Oracle® Fusion Middleware
Migration Guide for Oracle Application Integration Architecture
11g Release 1 (11.1.1.9.0)
E17361-09

December 2014
Describes the overall migration strategy, process, and utilities used to migrate from AIA 2.4 and 2.5 to AIA 11g Release 1 (11.1.1.9.0). Describes how to migrate custom process integrations built using Foundation Pack. Describes how to migrate extensions to AIA-delivered Foundation Pack 11g Release 1-based PIPs.
## Contents

**Preface** ........................................................................................................................................................................... ix
  Audience ............................................................................................................................................................................ ix
  Oracle AIA Guides ............................................................................................................................................................ ix
  Related Documents ............................................................................................................................................................ ix
  Documentation Accessibility ............................................................................................................................................. x
  Conventions ........................................................................................................................................................................ x

**What's New in This Guide for Release 11.1.1.9** ........................................................................................................... xi

### 1 Introduction to Migrating AIA 2.4 and 2.5 Artifacts to Foundation Pack 11g Release 1
  1.1 Overview of Migration Tasks for Custom Process Integrations Built Using Foundation Pack 1-1
  1.2 Overview of Process Integration Pack Extension Migration Tasks................................................. 1-2

### 2 Preparing to Migrate AIA Artifacts to AIA 11g Release 1
  2.1 Preparing Your Migration Environment................................................................................................. 2-1
  2.1.1 Preparing a Certified Database for the Migration .............................................................................. 2-1
  2.1.2 Preparing Oracle Fusion Middleware 11g Release 1 for Migration ................................................. 2-2
  2.1.3 Preparing AIA Foundation Pack 11g Release 1 for Migration................................................................. 2-2
  2.1.4 Preparing Oracle Enterprise Repository 11g Release 1 Setup for the Migration (Optional) 2-3
  2.1.5 Preparing Oracle Service Registry 11g Release 1 for the Migration (Optional) ....................... 2-3
  2.2 Before You Begin.................................................................................................................................................... 2-3
  2.3 Creating and Deploying AIA Components as Web Archive (WAR) Files..................................... 2-3

### 3 Migrating Enterprise Object Library Customizations
  3.1 EOL Migration Prerequisites......................................................................................................................... 3-1
  3.2 Performing the Migration........................................................................................................................................ 3-1
  3.2.1 Step 1: Copy Customized EnterpriseObjectLibrary Directory from the Old AIA Release Installation 3-2
  3.2.2 Step 2: Copy Uncustomized Old EnterpriseObjectLibrary Directory from the Old AIA Release Installation 3-2
3.2.3 Step 3: Grant Write Permission to the EnterpriseObjectLibrary Directory in the New AIA Release Installation 3-2
3.2.4 Step 4: Execute the EOL Upgrade Utility ................................................................. 3-2
3.2.5 Step 5: Review the Output File ................................................................................. 3-4
3.2.5.1 Scenario 1: Namespace Declaration Conflicts ...................................................... 3-5
3.2.5.2 Scenario 2: complexType Name Conflicts ......................................................... 3-6
3.2.6 Step 6: Handling Imports and Includes of Non-AIA-Delivered Schema Modules ... 3-6
3.3 Running Schema Validations ......................................................................................... 3-7
3.4 Running XML Instance Validations ............................................................................. 3-10

4 Migrating New Enterprise Business Objects

5 Migrating New Enterprise Business Service Definitions

6 Migrating Domain Value Map and Cross-Reference Metadata

6.1 Migrating DVM and Cross-Reference Metadata.......................................................... 6-1

7 Migrating Services

7.1 Overview of AIA Service Migration Using the AIA Migration Utility......................... 7-1
7.2 How to Migrate Adapter Services ............................................................................. 7-2
7.3 How to Migrate Provider ABCSs ............................................................................. 7-3
7.4 How to Migrate EBSs ......................................................................................... 7-4
7.5 How to Migrate Requester ABCSs ........................................................................ 7-8

8 Migrating Extension Services

8.1 Migrating Services Developed Using BPEL or Oracle Mediator ............................... 8-1
8.2 Migrating Services Developed Using Other Technologies ....................................... 8-1

9 Migrating Routing Rules

10 Migrating and Configuring Error Handling Artifacts

10.1 How to Migrate Fault Policies .................................................................................. 10-1
10.2 How to Configure Transaction Configurations ...................................................... 10-2
10.3 How to Configure AIAConfigurationProperties.xml for Error Handling ............ 10-2
10.4 How to Migrate the Error Notification Setup ....................................................... 10-3
10.5 How to Configure the Oracle User Messaging Service for Email Notifications ........ 10-3
10.6 How to Configure User Email Addresses ............................................................. 10-4
10.7 How to Migrate Custom Error Handlers ............................................................... 10-4
10.8 How to Migrate Custom Subscribers ..................................................................... 10-6

11 Migrating Composite Application Validation System Definitions

11.1 Migrating Custom CAVS Definitions ..................................................................... 11-1

12 Making Migrated Services Deployable

12.1 Step 1: Annotating Composites .............................................................................. 12-1
12.2 Step 2: Harvesting AIA Content ................................................................. 12-1
12.3 Step 3: Generate Bill of Material ............................................................. 12-2
12.4 Step 4: Deploy Composites .............................................................. 12-2

13 Migrating AIA 2.4 and 2.5 Business Service Repository Artifacts to Oracle Enterprise Repository Artifacts
13.1 Preparing to Harvest the Services ...................................................... 13-1
13.2 Harvesting Services to Oracle Enterprise Repository .................... 13-1

14 Migrating Actual Domain Value Map and Cross-Reference Data
14.1 Migrating Cross-Reference Data .......................................................... 14-1
14.2 Migrating Extended DVM Data .......................................................... 14-2

15 Migrating to an AIA 11g Release 1 Production Environment
15.1 General Guidelines for Migrating to an AIA 11g Release 1 Production Environment. 15-2

16 Migrating ODI Flows from AIA Foundation Pack 2.4 and 2.5 to AIA Foundation Pack 11g Release 1
16.1 Overview of ODI Flow Migration Tasks .............................................. 16-1
16.2 Describing the Source 10g ODI Server ................................................ 16-2
16.3 Describing the Target 11g ODI Server .................................................. 16-3
16.4 Migrating ODI ...................................................................................... 16-3
16.5 Performing Post-Migration Steps ........................................................ 16-11
16.6 Handling Cross-Reference Schema Changes ..................................... 16-11
16.7 Importing the Latest Knowledge Module for Cross-References .......... 16-11
16.8 Changing the Model and Interfaces That Involve Cross-References .... 16-12

17 Using the AIA Migration Utility
17.1 Overview of the AIA Migration Utility .................................................... 17-1
17.2 How to Use the AIA Migration Utility .................................................. 17-2

18 Loading AIA Metadata to Oracle Metadata Services Repository
18.1 Accessing MDS from the AIA Service Artifact ..................................... 18-1
18.2 Moving Abstract Service WSDLs into MDS ......................................... 18-2
18.3 Migrating the AIA Configuration Properties File ................................ 18-5
18.4 Moving Custom Extensions to EOL, ESL, and AOL to MDS ............... 18-6
18.5 Updating MDS .................................................................................... 18-6

Index
List of Examples

<table>
<thead>
<tr>
<th>Page</th>
<th>Example Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Editing weblogic.xml</td>
</tr>
<tr>
<td>2-2</td>
<td>Editing build.xml</td>
</tr>
<tr>
<td>3-1</td>
<td>EOL Upgrade Utility Run in Report Mode for Linux, Solaris SPARC, IBM AIX, or HP-UX</td>
</tr>
<tr>
<td>3-2</td>
<td>EOL Upgrade Utility Run in Report Mode for Windows</td>
</tr>
<tr>
<td>3-3</td>
<td>EOL Upgrade Utility Run in Upgrade Mode for Linux, Solaris SPARC, IBM AIX, or HP-UX</td>
</tr>
<tr>
<td>3-4</td>
<td>EOL Upgrade Utility Run in Upgrade Mode for Windows</td>
</tr>
<tr>
<td>3-5</td>
<td>Sample Run of EOL Upgrade Utility in Upgrade Mode for Linux, Solaris SPARC, IBM AIX, or HP-UX</td>
</tr>
<tr>
<td>7-1</td>
<td>Application Service WSDL Reference in bpel.xml</td>
</tr>
<tr>
<td>7-2</td>
<td>wsdlRuntimeLocation in bpel.xml</td>
</tr>
<tr>
<td>7-3</td>
<td>Filter Expression Before Migration</td>
</tr>
<tr>
<td>7-4</td>
<td>Filter Expression After Migration</td>
</tr>
<tr>
<td>7-5</td>
<td>Concrete URL in the composite.xml binding,ws Section</td>
</tr>
<tr>
<td>7-6</td>
<td>Part Name for the Source Service</td>
</tr>
<tr>
<td>7-7</td>
<td>Part Name for the Target Service</td>
</tr>
<tr>
<td>7-8</td>
<td>Pointing the CAVS Endpoint to the Abstract WSDL in composite.xml</td>
</tr>
<tr>
<td>7-9</td>
<td>Pointing the CAVS Endpoint to the Abstract WSDL in the Component Type File</td>
</tr>
<tr>
<td>7-10</td>
<td>Changing Filter Expressions in the .mplan File: Filter Expression with Xref Calls Before Migration</td>
</tr>
<tr>
<td>7-11</td>
<td>Changing Filter Expressions in the .mplan File: Filter Expression with xref Calls After Migrating Manually</td>
</tr>
<tr>
<td>7-12</td>
<td>Invoked Service Pointing to abstract WSDL in composite.xml</td>
</tr>
<tr>
<td>7-13</td>
<td>bpel.xml with wsdlRuntimeLocation Commented Out</td>
</tr>
<tr>
<td>10-1</td>
<td>Sample QueryCustomerPartyPortalProvABCSImplFaultPolicy.xml with Fault Policies Defined</td>
</tr>
<tr>
<td>10-2</td>
<td>fault-bindings.xml File</td>
</tr>
<tr>
<td>10-3</td>
<td>Sample Code from the AIA Foundation Pack Demo Implementation</td>
</tr>
<tr>
<td>10-4</td>
<td>Sample JCA file from the AIA Demo</td>
</tr>
<tr>
<td>18-1</td>
<td>Sample adf-config.xml</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>3-1</td>
<td>Project Properties</td>
</tr>
<tr>
<td>3-2</td>
<td>Application Navigator</td>
</tr>
<tr>
<td>3-3</td>
<td>Audit Application Sources</td>
</tr>
<tr>
<td>3-4</td>
<td>Run Audit Application Sources</td>
</tr>
<tr>
<td>3-5</td>
<td>Audit Log</td>
</tr>
<tr>
<td>3-6</td>
<td>Sample XML Instance File Copied to the Project Directory</td>
</tr>
<tr>
<td>3-7</td>
<td>Sample XML file in the Application Navigator</td>
</tr>
<tr>
<td>7-1</td>
<td>AIA 2.4 and 2.5 to AIA Foundation Pack 11g Release 1 Technical Upgrade Path</td>
</tr>
<tr>
<td>15-1</td>
<td>Migrating to the AIA 11g Release 1 Production Environment</td>
</tr>
<tr>
<td>16-1</td>
<td>ODI Migration Task Flow</td>
</tr>
<tr>
<td>16-2</td>
<td>Specify Operation</td>
</tr>
<tr>
<td>16-3</td>
<td>Prerequisites</td>
</tr>
<tr>
<td>16-4</td>
<td>Specify Target Database</td>
</tr>
<tr>
<td>16-5</td>
<td>Specify Work Repositories User Name</td>
</tr>
<tr>
<td>16-6</td>
<td>Examining Components</td>
</tr>
<tr>
<td>16-7</td>
<td>Upgrade Summary</td>
</tr>
<tr>
<td>16-8</td>
<td>Upgrading Components</td>
</tr>
<tr>
<td>16-9</td>
<td>Upgrade Complete</td>
</tr>
<tr>
<td>16-10</td>
<td>Update XREF_TABLE_NAME</td>
</tr>
<tr>
<td>16-11</td>
<td>Mandatory Option Cleared for the LAST_ACCESSSED Column</td>
</tr>
<tr>
<td>16-12</td>
<td>Active Mapping Option Cleared for the LAST_ACCESSSED Column</td>
</tr>
<tr>
<td>16-13</td>
<td>Delete Option in the Context Menu</td>
</tr>
<tr>
<td>16-14</td>
<td>Edits to the LAST_MODIFIED Column</td>
</tr>
<tr>
<td>16-15</td>
<td>Implementation Tab Value Set to SYSTIMESTAMP</td>
</tr>
<tr>
<td>16-16</td>
<td>Mapping Value Set to the Fully Qualified Name of the Cross-Reference</td>
</tr>
<tr>
<td>16-17</td>
<td>Update of the Physical Schema to the Cross-Reference Schema</td>
</tr>
<tr>
<td>17-1</td>
<td>AIA HOME/util Folder Content</td>
</tr>
</tbody>
</table>
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–1</td>
<td>PIP Extension Migration Tasks</td>
<td>1-3</td>
</tr>
<tr>
<td>7–1</td>
<td>AIA 2.4 and 2.5 Artifacts and Their Equivalent AIA Foundation Pack 11g Release 1 Artifacts</td>
<td>7-2</td>
</tr>
<tr>
<td>10–1</td>
<td>AIA 11g Release 1 ConfigurationProperties.xml for Error Handling</td>
<td>10-3</td>
</tr>
<tr>
<td>17–1</td>
<td>AIAMigrationUtility.properties Values</td>
<td>17-2</td>
</tr>
</tbody>
</table>
Welcome to *Oracle Fusion Middleware Migration Guide for Oracle Application Integration Architecture*. This document describes the overall migration strategy, process, and utilities used to migrate from AIA 2.4 and 2.5 to AIA 11g Release 1 (11.1.1.9.0). It describes how to migrate custom process integrations built using Foundation Pack. Additionally, it describes how to migrate extensions to AIA-delivered Foundation Pack 11g Release 1-based PIPs.

**Audience**

This document is intended for users migrating Oracle Application Integration Architecture (AIA) 2.4 or 2.5 to AIA 11g Release 1 (11.1.1.9.0).

**Oracle AIA Guides**

In addition to this *Oracle Fusion Middleware Migration Guide for Oracle Application Integration Architecture*, we provide the following Oracle AIA guides for this 11.1.1.9.0 release:

- *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle Application Integration Architecture Foundation Pack*
- *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*
- *Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack*
- *Oracle Fusion Middleware Infrastructure Components and Utilities User’s Guide for Oracle Application Integration Architecture Foundation Pack*
- *Oracle Fusion Middleware Product-to-Guide Index for Oracle Application Integration Architecture Foundation Pack*

**Related Documents**

For more information, see the following documents delivered as a part of the overall Oracle Fusion Middleware 11.1.1.9.0 documentation library:

- *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite and Oracle Business Process Management Suite*
Oracle Fusion Middleware Developer’s Guide for Oracle SOA Suite
Oracle Fusion Middleware Security and Administrator’s Guide for Web Services

Documentation Accessibility
For information about Oracle’s commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support
Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Conventions
The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td>italic</td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
What's New in This Guide for Release 11.1.1.9

This guide has no updates for release 11.1.1.9.

For a list of known issues (release notes), see the "Known Issues for for Oracle SOA Products and Oracle AIA Foundation Pack" at http://www.oracle.com/technetwork/middleware/docs/soa-aiafp-knownissuesindex-364630.html.
This guide provides details about the overall migration strategy, process, and utilities used to migrate from Oracle Application Integration Architecture (AIA) 2.4 and 2.5, built on Oracle Fusion Middleware 10g, to AIA 11g Release 1, built on Fusion Middleware 11g Release 1.

The migration involves upgrades at various levels, from individual services to server environments. AIA 2.4 and 2.5 installations based on Oracle Fusion Middleware 10g use BPEL and Enterprise Service Bus, as well as Oracle Fusion Middleware’s Oracle Data Integrator (ODI) components.

**Note:** Migrate all the services in AIA 2.4 or 2.5 to AIA 11g R1. AIA does not support services that are migrated and deployed on AIA 11g R1 to make call to the AIA 2.5 or AIA 2.4 services. The services that are migrated cannot have references to AIA 2.4 or AIA 2.5 services.

Migration to Oracle Fusion Middleware 11g-based AIA Foundation Pack 11g Release 1 installation requires completion of the tasks covered in this guide.

We discuss how to migrate two AIA offerings: Foundation Pack, and Process Integration Packs (PIPs).

This chapter includes the following sections:

- **Overview of Migration Tasks for Custom Process Integrations Built Using Foundation Pack**

- **Overview of Process Integration Pack Extension Migration Tasks**

### 1.1 Overview of Migration Tasks for Custom Process Integrations Built Using Foundation Pack

To migrate custom process integrations built using Foundation Pack, refer to the following set of tasks and links.

1. **Chapter 2, "Preparing to Migrate AIA Artifacts to AIA 11g Release 1"**
   
   Guides you through the environment setup and prerequisite steps prior to performing the migration.

2. **Chapter 3, "Migrating Enterprise Object Library Customizations"**
1.2 Overview of Process Integration Pack Extension Migration Tasks

Caution: This section applies to customers who have licensed an AIA-delivered Foundation Pack 11g Release 1-based PIP. Customers should not migrate any AIA-delivered 2.4 or 2.5 PIPs.

PIP customers should complete the following tasks to migrate only the extensions made to AIA-delivered PIPs.
Extensions include any of the modifications:

■ Adding extension services to delivered ABCSs.
■ Adding new ABCSs.
■ Adding new EBOs.
■ Adding new EBSs.
■ Extending the delivered Business Service Objects.
■ Adding or modifying DVM metadata and actual data.
■ Adding or modifying the cross-reference metadata and actual data.
■ Adding or modifying the routing rules.

Prior to performing the PIP extension migration, you must:

1. Catalog all of your extensions and customizations to the 10g-based 2.4 or 2.5 PIP. Review AIA-delivered Foundation Pack 11g Release 1-based PIP functionality and documentation to determine which 10g-based extensions and customizations you will need to migrate.
2. Install WebLogic Server.
3. Install Oracle SOA Suite.
4. Install Foundation Pack 11g Release 1.
5. Install the AIA-delivered Foundation Pack 11g Release 1-based PIP.

Table 1–1 will guide you through the tasks and documentation required for your migration.

Table 1–1  PIP Extension Migration Tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Set up the environment.</td>
<td>Perform tasks in Chapter 2, &quot;Preparing to Migrate AIA Artifacts to AIA 11g Release 1.&quot;</td>
</tr>
</tbody>
</table>
No: Go to step 3. |
No: Go to step 4. |
No: Go to step 5. |
No - go to step 6. |
### Table 1–1 (Cont.) PIP Extension Migration Tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>6. Have you extended the Oracle delivered PIPs?</td>
<td>■ Yes: Perform tasks in Chapter 7, &quot;Migrating Services.&quot;</td>
</tr>
<tr>
<td>■ Adding additional Application Business Connector Services?</td>
<td>■ How to Migrate Provider ABCSs</td>
</tr>
<tr>
<td>■ Adding additional ABCS to implement new integration flows?</td>
<td>■ How to Migrate Requester ABCSs</td>
</tr>
<tr>
<td>Then perform tasks in Chapter 12, &quot;Making Migrated Services Deployable.&quot;</td>
<td></td>
</tr>
<tr>
<td>■ No: Go to step 7.</td>
<td></td>
</tr>
<tr>
<td>7. Have you built your own custom Adapter Services or Enterprise Business Services using Foundation Pack?</td>
<td>■ Yes: Perform tasks in Chapter 7, &quot;Migrating Services.&quot;</td>
</tr>
<tr>
<td>■ How to Migrate Adapter Services</td>
<td>■ How to Migrate EBSs</td>
</tr>
<tr>
<td>Then, perform tasks in Chapter 12, &quot;Making Migrated Services Deployable.&quot;</td>
<td></td>
</tr>
<tr>
<td>■ No: Go to step 8.</td>
<td></td>
</tr>
<tr>
<td>8. Have you extended PIP ABCSs using delivered extension hooks?</td>
<td>■ Yes: Perform tasks in Chapter 8, &quot;Migrating Extension Services.&quot;</td>
</tr>
<tr>
<td>■ Then, if the extension services have been developed using the Oracle SOA Suite 11g Service Component Architecture, perform tasks in Chapter 12, &quot;Making Migrated Services Deployable.&quot;</td>
<td></td>
</tr>
<tr>
<td>■ No: Go to step 9.</td>
<td></td>
</tr>
<tr>
<td>■ No: go to step 10.</td>
<td></td>
</tr>
<tr>
<td>10. Have you defined custom fault policies or developed custom error handlers or customized the AIA delivered AIA Error Handling framework?</td>
<td>■ Yes: Perform tasks in Chapter 10, &quot;Migrating and Configuring Error Handling Artifacts.&quot;</td>
</tr>
<tr>
<td>■ No: Go to step 11.</td>
<td></td>
</tr>
<tr>
<td>■ No: Go to step 12.</td>
<td></td>
</tr>
<tr>
<td>12. Do you need to migrate any custom-built content (EBO, EBM, ABCS, EBS, composites) from the Business Service Repository to Oracle Enterprise Repository?</td>
<td>■ Yes: Perform tasks in Chapter 13, &quot;Migrating AIA 2.4 and 2.5 Business Service Repository Artifacts to Oracle Enterprise Repository Artifacts.&quot;</td>
</tr>
<tr>
<td>■ No - go to step 13.</td>
<td></td>
</tr>
<tr>
<td>13. Do you have additional custom DVM and cross-reference values on top of the Oracle seeded values?</td>
<td>■ Yes: Perform tasks in Chapter 14, &quot;Migrating Actual Domain Value Map and Cross-Reference Data.&quot;</td>
</tr>
<tr>
<td>■ No - go to step 14.</td>
<td></td>
</tr>
<tr>
<td>14. Migrate to an AIA 11g Production Environment</td>
<td>Perform tasks in Chapter 15, &quot;Migrating to an AIA 11g Release 1 Production Environment.&quot;</td>
</tr>
<tr>
<td>Task</td>
<td>Instructions</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>15. Any ODI flows?</td>
<td>Yes: Perform tasks in Chapter 16, &quot;Migrating ODI Flows from AIA Foundation Pack 2.4 and 2.5 to AIA Foundation Pack 11g Release 1.&quot;</td>
</tr>
</tbody>
</table>
This chapter describes how to prepare your migration environment, lists the prerequisites, and describes how to deploy Oracle Application Integration Architecture (AIA) components as Web Archive (WAR) files.

In Oracle Fusion Middleware 11g Release 1, any SOA composite can be deployed by referring to the abstract WSDL. This allows AIA projects to be migrated independently and in any order.

---

**Note:** Migrate all the services in AIA 2.4 or 2.5 to AIA 11g R1. AIA does not support services that are migrated and deployed on AIA 11g R1 to make call to the AIA 2.5 or AIA 2.4 services. The services that are migrated cannot have references to AIA 2.4 or AIA 2.5 services.

---

This chapter includes the following sections:

- Preparing Your Migration Environment
- Before You Begin
- Creating and Deploying AIA Components as Web Archive (WAR) Files

### 2.1 Preparing Your Migration Environment

This section includes the following topics:

- Section 2.1.1, "Preparing a Certified Database for the Migration"
- Section 2.1.2, "Preparing Oracle Fusion Middleware 11g Release 1 for Migration"
- Section 2.1.3, "Preparing AIA Foundation Pack 11g Release 1 for Migration"
- Section 2.1.4, "Preparing Oracle Enterprise Repository 11g Release 1 Setup for the Migration (Optional)"
- Section 2.1.5, "Preparing Oracle Service Registry 11g Release 1 for the Migration (Optional)"

#### 2.1.1 Preparing a Certified Database for the Migration

AIA Foundation Pack 2.4 and 2.5 installations use a database to make the best use of the following:

- Storing cross-reference data.
Cross-references enable you to dynamically map values for equivalent entities created in different applications. When you create or update objects in one application, you may also want to propagate the changes to other applications.

For example, when a new customer is created in one of your applications, you may want to create a new entry for the same customer in another application. However, the applications you are integrating may be using different entities to represent the same information.

- Oracle Advanced Queuing (AQ) for message persistence (only available for use with Oracle Database).
- Storing AIA system registration and error notification information.

When migrating to AIA Foundation Pack 11g Release 1, AIA recommends using a fresh installation of the database and then migrating data from the existing AIA Foundation Pack 2.4 or 2.5 environments.

Another option is to have an existing 10g SOA database hosting the AIA instance to be upgraded to the 11g Release 1. Oracle databases do support two types of database upgrade: side-by-side migration or grace period cutover migration.

For more information, see the Oracle Database 11g Upgrade Guide.

If you have selected an existing 10g database for the upgrade, you must ensure that all in-flight instances are allowed to complete before the database upgrade.


### 2.1.2 Preparing Oracle Fusion Middleware 11g Release 1 for Migration

When migrating to Foundation Pack 11g Release 1 you must use a fresh installation of Oracle Fusion Middleware 11g Release 1. Oracle Fusion Middleware enables enterprises to create and run agile, intelligent business applications while maximizing IT efficiency through full utilization of modern hardware and software architectures.

For more information, see the Oracle Fusion Middleware Installation Planning Guide.

Set up the Oracle Fusion Middleware environment for AIA Foundation Pack 11g Release 1.

For more information, see Oracle Fusion Middleware Installation and Upgrade Guide for Oracle Application Integration Architecture Foundation Pack.

### 2.1.3 Preparing AIA Foundation Pack 11g Release 1 for Migration

Foundation Pack is a prebuilt integration solution that provides the programming model, best practices, and application-independent data model, as well as supporting tools to implement, test, diagnose, and govern your service-oriented architecture.

To migrate your existing services and integrations to the 11g R1 platform, you must use a fresh installation of AIA Foundation Pack 11g Release 1.

For information about how to install AIA Foundation Pack 11g Release 1, see "Understanding the Oracle AIA Foundation Pack Installation Process" in Oracle Fusion Middleware Installation and Upgrade Guide for Oracle Application Integration Architecture Foundation Pack.
2.1.4 Preparing Oracle Enterprise Repository 11g Release 1 Setup for the Migration (Optional)

Oracle Enterprise Repository is an optional component for AIA Foundation Pack and the Process Integration Packs. You can install and use AIA products without Oracle Enterprise Repository. If you want to use Oracle Enterprise Repository to obtain visibility into objects, services, composites, and so forth throughout the SOA lifecycle, you must purchase a Full Use License (FUL) for Oracle Enterprise Repository.

For information about how to install Oracle Enterprise Repository 11g Release 1, see Oracle Fusion Middleware Installation Guide for Oracle Enterprise Repository.

2.1.5 Preparing Oracle Service Registry 11g Release 1 for the Migration (Optional)

Oracle Service Registry is an optional component for AIA Foundation Pack and the Process Integration Packs. You can install and use AIA products without Oracle Service Registry.

For information about how to install Oracle Service Registry 11g Release 1, see "Installation Guide" in Oracle Fusion Middleware Service Registry product documentation at http://download.oracle.com/otndocs/tech/soa/OSR11gR1ProductDocumentation.pdf.

2.2 Before You Begin

The prerequisites for migration of the AIA 2.4 and 2.5 service artifacts are as follows:

- Installation of Oracle Fusion Middleware 11g Release 1, including Oracle SOA Suite and Oracle Database 11g Release 1.
- Installation of AIA Foundation Pack 11g Release 1. As a part of the Foundation Pack, infrastructure services such as Error Handling, Composite Application Validation System (CAVS), Service Constructor, Project Lifecycle Workbench, and the AIA Migration Utility will be installed. Foundation Pack requires a new install and cannot be migrated from earlier versions.
- Access to the AIA artifacts to be migrated.
- All of the service references (adapter services, extension services, infrastructure services, and external web services, for example) should be reachable. In other words, the services should be up and running. If the services are not up and running, a copy of the abstract service WSDLs should be placed in the appropriate 2.4 or 2.5 service folder.
- Ensure AIA 2.4 or 2.5 service projects are compilable, deployable, and executable on the 10g environment prior to migrating them to the 11g environment.
- The 11g version of your custom adapters (binary) should be installed prior to the start of the migration process. For example, the BRM JCA installation should be done after the 11g FMW installation.
- AIA components should be deployed as a .war file on the 11g server.

2.3 Creating and Deploying AIA Components as Web Archive (WAR) Files

The AIA 11g R1 recommendation is to migrate projects independently without needing the 10g server instance up and running. To avoid the dependency on the 10g
Creating and Deploying AIA Components as Web Archive (WAR) Files

server, you must deploy all the library components, namely AIA Components, as a war file onto the WebLogic Server.

To create and deploy AIA components as a .war file:
1. Copy the Application Object Library artifacts and Enterprise Library of the corresponding application to the $AIA_HOME/AIAComponents folder if not already present.

   For example: If you are trying to migrate any provider or requester service in AIADemo in AIA FP 2.5, you may need to copy the ApplicationObjectLibrary/AIADemo and AIADemoEnterpriseBusinessServiceLibrary folders from <$AIA10g_HOME>/samples/AIADemo/htdocs/AIAComponents to <$AIA11g_HOME>/AIAComponents.

   Then copy the entire A10g library under AIAComponents into a local folder.

   For example:
   - For Windows: c:\Metadata
   - For Linux: /slot/ems1739/oracle/Metadata

2. Create a subfolder named WEB-INF under c:\Metadata for Windows, or /slot/ems1739/oracle/Metadata for Linux. For example:
   - C:\Metadata\WEB-INF for windows
   - /slot/ems1739/oracle/Metadata/WEB-INF for linux

3. Create a file named web.xml in the folder WEB-INF.

4. Open web.xml using a text editor, add the following content, and save.

   ```xml
   <web-app xmlns:j2ee="http://java.sun.com/xml/ns/j2ee"></web-app>
   ```

5. Create a file named weblogic.xml in the folder WEB-INF.

6. Open weblogic.xml using a text editor, add the content shown in Example 2–1, and save.

   **Example 2–1 Editing weblogic.xml**

   ```xml
   <?xml version="1.0" encoding="ISO-8859-1"?>
   <weblogic-web-app xmlns="http://www.bea.com/ns/weblogic/90">
   <container-descriptor>
   <index-directory-enabled>true</index-directory-enabled>
   </container-descriptor>
   <context-root>/</context-root>
   </weblogic-web-app>
   ```

7. Create a file named build.xml in the folder c:\Metadata for Windows, or /slot/ems1739/oracle/Metadata for Linux.

8. Open build.xml using a text editor and add the content shown in Example 2–2. Replace the content in square brackets [] with the values relevant to your environment and save.

   Provide the values for the bold attributes.

   **Example 2–2 Editing build.xml**

   ```xml
   <?xml version="1.0" encoding="UTF-8" ?>
   <project name="AIAComponentsAsWAR" default="copyAIAComponents" basedir="." >
   ```
Creating and Deploying AIA Components as Web Archive (WAR) Files

Preparing to Migrate AIA Artifacts to AIA 11g Release 1

9. Set environment variable **ANT_HOME** to **JDEV install folder on your local machine\jdeveloper\ant**.

10. Set environment variable **PATH** to **$ANT_HOME\bin**.

   **Tip:** Ensure that the admin and SOA servers are up and the Java path is set correctly before running the ant command. Without this configuration, the deployment may fail.

11. In the command prompt window, change to directory to **c:\Metadata (for Linux /slot/ems1739/oracle/Metadata)**.

12. Run the ANT build file by executing the ant command. This builds and deploys AIAComponents as a .war file to the 11g Release 1 server.

   For example:
   - For Windows
     C:\Metadata>ant
   - For Linux
     -bash-3.00$ cd /slot/ems1739/oracle/Metadata
     -bash-3.00$ ant
This chapter discusses how to migrate Enterprise Object Library (EOL) customizations from Oracle Application Integration Architecture (AIA) 2.4 or 2.5 to AIA 11g Release 1 using the EOL Upgrade Utility.

This chapter includes the following sections:

- **EOL Migration Prerequisites**
- **Performing the Migration**
- **Running Schema Validations**
- **Running XML Instance Validations**

Use of the EOL Upgrade Utility requires that you have customized the EOL using only the `Custom*.xsd` schema modules located in the following subdirectories of `<AIA_HOME directory>/AIAComponents/EnterpriseObjectLibrary`:

- Core/Custom
- Industry/<IndustryName>/Custom
- Infrastructure/Custom

The purpose of the EOL Upgrade Utility is to migrate additions made to the `Custom*.xsd` schema modules from the old customized 2.4 or 2.5 AIA release installation (source) to the new 11g Release 1 AIA release installation (target).

### 3.1 EOL Migration Prerequisites

Read and understand the contents of this chapter completely. Understand the tasks involved in the migration process before starting the migration.

Make a back up copy of your target AIA 11g Release 1 EnterpriseObjectLibrary folder. This folder is located in `<AIA_HOME directory>/AIAMetaData/AIAComponents`.

### 3.2 Performing the Migration

This section includes the following topics:

- **Section 3.2.1**, "Step 1: Copy Customized EnterpriseObjectLibrary Directory from the Old AIA Release Installation"
- **Section 3.2.2**, "Step 2: Copy Uncustomized Old EnterpriseObjectLibrary Directory from the Old AIA Release Installation"
3.2.1 Step 1: Copy Customized EnterpriseObjectLibrary Directory from the Old AIA Release Installation

Copy contents of the customized old EnterpriseObjectLibrary directory located in `<AIA_HOME directory>/AIAComponents/EnterpriseObjectLibrary` to a temporary directory within the new AIA release installation.

3.2.2 Step 2: Copy Uncustomized Old EnterpriseObjectLibrary Directory from the Old AIA Release Installation

Copy contents of the uncustomized old EnterpriseObjectLibrary to a temporary directory accessible from the target AIA 11g Release 1 installation.

Tip: If a backup of the uncustomized EnterpriseObjectLibrary prior to customization is not available then you can obtain it from Oracle eDelivery, AIA Media Pack. Extract EnterpriseObjectLibrary from the downloaded installable, and obtain file named filegroup1.jar present at `<zip file extract>/cd/Disk1/stage/Components/oracle.aia.fp.base/2.5.0.0.0/1/DataFiles`. Unzip filegroup1.jar and find EnterpriseObjectLibrary at `<extract of filegroup.jar>/AIAComponents/EnterpriseObjectLibrary`.

3.2.3 Step 3: Grant Write Permission to the EnterpriseObjectLibrary Directory in the New AIA Release Installation

Grant `write` permission to files in the `<AIA_HOME directory>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary` directory in the new AIA release installation. The migration process must be able to modify the Custom*.xsd files present in this directory.

3.2.4 Step 4: Execute the EOL Upgrade Utility

The EOL Upgrade Utility migrates each Custom*.xsd file from the old AIA release installation to the corresponding Custom*.xsd file in `<AIA_HOME directory>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary`.

All allowed XML schema specification-compliant changes made to the Custom*.xsd schema module from the old AIA release are copied over to the corresponding Custom*.xsd schema module in the new AIA release installation.

This includes:
- Customizations to Custom*Type complexType
- Any additional global element, simpleType, complexType, group, and attributeGroup definitions in the old Custom*.xsd
- Any additional includes and imports
Any additional namespace declarations

Any additional annotations and comments

The steps performed during the migration of each file are logged in the following output files, which are generated in a location you specify when you run the utility.

- EOLMergeUtility_AutoLog_<TimeStamp>.log
  Captures all automated (trivial) merges.

- EOLMergeUtility_ManualActionLog_<TimeStamp>.log
  Captures all conflicts and the manual actions the user must perform.

- EOLMergeUtility_DetailedLog_<TimeStamp>.log
  Captures a detailed merge report, including post-merge validations.

Output logs are created with timestamps. Each time the utility is run, a new set of log files are created.

Every trivial merge performed by the utility is prefixed with [AUTOMATED] in the EOLMergeUtility_AutoLog_<TimeStamp> log file.

If the EOL Upgrade Utility encounters any conflicts that it cannot automatically resolve, the merge encountering the conflict is prefixed with [MANUAL_ACTION_NEEDED] in the EOLMergeUtility_ManualActionLog_<TimeStamp> log file.

We recommend that you run the EOL Upgrade Utility in a two-step process.

1. Run the utility in Report mode, as shown in Example 3–1 and Example 3–2.

   In this mode, the utility does not perform the actual merge, but it does generate output logs that reflect the results of the merge, has the utility been run in Upgrade mode.

   **Example 3–1 EOL Upgrade Utility Run in Report Mode for Linux, Solaris SPARC, IBM AIX, or HP-UX**

   ```bash
   source <AIA_HOME directory>/aia_instances/<aia_instance_name>/bin/aiaenv.bat
   ```

   **Example 3–2 EOL Upgrade Utility Run in Report Mode for Windows**

   ```bat
   source <AIA_HOME directory>/aia_instances/<aia_instance_name>/bin/aiaenv.bat
   java -classpath %AIA_HOME%/lib/EOLUpgradeUtility.jar;%MW_HOME%/oracle_common\modules\oracle.xdk_11.1.0\xmlparserv2.jar oracle.apps.aia.core.extensiontracer.parser.CustomExtnHandler <Customized Old EnterpriseObjectLibrary Directory> %AIA_HOME%/AIMetaData/AIAComponents/EnterpriseObjectLibrary 'report' 'new' <logFileLocation> <Un-customized Old EnterpriseObjectLibrary Directory>
   ```

   We recommend that you perform all of the changes suggested in the EOLMergeUtility_ManualActionLog before you perform the actual merge by running the utility in Upgrade mode.

   For information about handling items listed in EOLMergeUtility_ManualActionLog, see Section 3.2.5, "Step 5: Review the Output File."
2. Run the utility in Upgrade mode, as shown in Example 3–3 and Example 3–4.

   In this mode, the actual merge process is executed. If in the earlier step you performed the changes suggested in the EOLMergeUtility_ManualActionLog, this log should now be empty, indicating that the merge has completed successfully.

   **Example 3–3  EOL Upgrade Utility Run in Upgrade Mode for Linux, Solaris SPARC, IBM AIX, or HP-UX**

   ```
   source /AIA_HOME/directory/aia_instances/<aia_instance_name>/bin/aiaenv.sh
   ```

   **Example 3–4  EOL Upgrade Utility Run in Upgrade Mode for Windows**

   ```
   source /AIA_HOME/directory\aia_instances\<aia_instance_name>\bin\aiaenv.bat
   java -classpath %AIA_HOME%\lib\EOL UpgradeUtility.jar:%MW_HOME%\oracle_common\modules\oracle.xdk_11.1.0\xmlparserv2.jar oracle.apps.aia.core.extensiontracer.parser.CustomExtnHandler <Customized Old EnterpriseObjectLibrary Directory> %AIA_HOME%\AIMetaData\AIAComponents\EnterpriseObjectLibrary 'upgrade' 'new' <logFileLocation> <Un-customized Old EnterpriseObjectLibrary Directory>
   ```

   Here are descriptions of the variables that you need to supply:

   - **<Customized Old EnterpriseObjectLibrary Directory>**
     Provide the path to the copy of the customized EnterpriseObjectLibrary directory from the old AIA release installation that you created in step 1.

   - **<Un-customized Old EnterpriseObjectLibrary Directory>**
     Provide the path to the copy of the un-customized EnterpriseObjectLibrary directory from the old AIA release installation that you created in step 2.

   - **<logFileLocation>**
     Provide the location in which you want the process log file to be generated.

   **Example 3–5** provides code with all variables populated.

   **Example 3–5  Sample Run of EOL Upgrade Utility in Upgrade Mode for Linux, Solaris SPARC, IBM AIX, or HP-UX**

   ```
   source /slot6035/AIAHOME/aia_instances/devInstance/bin/aiaenv.sh
   java -cp $AIA_HOME/lib/EOL UpgradeUtility.jar:$MW_HOME/oracle_common/modules/oracle.xdk_11.1.0/xmlparserv2.jar oracle.apps.aia.core.extensiontracer.parser.CustomExtnHandler 2.5EOL2/EnterpriseObjectLibrary PS3EOL2/EnterpriseObjectLibrary 'upgrade' 'new' $AIA_HOME/EOLUpgrade/logs 2.5EOL2Orig/EnterpriseObjectLibrary
   ```

3.2.5 Step 5: Review the Output File

   This section includes the following topics:

   - Section 3.2.5.1, "Scenario 1: Namespace Declaration Conflicts"
Section 3.2.5.2, "Scenario 2: complexType Name Conflicts"

Review the output file, EOLMergeUtility_ManualActionLog, created by the EOL Upgrade Utility. Resolve any conflicts that require manual resolution. These conflicts are prefixed with [MANUAL_ACTION_NEEDED] in the output file.

Following are two scenarios in which conflicts may arise.

3.2.5.1 Scenario 1: Namespace Declaration Conflicts

The customized old Custom*.xsd module has a namespace declaration that conflicts with a namespace declaration used in the new Custom*.xsd.

For example, assume that the AIA Foundation Pack 2.5 CustomCommonComponents.xsd file has been customized by adding a namespace declaration:

xmlns:corecomcust="http://xmlns.mycompany.com/common"

The AIA Foundation Pack 11g Release 1 CustomCommonComponents.xsd file has an additional namespace declaration:

xmlns:corecomcust="http://xmlns.oracle.com/EnterpriseObjects/Core/Custom/Common/V2"

The namespace declarations are in conflict. When the EOL Upgrade Utility encounters this conflict, it generates an output message similar to the following:

--> Prefix being used needs to be changed for successful merge.

To resolve this conflict, make the following changes to the migrated file in <AIA_HOME directory>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/ in the new AIA release installation:

1. Change the namespace prefix in the customized namespace declaration to a unique name.

   In this specific case, in CustomCommonComponents.xsd in the new AIA release installation, change the namespace prefix associated with the URI "http://xmlns.mycompany.com/common" to a unique name, such as "mycorecomcust."

2. Change all occurrences of the namespace prefix to the new name.

   In this specific case, change all custom occurrences of the prefix "corecomcust" in the CustomCommonComponents.xsd in the new AIA release installation to "mycorecomcust."

3. Add the namespace prefix mentioned in the output log to the xsd:schema element in the file.

   In this specific case, add the namespace declaration mentioned in the output log to the xsd:schema element of CustomCommonComponents.xsd in the new AIA release installation. For example:

   xmlns:corecomcust="http://xmlns.oracle.com/EnterpriseObjects/Core/Custom/Common/V2"
3.2.5.2 Scenario 2: complexType Name Conflicts
The old Custom*.xsd module has been customized by adding a new global complex type definition that has the same name as a new complexType definition delivered in the corresponding Custom*.xsd file in the new AIA release installation.

For example, assume that the AIA Foundation Pack 2.5 CustomCommonComponents.xsd has been customized with a new complex type definition named "CustomFinancialAccountReferenceType."

The new AIA release installation delivers CustomCommonComponents.xsd with a new complex type definition with the same name, "CustomFinancialAccountReferenceType."

There is a naming conflict with the custom type in the customized AIA Foundation Pack 2.5 EnterpriseObjectLibrary. When the EOL Upgrade Utility encounters this conflict, it generates an output message similar to the following:

[MANUAL_ACTION_NEEDED] Duplicate CustomXXType Notification ::
CustomFinancialAccountReferenceType is being used in the latest EOL.
-->  CustomType name being used needs to be changed for successful merge

There are two methods for resolving this conflict. Select the method that is appropriate for your implementation and programming model.

1. **Method 1**: Make no changes and retain the custom type definition from the EOL in the old AIA release installation. If you decide to implement this approach, the customized content will appear in places in which it has not appeared in the past. For example, the customized content will appear in the content delivered in the new AIA release installation. This side effect can invalidate XML instances or implementations if any of the customized content includes constraints, such as a minOccurs attribute set to 1. To address this side effect, you must relax these constraints.

2. **Method 2**: Make the following changes to the migrated file in <AIA_HOME directory>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/ in the new AIA release installation:
   a. Change the customized type you added in AIA Foundation Pack 2.5 to a unique name, ensuring that you do not start the name with "Custom."
   b. Modify all occurrences of this customized type.
   c. Add an empty type definition corresponding to what has been delivered in the new EOL custom schema module to the old EOL custom schema module. You can copy the empty definition from the backup copy of <AIA_HOME directory>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary that you created as a prerequisite.

3.2.6 Step 6: Handling Imports and Includes of Non-AIA-Delivered Schema Modules
You may have customized the Custom*.xsd schema modules by adding additional imports or includes of schema modules that have not been delivered in the EOL.

For example, your implementation, "MyCompany," may have a central XSD components library named "MyCompanyComponents.xsd," which was not delivered as a part of AIA. The custom schema modules in the EnterpriseObjectLibrary from the old AIA release installation may include an import of the above file.

Ensure that the references to these non-AIA-delivered schema modules are accessible from the migrated EnterpriseObjectLibrary in the new AIA release installation.
If relative URLs are used in the schemaLocation attribute while importing or including non-AIA-delivered schema modules, you must upload these schema modules into the Oracle Metadata Service repository (MDS).

If absolute URLs are used in the schemaLocation attribute while importing or including non-AIA-delivered schema modules, ensure that these absolute URLs are accessible from the new AIA release installation.

3.3 Running Schema Validations

At this point, the EnterpriseObjectLibrary of the new AIA release installation has been updated with customized content migrated from the EnterpriseObjectLibrary from the old AIA release installation.

You must verify that the customizations made to Custom*.xsd files in the EnterpriseObjectLibrary from the old AIA release installation were migrated to the corresponding files in the EnterpriseObjectLibrary in <AIA_HOME>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary in the new AIA release installation.

To do this, run schema validations. AIA recommends that you run schema validations for the entire schema library in the new AIA release installation.

You can use any XML schema validation tool or API to validate the contents of the EnterpriseObjectLibrary. For example, you can use the Oracle JDeveloper Audit feature to validate the entire Enterprise Object Library.

To use the Oracle JDeveloper Audit feature:


2. Make a local copy of the migrated EnterpriseObjectLibrary folder present at <AIA_HOME>/AIAMetaData /AIAComponents / of the new AIA release installation.

3. To avoid HTML from being validated in the Oracle JDeveloper validation, you can delete the all .html, .css, and .txt files from the local copy of the EnterpriseObjectLibrary folder used for validation.

4. Edit the project properties of the newly created SOA project and add the local copy created previously to the Java Source Paths: list. Figure 3–1 shows that the contents of the EnterpriseObjectLibrary are now included in the project.
5. Navigate to the **Application Navigator** window. Select the **Application Sources** folder under the project name by clicking on it as shown in **Figure 3–2**.

**Figure 3–2  Application Navigator**
6. Keep the Application Sources folder highlighted and navigate to Build, Audit Application Sources as shown in Figure 3–3.

**Figure 3–3  Audit Application Sources**

7. In the Audit Application Sources window, choose the default profile Audit Rules shown in Figure 3–4 and click Run.

**Figure 3–4  Run Audit Application Sources**

8. The Audit Log window shown in Figure 3–5 displays the status of the audit. The number of files audited is displayed in the report title. Any issues found in the audit are displayed below, grouped by the file in which the issues were found.

**Figure 3–5  Audit Log**

9. If there are any issues, correct them, and redo the migration, if necessary. You can ignore errors related to HTML files.
3.4 Running XML Instance Validations

AIA recommends that you take sample XML instances that were created based on the old AIA release installation and run XML instance validations against the migrated version in the new AIA release installation, particularly for customized Enterprise Business Objects (EBOs).

You can skip this validation if you are going to put your migrated content through integration testing with schema validation turned On.

You can use any XML schema tool or API to validate the sample XML instance against its corresponding XSD schema.

**To use Oracle JDeveloper 11g to validate an XML instance against its corresponding XSD schema:**

1. Copy the sample XML instance file based on the old AIA release installation to the project directory of a new or existing Oracle JDeveloper project as shown in Figure 3–6.

**Figure 3–6 Sample XML Instance File Copied to the Project Directory**

2. Open the project in Oracle JDeveloper and refresh the Application Navigator. Figure 3–7 shows the sample XML instance file displayed in the Application Navigator. If the Application Navigator pane is not visible, navigation to View, Application Navigator.
3. Double-click to open the sample XML instance. Add or modify the following information as appropriate to the root element of the XML instance, if it is not already present.

- Namespace prefix declarations, such as the prefix of the EBO namespace.
  For example:
  ```xml
  xmlns:coreinvoice=<namespace of the XML instance or targetNamespace of the schema>
  xmlns:coreinvoice="http://xmlns.oracle.com/EnterpriseObjects/Core/EBO/Invoice/V2"
  ```

- Declaration of the XMLSchema-instance namespace prefix.
  For example:
  ```xml
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  ```

- The XMLSchema-instance schemaLocation attribute.
  For example:
  ```xml
  xsi:schemaLocation="<namespace> <physical-schema location>"
  ```

  The physical schema location should point to the .xsd file in the new AIA release installation:
  ```xml
  ```

4. Save any edits you made.

5. In the **Application Navigator**, right-click the sample XML file and select **Validate**.
6. The XML Validation Errors - Log displays the validation results. If any issues are logged, correct them, and redo the migration, if necessary.
This chapter discusses how to migrate new Enterprise Business Objects (EBOs) you created and added to the EBO Library.

To migrate new EBOs:
1. Copy the new EBO(s) to the respective library folder in `<AIA_HOME directory>/AIAMetaData/AIAComponents/EnterpriseObjectLibrary`.
2. Update the EBO Library in the Oracle Metadata Services (MDS) repository.
   a. Browse to the folder at `$AIA_HOME/aia_instances/$INSTANCE_NAME/bin` and open the deployment plan file, `UpdateMetaDataDP.xml`.
   b. Source the file `aiaenv.sh`:

```
source aiaenv.sh
```
   c. Update the file `UpdateMetaDataDP.xml`, by inserting include tag(s), for each resource group that you want to add to the MDS.

```
<include name ="AIAComponents/EnterpriseObjectLibrary/<respective folders/>">
```
   d. Browse to `AIA_HOME/Infrastructure/Install/scripts`. Execute the script `UpdateMetaData.xml`:

```
ant -f $AIA_HOME/Infrastructure/Install/AID/AIAInstallDriver.xml -DDeploymentPlan=<path of UpdateMetaDataDP> -DPropertiesFile= $AIA_HOME/aia_instances/<AIA_INSTANCE>/config/AIAInstallProperties.xml
```
This chapter discusses how to migrate new Enterprise Business Service (EBS) definitions you created and added to the EBS Library.

**To migrate new EBSs:**

1. Copy the new EBS definition(s) to the respective library folder in `<AIA_HOME directory>/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary`.

2. Update the EBS Library in the Oracle Metadata Services (MDS) repository.
   a. Browse to the folder at `$AIA_HOME/aia_instances/$INSTANCE_NAME/bin` and open the deployment plan file, `UpdateMetaDataDP.xml`.
   b. Source the file `aiaenv.sh`:
      ```
      source aiaenv.sh
      ```
   c. Update the file `UpdateMetaDataDP.xml` by inserting include tag(s), for each resource group that you want to add to the MDS.
      ```xml
      <include name="AIAComponents/EnterpriseBusinessServiceLibrary/<respectivefolders/>">
      ```
   d. Browse to `AIA_HOME/Infrastructure/Install/scripts`. Execute the script `UpdateMetaData.xml`:
      ```
      ant -f $AIA_HOME/Infrastructure/Install/AID/AIAInstallDriver.xml 
      -DDeploymentPlan=<path of UpdateMetaDataDP> -DPropertiesFile= 
      $AIA_HOME/aia_instances/<AIA_INSTANCE>/config/AIAInstallProperties.xml. 
      ```
This chapter discusses how to migrate the metadata associated with Oracle Application Integration Architecture (AIA) Domain Value Maps (DVMs) and Cross-References.

When you install AIA Foundation Pack 11g Release 1, cross-reference and DVM metadata setup is part of the installation. If you have extended or customized cross-references and DVMs, follow the procedures covered in this chapter to migrate these artifacts.

Migrating DVMs and cross-references is a two-step process. Migration involves migrating both metadata and actual data. The DVMs have metadata, as well as static data. For cross-references, the data is dynamic and is stored in a database.

This chapter discusses migrating DVM and cross-reference metadata.

For information about how to migrate actual DVM and cross-reference data, see Chapter 14, "Migrating Actual Domain Value Map and Cross-Reference Data."

### 6.1 Migrating DVM and Cross-Reference Metadata

DVM and cross-reference metadata are the definitions of the DVM and cross-reference objects. The definitions have changed in 11g.

**To migrate DVM and cross-reference metadata:**

Run the AIA Migration Utility.

Input and output directories can be in any location, however it is preferable to create the directories in the root directory of the AIA Migration Utility.

- **For DVMs:**
  ```
  ant -f AIAMigrationUtility.xml upgradeDVM -DinputDir=<path> -DoutputDir=<path>
  ```

- **For cross-references:**
  ```
  ant -f AIAMigrationUtility.xml upgradeXRef -DinputDir=<path> -DoutputDir=<path>
  ```

For more information, see Chapter 18, "Loading AIA Metadata to Oracle Metadata Services Repository."

Migrating Domain Value Map and Cross-Reference Metadata
This chapter discusses the steps needed to migrate the services built leveraging Oracle 10g SOA and AIA technologies.

This chapter includes the following sections:

- Overview of AIA Service Migration Using the AIA Migration Utility
- How to Migrate Adapter Services
- How to Migrate Provider ABCSs
- How to Migrate EBSs
- How to Migrate Requester ABCSs

### 7.1 Overview of AIA Service Migration Using the AIA Migration Utility

Migration of AIA Foundation Pack 2.4 and 2.5 service artifacts to AIA Foundation Pack 11g Release 1 artifacts facilitates the easy adoption of the component-based architecture of Oracle Fusion Middleware 11g Release 1 for AIA solution implementation.

Figure 7–1 depicts the migration process.
Table 7–1 provides a list of AIA 2.4 and 2.5 artifacts and their equivalent AIA Foundation Pack 11g Release 1 artifacts.

<table>
<thead>
<tr>
<th>AIA 2.4 and 2.5 Artifacts</th>
<th>Technology Implementation Used in FMW 10g</th>
<th>Technology Implementation Used in FMW 11g</th>
<th>AIA Foundation Pack 11g Release 1 Artifact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requester Application Business Connector Service (ABCS)</td>
<td>BPEL</td>
<td>BPEL</td>
<td>Requester ABCS BPEL Composite</td>
</tr>
<tr>
<td>Enterprise Business Service (EBS)</td>
<td>Enterprise Service Bus (ESB)</td>
<td>Mediator</td>
<td>EBS Mediator Composite</td>
</tr>
<tr>
<td>Provider ABCS</td>
<td>BPEL</td>
<td>BPEL</td>
<td>Provider ABCS BPEL Composite</td>
</tr>
<tr>
<td>Sub-BPEL Services for an ABCS</td>
<td>BPEL</td>
<td>BPEL</td>
<td>Service Component in an ABCS Composite</td>
</tr>
<tr>
<td>Adapter Services</td>
<td>ESB/BPEL</td>
<td>Mediator/BPEL</td>
<td>Adapter Service Composite</td>
</tr>
<tr>
<td>Enterprise Business Flow (EBF)</td>
<td>BPEL</td>
<td>BPEL</td>
<td>EBF Composite</td>
</tr>
</tbody>
</table>

7.2 How to Migrate Adapter Services

To migrate adapter services:
1. Complete the pre-migration tasks.
   - Change the references in the adapter WSDL, if there are any changes to the JNDI references. Change the value of the `jca:address location` attribute value in the service section.
     For example: `jca:address location="eis/DB/AIASamplesDB`
   - Post migration if you want to view the adapter using 11g JDeveloper, create the respective UI Connection with the value present in the adapter WSDL service section.
     For example: `UIConnectionName="AIASamplesDB"
     Ensure that the service builds in 10g.
2. Run the AIA Migration Tool.
   In the `AIAMigrationUtility.Properties` file, provide the value for the `process.type` attribute.
   For example: `process.type= DBAdapter`
   For more information, see Chapter 17, “Using the AIA Migration Utility.”
3. Complete the post-migration tasks.
   a. Annotate the Adapter Service composite.
      For more information about annotation composites for harvesting, see "Annotating Composites" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.
   b. Configure `adf-config.xml` to access MDS store.
      For more information, see Section 18.1, "Accessing MDS from the AIA Service Artifact."
For information, see Chapter 12, "Making Migrated Services Deployable."

7.3 How to Migrate Provider ABCSs

To migrate Provider ABCS:

1. Complete the pre-migration tasks.
   a. Verify the service builds in 10g.
   b. Ensure that all WSDLs are reachable on the 11g server or from the file system.
      If the application service WSDLs are not up and running, you should copy them to the Provider ABCS folder and change the reference accordingly in bpel.xml as shown in Example 7–1.

Example 7–1 Application Service WSDL Reference in bpel.xml

```xml
<partnerLinkBinding name="SamplesCreateCustomerPartyPortalProvider">
  <property name="wsdlLocation">SamplesCreateCustomerPartyPortalProvider.wsdl</property> (this wsdl is a local reference)
</partnerLinkBinding>
```

c. Comment out the `wsdlRuntimeLocation` in the bpel.xml as shown in Example 7–2.

Example 7–2 wsdlRuntimeLocation in bpel.xml

```xml
<partnerLinkBinding>
  <!--
  -->
</partnerLinkBinding>
```

2. Run the AIA Migration Tool.
   For more information, see Chapter 17, "Using the AIA Migration Utility."

   In the AIAMigrationUtility.Properties file, provide the value for the process.type attribute.

   For example: process.type= Provider

3. Complete the post-migration tasks.
   a. Create XSD and XSL folders in the project folder if there are any XSD, XSL files. Then move XSD and XSL files to their respective folders in the project.
      After moving the XSDs and XSLs to their respective folders, change the references in BPEL, WSDL, and XSL files.
   b. Configure the abstract WSDLs.
      For more information, see Section 18.2, "Moving Abstract Service WSDLs into MDS."
   c. Configure fault policies.
      For more information, see Section 10.1, "How to Migrate Fault Policies."
   d. Specify the transaction-related properties.
      For more information, see "Enabling Transactions" in Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack.
e. Complete the annotations in Composite.xml. This can be performed after unit testing, but must be completed prior to publishing.

For more information about annotation composites for harvesting, see "Annotating Composites" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

f. Change any remaining 10g artifact (WSDL files) locations (Host/Port) to 11g artifact locations.

g. The function `ora:getUserProperty` was deprecated in 10.1.3.4 and the replacement is `ids:getUserProperty`. In 10.1.3.4 JDeveloper, `ora:getUserProperty` is under "Deprecated Identity Service Functions". So in 11g, you need to use `ids:getUserProperty`.

h. Configure `adf-config.xml` to access the MDS store.

For more information, see Section 18.1, "Accessing MDS from the AIA Service Artifact."

For information, see Chapter 12, "Making Migrated Services Deployable."

7.4 How to Migrate EBSs

**To migrate EBSs:**

1. Complete the pre-migration tasks.
   a. Ensure that all WSDLs are reachable on 11g or from the file system.
   b. Browse through all `.esbsvc` files and remove the service and system group information from the `qname` attribute of the Service and `operationInfo` tags. For example:

   **Before change:**
   ```xml
   <service name="CustomerPartyEBS" guid="88D6459098C711DDBF83C6AC4E8A30F
   " qname="AIASamples.CustomerPartyEBS" status="ENABLED"
   serviceType="RoutingService" typeDescription="Routing Service"
   isWSDLEditable="false" soapEndpointStatus="ENABLED">
   </service>
   ```

   **After change:**
   ```xml
   <service name="CustomerPartyEBS" guid="88D6459098C711DDBF83C6AC4E8A30F
   " qname="CustomerPartyEBS" status="ENABLED" serviceType="RoutingService"
   typeDescription="Routing Service" isWSDLEditable="false"
   soapEndpointStatus="ENABLED">
   ```

   c. Change the service references in `.esbsvc` files (wsdlURL).

   If the WSDLs are not reachable then copy the reference WSDLs to the local folder and change the value of the `wsdlURL` attribute in the `.esbsvc` file.

   The `http reference` of the service must be changed to the local folder reference. For example:

   **Before change:**
   ```xml
   <wsdlURL>http://ap6014rems.example.com:7839/AIAComponents/
   EnterpriseBusinessServiceLibrary/Core/EBO/CustomerParty/V2/
   CustomerPartyEBS.wsd1</wsdlURL>
   ```
The assumption is this reference WSDL is not accessible on the web service, hence the user needs to copy the WSDLs to local folder and change the references.

After change:

```xml
<wSDLURL>CustomerPartyEBS.wsdl</wSDLURL>
```

2. Run the AIA Migration Utility.

**Note:** In the AIAMigrationUtility.Properties file provide the value for the process.type attribute.

For example: `process.type=EBS`

For more information, see Section 17.1, "Overview of the AIA Migration Utility."

3. Complete post-migration tasks.
   a. If filter expressions have DVM and cross-reference calls, those filter expressions must be manually migrated. For example:

   **Example 7–3** provides an example of a filter expression before migration:

   ```xml
   Example 7–3 Filter Expression Before Migration
   ``

   **Example 7–4** provides an example of a filter express after migration:

   ```xml
   Example 7–4 Filter Expression After Migration
   ``

   b. Configure the concrete URLs in the **composite.xml binding.ws section**:

   ```xml
   <binding.ws port="NAMESPACE OF THE REFERENCE SERVICE#wsdl.endpoint
   (Servicename Of The Reference Service/Port Name Of The Reference Service)
   * location="ENDPOINT LOCATION OF THE REFERENCE SERVICE"/>
   ```
Example 7–5 shows concrete URL populated in the composite.xml binding.ws section.

**Example 7–5  Concrete URL in the composite.xml binding.ws Section**

```
<reference ui:wsdlLocation="CustomerPartyEBSRef1.wsdl"
        name="SamplesCustomerPartyEBS">

  ..
  <binding.ws port="http://xmlns.oracle.com/EnterpriseServices/Core/CustomerParty/
              V2#wsdl.endpoint(SamplesCustomerPartyEBS_ep/CustomerPartyEBS_pt)"
              location="http://sdc60024sems.example.com:8097/soa-infra/services/default/
              SamplesCustomerPartyEBS/SamplesCustomerPartyEBS_ep?WSDL"/>

</reference>
```

c. Configure `adf-config.xml` to access the MDS store.

For more information, see Section 18.1, "Accessing MDS from the AIA Service Artifact."

d. Add an explicit assign statement to the routing rule when the part names for the source service and target service from the mediator are different.

Example 7–6 and Example 7–7 illustrate a case in which source service part names are different. In this case, an explicit transformation or assign must be added in the routing rule.

**Example 7–6  Part Name for the Source Service**

```
<message name="CreateCustomerPartyListReqMsg">
  <documentation>
    <svcdoc:Message>
      <svcdoc:Description>This message is used as the request payload to the CreateCustomerPartyList operation</svcdoc:Description>
    </svcdoc:Message>
  </documentation>
  <part name="CreateCustomerPartyListEBM" element="customerpartyeo:
         CreateCustomerPartyListEBM"/>
</message>
```

**Example 7–7  Part Name for the Target Service**

```
<message name="CreateCustomerPartyListReqMsg">
  <documentation>
    <svcdoc:Message>
      <svcdoc:Description>This message is used for sending payload to Create Customer operation</svcdoc:Description>
    </svcdoc:Message>
  </documentation>
  <part name="CreateCustomerPartyList" element="aiasamples:
         CreateCustomerPartyListEBM"/>
</message>
```

e. The CAVS programming model will work as-is for migrated requester and provider ABCSs. However, you should perform the following changes:

Point the CAVS endpoint in the composite.xml and component type files to the abstract WSDL, as shown in Example 7–8 (composite.xml) and Example 7–9 (component type file).
Example 7–8  Pointing the CAVS Endpoint to the Abstract WSDL in composite.xml

```xml
<referenceur:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/Infrastructure
ServiceLibrary/V1/wsdl/AsyncRequestRecipient.wsdl" name="AsyncRequest
Recipient">  
interface(CAVSAnyPort)="/>
  <binding.ws port="http://xmlns.oracle.com/apps/aia/cavs/Any#wsdl.endpoint
(CAVSAnyService/CAVSAnyPort)="/" location="http://sdc60008sems.example.com:
8088/AIAValidationSystemServlet/asyncrequestrecipient.wsdl="/>
</reference>
```

Example 7–9  Pointing the CAVS Endpoint to the Abstract WSDL in the Component Type
File

```xml
<reference ur:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/
InfrastructureServiceLibrary/V1/wsdl/AsyncRequestRecipient.wsdl " name=
'AsyncRequestRecipient'>  
interface(CAVSAnyPort)="/>
</reference>
```

f. Change filter expressions in the .mplan file, as shown in Example 7–10 and
Example 7–11.

g. If you used XSLT functions to manipulate SOAP headers in Enterprise Service
Bus (ESB) 10g, then after you upgrade your application, those header
manipulations will be modeled as assignments in Oracle Mediator 11g. For
complex header manipulations, you should manually verify the SOAP headers before deploying the upgraded Oracle Mediator 11g projects.

Most of the standard header properties are upgraded to their equivalent 11g headers. Customized headers are upgraded using a best effort approach, but it is not possible to upgrade all the variations of customized headers. As a result, you must manually upgrade any customized headers the equivalent condition or assign element and equivalent mplan artifacts.

For more information, see "Getting Started with Oracle Mediator" in the Oracle Fusion Middleware Developer’s Guide for Oracle SOA Suite and "Upgrading Filtering or Setting Message Headers in Oracle Enterprise Service Bus" in Oracle Fusion Middleware Upgrade Guide for Oracle SOA Suite, WebCenter, and ADF.

For information, see Chapter 12, "Making Migrated Services Deployable."

### 7.5 How to Migrate Requester ABCSs

**To migrate requester ABCSs:**

1. Complete pre-migration tasks.
   a. Ensure that all WSDLs are reachable on 11g or from the file system.
   b. Verify that the service is compiled and deployed successfully on Oracle Fusion Middleware 10g.
   c. Point references for all invoked services to abstract WSDLs in composite.xml, as shown in Example 7–12.

   **Example 7–12 Invoked Service Pointing to abstract WSDL in composite.xml**

   ```xml
   <service ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/ApplicationConnectorServiceLibrary/SampleSEBL/V1/RequestorABCS/SamplesCreateCustomerSiebelReqABCSImpl.wsdl" name="SamplesCreateCustomerSiebelReqABCSImpl">
   ----------------------------------
   </service>
   d. Comment out the wsdlRuntimeLocation in the bpel.xml as shown in Example 7–13.

   **Example 7–13 bpel.xml with wsdlRuntimeLocation Commented Out**

   ```xml
   <partnerLinkBinding name="SamplesCustomerPartyEBS">
   <property name="wsdlLocation">SamplesCustomerPartyEBSRef.wsdl</property>
   <property name="transaction">participate</property>
   <!--property name="wsdlRuntimeLocation">http://ap6060fems.example.com:7817/esb/wsil/AIASamples/CustomerPartyEBS/wsdl</property-->
   </partnerLinkBinding>
   ```

2. Run the AIA Migration Utility.
For example: process.type= Requester
For more information, see Section 17.1, "Overview of the AIA Migration Utility."

3. Complete post-migration tasks.
For information, see post-migration tasks under Section 7.3, "How to Migrate Provider ABCSs."
For information, see Chapter 12, "Making Migrated Services Deployable."

**Note:** In the AIAMigrationUtility.Properties file provide the value for process.type attribute.
This chapter describes how to migrate services developed using BPEL, Oracle Mediator, or other technologies.

This chapter includes the following sections:

- **Migrating Services Developed Using BPEL or Oracle Mediator**
- **Migrating Services Developed Using Other Technologies**

After you have completed the tasks in this chapter and if your extension services have been developed using the Oracle SOA Suite 11g Service Component Architecture, then perform tasks in Chapter 12, "Making Migrated Services Deployable."

### 8.1 Migrating Services Developed Using BPEL or Oracle Mediator

If you have developed services using BPEL or Oracle Mediator, follow these steps:

1. Migrate BPEL or Mediator based extension service to 11g following the steps outlined in Chapter 7, "Migrating Services."
2. After migrating the extension services, unit test them.
3. Deploy the extension service on the Fusion Middleware 11g server.
4. Run the flows to test ABCS with extension hooks.

### 8.2 Migrating Services Developed Using Other Technologies

If you have developed extension services using technologies other than using BPEL or Oracle Mediator, follow these steps:

1. Move the service to 11g and make 11g compatible changes, if required.
2. Unit test the extension service.
3. Deploy the migrated service.
4. Run the flows to test ABCS with extension hooks.
This chapter discusses how to migrate your custom routing rules for Oracle-delivered Enterprise Business Services (EBSs).

If you have developed custom routing rules, then you need to manually migrate these filter expressions.

When upgrading Oracle Enterprise Service Bus projects, each routing rule is upgraded into case, and the filter expression is converted to condition. These changes can be found in the .mplan file of the Routing Service.

AIA recommends that you re-create the routing rules using Oracle JDeveloper.

**To re-create routing rules:**

1. Open Oracle JDeveloper and import the Mediator project into the application.
2. Open the .mplan file from Oracle JDeveloper.
3. Select the operation where you want to add the routing rule.
4. Select the + symbol and create the static routing rule.
5. Redeploy the EBS project.
This chapter describes how to migrate and configure Oracle AIA Error Handling artifacts.

You may need to migrate fault policies or error handling services developed using AIA 2.4 or 2.5. Since there is no utility to support migration of error handling artifacts, this task must be performed manually.

For information about error handling, see "Introduction to Oracle AIA Error Handling" in Oracle Fusion Middleware Infrastructure Components and Utilities User’s Guide for Oracle Application Integration Architecture Foundation Pack.

This chapter includes the following sections:

- How to Migrate Fault Policies
- How to Configure Transaction Configurations
- How to Configure AIAConfigurationProperties.xml for Error Handling
- How to Migrate the Error Notification Setup
- How to Configure the Oracle User Messaging Service for Email Notifications
- How to Configure User Email Addresses
- How to Migrate Custom Error Handlers
- How to Migrate Custom Subscribers

### 10.1 How to Migrate Fault Policies

AIA Foundation Pack provides a default fault policy that is stored in MDS in the folder AIAMetaData/faultPolicies/V1. When default fault policies are used in a process, you must add the following elements to the composite.xml file:

```
<property name="oracle.composite.faultPolicyFile">[pointer to the fault policy xml file in the MDS]</property>
<property name="oracle.composite.faultBindingFile">[pointer to the fault policy bindings file fault-bindings.xml in the MDS]</property>
```

If custom fault policies are used in AIA 2.4 and 2.5, then follow these steps to update them to AIA Foundation Pack 11g Release 1.

1. Consider a sample custom fault policy file. Example 10–1 shows how to associate fault policies defined in this sample fault-policy file with a fault-policy binding.xml file. These two files should be placed in the composite process folder.
10.2 How to Configure Transaction Configurations

For information, see "Enabling Transactions" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

10.3 How to Configure AIAConfigurationProperties.xml for Error Handling

In AIA 2.4 and 2.5, and in AIA Foundation Pack 11g Release 1, the AIAConfigurationProperties.xml file contains settings related to AIA Error Handling. Table 10–1 provides a summary of the new settings introduced with AIA Foundation
How to Configure the Oracle User Messaging Service for Email Notifications

10.4 How to Migrate the Error Notification Setup

To migrate the Error Notification Setup from an AIA 2.4 or 2.5 environment to a new AIA Foundation Pack 11g Release 1 environment, you must review all custom notification records and manually duplicate these entries in the new AIA Foundation Pack 11g Release 1 environment. The Error Notification page can be accessed by clicking the Go button in the Setup area of the AIA Home Page.

There is no functional difference for the record attributes that existed in AIA 2.4 and 2.5. The new attributes relate to error notification throttling and default when entering new records. There is no need to change these values if the throttling feature is not required.

For more information about error notification throttling, see "Using Error Notifications" in Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack.

10.5 How to Configure the Oracle User Messaging Service for Email Notifications

To enable and configure email messaging:

1. In Oracle Enterprise Manager, select the SOA infrastructure and open the context menu SOA Administration and Workflow Notification Properties. Set Notification Mode to Email, click Apply, and then click Go to the Messaging Driver page.
2. On the Messaging Driver page, click the **Configure Driver** button for the **User Messaging Email Driver**. Provide a valid value for the **OutgoingMailServer** property, such as mail.oracle.com, and enter any other settings that are relevant for your environment.

3. Restart the SOA server for the changes to take effect.


### 10.6 How to Configure User Email Addresses

**To configure user email addresses:**


2. Log in as the user who is supposed to receive error notifications, such as AIAIntegrationAdmin.

3. Create a new default messaging channel of type **EMAIL**, and assign the email address to which notifications will be sent.

### 10.7 How to Migrate Custom Error Handlers

Similar to AIA 2.4 and 2.5 Error Handling, AIA Foundation Pack 11g Release 1 Error Handling allows you to implement a custom error handler to extend the default capabilities of the AIA Error Handler. For example, it allows you to add further details to the fault payload.

Note a few key changes made to the custom error handler:

- Implements **handleCompositeSystemError** as opposed to **handleBPELSystemError** in AIA 2.4 and 2.5.
- Handles errors for BPEL and Mediator components within composites.
- Distinguishes component types (BPEL and Mediator) and responds accordingly.
- Code changed to provide access to runtime variables. Specifically, see the reference to **getVariableData** in Example 10–3.

For your reference, Example 10–3 provides a code sample from the AIA Foundation Pack Demo available on Oracle Technology Network at http://oracle.com/technetwork.

Note the highlighted area to understand the differences between AIA Foundation Pack 11g Release 1 and 2.4 and 2.5 implementations, and make modifications accordingly to your subscriber code.

**Example 10–3 Sample Code from the AIA Foundation Pack Demo Implementation**

```java
package oracle.apps.aia.demo;
import oracle.integration.platform.faultpolicy.IFaultRecoveryContext;
import com.collaxa.cube.engine.fp.BPELFaultRecoveryContextImpl;
import com.collaxa.cube.engine.types.bpel.CXMessageVariable;
import oracle.apps.aia.core.eh.IAIAErrorHandlerExtension;
import oracle.apps.aia.core.eh.logging.AIALogger;
```
public class AIADemoErrorHandlerExtImpl implements IAIAErrorHandlerExtension
{
    public AIADemoErrorHandlerExtImpl()
    {
    }

    public String handleBusinessError(String faultMessageConstructed)
    {
        return faultMessageConstructed;
    }

    public String handleCompositeSystemError(IFaultRecoveryContext iFaultRecoveryContext, String faultMessageConstructed, String componentType)
    {
        String enrichedFaultMessageConstructed = faultMessageConstructed;

        if (componentType.equals("bpel"))
        {
            try
            {
                BPELFaultRecoveryContextImpl bpelCtx = (BPELFaultRecoveryContextImpl) iFaultRecoveryContext;
                CXMessageVariable bpelVar = (CXMessageVariable) bpelCtx.getVariableData("AIADemoProcess CreditChargeAuthorizationReqMsg");
                if (bpelVar != null)
                {
                    String ebmPayload = bpelVar.toXML();
                    enrichedFaultMessageConstructed = faultMessageConstructed.replaceFirst("<IntermediateMessageHop>",
                    (new StringBuilder()).append("<InputPayload>").append(ebmPayload).append("</InputPayload><IntermediateMessageHop>").toString());
                }
            }
            catch (Exception ex)
            {
                ex.printStackTrace();
                AIALogger.logErrorMessage(null, ex.getLocalizedMessage());
            }
        }
        else if (componentType.equals("mediator") )
        {
            //do your mediator custom error handling..
            //This block gets triggered only in case parallel routing rules are enabled.
            }

        return enrichedFaultMessageConstructed;
    }
}

Detailed Steps
1. Compile your code by including the following jar files in your project or from ant task.

   <AIA_HOME>/lib/aia.jar
How to Migrate Custom Subscribers

2. After you have your jar file constructed, compile the custom classes, for example `AIADemoErrorHandler.jar`.

   Extract the contents to `${mw.home}/Oracle_SOA1/soa/modules/oracle.soa.ext_11.1.1/classes` folder.

3. Restart the SOA managed server for the extension to work.

   Make sure that your AIA Error notification rules have the custom listener stamp set up which is the same as that in 10g.

   Note that the table name changed from BSR_ERROR_NOTIFICATIONS to `AIA_ERROR_NOTIFICATIONS`.

10.8 How to Migrate Custom Subscribers

The migration of custom subscribers to the `AIA_ERROR_TOPIC` that was implemented to pick up fault messages for further processing depends on how the subscriber was implemented in AIA 2.4 and 2.5/Oracle SOA Suite 10g.

Assuming that the subscriber was implemented as either a BPEL- or ESB-based process leveraging a JMS adapter, the usual technical migration applies as for any other AIA service.

It is essential to ensure that the JMS adapter configuration located in the JCA file is adjusted to point to the correct JNDI name of the AIA error topic and the connection factory in AIA Foundation Pack 11g Release 1. Note that these names have changed since AIA 2.4 and 2.5. See Example 10–4, which is a sample JCA file from the AIA Demo, to understand the relevant properties in bold and their correct settings.

Example 10–4 Sample JCA file from the AIA Demo

```xml
<adapter-config name="AIADemoErrorJMSListener" adapter="Jms Adapter"
xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
  <connection-factory location="eis/jms/aiaErrorTopicCF" UIConnectionName="dev_soa_server" UIJmsProviders="WLSJMS" adapterRef=""/>
  <endpoint-activation portType="Consume_Message_ptt" operation="Consume_Message">
    <activation-spec className="oracle.tip.adapter.jms.inbound.JmsConsumeActivationSpec">
      <property name="DestinationName" value="jms/aia/aiaErrorTopic"/>
      <property name="UseMessageListener" value="false"/>
      <property name="MessageSelector" value="JMSCorrelationID = 'AIADEMO_CREDITCARD_FAILURE'"/>
      <property name="PayloadType" value="TextMessage"/>
    </activation-spec>
  </endpoint-activation>
</adapter-config>
```

In 11g, AIA Error Topic JMS infrastructure is moved from AQ implementation to that of WebLogic JMS (default behavior). WLS JMS can be configured to use any foreign JMS server or use database as persistent point for the JMS resource configured. As delivered, Foundation Pack configures it as file based artifact.

The custom listener stamping logic and enrichment extension handler in 10g is applicable for 11g with the changes mentioned above.
This chapter describes how to migrate any custom Composite Application Validation System (CAVS) definitions.

This chapter includes this section:

- Migrating Custom CAVS Definitions

### 11.1 Migrating Custom CAVS Definitions

**To migrate any custom CAVS definitions:**

1. Log in to the Oracle AIA Console for AIA 2.4 or 2.5.
2. Select the Validation System tab. Click the Definitions link. The Definitions page displays.
3. Select the definitions to migrate and click the **Export** button to export the CAVS definitions into an XML file.
4. Log in to the AIA Home Page for 11g Release 1: http://<host>:<port>/AIA. In the **Composite Application Validation System** area, click the **Go** button.
5. The Definitions page displays. Click the **Import** button and import the CAVS definitions exported in Step 3.

For more information about the export and import features in CAVS, see "Exporting and Importing CAVS Definitions and Instances" in *Oracle Fusion Middleware Infrastructure Components and Utilities User’s Guide for Oracle Application Integration Architecture Foundation Pack.*
This chapter discusses how to make migrated services into deployable artifacts. The following sections describe the steps to follow after migrating the services:

- **Step 1: Annotating Composites**
- **Step 2: Harvesting AIA Content**
- **Step 3: Generate Bill of Material**
- **Step 4: Deploy Composites**

### 12.1 Step 1: Annotating Composites

AIA recommends using annotations in the composite XML file to provide detailed information about:

- AIA artifacts and their relationships to other AIA artifacts.
- Composite-level descriptor properties that are used to configure the component at deployment and runtime.

AIA architecture categorizes SOA composites further as adapter services, requester services, provider services, and so on based on their usage. The meta information of these AIA services is used in maintaining Oracle Enterprise Repository assets of AIA asset types and linking them to Oracle Enterprise Repository assets with native asset types. This is accomplished with the help of AIA Harvester, which harvests the SOA composites.

For more information, see "Annotating Composites" in *Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack*.

### 12.2 Step 2: Harvesting AIA Content

After you have unit-tested, source-controlled, and completed your migrated service you can harvest these design-time composites into the Project Lifecycle Workbench and, optionally, Oracle Enterprise Repository.

When you choose to harvest into the Project Lifecycle Workbench, annotations in composite XML files are published to Project Lifecycle Workbench. Annotations published to Project Lifecycle Workbench are instrumental in facilitating downstream automation, such as bill of material (BOM) generation and deployment plan generation. Annotations and harvesting are required to enable this downstream automation.
Step 3: Generate Bill of Material

For more information, see "Harvesting Oracle AIA Content" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

12.3 Step 3: Generate Bill of Material

After the design-time composites are complete, they are harvested into the Project Lifecycle Workbench. At this point, you can use the Project Lifecycle Workbench to create a bill of material (BOM) for a PIP.

The BOM specifies the inventory of composites that make up the PIP. Using this feature can help expedite deployment plan generation for a PIP project by enabling the automatic or semiautomatic generation of the PIP’s BOM.

For more information, see "Working with Project Lifecycle Workbench Bills of Material" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

12.4 Step 4: Deploy Composites

Deploying custom built services is a two step process:

1. Generate the deployment plans and deploy using the AID (Application Integration Architecture Installation Driver).

2. Deploy custom-built services using the AID.

---

**Note:** You can deploy custom-built services only after installing Foundation Pack.

---

For more information, see "Generating Deployment Plans and Deploying Artifacts" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.
This chapter describes how to prepare and then migrate artifacts from the Business Service Repository (BSR) to the Oracle Enterprise Repository (OER).

This chapter includes the following sections:

- Preparing to Harvest the Services
- Harvesting Services to Oracle Enterprise Repository

13.1 Preparing to Harvest the Services

After you have completed the environment setup (including Oracle Enterprise Repository), prerequisites, service migration, and have annotated the services, you are ready to harvest the services.

For information about the prerequisites and environment setup, see Chapter 2, "Preparing to Migrate AIA Artifacts to AIA 11g Release 1."

For information about the service migration, see Section 7, "Migrating Services."

For information about annotating composites, see "Annotating Composites" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

13.2 Harvesting Services to Oracle Enterprise Repository

This section provides details about how to harvest services to Oracle Enterprise Repository.

The Business Service Repository (BSR) is an integral part of the AIA 2.4 and 2.5 offering. It provides users with SOA visibility, since a BSR instance is created and populated each time a new AIA 2.4 or 2.5 installation occurs.

In AIA Foundation Pack 11g Release 1, this functionality is consolidated in Oracle's SOA repository, Oracle Enterprise Repository. Oracle Enterprise Repository brings you additional asset management capabilities, such as asset lifecycle control, and analytics around asset usage and tracking.

The migration of 2.4 and 2.5 BSR content to an 11g Release 1 Oracle Enterprise Repository instance does not occur in isolation. Instead, it occurs as a part of the overall migration path from AIA 2.4 and 2.5 (on Oracle SOA Suite 10g) to AIA Foundation Pack 11g Release 1 (on Oracle SOA Suite 11g). For instance, it does not
Harvesting Services to Oracle Enterprise Repository

make sense to remain on AIA 2.4 or 2.5 (and therefore maintain respective BSR instances) on the 10g SOA Suite, while operating an 11g Oracle Enterprise Repository.

- For prebuilt AIA 2.4 and 2.5 content released by Oracle, you need not do anything. The content can be published into Oracle Enterprise Repository using the AIA Solution Pack.

- For custom-built content, you must harvest the content into Oracle Enterprise Repository by using the AIA Harvester following the samples and documents delivered in AIA Foundation Pack 11g Release 1.
  - Custom-build content (EBS.wsdl, EBM.xsd, EBO.xsd, ABCS.wsdl, and composites, for example) must be migrated and annotated in accordance with AIA Foundation Pack 11g Release 1 extension and annotation standards.
  - You can harvest your custom-built interfaces dynamically by running the AIA Harvester.

For more information about the AIA Harvester, see "Harvesting Design-Time Composites into Project Lifecycle Workbench and Oracle Enterprise Repository" in Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack.

Note that Oracle Enterprise Repository is about SOA asset lifecycle management, whereas, AIA Foundation Pack 11g Release 1 components, such as the Project Lifecycle Workbench, focus on facilitating development lifecycle management from functional definition and process decomposition to bill-of-material auto-generation in the context deployment plan construction.

You may find it convenient to manage and reuse the many types of SOA artifacts produced by AIA’s development lifecycle in a SOA repository, such as Oracle Enterprise Repository. If so, you can operate the AIA Project Lifecycle Workbench in concert with Oracle Enterprise Repository.

Oracle Enterprise Repository is an optional component to AIA Foundation Pack installation and execution. You can operate AIA Foundation Pack (including the Project Lifecycle Workbench) without using Oracle Enterprise Repository.
This chapter describes how to migrate the actual Domain Value Map and Cross-Reference data.

Migrating domain value maps (DVMs) and cross-references is a two-step process. Migration involves migrating both actual data and metadata. The DVMs have static data, as well as metadata. For cross-references, the data is dynamic and is stored in a database.

This chapter includes the following sections:

- Migrating Cross-Reference Data
- Migrating Extended DVM Data

If you are migrating actual DVM and cross-reference data, or if you have extended or customized cross-reference and DVMs, follow the procedures in this chapter to migrate this data.

For information about how to migrate DVM and cross-reference metadata, see Chapter 6, "Migrating Domain Value Map and Cross-Reference Metadata."

14.1 Migrating Cross-Reference Data

For cross-references in AIA Foundation Pack 11g Release 1, the schema has changed. The migration of cross-reference data should be performed using the **Oracle SOA Suite Command-Line Upgrade Tool**, which is provided as a part of Oracle Fusion Middleware 11g Release 1.

For more information, see "Using the Oracle SOA Suite Command-Line Upgrade Tool" in *Oracle Fusion Middleware Upgrade Guide for Oracle SOA Suite, WebCenter, and ADF*. 
If you have customized delivered DVM data to include additional values, follow these steps to migrate the data.

1. Add DVM data to the existing DVMs shipped in the physical location `<AIA_HOME>AIAMetaData/dvm/<DVM>`. In other words, edit the shipped 11g Release 1 DVMs to include the customizations to the 2.4 or 2.5 DVMs.

2. Update the `<AIA_HOME>/aia_instances/<INSTANCE_NAME>/config/UpdateMetadataDP.xml` with the respective DVM entry to update the Oracle Metadata Services (MDS) repository with actual values.

**Note:** Prior to executing an insert, identify the number of rows that will be inserted by running the select statement with the same selection criteria (where clause).

Inserting a very large number of rows (several hundreds of thousands to millions) using a single request could warrant database tuning, which might require the completion of tasks such as sufficient allocation of Oracle rollback segments.

Consult your database administrator for further help in this area.

Another option would be to fine-tune the selection criteria, thereby resulting in a smaller set of records to be processed in one request.

### 14.2 Migrating Extended DVM Data

If you have customized delivered DVM data to include additional values, follow these steps to migrate the data.

1. Add DVM data to the existing DVMs shipped in the physical location `<AIA_HOME>AIAMetaData/dvm/<DVM>`. In other words, edit the shipped 11g Release 1 DVMs to include the customizations to the 2.4 or 2.5 DVMs.

2. Update the `<AIA_HOME>/aia_instances/<INSTANCE_NAME>/config/UpdateMetadataDP.xml` with the respective DVM entry to update the Oracle Metadata Services (MDS) repository with actual values.
This chapter describes how to migrate from an AIA 2.4 or 2.5 production environment to an AIA 11g Release 1 environment.

Migration from an AIA 2.4 or 2.5 production environment to an AIA 11g Release 1 environment is a carefully planned exercise. The methodology must be determined by the project team on site, taking into account actual requirements. The overall process flow is illustrated in Figure 15–1.
15.1 General Guidelines for Migrating to an AIA 11g Release 1 Production Environment

Use the following steps as general guidelines for your migration:

1. Set up an AIA Foundation Pack 11g Release 1 environment.
2. Set up a fresh SOA instance data store and AQ JMS queues.
3. Migrate Composite Application Validation System (CAVS) definitions and system registrations to the production system.
4. Migrate services and deploy the services to the new 11g production environment.
5. Upload AIA metadata to the Oracle Metadata Services (MDS) repository.
6. As a planned outage, suspend and stop all events and applications triggering the pushing of messages to the AIA 2.4 or 2.5 environment.

7. Allow all in-flight messages in the AIA 2.4 or 2.5 environment to complete.

8. Move cross-reference data from the 10g cross-reference database to the 11g cross-reference database.

9. Modify applications to divert messages to the new AIA Foundation Pack 11g Release 1 environment.

10. Restart the events and applications triggering the pushing of messages to the AIA Foundation Pack 11g Release 1 environment.
This chapter discusses how to migrate Oracle Data Integrator (ODI) flows. After you have set up the environment, the ODI 11g server, and the Oracle 11g database, you will be ready to migrate ODI flows.

This chapter includes the following sections:

- Overview of ODI Flow Migration Tasks
- Describing the Source 10g ODI Server
- Describing the Target 11g ODI Server
- Migrating ODI
- Performing Post-Migration Steps
- Handling Cross-Reference Schema Changes
- Importing the Latest Knowledge Module for Cross-References
- Changing the Model and Interfaces That Involve Cross-References

16.1 Overview of ODI Flow Migration Tasks

To migrate ODI flows:

1. Set up the environment.
2. Export and move master repository and work repository schemas from the Oracle 10g database to the Oracle 11g database.
3. Import the source code Oracle 11g database master repository and work repository schemas.
4. Run upgrade assistant.

Figure 16–1 illustrates the overall ODI migration task flow.
Figure 16–1  ODI Migration Task Flow

16.2 Describing the Source 10g ODI Server

- The ODI code that you want to migrate should be up and running in ODI 10g repositories (in database).
- Ensure that the master repository is associated with only one work repository.
- Compile the following information about the source ODI system:
  - Host login user name and password.
  - Database installation location.
  - ODI repository details: master repository database user name and password and work repository database user name and password.
  - ODI user name and password.
16.3 Describing the Target 11g ODI Server

Compile the following information about the target ODI system:

- Host login user name and password.
- Database installation location.
- Database administrator user name and password.
- ODI repository details: master repository database user name and password and work repository database user name and password.

If these users are not present in the database, create the users and grant them session, connect, and resource privileges. Use the following Oracle database commands:

```
create user <User name> identified by <Password>
grant create session, connect, resource to <User name>
```

- ODI installation location.

Gather connection details for data servers in the ODI master repository (source database, target database, cross-reference database, DVM XML file location, AIAConfiguration.Properties file location, and so forth).

16.4 Migrating ODI

To migrate ODI:

1. Export the master repository and work repository schemas from the ODI 10g database.

To export the database schema, run the following commands from the bin folder of the Oracle database installation:

```
exp <master db username of 10g>/< master db password of 10g>
FILE=<location>/master.dmp
exp < work db username of 10g >/ < work db password of 10g>
FILE=<location>/work.dmp
```

**Note:** You must source the environment correctly for these commands to run correctly.

2. Move the master.dmp and work.dmp files to the server where the database for ODI 11g is installed.

3. Import the master and work repositories to the 11g database.

To import the database schema run the following commands from the bin folder of the Oracle database installation:

```
imp <MASTER REP DB USERNAME for 11g>/<MASTER REP DB PASSWORD for 11g>
FROMUSER=<MASTER REP DB USERNAME of 10 g> Touser=<MASTER REP DB USERNAME for 11g> FILE=< location >/master.dmp
imp <WORK REP DB USERNAME for 11g>/<WORK REP DB PASSWORD for 11g>
FROMUSER=<WORK REP DB USERNAME for 10g> TOUSER=<WORK REP DB USERNAME for 11g> FILE=<location>/work.dmp
```
4. Run the ODI upgrade operation of the Oracle Fusion Middleware Upgrade Assistant from the bin folder under ODI 11g installation home.

For Windows, run `ua.bat`.

For Linux, run `ua`.

a. The Oracle Fusion Middleware Upgrade Assistant Welcome page displays. Click Next.

b. The Specify Operation page displays, as shown in Figure 16–2. Select the Upgrade Oracle Data Integrator Repositories option. Upgrade the ODI database repositories only. Click Next.

Figure 16–2 Specify Operation

![Specify Operation](image)

c. The Prerequisites page displays, as shown in Figure 16–3. Verify that prerequisites have been met. Click Next.
**Figure 16–3 Prerequisites**

![Prerequisites](image)

d. The Specify Target Database page displays, as shown in Figure 16–4. Enter the details of the master repository to be migrated. Click Next.
The Specify Work Repositories User Name page displays, as shown in Figure 16–5. Enter the details of the work repository to be migrated. Click Next.
f. The Examining Components page displays, as shown in Figure 16–6. Enable the Upgrade Assistant to complete its component examination. Click Next.
g. The Upgrade Summary page displays, as shown in Figure 16–7. Verify that the upgrade information displayed is correct. Click Upgrade.
h. The Upgrading Components page displays, as shown in Figure 16–8. The page displays the progress of the upgrade.
i. After the upgrade process is complete, the Upgrade Complete page displays, as shown in Figure 16–9.
16.5 Performing Post-Migration Steps

1. After the ODI upgrade has been completed, connect to repositories using the ODI Studio and update the connections in the physical topology to point to the data servers to be used for development.

2. If the ODI projects use OdiInvokeWebService to invoke a web service, as with AIA Error Handling, for example, then reenter the password for all of the web service callouts in the projects.
   Also, update the web service URL to conform to AIA 11g Release 1 standards. Note that the password is encrypted using the ODI 11g OdiInvokeWebService command.

3. If ODI projects use cross-references, follow the steps specified in Section 16.6, "Handling Cross-Reference Schema Changes."

4. After all post-ODI upgrade changes are complete, regenerate all scenarios and test the flows.

16.6 Handling Cross-Reference Schema Changes

Oracle SOA Suite 11g Release 1 includes the following changes to the cross-reference schema (XREF_DATA table):

- The LAST_ACCESSSED column in the XREF_DATA table has been removed.
- The data type of the LAST_MODIFIED column in the XREF_DATA table has been changed from varchar2 to timestamp.
- The XREF_TABLE_NAME value is the fully qualified name of the cross-reference in Oracle Metadata Services repository (MDS).
   In 10g, the XREF name was populated as the name of the cross-reference. In 11g Release 1, it is populated as MDS_XREF_PATH + XREF NAME.
   For example, oramds:/apps/AIAMetaData/xref/RESOURCE_ID.xref.

To enable ODI code to work with the Oracle SOA Suite 11g Release 1 cross-reference schema:

1. Import the latest Knowledge Module for cross-references.
   This is delivered by ODI.
   For more information, see Section 16.7, "Importing the Latest Knowledge Module for Cross-References."

2. Change the ODI models and interfaces that involve cross-references.
   For more information, see Section 16.8, "Changing the Model and Interfaces That Involve Cross-References."

16.7 Importing the Latest Knowledge Module for Cross-References

To import the latest Knowledge Module for cross-references:

1. Import the latest IKM SQL Control Append (ESB XREF) into the project using INSERT_UPDATE mode.

2. In all interfaces that use this KM, update the XREF_TABLE_NAME option with the fully qualified name of the cross-reference table, as shown in Figure 16–10.
16.8 Changing the Model and Interfaces That Involve Cross-References

Because the XREF_DATA table has undergone data model changes, the model in ODI must also be changed accordingly.

**Note:** We assume that the XREF_DATA table is used only as the target of the interfaces in the project. The cross-reference view, which is created per PIP, is used as the source of interface.

**To update the model and interfaces:**

1. Log in to ODI and access the Designer tab.
2. Open the LAST_ACCESSED column of the XREF_DATA table in the model.
3. Access the Control tab and clear the Mandatory option, as shown in Figure 16–11.
4. Open the interface that uses the XREF_DATA table as the target and access the Mapping tab.

5. Clear the Active Mapping option for the LAST_ACCESSSED column on the target. Also delete any code in the Implementation area, as shown in Figure 16–12.
6. Perform steps 4 and 5 for all interfaces that use the XREF_DATA table.
7. Save the interface and ignore any warnings displayed while saving.
8. Regenerate all scenarios.
9. Disconnect from and reconnect to the repository. This is required.
10. Delete the LAST_ACCESSSED column of the XREF_DATA table in the model using the **Delete** option in the context menu of the LAST_ACCESSSED column, as shown in Figure 16–13.
11. Open the LAST_MODIFIED column of the XREF_DATA table in the model and change the **Datatype** field value to **TIMESTAMP**. Clear the **Length** and **Precision** fields in the Logical Format area, as shown in **Figure 16–14**.
12. Open the interface that uses the XREF_DATA table in the target and access the **Mapping** tab.

13. On the Implementation tab, change the value to `SYSTIMESTAMP`. Save the implementation code and interface. Ensure that the **Execute on** option is set to **Target**, as shown in **Figure 16–15**.
14. Perform steps 12 and 13 for all interfaces that use XREF_DATA.

15. Change the Mapping value of XREF_TABLE_NAME to the fully qualified name of the cross-reference, as shown in Figure 16–16.
16. Update the physical data source to connect to the new 11g XREF database.

17. Update the physical schema to the cross-reference schema on the 11g database, as shown in Figure 16–17.
18. In the Logical connection of the cross-reference schema, change the physical connection to point to the new cross-reference physical schema.

19. Regenerate all scenarios.

20. Test the execution of the code.
This chapter provides an overview and describes how to use the Oracle Application Integration Architecture (AIA) Migration Utility.

The AIA Migration Utility seamlessly migrates AIA 2.4 and 2.5 BPEL and Enterprise Service Bus (ESB) services to AIA Foundation Pack 11g Release 1. The utility leverages the SOA upgrade utility to perform the bulk of the migration tasks and automates most of the pre- and post-migration tasks.

For more information about the SOA upgrade utility, see “Considerations When Upgrading All Oracle SOA Applications” in Oracle Fusion Middleware Upgrade Guide for Oracle SOA Suite, WebCenter, and ADF.

This chapter includes the following sections:

■ Overview of the AIA Migration Utility
■ How to Use the AIA Migration Utility

17.1 Overview of the AIA Migration Utility

The following operations are performed by the AIA Migration Utility:

■ Detects the type of service, and depending on whether it is a BPEL process or an ESB process, invokes the appropriate version of the SOA upgrade utility to perform the migration.

■ Updates AIAComponents references in WSDLs and XSDs to point to Oracle Metadata Services (MDS) repository.

■ Updates the AIAAsyncErrorHandlingProcess reference in bpel.xml to point to the AIA Foundation Pack 11g Release 1 error handling service.

■ Updates domain-value map (DVM) and cross-reference (XREF) signatures in XSL files to point to AIAComponents in MDS. This step takes care of updating references to template and function calls in the XSL files used by the process.

■ Inserts annotations in composite.xml.

Annotations play a significant role in the AIA lifecycle and governance in AIA Foundation Pack 11g Release 1. The AIA Migration Utility injects annotations (empty placeholders) into the migrated composites.

Depending on whether a process is a Requester Application Business Connector Service (ABCS), Enterprise Business Service (EBS), Provider ABCS, or Adapter service, appropriate annotations are inserted into the composites. The utility uses the process.type variable in AIAMigrationUtility.properties for this purpose.
As a post-migration task, you must populate appropriate annotation values into the skeletal structure provided by the utility to be able to harvest AIA artifacts into Oracle Enterprise Repository.

For more information about annotation composites for harvesting, see "Annotating Composites" and "Harvesting Oracle AIA Content" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

17.2 How to Use the AIA Migration Utility

The AIA Migration Utility is delivered as a part of AIA Foundation Pack 11g Release 1.

**To use the AIA Migration Utility:**

1. The AIA Migration Utility is available in the AIA_HOME\utils folder. This is a standalone utility that can be used by unzipping it in any location, whether it is on a server or a developer’s machine.

   Figure 17–1 shows the content of the AIA_HOME\utils folder.

   **Figure 17–1  AIA HOME/util Folder Content**

   ```
   AIAMigrationUtility\AIAMigrationUtility.properties
   AIAMessageResubmissionUI
   AIAMigrationUtility
   DeploymentPlanGenerator
   EBODesignerDocGenerator
   UpdateStore
   SQLMerge
   ...
   ```

2. Update AIAMigrationUtility\AIAMigrationUtility.properties using the values provided in Table 17–1.

   Ensure that your values use only forward slashes "/".

   **Table 17–1  AIAMigrationUtility.properties Values**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>jdev.home</td>
<td>Value of JDeveloper home.</td>
</tr>
<tr>
<td></td>
<td>Example for Windows:</td>
</tr>
<tr>
<td></td>
<td>jdev.home = C:/oracle/Middleware/jdeveloper</td>
</tr>
<tr>
<td></td>
<td>Example for Linux:</td>
</tr>
<tr>
<td></td>
<td>jdev.home = /apps/oracle/Middleware/jdeveloper</td>
</tr>
<tr>
<td></td>
<td>Example for ADE:</td>
</tr>
<tr>
<td></td>
<td>jdev.home = /net/adc1006nap.example.com/vol/fade_</td>
</tr>
<tr>
<td></td>
<td>adc_txn/.fusionapps_ide/AIA/JDevs/JDEV_5536/jdeveloper</td>
</tr>
</tbody>
</table>

17-2 Oracle Fusion Middleware Migration Guide for Oracle Application Integration Architecture
3. Set the ANT and JDK paths by sourcing aiaenv.sh:

source <AIA_HOME>/aia_instances/<AIA_INSTANCE>/bin/aiaenv.sh

4. Invoke the AIAMigrationUtility as follows:

ant -f AIAMigrationUtility.xml

5. Open the migrated project in Oracle JDeveloper.
6. Point to the JPR file created in the project folder to which the service has been migrated. Open the project.
This chapter describes how to load Oracle Application Integration Architecture (AIA) metadata artifacts to the Oracle Metadata Services (MDS) repository.

You must upload AIA metadata artifacts to the AIA MetaData folder in the SOA partition of the Oracle Metadata Services repository (MDS) after performing any of the following tasks:

- Migrating services.
- Making schema changes.
- Making changes to configuration files AIAConfigurationProperties.xml and AIAEHNotification.xml.
- Making changes to or adding new WSDLs, XSLs, DVM, and Xref metadata.
- Making changes to default fault policies.

This chapter discusses the following topics:

- Accessing MDS from the AIA Service Artifact
- Moving Abstract Service WSDLs into MDS
- Migrating the AIA Configuration Properties File
- Moving Custom Extensions to EOL, ESL, and AOL to MDS
- Updating MDS

18.1 Accessing MDS from the AIA Service Artifact

To access the MDS store from the AIA service artifact, configure the adf-config.xml file located in `<application folder>.adf\META-INF\adf-config.xml`.

Example 18–1 provides a sample adf-config.xml file:

In your respective adf-config.xml files, adjust content in bold to reflect values for your MDS configuration.

Example 18–1 Sample adf-config.xml

```xml
<?xml version='1.0' encoding='windows-1252' ?>
<adf-config xmlns='http://xmlns.oracle.com/adf/config'
xmlns:sec='http://xmlns.oracle.com/adf/security/config'>
  <adf-mds-config xmlns='http://xmlns.oracle.com/adf/mds/config'>
    <mds-config xmlns='http://xmlns.oracle.com/mds/config'>

```
18.2 Moving Abstract Service WSDLs into MDS

Abstract WSDLs of all AIA services must be stored in MDS.

For information about how MDS is used in AIA and the utility used to move artifacts to MDS, see "Using MDS in AIA" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

**Note:** Abstract WSDLs should be modified to access the required schemas from MDS and then be moved to MDS.

For Application Business Connector Services (ABCS) and adapter services, abstract WSDLs are stored in MDS using the following structure:

AIAMetaData/AIAComponents/ApplicationConnectorServiceLibrary/<PRODUCT_CODE>/<Version Number>/<Service Type>.

Possible values for <Version Number> are V1, V2, and so on.

Possible values for <Service Type> are RequesterABCS, ProviderABCS, and AdapterServices.
For example:

- AIAMetaData/AIAComponents/ApplicationConnectorServiceLibrary/Siebel/V1/RequesterABCS
- AIAMetaData/AIAComponents/ApplicationConnectorServiceLibrary/Siebel/V1/ProviderABCS

**To prepare abstract service WSDLs to be moved into MDS:**

1. In AIA Foundation Pack 11g Release 1, the abstract WSDL of an ABCS that is being developed should be accessed from MDS. The exceptions to this rule are:
   - The Enterprise Business Service (EBS) reference WSDLs that define PartnerLink types.
   - Participating application reference WSDLs that define PartnerLink types.
   - Adapter WSDLs that are generated by Oracle JDeveloper.
   - Abstract WSDLs of services defined at extension points.

2. Modify the abstract WSDLs to access the required schemas from the MDS and then load them to MDS.
   - The application entity schemas, or Application Business Message (ABM) schemas, should be accessible from MDS. They should not be a part of each ABCS project.
   - For ABCSs, abstract WSDLs are located here in MDS:
     AIAMetaData/AIAComponents/ApplicationConnectorServiceLibrary/\<PRO DUCT_CODE>/\<Version Number>/\<Service Type>
     Possible values for \<Version Number> - V1, V2, and so on.
     Possible values for \<Service Type> are RequesterABCS and ProviderABCS.
   - For Composite Business Processes (CBPs) and Enterprise Business Flows (EBFs), abstract WSDLs are stored in MDS using the following structure:
     AIAMetaData\AIAComponents\BusinessProcessServiceLibrary\\<Service Type>
     For example:
     AIAMetaData\AIAComponents\BusinessProcessServiceLibrary\CBP
     AIAMetaData\AIAComponents\BusinessProcessServiceLibrary\EBF

**To configure the artifacts before moving the abstract WSDLs into MDS:**

1. Configure the ComponentType file to point to the abstract WSDLs in the MDS.
   a. The ui:wsdlLocation attribute in the service element should point to the abstract WSDLs in the MDS.
      For example:
      <service ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/ ApplicationConnectorServiceLibrary/SampleSEBL/RequesterABCS/Samples CreateCustomerSiebelReqABCSImpl.wsdl" name="SamplesCreateCustomerSiebel ReqABCSImpl">
   b. If the composite has a reference, then the ui:wsdlLocation attribute in the Reference element should point to abstract WSDLs in the MDS.
      For example:
Moving Abstract Service WSDLs into MDS

<reference name="SamplesCreateCustomerPartyPortalProvider"
ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/Application
ConnectorServiceLibrary/SamplePortal/V1/SamplesCreateCustomerPartyPortal
Provider.wsdl"<interface.wsdlinterface="http://xmlns.oracle.com/
SamplePortalApp#wsdl.interface(SamplesCreateCustomerPartyPortal
Provider? )"/>

2. Configure composite.xml.

a. In composite.xml, the location attribute of the import element should point to abstract WSDLs in the MDS.

For example:

```xml
<import
    location="oramds:/apps/AIAMetaData/AIAComponents/ApplicationConnectorServiceLibrary/SampleSEBL/RequesterABCS/SamplesCreateCustomerSiebelReqABCSImpl.wsdl" namespace= "http://xmlns.oracle.com/ABCSImpl/Siebel/Samples/CreateCustomerSiebelReqABCSImpl/V1"/>
```

b. The ui:wsdlLocation attribute in the service and reference elements should point to abstract WSDLs in the MDS.

For example:

```xml
<service ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/AIADServiceLibrary/SampleSEBL/RequesterABCS/SamplesCreateCustomerSiebelReqABCSImpl.wsdl" name="SamplesCreateCustomerSiebelReqABCSImpl">
```

c. If the composite has a mediator as a reference service, change the location attribute in the <EBS Reference WSDL> file as shown in this example:

```xml
<wsdl:import
```


Service Constructor generates empty port and location attributes for the binding.ws element under the reference element in the composite.xml file. Populate the attributes with relevant values as follows.

When the reference service is a BPEL component, the binding.ws element should be populated as shown here.

```xml
<binding.ws
    port="[Namespace of the Service as defined in the wsdl]#/wsdl.endpoint([Name of the Service as given in the WSDL]/[ Name of the Porttype as given in the WSDL])"
    location="< URL of the concrete WSDL/>
```

The name of the service is the value of the attribute definition/name in the abstract WSDL. This follows the Service naming convention in the ABCS composite.
For example, Name of the Service is the <name of the composite>, which in turn is the value of the name attribute of the definitions element in the WSDL.

The URL of the concrete WSDL should use the following format:
http://{host}:{port}/soa-infra/services/default/[Name of the Service as given in the name attribute of the definition element in the WSDL]/[Name of the Porttype element as given in the WSDL]?WSDL

For example:

```xml
<binding.wsport="http://xmlns.example.com/SamplePortalApp#wsdl.endpoint
(SamplesCreateCustomerPartyPortalProvider/SamplesCreateCustomerParty
PortalProvider)" location="http://ap6014rems.example.com:8001/soa-infra
/services/default/SamplesCreateCustomerPartyPortalProvider/Samples
CreateCustomerPartyPortalProvider?WSDL"/>
```

When the reference service is a mediator component, the binding.ws element should be populated as shown here:

```xml
<binding.ws port="[Namespace of the Service as defined in thewsdl]
#wsdl.endpoint([Name of the Porttype element as given in the WSDL]_ep/
[Name of the Porttype element as given in the WSDL]_pt)" location="
[URL of the concrete WSDL]"/>
```

The URL of the concrete WSDL should use the following format:
http://{host}:{port}/soa-infra/services/default/[Name of the Porttype element as given in the WSDL]/[Name of the Porttype element as given in the WSDL]_ep?WSDL

For example:

```xml
<binding.wsport="http://xmlns.example.com/EnterpriseServices/Core/
CustomerParty/V2#wsdl.endpoint(SamplesCustomerPartyEBS_ep/
CustomerPartyEBS_pt)" location="http://ap6014rems.example.com:
8001/soa-infra/services/default/SamplesCustomerPartyEBS/SamplesCustomer
PartyEBS_ep?WSDL"/>
```

3. Move the WSDLs to MDS.
   For information, see "Using MDS in AIA" in Oracle Fusion Middleware Developer’s Guide for Oracle Application Integration Architecture Foundation Pack.

### 18.3 Migrating the AIA Configuration Properties File

The AIAConfigurationProperties.xml file is a central place for system-, module-, and service-related settings in an AIA Foundation Pack environment. Starting with AIA Foundation Pack 11g Release 1, this file is placed in MDS.

The installation of AIA Foundation Pack 11g Release 1 includes a default configuration file and you must manually migrate any custom system and module settings from the AIA 2.4 or 2.5 environment. Specifically, these are the configurations that are part of the AIA configuration properties file, which is available on the server but not at the project folder.

See the following procedure for information about how to update AIAConfigurationProperties.xml with custom Service Configuration elements.

Service-related settings are handled by the AIA Migration Utility, which migrates them into a file called AIAServiceConfigurationProperties.xml for each migrated service.
AIA Installation Driver merges the content in AIAServiceConfigurationProperties.xml into AIAConfigurationProperties.xml as a part of the service deployment and uploads it into MDS.

Note that this does not happen when deploying from Oracle JDeveloper. In this case, the AIAConfigurationProperties.xml must be manually updated and uploaded to this location in MDS: SOA-MDS > apps/AIAMetaData/config.

To update AIAConfigurationProperties.xml with custom service configuration elements:
1. Access the $AIA_HOME/aia_instances/$INSTANCE_NAME/config folder.
   ■ Open AIAConfigurationProperties.xml and add the custom elements under respective service configurations.
   ■ Open the deployment plan file, UpdateMetaDataDP.xml. Update UpdateMetaDataDP.xml by inserting include tags for AIAConfigurationProperties.xml, which you want to add to the MDS:
     
     Add the following:
     
     `<fileset dir="${AIA_HOME}/aia_instances/${INSTANCE_NAME}"/>
     <include name="config/AIAConfigurationProperties.xml"/>
     </fileset>`

   Note: In the include tag, the folder path must be relative to the folder AIAMetaData, which is under $AIA_HOME.

2. Access the $AIA_HOME/aia_instances/$INSTANCE_NAME/bin folder.
3. Source aiaenv.sh by executing the following command:
   
   `source aiaenv.sh`

4. Access $AIA_HOME/Infrastructure/Install/scripts. Execute the script UpdateMetaData.xml by using the following command:
   
   `ant -f $AIA_HOME/Infrastructure/Install/scripts/UpdateMetaData.xml`

18.4 Moving Custom Extensions to EOL, ESL, and AOL to MDS

For information about migrating custom extensions to the Enterprise Object Library, see Chapter 3, "Migrating Enterprise Object Library Customizations."

For information about migrating custom extensions to the Enterprise Service Library, see Chapter 7, "Migrating Services."

Any custom extensions to the Application Object Library must be performed manually.

Foundation Pack provides MDS update scripts to add custom extensions to the existing library. The folder structure remains the same for the custom extensions.

18.5 Updating MDS

You must repeat this procedure each time a file needs to be added to MDS.
To update SOA-MDS (apps/AIAMetaData):
1. Access the $AIA_HOME/aia_instances/$INSTANCE_NAME/bin folder.
2. Source aiaenv.sh by executing the following command:
   
   ```
   source aiaenv.sh
   ```
3. Access the $AIA_HOME/aia_instances/$INSTANCE_NAME/config folder. Open the deployment plan file, UpdateMetaDataDP.xml.
4. Update UpdateMetaDataDP.xml by inserting include tags for each resource group that you want to add to the MDS:
   a. To upload all files under AIAMetaData, add the following:
      
      ```
      <include name ="AIAMetadata/**/>
      ```
   b. To upload AOL objects to MDS, see the following example.
      For example, to upload SEBL AOL objects in the AIAComponents/ApplicationObjectLibrary/SEBL/schemas folder, add the following:
      
      ```
      <include name="AIAComponents/ApplicationObjectLibrary/SEBL/schemas/**'/
      ```
   
   **Note:** In the include tag, the folder path must be relative to the folder AIAMetaData.
5. Access the AIA_HOME/Infrastructure/Install/config folder. Execute the UpdateMetaData.xml script:

   ```
   ant -f UpdateMetaData.xml
   ```
adapter service migration, 7-1
AIA Migration Utility, 17-1

BSR
See Business Service Repository, 13-1

Build.xml, 2-4

Business Service Repository
migrating artifacts to Oracle Enterprise Repository artifacts, 13-1

CAVS
See Composite Application Validation System, 11-1
Composite Application Validation System definition migration, 11-1
cross-references
migrating data, 14-1
migrating metadata, 6-1
custom process integration migration, 1-1
domain value maps
migrating data, 14-1
migrating metadata, 6-1
domain value maps
migrating data, 14-1
migrating metadata, 6-1

fault policy migration, 10-1
Foundation Pack migration preparation, 2-2

loading AIA metadata to Oracle Metadata Services repository, 18-1
Oracle Data Integrator flows, 16-1
overview, 1-1
preparing a certified database, 2-1
preparing for, 2-1
preparing Foundation Pack, 2-2
preparing Oracle Enterprise Repository, 2-3
preparing Oracle Fusion Middleware, 2-2
preparing Oracle Service Registry, 2-3
prerequisites, 2-3
production environment, 15-1
provider Application Business Connector Services, 7-1
requester Application Business Connector Services, 7-1

running XML instance validations, 3-10
EOL
See Enterprise Object Library, 3-1
error handling artifact migration, 10-1
extension migration for Process Integration Packs, 1-2
extension service migration, 8-1

Enterprise Business Object migration, 4-1
Enterprise Business Services, 5-1, 7-1
Enterprise Object Library, 3-1
Enterprise Object Library migration
customizations, 3-1
performing, 3-1
prerequisites, 3-1
running schema validations, 3-7

domain value maps
migrating data, 14-1
migrating metadata, 6-1

Enterprise Business Object migration, 4-1
Enterprise Business Services
migrating definitions, 5-1
migration, 7-1
Enterprise Object Library migration
customizations, 3-1
performing, 3-1
prerequisites, 3-1
running schema validations, 3-7
routing rules, 9-1
using AIA Migration Utility, 17-1

O
ODI
See Oracle Data Integrator, 16-1
OER
See Oracle Enterprise Repository, 2-3
Oracle Data Integrator flow migration, 16-1
Oracle Enterprise Repository
migrating Business Service Repository artifacts
to, 13-1
preparing for migration, 2-3
Oracle Fusion Middleware migration
preparation, 2-2
Oracle Metadata Services repository AIA metadata
load, 18-1
Oracle Service Registry migration preparation, 2-3
OSR
See Oracle Service Registry, 2-3

P
prerequisites, 2-3
Process Integration Pack extension migration, 1-2
production environment migration, 15-1
provider Application Business Connector Service
migration, 7-1

R
requester Application Business Connector Services
migration, 7-1
routing rule migration, 9-1

S
service deployment after migration, 12-1
service migration overview, 7-1

W
.war files
See web archive files, 2-3
web archive files
creating, 2-3
deploying, 2-3