited from ...]	odl-handler	
oracle.soa.adapter.iwaf.PeopleSoft	NOTIFICATION:1 (INFO) [Inherited from ...]	odl-handler	
oracle.soa.adapter.iwaf.Sample	NOTIFICATION:1 (INFO) [Inherited from ...]	odl-handler	
oracle.soa.adapter.iwaf.Siebel	NOTIFICATION:1 (INFO) [Inherited from ...]	odl-handler	
oracle.soa.adapter.iwaf.connection	NOTIFICATION:1 (INFO) [Inherited from ...]	odl-handler	
oracle.soa.adapter.iwaf.inbound	NOTIFICATION:1 (INFO) [Inherited from ...]	odl-handler	

7. 図 9-3 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。

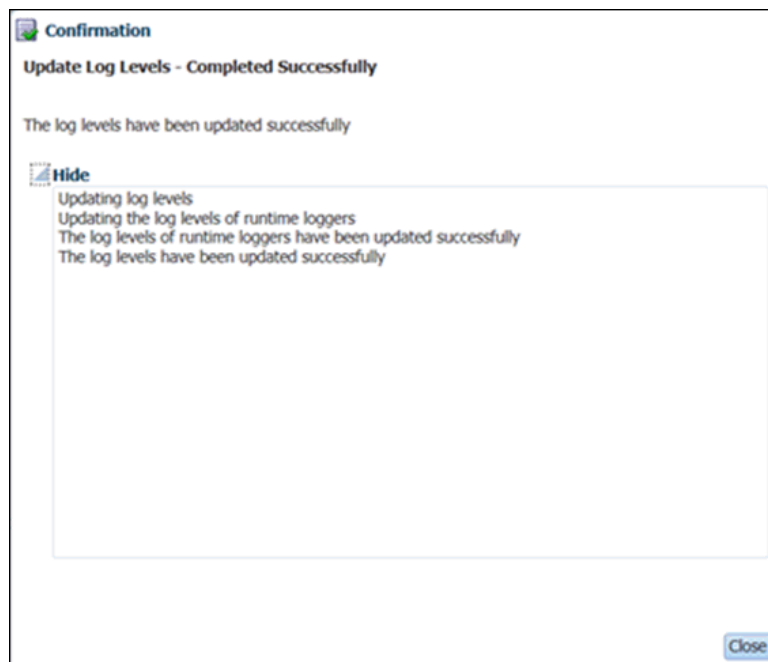
図 9-3 「Oracle Diagnostic Logging レベル (Java レベル)」列



8. 「適用」をクリックします。

図 9-4 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 9-4 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 9-1 「ログ・レベル」プロパティ設定

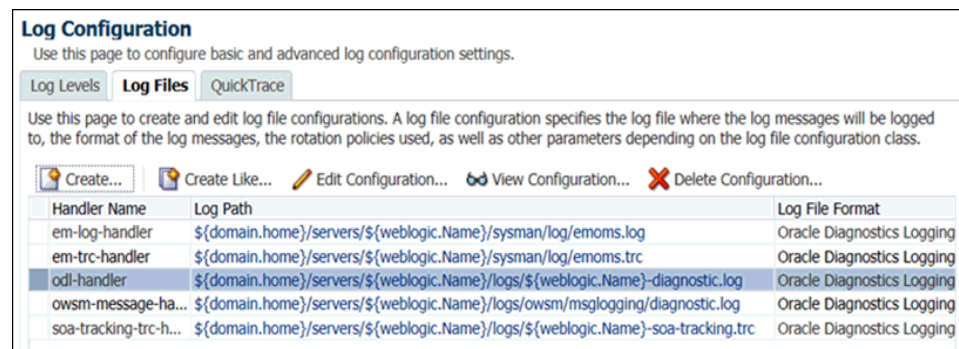
EM コンソールで設定された ログ・レベル	<code>{server-name}-diagnostic.log</code> で更新された ログ・レベル
ERROR:1	ERROR
WARNING:1	WARNING
NOTIFICATION:1	NOTIFICATION
NOTIFICATION:16	NOTIFICATION
NOTIFICATION:32	NOTIFICATION
TRACE:1	NOTIFICATION
TRACE:16	NOTIFICATION
TRACE:32	TRACE:32

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 9-5 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 9-5 「ログ構成」 ペイン

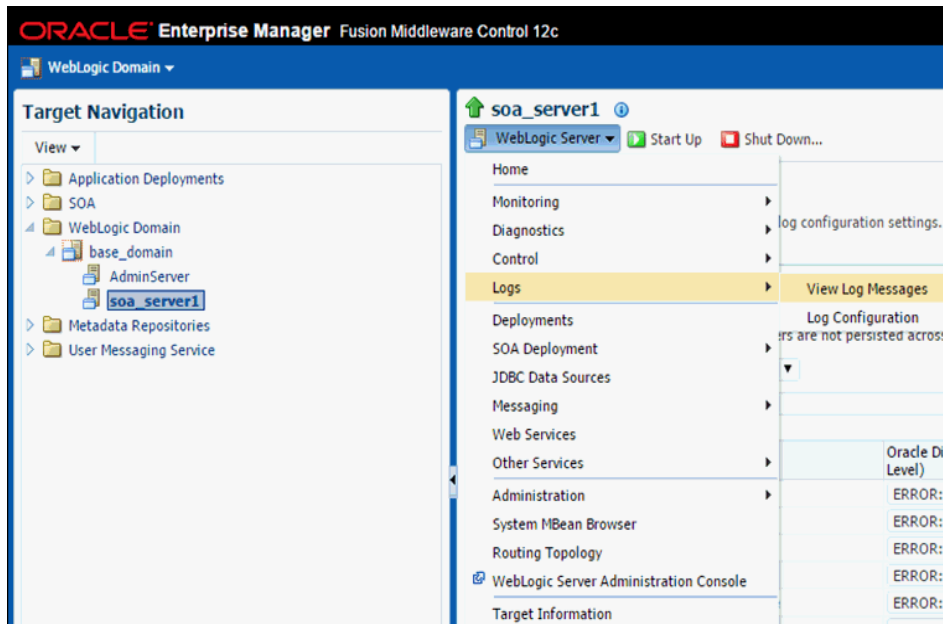


ログは、次の場所で入手できる `{server-name}-diagnostic.log` で更新されます。

`<ORACLE_HOME>\user_projects\domains\base_domain\servers\<server_Name>\logs`

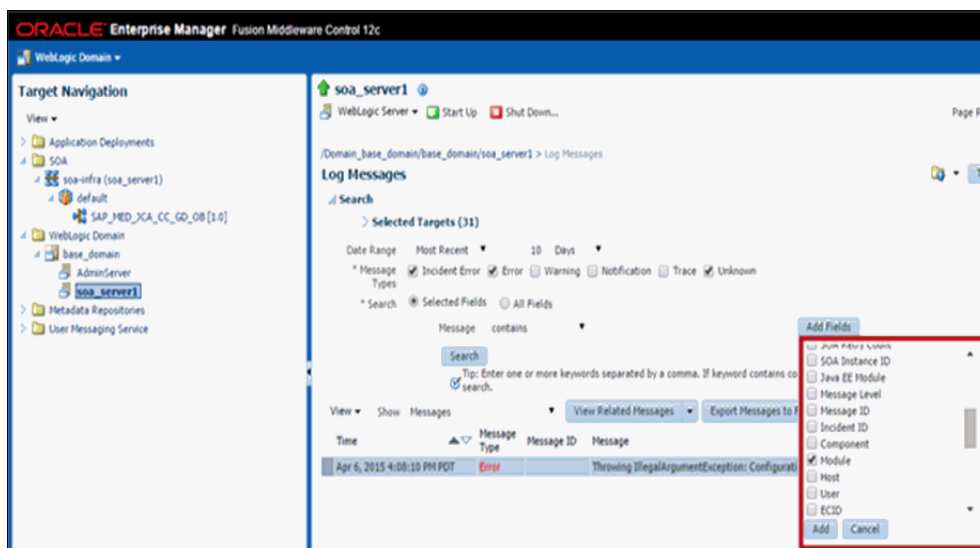
10. 図 9-6 に示すように、「サーバー」 ペインで、「WebLogic Server」 ドロップダウン・リストをクリックし、「ログ」 を選択し、「ログ・メッセージの表示」 をクリックすると、EM コンソールにログ・メッセージが表示されます。

図 9-6 「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 9-7 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 9-7 「モジュール」 フィールド



12. 「追加」 をクリックします。
13. 図 9-8 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 9-8 「ログ・メッセージ」 ペイン

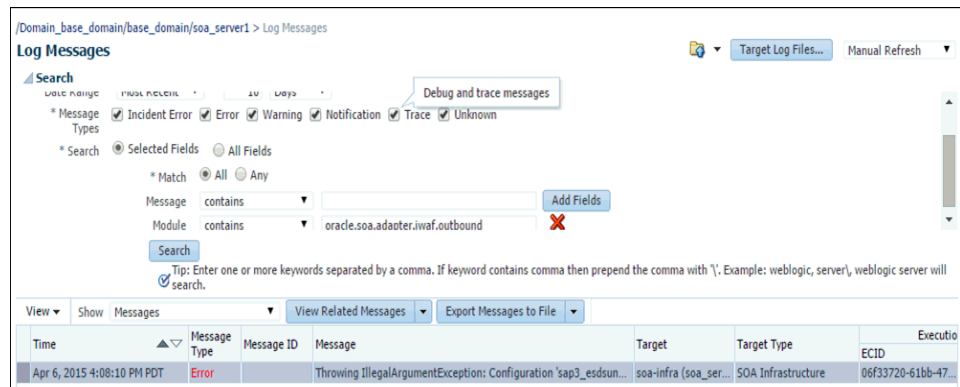
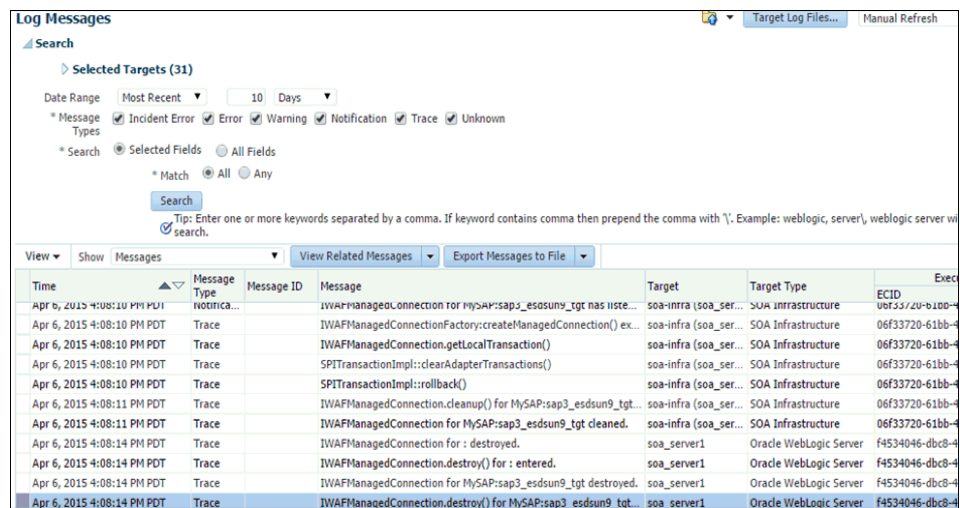


図 9-9 に示すように、指定されたログ出力のメッセージが、検索基準の下の表に表示されます。

図 9-9 ログ出力のメッセージ



14. 表の任意の行を選択します。図 9-10 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 9-10 「ECID」 オプション

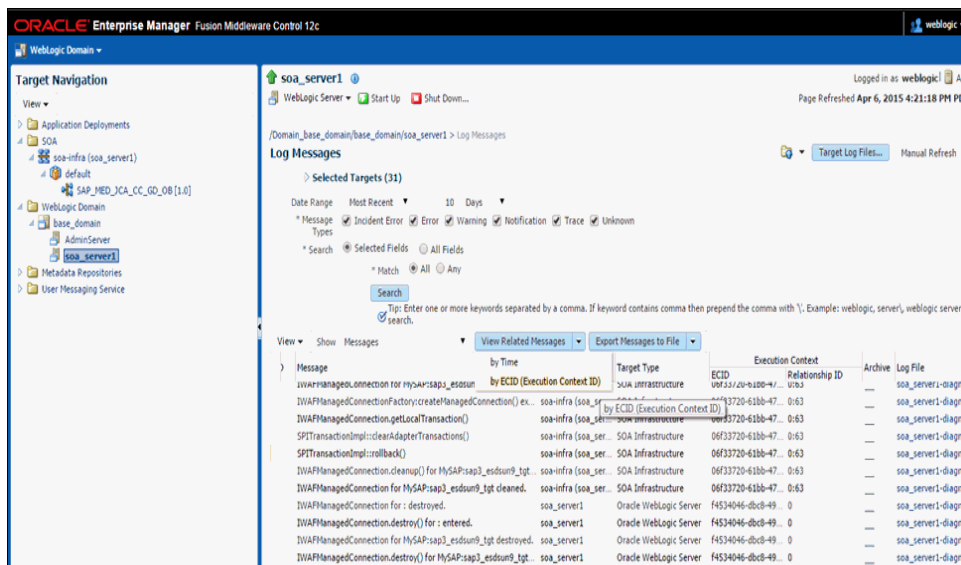
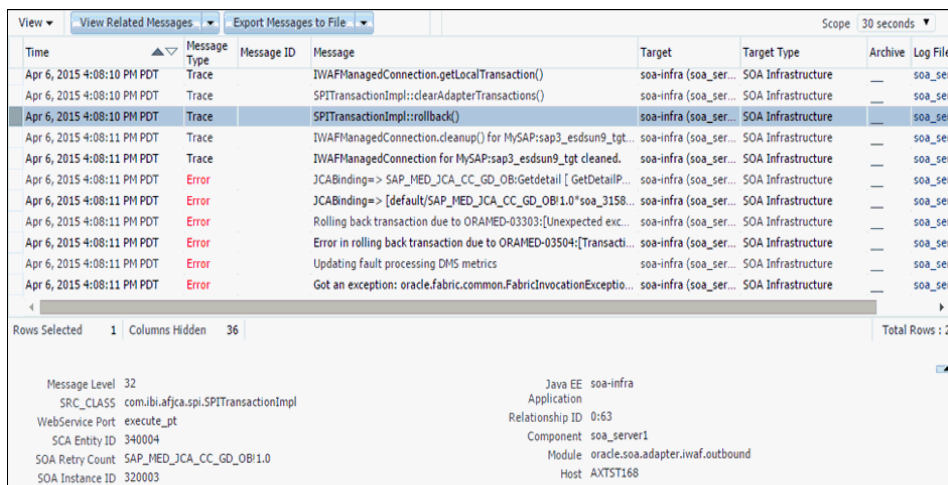


図 9-11 に示すように、詳細が表示されます。

図 9-11 メッセージの詳細



9.1.2 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。

- 9.1.1 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 9-12 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 9-12 「ログ・レベル」タブ

Logger Name	Oracle Diagnostic Logging Level (Java Level)	Log File	Persistent Log Level State
oracle.soa.adapter.aq	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.coherence	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.db	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.file	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.ftp	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.ibse	NOTIFICATION:1 (INFO) ▼	odi-handler	NOTIFICATION:1
oracle.soa.adapter.iwaf	TRACE:32 (FINEST) ▼	odi-handler	ERROR:1
oracle.soa.adapter.jdeworld	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.jms	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.ldap	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.mq	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	

- 図 9-13 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。

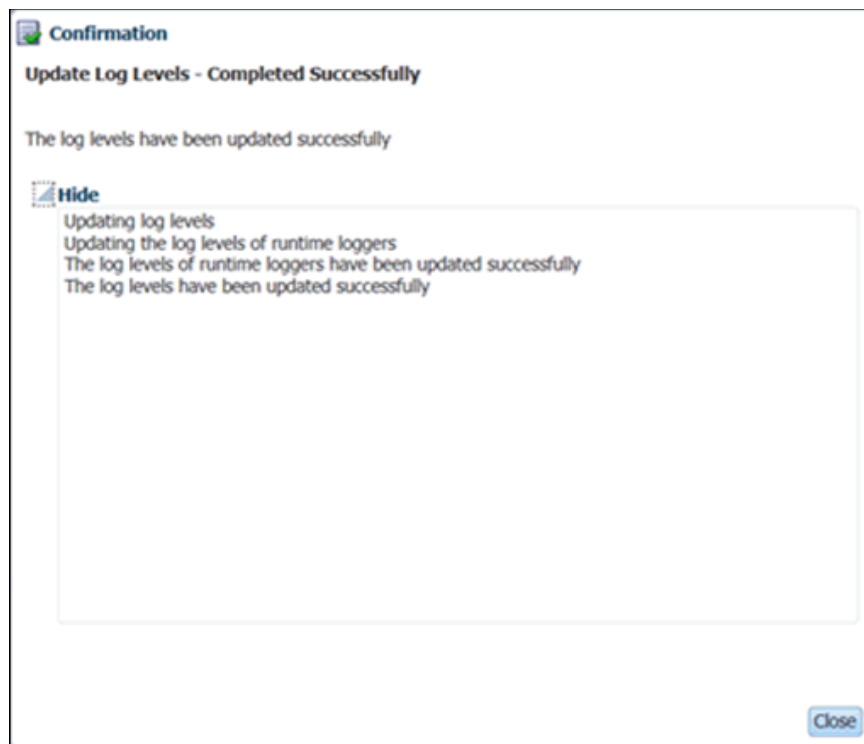
図 9-13 「Oracle Diagnostic Logging レベル (Java レベル)」列

Logger Name	Oracle Diagnostic Logging Level (Java Level)	Log File	Persistent Log Level State
oracle.soa.adapter.aq	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.coherence	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.db	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.file	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.ftp	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.ibse	NOTIFICATION:1 (INFO) ▼	odi-handler	NOTIFICATION:1
oracle.soa.adapter.iwaf	INCIDENT_ERROR:1 (SEVERE+100)	odi-handler	ERROR:1
oracle.soa.adapter.jdeworld	ERROR:1 (SEVERE) [Inherited f ▼]	odi-handler	
oracle.soa.adapter.jms	WARNING:1 (WARNING)	odi-handler	
oracle.soa.adapter.jms	NOTIFICATION:1 (INFO)	odi-handler	
oracle.soa.adapter.jms	NOTIFICATION:16 (CONFIG)	odi-handler	
oracle.soa.adapter.jms	NOTIFICATION:32	odi-handler	
oracle.soa.adapter.jms	TRACE:1 (FINE)	odi-handler	
oracle.soa.adapter.jms	TRACE:16 (FINER)	odi-handler	
oracle.soa.adapter.jms	TRACE:32 (FINEST)	odi-handler	
oracle.soa.adapter.jms	Inherited from parent	odi-handler	

- 「適用」をクリックします。

図 9-14 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 9-14 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 9-2 「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、[9.1.1 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。

9.2 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。

- 9.2.1 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。

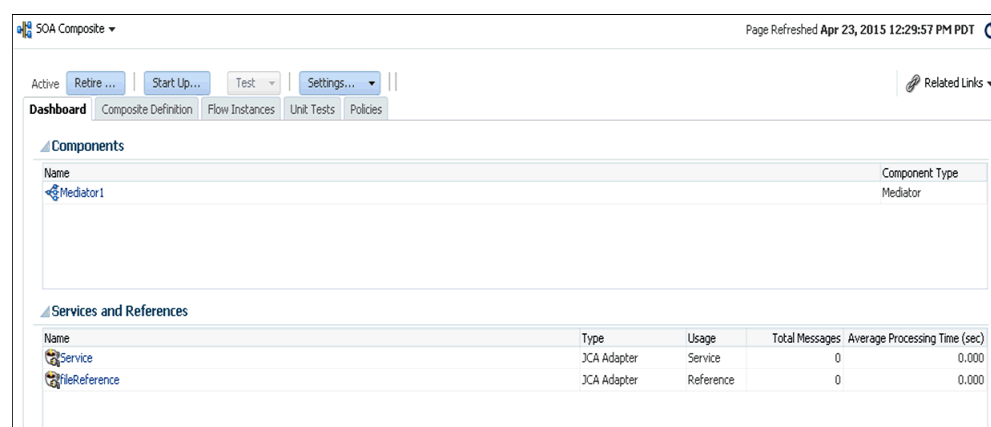
1. Oracle WebLogic Server を起動し、Web ブラウザで次の URL を入力して、Oracle WebLogic Server Enterprise Manager Console を起動します。

`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 9-15](#) 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 9-15 「サービスと参照」セクション



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 9-16](#) 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 9-16 スタック・トレース

The screenshot shows the 'SAMPACC (Custom Adapter)' configuration page in the Oracle Fusion Middleware Administration Console. The page is titled 'SOA Composite' and 'Page Refreshed Apr 10, 2015 5:24:59 AM EDT'. The main content area is divided into several sections: 'Dashboard', 'Policies', 'Properties', and 'Adapter Reports'. Under 'Adapter Reports', there are sub-sections for 'Diagnosability Reports', 'Configuration Reports', and 'Monitoring Reports'. The 'Monitoring Reports' section contains a table with the following columns: 'Node', 'EIS Connection Status', 'Managed Connections' (sub-columns: 'Currently Open', 'Average Number Used', 'Currently Free', 'Maximum Pool Size'), and 'Most Recent Time Stamp' (sub-columns: 'Last Message Publication', 'Last Service Activation'). Below this table, there is a 'Snapshot Reports' section with a 'Write to Database' checkbox. The 'Snapshot Reports' section includes a 'Retrieve Data' section with a 'Recent time period' dropdown set to 'Last*' and a '24 Hours' range. Below this, there is a table with the following columns: 'Server Name', 'Start EIS Downtime', 'End EIS Downtime', 'Error Summary', 'Total EIS Connection Downtime (Seconds)', and 'Data Location'. The 'Error Summary' column for the 'soa_server1' row contains the text 'Connection reset', which is highlighted with a red box.

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	

9.2.1 サポートされるプロトコル

この項では、Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for Siebel 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。

- 9.2.1.1 項「Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for Siebel のエンドポイント・ステータス」

9.2.1.1 Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for Siebel のエンドポイント・ステータス

Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for Siebel (インバウンド) で使用できるプロトコルは次のとおりです。

- HTTP
- MQ
- ファイル (サポートされていません)

HTTP のリスニング・プロトコル・アダプタは、ソケットでリスニングします。そのため、その性質上、ネットワークの反対側で何かが動作しているかどうかは、リクエストが受信されるまでわかりません。リクエストが受信された場合も、そのリクエストの送信元を確実に特定することはできません。

これらのプロトコルでは、リクエストが受信されると、送信元は EIS と見なされ、EIS が使用されます。EIS は、リクエストの受信時に通信エラーが発生していないかどうかを確認します。ただし、この確認はごく基本的なものにとどまります。

ただし、MQ インバウンド・リスナー・アダプタは MQ イベントをポーリングするため、MQ API を使用したポーリングの発生時に通信障害 (または正常な通信) が検出されます。

9.3 SOA デバッグ機能の構成

この項では、Oracle Application Adapter for Siebel 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。内容は次のとおりです。

- 9.3.1 項「SOA デバッガの使用のガイドライン」
- 9.3.2 項「前提条件」
- 9.3.3 項「Oracle JDeveloper での BPEL プロセスのデバッグ」
- 9.3.4 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。

9.3.1 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。

9.3.2 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。

9.3.3 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。内容は次のとおりです。

- 9.3.3.1 項「Oracle JDeveloper でのアウトバウンド BPEL プロセスのデバッグ」
- 9.3.3.2 項「Oracle JDeveloper でのインバウンド BPEL プロセスのデバッグ」

9.3.3.1 Oracle JDeveloper でのアウトバウンド BPEL プロセスのデバッグ


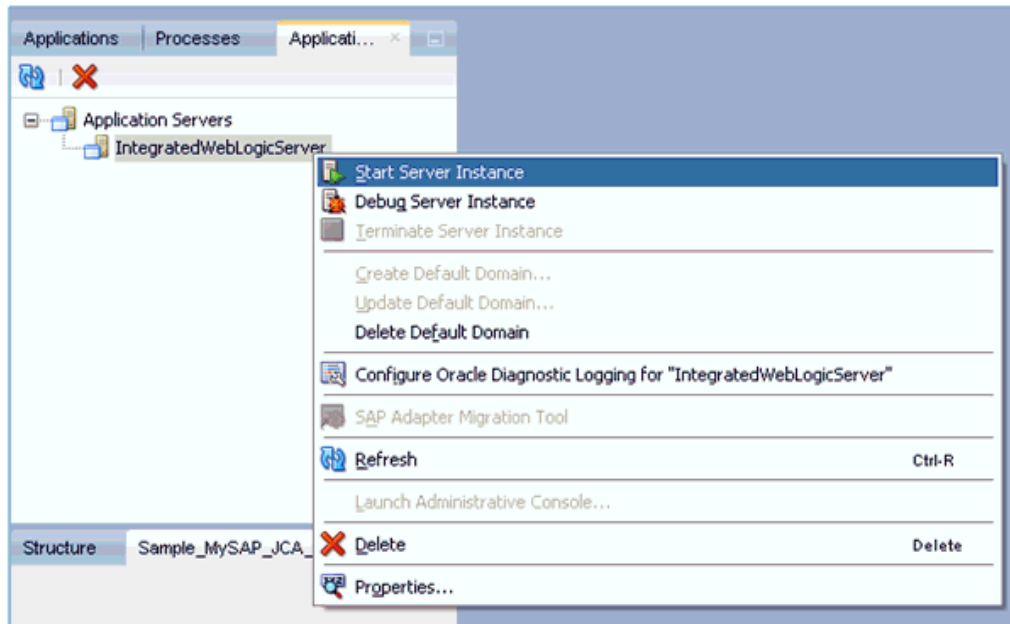
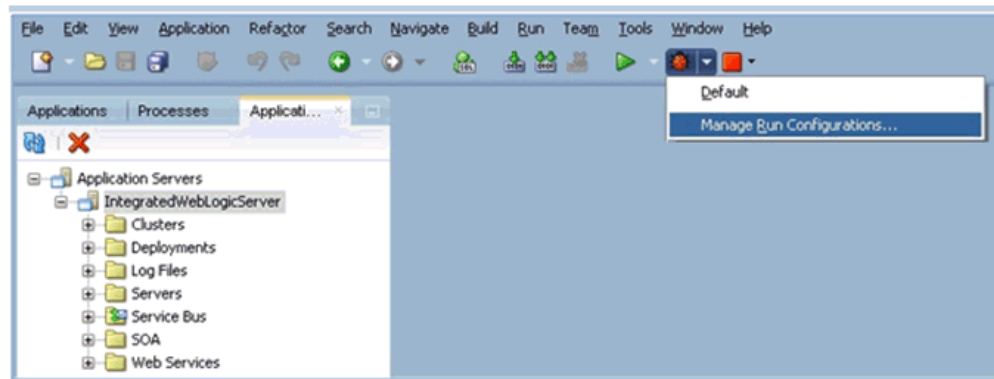
1. Oracle JDeveloper を開きます。
2. *IntegratedWeblogicServer* ドメインを起動します。
 - a. 左ペインの「アプリケーション・サーバー」タブをクリックします。
 - b.  9-17 に示すように、「アプリケーション・サーバー」ノードの下の「IntegratedWeblogicServer」を右クリックし、コンテキスト・メニューから「サーバー・インスタンスの起動」を選択します。

図 9-17 サーバー・インスタンスの起動



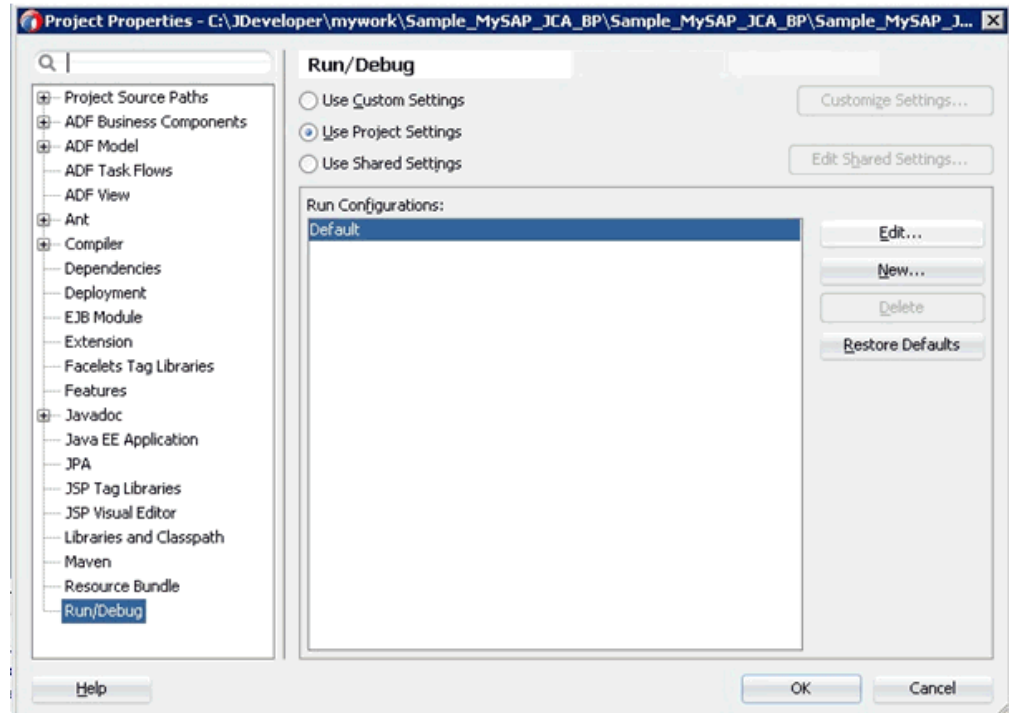
3. デバッグ環境を設定します。
 - a. 図 9-18 に示すように、「デバッグ」アイコンの横の下矢印をクリックし、コンテキスト・メニューから「実行構成の管理」を選択します。

図 9-18 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 9-19 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。

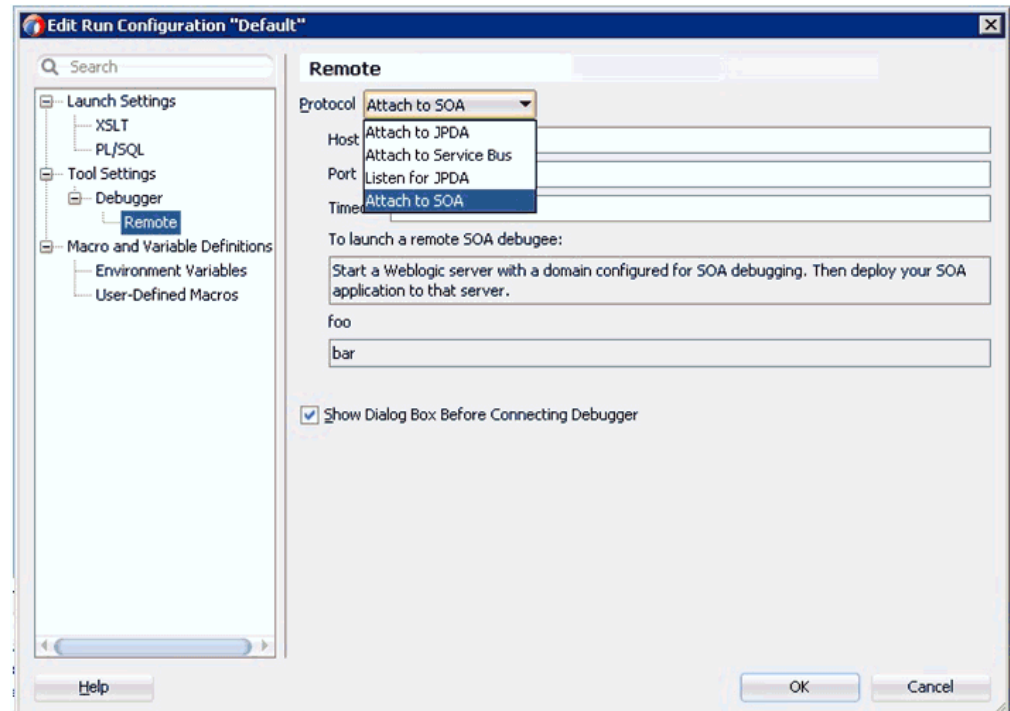
図 9-19 「プロジェクト・プロパティ」ダイアログ



- 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 9-20 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 9-20 「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.  9-21 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 9-21 デバッグする BPEL プロセスを選択

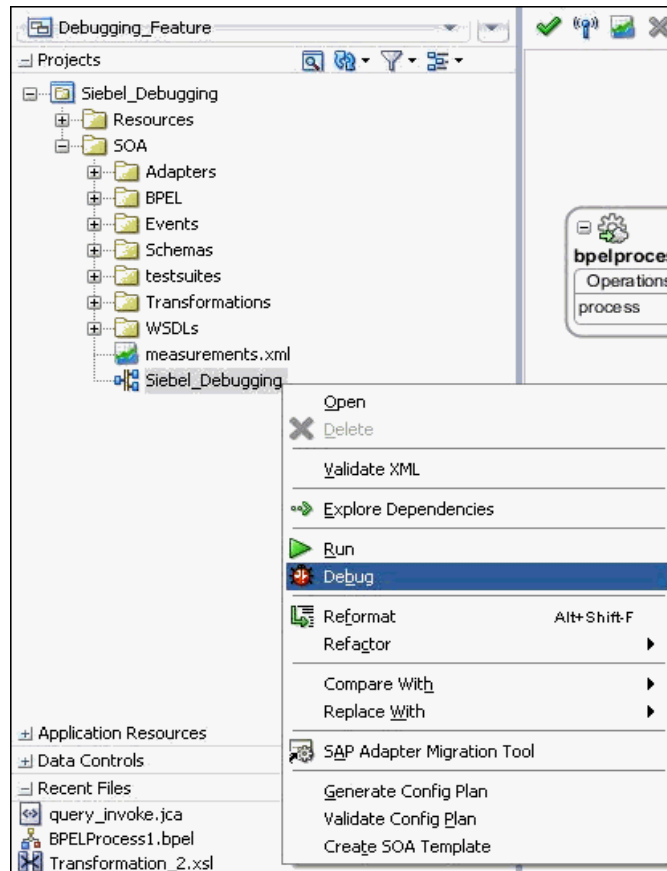
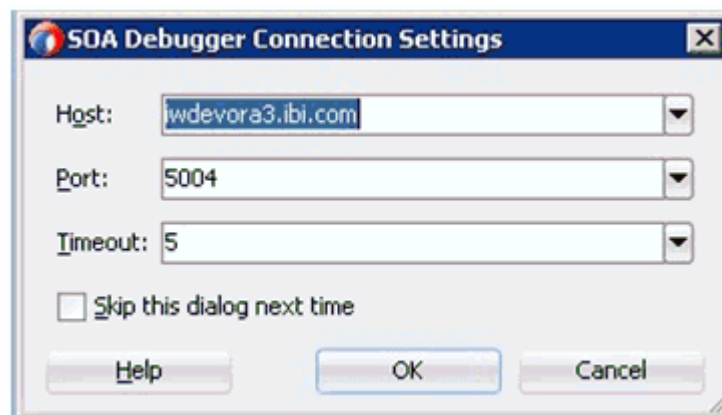


図 9-22 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 9-22 「SOA デバッガ接続設定」ダイアログ



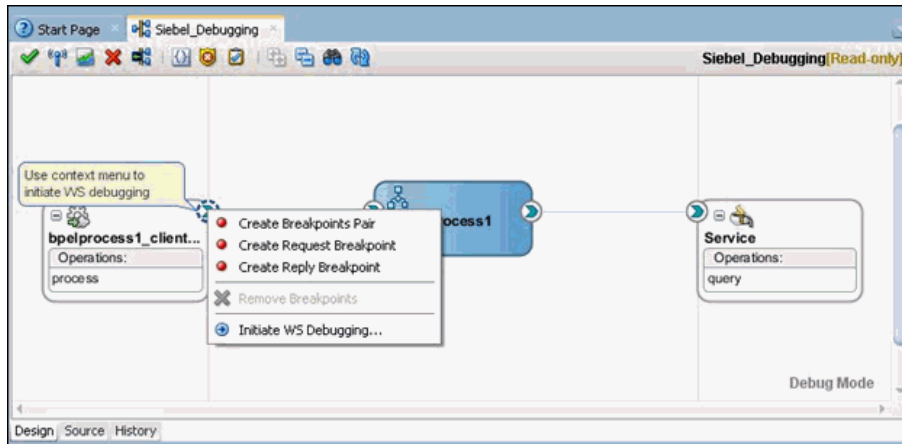
- b. 「OK」をクリックします。

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 9-23 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 9-23 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 9-24 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 9-24 WS デバッグの開始

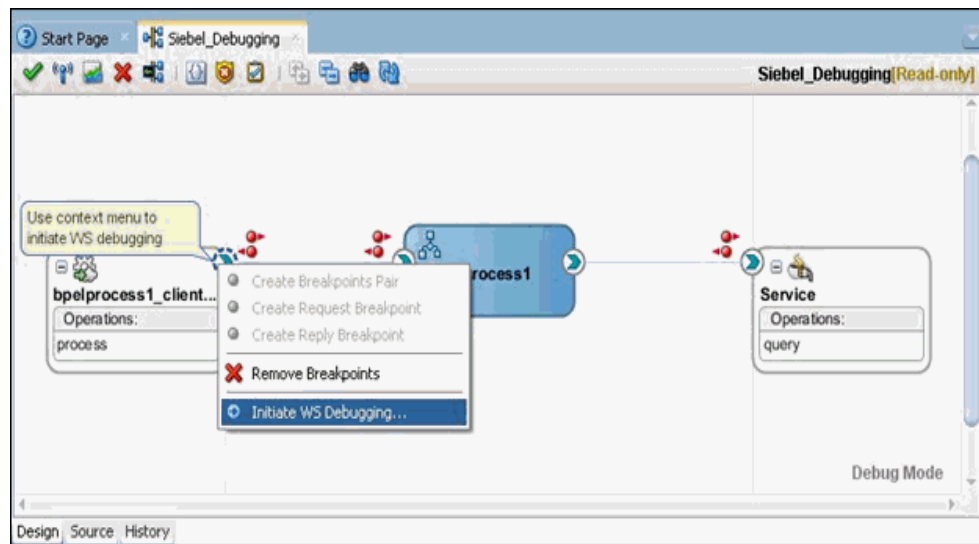
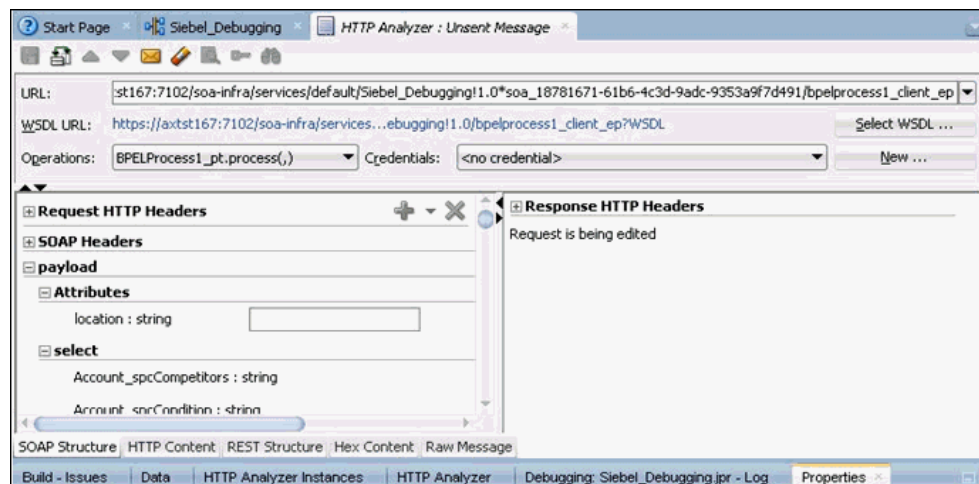


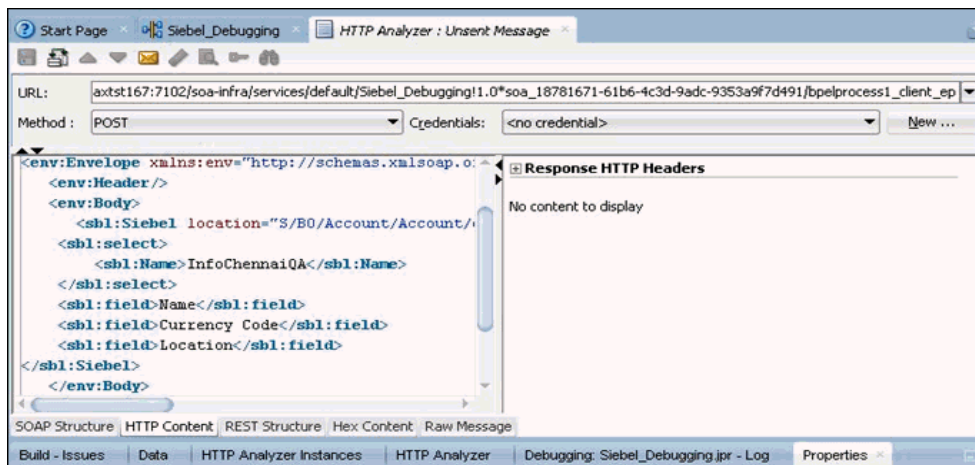
図 9-25 に示すように、「HTTP アナライザ」ダイアログが表示されます。

図 9-25 「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 9-26 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。

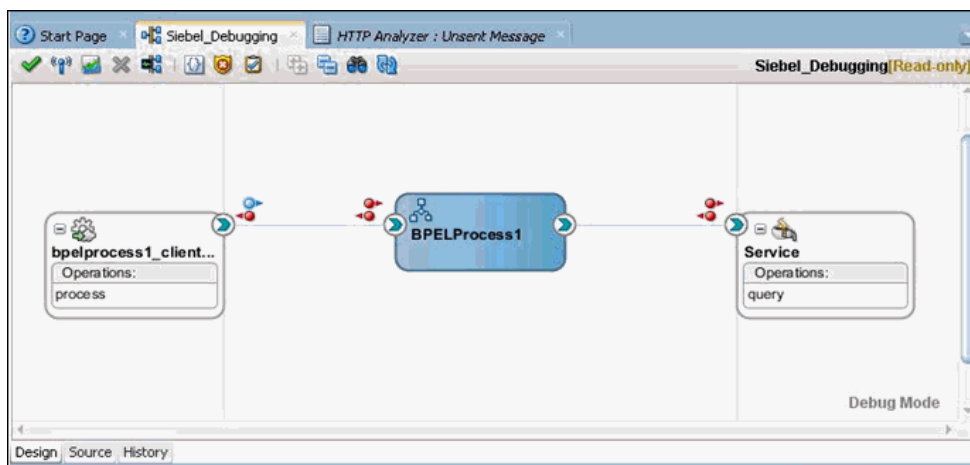
図 9-26 「HTTP コンテンツ」を選択



d. 「リクエストの送信」をクリックします。

図 9-27 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 9-27 BPEL プロセスがブレークポイントで停止



e. 図 9-28 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 9-28 「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。

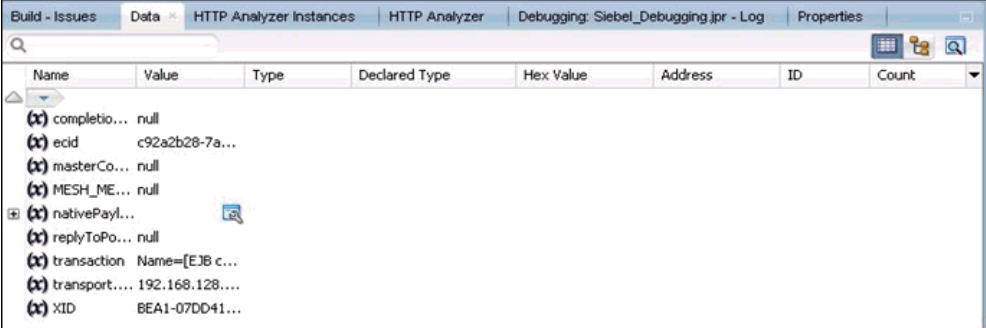
アイコン	説明
	<p>フレームをステップ実行します。</p> <p>これにより、次のブレークポイント (ブレークポイントが設定された BPEL プロセスの受信アクティビティなど) に移動します。ブレークポイントがない場合は、すべてのフレームをステップ実行して最初のフレームに戻ります。</p> <p>[F8] を押してフレームをステップ実行することもできます。</p>
	<p>次の有効な位置までトレース実行します。</p> <p>これは、新しいフレームである場合もあれば、同じフレームの別の位置である場合もあります。</p> <p>[F7] を押してフレームをトレース実行することもできます。</p>
	<p>フレームをステップ・アウトします。</p> <p>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。</p>
	<p>ステップ操作を再開します。</p> <p>[F9] を押して再開することもできます。</p>

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、「デバッガ」、「データ」の順に選択します。

Oracle Application Adapter for Siebel 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 9-29 に示します。

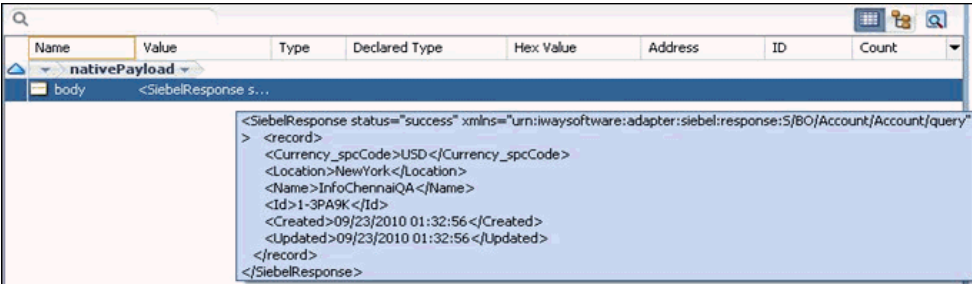
図 9-29 ヘッダー情報



Name	Value	Type	Declared Type	Hex Value	Address	ID	Count
(x) completio...	null						
(x) ecid	c92a2b28-7a...						
(x) masterCo...	null						
(x) MESH_ME...	null						
(x) nativePayl...							
(x) replyToPo...	null						
(x) transaction	Name=[EJB c...						
(x) transport...	192.168.128....						
(x) XID	BEA1-07DD41...						

Oracle Application Adapter for Siebel のレスポンス・ペイロードのサンプルを図 9-30 に示します。

図 9-30 レスポンス・ペイロードのサンプル

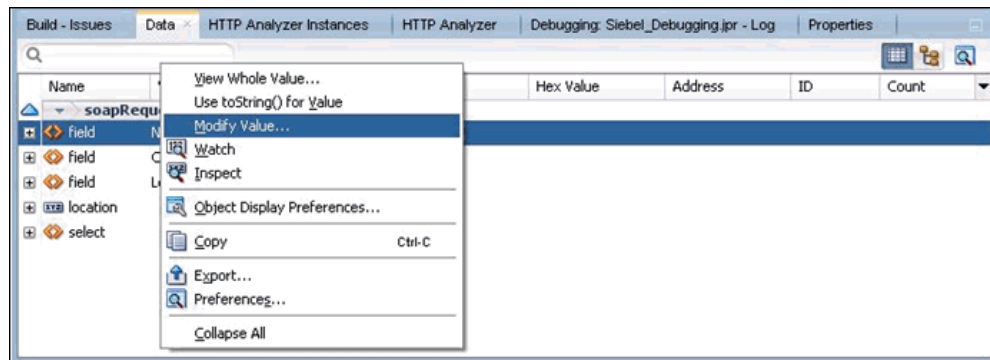


Name	Value	Type	Declared Type	Hex Value	Address	ID	Count
nativePayload	<SiebelResponse s...						
body	<SiebelResponse status="success" xmlns="urn:iwaysoftware:adapter:siebel:response:5/BO/Account/Account/query"> <record> <Currency_spcCode>USD</Currency_spcCode> <Location>NewYork</Location> <Name>InfoChennaiQA</Name> <Id>1-3PA9K</Id> <Created>09/23/2010 01:32:56</Created> <Updated>09/23/2010 01:32:56</Updated> </record> </SiebelResponse>						

注意: 図 9-30 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。

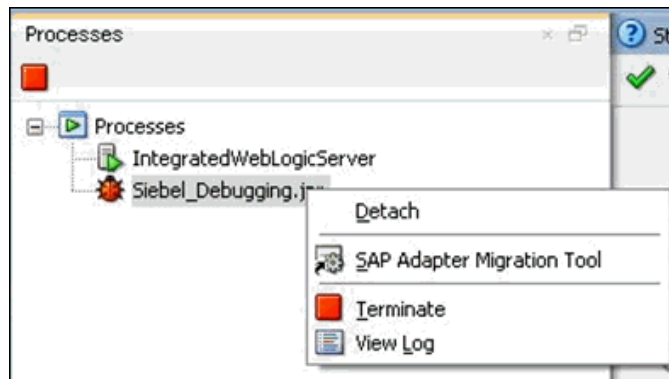
8. リクエスト・ペイロードのコンテンツを変更します。
 - a. 図 9-31 に示すように、SOAP リクエストを展開し、変更するフィールドを右クリックして、コンテキスト・メニューから「値の変更」を選択します。

図 9-31 値の変更



9. 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 9-32 に示すように、「プロセス」タブでプロセスを右クリックし、「連結解除」または「終了」を選択します。

図 9-32 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。

9.3.3.2 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 9-33 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 9-33 デバッグする BPEL プロセスを選択

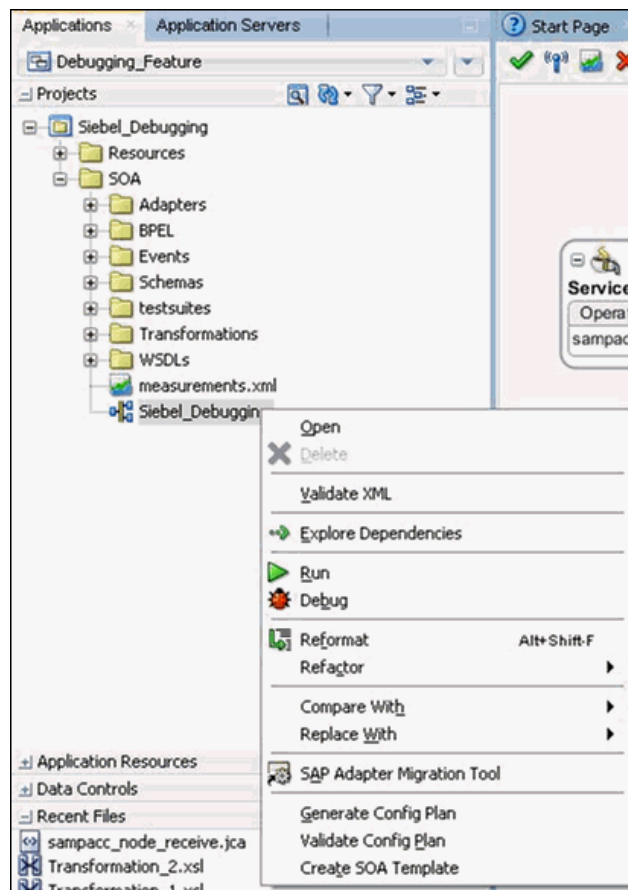
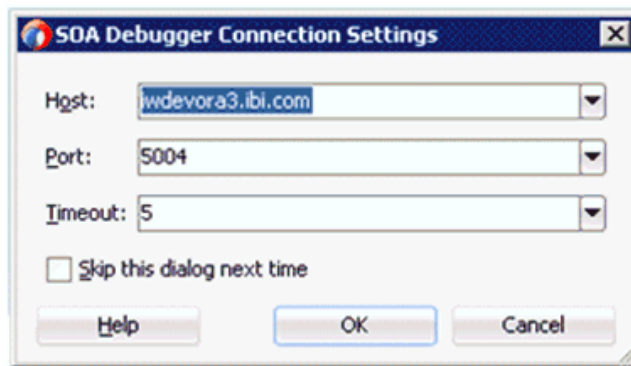


図 9-34 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 9-34 「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。

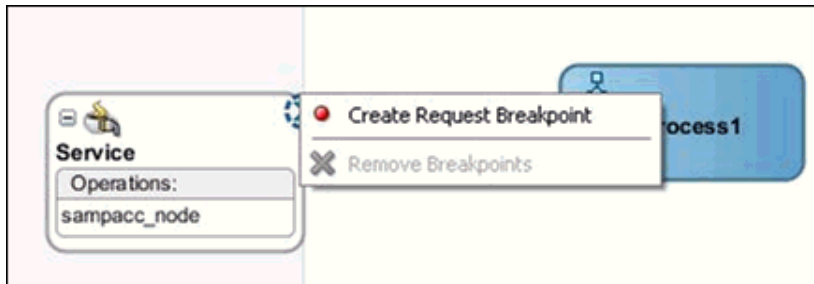
デバッガは iwdevora3.ibi.com 5004 でリモート・プロセスに接続しようとしています。

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。

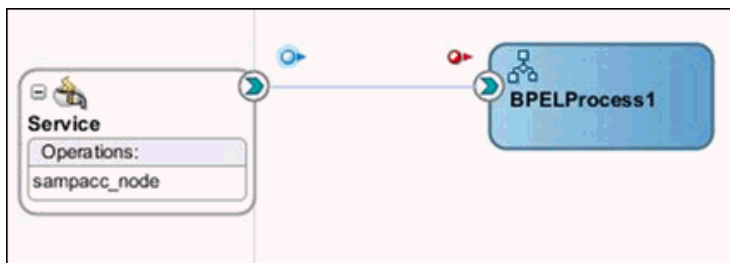
3. 図 9-35 に示すように、ブレークポイントを設定します。

図 9-35 ブレークポイントの設定



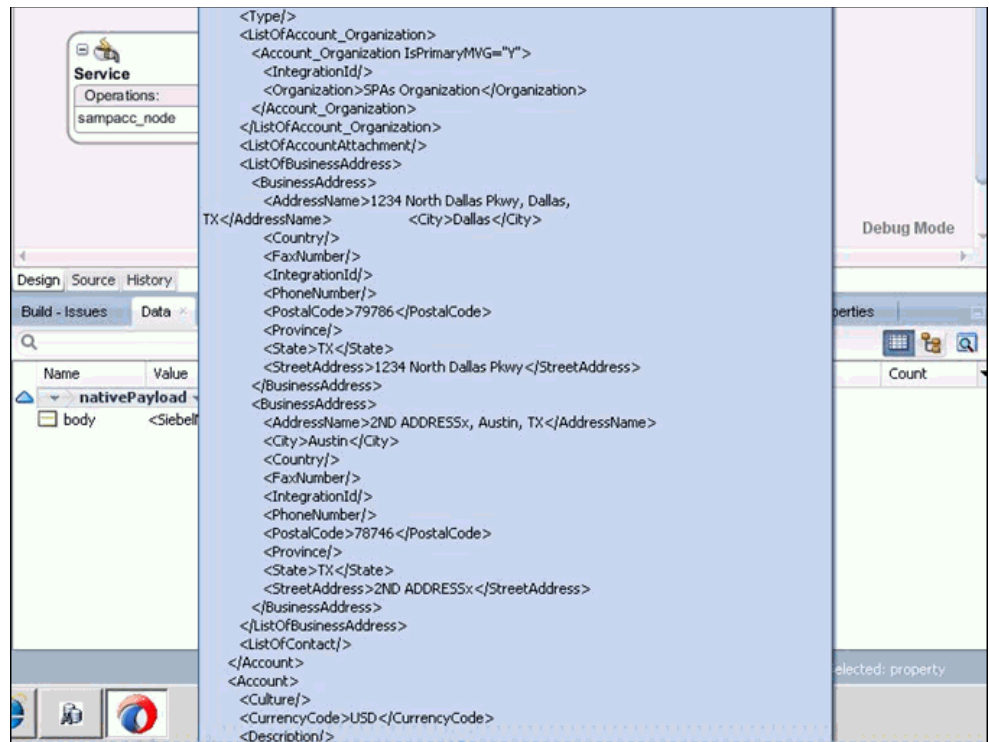
4. ブレークポイントが設定されて、インバウンド・プロセスから (Siebel からトリガーされるなどして) メッセージが受信されると、図 9-36 に示すように、指定したブレークポイントでプロセスが停止します。

図 9-36 ブレークポイントでプロセスを停止



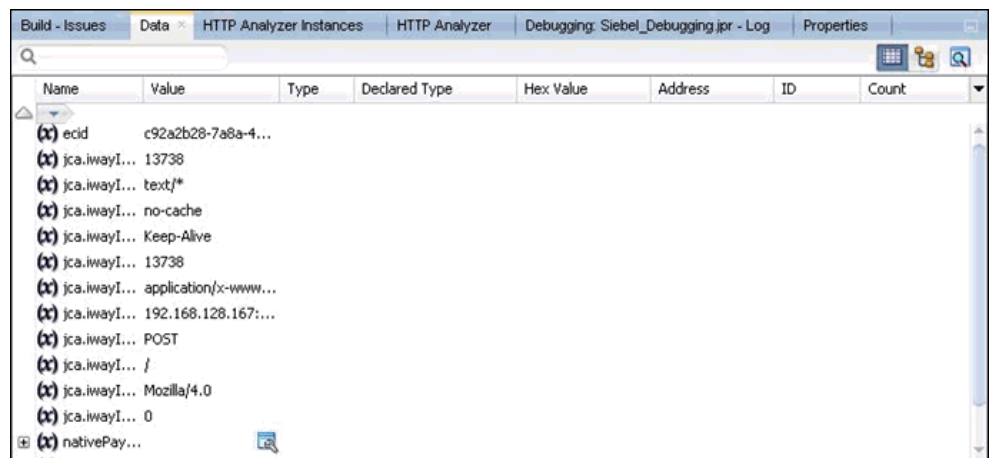
5. リクエストのペイロードとヘッダー情報を表示します。
- a. 図 9-37 に示すように、「ウィンドウ」をクリックし、「デバッガ」、「データ」の順に選択します。

図 9-37 デバッガのデータ



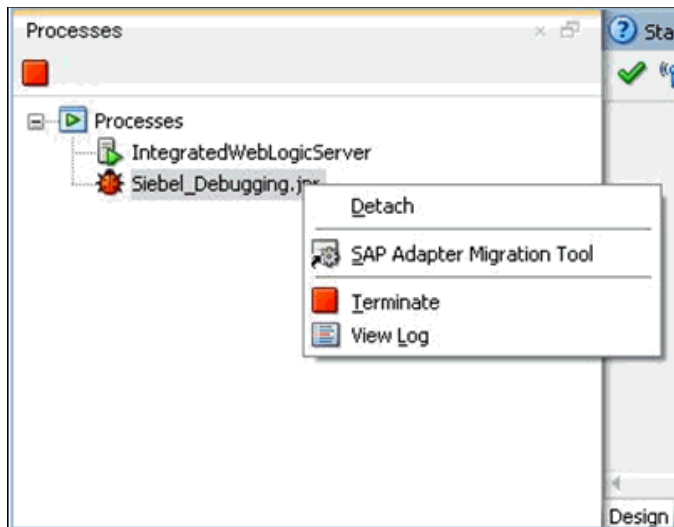
- b. インバウンド・リクエスト・ブレイクポイント (SampleAccount) に対して表示されるヘッダー情報を図 9-38 に示します。

図 9-38 インバウンド・リクエスト・ブレイクポイント (SampleAccount)



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 9-39 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 9-39 デバッグ・セッションの連結解除



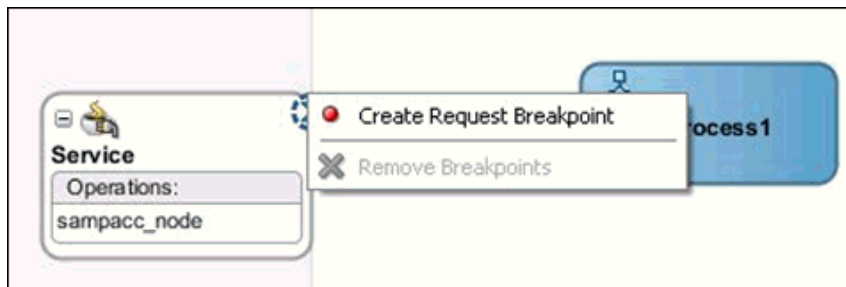
- b. 次のいずれかのオプションを選択します。

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 9-40 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 9-40 連結解除された BPEL プロセス



9.3.4 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。内容は次のとおりです。

- 9.3.4.1 項「前提条件」
- 9.3.4.2 項「Oracle JDeveloper でのアウトバウンド OSB プロセスのデバッグ」
- 9.3.4.3 項「Oracle JDeveloper でのインバウンド OSB プロセスのデバッグ」

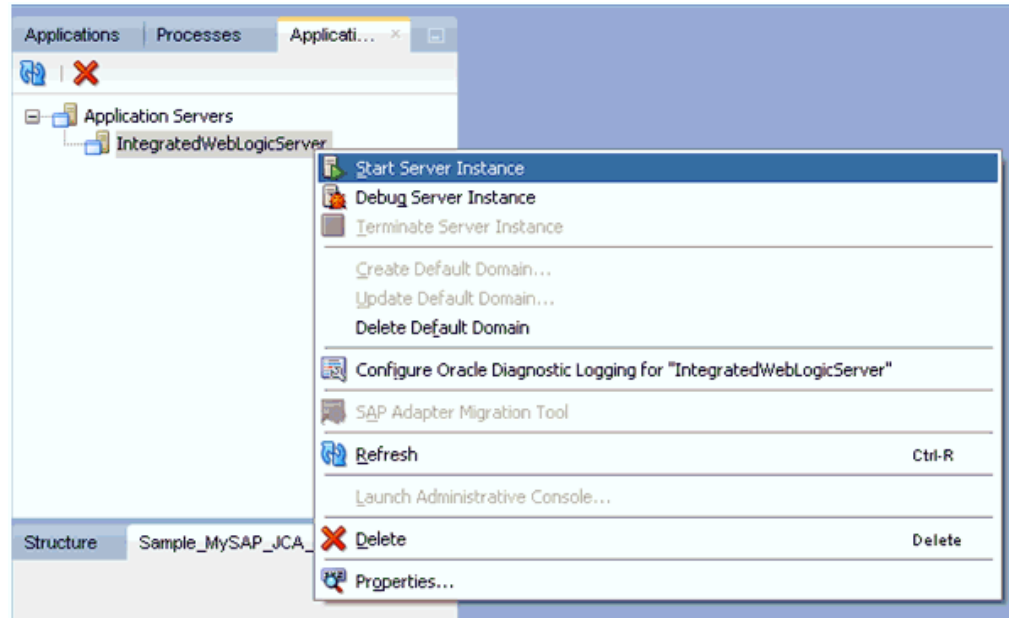
9.3.4.1 前提条件

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。

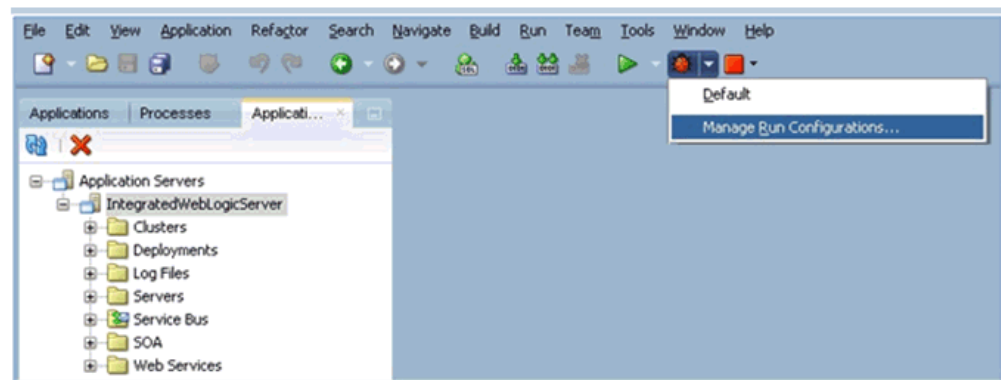
1. Oracle JDeveloper を開き、*IntegratedWebLogicServer* または *startWebLogic.cmd* を起動します。
 - a. 左ペインの「アプリケーション・サーバー」タブをクリックします。
 - b. 図 9-41 に示すように、「アプリケーション・サーバー」ノードの下の「**IntegratedWebLogicServer**」を右クリックし、コンテキスト・メニューから「サーバー・インスタンスの起動」を選択します。

図 9-41 サーバー・インスタンスの起動



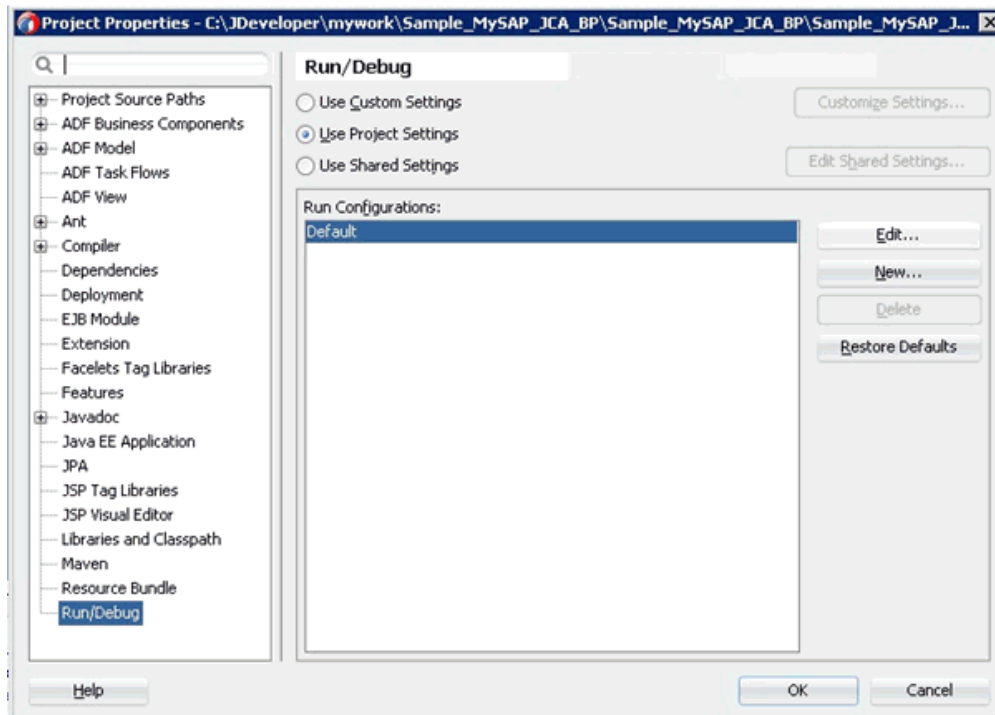
- c. または、*startWebLogic.cmd* を使用して、コマンド・プロンプトから Oracle WebLogic Server を起動します。
2. デバッグ環境を設定します。
 - a. 図 9-42 に示すように、「デバッグ」アイコンの横の下矢印をクリックし、コンテキスト・メニューから「実行構成の管理」を選択します。

図 9-42 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 9-43 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。

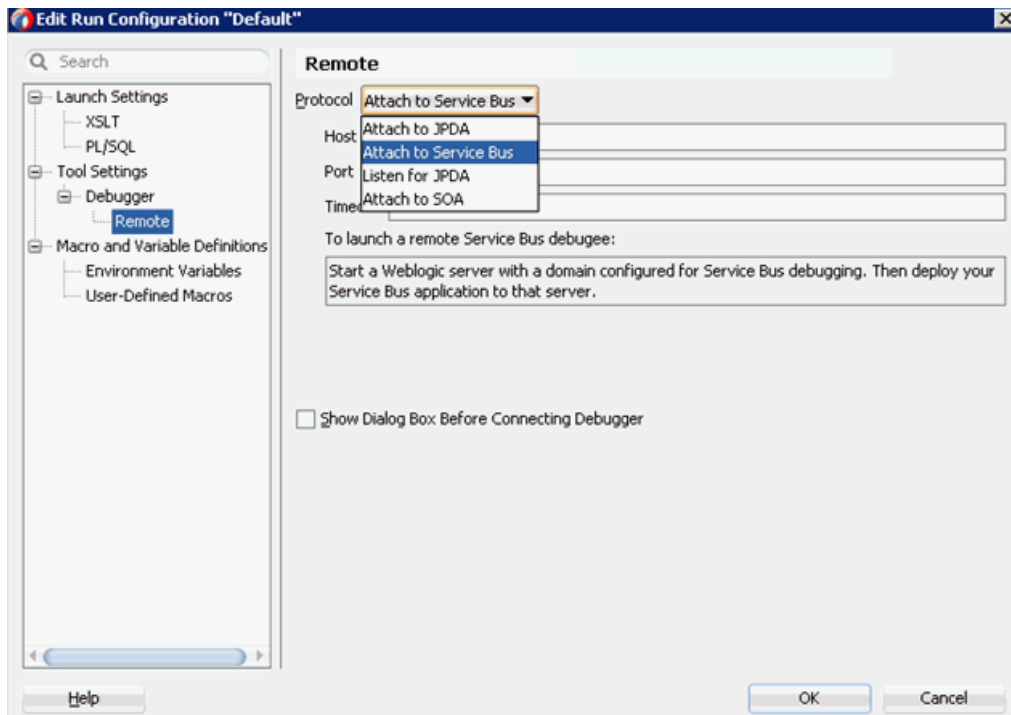
図 9-43 「プロジェクト・プロパティ」 ダイアログ



- 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 9-44 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 9-44 「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。

9.3.4.2 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. 図 9-45 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 9-45 デバッグする OSB プロセスを選択

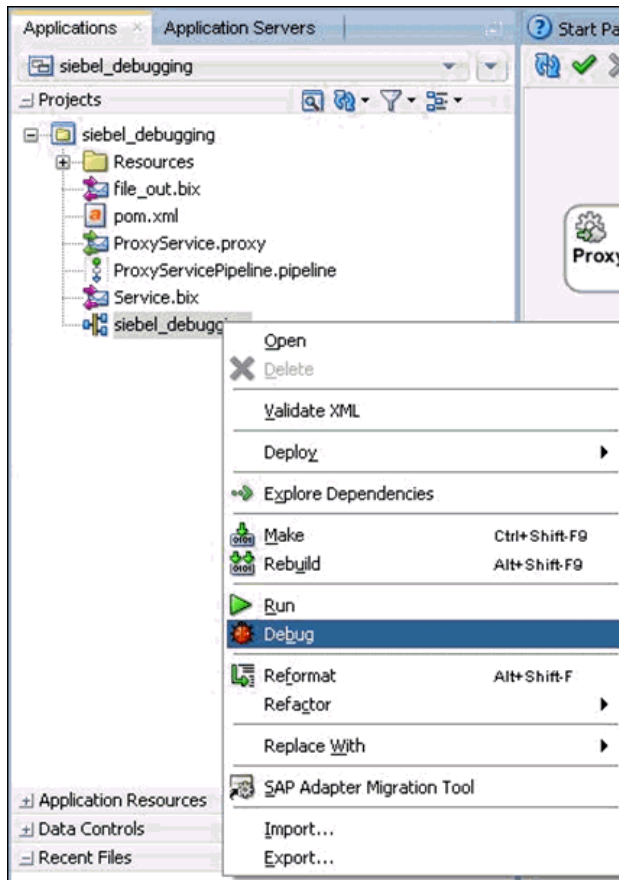
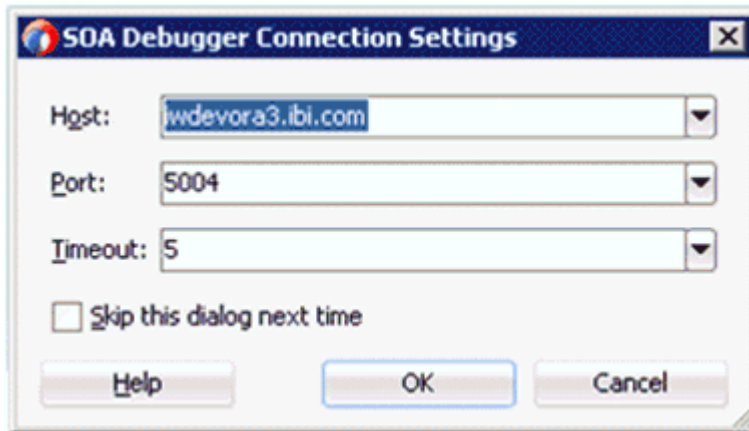


図 9-46 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 9-46 「SOA デバッガ接続設定」ダイアログ



- b. 「OK」をクリックします。

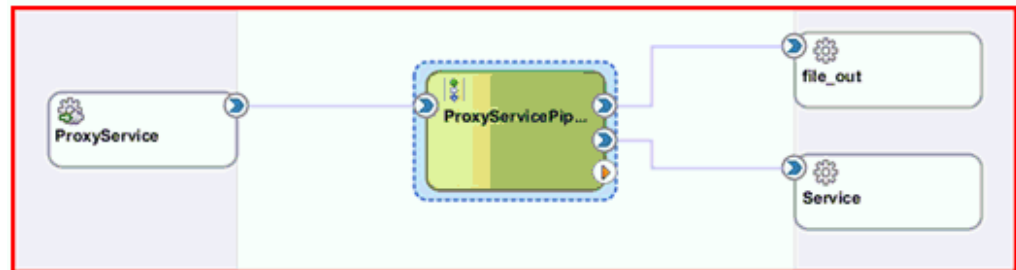
OSB プロセスが SOA デバッガに接続されると、デバッグ・ログに次のメッセージが表示されます。

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。

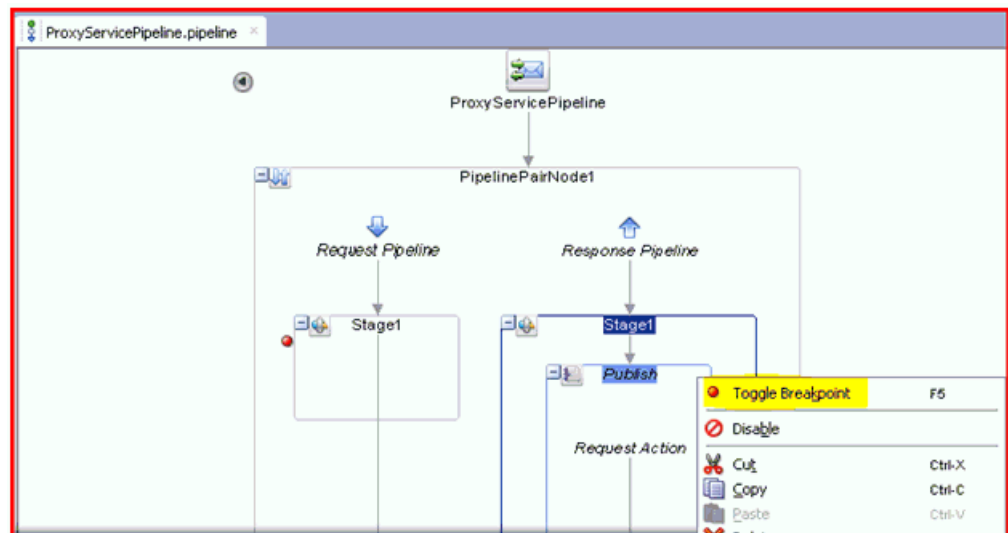
3. ブレークポイントを設定します。
 - a. 図 9-47 に示すように、パイプラインをダブルクリックしてエディタで開きます。

図 9-47 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 9-48 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 9-48 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 9-49 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 9-49 「デバッグ」を選択

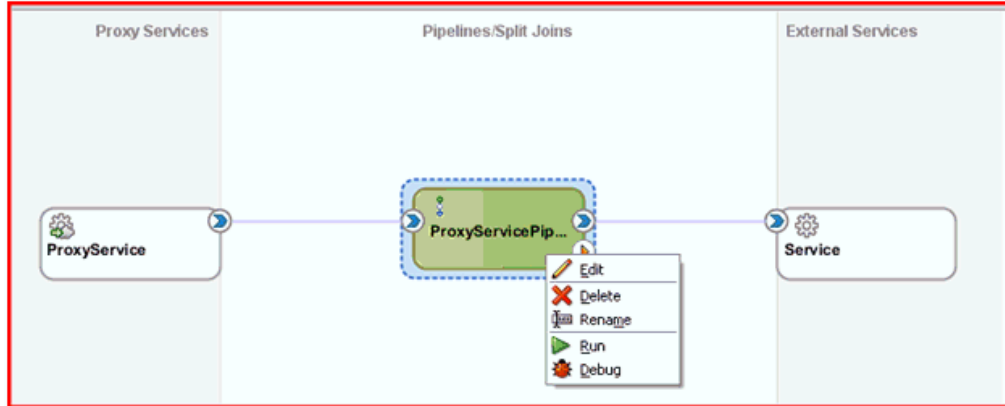
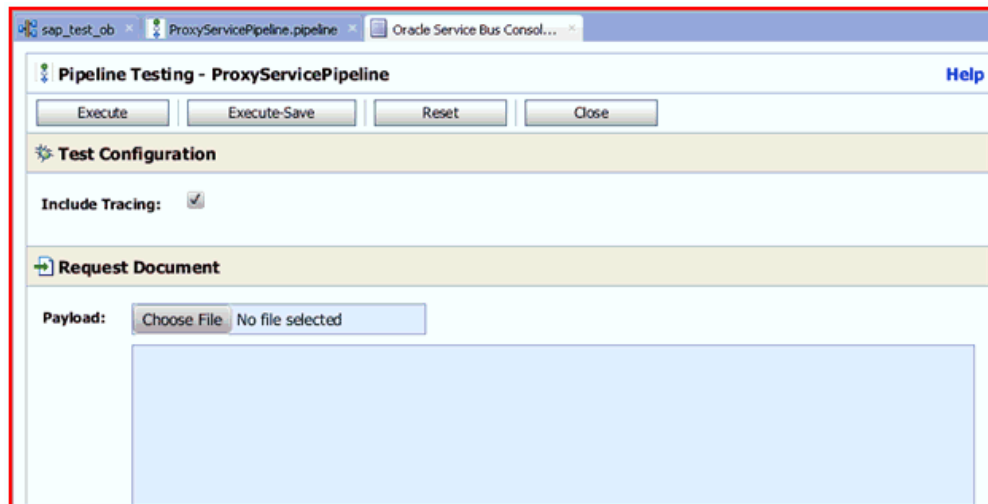


図 9-50 に示すように、プロセスが統合サーバーにデプロイされて、「構成のテスト」ペインが表示されます。

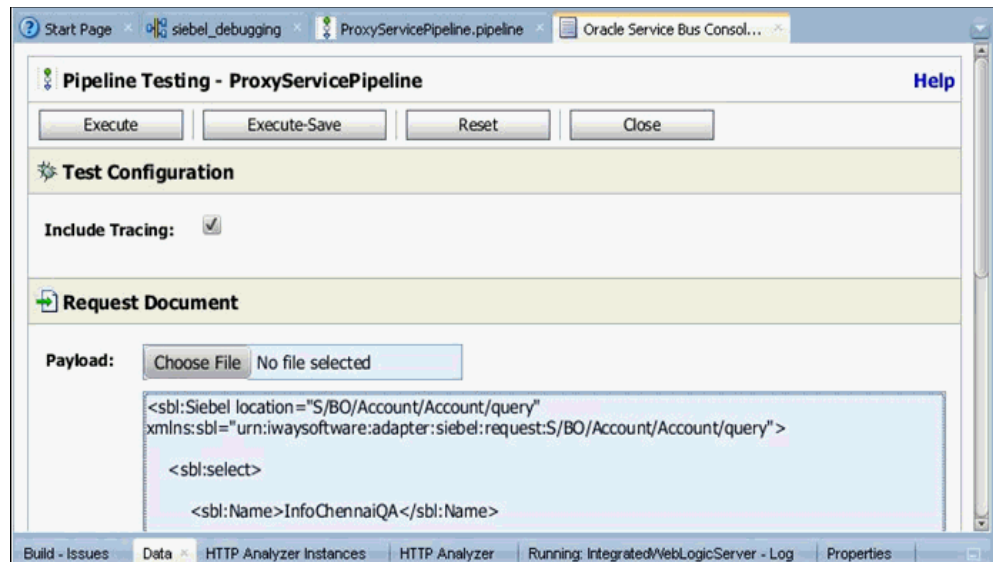
図 9-50 「構成のテスト」ペイン



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 9-51 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。

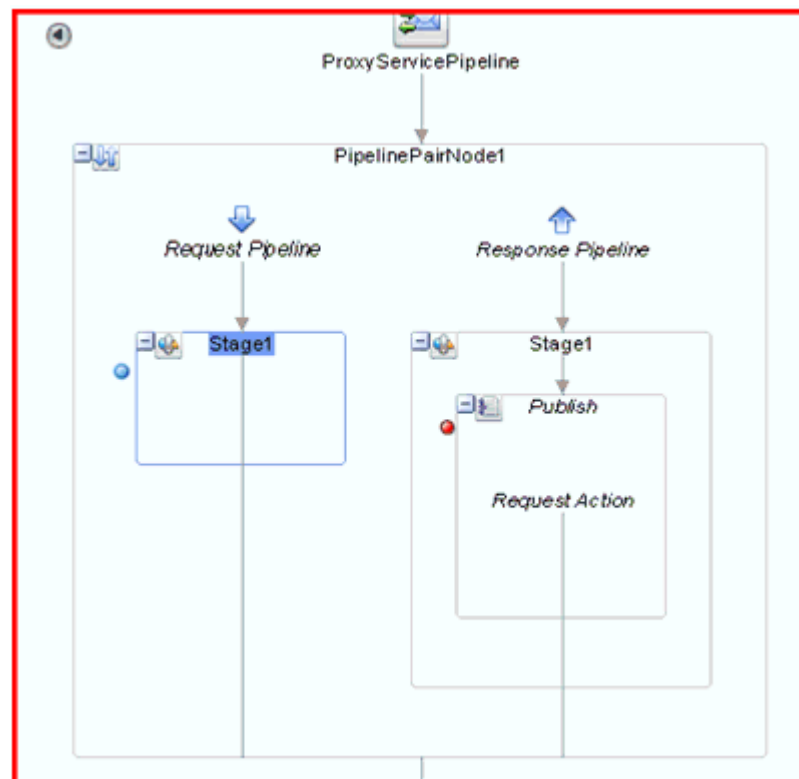
図 9-51 リクエスト・ドキュメントのテスト



- c. 「実行」をクリックします。

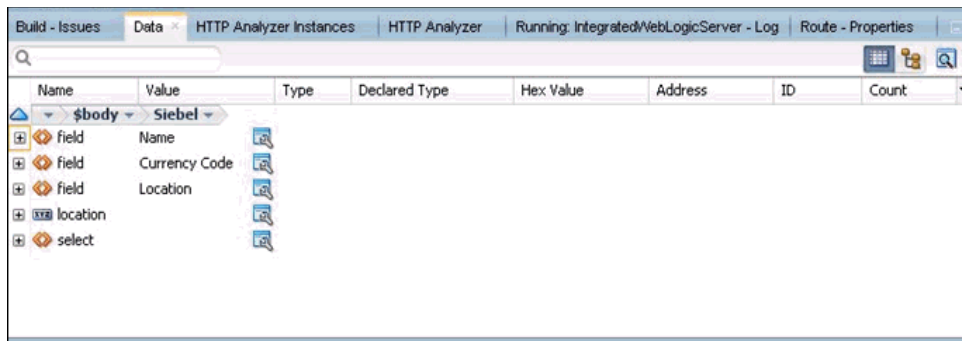
「構成のテスト」ペインでコマンドが実行されますが、図 9-52 に示すように、指定したブレークポイントで OSB プロセスが停止し、青く点滅します。

図 9-52 OSB プロセスがブレークポイントで停止



- d. 図 9-53 に示すように、下部の「データ」タブで、入力が渡されたことを確認します。






図 9-53 「データ」タブ



- e. 図 9-54 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 9-54 「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 9-55 に示すように、「データ」タブで出力を確認できます。

図 9-55 「データ」タブの出力

Name	Value	Type	Declared Type	Hex Value	Address	ID	Count
Created	09/23/2010 0...						
Currency...	USD						
Id	1-3PA9K						
Location	NewYork						
Name	InfoChennaiQA						
Updated	09/23/2010 0...						

また、図 9-56 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 9-56 Oracle Service Bus コンソール

```

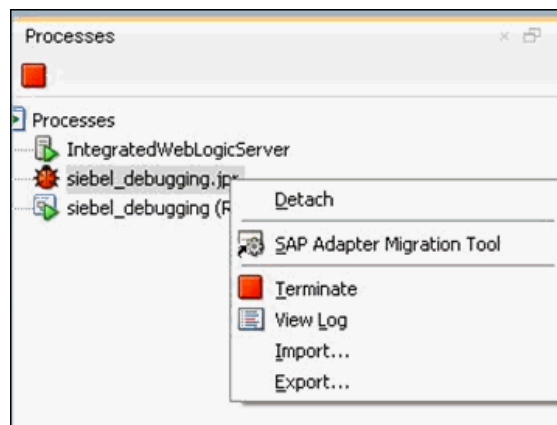
<SiebelResponse status="success" xmlns="urn:iwaysoftware:adapter:siebel:response:S/BO/Account/Account/query">
  <record>
    <Currency_spcCode>USD</Currency_spcCode>
    <Location>NewYork</Location>
    <Name>InfoChennaiQA</Name>
    <Id>1-3PA9K</Id>
    <Created>09/23/2010 01:32:56</Created>
    <Updated>09/23/2010 01:32:56</Updated>
  </record>
</SiebelResponse>

```

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 9-57 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 9-57 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。

9.3.4.3 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 9-58](#) 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 9-58 デバッグする OSB プロセスを選択

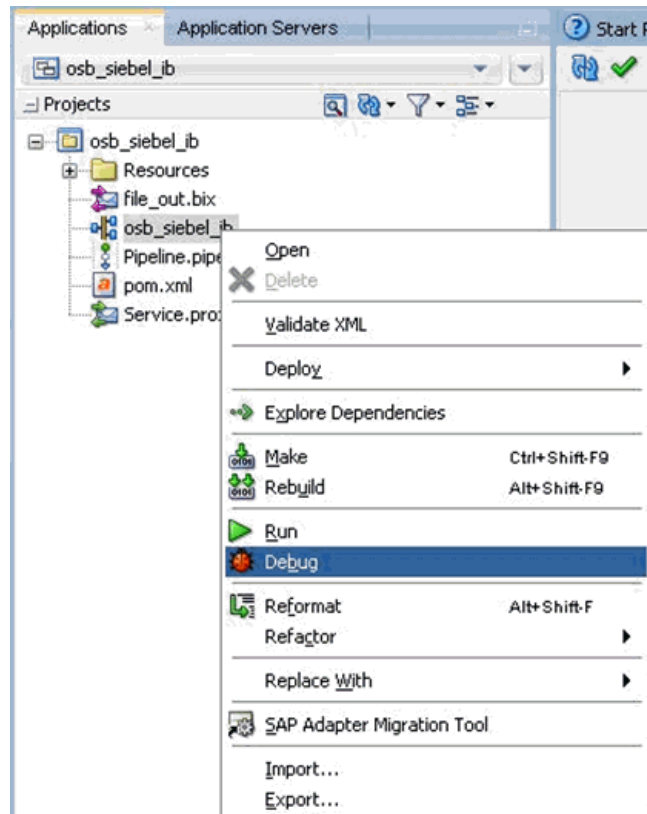
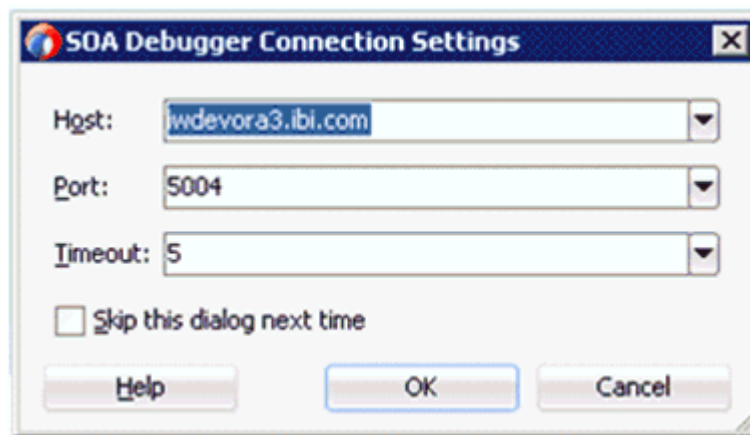


図 9-59 に示すように、「SOA デバッガ接続設定」ダイアログが表示されます。

図 9-59 「SOA デバッガ接続設定」ダイアログ



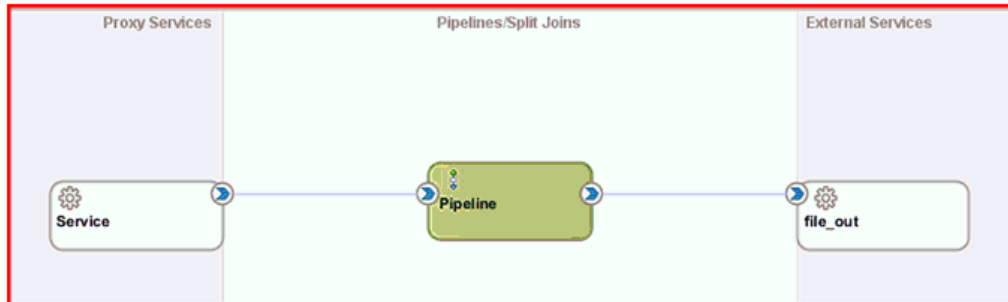
- b. 「OK」をクリックします。

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。

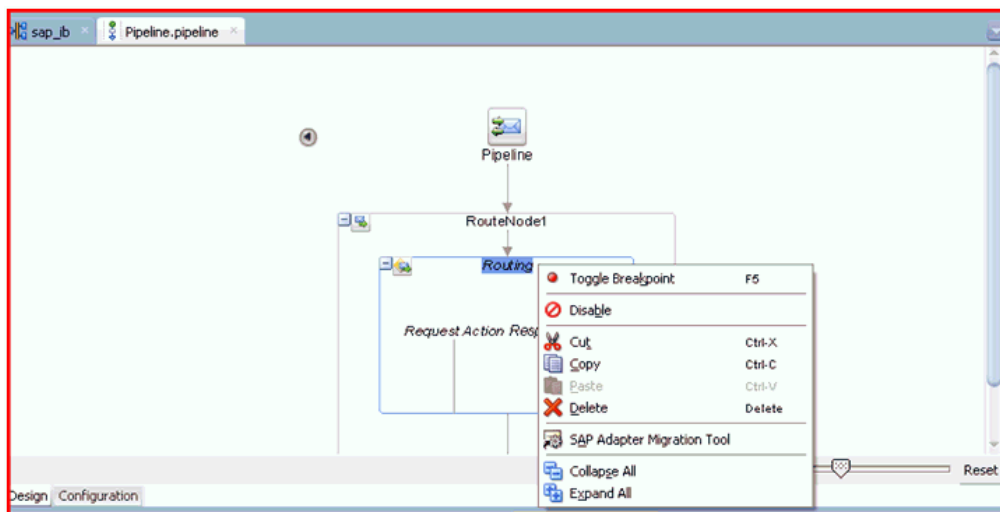
4. ブレークポイントを設定します。
 - a. 図 9-60 に示すように、パイプラインをダブルクリックしてエディタで開きます。

図 9-60 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 9-61 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。

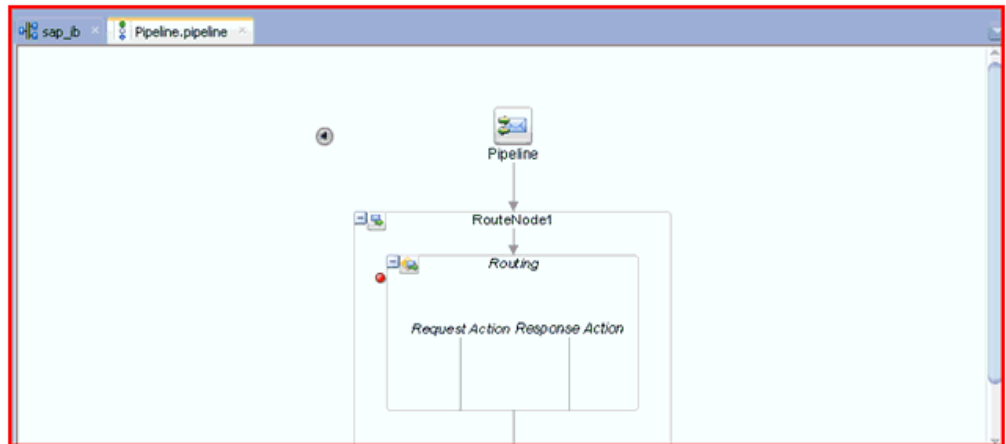
図 9-61 ブレークポイントの切替え



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 9-62 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 9-62 設定されたブレークポイントの赤いアイコン



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。

5. デバッグを開始します。
 - a. 図 9-63 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 9-63 「デバッグ」を選択

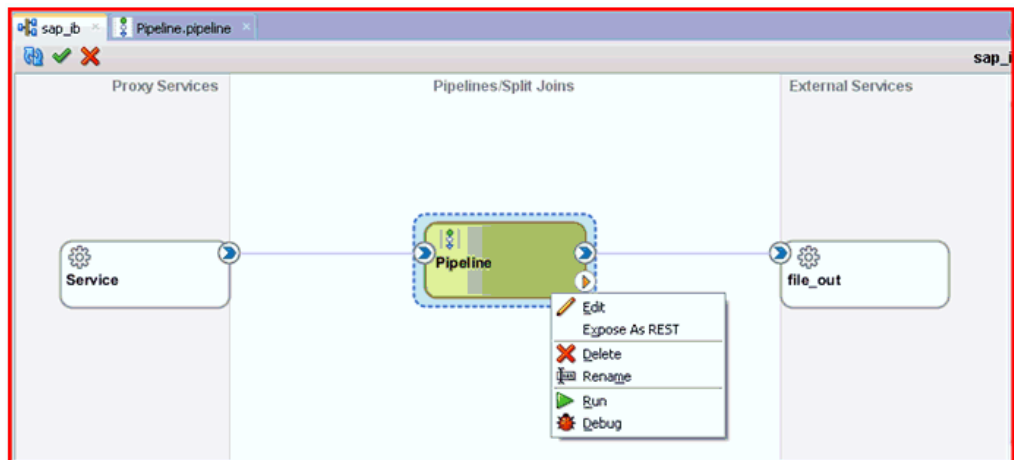
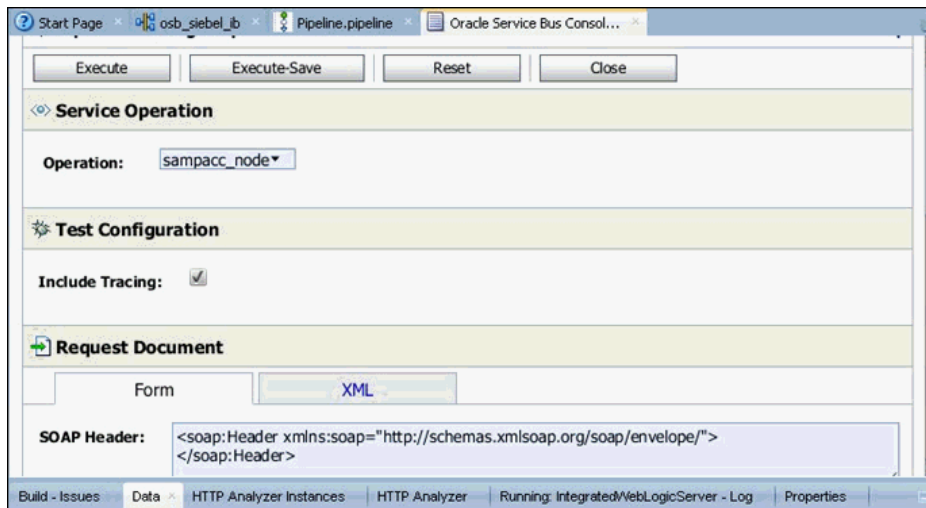


図 9-64 に示すように、プロセスが統合サーバーにデプロイされて、「構成のテスト」ペインが表示されます。

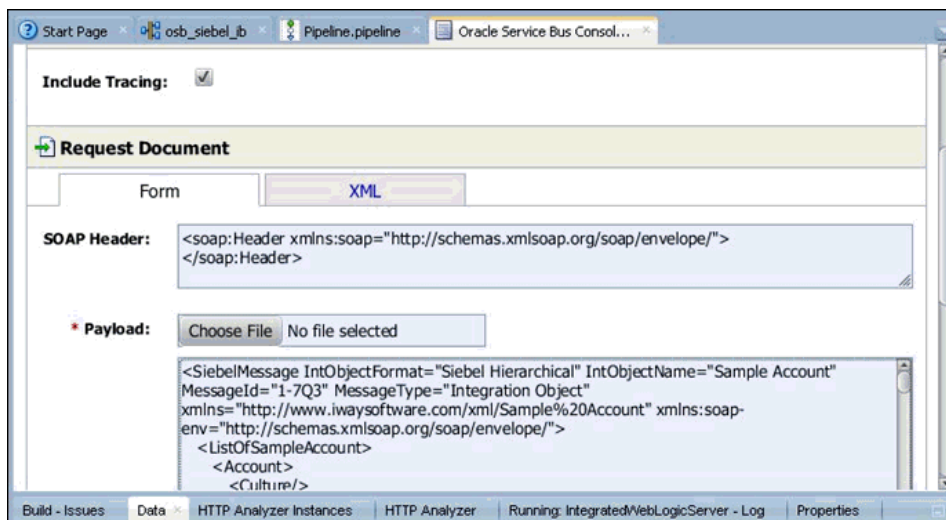
図 9-64 「構成のテスト」ペイン



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 9-65 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。

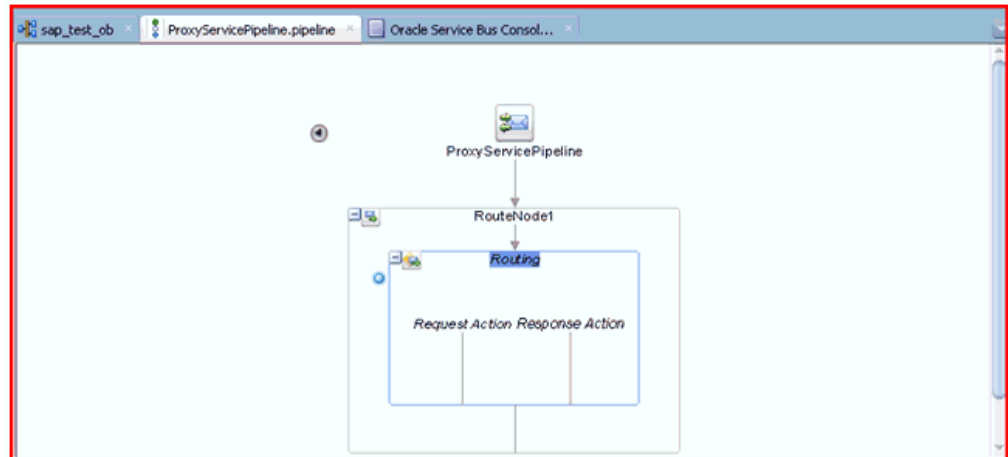
図 9-65 リクエスト・ドキュメントのテスト



- 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 9-66 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 9-66 OSB プロセスがブレークポイントで停止



- d. 図 9-67 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 9-67 「ステップ」オプション



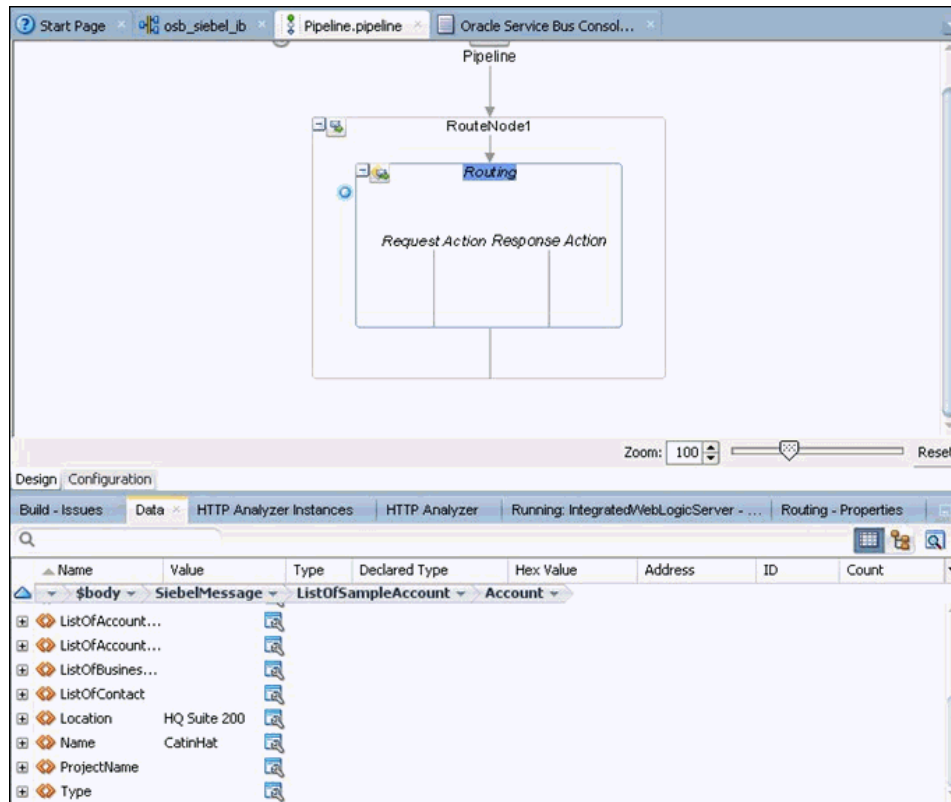
アイコン	説明
	デバッグ・セッションを終了または連結解除します。
	フレームをステップ実行します。 これにより、次のブレークポイント (ブレークポイントが設定された OSB プロセスの受信アクティビティなど) に移動します。ブレークポイントがない場合は、すべてのフレームをステップ実行して最初のフレームに戻ります。 [F8] を押してフレームをステップ実行することもできます。
	次の有効な位置までトレース実行します。 これは、新しいフレームである場合もあれば、同じフレームの別の位置である場合もあります。 [F7] を押してフレームをトレース実行することもできます。
	フレームをステップ・アウトします。 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。

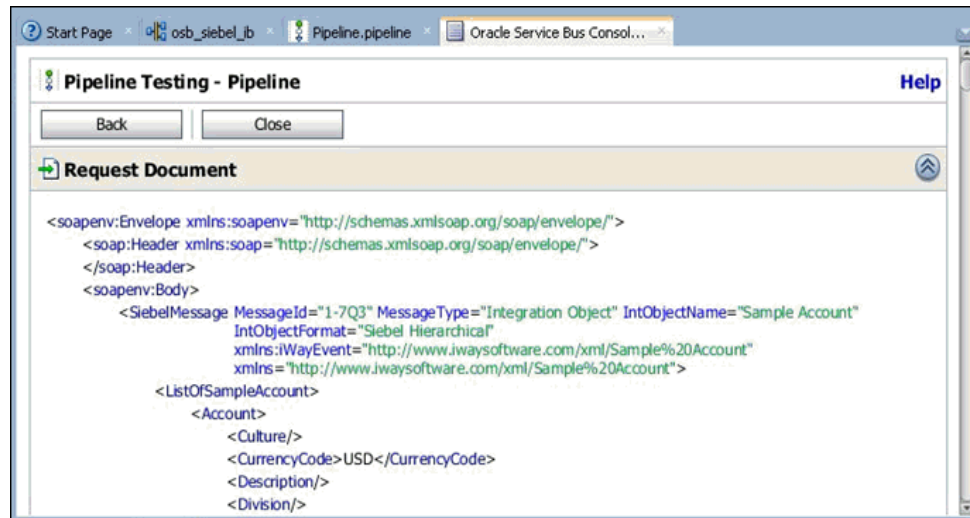
図 9-68 に示すように、「データ」タブでレスポンス・ドキュメントを確認できます。

図 9-68 「データ」タブの出力



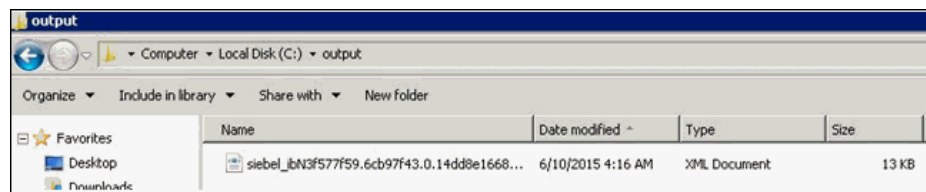
また、図 9-69 に示すように、「Oracle Service Bus コンソール」でレスポンスを確認することもできます。

図 9-69 Oracle Service Bus コンソール



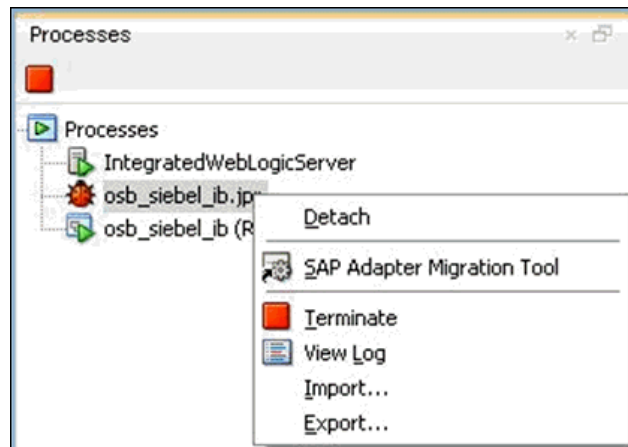
出力は、図 9-70 に示すように、構成済の出力場所にもあります。

図 9-70 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 9-71 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 9-71 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。

9.4 例外フィルタ

この項では、Oracle Application Adapter for Siebel 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。この項にはサンプルのテスト・シナリオが含まれます。

この項では、次のトピックについて説明します。

■ 9.4.1 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・プロセスに対してのみサポートされます。この機能は BSE 構成や J2CA 構成を使用するインバウンド・プロセスではサポートされません。

例外フィルタは `com.ibi.afjca.oracle.AdapterExceptionHandler` クラスを使用して、生成された例外をフィルタリングします。このクラスは例外をフィルタリングし、次のカテゴリに分類します。

- `PCRetriableResourceException`
- `PCResourceException`

次の例外はフォルト・ポリシー・ファイル内で表されます。

- `PCRetriableResourceException` - リモート・フォルト
- `PCResourceException` - バインド・フォルト

9.4.1 例外フィルタの構成

例外フィルタ構成は次のステップとトピックで構成されています。

1. [9.4.1.1 項「WSDL ファイルの生成」](#)
2. [9.4.1.2 項「例外フィルタ機能を使用する BPEL プロセスの作成」](#)
3. [9.4.1.3 項「フォルト・ポリシーおよびフォルト・バインド・ファイルの作成」](#)
4. [9.4.1.4 項「12c のデプロイメントに関する既知の問題の調整」](#)
5. [9.4.1.5 項「例外フィルタ機能を使用する BPEL プロセスのデプロイおよびテスト」](#)

9.4.1.1 WSDL ファイルの生成

WSDL ファイルを生成する手順

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、2-4 ページ [「J2CA 用の構成の作成」](#) を参照してください。
2. Siebel アダプタのターゲットを作成し、ターゲットに接続します。
詳細は、2-5 ページ [「Siebel 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用の WSDL を生成します。
詳細は、2-27 ページ [「WSDL の生成 \(J2CA 構成のみ\)」](#) を参照してください。

9.4.1.2 例外フィルタ機能を使用する BPEL プロセスの作成

例外フィルタ機能を使用する BPEL プロセスを作成する手順

1. JDeveloper を開いて新規の SOA アプリケーションを作成します。
詳細は、4-9 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. 新規 SOA プロジェクトを作成します (`Exception_Filter` など)。

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、4-11 ページ「[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 ファイルとともに) は JDeveloper プロジェクトにインポートされます。

詳細は、4-11 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。

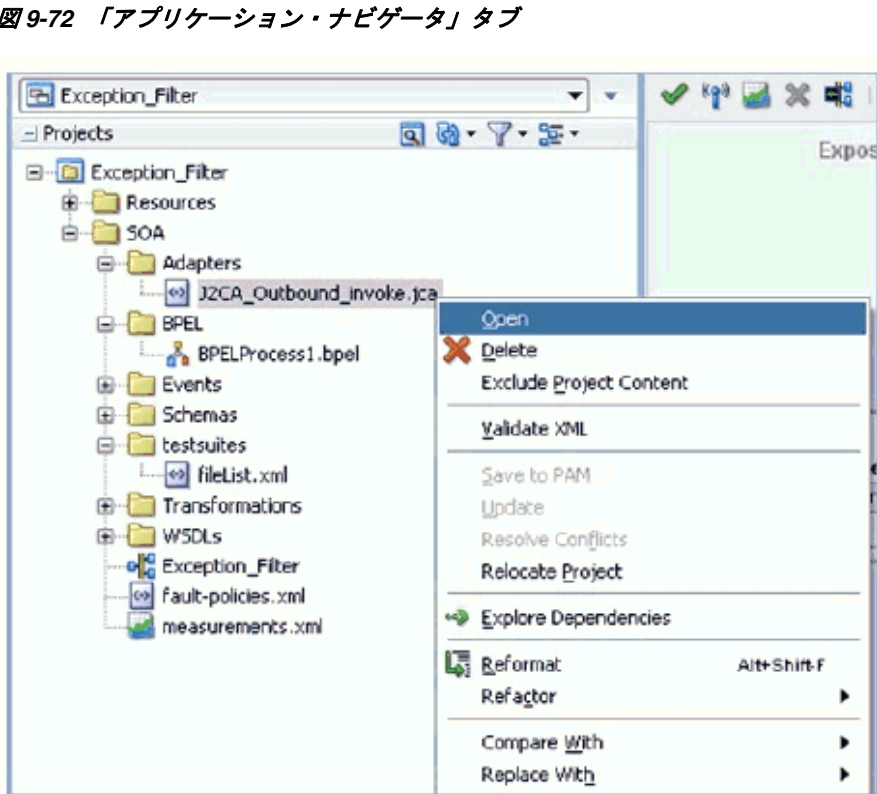
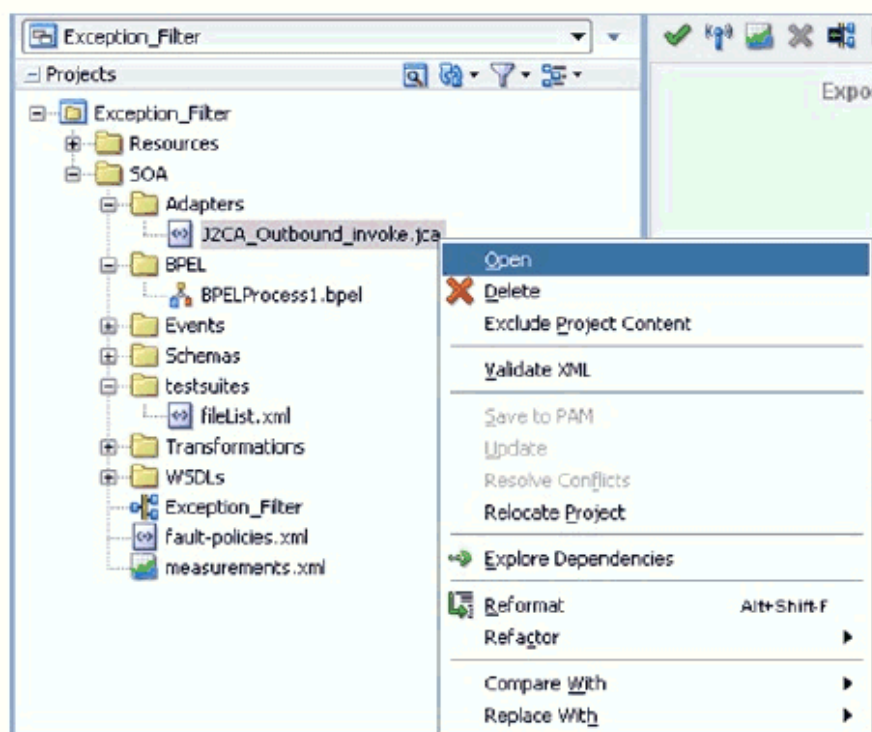
4. インポートされた JCA ファイルを変更します。
 - a.  図 9-72 に示すように、インポートされた JCA ファイルを右クリックし、メニューから「開く」を選択します。

図 9-72 「アプリケーション・ナビゲータ」タブ



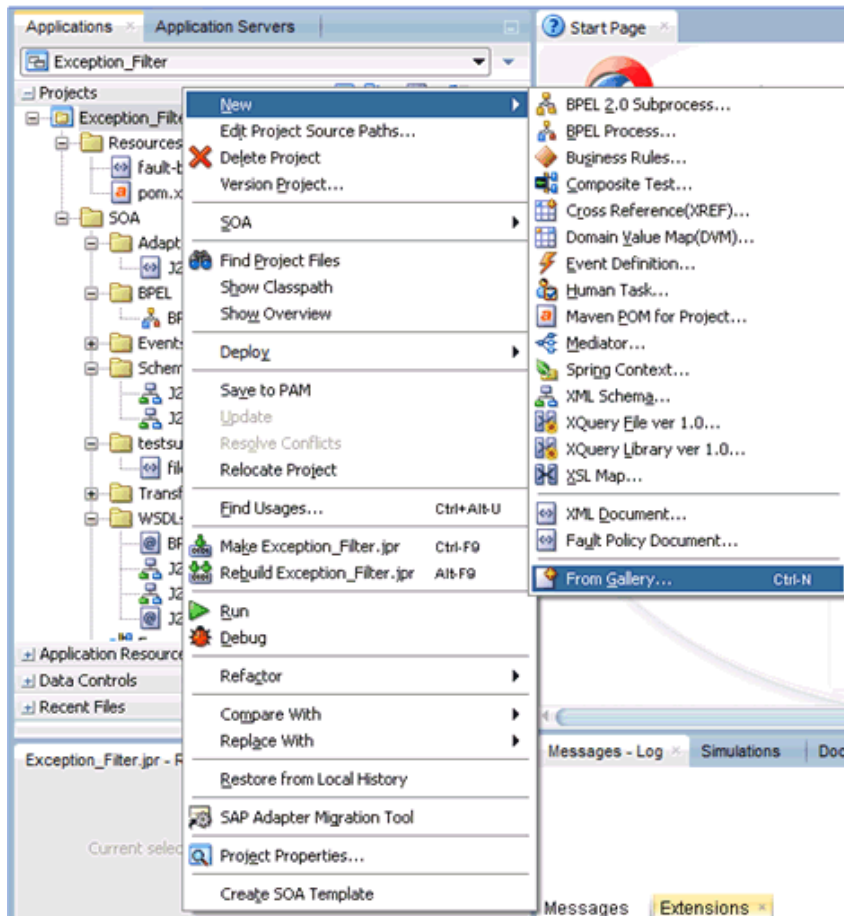
- b. <interaction-spec> 要素に ExceptionFilter プロパティを追加します。次に例を示します。


```
<interaction-spec className="com.ibi.afjca.cci.IWAFInteractionSpec">
  <property name="FunctionName" value="PROCESS"/><property
  name="ExceptionFilter"
  value="com.ibi.afjca.oracle.AdapterExceptionFilter"/></interaction-spec>
```
 - c. 変更した 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、残りの BPEL プロセス作成を続行します。
- 詳細は、4-11 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。

9.4.1.3 フォルト・ポリシーおよびフォルト・バインド・ファイルの作成 フォルト・バインド・ファイルを作成する手順

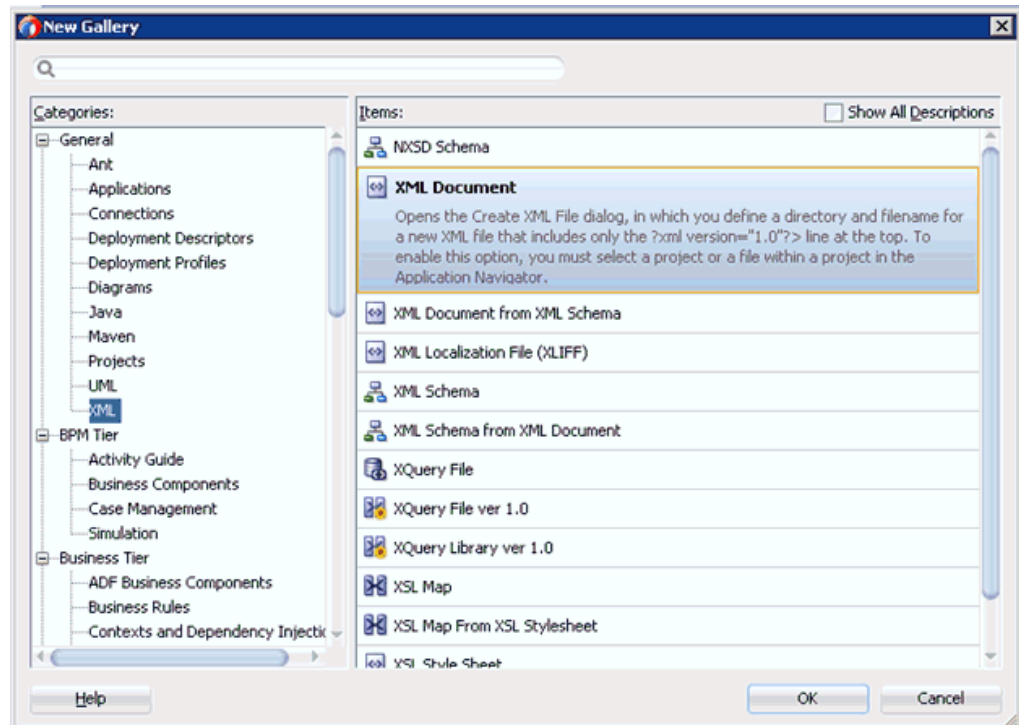
1. 図 9-73 に示すように、作成された SOA プロジェクトを右クリックし (Exception_Filter など)、「新規」を選択してから、「ギャラリーから」をクリックします。

図 9-73 「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 9-74 に示すように、「一般」カテゴリで「XML」をクリックします。

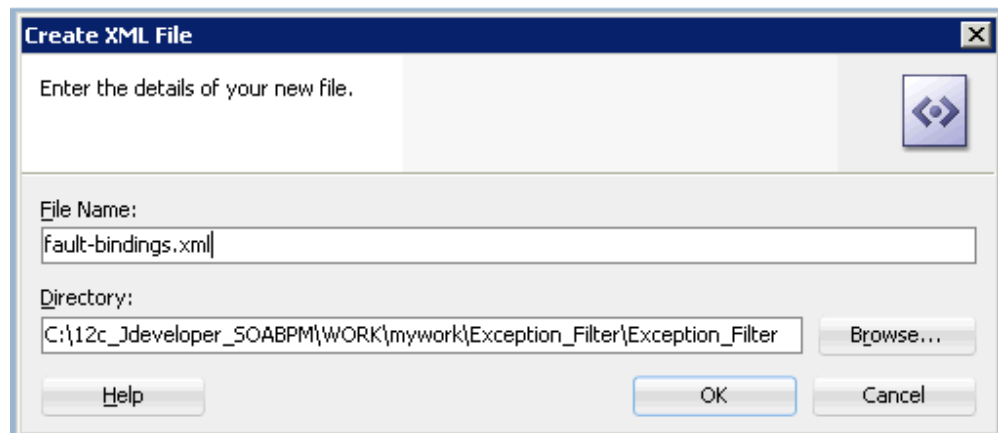
図 9-74 「新規ギャラリー」 ダイアログ



2. 項目から「XML ドキュメント」を選択し、「OK」をクリックします。

図 9-75 に示すように、「XML ファイル」ダイアログが表示されます。

図 9-75 「XML ファイル」ダイアログ



3. 「ファイル名」フィールドで「**fault-bindings.xml**」と入力し、「OK」をクリックします。
4. 適切なフォルト・バインド機能を **fault-bindings.xml** ファイルに追加します。
サンプル **fault-bindings.xml** ファイルの表示については、9-48 ページ「[サンプル Fault-Bindings.xml ファイル](#)」を参照してください。

注意： <name> 要素のパラメータは作成した 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 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>

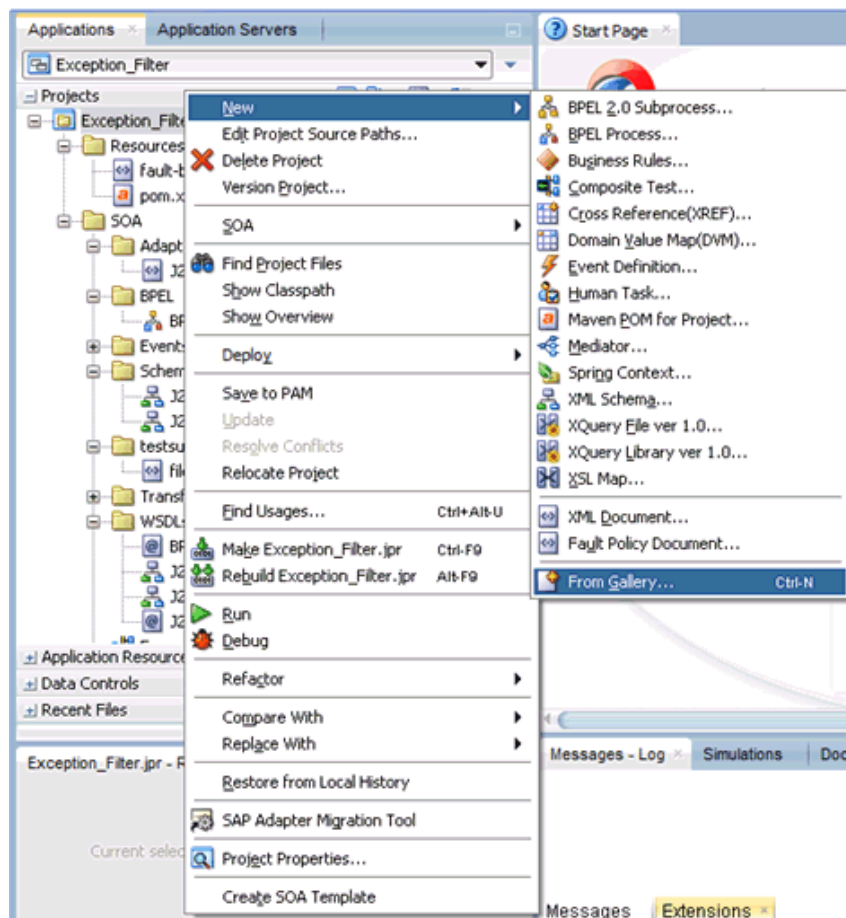
</faultPolicyBindings>
```

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手順

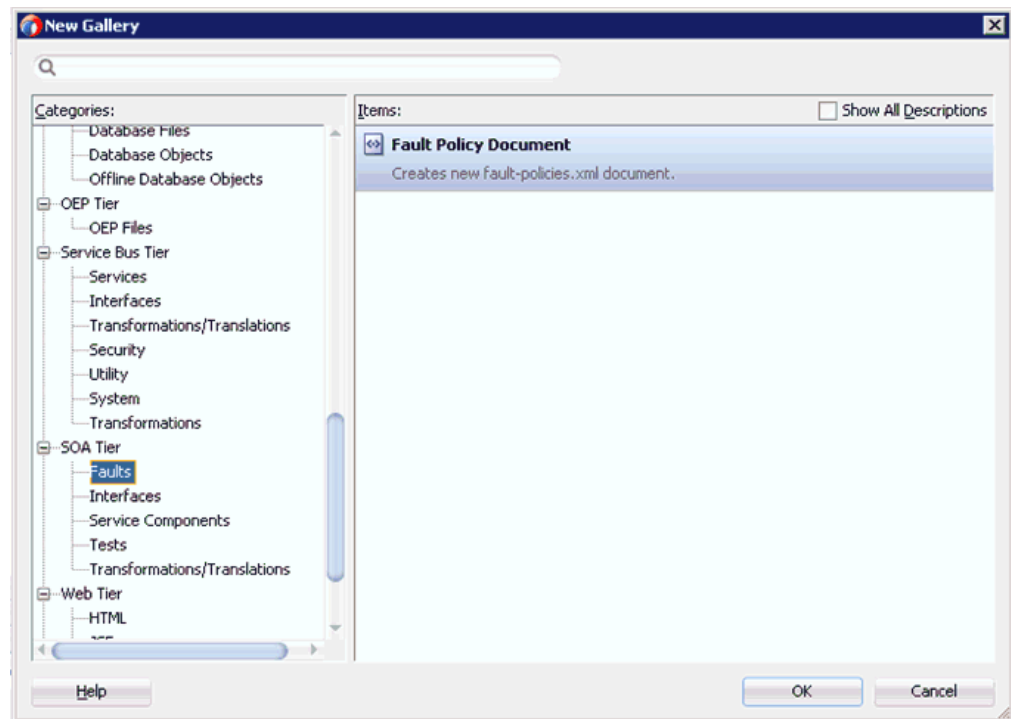
1. 図 9-76 に示すように、作成された SOA プロジェクトを右クリックし (Exception_Filter など)、「新規」を選択してから、「ギャラリーから」をクリックします。

図 9-76 「アプリケーション」タブ



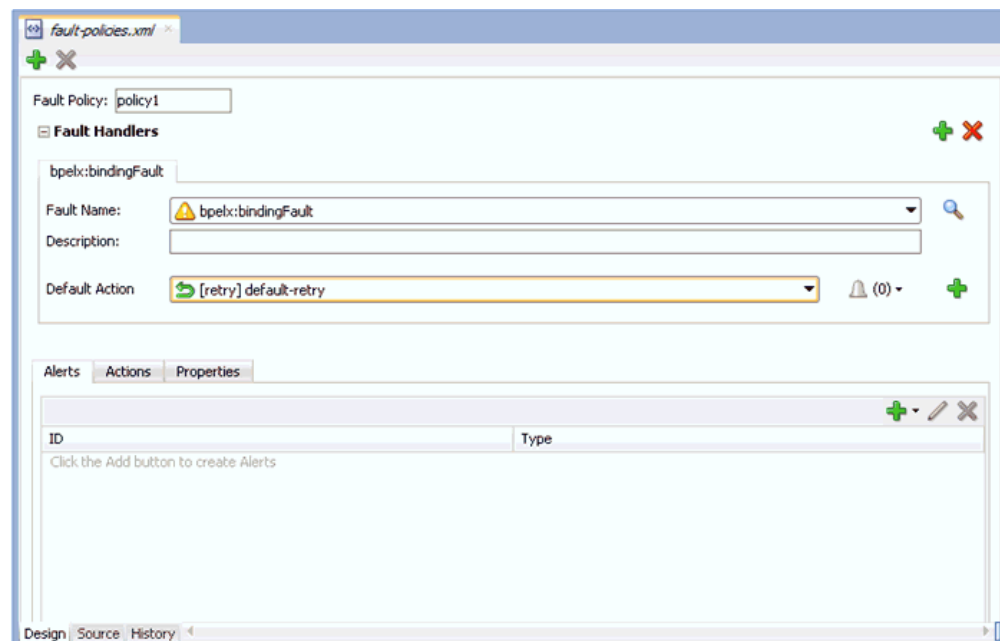
「新規ギャラリー」ダイアログが表示されます。図 9-77 に示すように、「SOA 層」カテゴリで「フォルト」を選択します。

図 9-77 「新規ギャラリー」 ダイアログ



2. 「アイテム」の下にある「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 9-78 に示すように、「fault-policies.xml」タブで、「フォルト名」ドロップダウン・リストから「**bpel:bindingFault**」を選択し、「デフォルト・アクション」ドロップダウン・リストから「**[retry] default-retry**」を選択します。

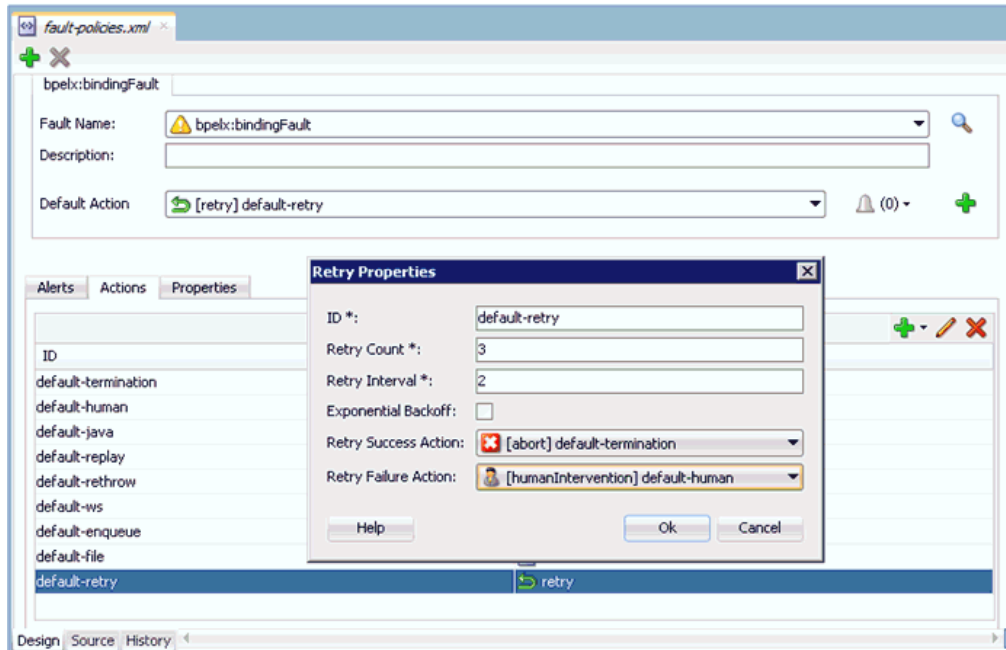
図 9-78 「Fault-policies.xml」タブ



4. 「アクション」タブをクリックし、「default-retry」をダブルクリックします。

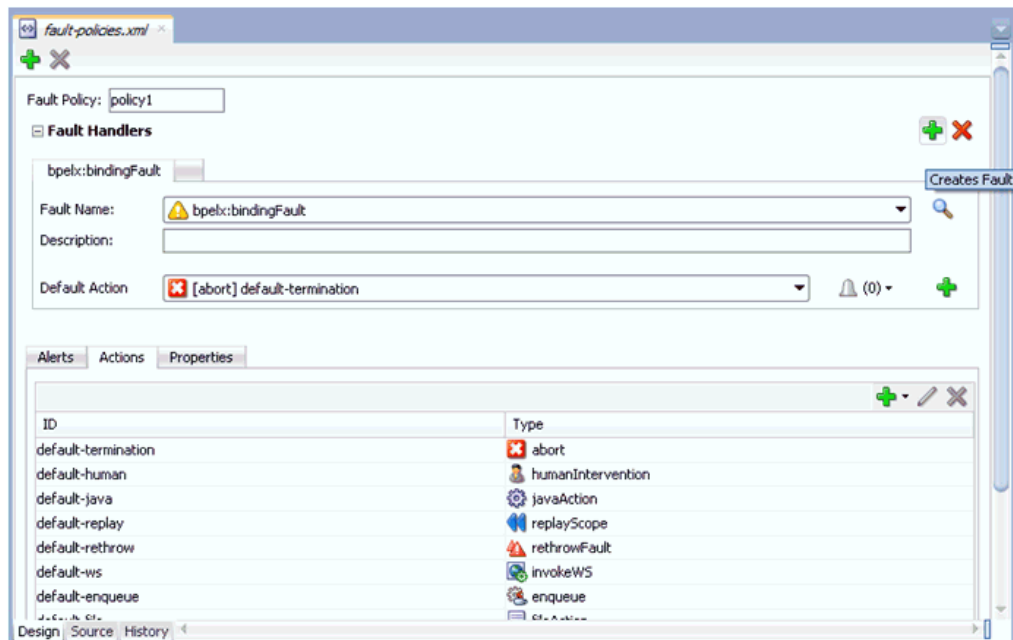
図 9-79 に示すように、「再試行プロパティ」ダイアログ・ボックスが表示されます。

図 9-79 「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・リストから「**[abort] default-termination**」を選択し、「再試行失敗アクション」ドロップダウン・リストから「**[humanIntervention] default-human**」を選択します。
6. 「OK」をクリックします。
7. 図 9-80 に示すように、「追加」をクリックして別のフォルト・ハンドラを作成します。

図 9-80 「Fault-policies.xml」タブ



8. 「fault-policies.xml」タブで、「フォルト名」ドロップダウン・リストから「**bpelx:remoteFault**」を選択し、「デフォルト・アクション」ドロップダウン・リストから「**[abort] default-termination**」を選択します。
9. 図 9-81 に示すように、「アクション」タブで、「追加」をクリックし、「**retry**」を選択します。

図 9-81 「アクション」タブ

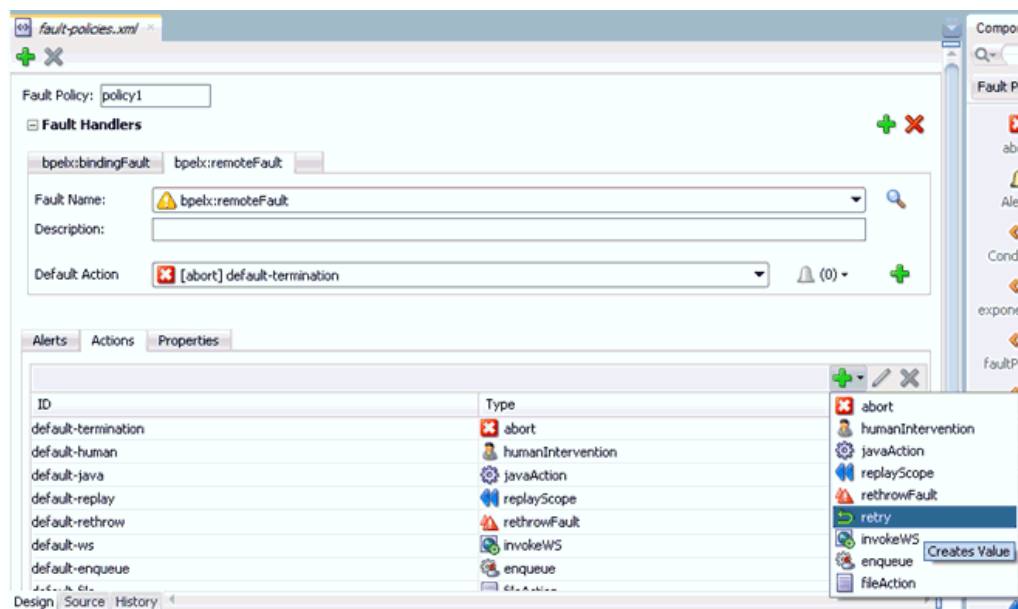
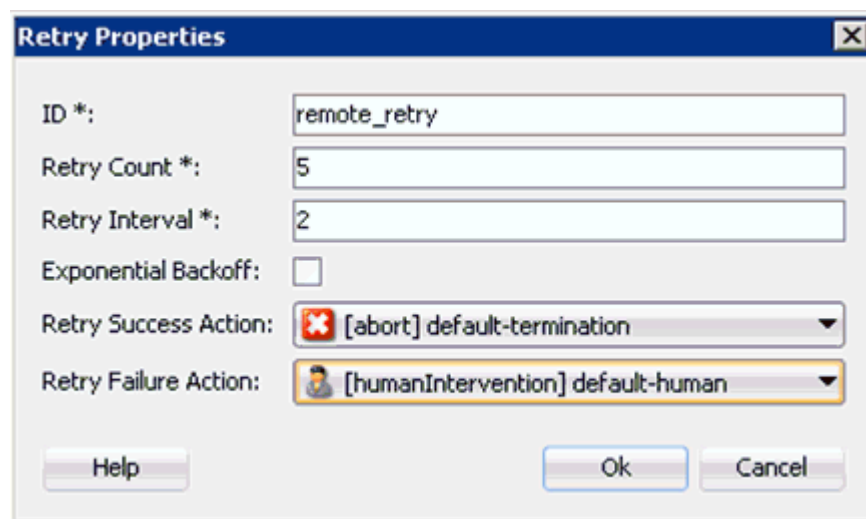


図 9-82 に示すように、「再試行プロパティ」ダイアログが表示されます。

図 9-82 「再試行プロパティ」ダイアログ・ボックス

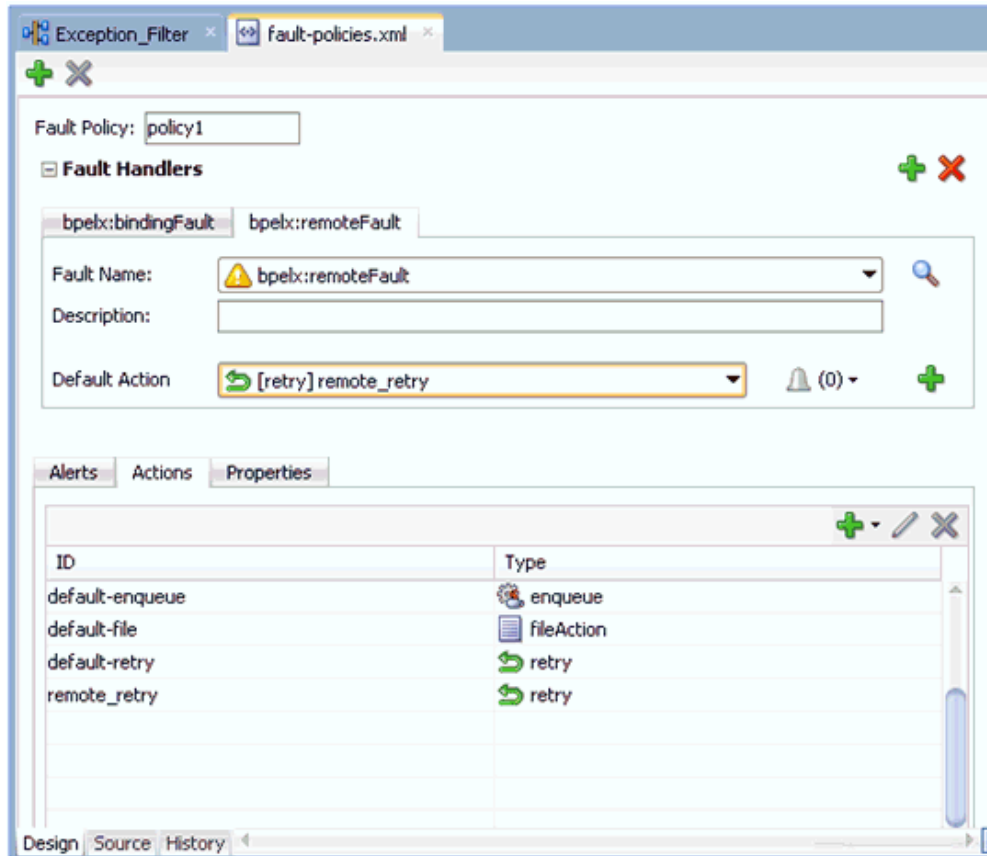


10. 「ID」の各フィールドに値を入力します。
11. 「再試行成功アクション」ドロップダウン・リストから「**[abort] default-termination**」を選択し、「再試行失敗アクション」ドロップダウン・リストから「**[humanIntervention] default-human**」を選択します。
12. 「OK」をクリックします。

作成した再試行 ID は「アクション」タブの下に表示されます。

図 9-83 に示すように、「デフォルト・アクション」ドロップダウン・リストから新しく作成した再試行 ID(remote_retry など) を選択します。

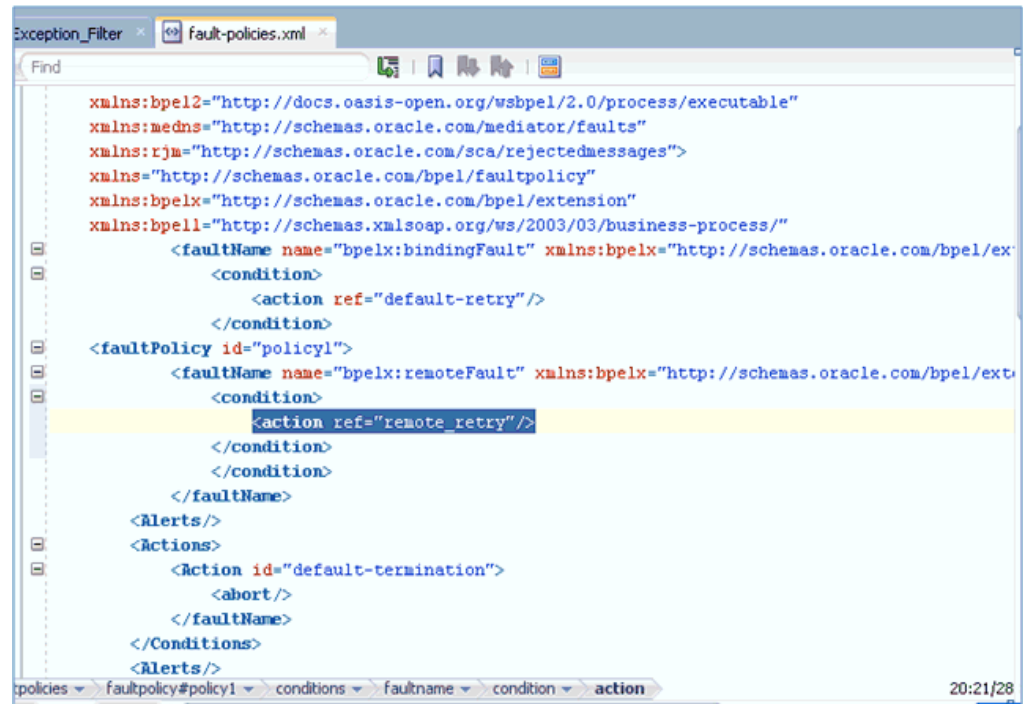
図 9-83 「Fault-policies.xml」タブ



13. 「すべて保存」をクリックします。

14. 図 9-84 に示すように、「ソース」タブをクリックし、フォルト・ポリシーが正しく追加されたことを確認します。

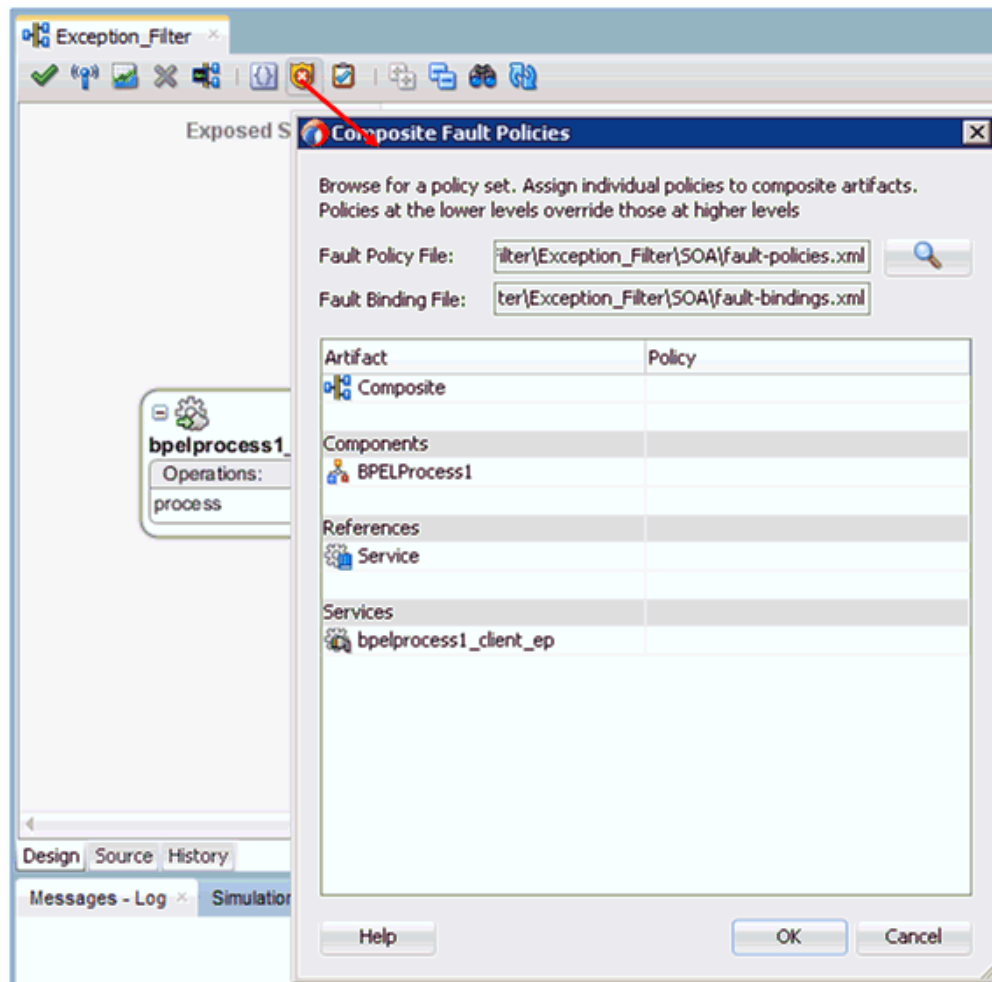
図 9-84 「ソース」タブ



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。

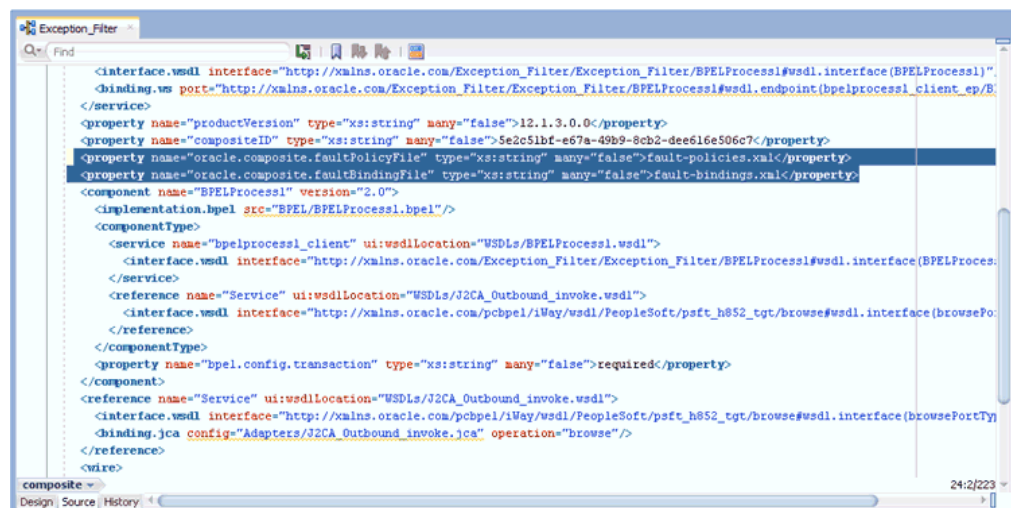
「コンポジット・フォルト・ポリシー」ウィンドウが表示されます。図 9-85 に示すように、フォルト・ポリシーと fault-bindings が正しく選択されていることを確認します。

図 9-85 「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 9-86 に示すように、「ソース」タブをクリックし、*fault-bindings.xml* ファイルと *fault-policies.xml* ファイルが正しく追加されたことを確認します。

図 9-86 「ソース」タブ



9.4.1.4 12c のデプロイメントに関する既知の問題の調整

12c のデプロイメントに関する既知の問題の調整方法の詳細は、4-26 ページ [4.4.3.3 項「12c のデプロイメントに関する既知の問題の調整」](#) を参照してください。

9.4.1.5 例外フィルタ機能を使用する BPEL プロセスのデプロイおよびテスト

例外フィルタ機能を使用する BPEL プロセスをデプロイおよびテストする方法

1. 作成した BPEL プロセスをデプロイします。
詳細は、4-28 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. システム (サーバーが稼働している) をネットワークから切断し、通信エラーをシミュレートします。
3. 有効な入力で、デプロイした BPEL プロセスを呼び出します。
詳細は、4-31 ページ [「Oracle Enterprise Manager Console 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、BPEL プロセスが再試行または中断されるのを観察できます。

9.5 Oracle SOA Suite (BPEL、メディエータまたは BPM) 用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) を使用する構成内で Oracle Application Adapter for Siebel 用の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・シナリオも含まれています。この項では、次のトピックについて説明します。

■ 9.5.1 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・プロセスに対してのみサポートされます。この機能は BSE 構成や J2CA 構成を使用するインバウンド・プロセスではサポートされません。

注意： J2CA コネクタは4つのすべてのアダプタ (SAP R/3、PeopleSoft、Siebel、および J.D. Edwards OneWorld) で共通です。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1つのアプリケーション・アダプタのみが使用されることを確認します。たとえば、1つのアダプタ・インスタンスで使用できるのは Siebel アプリケーション・アダプタのみです。資格証明マッピングを個別のアダプタ・レベルで構成することはできません。2つのアダプタで資格証明マッピングが必要な場合は、両方のアダプタが2つの個別のアダプタ・インスタンスで稼働している必要があります。

ユーザーの資格証明を J2CA リソース・アダプタに渡すには、Oracle WebLogic Server のユーザー資格証明から EIS ユーザー資格証明 (Siebel 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 BPEL、メディアータ、または BPM Web サービスに関連付けて、Oracle WebLogic Server の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ピングされてから、J2CA コンテナに渡されます。J2CA コンテナではこれらの資格情報を使用して EIS アダプタ (Siebel) に接続します。

9.5.1 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。次のステップとトピックから構成されています。

1. アダプタをデプロイします。
詳細は、第3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 資格証明を EIS 資格証明に関連付けます。
詳細は、9-56 ページ 9.5.1.1 項「Oracle WebLogic Server 資格証明の EIS 資格証明との関連付け」を参照してください。
3. WSDL ファイルを生成します。
詳細は、9-59 ページ 9.5.1.2 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・プロセスを作成し、デプロイします。
詳細は、9-59 ページ 9.5.1.3 項「アウトバウンド・プロセスの作成とデプロイ」を参照してください。
5. EIS 資格証明を呼び出し、EIS 資格証明が渡されたことを確認します。
詳細は、9-60 ページ 9.5.1.4 項「EIS 資格証明の確認」を参照してください。

9.5.1.1 Oracle WebLogic Server 資格証明の EIS 資格証明との関連付け

Oracle WebLogic Server 資格証明を EIS 資格証明に関連付ける方法


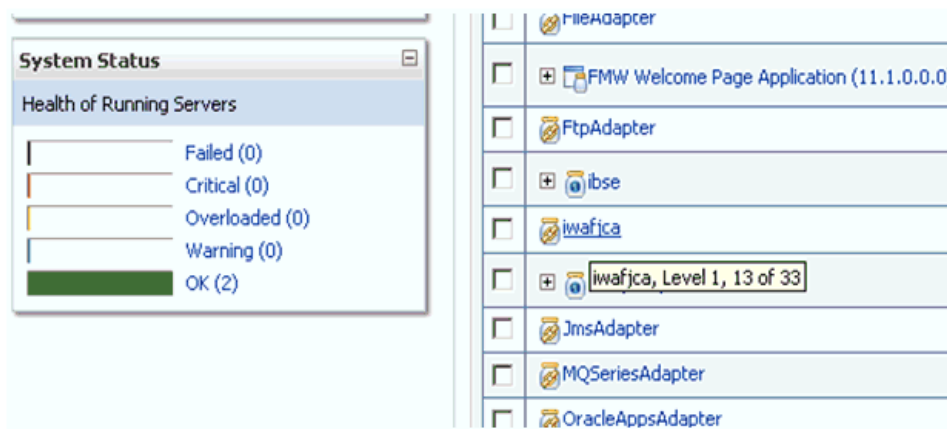
1. Oracle WebLogic Server 管理コンソールにログインします。
2.  9-87 に示すように、左ペインのドメイン構造セクションで、「デプロイメント」をクリックします。

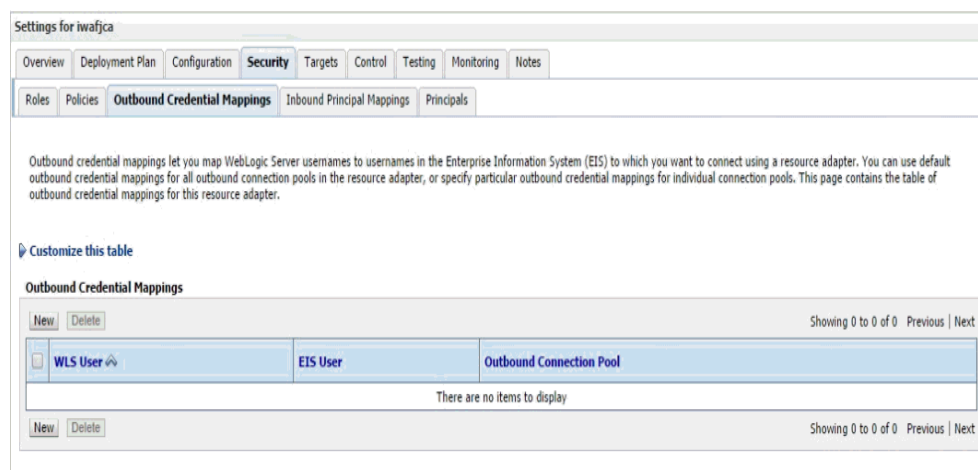
図 9-87 「デプロイメント」 ページ



3. 「iwaifjca」リソース・アダプタをクリックします。

図 9-88 に示すように、「iwaifjca の設定」ページの設定が表示されます。

図 9-88 「iwaifjca 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 9-89 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 9-89 新規セキュリティ資格証明マッピングの作成ページ

Create a New Security Credential Mapping

Back Next Finish Cancel

Outbound Connection Pool

Which Outbound Connection Pool would you like the credential map to be associated with this resource adapter. Each Outbound Connection Pool can then configure themselves

Customize this table

Create a New Security Credential Map Entry for:

<input type="checkbox"/>	Outbound Connection Pool
<input checked="" type="checkbox"/>	eis/OracleJCAAdapter/DefaultConnection
<input type="checkbox"/>	Resource Adapter Default

Back Next Finish Cancel

- アウトバウンド・コネクタ・プールを選択します。
次に例を示します。
eis/OracleJCAAdapter/DefaultConnection
- 「次へ」をクリックします。
図 9-90 に示すように、「WebLogic Server ユーザー」ページが表示されます。

図 9-90 「WebLogic Server ユーザー」ページ

Create a New Security Credential Mapping

Back Next Finish Cancel

WebLogic Server User

Select the WebLogic Server User that you would like to map an EIS user to. Selecting 'User for creating initial connections' will configure the user that will be used for creating the initial connections when the resource adapter is first started. Selecting 'Default User' will configure the user that will be used as the default for any authenticated WebLogic Server user that does not have a credential mapping specifically for them. Selecting 'User for unauthenticated user' will configure the user that will be used for an unauthenticated WebLogic Server user. If you select 'Configured User' you must type in the WebLogic Server user that you are configuring. This user must be a configured WebLogic Server user.

User for creating initial connections
 Default User
 Unauthenticated WLS User
 Configured User Name

WebLogic Server User Name:

Back Next Finish Cancel

- 「デフォルト・ユーザー」を選択し、有効な Oracle WebLogic Server ユーザー名を入力して、「次へ」をクリックします。
図 9-91 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 9-91 「EIS ユーザー名およびパスワード」 ページ

8. EIS のユーザー名と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が EIS ユーザー (Siebel) にマッピングされました。マッピングは J2CA サービスを呼び出す前に自動的に呼び出されます。

9.5.1.2 WSDL ファイルの生成

WSDL ファイルを生成する手順

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、2-4 ページ [2.3.2 項「J2CA 用の構成の作成」](#) を参照してください。
2. Siebel アダプタのターゲットを作成し、ターゲットに接続します。
詳細は、2-5 ページ [2.4 項「Siebel 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用の WSDL を生成します。
詳細は、2-27 ページ [2.12 項「WSDL の生成 \(J2CA 構成のみ\)」](#) を参照してください。

9.5.1.3 アウトバウンド・プロセスの作成とデプロイ

この項では、アウトバウンド・プロセスの構成方法について説明します。デモ用に、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、メディエータと BPM のアウトバウンド・プロセスについても同じステップが適用されます。

メディエータ・アウトバウンド・プロセスの作成の詳細は、[第 5 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 6 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・プロセスの作成の詳細は、次の項を参照してください。

- [4.4.2 項「SOA 用の空のコンポジットの作成」](#)

- 4.4.3 項「BPEL アウトバウンド・プロセスの定義」
- 4.4.4 項「BPEL アウトバウンド・プロセスのデプロイ」

9.5.1.4 EIS 資格証明の確認

この項で説明するように、入力 XML を呼び出して、デフォルト・ユーザーについては EIS ターゲット資格証明が WebLogic 管理コンソールで構成した資格証明で上書きされることを確認します。

1. 有効な入力で、デプロイした BPEL アウトバウンド・プロセスを呼び出します。
詳細は、4-31 ページ 4.4.5 項「Oracle Enterprise Manager Console 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、暗号化されたパスワードを見つけます。これは、ユーザー資格証明が Oracle WebLogic Server 経由で EIS に渡されたことを示しています。

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)
```

9.6 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) を使用する構成内で Oracle Application Adapter for Siebel 用の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・シナリオも含まれています。この項では、次のトピックについて説明します。

- 9.6.1 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・プロセスに対してのみサポートされます。この機能は BSE 構成や J2CA 構成を使用するインバウンド・プロセスではサポートされません。

注意： J2CA コネクタは 4 つのすべてのアダプタ (SAP R/3、PeopleSoft、Siebel、および J.D. Edwards OneWorld) で共通です。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 1 つのアプリケーション・アダプタのみが使用されることを確認します。たとえば、1 つのアダプタ・インスタンスで使用できるのは Siebel アプリケーション・アダプタのみです。資格証明マッピングを個別のアダプタ・レベルで構成することはできません。2 つのアダプタで資格証明マッピングが必要な場合は、両方のアダプタが 2 つの個別のアダプタ・インスタンスで稼働している必要があります。

ユーザーの資格証明を J2CA リソース・アダプタに渡すには、Oracle WebLogic Server のユーザー資格証明から EIS ユーザー資格証明 (Siebel 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 の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ピングされてから、J2CA コンテナに渡されます。J2CA コンテナではこれらの資格情報を使用して EIS アダプタ (Siebel) に接続します。

9.6.1 資格証明マッピングの構成

資格証明マッピングの構成は次のステップとトピックで構成されています。

1. アダプタをデプロイします。
詳細は、第3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. 9.6.1.1 項「Oracle WebLogic Server 資格証明の EIS 資格証明との関連付け」
3. 9.6.1.2 項「WSDL ファイルの生成」
4. 9.6.1.3 項「Oracle Service Bus (OSB) アウトバウンド・プロセスの作成」

9.6.1.1 Oracle WebLogic Server 資格証明の EIS 資格証明との関連付け

Oracle WebLogic Server 資格証明を EIS 資格証明に関連付ける方法

1. Oracle WebLogic Server 管理コンソールにログインします。
2. 図 9-92 に示すように、左ペインのドメイン構造セクションで、「デプロイメント」をクリックします。

図 9-92 ドメイン構造セクション

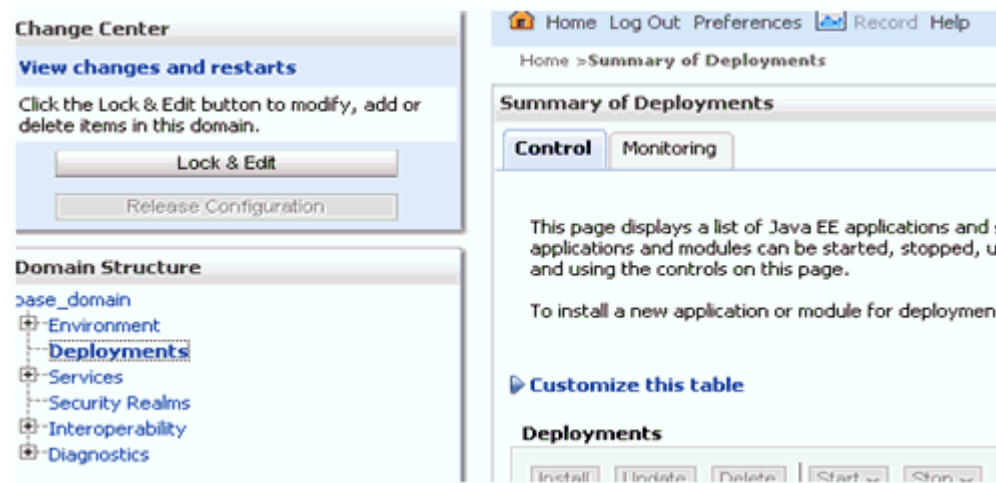
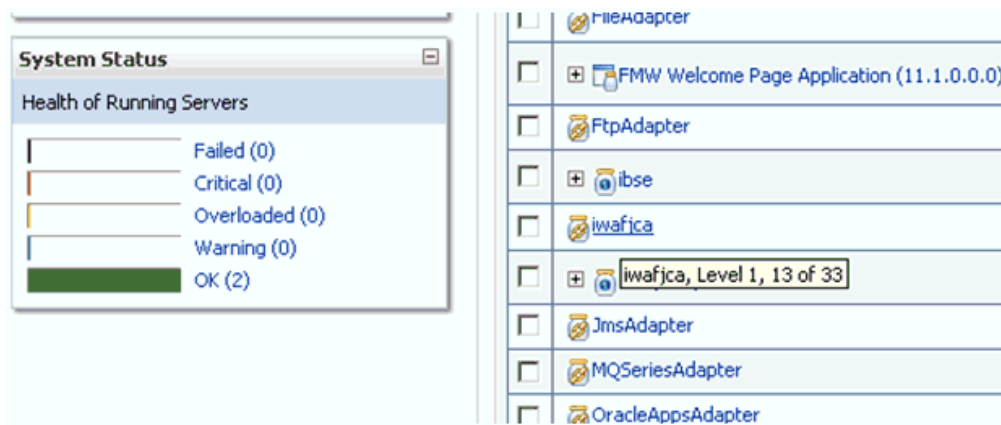


図 9-93 に示すように、「デプロイメント」ページが表示されます。

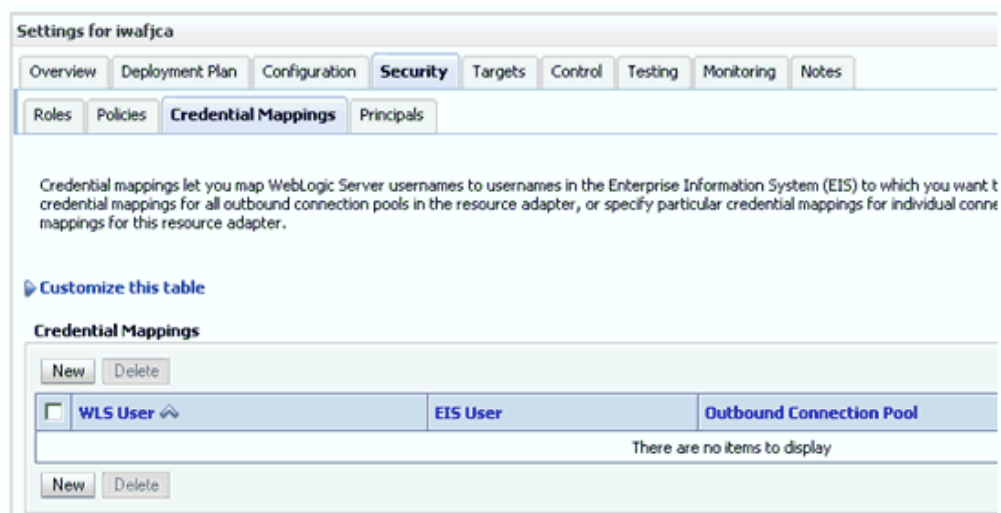
図 9-93 「デプロイメント」 ページ



3. 「iwafjca」リソース・アダプタをクリックします。

図 9-94 に示すように、「iwafjca の設定」ページの設定が表示されます。

図 9-94 「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 9-95 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 9-95 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 wizard window titled "Create a New Security Credential Mapping". At the top, there are navigation buttons: "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 text is a table titled "Create a New Security Credential Map Entry for:" with the following entries:

<input type="checkbox"/>	Outbound Connection Pool
<input checked="" type="checkbox"/>	eis/OracleJCAAdapter/DefaultConnection
<input type="checkbox"/>	Resource Adapter Default

At the bottom of the wizard, there are more navigation buttons: "Back", "Next", "Finish", and "Cancel".

5. アウトバウンド・コネクタ・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 9-96 に示すように、「WebLogic Server ユーザー」ページが表示されます。

図 9-96 「WebLogic Server ユーザー」ページ

The screenshot shows the "WebLogic Server User" section of the wizard. It contains a text box with instructions: "Select the WebLogic Server User that you would like to map an EIS user to. Selecting 'User for creating initial connections' will the resource adapter is first started. Selecting 'Default User' will configure the user that will be used as the default for any a specifically for them. Selecting 'User for unauthenticated user' will configure the user that will be used for an unauthenticated WebLogic Server user that you are configuring. This user must be a configured WebLogic Server user." Below the text are four radio button options:

- User for creating initial connections
- Default User
- Unauthenticated WLS User
- Configured User Name

Below these options is a text field labeled "WebLogic Server User Name:" with the value "weblogic" entered. At the bottom, there are navigation buttons: "Back", "Next", "Finish", and "Cancel".

7. 「構成済みユーザー名」を選択し、有効な Oracle WebLogic Server ユーザー名を入力して、「次へ」をクリックします。
図 9-97 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 9-97 「EIS ユーザー名およびパスワード」 ページ

8. EIS のユーザー名と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が EIS ユーザー (Siebel) にマッピングされました。マッピングは J2CA サービスを呼び出す前に自動的に呼び出されます。

9.6.1.2 WSDL ファイルの生成

WSDL ファイルを生成する手順

1. Oracle Service Bus (OSB) と統合するためのアプリケーション・エクスプローラのクラス・パスを設定します。
詳細は、7-6 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、2-4 ページ「[J2CA 用の構成の作成](#)」を参照してください。
3. Siebel アダプタのターゲットを作成し、ターゲットに接続します。
詳細は、2-5 ページ「[Siebel 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用の WSDL を生成します。
詳細は、4-8 ページ [4.4.1 項「リクエスト / レスポンス・サービス用の WSDL の生成」](#) を参照してください。

9.6.1.3 Oracle Service Bus (OSB) アウトバウンド・プロセスの作成

Oracle Service Bus (OSB) アウトバウンド・プロセスの作成の詳細は、8-3 ページ [8.1.2 項「OSB アウトバウンド処理の定義」](#) を参照してください。

1. [図 9-98](#) に示すように、OSB プロジェクトを右クリックし、「新規」を選択したら、「サービス・アカウント」をクリックしてサービス・アカウントを構成します。

図 9-98 サービス・アカウント・オプションの選択

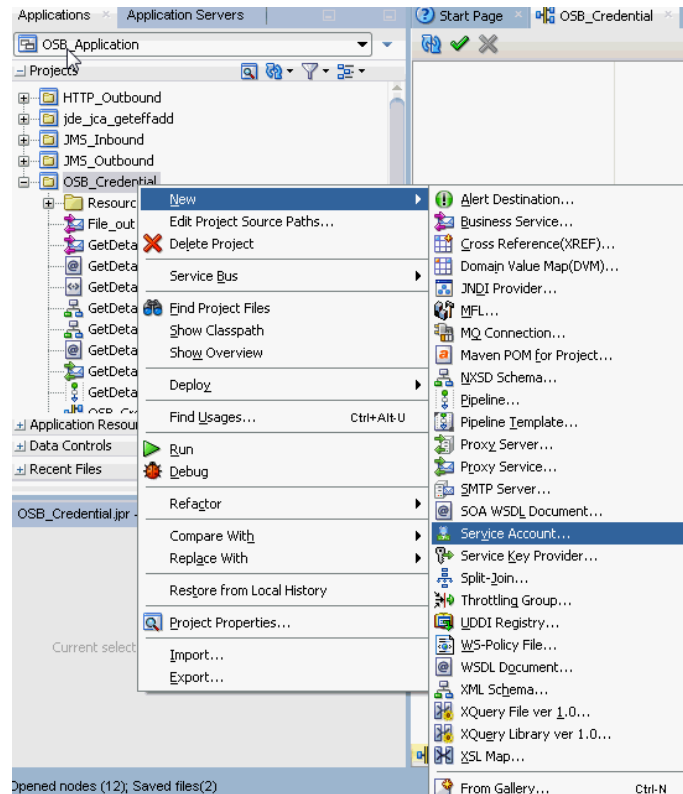
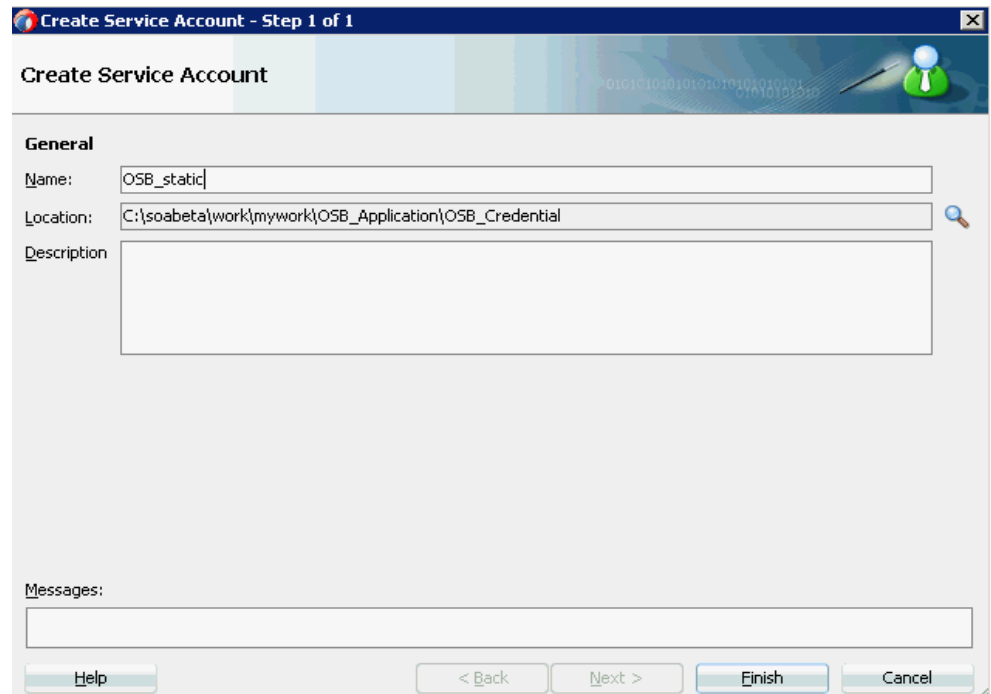


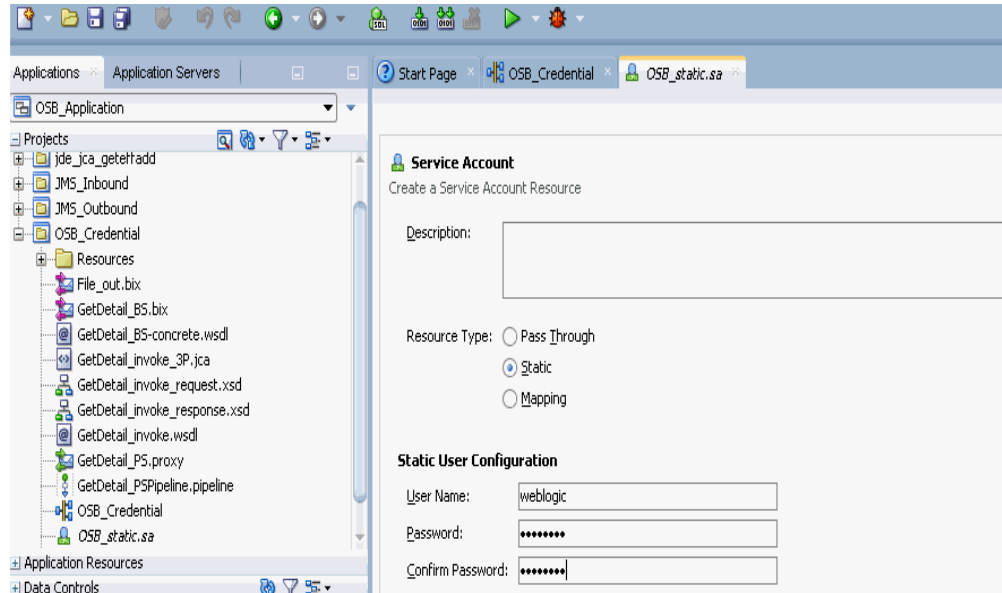
図 9-99 に示すように、サービス・アカウントの作成ペインが表示されます。

図 9-99 サービス・アカウントの作成ペイン



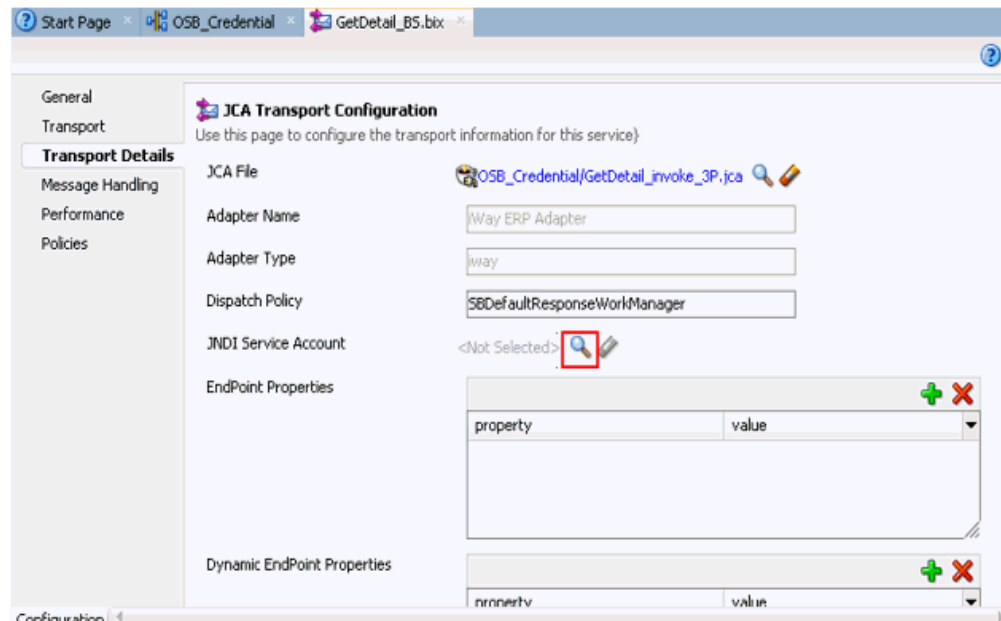
2. サービス・アカウントの名前を入力し、「終了」をクリックします。
サービス・アカウントの構成ページが表示されます。
3. 「リソース・タイプ」セクションで「静的」を選択します。
4. 図 9-100 に示すように、Oracle WebLogic Server の有効なユーザー名とパスワードを入力します。

図 9-100 「サービス・アカウント構成」構成ページ



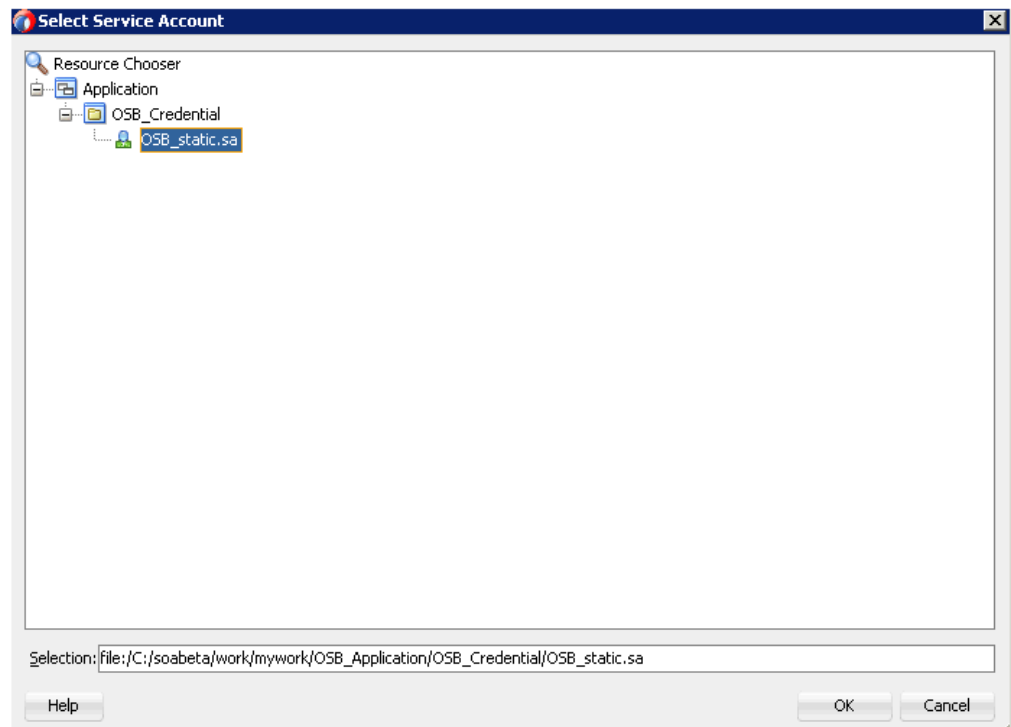
5. 構成ページを保存して閉じます。
6. コンポジット・エディタ・ウィンドウで、ステップ 3 で作成した WSDL ベースのビジネス・サービスをダブルクリックします。
WSDL ベースのビジネス・サービスの構成ページが表示されます。
7. 図 9-101 に示すように、「トランスポートの詳細」タブを選択します。

図 9-101 「トランスポートの詳細」タブ



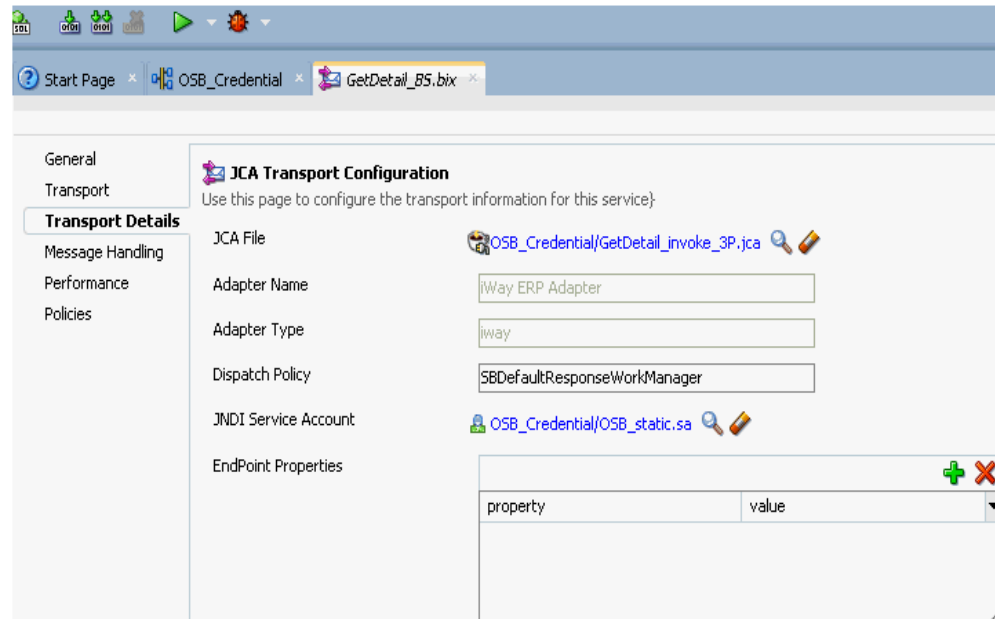
8. 「JNDI サービス・アカウント」アイコンをクリックします。
「サービス・アカウントの選択」ウィンドウが表示されます。
9. 図 9-102 に示すように、作成したサービス・アカウントを選択して、「OK」をクリックします。

図 9-102 サービス・アカウントの選択



10. 図 9-103 に示すように、構成ページを保存して閉じます。

図 9-103 ビジネス・サービスの構成ページ



11. BPEL プロセスをデプロイします。

詳細は、8-17 ページ 8.1.3 項「OSB アウトバウンド処理のデプロイ」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、有効な入力 XML ファイルをコピーし、構成した入力フォルダに貼り付け、構成済の出力ロケーションで出力が受信されるかを確認します。

13. J2CA ログ・ファイルを確認し、暗号化されたパスワードを見つけます。これは、ユーザー資格証明が Oracle WebLogic Server 経由で EIS に渡されたことを示しています。

次に例を示します。

```
FINEST IWAManagedConnectionFactory com.ibi.afjca.Util
getPasswordCredential(78) InLoop:
User-iwayqa:Password-ENCR(3189319731831132182333215323332323192322731773252)
FINEST IWAManagedConnectionFactory com.ibi.afjca.Util
getPasswordCredential(90) Use the system PasswordCredential:
User-iwayqa:Password-ENCR(3109313331831131702333215320132323192322731773236)
```

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、Siebel 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。内容は次のとおりです。

- 10.1 項「トラブルシューティング」
- 10.2 項「BSE エラー・メッセージ」

10.1 トラブルシューティング

この項では、Siebel に関するトラブルシューティング情報を次の 4 つのカテゴリに分けて説明します。

- 10.1.1 項「Oracle Application Adapter for Siebel に関する一般的な使用上の注意」
- 10.1.2 項「アプリケーション・エクスプローラ」
- 10.1.3 項「Siebel」
- 10.1.4 項「Oracle Adapter J2CA」

トラブルシューティングに関連する可能性があるログ・ファイル情報は、アダプタのインストールに基づき次の場所にあります。

- 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`

10.1.1 Oracle Application Adapter for Siebel に関する一般的な使用上の注意

Oracle Application Adapter for Siebel には、次の制約があります。

- HTTPS プロトコルは、サービスおよびイベントに対してサポートされていません。
- 結合が指定された複数値 (MVG) フィールドの更新はサポートされていません。
- 接続が失われた場合、アダプタは Siebel に自動的に再接続しません。

10.1.2 アプリケーション・エクスプローラ

この項では、アプリケーション・エクスプローラを使用すると発生する可能性がある様々なエラー・タイプについて説明します。

エラー	ソリューション
<p>Siebel がアプリケーション・エクスプローラの「アダプタ」ノード・リストに表示されません。</p>	<p>Siebel の配布メディアとともに提供されている Siebel jar ファイルが <ADAPTER_HOME>\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\lib ディレクトリにあることを確認します。たとえば Siebel 7.03 環境の場合は、SiebelJI_Common.jar と SiebelJI_enu.jar がこのディレクトリに置かれている必要があります。</p>
<p>ターゲット・タイプ・ドロップ・ダウンに、「Java データ Bean 接続」のみが含まれていますが、COM 接続タイプが必要です。</p>	<p>アプリケーション・エクスプローラのホストとなるシステム上に Siebel シン・クライアントが正しくインストールされ、適切な COM 環境が使用可能なことを確認します。</p>
<p>Siebel ターゲットに接続しようとする、Siebel Gateway サーバーの名前を含んだエラー・メッセージが表示されます。例：</p>	<p>使用するターゲットに対して Siebel Gateway サーバーの名前が正しく定義されていることを確認します。</p>
<p>アダプタのアクティブ化で問題が発生しました (<server_name>)。詳細はログを確認してください。</p>	<p>Siebel システムへ接続するためのユーザー ID およびパスワード・パラメータの値が正しいことを確認します。</p>
<p>Siebel ターゲットに接続しようすると、次のエラーが表示されます。 アダプタのアクティブ化で問題が発生しました。(無効なログオン・パラメータ・セットを入力しました。ログオン・パラメータを再入力してください。) 詳細はログを確認してください。</p>	<p>Siebel システムへ接続するためのユーザー ID およびパスワード・パラメータの値が正しいことを確認します。</p>
<p>Siebel ターゲットに接続しようすると、次のエラーが表示されます。 アダプタのアクティブ化で問題が発生しました。(ネームサーバー接続を取得できませんでした) 詳細はログを確認してください。</p>	<p>Siebel 環境へのネットワーク接続状況を確認します。ネットワークの問題を修正し、接続を再試行します。</p>
<p>Siebel ターゲットに接続しようすると、次のエラーが表示されます。 アダプタのアクティブ化で問題が発生しました。(NSReadKey リクエストが失敗しました (エラー情報なし)...) 詳細はログを確認してください。</p>	<p>Siebel サーバー、企業名および使用中のターゲットのオブジェクト・マネージャに対して定義された値が正しいことを確認し、接続を再試行します。</p>
<p>Siebel ターゲットに接続しようすると、次のエラーが表示されます。 アダプタのアクティブ化で問題が発生しました。(翻訳可能なメッセージのロード時のエラー : com.siebel.locale.enu.messages.SSAMessages_enu) 詳細はログを確認してください。</p>	<p>Siebel システムへの接続に使用しているターゲットに対して、「詳細」パラメータの値が正しく定義されていることを確認します(たとえば英語の場合は enu)。</p>

エラー	ソリューション
<p>Siebel 環境への接続には成功しましたが、アプリケーション・エクスプローラ・ツリーの「ビジネス・オブジェクト」の各ノードで使用可能な値がありません。</p>	<p>Siebel のターゲット構成の「詳細」タブで指定したリポジトリ名にターゲットの Siebel 環境のコンポーネントがないか、そのリポジトリ名がターゲットの Siebel 環境に対して有効ではありません。リポジトリ名が有効でコンポーネントを含んでいることを確認し、再接続します。</p>
<p>実行時にログオン失敗エラーが発生します。</p>	<p>Siebel 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、Siebel には接続できません。接続パスワードは repository.xml には保存されません。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、アプリケーション・サーバーを再起動します。</p>
<p>ae.bat(iaexplorer.exe ではない)をアクティブ化してアプリケーション・エクスプローラを起動すると次の例外が発生します。</p>	<p>これは無害な例外です。アダプタの機能には影響しません。次の URL から BouncyCastle ファイルをダウンロードします。</p>
<pre>java.lang.ClassNotFoundException: org.bouncycastle.jce.provider.BouncycastleProvider</pre>	<p>ftp://ftp.bouncycastle.org/pub</p>
<p>Solaris 環境でアプリケーション・エクスプローラを起動できません。次の例外がコンソールに表示されます。</p>	<p>JAVACMD がユーザーの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>
<pre>javax.resource.ResourceException: IWAFFManagedConnectionFactory: License violation.at com.ibi.afjca.spi.IWAFFManagedConnectionFactory.createConnectionFactory(IWAFFManagedConnectionFactory.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)</pre>	<p>JAVACMD=/<jdk_home>/bin/java。ここで、<jdk_home>は、システム上でJDKがインストールされているディレクトリです。</p>
<p>接続ファクトリを作成できませんでした。</p>	

10.1.3 Siebel

リストされたエラー・メッセージは、アダプタを BSE または Oracle Adapter J2CA のリポジトリ・プロジェクトとともに使用しているときに発生する可能性があります。

エラー	ソリューション
<p>Siebel 環境への接続には成功したが、アプリケーション・エクスプローラ・ツリーの「ビジネス・オブジェクト」の各ノードで使用可能な値がありません。</p>	<p>Siebel のターゲット構成の「詳細」タブで指定したリポジトリ名にターゲットの Siebel 環境のコンポーネントがないか、そのリポジトリ名がターゲットの Siebel 環境に対して有効ではありません。リポジトリ名が有効でコンポーネントを含んでいることを確認し、再接続します。</p>
<p>リクエストの実行時、次のエラー・メッセージが表示されます。</p> <p>AdapterException: サポートされていないアクション: {0} Tquery</p>	<p>スキーマを確認して特定のリクエストに対してメソッドが使用可能なことを確認してください。</p>
<p>リクエストの実行時、次のエラー・メッセージが表示されます。</p> <p>AdapterException: フィールド 'NFame' がビジネス・コンポーネント 'Account' の定義内にありません。システム管理者にアプリケーション構成の確認を依頼してください。</p>	<p>その特定のオブジェクトに対するスキーマを参照してフィールド名がリクエスト・ドキュメント内で有効であることを確認した後、リクエストを再発行します。</p>
<p>Java データ Bean インタフェースを使用して Siebel 7.7 より前のリリースに接続する場合、最初の接続が失われた後、再接続できません。これは、アプリケーション・エクスプローラで短時間のネットワーク接続損失があったか、アプリケーション・エクスプローラが Siebel アプリケーションにログインしている間に Siebel サーバーまたはゲートウェイ・サービスが再起動した場合に発生することがあります。</p>	<p>Siebel アプリケーションに正常にログインするために、Oracle WebLogic Server とアプリケーション・エクスプローラを再起動します。これは Siebel API の既知の問題です。詳細は Siebel Alert 984 を参照してください。</p>
<p>階層データ型のメソッド引数を持つメソッドが含まれるビジネス・サービスに対してサービス・ノードを追加すると、次のエラーが発生する場合があります。</p> <p>有効な XMLCharEncoding 値 (UTF-8、UTF-16 など) を入力すると、次のエラーが表示されます。</p> <p>サービスの呼び出しに失敗しました。</p>	<p>メソッド引数 XMLCharEncoding はサポートされていません。XML ペイロードでこの要素を空白にします。</p>

10.1.4 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・エクスプローラに次のメッセージが表示されます。</p> <p>JCA を初期化できませんでした</p>	<p>次の例に示すように、右ペインの「詳細」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>

10.2 BSE エラー・メッセージ

この項では、BSE を使用して Web サービスを処理する場合に発生する可能性がある様々なエラー・タイプ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。

- [10.2.1 項「BSE での一般的なエラー処理」](#)
- [10.2.2 項「アダプタ固有のエラー処理」](#)

10.2.1 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 ゲートウェイと一部のアダプタのエンジンとの両方の役割を果たします。アダプタを使用する Web サービスが実行される場合、設計時と実行時の両方で、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るものと、使用されるアダプタによって異なった形態で公開されるものがあります。この項では、アダプタに固有のもので、いくつかの一般的なエラー条件の場合に発生するエラー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) は、SOAP リクエスト・メッセージを Web サービスに必要なアダプタに渡します。エラーが発生した場合、公開方法はアダプタおよび API、またはアダプタで使用されるインタフェースによって異なります。いくつかのシナリオで、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 エージェントが無効な SOAP リクエストを受信するたびに、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。

10.2.2 アダプタ固有のエラー処理

この項では、次のトピックについて説明します。

- [10.2.2.1 項「Oracle Application Adapter for Siebel の無効な SOAP リクエスト」](#)
- [10.2.2.2 項「Siebel リクエストからの空の結果」](#)
- [10.2.2.3 項「Oracle WebLogic Server Integration Adapter」](#)
- [10.2.2.4 項「無効な SOAP リクエスト」](#)
- [10.2.2.5 項「Oracle WebLogic Server Adapter リクエストからの空の結果」](#)

アダプタで実行時に例外が発生すると、BSE 内の SOAP エージェントは生成された SOAP レスポンス内に SOAP フォルト要素を含めます。SOAP フォルト要素にはフォ

ルト・コード要素およびフォルト文字列要素が含まれます。フォルト文字列には、アダプタのターゲット・システムからのネイティブ・エラー記述が含まれます。アダプタはターゲット・システムのインタフェースおよび API を使用するので、例外が発生するかどうかはターゲット・システムのインタフェースまたは API によるエラー条件の処理方法によって決定されます。BSE 内の SOAP エージェントによって SOAP リクエスト・メッセージがアダプタに渡される時、リクエストがそのサービスの WSDL に基づいて無効な場合、アダプタは SOAP フォルトを起こす例外が発生することがあります。

アダプタで起こる可能性があるすべてのエラー条件を予測することはほぼ不可能ですが、一般的なエラー条件をアダプタがどのように処理し、どのように Web サービスのコンシューマ・アプリケーションに公開されるかについて次に説明します。

10.2.2.1 Oracle Application Adapter for Siebel の無効な SOAP リクエスト

Oracle Application Adapter for Siebel で、実行される Web サービスの WSDL に準拠しない SOAP リクエスト・メッセージが受信されると、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>XD[FAIL] Parse failure (IS) 3: org.xml.sax.SAXParseException:
Premature end of file.</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

10.2.2.2 Siebel リクエストからの空の結果

Web サービスの実行時に Oracle Application Adapter for Siebel が Siebel 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><Exception> - major:4096 minor: -1 message:NSReadKey request 11 was
abandoned
after 37846ms connection:12a due to Connection shutdown request
Connection reset by peer:JVM_recv in socket input stream
stream read DetailedMessage:Unknown</Exception></faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

10.2.2.3 Oracle WebLogic Server Integration Adapter

Oracle Adapter は、エンジンが他の Oracle サーバーであるアダプタに BSE を接続します。したがって、このタイプのアダプタは多くの異なるターゲット・システムへの BSE の接続に使用されるため、エラー処理の動作には一貫性があります。アダプタのユーザー・ガイドで、Web サービスの実行時に Oracle WebLogic Server Integration Adapter が必要かどうかを確認してください。

10.2.2.4 無効な SOAP リクエスト

Oracle Application Adapter for Siebel で、実行される Web サービスの WSDL に準拠しない SOAP リクエスト・メッセージが受信されると、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>RPC server connection failed: Connection refused:
connect</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

10.2.2.5 Oracle WebLogic Server Adapter リクエストからの空の結果

渡された入力パラメータを使用して Oracle Application Adapter for Siebel で SOAP リクエストを実行する場合に、入力パラメータがターゲット・システム内のレコードと一致しないと、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>
```

Siebel ワークフローの使用

Siebel XML を使用して Siebel 統合オブジェクトと統合する場合、インタフェースでは Siebel ワークフローが使用されます。

注意： この付録は Oracle Application Adapter for Siebel ユーザー向けのドキュメントを補足するためのもので、Siebel のドキュメントにかわるものではありません。Siebel ワークフローおよびポリシーに関する完全で最新の情報は、Siebel システムの Siebel Bookshelf を参照してください。

この付録の内容は次のとおりです。

- [A.1 項「概要」](#)
- [A.2 項「Siebel ワークフローの作成」](#)

A.1 概要

Siebel ワークフローは、Siebel XML を発行または受信するために Siebel 内で定義されます。どちらの場合も、発行や受信は MQSeries、ファイルまたは HTTP 用の Siebel トランSPORT・サービスにより処理されます。この項に含まれる次のトピックでは、サポートされているトランSPORT・サービスを用いたワークフローの使用と作成について説明します。

- [A.1.1 項「Siebel ワークフロー」](#)
- [A.1.2 項「ポリシーを使用した Siebel EAI ワークフローの起動」](#)
- [A.1.3 項「Siebel ワークフロー - アウトバウンド」](#)
- [A.1.4 項「Siebel ワークフロー - インバウンド」](#)

A.1.1 Siebel ワークフロー

Siebel ワークフローは、一連の Siebel ビジネス・サービスが相互にリンクされて 1 つのビジネス・タスクを構成したものです。ワークフローは、Siebel クライアントのワークフロー管理用画面を使用して作成します。ワークフローは次のいずれかの方法で起動されます。

- ワークフロー・ポリシーの使用
- ランタイム・イベント (Siebel イベント) の使用
- スクリプト (eScript または Siebel VB) の使用

次の項では、ポリシー条件を使用してワークフローを起動する方法について簡単に説明します。

関連項目：

ポリシーおよびその他の方法の詳細は、Siebel Bookshelf のドキュメントを参照してください。

A.1.2 ポリシーを使用した Siebel EAI ワークフローの起動

ワークフロー・ポリシーは、一連の定義済アクションを実行する一連の条件により定義されます。Siebel ワークフロー・ポリシーは次のもので構成されます。

- Siebel データベースの状態の変化に基づいて状況を定義した条件。
- 条件が満たされたときに実行される手順を定義したアクション。

ワークフローをアクションとして起動するポリシーを作成するには、次の手順を実行します。

1. ポリシーのトリガー後に実行されるアクションを定義します。統合プロセスの実行プログラムを使用します。
2. 条件を設定して適切なポリシー・グループとアクションを選択することにより、ポリシーを作成します。
3. 開始日を選択してポリシーをアクティブ化します。
4. 「サーバー管理」ウィンドウから「トリガーの生成」サーバー・タスクを実行し、モニターする条件を設定します。
5. 適切なポリシー・グループ (使用するポリシーが属しているグループ) で編集した後、ワークフロー・モニター・エージェントを起動し、アクションを実行するかどうかを評価します。
6. 「サーバー管理」ウィンドウからワークフロー・アクション・エージェント・サーバー・タスクを開始してアクションを実行します。

A.1.3 Siebel ワークフロー - アウトバウンド

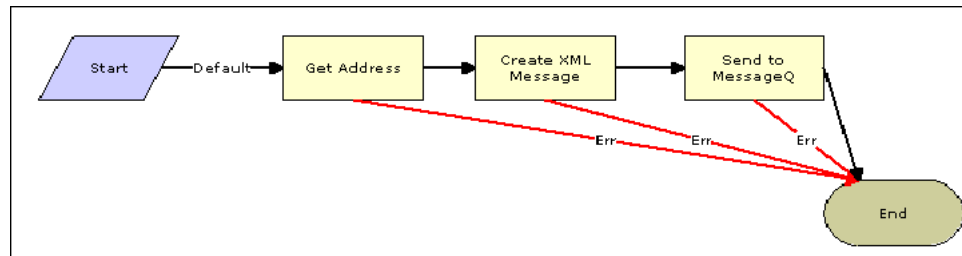
Siebel ポリシー、ランタイムまたはスクリプト (eScript または Siebel VB) イベントに基づいて Siebel ワークフローがトリガーされると、その結果として、Siebel XML ドキュメントが生成され Siebel トランスポートの 1 つに置かれます。たとえば、Siebel Call Center アプリケーションに新規アカウントを追加すると、そのアカウント・トランザクション上でトリガーされるワークフローを設計および構成できます。このワークフローを、新規レコードのデータを取得して Siebel XML に変換し、MQSeries メッセージ・キューに入れるように設計できます。

この例では、Siebel ワークフローのプロセスで次の一連の Siebel ビジネス・サービスが実行されます。

1. 新規更新されたアカウント・レコードを問い合わせ、そのレコードの元の内部構造にあったデータをメモリーに入れる Siebel EAI Siebel Adapter をコールします。
2. データを XML メッセージに変換する Siebel EAI XML Converter をコールします。
3. 新規作成された XML メッセージを適切な MQSeries メッセージ・キューに入れる Siebel EAI MQSeries Transport をコールします

メッセージ・キューに入った後、メッセージは Oracle Application Adapter for Siebel 6.3 以上により取得されます。次のワークフロー・シーケンス (図 A-1) は、前述の手順を示しています。

図 A-1 Siebel の更新データをエクスポートする Siebel ワークフローの作成方法



A.1.4 Siebel ワークフロー - インバウンド

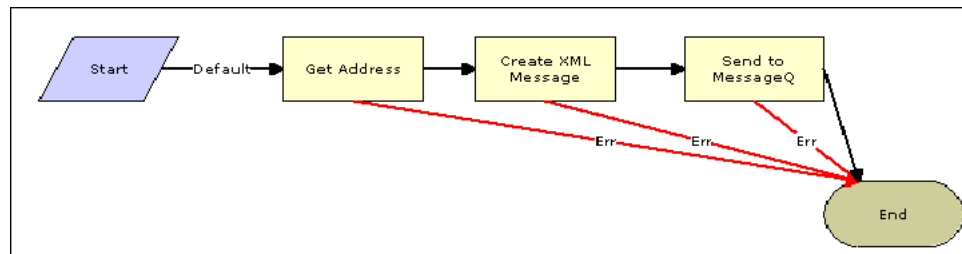
外部イベントによりトリガーされる Siebel ワークフローは、そのトランスポートの 1 つに置かれた Siebel XML ドキュメントを受信することによって開始されます。その結果、XML を入力に使用した Siebel レコードが更新される場合があります。たとえば、別の CRM システムに新規アカウントが追加され、Siebel Call Center アプリケーションでも更新が必要な場合があります。MQSeries メッセージ・キュー上で受信またはリスニングするようにワークフローを設計および構成できます。XML メッセージを受信すると、ワークフローはトランザクションを処理して Siebel システムに移し、レコードを更新します。

この例では、Siebel XML メッセージをメッセージ・キュー内で受信したときに Siebel MQSeries Receiver サーバー・タスクが Siebel ワークフロー・プロセスを開始し、このプロセスが次に示すように一連の Siebel ビジネス・サービスを実行します。

1. Siebel EAI XML Converter をコールし、それにより XML メッセージを Siebel の内部形式に変換します。
2. Siebel EAI Siebel Adapter をコールし、それにより、そのサービス内に定義されたメソッドに基づいて新規更新されたアカウント・レコードを適用します。

次にワークフロー・プロセスの例を示します (図 A-2)。

図 A-2 サンプル・ワークフロー・プロセス



A.2 Siebel ワークフローの作成

この項に含まれる次のトピックでは、Siebel ワークフロー管理ウィンドウで Siebel ワークフローを作成する手順を示します。

- A.2.1 項「MQSeries トランスポートを使用したイベント用の Siebel ワークフローの作成」
- A.2.2 項「ファイル・トランスポートを使用したイベント用の Siebel ワークフローの作成」
- A.2.3 項「HTTP トランスポートを使用したイベント用の Siebel ワークフローの作成」

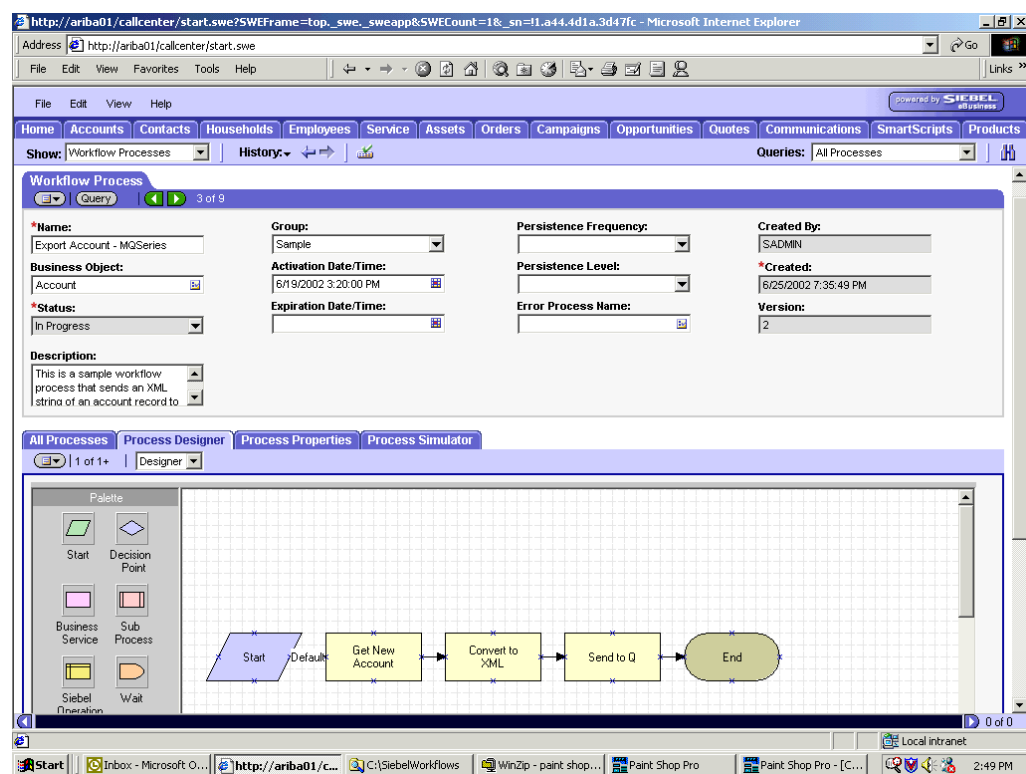
- A.2.4 項「MQSeries トランスポートを使用したサービス用の Siebel ワークフローの作成」
- A.2.5 項「ファイル・トランスポートを使用したサービス用の Siebel ワークフローの作成」
- A.2.6 項「HTTP トランスポートを使用したサービス用の Siebel ワークフローの作成」

A.2.1 MQSeries トランスポートを使用したイベント用の Siebel ワークフローの作成

次の手順は、Siebel ワークフロー管理ウィンドウに示された Siebel ワークフローの例です。このワークフローは、MQSeries トランスポートを使用して Siebel Account レコード情報をエクスポートするために設計されたものです。

次の図 (図 A-3) は、Siebel ワークフロー管理ウィンドウです。

図 A-3 Siebel ワークフロー管理ウィンドウ



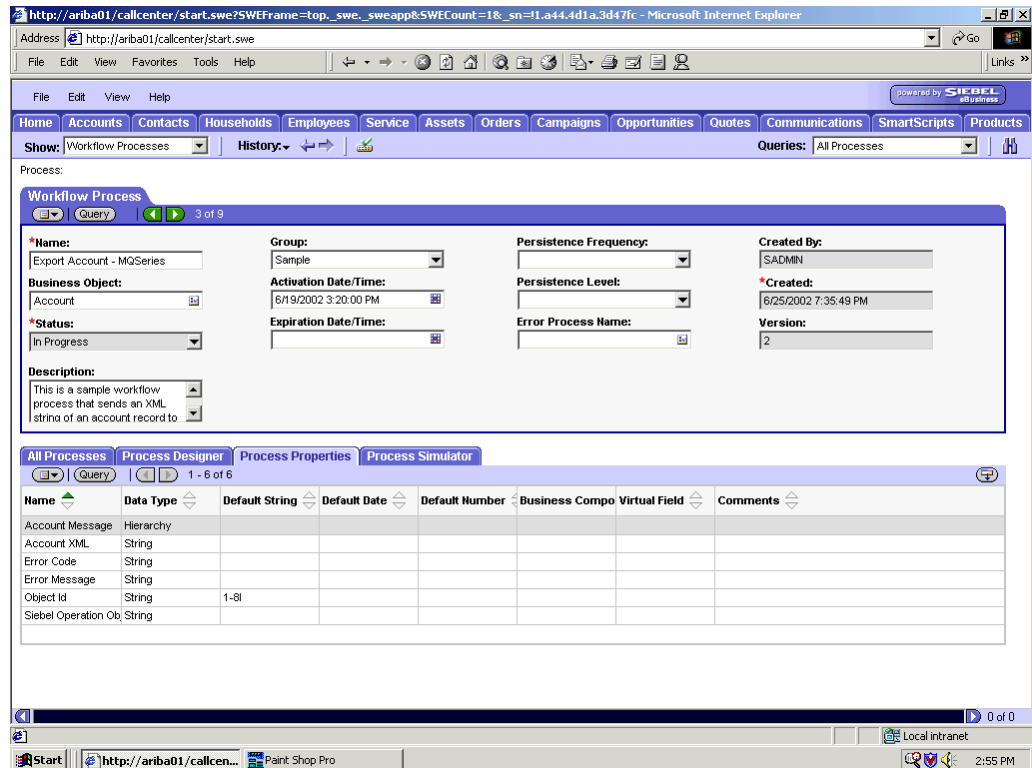
次の手順では、Siebel Call Center アプリケーションで Account レコードが更新されたときに Siebel XML を生成する Siebel ワークフローの作成方法を説明します。ワークフローはその後 MQSeries メッセージ・キューに置かれます。

Siebel ワークフローを作成する手順

1. ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブで、Account メッセージと Account XML プロセス・プロパティを定義します。
Account メッセージには Siebel Account データが階層形式で含まれています。
Account XML は、ワークフローで XML に変換された Siebel Account データを指定します。

図 A-4 に示すように、ウィンドウが表示され、「プロセスのプロパティ」タブがアクティブな状態で表示されます。

図 A-4 ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブ

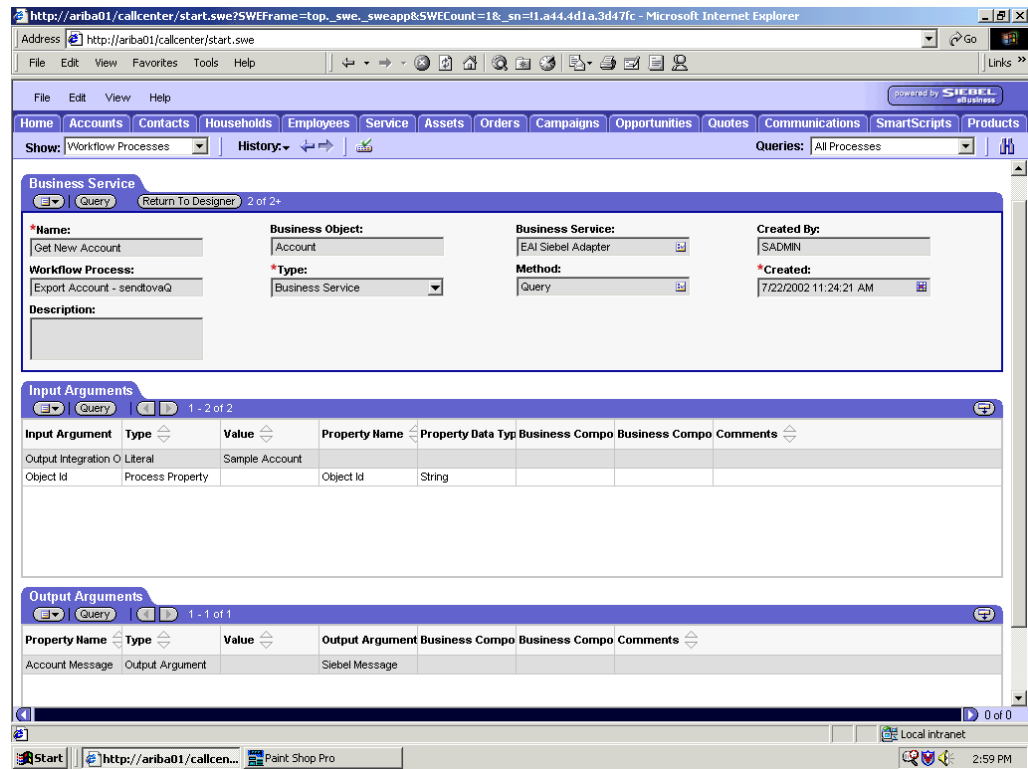


2. Siebel ワークフロー管理ウィンドウを使用して、ワークフローを作成します。
3. Account データのインスタンスを受信するために EAI Siebel Adapter ビジネス・サービス・ステップを定義し、Get New Account という名前を付けます。

このビジネス・サービスは、Query メソッドを使用して Siebel から Account 情報を取得します。

図 A-5 に示すように、このビジネス・サービスからの出力は階層形式で生成されます。

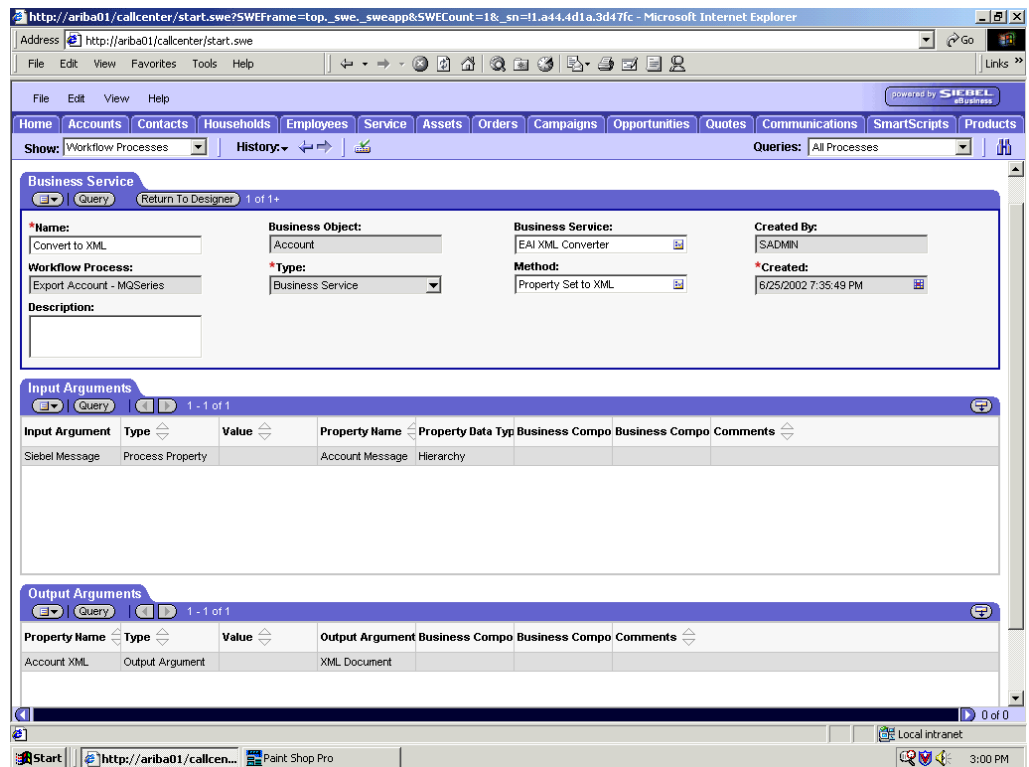
図 A-5 階層形式から生成されるビジネス・サービスからの出力



- EAI XML Converter ビジネス・サービス・ステップを定義し、Convert to XML という名前を付けます。

図 A-6 に示すように、EAI Siebel Adapter ビジネス・サービスから Account データを階層形式で受信し、それを XML 形式に変換するように定義されています。

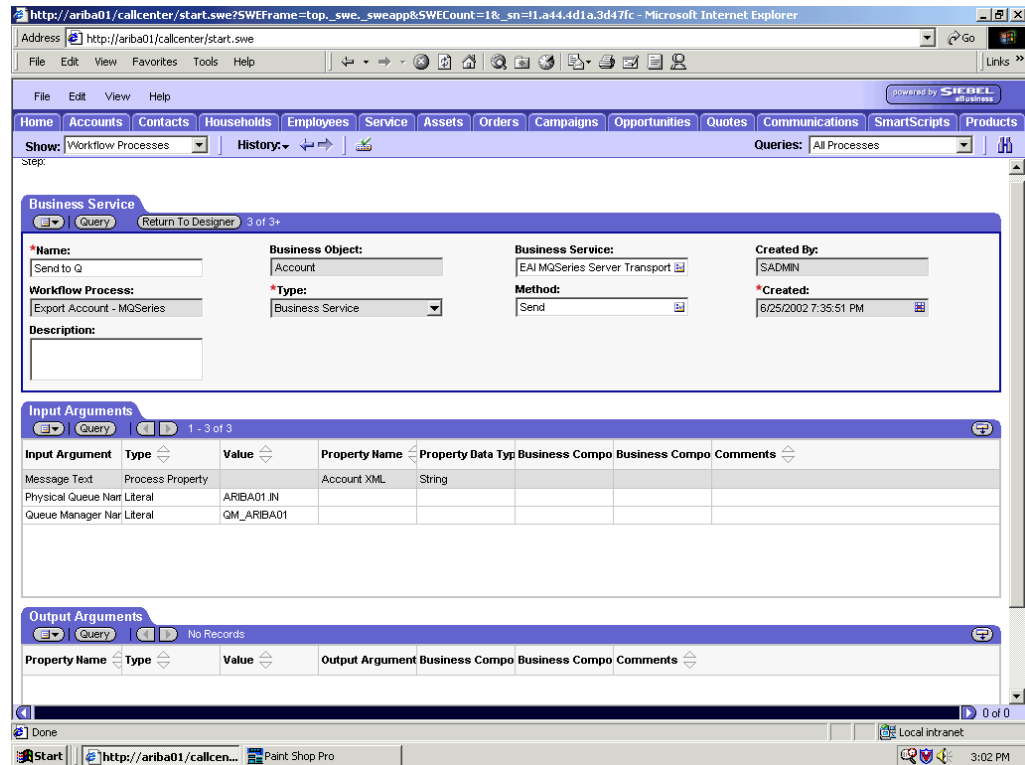
図 A-6 EAI XML Converter ビジネス・サービス



5. 図 A-7 に示すように、EAI MQSeries Server Transport ビジネス・サービス・ステップを定義し、Send to Q という名前を付けます。

EAI XML Converter ビジネス・サービスから Account データを Siebel XML 形式で受信し、Send メソッドを使用して Account XML を MQSeries に送信するように定義されています。

図 A-7 EAI MQ Series Server Transport ビジネス・サーバー「Send to Q」

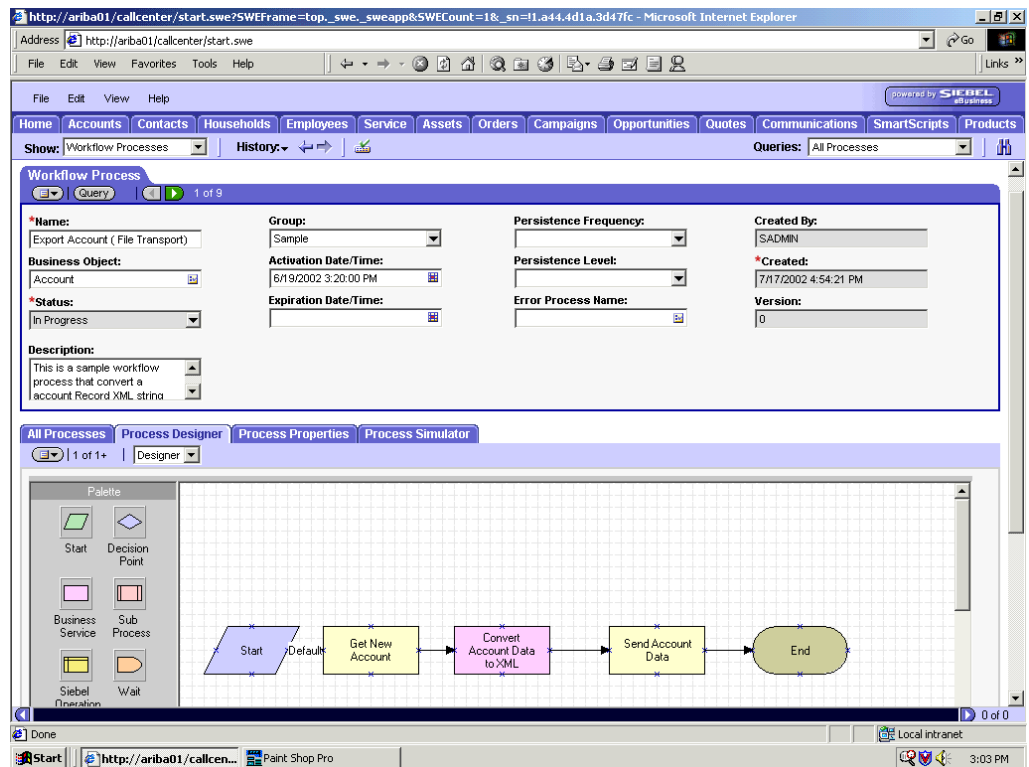


A.2.2 ファイル・トランスポートを使用したイベント用の Siebel ワークフローの作成

次の手順は、Siebel ワークフロー管理ウィンドウに示された Siebel ワークフローの例です。このワークフローは、ファイル・トランスポートを使用して Siebel Account レコード情報をエクスポートするために設計されたものです。

図 A-8 に示すように、ウィンドウが表示され、プロセス・デザイナー・タブがアクティブな状態で表示されます。

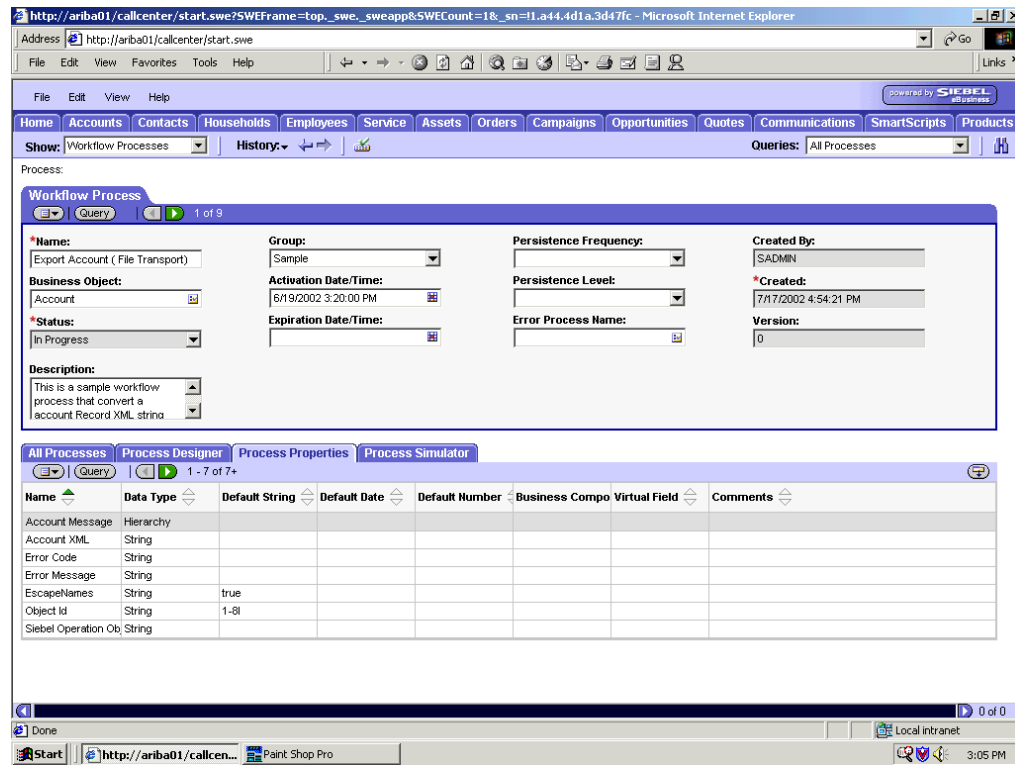
図 A-8 Siebel ワークフロー管理ウィンドウ



この手順では、Siebel Call Center アプリケーションで Account レコードが更新されたときに Siebel XML を生成し、Siebel XML をファイル・システム上に置く Siebel ワークフローの作成方法を説明します。

Siebel ワークフローを作成する手順

図 A-9 ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブ



1. 図 A-9 に示すように、ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブで、Account メッセージと Account XML プロセス・プロパティを定義します。

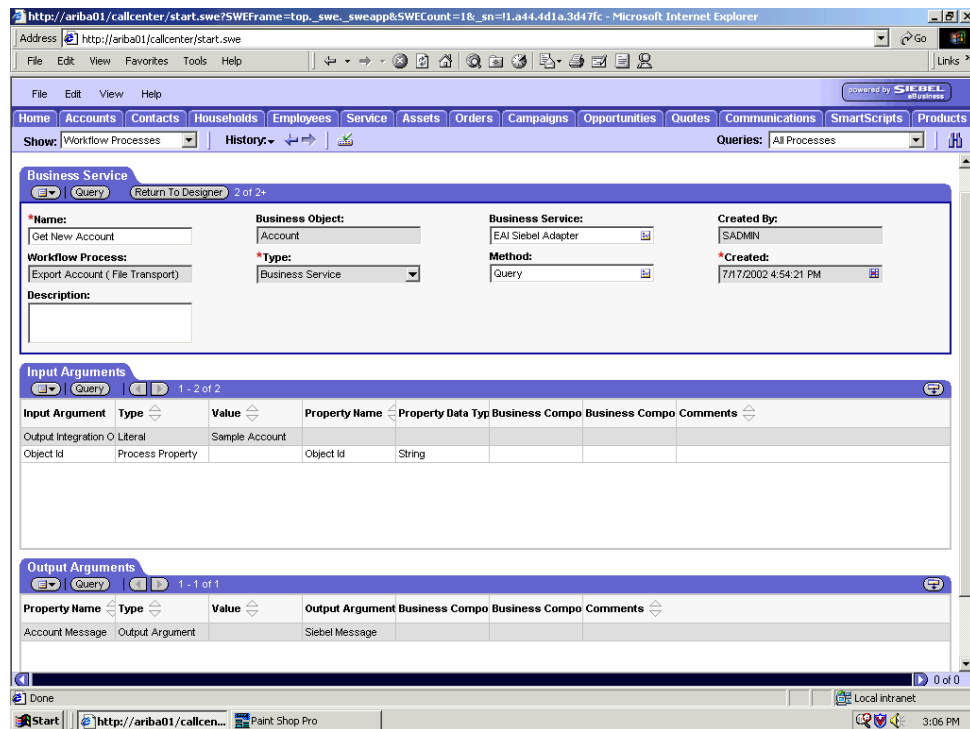
Account メッセージには Siebel Account データが階層形式で含まれています。

Account XML は、ワークフローで XML に変換された Siebel Account データを指定します。

2. Siebel ワークフロー管理ウィンドウを使用して、ワークフローを作成します。

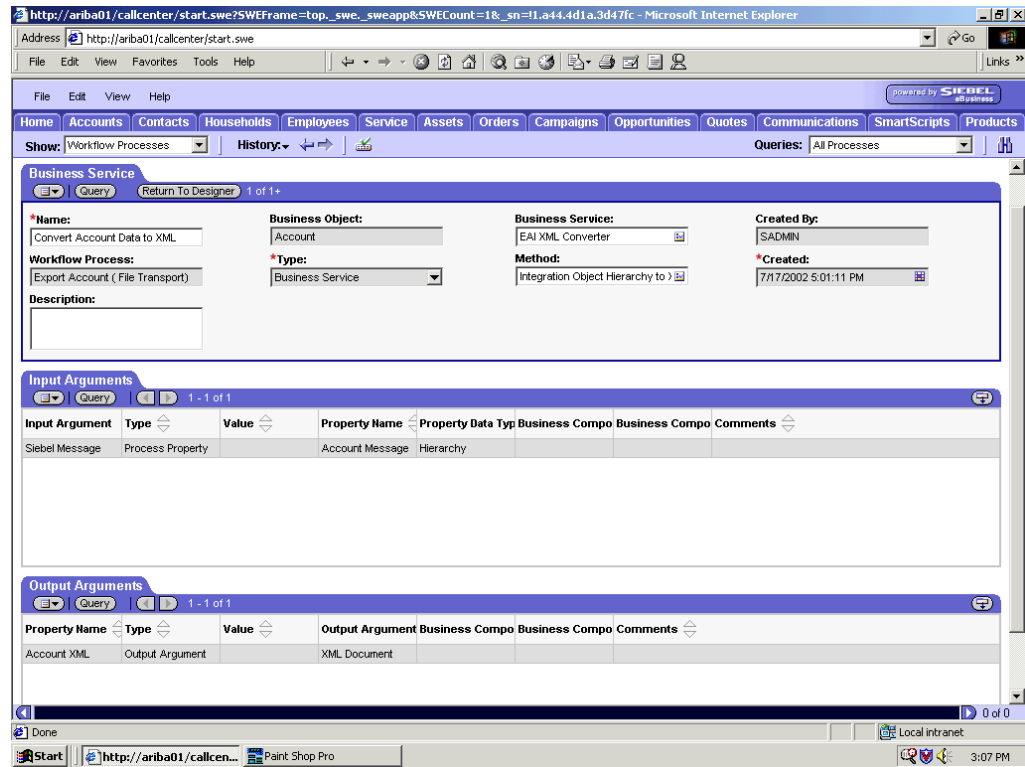
図 A-10 は、Siebel ワークフロー管理ウィンドウの例です。

図 A-10 Siebel ワークフロー管理ウィンドウ



- Account データのインスタンスを受信するために EAI Siebel Adapter ビジネス・サービス・ステップを定義し、Get New Account という名前を付けます。
このビジネス・サービスは、Query メソッドを使用して Siebel から Account 情報を取得します。
このビジネス・サービスからの出力は階層形式で生成されます。

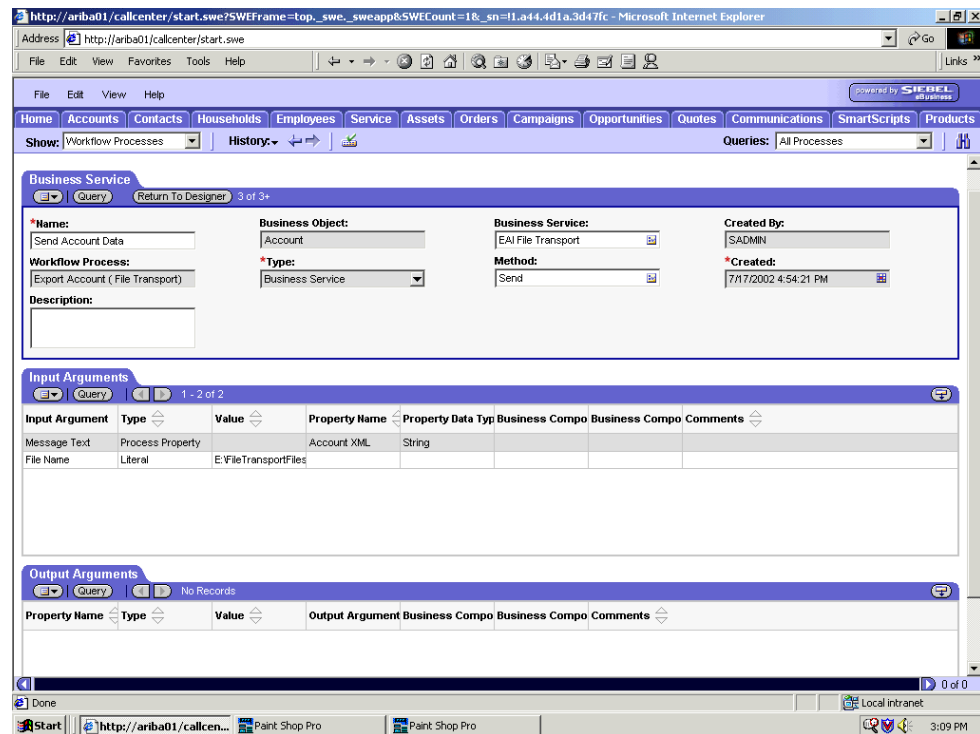
図 A-11 EAI XML Converter の作成



4. 図 A-11 に示すように、EAI XML Converter ビジネス・サービス・ステップを定義し、Convert Account Data to XML という名前を付けます。

このビジネス・サービスは、EAI Siebel Adapter ビジネス・サービスから Account データを階層形式で受信し、それを XML 形式に変換するように定義されています。

図 A-12 EAI XML Converter ビジネス・サービスのファイル・トランスポート



5. 図 A-12 に示すように、EAI File Transport ビジネス・サービス・ステップを定義し、Send Account Data という名前を付けます。

このビジネス・サービスは、EAI XML Converter ビジネス・サービスから Account データを Siebel XML 形式で受信し、Send メソッドを使用して Account XML をファイル・システムの指定されたディレクトリに送信するように定義されています。

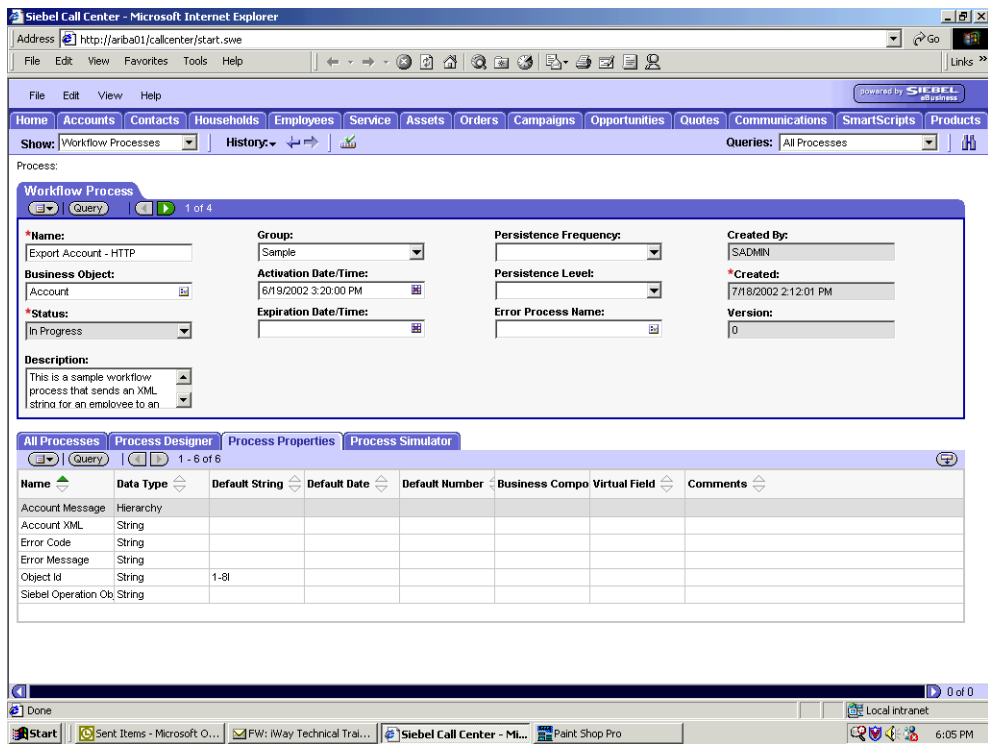
A.2.3 HTTP トランスポートを使用したイベント用の Siebel ワークフローの作成

次の手順は、Siebel ワークフロー管理ウィンドウに示された Siebel ワークフローの例です。このワークフローは、HTTP トランスポートを使用して Siebel Account レコード情報をエクスポートするために設計されたものです。

この手順では、Siebel Call Center アプリケーションで Account レコードが更新されたときに Siebel XML を生成する Siebel ワークフローの作成方法を説明します。

Siebel ワークフローを作成する手順

図 A-13 Siebel ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブ



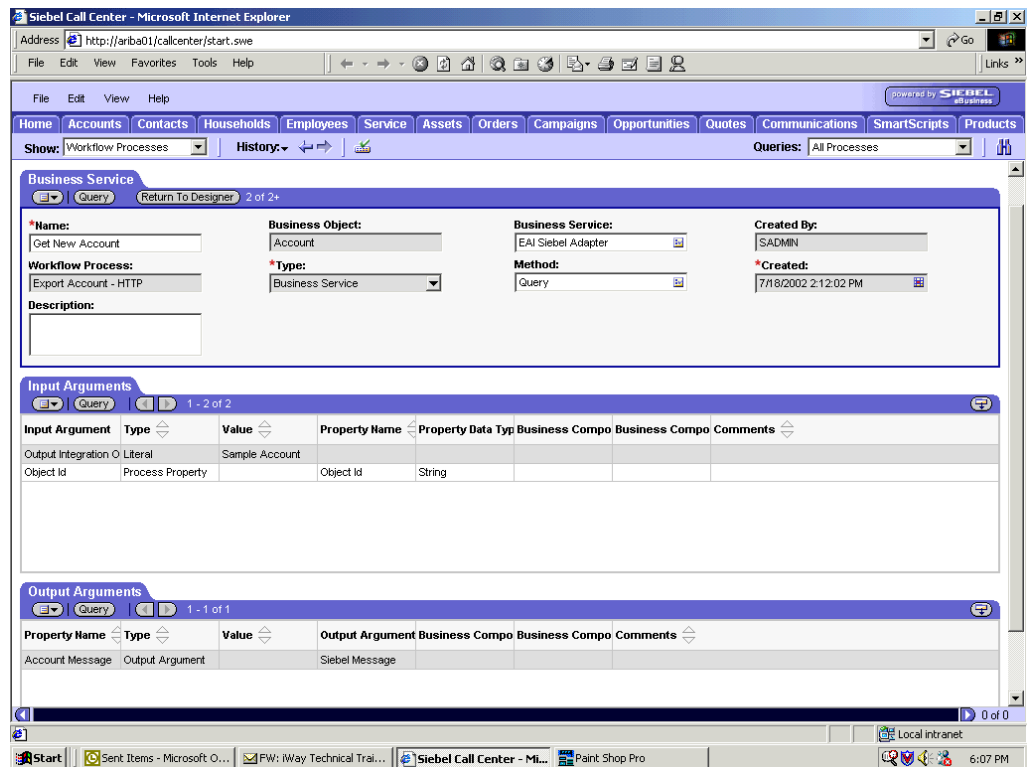
1. 図 A-13 に示すように、ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブで、Account メッセージと Account XML プロセス・プロパティを定義します。

Account メッセージには Siebel Account データが階層形式で含まれています。

Account XML は、ワークフローで XML に変換された Siebel Account データを指定します。

2. Siebel ワークフロー管理ウィンドウを使用して、ワークフローを作成します。

図 A-14 Siebel ワークフロー管理ウィンドウ



3. 図 A-14 に示すように、Account データのインスタンスを受信するために EAI Siebel Adapter ビジネス・サービス・ステップを定義し、Get New Account という名前を付けます。

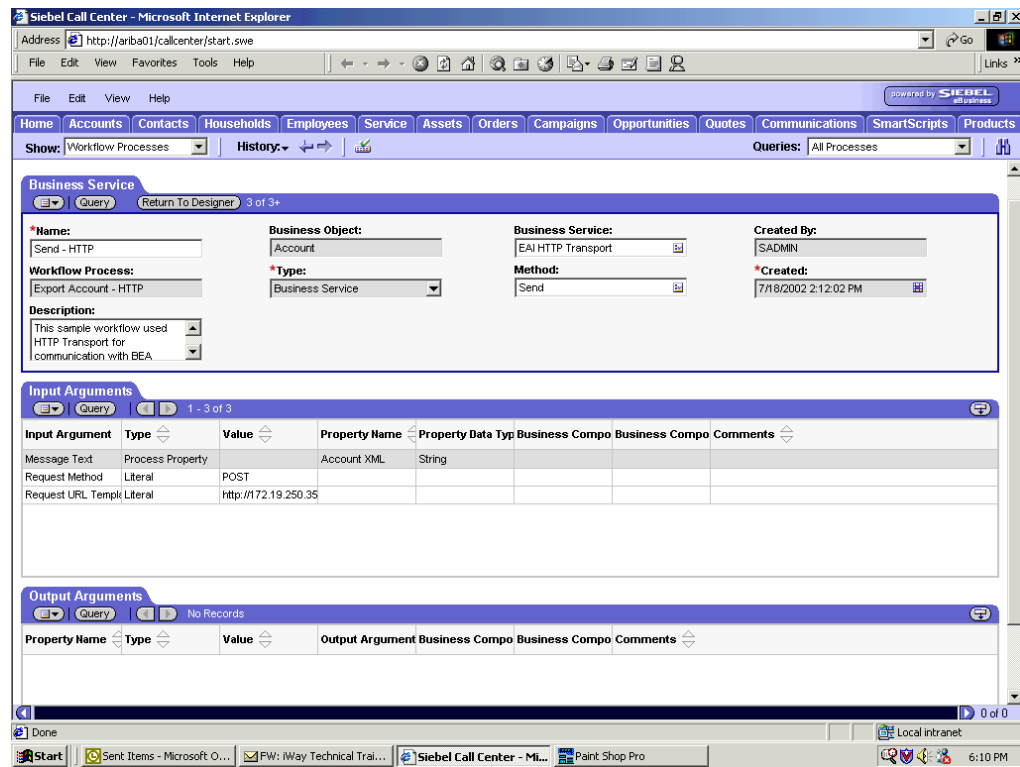
このビジネス・サービスは、Query メソッドを使用して Siebel から Account 情報を取得します。

このビジネス・サービスからの出力は階層形式で生成されます。

4. EAI XML Converter ビジネス・サービス・ステップを定義し、Convert to XML という名前を付けます。

このビジネス・サービスは、EAI Siebel Adapter ビジネス・サービスから Account データを階層形式で受信し、それを XML 形式に変換するように定義されています。

図 A-15 EAI XML Converter ビジネス・コンバータ・ビジネス・サービス



5. 図 A-15 に示すように、EAI HTTP Transport ビジネス・サービス・ステップを定義し、Send - HTTP という名前を付けます。

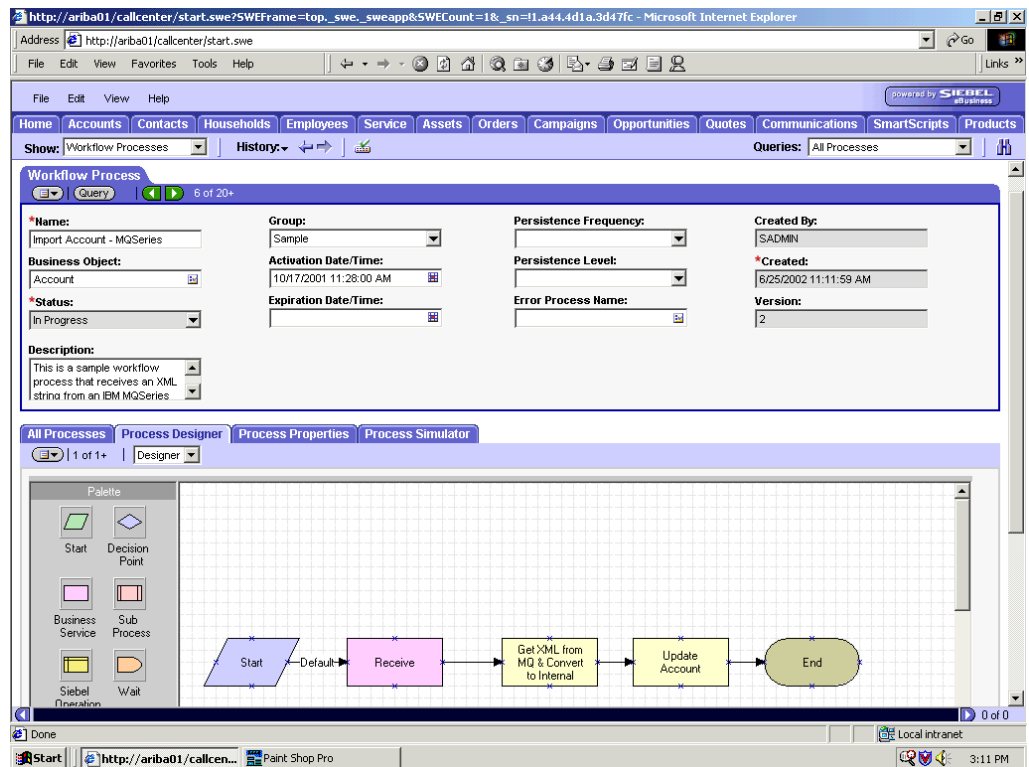
このビジネス・サービスは、EAI XML Converter ビジネス・サービスから Account データを Siebel XML 形式で受信し、Send メソッドを使用して Account XML を HTTP に送信するように定義されています。

A.2.4 MQSeries トランスポートを使用したサービス用の Siebel ワークフローの作成

次の手順は、Siebel ワークフロー管理ウィンドウに示された Siebel ワークフローの例です。このワークフローは、MQSeries トランスポートを使用して Siebel Account レコード情報をインポートするために設計されたものです。

図 A-16 に、サンプルの Siebel ワークフロー管理ウィンドウを示します。

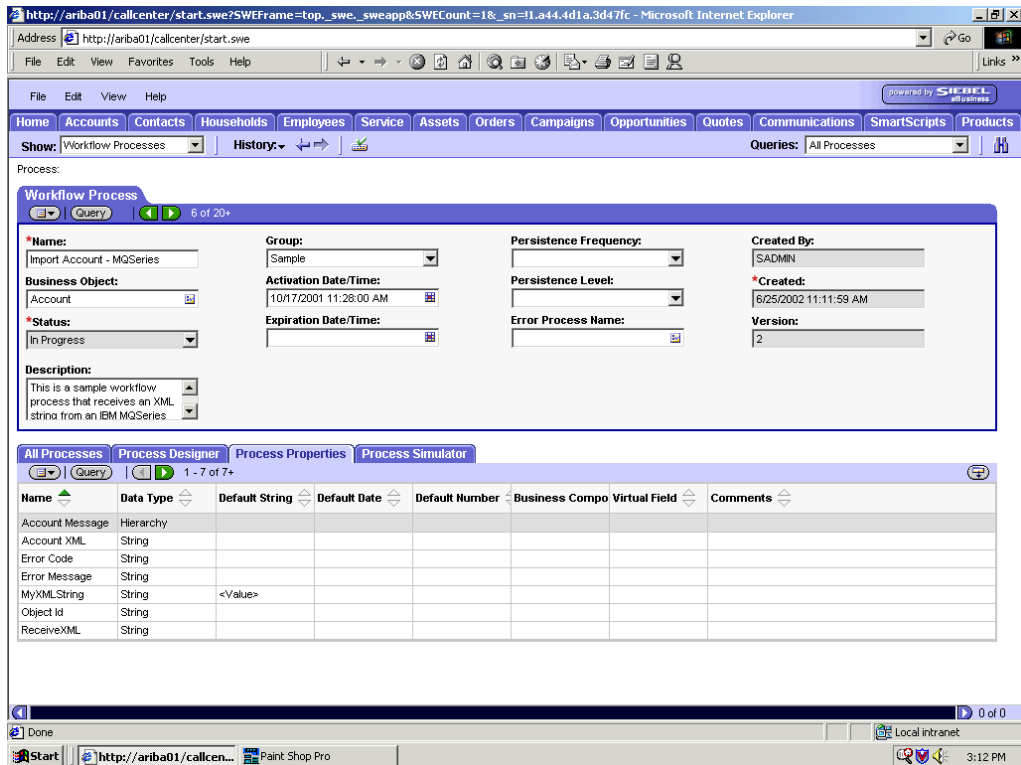
図 A-16 Siebel ワークフロー管理ウィンドウ



この手順では、Siebel Call Center アプリケーションで Account レコードが更新されたときに Siebel XML を生成する Siebel ワークフローの作成方法を説明します。

Siebel ワークフローを作成する手順

図 A-17 Siebel ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブ

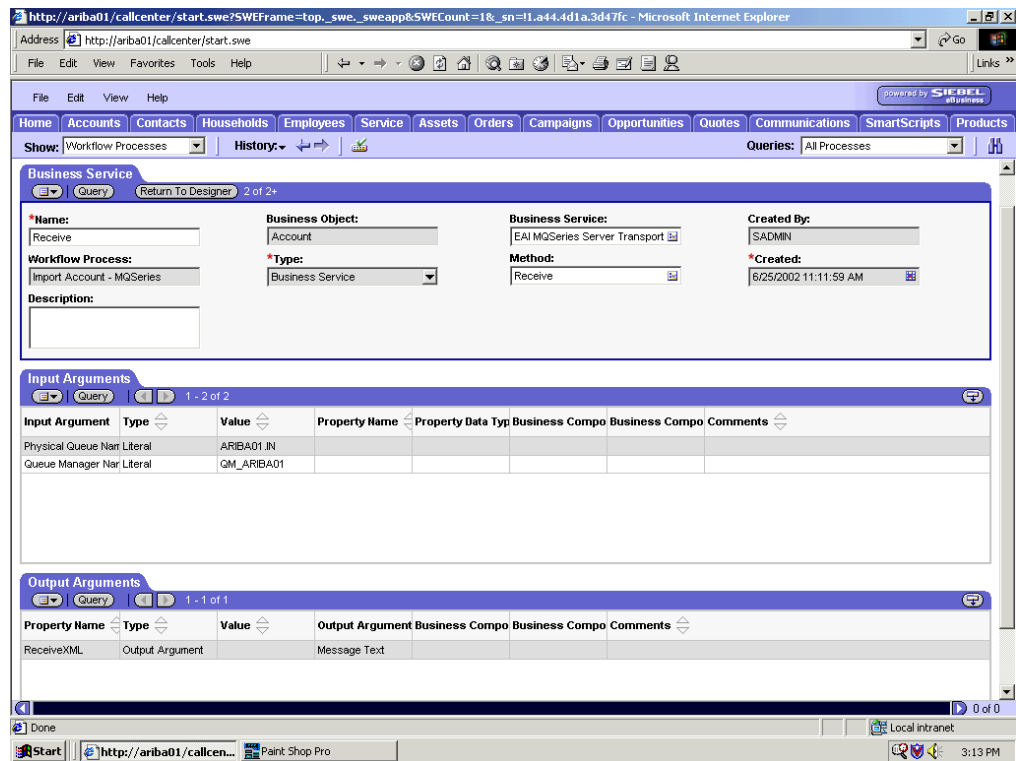


1. 図 A-17 に示すように、ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブで、Account メッセージと Account XML プロセス・プロパティを定義します。

Account メッセージには Siebel Account データが階層形式で含まれています。

Account XML は、ワークフローで XML に変換された Siebel Account データを指定します。

図 A-18 EAI MQ Series Server Transport ビジネス・サービスの作成

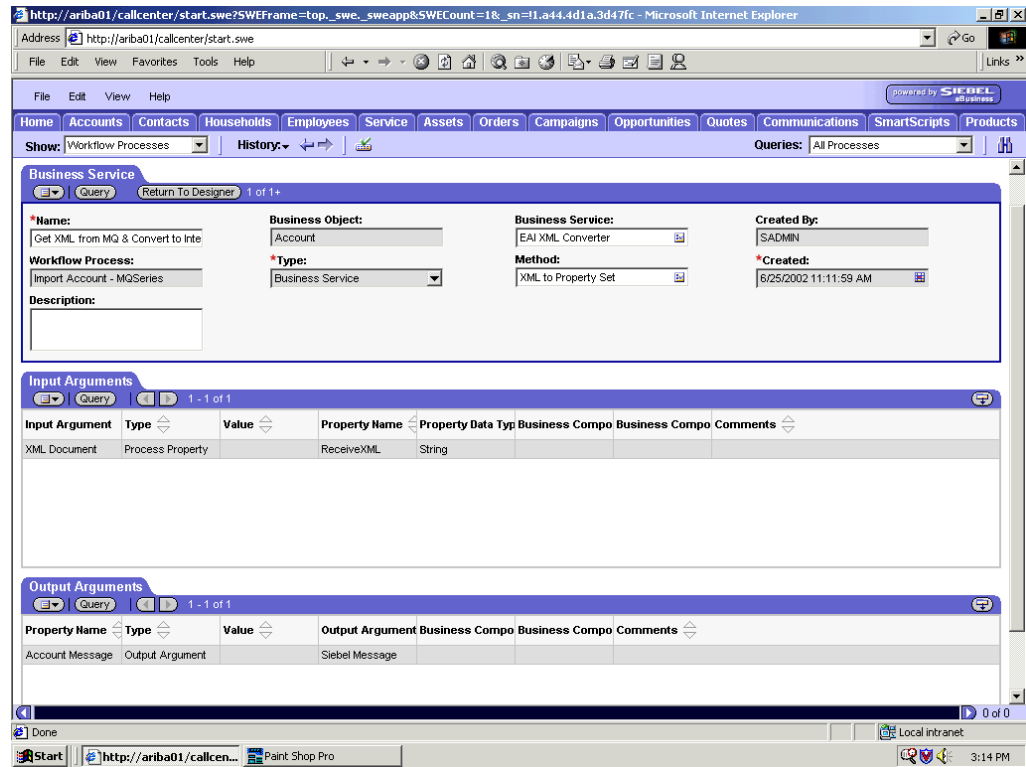


2. 図 A-18 に示すように、EAI MQSeries Server Transport ビジネス・サービス・ステップを定義し、Receive という名前を付けます。

このビジネス・サービスは、MQSeries メッセージ・キューから Account データを受信するように定義されています。

EAI MQSeries Server Transport ビジネス・サービスは、Account データを Siebel XML 形式で受信して EAI XML Converter ビジネス・サービスに送信します。

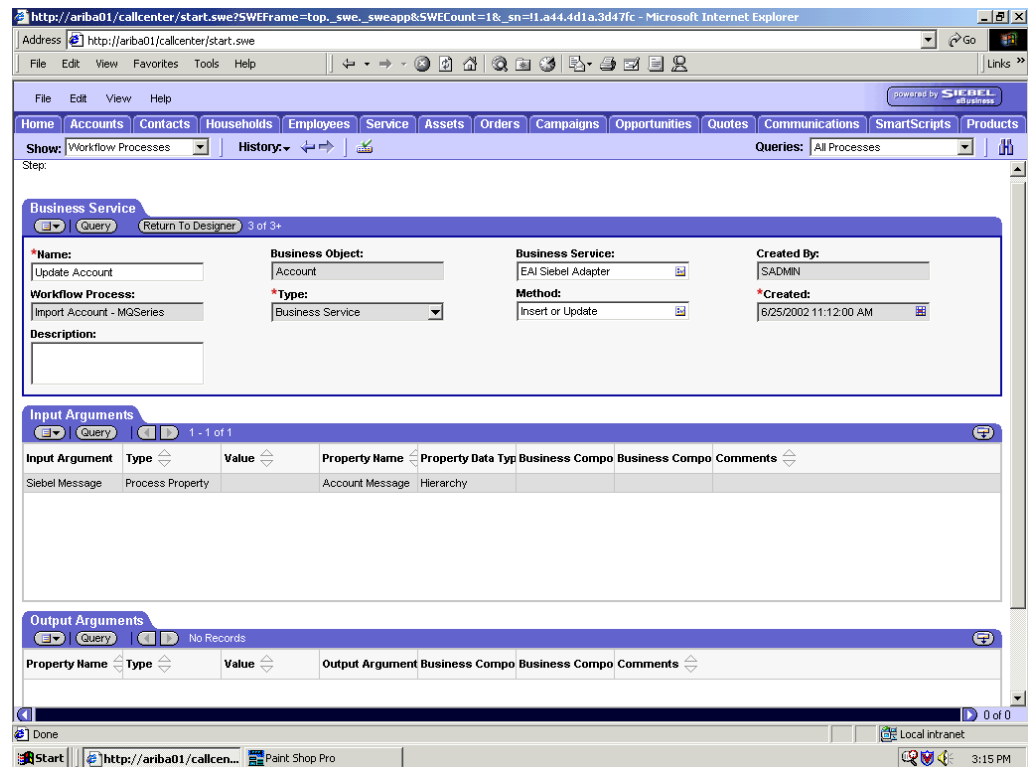
図 A-19 EAI XML Converter ビジネス・サービスの構成



3. 図 A-19 に示すように、EAI XML Converter ビジネス・サービス・ステップを定義し、Get XML from MQ & Convert to XML という名前を付けます。

このビジネス・サービスは、EAI MQSeries Server Transport ビジネス・サービスから Account データを XML 形式で受信し、それを階層形式に変換するように定義されています。

図 A-20 MQ Series トランスポートを使用する EAI XML Converter の Update Account 構成



4. 図 A-20 に示すように、EAI Siebel Adapter ビジネス・サービス・ステップを定義し、Update Account という名前を付けます。

このビジネス・サービスは、EAI XML Converter ビジネス・サービスから階層形式で Account データのインスタンスを受信するように定義されています。

このビジネス・サービスは、Account 情報を Insert または Update メソッドを使用して Siebel に適用します。

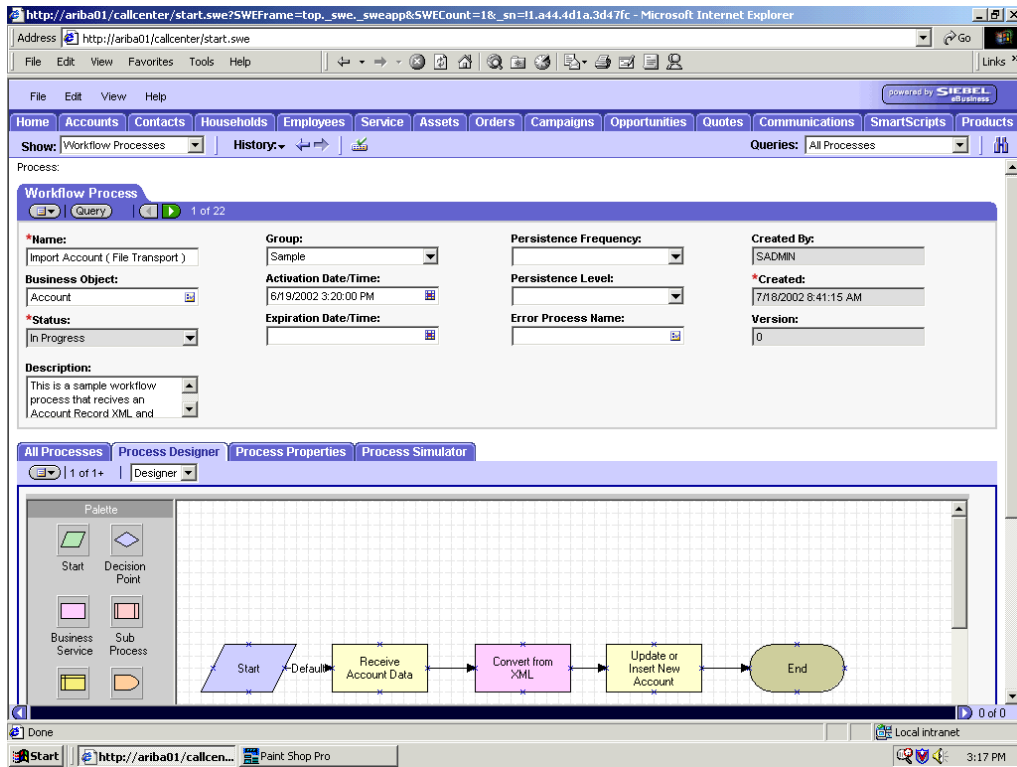
A.2.5 ファイル・トランスポートを使用したサービス用の Siebel ワークフローの作成

次の手順は、Siebel ワークフロー管理ウィンドウに示された Siebel ワークフローの例です。このワークフローは、ファイル・トランスポートを使用して Siebel Account レコード情報をインポートするために設計されたものです。

この手順では、Siebel Call Center アプリケーションで Account レコードが更新されたときに Siebel XML を生成し、Siebel XML をファイル・システム上に置く Siebel ワークフローの作成方法を説明します。

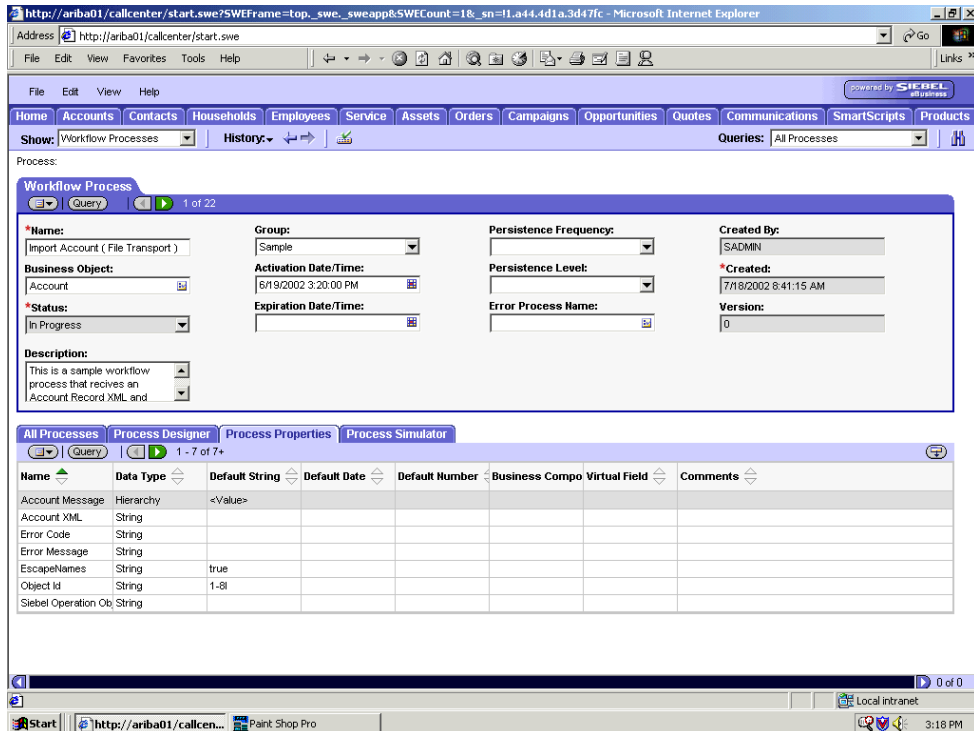
図 A-21 は、プロセス・デザイナー・タブがアクティブな状態の Siebel ワークフロー管理ウィンドウです。

図 A-21 Siebel ワークフロー管理ウィンドウ



Siebel ワークフローを作成する手順

図 A-22 ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブ

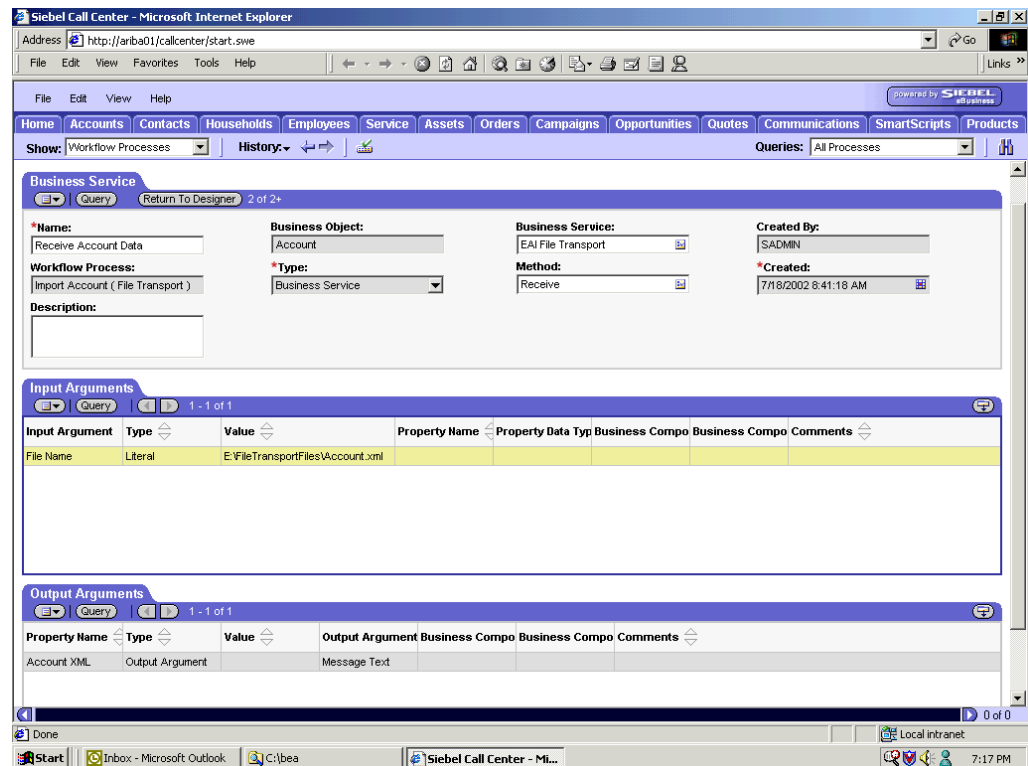


1. 図 A-22 に示すように、ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブで、Account メッセージと Account XML プロセス・プロパティを定義します。

Account メッセージには Siebel Account データが階層形式で含まれています。

Account XML は、ワークフローで XML に変換された Siebel Account データを指定します。

図 A-23 EAI File Transport ビジネス・サービス・ステップ

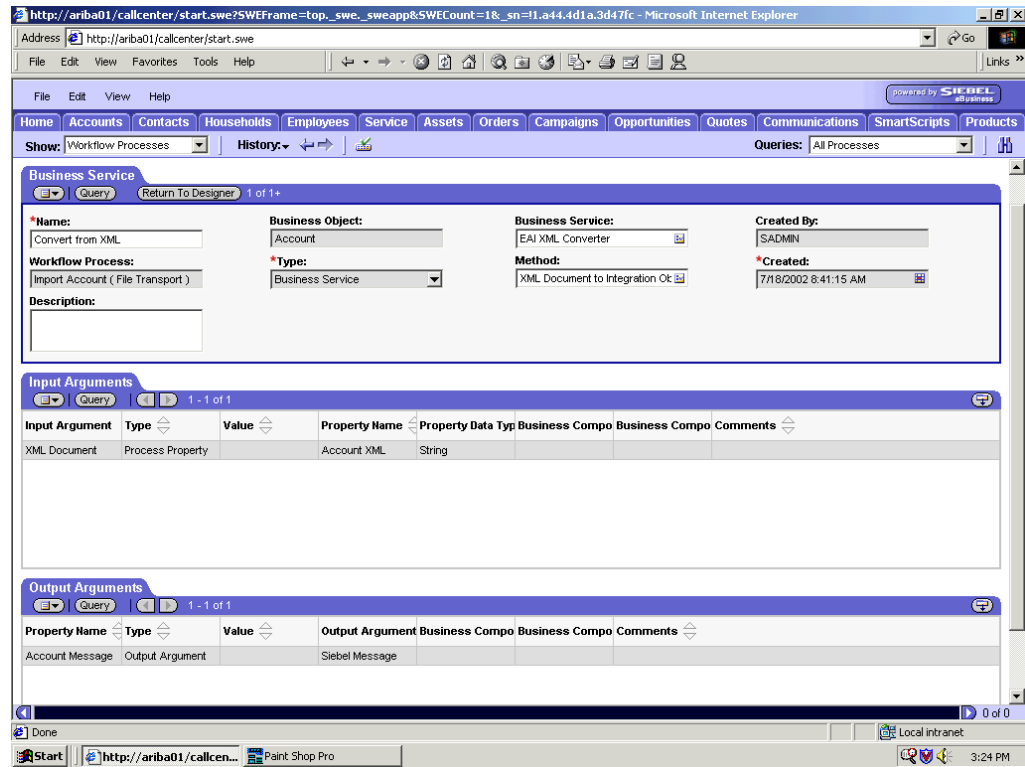


2. 図 A-23 に示すように、EAI FileTransport ビジネス・サービス・ステップを定義し、Receive Account Data という名前を付けます。

このビジネス・サービスは、ファイル・システムから Account データを受信するように定義されています。

EAI File Transport ビジネス・サービスは、Account データを Siebel XML 形式で受信して EAI XML Converter ビジネス・サービスに送信します。

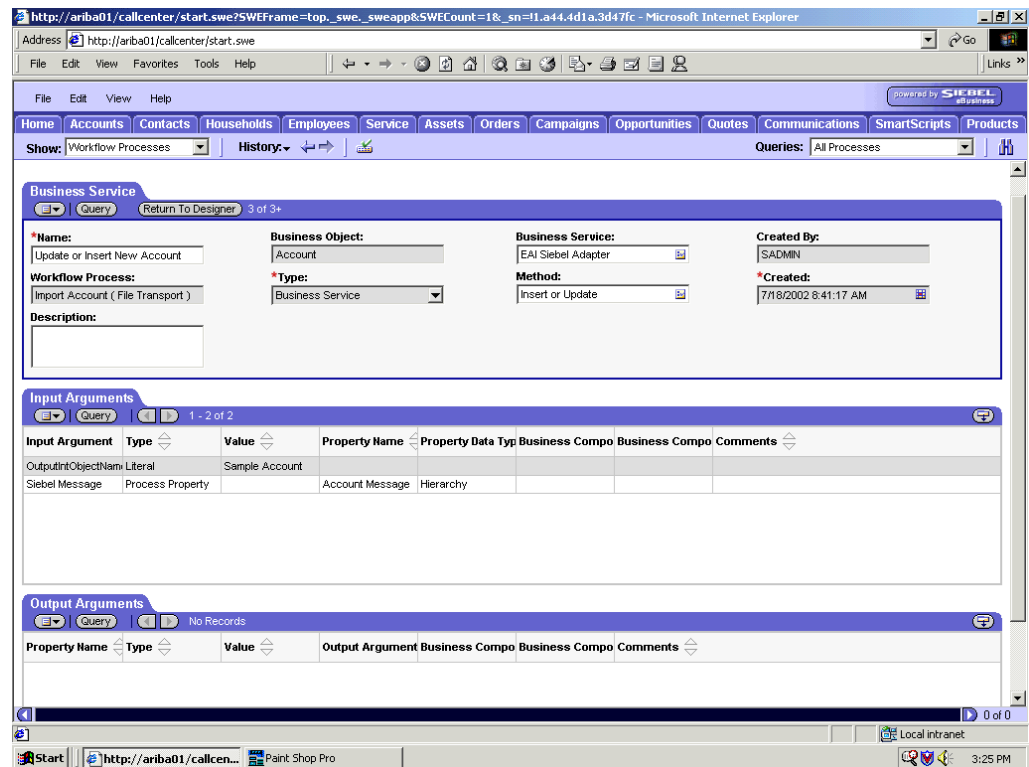
図 A-24 EAI XML Converter ビジネス・サービス・ステップ



3. 図 A-24 に示すように、EAI XML Converter ビジネス・サービス・ステップを定義し、Convert from XML という名前を付けます。

このビジネス・サービスは、EAI File Transport ビジネス・サービスから Account データを XML 形式で受信し、それを階層形式に変換するように定義されています。

図 A-25 Update という名前の EAI Adapter ビジネス・サービス・ステップの作成



4. 図 A-25 に示すように、EAI Siebel Adapter ビジネス・サービス・ステップを定義し、Update or Insert New Account という名前を付けます。

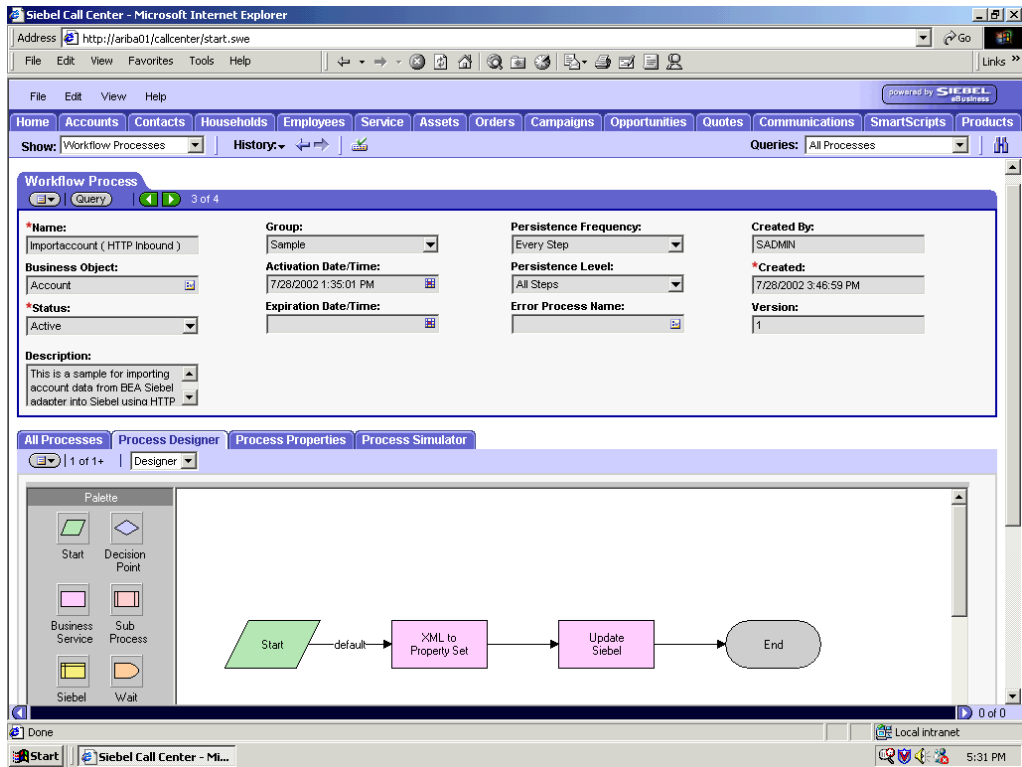
このビジネス・サービスは、EAI XML Converter ビジネス・サービスから階層形式で Account データのインスタンスを受信するように定義されています。

このビジネス・サービスは、Account 情報を Insert または Update メソッドを使用して Siebel に適用します。

A.2.6 HTTP トランスポートを使用したサービス用の Siebel ワークフローの作成

次の手順は、図 A-26 の Siebel ワークフロー管理ウィンドウに示された Siebel ワークフローの例です。次に示すように、このワークフローは、HTTP トランスポートを使用して Siebel Account レコード情報をインポートするために設計されたものです。

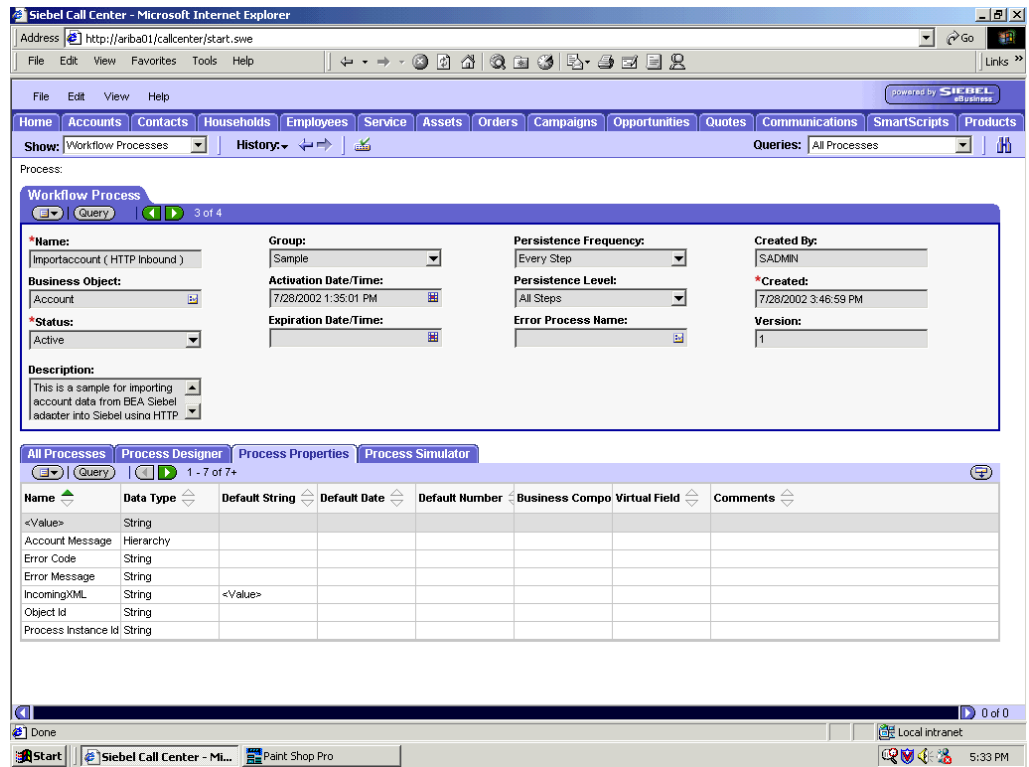
図 A-26 Siebel ワークフロー管理ウィンドウ



次の手順では、Siebel Call Center アプリケーションで Account レコードが更新されたときに Siebel XML を生成し、Siebel XML をファイル・システム上に置く Siebel ワークフローの作成方法を説明します。

Siebel ワークフローを作成する手順

図 A-27 ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブ

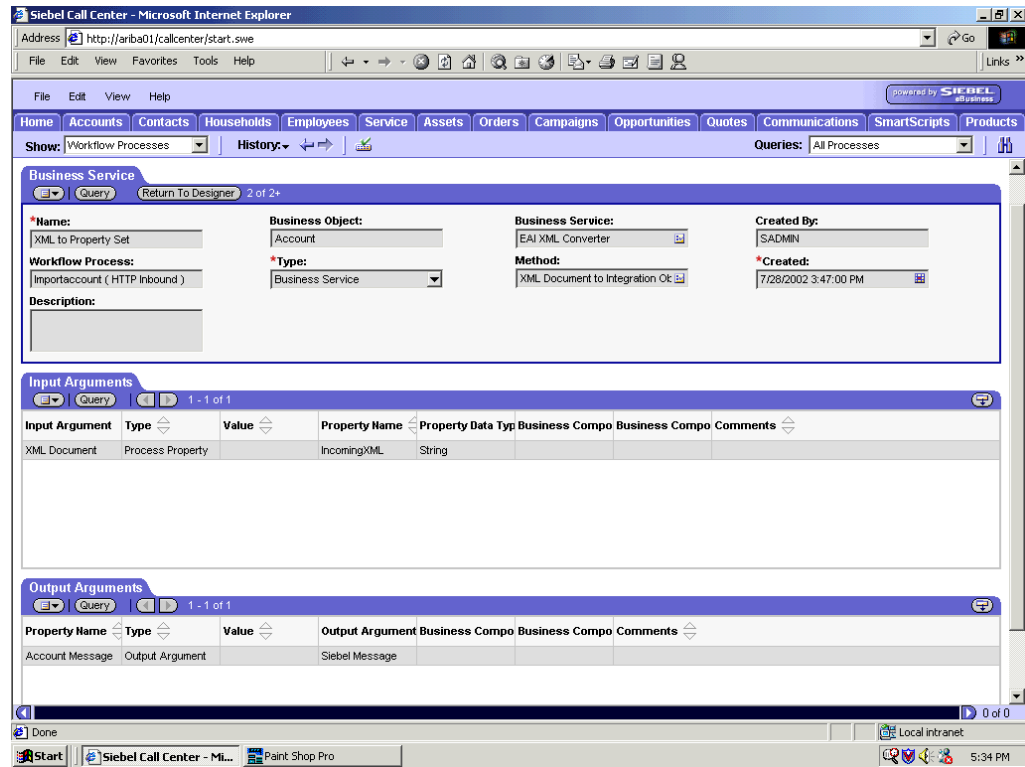


1. 図 A-27 に示すように、ワークフロー・プロセス・ウィンドウの「プロセスのプロパティ」タブで、Account メッセージと Account XML プロセス・プロパティを定義します。

Account メッセージには Siebel Account データが階層形式で含まれています。

Account XML は、ワークフローで XML に変換された Siebel Account データを指定します。

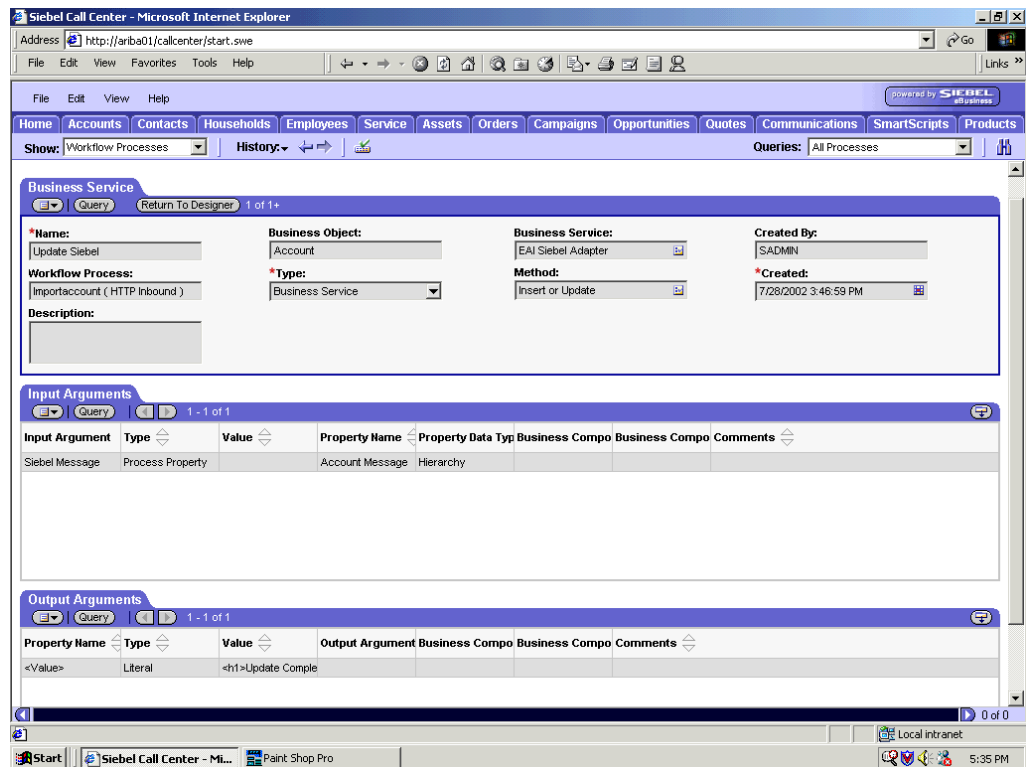
図 A-28 EAI XML Converter ビジネス・サービス・ステップ



2. 図 A-28 に示すように、EAI XML Converter ビジネス・サービス・ステップを定義し、XML to Property Set という名前を付けます。

このビジネス・サービスは、EAI HTTP Transport ビジネス・サービスから Account データを XML 形式で受信し、それを階層形式に変換するように定義されています。

図 A-29 EAI Adapter ビジネス・サービス Update ステップ



3. 図 A-29 に示すように、EAI Siebel Adapter ビジネス・サービス・ステップを定義し、Update Siebel という名前を付けます。

このビジネス・サービスは、EAI XML Converter ビジネス・サービスから階層形式で Account データのインスタンスを受信するように定義されています。

このビジネス・サービスは、Account 情報を Insert または Update メソッドを使用して Siebel に適用します。

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

- BPEL Process Manager
 - OracleAS Adapter for Siebel および, 4-1
 - アダプタとの統合, 4-1, 6-1
- BPEL デザイナ, 4-1, 6-1
- BSE (OracleAS Adapter Business Services Engine) 構成, 2-3
- トラブルシューティング, 10-5
- Business Integration Manager, 1-5

C

- CCI (Common Client Interface), 3-1
- CCI コール, 3-1
- Common Client Interface (CCI), 3-1
- COM 環境, 10-2

E

- EAI (Enterprise Application Integration), 1-5
- Eclipse。『*JDeveloper*』を参照
- EJB (Enterprise Java Beans), 3-1
- Enterprise Application Integration (EAI), 1-5
- Enterprise Connector for J2EE Connector Architecture (J2CA), 1-1
- Enterprise Java Beans (EJB), 3-1
- eScript スクリプト, A-1

H

- HTTP チャンネル, 2-30, 4-35 ~ 4-38
- HTTP トランスポート, A-13, A-25

I

- 「iBSE URL」フィールド, 2-3
- IO (統合オブジェクト) ノード, 2-20
- 作成, 4-39
- 「IWayConfig」パラメータ, 3-4
- 「IWayHome」パラメータ, 3-4
- 「IWayRepoPassword」パラメータ, 3-4
- 「IWayRepoURL」パラメータ, 3-4
- 「IWayRepoUser」パラメータ, 3-4

J

- J2CA (Enterprise Connector for J2EE Connector Architecture), 1-1
- J2CA アーキテクチャ
 - Oracle Application Server Adapter および, 3-1
- J2CA 構成
 - アプリケーション・エクスプローラおよび, 3-3
- J2CA サービス, 4-34, 4-35
- J2CA リソース・アダプタ, 1-1, 3-1
- JAR ファイル, 10-2
- Java データ Bean, 1-5
- Java プログラム・クライアント, 3-1
- JDeveloper, 6-1

L

- 「Loglevel」パラメータ, 3-4

M

- 「ManagedConnectionFactory」パラメータ, 3-4
- MQSeries チャンネル, 2-32
- MQSeries トランスポート, A-4, A-16
- MQ サーバー・チャンネル・パラメータ, 2-33
- MQ サーバー・ポート・パラメータ, 2-33
- MQ サーバー・ホスト・パラメータ, 2-33

O

- Oracle Application Server
 - コネクタのデプロイ, 3-1
- Oracle Application Server Adapter
 - J2CA アーキテクチャおよび, 3-1
 - トラブルシューティング, 10-6
- Oracle JDeveloper, 6-1
- Oracle JDeveloper BPEL デザイナ。『*BPEL デザイナ*』、『*JDeveloper*』、または『*Oracle JDeveloper*』を参照
- Oracle Unified Method (OUM), ix
- OracleAS Adapter
 - インストール・ディレクトリおよび, 3-4
- OracleAS Adapter Business Services Engine (BSE), 1-1, 2-3
- トラブルシューティング, 10-5

OracleAS Adapter for Siebel
BPEL Process Manager および, 4-1
デプロイ, 1-1, 4-2
統合アクセス方法および, 1-5
トラブルシューティング, 10-1 ~ 10-7
OracleAS Adapter J2CA, 2-3 ~ 2-4
アプリケーション・エクスペローラおよび, 10-4
接続先, 10-4
OracleAS Adapter J2CA への接続, 10-4
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」も参照
Oracle データベース, 3-4

P

Process Manager。 「BPEL Process Manager」 を参照

Q

Query メソッド, A-11

R

REQUEST_ACK オプション, 2-34
REQUEST_RESPONSE オプション, 2-34
REQUEST オプション, 2-34

S

「Sample Account」 ノード, 2-20, 4-39
Siebel
接続先, 1-4, 2-5 ~ 2-9, 4-38, 10-2 ~ 10-4, 10-6
切断, 2-9
Siebel Call Center
起動, 4-49
Siebel Call Center の起動, 4-49
Siebel COM データ・インタフェース, 1-5
Siebel Gateway サーバー, 10-2
Siebel Java データ Bean, 1-5
Siebel Tools ウィンドウ, 2-17
Siebel Tools のスキーマ・ウィザード, 2-16, 2-19
Siebel VB スクリプト, A-1
SiebelJI_Common.jar ファイル, 10-2
SiebelJI_enu.jar ファイル, 10-2
Siebel イベント (ランタイム・イベント), A-1
Siebel オブジェクト・マネージャ, 1-2
Siebel からの切断, 2-9
Siebel クライアントのワークフロー管理用画面, A-1
Siebel 統合オブジェクト, 1-2, 2-17
参照, 2-9
Siebel トランスポート, 1-5
「Siebel」 ノード, 4-38 ~ 4-40
Siebel ビジネス・オブジェクト, 1-4, 2-24
参照, 2-9
Siebel ビジネス・コンポーネント, 1-2 ~ 1-6, 2-13, 2-24
Siebel ビジネス・サービス, 1-2 ~ 1-6, 2-13, A-1 ~ A-3
参照, 2-9

Siebel ビジネス・プロセス
起動, 1-4
Siebel への接続, 2-5 ~ 2-9, 4-38, 4-40, 10-2 ~ 10-4, 10-6
Siebel へのターゲットの定義, 2-6
Siebel ワークフロー, 1-5, A-1 ~ A-29
作成, A-3
Siebel ワークフロー管理ウィンドウ, 4-50
Siebel ワークフロー・ポリシー, A-2
SOAP エージェント, 10-5
SOAP フォルト, 10-5
SOAP リクエスト, 10-5 ~ 10-7
エラーおよび, 10-5 ~ 10-7
SOAP レスポンス, 10-5 ~ 10-7
SOA (サービス指向アーキテクチャ), 4-1

W

Web Service Definition Language (WSDL), 2-27 ~ 2-28
Web サービス, 1-1 ~ 1-2, 1-6, 2-3
作成, 2-24 ~ 2-25
テスト, 2-25 ~ 2-26
統合, 4-1
トラブルシューティング, 10-5 ~ 10-7
リポジトリ・プロジェクトおよび, 2-3
Web サービスの作成, 2-24 ~ 2-25
「Web サービスの作成」 ダイアログ, 2-24
Web サービスのテスト, 2-25 ~ 2-26
Web サービス名, 2-24
WSDL (Web Service Definition Language), 2-27 ~ 2-28
WSDL ドキュメント, 4-1
「WSDL のエクスポート」 ダイアログ・ボックス, 4-8
WSDL ファイル, 4-1, 6-1
アプリケーション・エクスペローラおよび, 4-34, 4-35
作成, 4-35
「WSDL ファイル名」 フィールド, 4-41

X

XDR スキーマ
作成, 2-16
XML スキーマ, 1-2, 1-6
作成, 2-12 ~ 2-14, 2-16
「XML スキーマ」 ウィザード, 2-18
XML ドキュメント, 1-4, A-2
XML メッセージ, 1-2, A-2 ~ A-3

あ

アウトバウンド処理, 1-1 ~ 1-5, A-2
アダプタ, 1-1
BPEL Process Manager との統合, 4-1, 6-1
構成, 3-3
デプロイ, 3-1 ~ 3-3, 4-2
トラブルシューティング, 10-1 ~ 10-7
アダプタ構成, 3-3
更新, 3-3

アダプタ構成の更新, 3-3
アダプタ・タイプ
リソース, 3-3
「アダプタ」ノード, 4-38, 4-40
アダプタのデプロイ, 3-1 ~ 3-3, 4-2
アダプタの例外, 10-4
アプリケーション・アダプタ
BPEL Process Manager との統合, 4-1, 6-1
構成, 3-3
デプロイ, 3-1 ~ 3-3, 4-2
トラブルシューティング, 10-1 ~ 10-7
アプリケーション・エクスプローラ, 1-2, 1-6, 2-10
~ 2-12, 4-7
J2CA 構成および, 3-3
OracleAS Adapter J2CA および, 10-4
WSDL ファイルおよび, 4-35, 4-34
ターゲットおよび, 4-40
テストおよび, 10-2 ~ 10-3
デバッグおよび, 10-2 ~ 10-3
トラブルシューティング, 10-2 ~ 10-3
アプリケーション・システム・オブジェクト
表示, 2-10

い

イベント, 1-2
構成, 2-28 ~ 2-36
作成, 4-40 ~ 4-41
トリガー, 4-49 ~ 4-53
ワークフローと, A-4 ~ A-16
イベント・アダプタ, 2-28 ~ 2-36
イベント・スキーマ, 2-16
イベント統合, 4-34
「イベント」ノード, 2-30
イベントの構成, 2-28 ~ 2-36
イベントの作成, 4-40 ~ 4-41
イベントのトリガー, 4-49 ~ 4-53
イベント・ポート
作成, 4-40 ~ 4-41
イベント・メッセージ, 4-34, 4-35
インストール・ディレクトリ, 3-4
インバウンド J2CA サービス, 4-34, 4-35
インバウンド処理, 1-1 ~ 1-5, A-3

え

「エラー・ディレクトリ」パラメータ, 2-34, 2-36
エラー・メッセージ, 10-2 ~ 10-7
ターゲット・システムおよび, 10-6
エンベロープ・タイプ・リスト, 2-19

お

オブジェクト・マネージャ, 1-2
「オブジェクト・マネージャ」パラメータ, 2-7

か

管理対象コネクタ・ファクトリ, 3-3
管理対象コネクタ・ファクトリ・オブジェクト, 3-3
複数, 3-5

き

「既存のサービス名」リスト, 2-24
「キュー・マネージャ名」パラメータ, 2-33

け

「言語」パラメータ, 2-7, 10-2

こ

構成
上書き, 3-4
接続先, 2-5
定義, 2-3 ~ 2-4
「構成」ノード, 2-4
構成パラメータ, 3-4
IWayConfig, 3-4
IWayHome, 3-4
IWayRepoPassword, 3-4
IWayRepoURL, 3-4
IWayRepoUser, 3-4
Loglevel, 3-4
コネクタ
Oracle Application Server へのデプロイ, 3-1
コネクタ・ファクトリ, 3-3
コネクタ・ファクトリ・オブジェクト, 3-3
複数, 3-5

さ

サーバー・ポート・パラメータ, 2-31, 4-37
サービス, 1-2
作成, 2-24 ~ 2-25
テスト, 2-25 ~ 2-26
ワークフローと, A-16 ~ A-29
サービス指向アーキテクチャ (SOA), 4-1
「サービス」ノード, 2-25
「サービス・プロバイダ」リスト, 2-3 ~ 2-4
サービス名, 2-24
サービス・スキーマ
作成, 2-12 ~ 2-14
サブレット, 3-1

し

システム・オブジェクト
表示, 2-10
システム・オブジェクトの表示, 2-10
実行時, 2-3, 4-38 ~ 4-41
「詳細」タブ, 2-7, 2-34, 2-36
「処理モード」パラメータ, 2-36
「新規構成」ダイアログ・ボックス, 2-3 ~ 2-4
シン・クライアント, 10-2

す

スキーマ
作成, 2-12 ~ 2-14, 2-16
スキーマに関する考慮事項, 2-13
スキーマの場所フィールド, 4-39

スキーマの作成, 2-12 ~ 2-14
スクリプト, A-1
「スレッド制限」パラメータ, 2-34, 2-36

せ

設計時, 2-3, 4-1, 4-34
構成, 4-7
接続
確立, 2-5 ~ 2-9, 2-19, 4-38
削除, 2-10
閉じる, 2-9
「接続」ダイアログ・ボックス, 4-38
接続タイプ, 10-2
接続パラメータ, 10-2
オブジェクト・マネージャ, 2-7
言語, 2-7
リポジトリ・マネージャ, 2-7
接続を閉じる, 2-9
「説明」フィールド, 2-24

そ

「操作モード」パラメータ, 2-34

た

ターゲット
削除, 2-10
接続先, 2-5 ~ 2-9, 4-38, 4-40, 10-2 ~ 10-4
切断, 2-9
編集, 2-10
定義, 2-6
「ターゲット・タイプ」リスト, 2-6
ターゲットの削除, 2-10
「ターゲットの追加」ダイアログ・ボックス, 2-6
ターゲットの編集, 2-10
ターゲット・システム
エラーおよび, 10-6

ち

チャンネル
テストおよびデバッグ, 4-38
起動, 2-32 ~ 2-36, 4-38
作成, 1-2, 2-29 ~ 2-36, 4-35 ~ 4-38
削除, 2-36
停止, 2-32 ~ 2-36, 4-38
編集, 2-36
チャンネル, 「リスナー」も参照
チャンネル構成パラメータ
MQ サーバー・チャンネル, 2-33
MQ サーバー・ポート, 2-33
MQ サーバー・ホスト, 2-33
エラー・ディレクトリ, 2-34, 2-36
キュー・マネージャ名, 2-33
サーバー・ポート, 2-31, 4-37
処理モード, 2-36
スレッド制限, 2-34, 2-36

操作モード, 2-34
同期タイプ, 2-31, 2-34, 2-35, 4-37
ドキュメント・タイプ XML, 2-33
ファイル・マスク, 2-35
ポート, 2-31, 4-37
ポーリング間隔, 2-36
ポーリング場所, 2-35
メッセージ待機間隔, 2-34
リクエスト・キュー名, 2-34
レスポンス /Ack ディレクトリ, 2-35
チャンネル・タイプ
HTTP, 2-30, 4-35 ~ 4-38
MQSeries, 2-32
ファイル, 2-35
「チャンネル」ノード, 2-30, 4-35
「チャンネルの追加」ダイアログ・ボックス, 2-30, 4-35
チャンネルの作成, 4-35 ~ 4-38
チャンネルの削除, 2-36
チャンネルの編集, 2-36
「チャンネル」フィールド, 4-41

て

データベース
Oracle, 3-4
接続先, 3-4
データベース接続
開く, 3-4

と

同期タイプ・パラメータ, 2-31, 2-34 ~ 2-35, 4-37
統合アクセス方法
OracleAS Adapter for Siebel および, 1-5
統合オブジェクト, 1-2, 2-17
参照, 2-9
統合オブジェクト (IO) ノード, 2-20
作成, 4-39
「ドキュメント・タイプ XML」パラメータ, 2-33
トラブルシューティング, 10-1 ~ 10-7
BSE, 10-5
Web サービス, 10-5 ~ 10-7
アプリケーション・エクスペローラ, 10-2 ~ 10-3
トランザクション処理, 1-2
トランスポート, 1-5
HTTP, A-13, A-25
MQSeries, A-4, A-16
ファイル, A-8 ~ A-13, A-21
トレース情報, 10-1

な

ナビゲーション・パス, 1-4

に

入力引数, 4-52
入力引数タブ, 4-52

の

ノード

- Sample Account, 2-20, 4-39
- Siebel, 4-38 ~ 4-40
- アダプタ, 4-38, 4-40
- イベント, 2-30
- 構成, 2-4
- サービス, 2-25
- 接続済, 2-9, 4-38
- 切断済, 2-9
- チャンネル, 2-30, 4-35
- 統合オブジェクト, 2-20, 4-39
- ポート, 4-35
- メソッド, 2-25
- ノードのリスト, 10-2
- 「ノード名」フィールド, 4-39
- ノード・リスト, 10-2

は

- パスワード, 3-4
- パスワード・パラメータ, 10-2
- バッチ・スクリプト, 10-2
- パラメータ・タイプ
 - 構成, 3-4
 - 接続, 2-7, 10-2
 - チャンネル構成, 2-31 ~ 2-36, 4-37

ひ

- ビジネス・イベント, 1-2
- ビジネス・オブジェクト, 1-4, 2-24
 - 参照, 2-9
- ビジネス・コンポーネント, 1-2 ~ 1-6, 2-13, 2-24
- ビジネス・サービス, 1-2 ~ 1-6, 2-13, A-1 ~ A-3
 - 作成, 2-24 ~ 2-25
 - 参照, 2-9
 - テスト, 2-25 ~ 2-26
- ビジネス・サービス・リスト, 2-18

ふ

- ファイル・チャンネル, 2-35
- 「ファイル・マスク」パラメータ, 2-35
- ファイル・トランスポート, A-8 ~ A-13, A-21
- フォルト・コード要素, 10-5
- フォルト文字列要素, 10-5
- プロジェクト
 - リポジトリ, 2-3 ~ 2-4
- プロセス・シミュレータ・タブ, 4-53
- プロセス・デザイナー・タブ, 4-52
- 「プロセスのプロパティ」タブ, A-4, A-23, A-27
- 「プロトコル」リスト, 2-20, 4-36, 4-39
- プロパティ, 3-4

ほ

- ポート
 - 作成, 4-40 ~ 4-41, 2-28
- 「ポート」ノード, 4-35

- ポート・パラメータ, 2-31, 4-37
- ポート番号パラメータ, 2-4
- 「ポーリング位置」パラメータ, 2-35
- 「ポーリング間隔」パラメータ, 2-36
- ホスト名パラメータ, 2-4

め

- 「メソッド」ノード, 2-25
- 「メソッド名」フィールド, 2-24
- メタデータ
 - 参照, 2-9
 - 表示, 2-10
- メタデータの参照, 2-9
- メタデータ表, 2-11
- メッセージ, 1-2
- 「メッセージ待機間隔」パラメータ, 2-34
- メッセージ・タイプ
 - イベント, 4-34, 4-35
- メッセージ交換
 - 同期, 1-2
 - 非同期, 1-2
- メディエータ・アウトバウンド・プロセス, 5-2
- メディエータ・インバウンド・プロセス, 5-11

ゆ

- ユーザー名パラメータ, 10-2

ら

- ライセンス, 2-24
- ライセンスおよびメソッド・ダイアログ・ボックス, 2-24
- 「ライセンス」フィールド, 2-24
- ランタイム・イベント (Siebel イベント), A-1
- ランタイム統合, 4-49

り

- リクエスト
 - 実行, 10-4
- 「リクエスト・キュー名」パラメータ, 2-34
- リクエスト・スキーマ, 2-13
- 「リクエスト」タブ, 2-33, 2-35
- リスナー。「チャンネル」も参照
- リポジトリ・マネージャ・パラメータ, 2-7
- リポジトリ・プロジェクト
 - Web サービスおよび, 2-3
 - 作成, 2-3 ~ 2-4
- リポジトリ・プロジェクトの作成, 2-3 ~ 2-4
- リポジトリ名パラメータ, 10-3 ~ 10-4

れ

- レスポンス /Ack ディレクトリ・パラメータ, 2-35
- レスポンス・スキーマ, 2-13
- 「レスポンス」タブ, 2-34 ~ 2-35

ろ

ログオン・パラメータ, 10-2
ログ・ファイル, 10-1
ログ・レベル
 上書き, 3-4

わ

ワークフロー, A-1 ~ A-29
 作成, A-3
ワークフロー・プロセス・ウィンドウ, 4-50, A-4 ~
 A-23, A-27
ワークフロー・ポリシー, A-2
ワークフロー要素, 4-52