

Oracle® Fusion Middleware Application Adapters

Application Adapter ベスト・プラクティス・ガイド for
12c (12.2.1.2.0)

E85895-01

2016 年 12 月

Oracle WebLogic Server Application Adapter のベスト・
プラクティスについて説明します。

Oracle Fusion Middleware Oracle WebLogic Server 12c (12.2.1.2.0) Application Adapters 12c (12.2.1.2.0)

ベスト・プラクティス・ガイド

E85895-01

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

原著者 : Stefan Kostial

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

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

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

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

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

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

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

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

目次

はじめに	v
対象読者	v
ドキュメントのアクセシビリティについて	v
関連ドキュメント	v
表記規則	vi
1 一般的なベスト・プラクティス	
管理対象接続について	1-1
SOA、BPM および OSB の高可用性クラスタ環境での Oracle Application Adapters の構成	1-3
2つのシステムを含むクラスタ環境への SOA または OSB のインストールと構成	1-4
Oracle Application Adapters のデプロイ	1-5
iBSE コネクタ・アプリケーションのデプロイと構成	1-6
iBSE 構成のデプロイ	1-15
アプリケーション・エクスプローラで BSE 構成を構成する	1-15
主な注意事項	1-17
アウトバウンド・プロセスのための高可用性 (HA) ファイル・アダプタの構成	1-17
アウトバウンド・プロセスの構成	1-22
インバウンド・プロセス用の Oracle HTTP Server の構成	1-23
インバウンド・プロセスの構成	1-26
シングルトン・テスト	1-26
推奨されるリポジトリ・タイプ	1-28
入力 XML ドキュメントの生成	1-28
前提条件	1-28
Application Explorer を使用した WSDL ドキュメントの作成	1-28
リクエスト・スキーマからの入力 XML ファイルの生成	1-29
Oracle Enterprise Manager コンソールでの生成済入力 XML ファイルの使用	1-32
2 Oracle Application Adapter for SAP R/3 のベスト・プラクティス	
接続プールについて	2-1
セキュリティ	2-2
ロード・バランシング	2-2
SAP メッセージ・サーバーによる SAP へのインバウンドのロード・バランシング	2-3
SAP ゲートウェイによる SAP からのアウトバウンドのロード・バランシング	2-3
エンコーディング	2-4

SAP サーバーの動作	2-5
IDoc の構成	2-5
IDoc メタデータ	2-6
3 Oracle Application Adapter for Siebel のベスト・プラクティス	
Siebel のベスト・プラクティス	3-1
Siebel の接続プーリングの構成	3-2
4 Oracle Application Adapter for PeopleSoft のベスト・プラクティス	
PeopleSoft イベント (インバウンド処理)	4-1
5 Oracle Application Adapter for J.D. Edwards OneWorld のベスト・プラクティス	
接続プーリングおよびロード・バランシング	5-1
セキュリティ	5-2
複数の接続および複数のユーザー	5-2
Unicode サポート	5-2
配信確認およびエラー・メッセージ	5-2

用語集

索引

はじめに

『Oracle Fusion Middleware Oracle WebLogic Server Application Adapter ベスト・プラクティス・ガイド』へようこそ。このドキュメントでは、すべての Oracle Application Adapters for Oracle WebLogic Server に共通する一般的なベスト・プラクティスについて説明します。

対象読者

このドキュメントは、Oracle Application Adapters for Oracle WebLogic Server を使用するシステム管理者およびアプリケーション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 for SAP R/3 ユーザーズ・ガイド
- 『Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for Siebel ユーザーズ・ガイド』

- 『Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for PeopleSoft ユーザーズ・ガイド』
- 『Oracle Fusion Middleware Oracle WebLogic Server Application Adapter for J.D. Edwards OneWorld ユーザーズ・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、次の場所にある OUM FAQ を参照してください。

http://my.oracle.com/portal/page/myo/ROOTCORNER/KNOWLEDGEAREAS1/BUSINESS_PRACTICE/Methods/Learn_about_OUM.html

表記規則

このドキュメントでは次の表記規則を使用します。

規則	意味
太字	太字は、操作に関連する Graphical User Interface 要素、または本文中で定義されている用語および用語集に記載されている用語を示します。
イタリック	イタリックは、ユーザーが特定の値を指定するプレースホルダ変数を示します。
固定幅フォント	固定幅フォントは、段落内のコマンド、URL、サンプル内のコード、画面に表示されるテキスト、または入力するテキストを示します。

一般的なベスト・プラクティス

この章では、すべての Oracle Application Adapters for Oracle WebLogic Server に共通する一般的なベスト・プラクティスについて説明します。

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

`<ADAPTER_HOME>` は、以下を意味します。

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

この章の内容は次のとおりです。

- [1.1 項「管理対象接続について」](#)
- [1.2 項「SOA、BPM および OSB の高可用性クラスタ環境での Oracle Application Adapters の構成」](#)
- [1.3 項「推奨されるリポジトリ・タイプ」](#)
- [1.4 項「入力 XML ドキュメントの生成」](#)

1.1 管理対象接続について

Oracle WebLogic Server (WLS) は、Oracle J2CA リソース・アダプタへの接続プールを作成して維持します。これらは管理対象接続と呼ばれます。管理対象接続はそれぞれ、(アダプタ・ターゲットとして公開された) バックエンドのエンタープライズ情報システム (EIS) システムへの固有の接続セットを維持します。このセットは最初は空で、その管理対象接続を使用している SOA プロセスが実行されて特定のアダプタ・ターゲット接続がリクエストされた場合にのみ追加されます。

管理対象接続は、特定のアダプタ・ターゲットへの接続を 1 つのみ維持し、そのターゲットへの接続がリクエストされるたびにその接続を再利用します。WLS が管理対象接続を破棄すると、管理対象接続はその内部 EIS 接続すべてをクローズします。

以降の手順で、複数の Oracle アダプタ・ターゲットへの参照を含む SOA プロセスを使用するシナリオを例に挙げて詳しく説明します。

1. SOA プロセスが Oracle リソース・アダプタ ("eis/OracleJCAAdapter/DefaultConnection") への管理対象接続をリクエストし、"Managed1" という新しい空の管理対象接続を受け取ります。
2. SOA プロセスは、Managed1 を使用して、アダプタ・ターゲット MySAP/SAP1 のサービスを呼び出します。
3. Managed1 は新しい MySAP アダプタ・インスタンスを作成してターゲット SAP1 に対してアクティブ化し、その内部 EIS 接続セットに追加します。
4. サービスが呼び出されてレスポンスが返され、SOA プロセスが続行されます。
5. SOA プロセスは、Managed1 を使用して、アダプタ・ターゲット MySAP/SAP2 のサービスを呼び出します。
6. Managed1 は新しい MySAP アダプタ・インスタンスを作成してターゲット SAP2 に対してアクティブ化し、その内部 EIS 接続セットに追加します。
7. サービスが呼び出されてレスポンスが返され、SOA プロセスが続行されます。
8. SOA プロセスは、Managed1 を使用して、アダプタ・ターゲット JDEdwards/JDE1 のサービスを呼び出します。
9. Managed1 は新しい JDEdwards アダプタ・インスタンスを作成してターゲット JDE1 に対してアクティブ化し、その内部 EIS 接続セットに追加します。
10. SOA プロセスが終了します。WLS は Managed1 を接続プールに返します。

SOA プロセスが終了すると、他の SOA プロセスが Managed1 を使用できるようになります。その内部 EIS 接続 (SAP1、SAP2、JDE1) はオープンしたままです。

Managed1 からこれらのターゲットへの接続がプロセスによって次回要求されると、これらの接続は再利用されます。これらは、WLS が Managed1 を破棄した場合のみクローズされます。

図 1-1 に示すように、WLS 管理コンソールで管理対象接続プールを監視できます。

図 1-1 「アウトバウンド接続プール」タブ

Outbound Connection Pool	Server	State	Current Connections	Created Connections
eis/OracleJCAAdapter/DefaultConnection	soa_server1	Running	0	0

「現在の接続」列は、実行中の SOA プロセスなどにより現在使用されている管理対象接続の数を表します。この数は、SOA プロセスが実行を開始すると増加し、一定時間非アクティブの状態が続くと減少します。

SOA プロセスは、管理対象接続が必要になると、接続プールから管理対象接続を取得しようとします。使用可能な管理対象接続がない場合、現在の接続数が最大プール・サイズを超えていなければ、新しい接続が作成されます。最大プール・サイズに達していて、使用可能な空きの接続がない場合は、リソース可用性エラーが生成されます。プロセスが終了すると、接続がプールに返されます。

これらの数は、特定の EIS への接続ではなく Oracle リソース・アダプタへの接続を表していることに注意してください。1 つの Oracle 管理対象接続には、その使用履歴に応じた数の EIS 接続が含まれている場合があります。したがって、Oracle リソース・アダプタによって使用されている EIS 接続の数を調べる方法はありません。

管理対象接続プールの最大サイズ、最小サイズおよびその他のプール・パラメータは、`weblogic-ra.xml` ファイル内で設定でき、[図 1-2](#) に示すように管理コンソールに表示されます。

図 1-2 「接続プール」タブ

The screenshot shows the 'Settings for javax.resource.cci.ConnectionFactory' page in the Oracle WebLogic Server Administration Console. The 'Connection Pool' tab is active. The 'Save' button is visible at the top left. Below the tabs, there is a description: 'This page allows you to view and modify the pool parameters of this outbound connection.' The parameters are as follows:

Parameter	Value
Initial Capacity:	0
Max Capacity:	10
Capacity Increment:	1
Shrinking Enabled:	true
Shrink Frequency Seconds:	900

多数のプール・パラメータがありますが、特定の EIS ではなく Oracle リソース・アダプタへの接続のプールを制御するものなので、その多くは重要ではありません。たとえば、初期プール・サイズを設定すると、空のリソース・アダプタ接続のセットが作成されるのみです。これを行う理由はありません。同様の理由で、「接続の最大追加数」パラメータも特に有用ではありません。

Oracle アダプタを使用する場合は、初期容量を常に 0 に設定する必要があります。

設定の中には、すべてのアダプタ・ターゲットに対して暗黙的な影響を与えるものもあります。たとえば、各管理対象接続では固有のアダプタ・ターゲットに対して最大 1 つの接続しか保持できないため、管理対象接続プールの最大サイズを設定すると、1 つのターゲットに対して確立できる最大接続数が暗黙的に設定されます。同様に、管理対象接続の存続期間を定義すると、その接続により作成されるすべての EIS 接続の存続期間が暗黙的に定義されます。ただし、Oracle の汎用的なリソース・アダプタを使用して特定の EIS ターゲットに対する接続プーリングを制御することはできません。

1.2 SOA、BPM および OSB の高可用性クラスタ環境での Oracle Application Adapters の構成

アプリケーション・アダプタは、Oracle WebLogic Server を各種のパッケージ・アプリケーションと統合します。このようなアダプタには、Oracle Application Adapters for PeopleSoft、SAP R/3、Siebel、J.D. Edwards OneWorld などがあります。

この項では、クラスタ環境 (のすべての使用可能な SOA または OSB ノード、この例では 2 つのシステム) に 12c (12.2.1.0.0) の Oracle Application Adapters をデプロイ

し、インバウンド・プロセスおよびアウトバウンド・プロセスを使用するために必要な手順について説明します。

注意： クラスタ環境内のすべてのノードで同じデータベースを使用します。このため、データベース・リポジトリを特定のデータベース用に構成する必要があり、すべてのノードでこのデータベースを指定する必要があります。BSE および J2CA のデータベース・リポジトリ設定を構成する方法の詳細は、『Oracle Fusion Middleware Oracle WebLogic Server Application Adapters インストール・ガイド』(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。

この節では、次のトピックについて説明します。

- 1.2.1 項「2つのシステムを含むクラスタ環境への SOA または OSB のインストールと構成」
- 1.2.2 項「Oracle Application Adapters のデプロイ」
- 1.2.3 項「iBSE コネクタ・アプリケーションのデプロイと構成」
- 1.2.4 項「主な注意事項」
- 1.2.5 項「アウトバウンド・プロセスのための高可用性 (HA) ファイル・アダプタの構成」
- 1.2.6 項「アウトバウンド・プロセスの構成」
- 1.2.7 項「インバウンド・プロセス用の Oracle HTTP Server の構成」
- 1.2.8 項「インバウンド・プロセスの構成」
- 1.2.9 項「シングルトン・テスト」

1.2.1 2つのシステムを含むクラスタ環境への SOA または OSB のインストールと構成

2つのシステムを含むクラスタ環境に SOA または OSB をインストールして構成するには、次を行います。

1. 両方のシステムに SOA または OSB スイートをインストールします。SOA または OSB スイートのインストールの詳細は、次の Web サイトを参照してください。
 - SOA の場合：
<http://docs.oracle.com/middleware/1221/core/INSOA/GUID-D5AFD830-8A7D-42CC-8C22-CE68C452CF4A.htm#INSOA369>
 - OSB の場合：
<http://docs.oracle.com/middleware/1221/core/INOSB/GUID-D5AFD830-8A7D-42CC-8C22-CE68C452CF4A.htm#INSOA369>
2. SOA または OSB のクラスタ・ドメインを作成して構成します。詳細は、次の Web サイトを参照してください。
 - SOA の場合：
<http://docs.oracle.com/middleware/1221/core/INSOA/GUID-36DFF16B-4891-46EB-9554-436A3CCF85BB.htm#INSOA377>

- OSB の場合：

<http://docs.oracle.com/middleware/1221/core/INOSB/GUID-36DFF16B-4891-46EB-9554-436A3CCF85BB.htm#INSOA377>

注意： 前述のリンクには単一システムのクラスタ環境を構成する手順が含まれています。2つのシステムでクラスタ環境を構成するように手順を変更してください(2番目の管理対象サーバー用に2番目のシステムのIPを指定する、1つのマシンではなく2つのマシンを作成する、2番目の管理対象サーバーで2番目のマシンを指定するなど)。

3. 最初のシステムでドメインをバックし、2番目のシステムでアンパックします。詳細は、次を参照してください。

http://docs.oracle.com/middleware/1221/core/ASHIA/scale_out_prototype.htm#ASHIA134

1.2.2 Oracle Application Adapters のデプロイ

J2CA コネクタ・アプリケーションをデプロイするには、次の手順に従います。

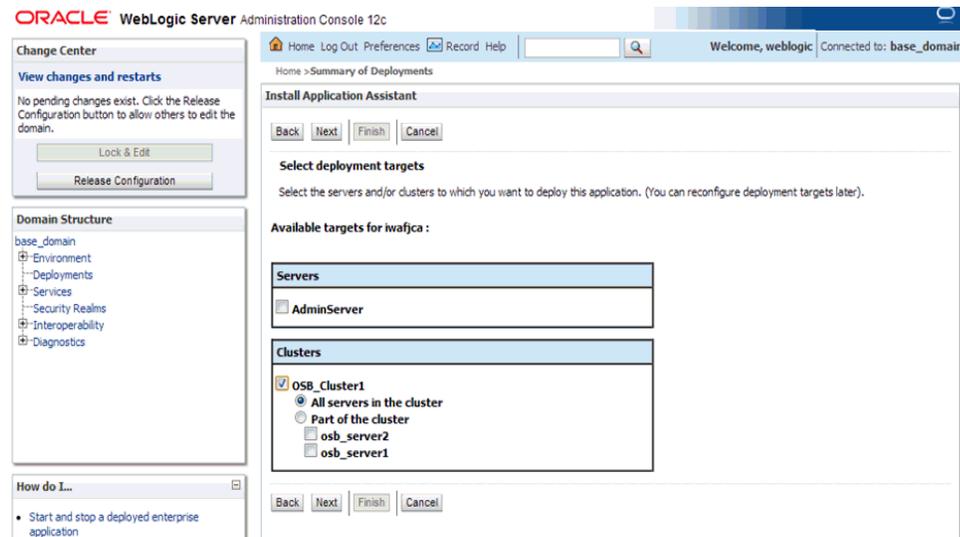
1. 構成されている Oracle WebLogic Server ドメインに対して Oracle WebLogic Server を起動します。
2. 両方のシステムで SOA または OSB の管理対象サーバーを起動します。
3. 次の URL を入力して、Web ブラウザで Oracle WebLogic Server 管理コンソールを開きます。

`http://host name:port/console`

説明は次のとおりです。

- `host name`: Oracle WebLogic Server を実行しているシステムの名前です。
 - `port`: 使用しているドメインのポート番号です。
4. 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。
 5. 左ペインの「ドメイン構造」セクションにある「デプロイメント」をクリックします。
 6. 「チェンジ・センター」の「ロックして編集」をクリックします。
 7. 「インストール」をクリックします。
 8. アダプタのインストール・ディレクトリを参照し、`iwafjca.rar` の隣のオプションを選択して「次へ」をクリックします。
 9. デフォルトの選択(「このデプロイメントをアプリケーションとしてインストールする」)を受け入れて、「次へ」をクリックします。
 10.  1-3 に示すように、「クラスタ」セクションで、「SOA_Cluster」または「OSB_Cluster」を選択し、「クラスタのすべてのサーバー」を選択して「次へ」をクリックします。

図 1-3 「クラスタ」セクション



「オプション設定」ページが表示されます。

11. デフォルト値を受け入れ、「次へ」をクリックします。
「サマリー」ページが表示されます。
 12. 「終了」をクリックします。
J2CA (iwafjca) コネクタ・アプリケーションの「設定」ページが表示されます。
 13. 「保存」をクリックします。
 14. 「チェンジ・センター」の「変更のアクティブ化」をクリックします。
 15. 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、すべてのデプロイ済アプリケーションが一覧表示された表をナビゲートして、J2CA (iwafjca) コネクタ・アプリケーションを見つけます。
 16. 「制御」タブを選択し、iwafjca の隣のチェック・ボックスを選択します。
 17. 「起動」サブメニュー (下向き矢印) をクリックして「すべてのリクエストを処理」を選択します。
「アプリケーション起動アシスタント」ページが表示されます。
 18. 「はい」をクリックします。
 19. 「デプロイメント」ページでアプリケーションが正常に開始されたことを確認します。
 20. 同様に、手順 5 から 19 を iwafjca.war デプロイメントについて繰り返します。
- iBSE コネクタ・アプリケーションのデプロイと構成を行うには、次の手順に従います。

1.2.3 iBSE コネクタ・アプリケーションのデプロイと構成

この節では、次のトピックについて説明します。

- 1.2.3.1 項「iBSE 構成のデプロイ」
- 1.2.3.2 項「アプリケーション・エクスプローラで BSE 構成を構成する」

注意： 次の手順を続ける前に、クラスタ・ドメインがすでに作成および構成されていることを確認してください。作成済でない場合は、これらの手順を進める前に、詳細を 1-4 ページ 1.2.1 項「[2つのシステムを含むクラスタ環境への SOA または OSB のインストールと構成](#)」で確認してクラスタ・ドメインを作成してください。

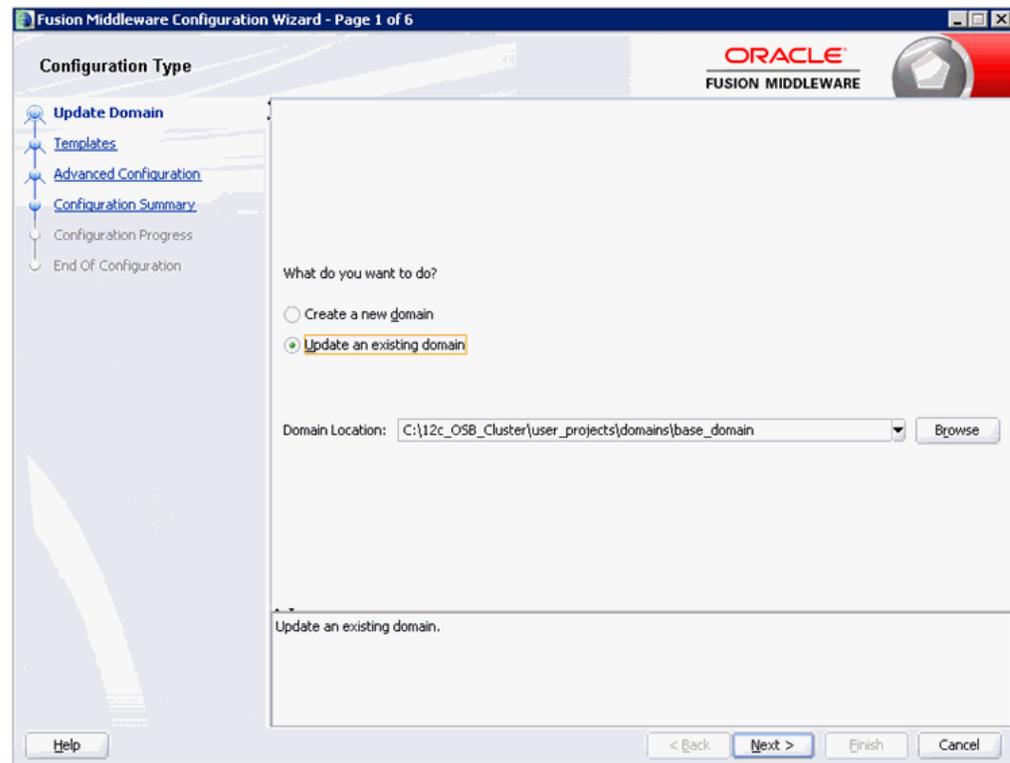
iBSE 用の HTTP プロキシ・サーバーでドメインを拡張するには、次の手順に従います。

1. 次のディレクトリに移動して config.cmd を実行します。

```
<ORACLE_HOME>\oracle_common\common\bin
```

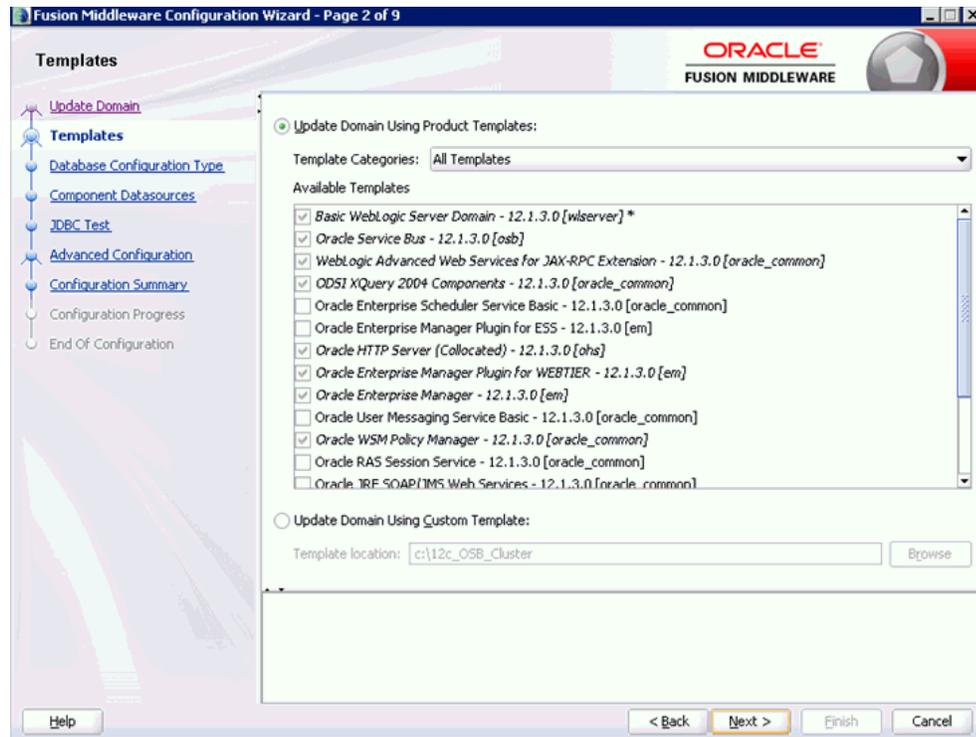
2. [図 1-4](#) に示すように、表示された構成ウィザードで「既存のドメインの更新」を選択し、作成したクラスタ・ドメインの場所を参照して「次へ」をクリックします。

図 1-4 「構成タイプ」 ペイン



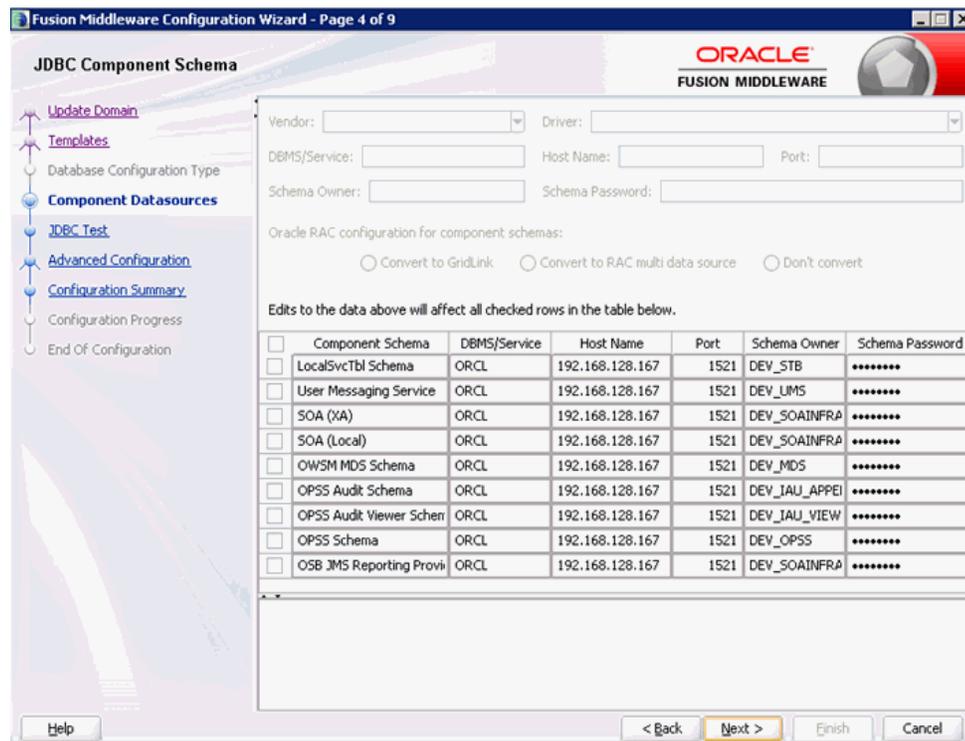
3. [図 1-5](#) に示すように、表示された「テンプレート」 ペインでデフォルト値をそのままにして「次へ」をクリックします。

図 1-5 「テンプレート」 ペイン



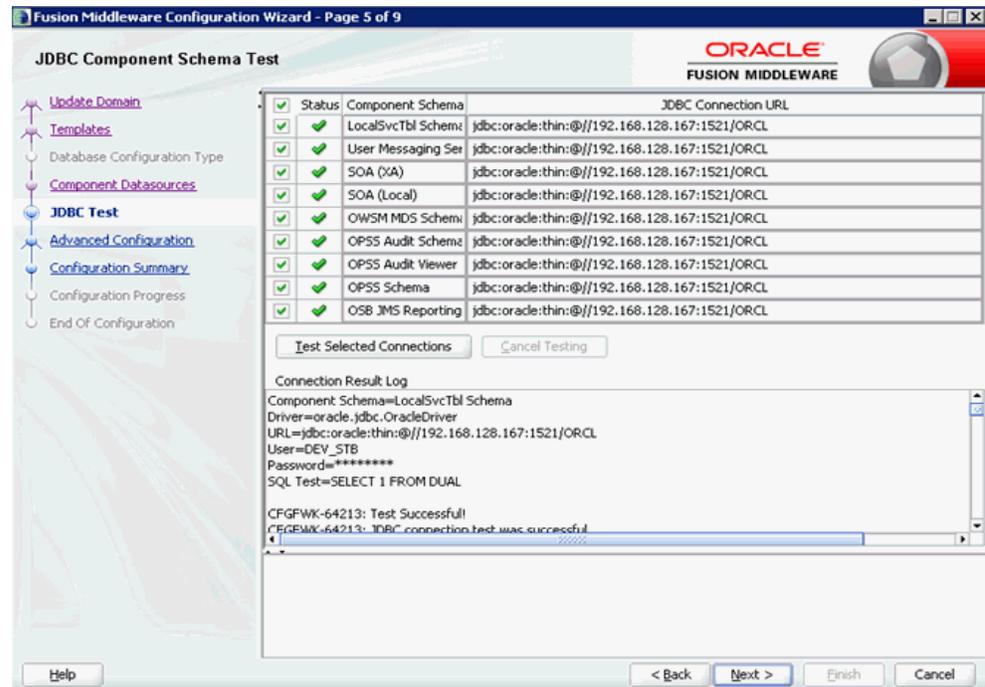
4. 図 1-6 に示すように、表示された「JDBC コンポーネント・スキーマ」 ペインでデフォルト値をそのままにして「次へ」をクリックします。

図 1-6 「JDBC コンポーネント・スキーマ」 ペイン



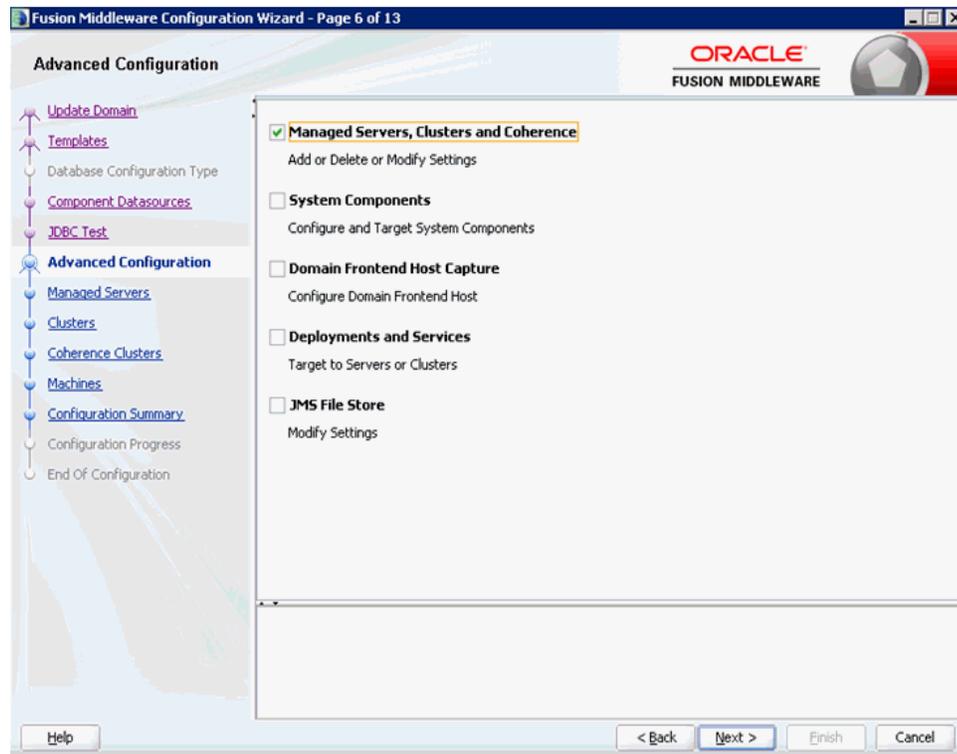
5. 図 1-7 に示すように、「JDBC コンポーネント・スキーマ・テスト」ペインで、すべてのテストに合格したことを確認して「次へ」をクリックします。

図 1-7 「JDBC コンポーネント・スキーマ・テスト」ペイン



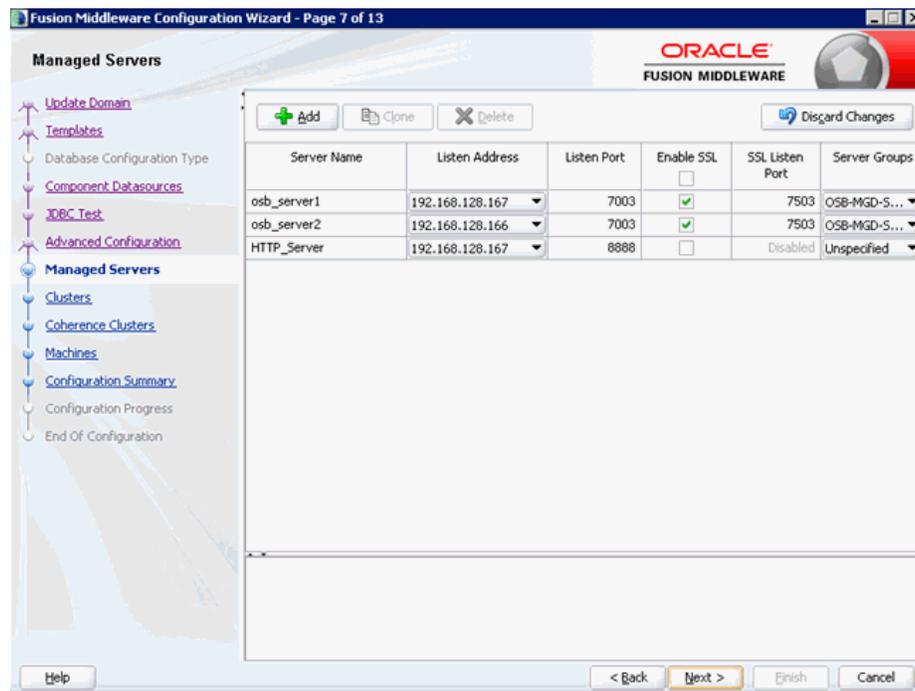
6. 図 1-8 に示すように、「管理対象サーバー、クラスタおよび Coherence」チェック・ボックスを選択して、「次へ」をクリックします。

図 1-8 「拡張構成」 ペイン



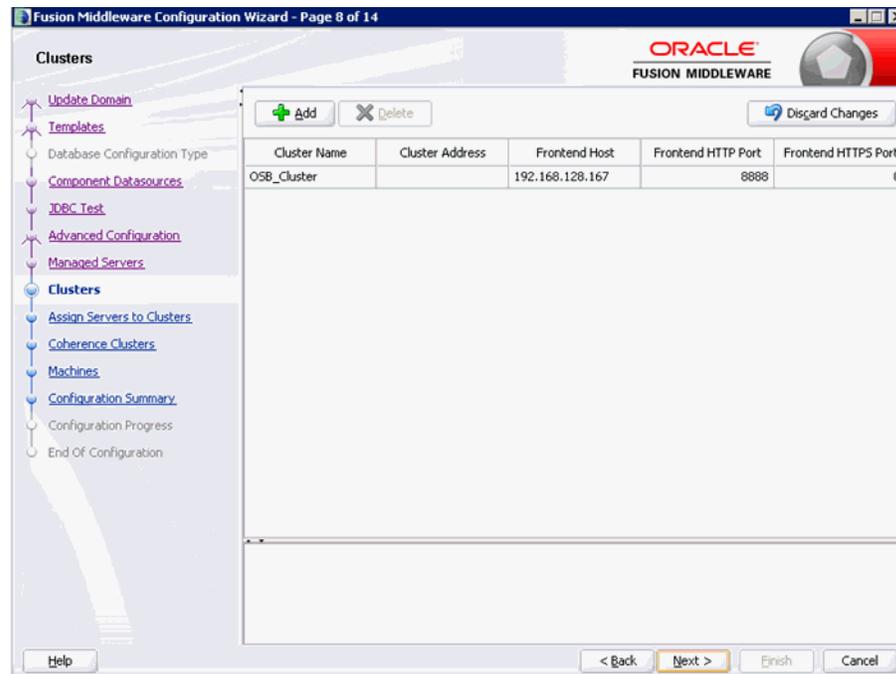
- 図 1-9 に示すように、「追加」をクリックして新しいサーバー (HTTP_Server など) を作成し、リスニング・ポート (8888 など) を入力します。

図 1-9 「管理対象サーバー」 ペイン



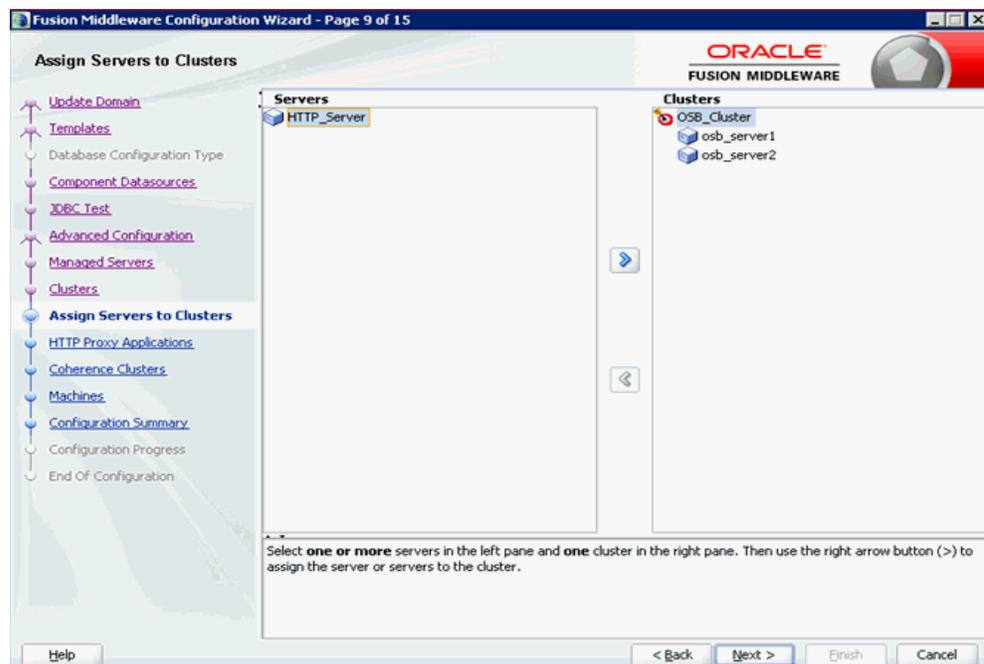
8. 図 1-10 に示すように、「クラスタ」ペインで使用可能なクラスタをそのままにして「次へ」をクリックします。

図 1-10 「クラスタ」ペイン



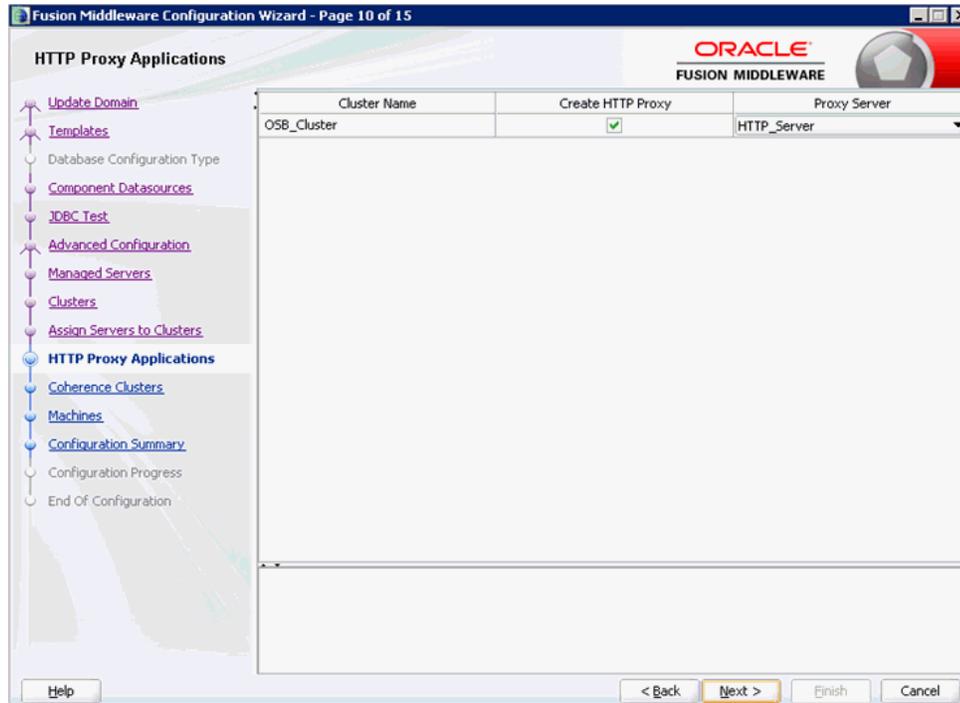
9. 図 1-11 に示すように、「クラスタへのサーバーの割当」ペインで、新たに作成した HTTP_Server がクラスタに割り当てられていないことを確認し、「次へ」をクリックします。

図 1-11 「クラスタへのサーバーの割当」ペイン



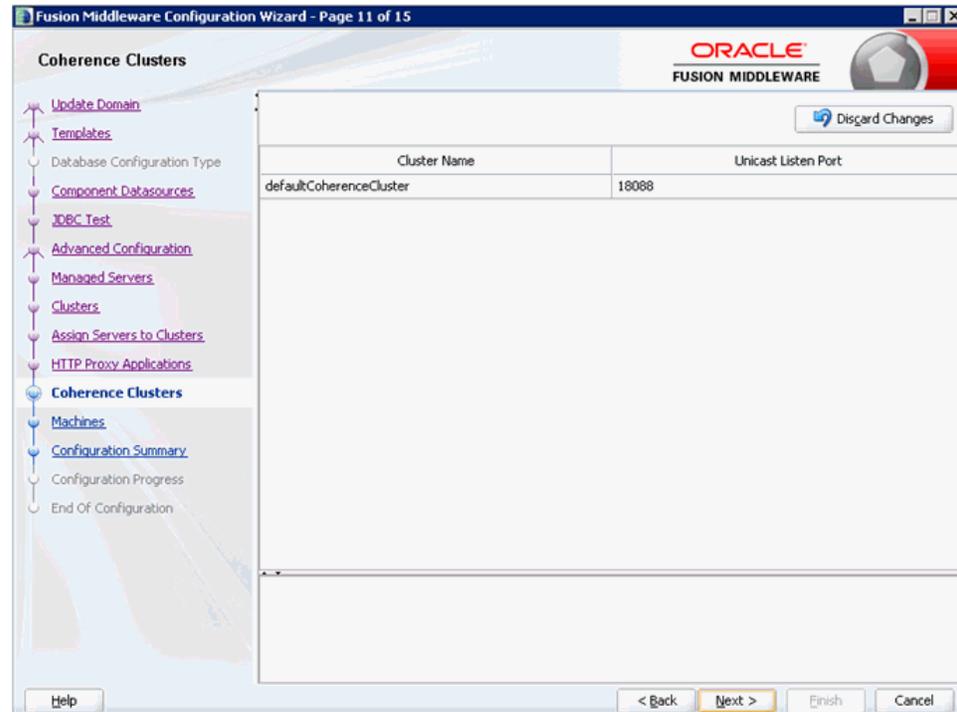
10. 図 1-12 に示すように、「HTTP プロキシ・アプリケーション」ペインで「HTTP プロキシの作成」チェック・ボックスを選択し、ドロップダウン・リストから「HTTP_Server」を選択して「次へ」をクリックします。

図 1-12 「HTTP プロキシ・アプリケーション」ペイン



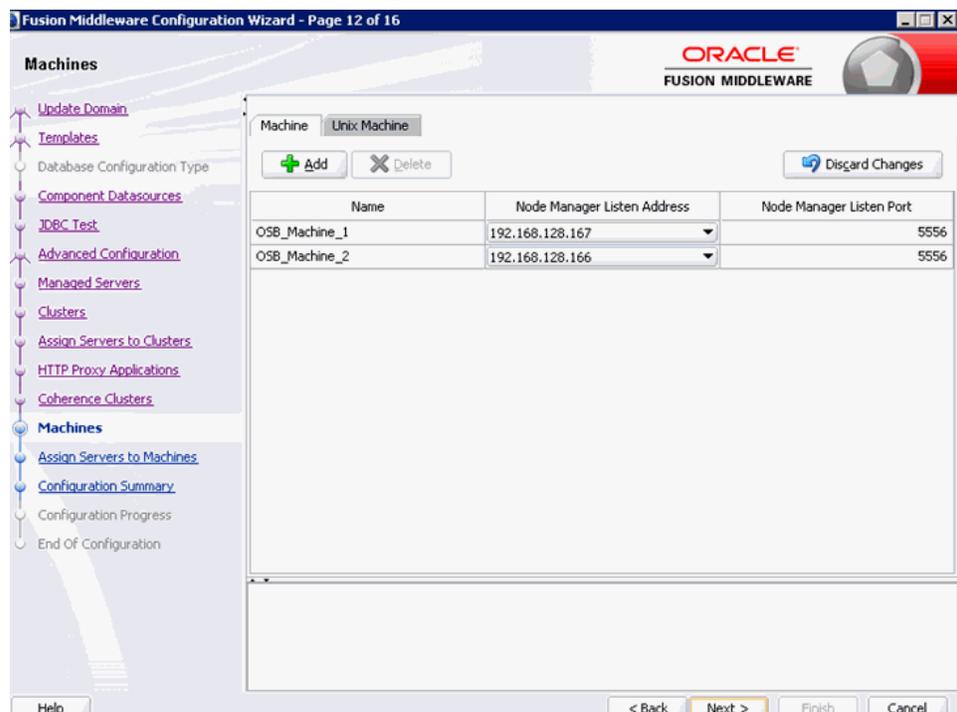
11. 図 1-13 に示すように、「Coherence クラスタ」ペインでデフォルト値をそのままにして「次へ」をクリックします。

図 1-13 「Coherence クラスタ」 ペイン



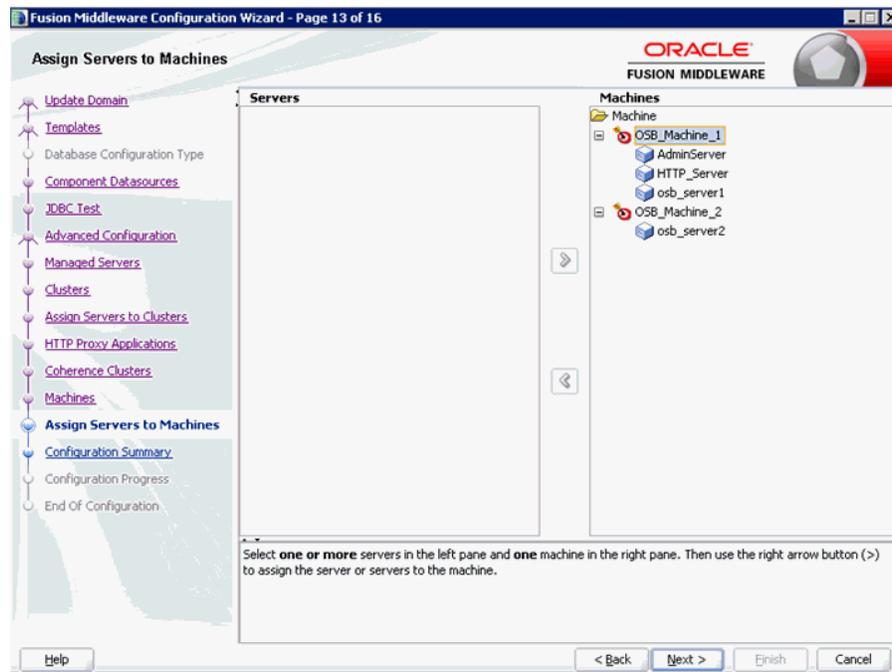
12. 図 1-14 に示すように、「マシン」 ペインで使用可能なマシンをそのままにして「次へ」をクリックします。

図 1-14 「マシン」 ペイン



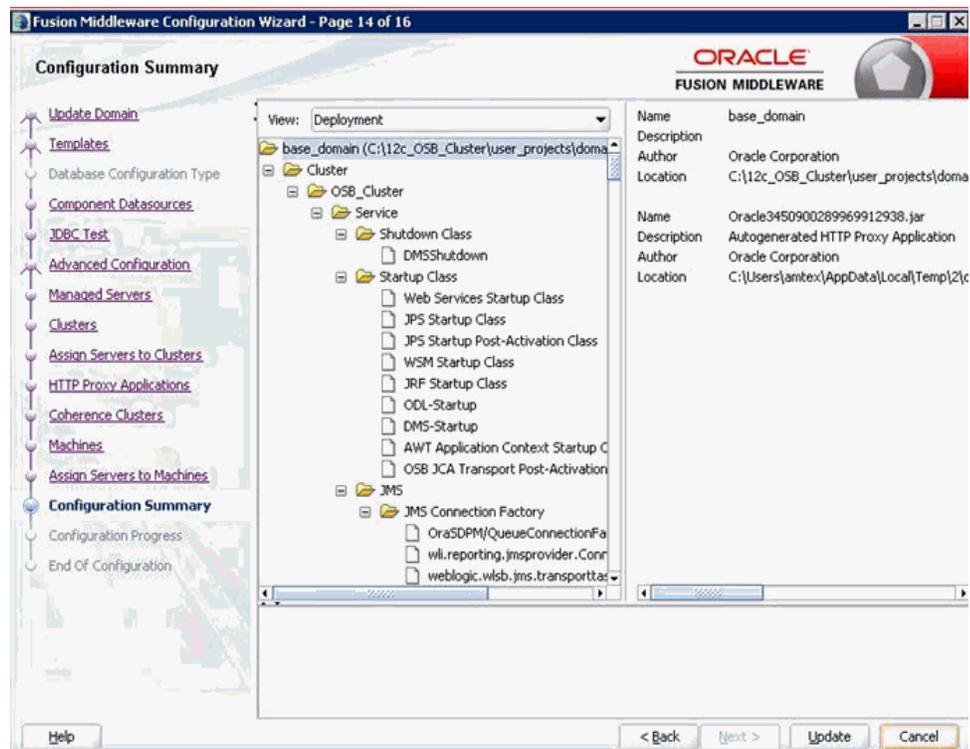
13. 図 1-15 に示すように、「サーバーのマシンへの割当」 ペインで、作成した HTTP サーバーを (管理サーバーが使用可能な) 最初のノードに割り当てて「次へ」をクリックします。

図 1-15 「サーバーのマシンへの割当」 ペイン



14. 図 1-16 に示すように、「構成のサマリー」 ペインでサマリーを確認して「更新」をクリックします。

図 1-16 「構成のサマリー」 ペイン



1.2.3.1 iBSE 構成のデプロイ

ibse.war をデプロイするには、1-5 ページ 1.2.2 項「Oracle Application Adapters のデプロイ」の J2CA のデプロイメント手順の手順 5 から 19 を、次の変更を行ったうえで繰り返します。

- 手順 8 で ibse.war を選択します。
- 手順 10 で作成した HTTP_Server を選択します(その他のサーバーは選択しません)。

1.2.3.2 アプリケーション・エクスプローラで BSE 構成を構成する

次の手順を実行してアプリケーション・エクスプローラで BSE 構成を構成します。

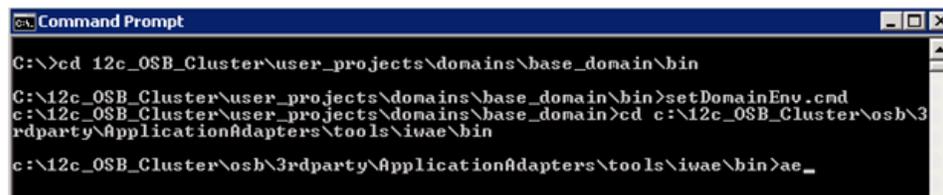
1. ターミナルを開いて次のディレクトリに移動し、setDomainEnv.sh (UNIX プラットフォームの場合) または setDomainEnv.cmd (Windows プラットフォームの場合) を実行します。

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

2. 図 1-17 に示すように、同じターミナルで次のディレクトリに移動して ae.exe (Windows プラットフォームの場合) または iwae.sh (UNIX プラットフォームの場合) を実行します。

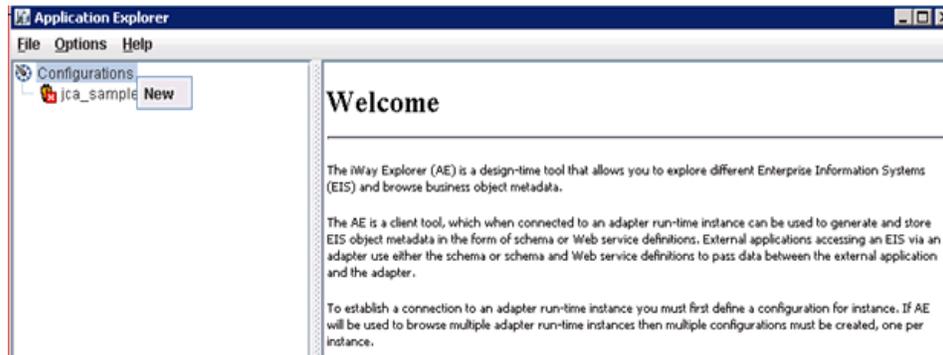
```
<ADAPTER_HOME>\tools\iwae\bin
```

図 1-17 コマンド・プロンプト



3. 図 1-18 に示すように、「構成」を右クリックし、「新規」を選択します。

図 1-18 アプリケーション・エクスプローラ



「新規構成」ダイアログが表示されます。

4. 図 1-19 に示すように、新しい構成の名前 (例: ibse) を入力して「OK」をクリックします。

図 1-19 「新規構成」ダイアログ



注意： ここで指定した BSE 構成の名前は BSE デプロイメント・プロセスで使用されます。

5. 「サービス・プロバイダ」リストから「iBSE」を選択します。
6. 図 1-20 に示すように、「iBSE URL」フィールドで、デフォルトの URL を受け入れるか、次の形式で別の URL に置き換えて「OK」をクリックします。

図 1-20 「新規構成」ダイアログ



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

説明は次のとおりです。

- *host name*: HTTP サーバーが存在するシステムです。
- *port*: HTTP サーバーがリスニングしている HTTP ポート番号です。

図 1-21 に示すように、ルート「構成」ノードの下に新規構成を表すノードが表示されます。

図 1-21 新規構成



注意：

1. 次の iBSE 構成ページを使用してデータベースの iBSE を構成します。
`http:// HTTP_PROXY_SERVER_IP:HTTP_PROXY_Server_PORT /ibse`
 詳細は、*Oracle WebLogic Server インストレーション・ガイド*を参照してください。
2. サーバーを再起動します。
3. HTTP プロキシ・サーバーの IP とポートを使用して ibse にアクセスします。次に例を示します。
`http://HTTP_PROXY_SERVER_IP:HTTP_PROXY_Server_PORT/ibse/IBSEServlet`

1.2.4 主な注意事項

アウトバウンド・プロセスの構成はすべての Oracle Application Adapters for Oracle WebLogic Server (SAP R/3、Siebel、PeopleSoft および J.D. Edwards) で同じです。

詳細は、1-17 ページ [1.2.5 項「アウトバウンド・プロセスのための高可用性 \(HA\) ファイル・アダプタの構成」](#) および 1-22 ページ [1.2.6 項「アウトバウンド・プロセスの構成」](#) を参照してください。

インバウンド処理について、1-23 ページ [1.2.7 項「インバウンド・プロセス用の Oracle HTTP Server の構成」](#) に記載されている手順は Siebel、PeopleSoft および J.D. Edwards アダプタに対してのみ実施する必要があります。SAP R/3 アダプタに対しては実施しません。PeopleSoft および J.D. E アダプタについては、HTTP リスナーのみを高可用性 (HA) 環境で使用する必要があります。TCP リスナーは使用しないでください。

1.2.5 アウトバウンド・プロセスのための高可用性 (HA) ファイル・アダプタの構成

アウトバウンド・プロセスを使用するには、Oracle WebLogic Server 管理コンソールで次の構成手順を実行する必要があります。

1. 次の URL を入力して、Web ブラウザで Oracle WebLogic Server 管理コンソールを開きます。

`http://host name:port/console`

ここで、*host name* は Oracle WebLogic Server が稼働しているシステムの名前、*port* は使用しているドメインのポート番号です。

Oracle WebLogic Server 管理コンソールのログオン・ページが表示されます。

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。

Oracle WebLogic Server 管理コンソールの「ホーム」ページが表示されます。

3. 左ペインの「ドメイン構造」セクションにある「デプロイメント」をクリックします。

[図 1-22](#) に示すように、「デプロイメント」ページが表示されます。

図 1-22 「デプロイメント」 ページ

Name	State	Health	Type	Deployment Order
adf.oracle.domain(1.0,11.1.1.2.0)	Active		Library	100
adf.oracle.domain.webapp(1.0,11.1.1.2.0)	Active		Library	100
AqAdapter	Active	OK	Resource Adapter	324
b2bui	Active	OK	Enterprise Application	313
composer	Active	OK	Enterprise Application	315
DbAdapter	Active	OK	Resource Adapter	322
DefaultToDoTaskFlow	Active	OK	Enterprise Application	314
DMS Application (11.1.1.1.0)	Active	OK	Web Application	5
em	Active	OK	Enterprise Application	400
emai	Active		Library	100
emas	Active		Library	100
emcore	Active		Library	100
FileAdapter	Active	OK	Resource Adapter	321
FMW Welcome Page Application (11.1.0.0.0)	Active	OK	Enterprise Application	5
FtpAdapter	Active	OK	Resource Adapter	325

4. 「FileAdapter」 をクリックします。

図 1-23 に示すように、「FileAdapter の設定」 ページが表示されます。

図 1-23 「FileAdapter の設定」 ページ

Settings for FileAdapter

Overview Deployment Plan **Configuration** Security Targets Control Testing Monitoring Notes

General Properties **Outbound Connection Pools** Admin Objects Workload Instrumentation

This page displays a table of Outbound Connection Pool groups and instances for this resource adapter. The top level entries in the table represent Outbound Connection Pool groups. Groups are listed by connection factory interface and the instances are listed by their JNDI names. Expand a group to obtain configuration information for a Connection Pool instance within an Outbound Connection Pool group. Click the name of a group or instance to configure it. Automatically generated Connection Pools are not displayed in the table below.

Outbound Connection Pool Configuration Table

Click the **Lock & Edit** button in the Change Center to activate all the buttons on this page.

New Delete Showing 1 to 1 of 1 Previous | Next

Groups and Instances	Connection Factory Interface
javax.resource.cci.ConnectionFactory	javax.resource.cci.ConnectionFactory

New Delete Showing 1 to 1 of 1 Previous | Next

5. 「構成」 タブ、「アウトバウンド接続プール」 タブの順にクリックします。

図 1-24 に示すように、「アウトバウンド接続プールの構成表」 ページが表示されます。

図 1-24 「アウトバウンド接続プールの構成表」ページ

Outbound Connection Pool Configuration Table

Click the *Lock & Edit* button in the Change Center to activate all the buttons on this page.

New Delete Showing 1 to 1 of 1 Previous | Next

<input type="checkbox"/>	Groups and Instances	Connection Factory Interface
<input type="checkbox"/>	javax.resource.cci.ConnectionFactory	javax.resource.cci.ConnectionFactory
<input type="checkbox"/>	eis/FileAdapter	javax.resource.cci.ConnectionFactory
<input type="checkbox"/>	eis/HAFileAdapter	javax.resource.cci.ConnectionFactory
<input type="checkbox"/>	eis/HAFileAdapterDB2	javax.resource.cci.ConnectionFactory
<input type="checkbox"/>	eis/HAFileAdapterMSSQL	javax.resource.cci.ConnectionFactory

New Delete Showing 1 to 1 of 1 Previous | Next

6. 「javax.resource.cci.ConnectionFactory」を展開し、「eis/HAFileAdapter」をクリックします。

図 1-25 に示すように、「javax.resource.cci.ConnectionFactory の設定」ページが表示されます。

図 1-25 「javax.resource.cci.ConnectionFactory」ページ

Settings for javax.resource.cci.ConnectionFactory

General **Properties** Transaction Authentication Connection Pool Logging

This page allows you to view and modify the configuration properties of this outbound connection pool. Properties you modify here are saved to a deployment plan.

Outbound Connection Properties

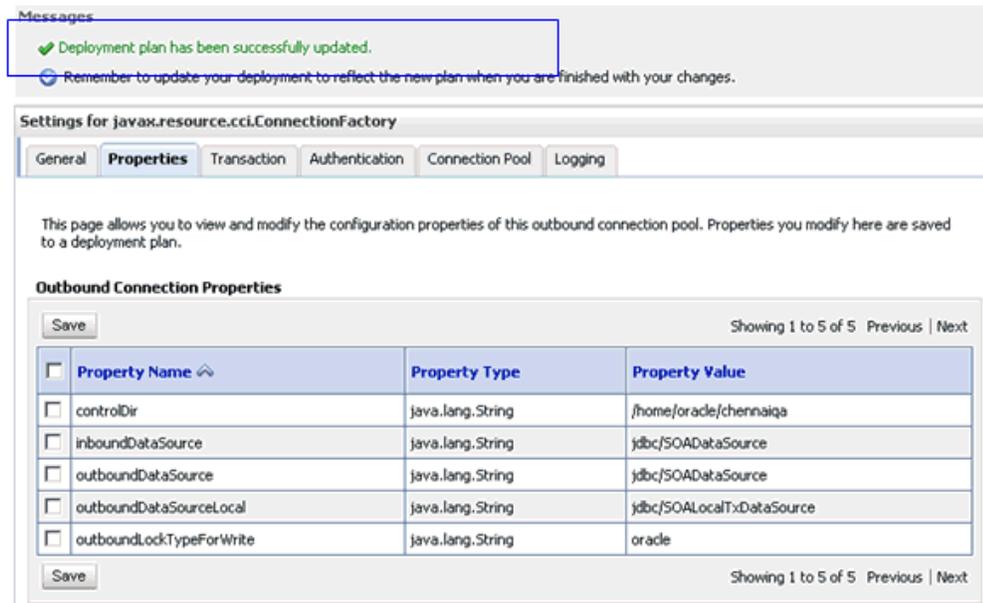
Save Showing 1 to 5 of 5 Previous | Next

<input type="checkbox"/>	Property Name	Property Type	Property Value
<input type="checkbox"/>	controlDir	java.lang.String	/home/oracle/chennaiqa
<input type="checkbox"/>	inboundDataSource	java.lang.String	jdbc/SOADDataSource
<input type="checkbox"/>	outboundDataSource	java.lang.String	jdbc/SOADDataSource
<input type="checkbox"/>	outboundDataSourceLocal	java.lang.String	jdbc/SOALocalTxDataSource
<input type="checkbox"/>	outboundLockTypeForWrite	java.lang.String	oracle

Save Showing 1 to 5 of 5 Previous | Next

7. 「プロパティ」タブをクリックします。
8. 「controlDir」プロパティに有効な場所を入力して「保存」をクリックします。
9. 下の図 1-26 に示すような成功メッセージが表示されているかどうかを確認します。

図 1-26 デプロイメント・プランの成功メッセージ



10. 図 1-27 に示すように、左ペインの「ドメイン構造」セクションにある「デプロイメント」をクリックします。

図 1-27 「デプロイメント」ページ



11. デプロイメント・リストから「FileAdapter」を選択して「更新」をクリックします。

図 1-28 に示すように、「アプリケーション更新アシスタント」ページが表示されます。

図 1-28 「アプリケーション更新アシスタント」 ページ

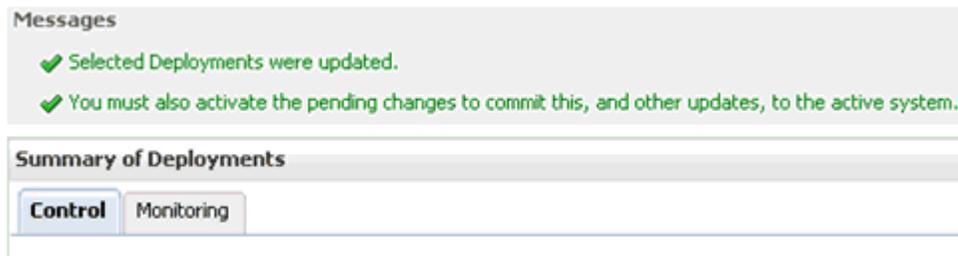
12. 「このアプリケーションを新しいデプロイメント・プランの変更とあわせた場所に更新します」を選択して「次へ」をクリックします。

図 1-29 に示すように、「サマリー」ページが表示されます。

図 1-29 「サマリー」 ページ

13. 「終了」をクリックします。
14. 図 1-30 に示すような成功メッセージが表示されているかどうかを確認します。

図 1-30 デプロイメントの成功メッセージ



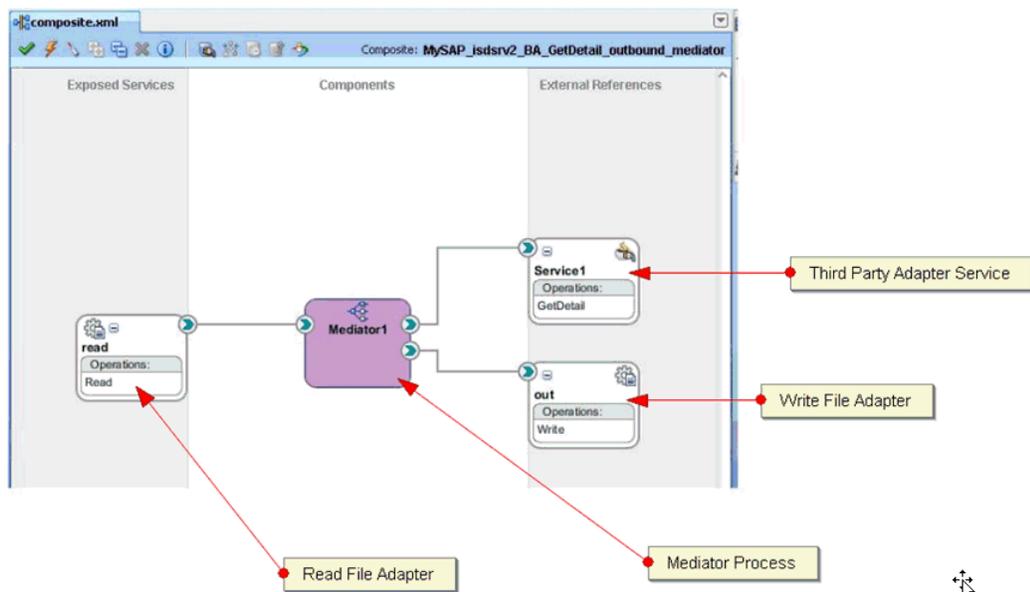
15. 生成された **plan.xml** ファイルを最初のシステム (クラスタ構成内の最初のノード) から 2 番目のシステム (クラスタ構成内の 2 番目のノード) の同じ場所 (Oracle_Home\Middleware\Oracle_SOA1\soa) にコピーします。
16. サーバー (Admin_Server、SOA_Server1、SOA_Server2) を再起動します。
17. 両方のシステムからアクセス可能な共有入力場所を作成します。
たとえば、入力をこの場所に貼り付けると、system1 および system2 から入力を使用できます。

1.2.6 アウトバウンド・プロセスの構成

アウトバウンド・プロセスを構成するには、次のようにします。

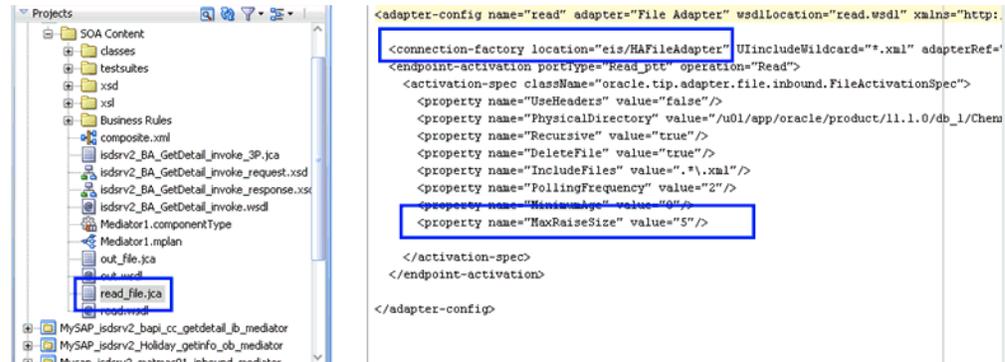
1. 最初のシステムでアプリケーション・エクスプローラを使用してターゲットを作成します。
2. (最初のシステム上の) ターゲットに接続し、Oracle Application Adapter for SAP R/3(MySAP ノード) のアウトバウンド WSDL を作成します。
3. 2 番目のシステム上でアプリケーション・エクスプローラを使用して、最初のシステムで指定したのと同じ名前のターゲットを作成します。
4. 両方のシステムで、管理サーバーおよび soa_server を再起動します。
5. 下の図 1-31 に示すように、Oracle JDeveloper を使用して、次のような構成でアウトバウンド・メディエータ・プロセスを作成します。

図 1-31 Oracle JDeveloper のアウトバウンド・メディエータ・プロセス



6. **Read_file_adapter.jca** ファイルを選択して次の変更を行います。
 - a. `<connection-factory location="eis/FileAdapter" UIincludeWildcard="*.xml" adapterRef="" />` を `<connection-factory location="eis/HFileAdapter" UIincludeWildcard="*.xml" adapterRef="" />` に変更します。
 - b. 図 1-32 に示すように、最後に値 `<property name="MaxRaiseSize" value="5" />` を追加します。

図 1-32 JCA プロパティのソース



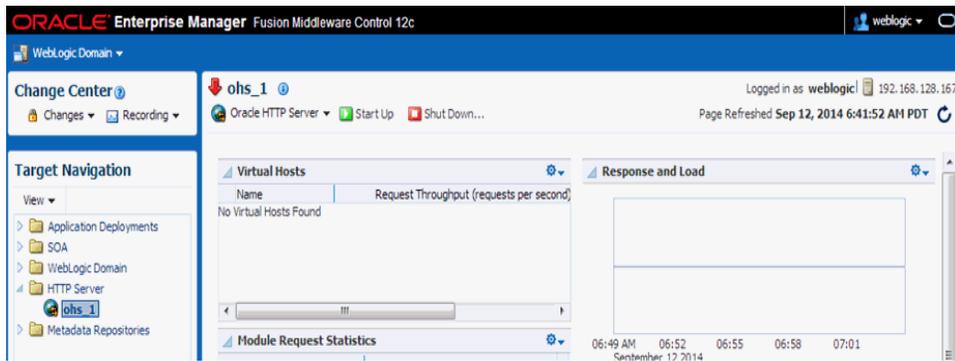
7. プロセスを保存し、両方の SOA サーバーにプロセスをデプロイします。
8. ファイル読取りアダプタで構成されている入力場所に入力ファイルを指定し、ファイル書込みアダプタで構成されている (2つのシステムの) 出力場所で出力が共有および配置されているかどうかを確認します。

1.2.7 インバウンド・プロセス用の Oracle HTTP Server の構成

インバウンド・プロセスを実行する前に、(PeopleSoft、Siebel および J.D. Edwards OneWorld Application Adapters に対して) 次の構成手順を実行する必要があります。

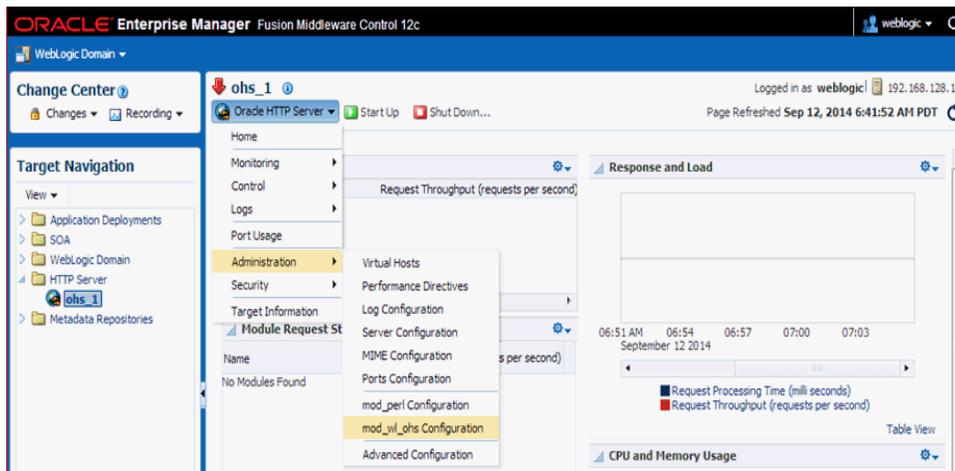
1. Web ブラウザで次の URL を入力して、Oracle EM コンソールを開きます。
`http://host name:port/em`
 説明は次のとおりです。
 - `host name`: Oracle WebLogic Server を実行しているシステムの名前です。
 - `port`: 使用しているドメインのポート番号です。
2. 管理者権限を持つアカウントを使用して、Oracle EM コンソールにログインします。
3. 図 1-33 に示すように、左ペインの「HTTP サーバー」を展開して、作成した OHS インスタンスを選択します。

図 1-33 OHS インスタンス



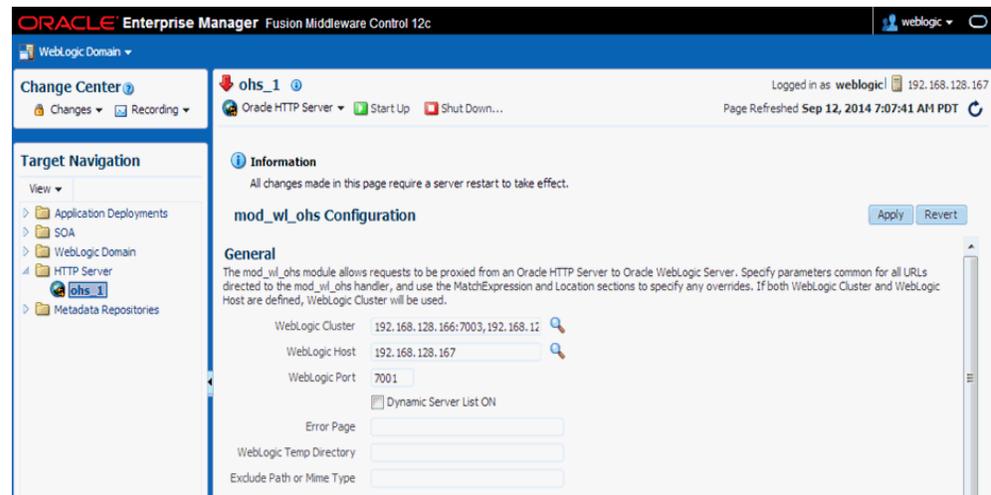
4. 図 1-34 に示すように、右ペインの「Oracle HTTP Server」ドロップダウン・リストをクリックして「管理」を選択し、「mod_wl_ohs 構成」をクリックします。

図 1-34 OHS の構成



5. 「チェンジ・センター」の「作成」をクリックし、「ロックして編集」を選択します。
6. 図 1-35 に示すように、画面上部の「一般」構成の下で「参照」ボタンを使用して、作成した WebLogic クラスタと WebLogic ホストを選択し、(まだ入力していない場合は) WebLogic ポートを入力します。

図 1-35 「一般」構成ペイン



7. 画面の下部にある「追加」をクリックして次のパラメータを入力します。

- 場所: Siebel (任意の適切な名前)
- WebLogic クラスタ: *system1_ip:port, system2_ip:port*

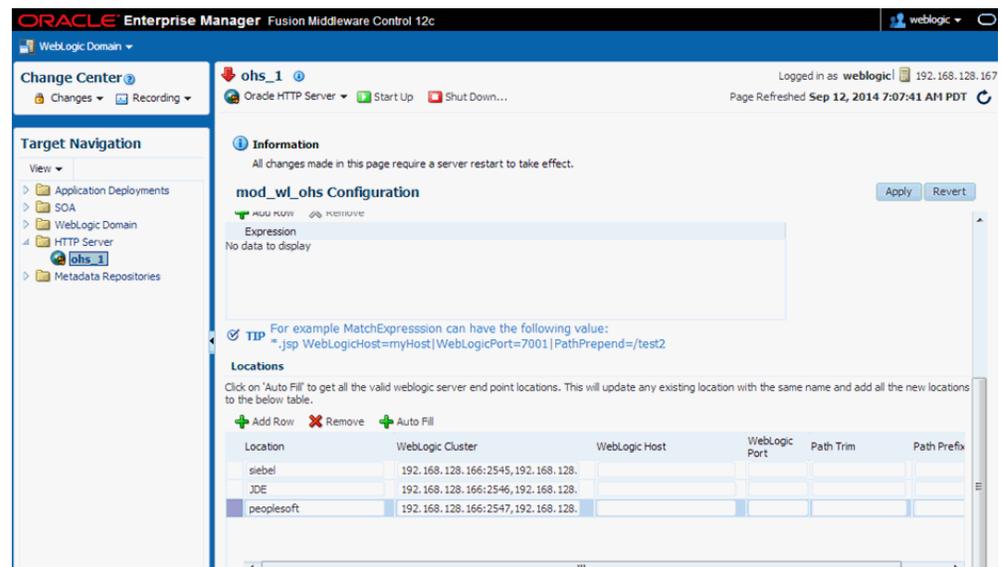
説明は次のとおりです。

- 場所: 任意の適切な名前です。
- *system1_ip* および *system2_ip*: SOA または OSB の管理対象サーバーが構成されているシステム IP アドレスです。
- *port*: *system1* および *system2* のチャネル構成で構成されているポート番号です。

8. 右ペインで「適用」をクリックし、「チェンジ・センター」の「変更」をクリックして「変更のアクティブ化」を選択します。

9. 図 1-36 に示すように、OHS サーバーを再起動します。

図 1-36 OHS サーバー

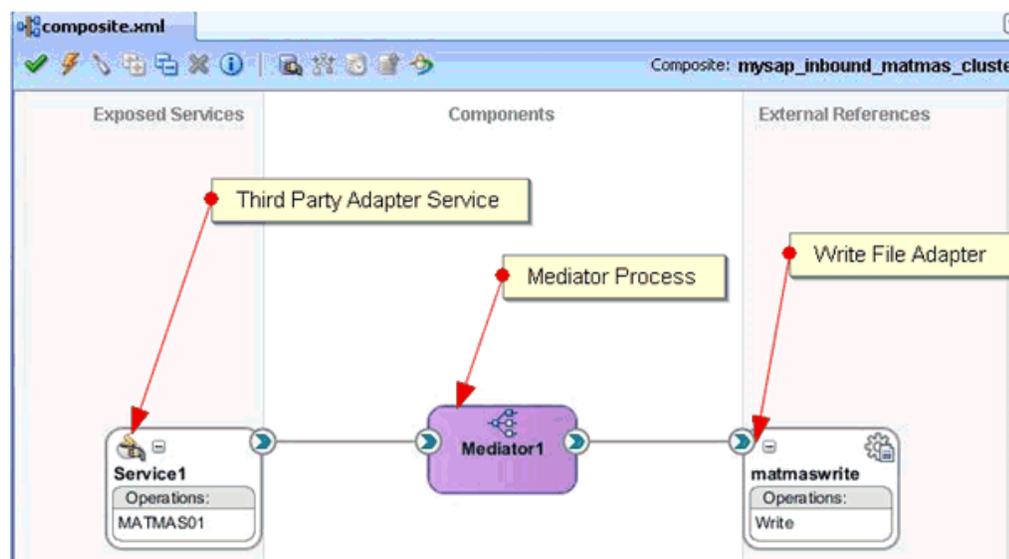


1.2.8 インバウンド・プロセスの構成

インバウンド・プロセスを構成するには、次のようにします。

1. 最初のシステムでアプリケーション・エクスプローラを使用してターゲットとチャンネルを作成します。
2. (最初のシステム上の) ターゲットに接続し、Oracle Application Adapter for SAP R/3(MySAP ノード) のインバウンド WSDL を作成します。
3. 2 番目のシステム上でアプリケーション・エクスプローラを使用して、最初のシステムで指定したのと同じ名前のターゲットとチャンネルを作成します。
4. 両方のシステムで、管理サーバーおよび soa_server を再起動します。
5. 図 1-37 に示すように、Oracle JDeveloper を使用して、次のような構成でインバウンド・メディエータ・プロセスを作成します。

図 1-37 Oracle JDeveloper のインバウンド・メディエータ・プロセス



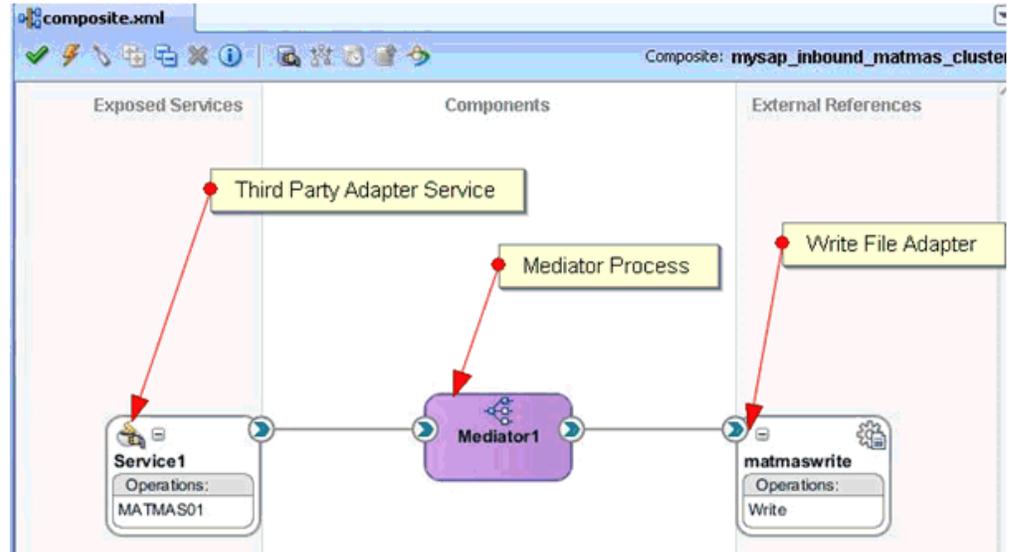
6. プロセスを保存し、(両方の SOA サーバーに) プロセスをデプロイします。
7. バックエンドからトリガーするか、HTTP パブリッシャを使用し、ファイル書込みアダプタで構成されている (両方のシステムの) 出力場所で出力が共有および配置されているかどうかを確認します。

1.2.9 シングルトン・テスト

シングルトン・テストを実行するには、次のようにします。

1. 最初のシステムでアプリケーション・エクスプローラを使用してターゲットとチャンネルを作成します。
2. (最初のシステム上の) ターゲットに接続し、Oracle Application Adapter for SAP R/3(MySAP ノード) のインバウンド WSDL を作成します。
3. 2 番目のシステム上でアプリケーション・エクスプローラを使用して、最初のシステムで指定したのと同じ名前のターゲットとチャンネルを作成します。
4. 両方のシステムで、管理サーバーおよび soa_server を再起動します。
5. 図 1-38 に示すように、Oracle JDeveloper を使用して、次のような構成でインバウンド・メディエータ・プロセスを作成します。

図 1-38 Oracle JDeveloper のインバウンド・メディエータ・プロセス



6. シングルトン・テストのために、**composite.xml** ファイルを開いて (ソース表示)、Service セクションに次のプロパティを追加します。

```
<property name="singleton">true</property>
```

次に例を示します。

```
<service name="Service1" ui:wSDLLocation="MATMAS01_receive_cluster.wsdl">
<interface.wSDL
interface="http://xmlns.oracle.com/pcbpel/iWay/wSDL/MySAP/isdsrv2_
cluster/MATMAS01#wSDL.interface (MATMAS01PortType) "/>
<binding.jca config="MATMAS01_receive_cluster_3P.jca"/>
<property name="singleton">true</property>
</service>
```

7. インバウンド・メディエータ・プロセスを保存し、両方の SOA サーバーにプロセスをデプロイします。
8. バックエンドからトリガーするか、HTTP パブリッシャを使用し、システムの出力量所でメッセージが受信されているかどうかを確認します。
9. メッセージを受信しているシステムの `soa_server` を停止します。
10. 2 番目のシステムでメッセージが受信されているかどうかを確認します。

Oracle Application Adapter for SAP R/3 でシングルトン・テスト機能を使用する場合、メッセージの損失があります (メッセージが SAP GUI にダンプされます)。これは、最初のシステムの停止後、2 番目のシステムのエンドポイント・アクティブ化が進行中になるためです。

失われたメッセージを取得するには、次の手順を実行します。

1. SAP GUI にログインします。
2. `/sm58` トランザクションを開始し、ダンプされたメッセージに移動します。
3. ダンプされたメッセージを右クリックし、**LUW の実行**を選択します。
4. ダンプされたメッセージすべてについて、ステップ 3 を繰り返します。

失われたすべてのメッセージが、Oracle Enterprise Manager コンソールで受信されます。

1.3 推奨されるリポジトリ・タイプ

ベスト・プラクティスとして、開発環境、テスト環境および本番環境で、アダプタに対してデータベース・リポジトリ (Oracle など) のみを使用することをお勧めします。初期起動の目的のみでデフォルトで提供されているファイル・リポジトリは使用しないでください。ファイル・リポジトリは、問題のトラブルシューティングを行う際にサポートされません。

1.4 入力 XML ドキュメントの生成

この項では、アウトバウンド BPEL およびメディアータ・プロセスでペイロードとして使用可能な入力 XML ドキュメントを生成する方法について説明します。ここでは、次の項目について説明します。

- 1.4.1 項「前提条件」
- 1.4.2 項「Application Explorer を使用した WSDL ドキュメントの作成」
- 1.4.3 項「リクエスト・スキーマからの入力 XML ファイルの生成」
- 1.4.4 項「Oracle Enterprise Manager コンソールでの生成済入力 XML ファイルの使用」

1.4.1 前提条件

続行する前に、次のコンポーネントおよびアプリケーションが使用可能であることを確認してください。

- Application Explorer を使用して作成されたアウトバウンド WSDL ドキュメント。
- XML エディタ (この項で例として使用されている Oracle JDeveloper や Altova XML Spy など)。
- Oracle WebLogic Server 12c (12.2.1.0.0) と Oracle JDeveloper。

1.4.2 Application Explorer を使用した WSDL ドキュメントの作成

アプリケーション・エクンプローラを使用して WSDL ドキュメントを作成するには、次のようにします。

1. Application Explorer がデプロイされた Oracle WebLogic Server が起動されていることを確認します。
2. コマンド・プロンプト・ウィンドウを開いて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。

setDomainEnv コマンドにより、Oracle WebLogic Server 環境のアプリケーション・エクンプローラのクラス・パスと他の環境変数が設定されます。
4. コマンド・プロンプト・ウィンドウを閉じないでください。
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・エクンプローラを起動します。
7. 使用可能な J2CA 構成を選択します。

8. 適切なアダプタ (MySAP など) を選択します。
9. 新規のターゲットを作成するか、既存のターゲットに接続します。
10. 作成したターゲット・アダプタを展開し、適切なオブジェクトを選択します。
11. オブジェクトを右クリックし、**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。
「WSDL のエクスポート」ダイアログが表示されます。
12. 「名前」フィールドのデフォルトの場所を受け入れて「OK」をクリックし、WSDL ドキュメントをデフォルトの場所にエクスポートします。
また、「参照」をクリックして別の場所を指定し、「OK」をクリックして、指定した場所に WSDL ドキュメントをエクスポートすることもできます。
13. WSDL ドキュメントがエクスポートされた場所に移動し、WSDL スキーマ、JCA スキーマ、リクエスト・スキーマおよびレスポンス・スキーマの各ファイルがエクスポートされて使用可能になっていることを確認します。

1.4.3 リクエスト・スキーマからの入力 XML ファイルの生成

リクエスト・スキーマから入力 XML ファイルを生成するには、次のようにします。

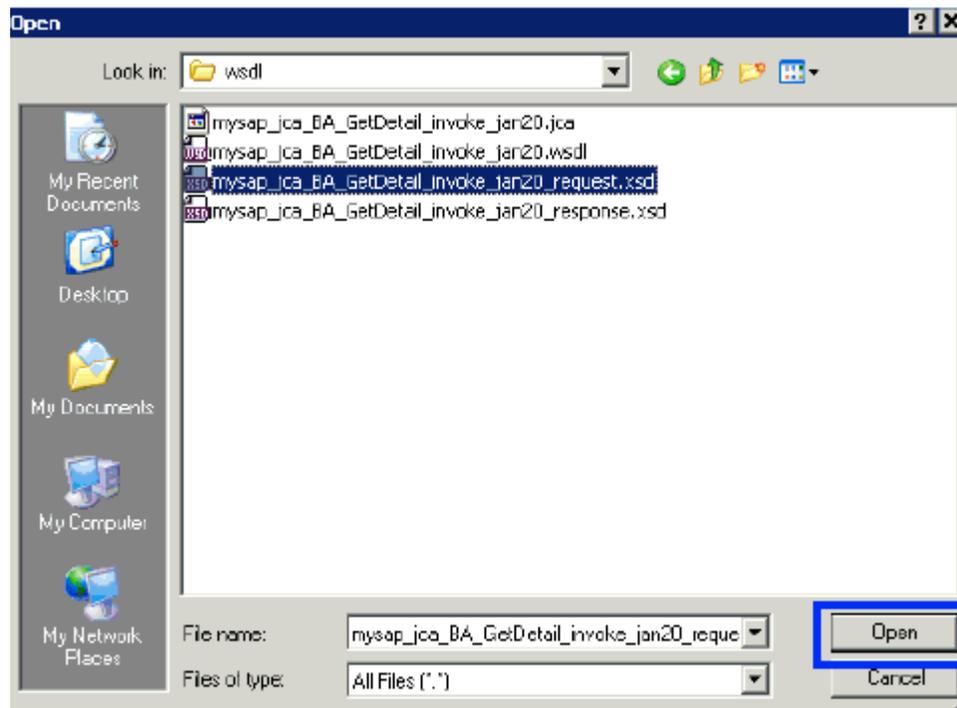
1. [図 1-39](#) に示すような XML エディタを開きます (この項で例として使用されている Altova XML Spy など)。

図 1-39 「Open」オプション



2. 「File」をクリックして「Open」を選択します。
[図 1-40](#) に示すように、「Open」ダイアログが表示されます。

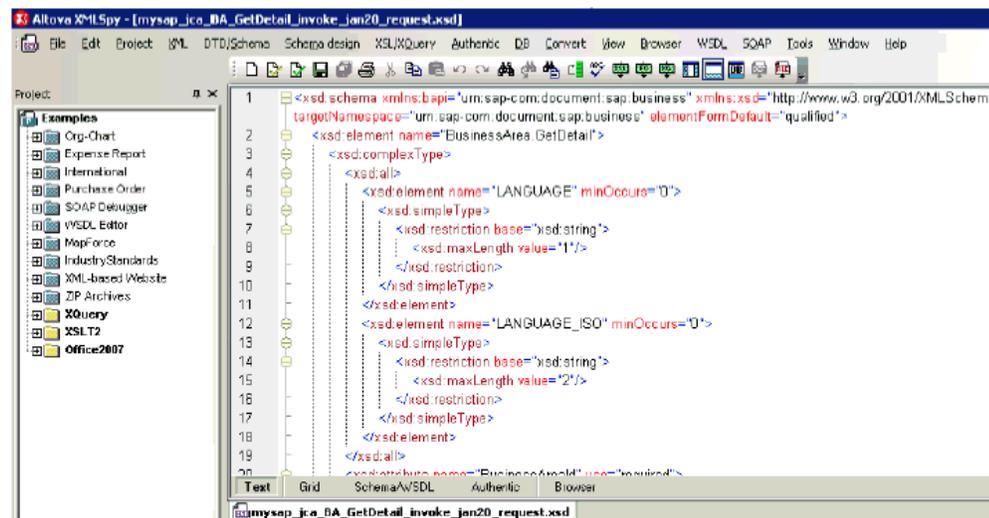
図 1-40 「Open」 ダイアログ



- XML リクエスト・スキーマがエクスポートされているファイル・システム上の場所に移動し、スキーマ・ファイルを選択して「Open」をクリックします。

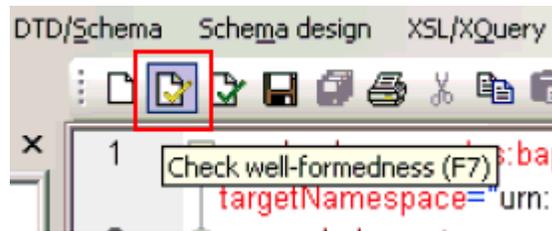
図 1-41 に示すように、Altova XML Spy で XML リクエスト・スキーマ・ファイルが開かれ、表示されます。

図 1-41 Altova XML Spy で表示された XML リクエスト・スキーマ・ファイル



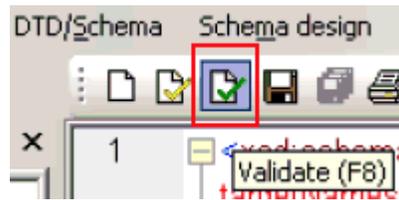
- 図 1-42 に示すように、「Check well-formedness」をクリックするか [F7] を押して、スキーマが整形形式であるかどうかを確認します。

図 1-42 「Check well-formedness」アイコン



5. 図 1-43 に示すように、「Validate」をクリックするか [F8] を押して、スキーマを検証します。

図 1-43 「Validate」アイコン



6. スキーマが整形形式で有効であることを確認したら、図 1-44 に示すように、「DTD/Schema」をクリックして「Generate Sample XML File」を選択します。

図 1-44 「Generate Sample XML File」オプション

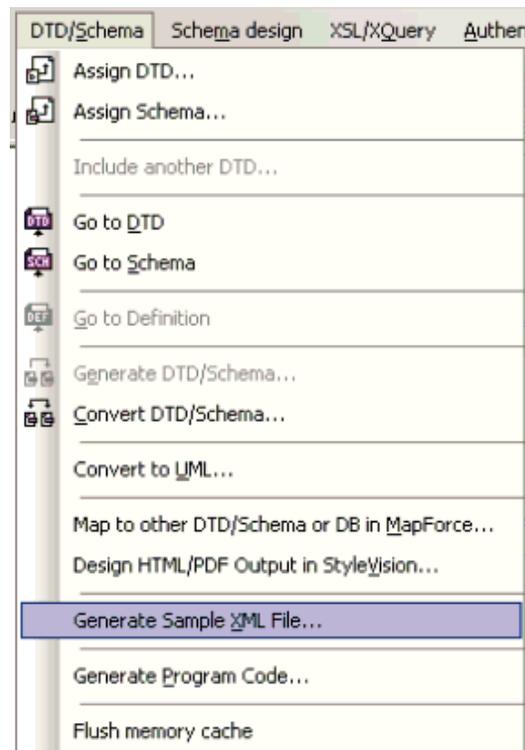
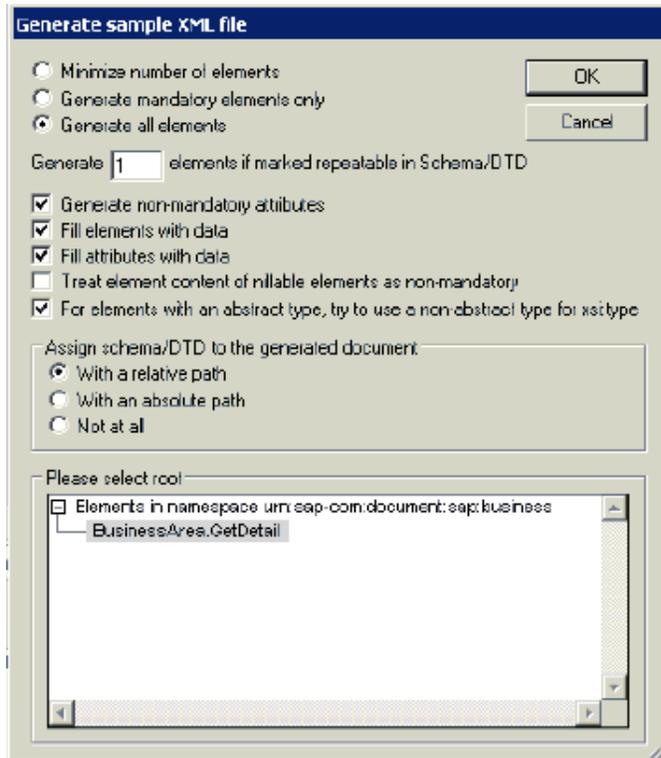


図 1-45 に示すような「Generate Sample XML File」ダイアログが表示されます。

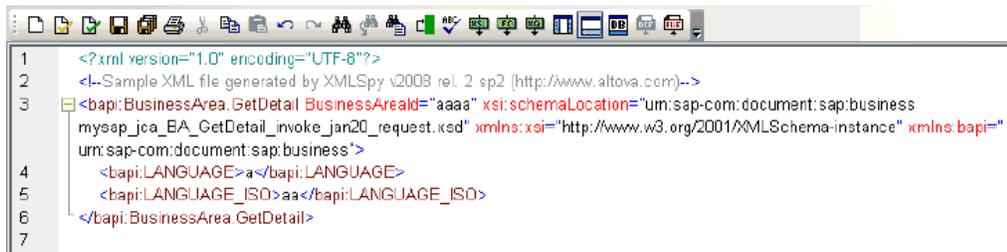
図 1-45 「Generate Sample XML File」 ダイアログ



7. サンプル XML ファイルの適切なパラメータを選択し、準備ができたなら「OK」をクリックします。

図 1-46 に示すように、Altova XML Spy でサンプル XML ファイルが生成されます。

図 1-46 Altova XML Spy で生成されたサンプル XML ファイル



8. 生成された入力 XML ファイルが整形形式で有効であることを確認します。
生成された入力 XML ファイルに必要な値を指定した後で、このファイルを使用して BPM プロセス、BPEL プロセスまたはメディエータ・プロセスを呼び出すことができます。

1.4.4 Oracle Enterprise Manager コンソールでの生成済入力 XML ファイルの使用

生成済の入力 XML ファイルを Oracle Enterprise Manager コンソールで使用する前に、次の前提条件が満たされていることを確認してください。

- JDeveloper で BPM アウトバウンド・プロセスが作成され、SOA_Server1 にデプロイされていること
- JDeveloper でメディエータ・アウトバウンド・プロセスが作成され、SOA_Server1 にデプロイされていること

- JDeveloper で BPEL アウトバウンド・プロセスが作成され、SOA_Server1 にデプロイされていること

メディエータ・アウトバウンド・プロセス

アウトバウンド・メディエータ・プロセスで入力 XML ファイルを使用するには、次の手順を実行します。

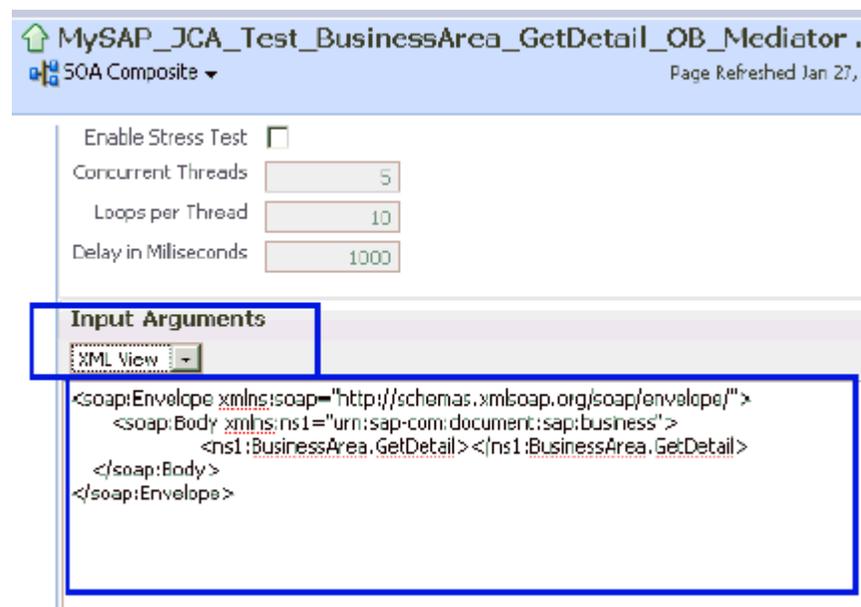
1. 次の URL を使用して、Oracle Enterprise Manager コンソールにログインします。
http://localhost:7001/em
2. 左ペインで「SOA」、「soa-infra (soa_server_name)」、「デフォルト」の順に展開します。
3. アウトバウンド・メディエータ・プロジェクトを選択します (MySAP_JCA_Test_BusinessArea_GetDetail_OB_Mediator など)。
4. 図 1-47 に示すように、コンソール右上隅の「テスト」ボタンをクリックします。

図 1-47 「テスト」ボタン



5. 図 1-48 に示すように、「引数の入力」セクションでリストから「XML 表示」を選択し、入力 XML が表示されていることを確認します。

図 1-48 「引数の入力」セクションの「XML 表示」オプション



注意： メディエータ・プロセスの場合、入力 XML をネームスペース付きで使用することは必須です。

6. Oracle Enterprise Manager コンソールに表示された XML は、次のように変更して使用できます。

Oracle Enterprise Manager コンソールに表示される XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="urn:sap-com:document:sap:business">
<ns1:BusinessArea.GetDetail> </ns1:BusinessArea.GetDetail> </soap:Body>
</soap:Envelope>
```

変更後の XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="urn:sap-com:document:sap:business">
<ns1:BusinessArea.GetDetail BusinessAreaId="1000">
</ns1:BusinessArea.GetDetail> </soap:Body> </soap:Envelope>
```

7. Altova XML Spy を使用して生成されたサンプルの入力 XML は、次の必要な変更を行った後で使用できます。

- a. XML ヘッダーを削除します。
- b. 必要な入力を追加します。
- c. Oracle Enterprise Manager コンソールに表示されている soap ヘッダーを追加します。
- d. ネームスペースを入力XML内のネームスペースと一致するように変更します。

Altova XML Spy を使用して生成された XML

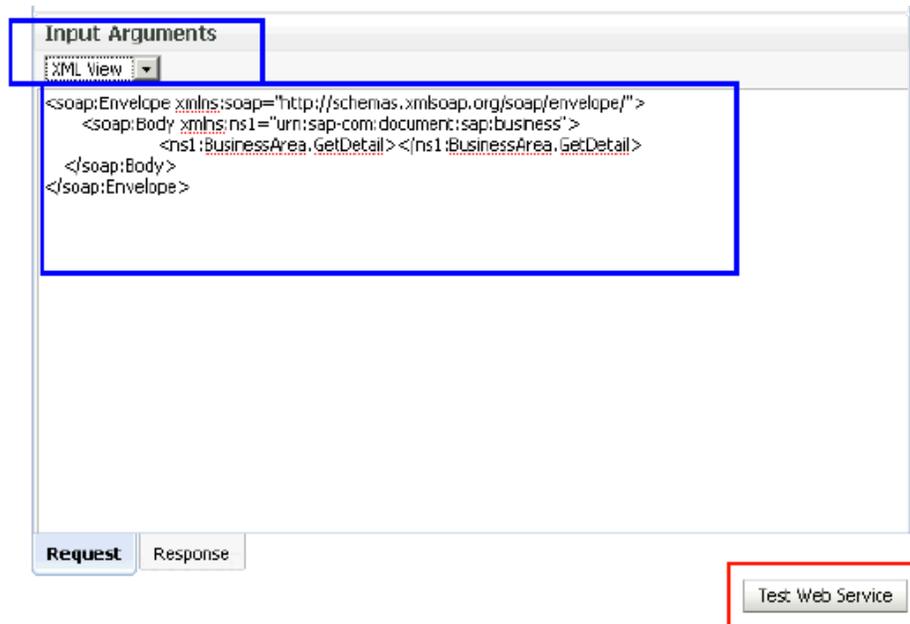
```
<?xml version="1.0" encoding="UTF-8"?> <!--Sample XML file generated by XMLSpy
v2008 rel. 2 sp2 (http://www.altova.com)--> <bapi:BusinessArea.GetDetail
BusinessAreaId="aaaa" xsi:schemaLocation="urn:sap-com:document:sap:business
mysap_jca_BA_GetDetail_invoke_jan20_request.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:bapi="urn:sap-com:document:sap:business">
<bapi:LANGUAGE>a</bapi:LANGUAGE> <bapi:LANGUAGE_ISO>aa</bapi:LANGUAGE_ISO>
</bapi:BusinessArea.GetDetail>
```

変更後の XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="urn:sap-com:document:sap:business">
<ns1:BusinessArea.GetDetail BusinessAreaId="1000"> <ns1:LANGUAGE>D</
ns1:LANGUAGE> < ns1:LANGUAGE_ISO>EN</ ns1:LANGUAGE_ISO>
</ns1:BusinessArea.GetDetail> </soap:Body> </soap:Envelope>
```

8.  1-49 に示すように、入力 XML を指定した後、「Web サービスのテスト」をクリックします。

図 1-49 「Web サービスのテスト」 ボタン



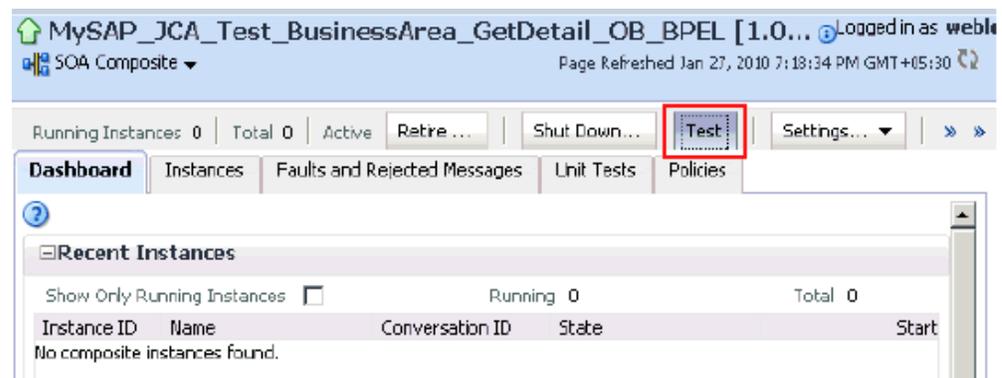
「レスポンス」 タブに出力が表示されます。

BPEL アウトバウンド・プロセス

アウトバウンド BPEL プロセスで入力 XML ファイルを使用するには、次の手順を実行します。

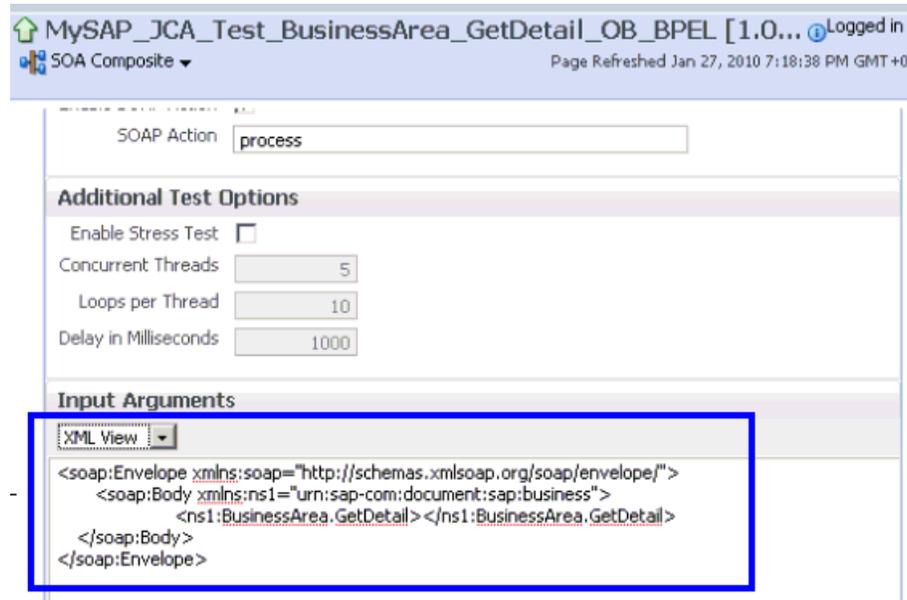
1. 次の URL を使用して、Oracle Enterprise Manager コンソールにログインします。
http://localhost:7001/em
2. 左ペインで「SOA」、「soa-infra (soa_server_name)」、「デフォルト」の順に展開します。
3. アウトバウンド BPEL プロジェクトを選択します (MySAP_JCA_Test_BusinessArea_GetDetail_OB_BPEL など)。
4. 図 1-50 に示すように、コンソール右上隅の「テスト」 ボタンをクリックします。

図 1-50 「テスト」 ボタン



5. 図 1-51 に示すように、「引数の入力」セクションでリストから「XML 表示」を選択し、入力 XML が表示されていることを確認します。

図 1-51 「引数の入力」セクション内の入力 XML ドキュメント



注意： BPEL プロセスの場合、入力 XML をネームスペース付きで使用することは必須ではありません。

- Oracle Enterprise Manager コンソールに表示された XML は、次のように変更して使用できます。

Oracle Enterprise Manager コンソールに表示される XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="urn:sap-com:document:sap:business">
<ns1:BusinessArea.GetDetail> </ns1:BusinessArea.GetDetail> </soap:Body>
</soap:Envelope>
```

変更後のネームスペース付き XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="urn:sap-com:document:sap:business">
<ns1:BusinessArea.GetDetail BusinessAreaId="1000">
</ns1:BusinessArea.GetDetail> </soap:Body>
</soap:Envelope>
```

変更後のネームスペースなし XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns="urn:sap-com:document:sap:business">
<BusinessArea.GetDetail BusinessAreaId="1000">
</BusinessArea.GetDetail> </soap:Body> </soap:Envelope>
```

- Altova XML Spy を使用して生成されたサンプルの入力 XML は、次の必要な変更を行った後で使用できます。
 - XML ヘッダーを削除します。
 - 必要な入力を追加します。
 - Oracle Enterprise Manager コンソールに表示されている soap ヘッダーを追加します。

d. ネームスペースを変更または削除します。

Oracle Enterprise Manager コンソールの「XML 表示」に表示される XML

```
<?xml version="1.0" encoding="UTF-8"?> <!--Sample XML file generated by XMLSpy
v2008 rel. 2 sp2 (http://www.altova.com)--> <bapi:BusinessArea.GetDetail
BusinessAreaId="aaaa" xsi:schemaLocation="urn:sap-com:document:sap:business
mysap_jca_BA_GetDetail_invoke_jan20_request.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:bapi="urn:sap-com:document:sap:business">
<bapi:LANGUAGE>a</bapi:LANGUAGE> <bapi:LANGUAGE_ISO>aa</bapi:LANGUAGE_ISO>
</bapi:BusinessArea.GetDetail>
```

変更後のネームスペース付き XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ns1="urn:sap-com:document:sap:business">
<ns1:BusinessArea.GetDetail BusinessAreaId="1000"> <ns1:LANGUAGE>D</
ns1:LANGUAGE> < ns1:LANGUAGE_ISO>EN</ ns1:LANGUAGE_ISO>
</ns1:BusinessArea.GetDetail> </soap:Body> </soap:Envelope>
```

変更後のネームスペースなし XML

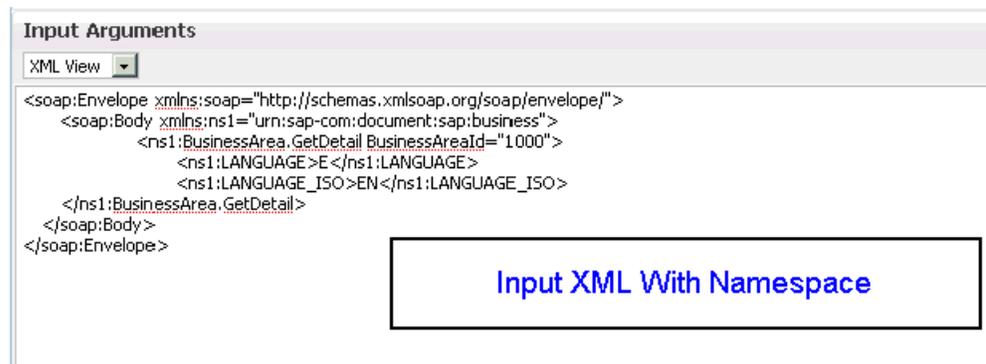
```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns="urn:sap-com:document:sap:business"> <BusinessArea.GetDetail
BusinessAreaId="1000"> <LANGUAGE>D</ LANGUAGE> < LANGUAGE_ISO>EN</ LANGUAGE_
ISO> </BusinessArea.GetDetail> </soap:Body> </soap:Envelope>
```

8. ネームスペース付きまたはネームスペースなしの入力 XML を指定した後、「Web サービスのテスト」をクリックします。

図 1-52 に示すように、「レスポンス」タブに出力が表示されます。

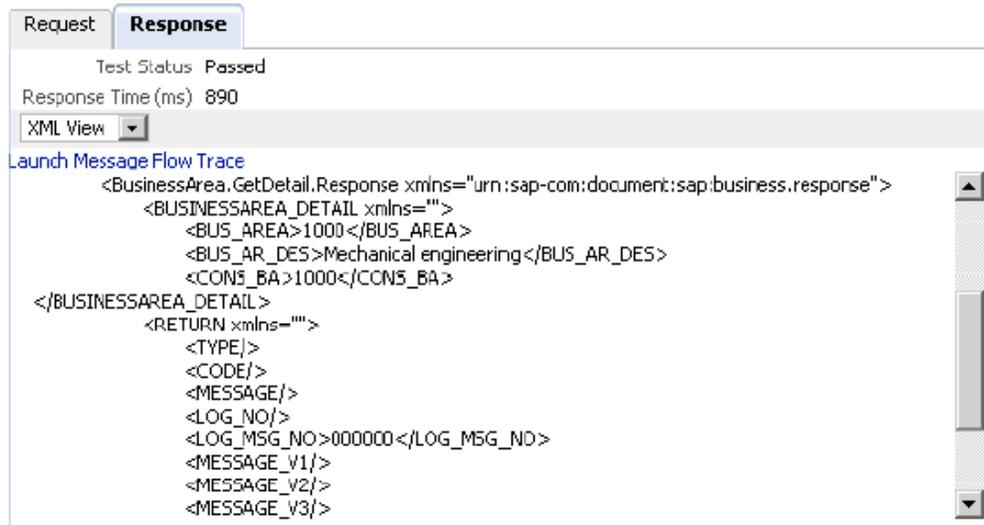
ネームスペース付きの入力 XML

図 1-52 ネームスペース付きの入力 XML ドキュメント



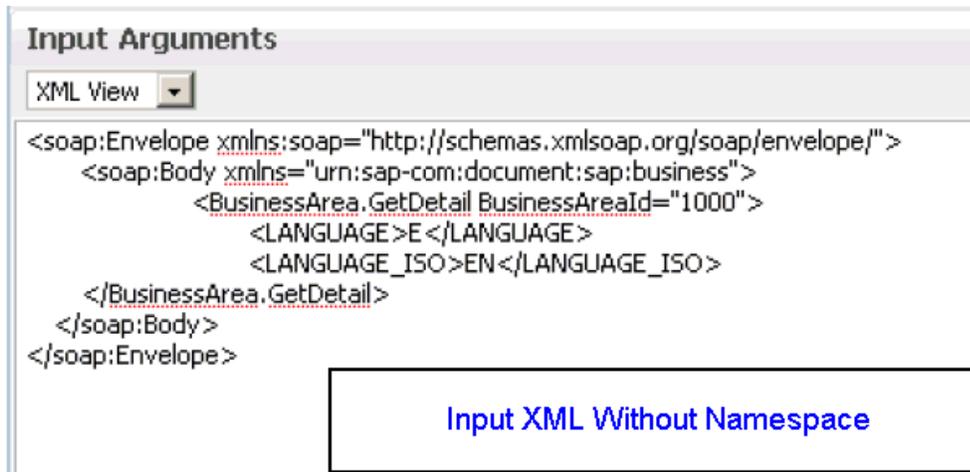
レスポンス

図 1-53 「レスポンス」 タブ



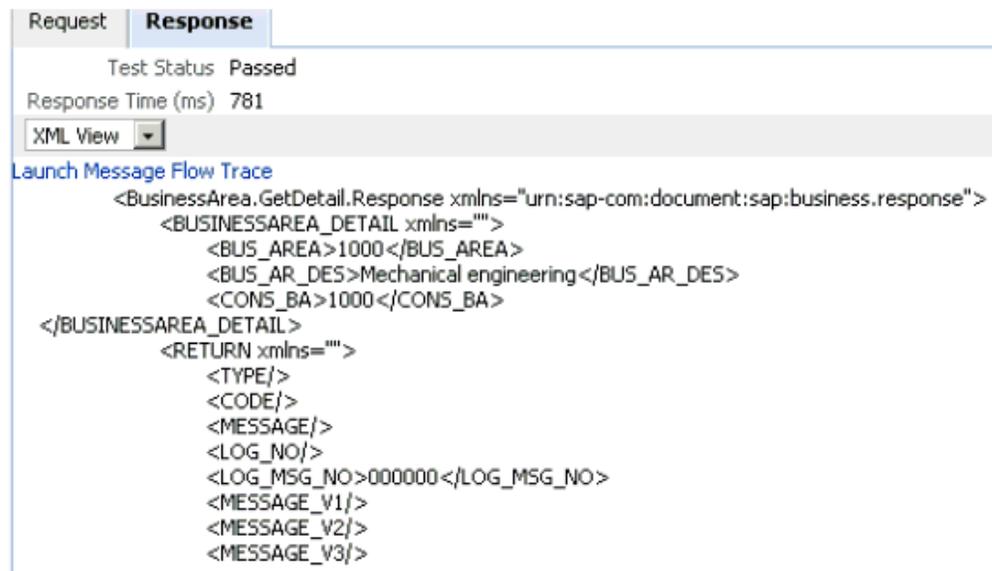
ネームスペースなしの入力 XML

図 1-54 ネームスペースなしの入力 XML ドキュメント



レスポンス

図 1-55 返される XML レスポンス・ドキュメント

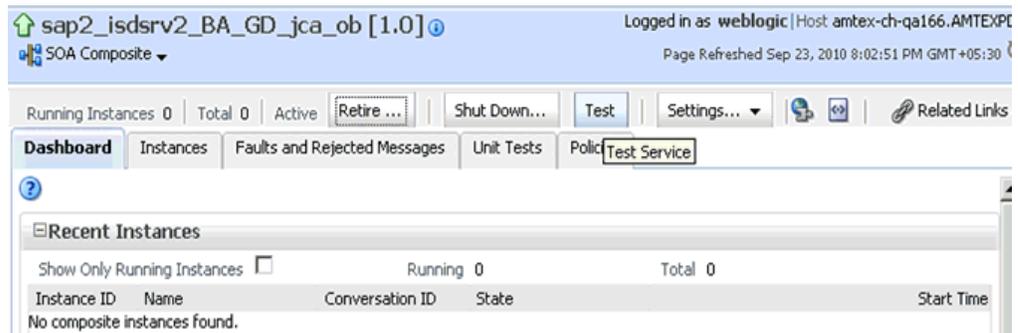


アウトバウンド BPM プロセスでの入力 XML ファイルの使用

アウトバウンド BPM プロセスで入力 XML ファイルを使用するには、次の手順を実行します。

1. 次の URL を使用して、Oracle Enterprise Manager コンソールにログインします。
<http://localhost:7001/em>
2. 左ペインで「SOA」、「soa-infra (soa_server_name)」、「デフォルト」の順に展開します。
3. アウトバウンド BPM プロジェクトを選択します (SAP2_isdsrv2_BA_GD_jca_ob など)。
4. 図 1-56 に示すように、コンソール右上隅の「テスト」をクリックします。

図 1-56 「テスト」ボタン



5. 図 1-57 に示すように、「引数の入力」セクションでリストから「XML 表示」を選択し、入力 XML が表示されていることを確認します。

図 1-57 「引数の入力」セクションで選択された「XML 表示」オプション

Additional Test Options

Enable Stress Test

Concurrent Threads

Loops per Thread

Delay in Milliseconds

Input Arguments

XML View

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body xmlns:ns1="http://xmlns.oracle.com/bpmn/bpmnProcess/Process">
    <ns1:operation xmlns:ns2="urn:sap-com:document:sap:business">
      <ns2:BusinessArea.GetDetail></ns2:BusinessArea.GetDetail>
    </ns1:operation>
  </soap:Body>
</soap:Envelope>
```

注意： BPM プロセスの場合、入力 XML をネームスペース付きで使用することは必須です。

- Oracle Enterprise Manager コンソールに表示された XML は、次のように変更して使用できます。

Oracle Enterprise Manager コンソールに表示される XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body xmlns:ns1="http://xmlns.oracle.com/bpmn/bpmnProcess/Process">
    <ns1:operation xmlns:ns2="urn:sap-com:document:sap:business">
      <ns2:BusinessArea.GetDetail/>
    </ns1:operation>
  </soap:Body>
</soap:Envelope>
```

変更後の XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body xmlns:ns1="http://xmlns.oracle.com/bpmn/bpmnProcess/Process">
    <ns1:operation xmlns:bapi="urn:sap-com:document:sap:business">
      <bapi:BusinessArea.GetDetail BusinessAreaId="0001">
        <bapi:LANGUAGE>D</bapi:LANGUAGE>
        <bapi:LANGUAGE_ISO>DE</bapi:LANGUAGE_ISO>
      </bapi:BusinessArea.GetDetail>
    </ns1:operation>
  </soap:Body>
</soap:Envelope>
```

- Altova XML Spy を使用して生成されたサンプルの入力 XML は、次の必要な変更を行った後で使用できます。
 - XML ヘッダーを削除します。
 - 必要な入力を追加します。
 - Oracle Enterprise Manager コンソールに表示されている SOAP ヘッダーを追加します。

- d. ネームスペースを入力XML内のネームスペースと一致するように変更します。

Altova XML Spy を使用して生成された XML

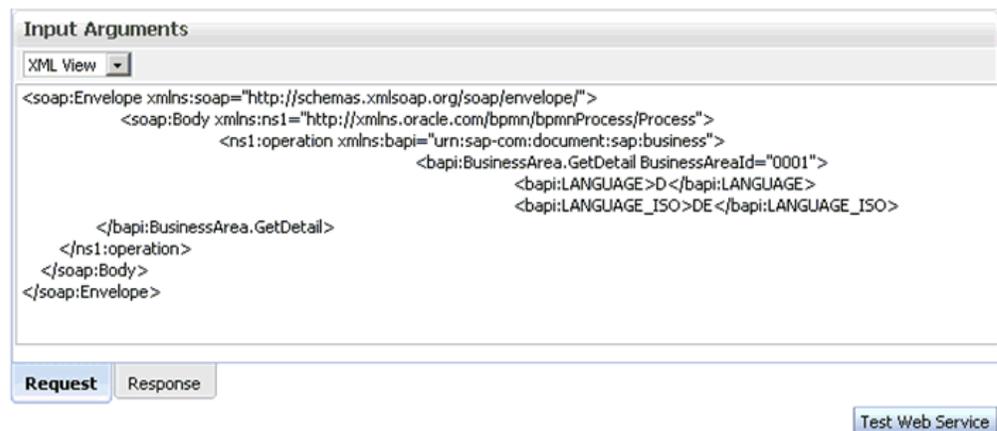
```
<?xml version="1.0" encoding="UTF-8"?> <!--Sample XML file generated by
XMLSpyv2008 rel. 2 sp2 (http://www.altova.com)--> <bapi:BusinessArea.GetDetail
BusinessAreaId="aaaa" xsi:schemaLocation="urn:sap-com:document:sap:business
mysap_jca_BA_GetDetail_invoke_jan20_request.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:bapi="urn:sap-com:document:sap:business">
  <bapi:LANGUAGE>a</bapi:LANGUAGE>
  <bapi:LANGUAGE_ISO>aa</bapi:LANGUAGE_ISO>
</bapi:BusinessArea.GetDetail>
```

変更後の XML

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body xmlns:ns1="http://xmlns.oracle.com/bpmn/bpmnProcess/Process">
    <ns1:operation xmlns:bapi="urn:sap-com:document:sap:business">
      <bapi:BusinessArea.GetDetail BusinessAreaId="0001">
        <bapi:LANGUAGE>D</bapi:LANGUAGE>
        <bapi:LANGUAGE_ISO>DE</bapi:LANGUAGE_ISO>
      </bapi:BusinessArea.GetDetail>
    </ns1:operation>
  </soap:Body>
</soap:Envelope>
```

8. 図 1-58 に示すように、入力 XML を指定した後、「Web サービスのテスト」をクリックします。

図 1-58 入力 XML



出力が宛先フォルダ内に受信されます。

Oracle Application Adapter for SAP R/3 の ベスト・プラクティス

この章では、Oracle Application Adapter for SAP R/3 (SAP JCo 3.0xx) に固有のベスト・プラクティスについて説明します。内容は次のとおりです。

- 2.1 項「接続プールについて」
- 2.2 項「セキュリティ」
- 2.3 項「ロード・バランシング」
- 2.4 項「エンコーディング」
- 2.5 項「SAP サーバーの動作」
- 2.6 項「IDoc の構成」
- 2.7 項「IDoc メタデータ」

2.1 接続プールについて

接続プールとは、特定の宛先へのクライアント接続セットのことです。プールは、指定されたリモート・システムへの新しい接続を自動的に作成したり、既存の接続を返したりします。また、不要になった接続をプールに返す方法も提供します。次のリストに重要な点を示します。

- 各アダプタ・ターゲットには、サイズおよびタイミング・パラメータを備えた固有の接続プールがあります。ターゲットを作成するときには、接続プールに割り当てられるすべてのタスクを処理する十分な接続を持つようにしてください。
- 指定した最大接続数をアプリケーションの実行中に増やすことはできません。ベスト・プラクティスとして、アプリケーションに十分な数を選択する必要があります。
- 異なるアプリケーションによってターゲットおよび接続プールが共有される場合、アプリケーションはその接続プールを作成した基礎となるユーザー ID の SAP 認可権限を共有します。
- 設計時にアプリケーション・エクスプローラを使用してターゲットを作成した場合、実行時には、そのターゲットを使用して作成されるすべてのオブジェクトに対して、指定した接続パラメータが使用されます。接続プロパティが使用環境をサポートできることを確認してください。
- 接続プール・サイズを計算するための一般的なパラメータは、時間 = (呼び出された関数に対する SAP アプリケーション・サーバー実行時間) + (ドキュメントのサイズ) + (ネットワーク遅延) です。特定の状況に対して許容される接続プールの最大サイズを SAP ゲートウェイ管理者に確認することをお勧めします。

- ドキュメントのサイズが大きい場合やトランザクションの実行時間が長い場合、接続タイムアウト・パラメータに大きい値を設定する必要があります。実行時間が非常に長いドキュメントは、フォアグラウンドで実行しないようにします。Oracle Application Adapter for SAP R/3 から呼出し可能なバッチ・ジョブの設定についてシステム管理者と確認することをお勧めします。
- デフォルトでは、SAP Java Connector (SAP JCo) は接続プールに対して 1 つの初期接続をオープンします。その初期接続がビジー状態のときに別のリクエストが到着すると、最大プール・サイズに達するまで SAP JCo は追加の接続をオープンします。最大プール・サイズに達すると、SAP JCo がタスクを中断せずに空き接続を待機する時間を決定する接続待機時間パラメータが呼び出されます。

Oracle Application Adapter for SAP R/3 から SAP への接続の数を監視するために、SAP GUI から SMGW トランザクションを使用できます。

接続プールを適切に最適化するには、アプリケーションおよびそのユーザーに関する知識が必要です。次の点について考慮する必要があります。

- この機能の実行に通常かかる時間
- この機能により返されるデータの量
- この機能を使用するユーザー数

複数の接続ターゲットを構成して、異なる機能ごとに異なる接続プールを使用すると便利です。たとえば、品目があるかどうかを確認する受注照会は実行時間が短いため、最大限のリソースは必要ありません。しかし、ヘッダーや行明細を作成する受注作成は多数の人に実行され、比較的長い実行時間がかかるため、より大きな接続プールを構成することをお勧めします。

2.2 セキュリティ

セキュリティのベスト・プラクティスに関する重要な点を次に示します。

- SAP Java Connector (SAP JCo) のデフォルトは、プレーン・テキストです。SAP JCo 通信にセキュアでないネットワーク・パスがある場合は、データのプライバシーを保護するために RFC 暗号化を備えた SAP ルーター・テクノロジーを使用することをお勧めします。
- SAP ゲートウェイおよびその機能の監視またはアクセスを行うユーザー権限を制限します。SAP JCo 通信を行う実行時ユーザー ID が、SAP ダイアログ・ユーザーではなく通信タイプ・ユーザーであることを確認してください。
- ベスト・プラクティスとして、ファイアウォール内のシステム間の通信には SAP ルーター・メカニズムを常に使用することを考慮してください。

2.3 ロード・バランシング

この項では、SAP を使用したインバウンドおよびアウトバウンドのロード・バランシングのベスト・プラクティスについて説明します。内容は次のとおりです。

- [2.3.1 項「SAP メッセージ・サーバーによる SAP へのインバウンドのロード・バランシング」](#)
- [2.3.2 項「SAP ゲートウェイによる SAP からのアウトバウンドのロード・バランシング」](#)

2.3.1 SAP メッセージ・サーバーによる SAP へのインバウンドのロード・バランシング

インバウンドのロード・バランシングは SAP メッセージ・サーバー経由のログオンに制限されます。メッセージ・サーバーはログオンを複数の SAP アプリケーション・サーバーへ分散するか、特定のアプリケーション・グループによってログオンを特定のサーバーへ分散します。アダプタの接続パラメータを構成するときに、アダプタが使用するメッセージ・サーバーおよびグループについての追加情報を指定する必要があります。Oracle Application Adapter for SAP R/3 は、メッセージ・サーバーを使用した SAP への接続をサポートします。アプリケーション・エクスプローラを使用してアダプタ・ターゲットまたはチャンネルを作成するときに、「メッセージ・サーバー」オプションを選択してください。

2.3.2 SAP ゲートウェイによる SAP からのアウトバウンドのロード・バランシング

登録済プログラムのロード・バランシングは、SAP からリモート宛先に大量のデータを送信する際に使用する手法です。アダプタ・チャンネルは、登録済のプログラム ID を使用して SAP 管理者によって SAP アプリケーションに登録されます。

アウトバウンド・メッセージのロード・バランシングのメカニズムは、管理者によって SAP ゲートウェイ・サーバーで構成されます。ただし、通常次のいずれかのルールに従います。

- 0: ロード・バランシングは行われません。最初の空きの登録済プログラムが使用されます。
- 1: カウンタが最も低いプログラムが使用されます。リクエストが登録済プログラムに割り当てられるたびに、カウンタが 1 つ増えます。
- 2: 負荷が最も少ないプログラム (SAP によって判断される) が使用されます。

アダプタ・チャンネルは SAP ゲートウェイに接続され、同じプログラム ID を SAP ゲートウェイに公開します。この時点で、1 つ以上のサーバーが接続を受け入れます。

登録済サーバー (アダプタ・チャンネル) がゲートウェイに接続する場合、特定の SAP アプリケーション・サーバーにではなく、ゲートウェイに対して接続します。プログラム ID に送信されたすべてのメッセージは、SAP をリスニングするように構成されたチャンネルに到着します。

ゲートウェイとプログラム ID

メッセージのロード・バランシングを増やすには、より多くのチャンネルで同じプログラム ID を公開する必要があります。SAP Java Connector 3 (JCo 3.0xx) では、1 つのインスタンスのプログラム ID とゲートウェイ・サーバーの資格証明の組合せは 1 対 1 の対応に制限されます。しかし、同じインスタンスの別のスレッドを使用するか、分散したサーバーに同じプログラム ID を持つ別のサーバーを使用することによって、ロード・バランシングが可能です。メッセージのロード・バランシングを増やすには、次の手順を実行します。

1. 「チャンネルの追加」ダイアログの「詳細」タブで、「スレッド数」パラメータがデフォルトで 3 スレッドに設定されています。
2. ベスト・プラクティスとして、「スレッド数」パラメータをデフォルト値の 3 の倍数に増やすことをお勧めします。
3. 無駄なリソースや SAP JCo によるスレッドプール・サイズの自動的なダウンキャストを避けるために、チャンネル構成のスレッド数がアプリケーション・サーバーのスレッド数と 1:3 の比で対応するようにします。

チャンネルからメッセージを受信するよう構成されている宛先は、そのチャンネルからのすべてのメッセージを受信します。これは、ビジネス・プロセスの構成において大きな意味を持つ場合があります。メッセージをタイプまたはコンテンツに基づいてルーティングするコーディング手法として、異なるメッセージ、メッセージ・フィルタまたはメッセージ・スプリッタに対して異なるプログラム ID を使用することを検討してください。

同じプログラム ID を使用してサーバーが構成されている場合、SAP システムでロード・バランシングが有効になっているかどうかによって、メッセージが重複したり、到着しないことがあります。プログラム ID のデプロイおよび使用は慎重に行い、その割当ては論理的かつ協調的に（たとえば、部門別またはメッセージ・タイプ別に）行ってください。通常、メッセージのスキーマのチャンネルへのバインドは、メッセージの重複を避けながら複数のサーバー間でプログラム ID を再利用するように行われます。

SAP からのメッセージの受信

サーバーに送信されるメッセージのタイプは、インタフェース・ドキュメント・スタイル (IDoc、RFC、BAPI) および RFC 宛先によって決まります。RFC 宛先は、プログラム ID を SAP 内部に保持するために使用され、またすべてのメッセージをアダプタ・チャンネルにルーティングするためにも使用されます。したがって、リモート・サーバーの IP アドレスを SAP 内部に隠すために、RFC 宛先を制御する SM59 トランザクションはロックしておくことをお勧めします。SAP 内部では、CALL FUNCTION 呼出しで DESTINATION パラメータを指定し、サーバーが存在する RFC 宛先を渡すことにより、RFC 関数モジュールがアダプタ・チャンネルにルーティングされます。次に例を示します。

```
CALL FUNCTION 'RFC_GET_SYSTEM_INFO DESTINATION ' DESTINATION 'MYDEST'
```

MYDEST は、SM59 トランザクション内でリモート TCP (T) 宛先として定義されており、パラメータの 1 つとして「Registered Server Program」を保持します。

IDoc の受信

SAP IDoc では、送信システムと受信システムを定義するために追加の構成が必要です。これらは SAP 論理システムに含まれています。SAP ALE を介してルーティングされる IDoc はすべて、チャンネルを介してメッセージを処理するために、RFC 宛先にリンクされた論理システムを使用します。IDoc の構成手順の完全なリストは、2-5 ページ 2.6 項「IDoc の構成」を参照してください。

非同期 BAPI

BAPI 出力は IDoc 形式で受信する可能性があります。詳細は、SAP のヘルプ・トピック *Distribution Using BAPIs* を参照してください。SAP システムで構成されると、非同期 BAPI は ALE IDoc と同じ方法で受信されます。

2.4 エンコーディング

Unicode システムのアダプタ・チャンネルは、Unicode モードでのみ動作します。SM59 トランザクションで、SAP GUI で宛先を作成する際に、RFC 宛先パラメータとして「Unicode」が選択されていることを確認してください。

SAP Java Connector (SAP JCo) の RFC コンポーネントは、送信側 (クライアント) のターゲット・コード・ページを自動的に判別し、それに応じてクライアントとサーバーの間のコード・ページ変換を調整します。これを変更する方法としては、ターゲット・システムのコード・ページを変更する方法しかなく、Windows ではコントロールパネルの「地域と言語のオプション」を使用します。その他のシステムの場合は、管理者に相談することをお勧めします。一般的に、システムにはコード・ページと言語パックが必要です。コード・ページ間のマッピングには、エンコーディングが使用されます。これは Unicode システムでは簡単に行えますが、Unicode 以外のシス

テムでは困難であったり可能でない場合もあります。原則として、Unicode 以外のシステムでは、システムのコード・ページと言語パックはデータ表示用に限定されます。

SAP からドキュメント (特に IDoc) を受信すると、セグメントに複数の言語が含まれていることがあります。Unicode 以外のシステムでは、ログオン時に使用された言語が出力として正しいと SAP は判断します。他の言語の場合 (特にマルチバイト)、1 つのドキュメント内のすべての言語を正しく取得することは、通常は不可能です。たとえば、Java エンコーディング変数を ISO-8859-2 に設定すると、ドイツ語のウムラウトは正しく送信されますが、日本語の漢字は歪みが生じます。この状況に対する唯一の解決策は、複数のテキスト・セグメントを送信してそれらを 1 つの結果に結合するか、Unicode の使用です。

2.5 SAP サーバーの動作

SAP サーバーの動作に関する重要な点を次に示します。

- 接続を開くことができないかまたはドロップした場合、SAP Java Connector (JCo) は自動的に再接続を試みます。
- JCo は自動的にリスナー構成に最大数のスレッドを設定します。JCo は、必要に応じて要求のディスパッチに使用されるワーカー・スレッドを管理します。スレッドの最大数は最大接続数によって定義されます。
- JCo は SAP からのすべてのイベントを内部キューに入れます。1 つ以上のワーカー・スレッドが使用可能な場合、キャッシュされた要求はディスパッチされます。
- 通信またはアプリケーションのエラーのために SAP から送信されたドキュメントを配信できない場合、SAP サーバーは一定時間待ってからドキュメントの再送信を試みます。SAP サーバーはこれを設定された回数行ってからタイムアウトします。この動作は SAP トランザクション SM59 を使用して、特に RFC 宛先の「*trfc options*」を設定することによって構成できます。

2.6 IDoc の構成

この項では、IDoc を送受信するための SAP システム構成のガイドラインについて説明します。

IDoc を SAP へ送信するためには、次のオブジェクトが必要になります。

- 論理システム: リモートシステム構成のための SAP のプレースホルダ。
- 配信モデル: 論理システムおよび IDoc メッセージ・タイプによってメッセージを送受信するパートナーを構成します。
- パートナ・プロファイル: 配信モデル内の各エントリについて、SAP がメッセージを処理する方法についての物理パラメータを構成します。

IDoc を SAP から受信するためには、次のオブジェクトが必要になります。

- 登録済プログラム ID を含む RFC 宛先: 送信設定を定義します。
- 論理システム: リモートシステム構成のための SAP のプレースホルダ。
- 配信モデル: 論理システムおよび IDoc メッセージ・タイプによってメッセージを送受信するパートナーを構成します。
- パートナ・プロファイル: 配信モデル内の各エントリについて、SAP がメッセージを処理する方法についての物理パラメータを構成します。
- ポート: IDoc の送信先を設定します。RFC 宛先に直接リンクします。

2.7 IDoc メタデータ

IDoc メッセージ・タイプは、セグメントのツリーで構成される IDoc タイプまたは基本タイプを持ちます。各セグメントにはリリースに依存しないセグメント・タイプと、リリースに依存するセグメント定義が含まれます。1つのアクティブな、またはリリース済の IDoc タイプのみである場合も、各 SAP バージョンに依存するセグメント定義の場合もあります。IDoc 処理の一部に、IDoc 内のセグメント情報を SAP の正しいセグメント情報に対応させる処理があります。この情報は、受信した IDoc の IDoc タイプおよび SAP リリースを、対応する SAP メタデータ・バージョンの IDoc タイプおよび SAP リリースと照合することによって得られ、これにより対応するメタデータに基づいて正しいセグメントとフィールド・データを取得します。セグメント・タイプに新しいセグメント定義が追加された場合、それらは次のバージョンの SAP までリリースできません。この情報は SAP に保存されているセグメント定義メタデータ情報には影響しませんが、現在の IDoc 処理には影響する可能性があります。

たとえば、セグメント・タイプ ZSEG には、SAP バージョン 000、SAP リリース 700 および 74 のフィールドを含むセグメント定義 ZSEG があります。新しいバージョンが追加された場合、ZSEG001 にはリリース・ステータスはなく、91 のフィールドが含まれます。これは SAP バージョン 701 がリリースされるまでリリースできません。

アウトバウンドの SAP IDoc を生成する ABAP プログラムが、送信する IDoc のリリース済のセグメントとリリースされていないセグメントをチェックしない場合、送信される IDoc には ZSEG001 の定義が含まれるので、IDoc メタデータを取得するときに問題が生じます。IDoc には新しいセグメント定義が含まれているのにメタデータには含まれていないため、不明なセグメントの例外が発生します。

アダプタ・チャネルの構成で IDoc リリース・プロバイダ・パラメータが SAP リリースに設定されている場合、セグメント ZSEG001 のメタデータを確認するときに、このセグメントはバージョン 700 でリリースされていないため、不明なセグメント・メタデータ・レイヤーから例外が生成されます。アダプタ・チャネルの構成で IDoc リリース・プロバイダ・パラメータが IDoc リリースに設定されていて、IDoc のリリースが SAP ポートで上書きされない場合、リリースは 700 のままであり、同じエラーになります。この状況で正しいメタデータを受信するには、IDoc リリース・プロバイダ・パラメータをユーザー入力に設定し、IDoc リリース・フィールドを空白のままにします。空白の値は SAP のメタデータ・レイヤーに固有の値です。これは、リリース済かどうかにかかわらず IDoc メタデータの最も大きいバージョンを返すことを示します。この例では ZSEG001 のメタデータが取得され、IDoc と一致するためドキュメントが正常に処理されます。

Oracle Application Adapter for Siebel の ベスト・プラクティス

この章では、Oracle Application Adapter for Siebel に固有のベスト・プラクティスについて説明します。内容は次のとおりです。

- 3.1 項「Siebel のベスト・プラクティス」
- 3.2 項「Siebel の接続プーリングの構成」

3.1 Siebel のベスト・プラクティス

この項では、Oracle Application Adapter for Siebel のベスト・プラクティスについて説明します。

- ネイティブの Siebel Java Data Bean (JDB) API を使用して Siebel システムと接続している場合は、Siebel ビジネス・オブジェクトおよびビジネス・サービスを使用して統合を実行する必要があります。HTTP プロトコルが必要な場合は、HTTP を使用する必要があります。ベスト・プラクティスとして、可能な場合は接続に Siebel JDB API を使用することをお勧めします。
- 大量のトランザクション・ボリュームの処理が必要な場合は、ベスト・プラクティスとして、`siebel.properties` ファイルを構成して接続プーリングを使用することをお勧めします。詳細は、3-2 ページ「Siebel の接続プーリングの構成」を参照してください。
- アウトバウンド (サービス) 処理中のセキュアな接続のためにのみ、アダプタ・ターゲット構成ステージで Application Explorer を使用して RSA 暗号化を有効にすることをお勧めします。
- ベスト・プラクティスとして、Siebel で統合オブジェクト・ノードとビジネス・サービス・ノードに生成された XML スキーマ (.xsd) ファイルを使用することをお勧めします。
- Siebel リポジトリ (.srf ファイル) に変更を加えた場合は、ベスト・プラクティスとして、.srf ファイルを再コンパイルし、Siebel サービスを再起動してアダプタに変更を反映させます。
- (COM 接続を使用する) Siebel バージョン 6.2 以下もまだサポートされていますが、ベスト・プラクティスとして、統合プロジェクトでは Siebel 6.3 以上で Siebel JDB API を使用することをお勧めします。
- 挿入、更新または問合せのリクエストを送信する際は、ベスト・プラクティスとして、データの挿入、更新または取得を行う必要のある要素のみをペイロードに含めます。空白の要素はすべて除去してください。
- ベスト・プラクティスとして、統合オブジェクト・ノードまたはビジネス・サービス・ノードを作成する際には名前に特殊文字を含めないでください。

3.2 Siebel の接続プーリングの構成

Siebel の接続プーリングを構成するには、Oracle Application Adapter for Siebel で使用する **siebel.properties** ファイルを作成する必要があります。このファイルを使用すると、Java Data Bean (JDB) API を使用して Siebel に接続するアプリケーションにデフォルトのパラメータを指定できます。この **siebel.properties** ファイルをシステム・クラスパスに追加する必要があります。

次の表に、**siebel.properties** ファイルに追加できるプロパティ値を示します。

プロパティ・タイプ	プロパティ	説明
リクエスト・タイムアウト	siebel.conmgr.txtimeout	<p>Siebel サーバーにリクエストを送信した後に待機する時間 (ミリ秒)。</p> <p>正の整数で指定する必要があります。その他の値は無視されます。デフォルト値は 600000 ミリ秒 (10 分) です。最大値は 2,147,483,647 ms (約 25 日) です。</p>
プールサイズ	siebel.conmgr.poolsize	<p>Application Object Manager プロセスごとに、開いている接続のプールが維持され、このプロセスのすべてのユーザーに共有されます。このパラメータは、プール内に保存される接続の最大数を指定します。</p> <p>パラメータの値は 500 よりも小さい正の整数にする必要があります。その他の値は無視されます。デフォルトは 2 です。</p>
セッション・タイムアウト	siebel.conmgr.sesstimeout	<p>Siebel サーバーがアイドル状態のクライアント・セッションを切断するまでに待機する時間 (秒数)。パラメータの値は非負の整数にする必要があります。</p> <p>デフォルトは、2700 秒 (45 分) です。最大値は 2,147,483,647 秒 (約 68 年) です。</p>

Oracle Application Adapter for PeopleSoft の ベスト・プラクティス

この章では、Oracle Application Adapter for PeopleSoft に固有のベスト・プラクティスについて説明します。

4.1 PeopleSoft イベント (インバウンド処理)

リリース 8.4 については、PeopleTools 8.4 シリーズ用に PeopleSoft から提供されている TCP/IP および HTTP ターゲット・コネクタを使用することをお勧めします。PeopleTools 8.1 シリーズ用に Oracle Software から提供されているターゲット・コネクタは使用しないでください。これらは、リリース 8.1 からリリース 8.4 に移行する既存ユーザーのために、PeopleTools 8.4 シリーズ用に Oracle Software がパッケージしたものです。

Oracle Application Adapter for J.D. Edwards OneWorld のベスト・プラクティス

この章では、Oracle Application Adapter for J.D. Edwards OneWorld に固有のベスト・プラクティスについて説明します。内容は次のとおりです。

- 5.1 項「接続プーリングおよびロード・バランシング」
- 5.2 項「セキュリティ」
- 5.3 項「複数の接続および複数のユーザー」
- 5.4 項「Unicode サポート」
- 5.5 項「配信確認およびエラー・メッセージ」

5.1 接続プーリングおよびロード・バランシング

セッションの接続プーリングは、標準の `jdeRequest` 要素のセッション属性を設定することにより確立されます。セッション属性が空の文字列の場合は、新しいセッションが開始されます。サーバーでは、`SessionManager` シングルトン・クラスが、ユーザー名、パスワードおよび環境名を指定されて、セッション・オブジェクトの新しいインスタンスを作成します。セッション初期化のオーバーヘッドを回避するため、セッションは有効期限が切れるまで再利用できます。以前のリクエストですでに確立されているセッションについては、セッション属性内にセッション ID を指定できません。

セッションの有効期限は、標準の `jdeRequest` 要素の `sessionidle` 属性により指定されます。この属性は、セッション作成リクエストにおいて、セッションのアイドル状態を許容する秒数を指定します。`SessionManager` は、セッションでこの時間内にリクエストが処理されていないと判断した場合、セッションを停止して、関連付けられたすべてのリソースを解放します。詳細は、*J.D. Edwards OneWorld 相互運用性ガイド* を参照してください。

サーバーの `jde.ini` ファイル内にあるランタイム設定によってロード・バランシング構成が制御されます。これらの設定は、`[JDENET]` セクションおよび適用可能な `[JDENET_KERNEL_DEFx]` セクション内にあります。関連するすべての設定については、前の項で説明しています。次の 2 つのタイプのプロセスを制御できます。

- ネットワーク・ジョブ (`JDENET_n`)
- 専用カーネル・プロセス・ジョブ (`JDENET_k`)

詳細は、*J.D. Edwards OneWorld システム管理ガイド* を参照してください。

5.2 セキュリティ

Oracle Application Adapter for J.D. Edwards OneWorld は、標準の JDE セキュリティをサポートしています。セキュリティは、システムで定義されているユーザー・プロファイルにより管理されます。詳細は、*J.D. Edwards OneWorld サーバーおよびワークステーションの管理ガイド*を参照してください。

5.3 複数の接続および複数のユーザー

複数の接続および複数のユーザーは、J.D. Edwards OneWorld で定義されているユーザー・プロファイルによって異なります。詳細は、*J.D. Edwards OneWorld サーバーおよびワークステーションの管理ガイド*を参照してください。

5.4 Unicode サポート

J.D. Edwards XML ソリューションでは、整形式の XML ドキュメントをサポートしています。また、インバウンド処理には UTF8 および UTF16 Unicode 標準を、アウトバウンド処理には UTF8 標準をサポートしています。

5.5 配信確認およびエラー・メッセージ

各トランザクションには、J.D. Edwards OneWorld によってセッション ID が割り当てられます。ベスト・プラクティスとして、アプリケーション P92002 を使用してサーバー側のエラー・メッセージを調べてください。

用語集

アダプタ

ある電子インタフェースから他の電子インタフェースへの (機能を失わない) 適応を可能にすることにより、汎用的な接続性を提供します。

エージェント

リスナーとドキュメントでサービス・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。チャンネルは、1つ以上のイベント・ポートをアダプタで管理される特定のリスナーにバインドします。

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

0

Oracle Application Adapter for Siebel

接続プーリング, 3-2

Oracle Unified Method (OUM), vi

い

一般的なベスト・プラクティス, 1-1

か

管理対象接続, 1-1

に

入力 XML ドキュメント

生成, 1-28

へ

ベスト・プラクティス

Oracle Application Adapter for J.D. Edwards

OneWorld, 5-1

Oracle Application Adapter for PeopleSoft, 4-1

Oracle Application Adapter for SAP R/3, 2-1

Oracle Application Adapter for Siebel, 3-1

一般的, 1-1

