Oracle® Application Integration Architecture Foundation Pack: Migration Guide to Foundation Pack 11gR1 (11.1.1.2.0)

Release 1 (11.1.1.2.0)
Part No. E17361-01

April 2010
Contents

Preface ................................................................................................................................. 5
Oracle AIA Guides .................................................................................................................. 5
Additional Resources ........................................................................................................... 5
1. Understanding Migration of AIA 2.4 and 2.5 Artifacts to AIA Foundation Pack 11gR1 (11.1.1.2.0) .......................................................................................................................... 7
  1.1. Overview of the Oracle Fusion Middleware Environment Migration ................................ 7
  1.2. Overview of AIA Service Code Migration Using the AIA Migration Utility ........................ 8
2. Prerequisites ...................................................................................................................... 11
3. Using the AIA Migration Utility ....................................................................................... 15
  3.1. How to Use the AIA Migration Utility ........................................................................... 15
4. Migrating AIA Services ...................................................................................................... 19
  4.1. How to Migrate Adapter Services ................................................................................ 19
  4.2. How to Migrate Provider ABCSs ................................................................................. 20
  4.3. How to Migrate EBSs .................................................................................................. 21
  4.4. How to Migrate Requester ABCSs ............................................................................. 22
  4.5. How to Migrate Extension Services ............................................................................ 23
5. Migrating Domain Value Maps and Cross-References ..................................................... 25
6. Migrating Error Handling .................................................................................................. 27
  6.1. How to Migrate Fault Policies ..................................................................................... 27
  6.2. How to Migrate Transaction Configurations ................................................................. 28
  6.3. How to Configure AIAConfigurationProperties.xml for Error Handling .................. 29
  6.4. How to Migrate Error Notification Setup .................................................................... 29
  6.5. How to Configure the Oracle User Messaging Service for Email Notifications ............ 30
  6.6. How to Configure User Email Addresses .................................................................. 30
  6.7. How to Migrate Custom Error Handlers ..................................................................... 30
  6.8. How to Migrate Custom Subscribers ......................................................................... 32
7. Loading AIA Metadata to Oracle Metadata Services Repository ...................................... 33
  7.1. How to Access MDS from the AIA Service Artifact ..................................................... 33
  7.2. How to Move Abstract Service WSDLs into MDS ....................................................... 34
  7.3. How to Migrate the AIA Configuration Properties File .............................................. 37
  7.4. How to Migrate Custom Extensions to EOL, AOL, and ESL ...................................... 38
7.5. How to Update MDS .................................................................................................................. 39
8. Migrating AIA 2.4 and 2.5 Business Service Repository Artifacts to Oracle Enterprise
   Repository Artifacts .................................................................................................................. 41
9. Cutting Over From a 10g- to an 11g-Based Production Environment ............................... 43
10. What You Need to Know About a SOA 11g Upgrade ......................................................... 45
Index ............................................................................................................................................. 47
Preface

Welcome to the Oracle Application Integration Architecture Foundation Pack: Migration Guide to Foundation Pack 11gR1 (11.1.1.2.0).

Oracle Application Integration Architecture (AIA) provides the following guides and resources for this release:

Oracle AIA Guides

- Oracle Application Integration Architecture Foundation Pack: Installation Guide
- Oracle Application Integration Architecture Foundation Pack: Getting Started with the Oracle AIA Foundation Pack and Demo
- Oracle Application Integration Architecture Foundation Pack: Concepts and Technologies Guide
- Oracle Application Integration Architecture Foundation Pack: Development Guide
- Oracle Application Integration Architecture Foundation Pack: Infrastructure Components and Utilities Guide
- Oracle Application Integration Architecture Foundation Pack: Reference Process Model Guide
- Oracle Application Integration Architecture Foundation Pack: Migration Guide to Foundation Pack 11gR1 (11.1.1.2.0)

Additional Resources

The following resources are also available:

<table>
<thead>
<tr>
<th>Resource</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Application Integration Architecture Foundation Pack: Product-to-Guide Index</td>
<td>My Oracle Support: <a href="https://support.oracle.com/">https://support.oracle.com/</a></td>
</tr>
<tr>
<td>Known Issues and Workarounds</td>
<td>My Oracle Support: <a href="https://support.oracle.com/">https://support.oracle.com/</a></td>
</tr>
<tr>
<td>Documentation updates</td>
<td>My Oracle Support: <a href="https://support.oracle.com/">https://support.oracle.com/</a></td>
</tr>
</tbody>
</table>
1. Understanding Migration of AIA 2.4 and 2.5 Artifacts to AIA Foundation Pack 11gR1 (11.1.1.2.0)

This guide details a comprehensive migration strategy to migrate Oracle Application Integration Architecture (AIA) 2.4 and 2.5, built on Oracle Fusion Middleware 10g, to AIA Foundation Pack 11g Release 1 (11.1.1.2.0), built on Oracle Fusion Middleware 11g. The migration involves upgrades at various levels, from individual services to server environments.

AIA Foundation Pack 11gR1 (11.1.1.2.0) provides infrastructure utilities that enable you to carry out the migration process efficiently and with ease. Specifics about the migration strategy and corresponding utilities are covered in this guide.

1.1. Overview of the Oracle Fusion Middleware Environment Migration

The AIA10g-based environment consists of the following:

- Oracle 10g Database

  For information about upgrading Oracle 10g Database to 11g, see Oracle Database Upgrade.

- Oracle 10g Oracle Fusion Middleware

  For information about upgrading Oracle 10g Fusion Middleware to 11g, see Oracle Fusion Middleware Upgrade Guides.

  For information about general upgrade planning, see the Oracle Fusion Middleware Upgrade Planning Guide.

AIA 2.4 and 2.5 installations use Oracle Database to make the best use of the following:

- Cross references.
- Oracle Advanced Queuing (AQ) for message persistence.
- AIA store for storing Business Service Repository (BSR), system registration, and error notification information.

The strategies for migrating each of the above to Oracle Database 11g are discussed in the following sections.
AIA 2.4 and 2.5 installations based on Oracle Fusion Middleware 10g use the Oracle Fusion Middleware components BPEL and Enterprise Service Bus. Migration to Oracle Fusion Middleware 11g-based AIA Foundation Pack 11gR1 installation requires completion of the following tasks:

- Set up the AIA Foundation Pack 11gR1 (11.1.1.2.0) Oracle Fusion Middleware environment.

For more information, see Oracle Application Integration Architecture Foundation Pack: Installation Guide.

- Migrate AIA 2.4 and 2.5 service artifacts to AIA Foundation Pack 11gR1 (11.1.1.2.0) artifacts. Subsequently, deploy the migrated artifacts to the Oracle Fusion Middleware 11g environment. This step is covered in detail in the following chapters.

1.2. Overview of AIA Service Code Migration Using the AIA Migration Utility

Migration of AIA 2.4 and 2.5 service artifacts to AIA Foundation Pack 11gR1 (11.1.1.2.0) artifacts facilitates the easy adoption of the component-based architecture of Oracle Fusion Middleware 11g for AIA solution implementation.

The following diagram depicts the migration process.
The following table provides a list of AIA 2.4 and 2.5 artifacts and their equivalent AIA Foundation Pack 11gR1 (11.1.1.2.0) 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 11gR1 (11.1.1.2.0) 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>
2. Prerequisites

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

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

- Access to the AIA artifacts to be migrated.
- Installation of Oracle Fusion Middleware 11gR1PS1, including Oracle SOA Suite and Oracle Database 11gR1.
- Installation of AIA Foundation Pack 11gR1 (11.1.1.2.0). 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.
- All of the service references (adapter services, extension service, infrastructure service, 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.
- The AIA 2.4 or 2.5 service project should be compiled, deployed, and executed without errors in the 10g environment.
- AIA components should be deployed as a .war file on the 11g server.

**To create and deploy AIA components as a .war file:**

1. Copy the AIAComponents folder from [aia_home/AIAComponents] on the 10g server to a folder on the server where Foundation Pack 11gR1 (11.1.1.2.0) is installed.

   For example:

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

   **Note:** Please apply your custom extensions to the AIAComponents, if any.

2. Create a subfolder named WEB-INF under c:\Metadata for Windows, or /slot/ems1739/oracle/Metadata 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 then save.
Prerequisites

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

6. Open weblogic.xml using a text editor, add the following content, and then save.

```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 following content. Replace the content in square brackets [] with the values relevant to your environment and then save.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<project name="AIAComponentsAsWAR" default="copyAIAComponents"
  basedir="." >
  <target name="copyAIAComponents">
    <taskdef name="wldeploy"
      classname="weblogic.ant.taskdefs.management.WLDeploy">
      <classpath>
        <pathelement location="[JDEV install folder on your local machine]/wlserver_10.3/server/lib/weblogic.jar"/>
      </classpath>
    </taskdef>
    <jar destfile="AIAComponents.war">
      <fileset dir="c:/Metadata">
        <include name="**/AIAComponents/**"/>
        <include name="**/WEB-INF/**"/>
      </fileset>
    </jar>
    <wldeploy action="deploy"
      name="AIAComponents" source="[c:/Metadata (for linux /slot/ems1739/oracle/Metadata)]/AIAComponents.war"
      user="[weblogic admin user name]" password="[weblogic admin user pwd]" verbose="true"
      adminurl="t3://[weblogic server host name]:[port num]"
      debug="true" targets="[WL_SOA_SERVER_NAME]" upload="true"
      failonerror="true">
      <echo message="- Deployment of AIAComponents.war completed"/>
    </wldeploy>
  </target>
</project>
```

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.
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 will build and deploy AIAComponents as a .war file to the 11gR1 PS1 server.

For example:

- For Windows
  
  C:\Metadata>ant

- For Linux
  
  -bash-3.00$ cd /slot/ems1739/oracle/Metadata
  -bash-3.00$ ant
3. Using the AIA Migration Utility

The Oracle Application Integration Architecture (AIA) Migration Utility seamlessly migrates AIA 2.4 and 2.5 BPEL and ESB services to AIA Foundation Pack 11gR1 (11.1.1.2.0). 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 Oracle Fusion Middleware Upgrade Guide for Oracle SOA Suite, WebCenter, and ADF, “Considerations When Upgrading All Oracle SOA Applications.”

The following operations are performed by the AIA Migration Utility:

- Updates the AIAAsyncErrorHandlingProcess reference in bpel.xml to point to the AIA Foundation Pack 11gR1 (11.1.1.2.0) error handling service.
- Detects the type of service and depending on whether it is a BPEL process or an Enterprise Service Bus (ESB) process, invokes the appropriate version of the SOA upgrade utility to perform the migration.
- Updates AIAComponents references in WSDLs and XDSs to point to Oracle Metadata Services (MDS) repository.
- 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 11gR1 (11.1.1.2.0). 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 Oracle Application Integration Architecture Foundation Pack: Development Guide, “Annotating Composites” and “Harvesting Oracle AIA Content.”

3.1. How to Use the AIA Migration Utility

The AIA Migration Utility is delivered as a part of AIA Foundation Pack 11gR1 (11.1.1.2.0)
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.

```
/slot/ems1095/oracle/AIA_HOME/util
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Ext</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIAMessageResubmissionUtil</td>
<td></td>
</tr>
<tr>
<td>AIAMigrationUtility</td>
<td></td>
</tr>
<tr>
<td>DeploymentPlanGenerator</td>
<td></td>
</tr>
<tr>
<td>EBODesignerDocGenerator</td>
<td></td>
</tr>
<tr>
<td>UpdateStore</td>
<td></td>
</tr>
<tr>
<td>XSLMerge</td>
<td></td>
</tr>
</tbody>
</table>

**Folder view from AIA HOME/util**

2. Update AIAMigrationUtility/AIAMigrationUtility.properties using the values provided in the following table. Ensure that your values use only forward slashes "/".

<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>• For windows</td>
</tr>
<tr>
<td></td>
<td>Example: jdev.home= C:/oracle/Middleware/jdeveloper</td>
</tr>
<tr>
<td></td>
<td>• For linux</td>
</tr>
<tr>
<td></td>
<td>Example: jdev.home = /apps/oracle/Middleware/jdeveloper</td>
</tr>
<tr>
<td></td>
<td>• For ADE</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>jdev.home=/net/adc1006nap.us.oracle.com/vol/fade_adc_txn/.fusionapps_ide/AIA/JDevs/JDEV_5536/jdeveloper</td>
</tr>
<tr>
<td>service.home</td>
<td>Location value of the source project.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>service.home=D:/Work/11g/workspace/source/SamplesAsyncCreateCustomerSiebelReqABCSImpl</td>
</tr>
<tr>
<td></td>
<td>service.home=/slot/ems1739/oracle/workspace/source/sampleapp/SamplesAsyncCreateCustomerSiebelReqABCSImpl</td>
</tr>
<tr>
<td>target.dir</td>
<td>Location value of the target folder.</td>
</tr>
<tr>
<td></td>
<td>Example:</td>
</tr>
<tr>
<td></td>
<td>target.dir=D:/Work/11g/workspaces/destination</td>
</tr>
<tr>
<td></td>
<td>service.home=/slot/ems1739/oracle/workspace/destination</td>
</tr>
<tr>
<td>service.name</td>
<td>Project name value. If the project name and the service name (BPEL Process Name) are different, then you must provide the service name (BPEL Process Name) value.</td>
</tr>
</tbody>
</table>
Using the AIA Migration Utility

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>service.name</td>
<td>Example: service.name=SamplesAsyncCreateCustomerSiebelReqABCSImpl</td>
</tr>
<tr>
<td>http.hostname</td>
<td>Server location where Foundation Pack 11gR1 (11.1.1.2.0) is installed.</td>
</tr>
<tr>
<td></td>
<td>Example: http.hostname=ap6014rems.us.oracle.com</td>
</tr>
<tr>
<td>http.port</td>
<td>Server port value.</td>
</tr>
<tr>
<td></td>
<td>Example: http.port=8001</td>
</tr>
<tr>
<td>application.name</td>
<td>Application name that the upgraded project should be a part of.</td>
</tr>
<tr>
<td></td>
<td>Example: application.name=sampleapp</td>
</tr>
<tr>
<td>process.type</td>
<td>Indicates the type of AIA artifact that needs to be migrated. Basically, this property provides empty placeholder annotations within composite.xml. Input to this attribute can be one of the following values: Provider, Requestor, DBAdapter, JMSAdapter, OracleAppsAdapter, EBF, or EBS.</td>
</tr>
<tr>
<td></td>
<td>Example: process.type=Provider</td>
</tr>
</tbody>
</table>

3. Set the ANT and JDK paths.

4. Invoke the AIAMigrationUtility.

   Use this command to invoke the Migration utility: `ant -f AIAMigrationUtility.xml`

5. Open the migrated project in Oracle JDeveloper.

6. Point to the .jpr file that is created in the project folder to which the service has been migrated and open the project.
4. Migrating AIA Services

This chapter discusses the following topics:

- How to Migrate Adapter Services
- How to Migrate Provider ABCSs
- How to Migrate EBSs
- How to Migrate Requester ABCSs
- How to Migrate Extension Services

4.1. How to Migrate Adapter Services

This procedure uses a DB adapter to illustrate its example.

To migrate adapter services:

1. Complete the pre-migration tasks.
   a. Change the references in DB Adapter wsdl, if there are any changes to the JNDI references.
   b. Change the jca:address location in the service section.
      For example: jca:address location="eis/DB/AIASamplesDB"
   c. Change the jca:UIConnectionName.
      For example: UIConnectionName="AIASamplesDB"
   d. Verify service builds in 10g.
   e. If any custom adapters are being developed and there are services developed using the custom adapters, then the custom adapters must first be ported to Oracle Fusion Middleware 11g SOA Suite.

2. Run the AIA Migration Tool.

   For more information, see How to Use the AIA Migration Utility.

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

   For example: process.type= DBAdapter

3. Complete the post-migration tasks.
a. Annotate the Adapter Service composite.

For more information about annotation composites for harvesting, see Oracle Application Integration Architecture Foundation Pack: Development Guide, “Annotating Composites.”

b. Configure adf-config.xml to access MDS store.

For more information, see How to Access MDS from the AIA Service Artifact.

### 4.2. How to Migrate Provider ABCSs

To migrate provider ABCS:

1. Complete the pre-migration tasks.
   a. Verify 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.
   c. Comment out the wsdlRuntimeLocation in the bpel.xml.

2. Run the AIA Migration Tool.

For more information, see How to Use the AIA Migration Utility.

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

For example: process.type= Provider

3. Complete the post-migration tasks.
   a. 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 file, WSDL file, and XSL files.
   b. Configure the abstract WSDLs.

For more information, see How to Move Abstract Service WSDLs into MDS.

c. Configure fault policies.

For more information, see How to Migrate Fault Policies.
d. Specify the transaction-related properties.


e. Complete the annotations in Composite.XML. This can be performed after unit testing, but must be completed prior to publishing.

f. Annotate the ABCS composite.

For more information about annotation composites for harvesting, see *Oracle Application Integration Architecture Foundation Pack: Development Guide*, “Annotating Composites.”

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

h. Some XPath functions have been deprecated in earlier versions of 10g; replace the deprecated functions with the latest 11g equivalents.

For more information, see *What You Need to Know About a SOA 11g Upgrade*.

i. Configure adf-config.xml to access the MDS store.

For more information, see *How to Access MDS from the AIA Service Artifact*.

### 4.3. 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. Remove system group files and references in the .esbsvc file. Change the qname attribute value in the .esbsvc file.

   c. 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: If the qname attribute value is “AIASystem.EBS.CustomerPartyEBS”, change it to “CustomerPartyEBS”

   d. Change the service references in .esbsvc files (wsdlURL).

      For example: BPELSystem_default_SamplesAsyncCreateCustomerPartyPortalProvABCSImpl_Sample sAsyncCreateCustomerPartyPortalProvABCSImpl_1_0.esbsvc
2. Run the AIA Migration Utility.
   In the AIAMigrationUtility.Properties file provide the value for the process.type attribute.
   For example: process.type= EBS

   For more information, see How to Use the AIA Migration Utility.

3. Complete post-migration tasks.
   a. Because the SOA upgrade utility may not migrate the expressions appropriately, you may need to manually change the filter expressions.
   b. Configure the concrete URLs in the composite.xml binding.ws section.
   c. Configure adf-config.xml to access the MDS store.

   For more information, see How to Access MDS from the AIA Service Artifact.

   d. Add an explicit assign statement to the routing rule, where the part names are different.

4.4. 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.
      d. Comment out the wsdlRuntimeLocation in the bpel.xml.

   2. Run the AIA Migration Utility.
      In the AIAMigrationUtility.Properties file provide the value for process.type attribute.
      For example: process.type= Requestor

      For more information, see How to Use the AIA Migration Utility.

   3. Complete post-migration tasks.
      a. See How to Migrate Provider ABCSs.
      b. 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.

For example:

In composite.xml:

```xml
<reference
  ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/InfrastructureServiceLibrary/V1/wsdls/AsyncRequestRecipient.wsdl"
  name="AsyncRequestRecipient">
  <interface.wsdl
    interface="http://xmlns.oracle.com/apps/aia/cavs/Any#wsdl.interface(CAVSAnyPort)"/>
  <binding.ws
    port="http://xmlns.oracle.com/apps/aia/cavs/Any#wsdl.endpoint(CAVSAnyService/CAVSAnyPort)"
    location="http://sdc60008sems.us.oracle.com:8088/AIAValidationSystemServlet/asyncrequestrecipient?wsdl"/>
</reference>
```

In the component type file:

```xml
<reference ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/InfrastructureServiceLibrary/V1/wsdls/AsyncRequestRecipient.wsdl"
  name="AsyncRequestRecipient">
  <interface.wsdl
    interface="http://xmlns.oracle.com/apps/aia/cavs/Any#wsdl.interface(CAVSAnyPort)"/>
</reference>
```

Change filter expressions in the .mplan file:

```xml
<condition language="xpath"
  xmlns:ebo="http://xmlns.oracle.com/EnterpriseObjects/Core/EBO/CurrencyExchange/V1"
  xmlns:corecom="http://xmlns.oracle.com/EnterpriseObjects/Core/Common/V2">
</condition>
```

### 4.5. How to Migrate Extension Services

To migrate extension services:

1. Modify extension concrete service WSDLs to use the relative paths for Enterprise Object Library references.
2. Push the concrete WSDL into the AIAMetadata/AIAComponents/ExtensionServiceLibrary folder in the MDS repository.

3. Customer extension services need to be migrated with the help of AIA Migration Utility or FMW upgrade utility.
5. **Migrating Domain Value Maps and Cross-References**

This is a two-step process. Migration involves upgrading both metadata and actual data. The domain value maps (DVM) have meta information, as well as the static data. For cross-references (XREFs), the data is dynamic and is stored in a database.

**To migrate DVM and XREF metadata:**

1. Set the input and output folder properties in the AIAMigrationUtility.properties file.
   
   ```
   inputDir = <directory where the dvm files exists>
   outputDir = <directory location where migrated dvm has to be placed>
   ```

2. Run the AIA Migration Utility
   
   - For DVMs
     ```
     ant -f AIAMigrationUtility.xml upgradeDVM -DinputDir=<path> -DoutputDir=<path>
     ```
   
   - FOR XREFs
     ```
     ant -f AIAMigrationUtility.xml upgradeXRef-DinputDir=<path> -DoutputDir=<path>
     ```

For XREFs in 11g, Oracle Database 11g is needed. Also the XREF schema for 11g has changed. The migration of XREF data may be performed manually, or by using the procedure provided here:

**To migrate XREF data from AIA 2.4 and 2.5 to AIA Foundation Pack 11gR1 (11.1.1.2.0) using a database script leveraging a database link:**

1. Connect as the user system to the database hosting the AIA Foundation Pack 11gR1 (11.1.1.2.0) XRef schema.

2. Adjust the following script to match your environments.

   ```
   create database link aia25 connect to aia identified by aia using
   '(
   (DESCRIPTION =
     (ADDRESS =
       (PROTOCOL = TCP)
       (Host = ap6060fems.us.oracle.com)
       (Port = 1561)
     )
     (CONNECT_DATA = (SID = aiafpdev))
   );
   ```
insert into as1_xref.xref_data
select 'oramds:/apps/AIAMetaData/xref/' || xref_table_name || '.xref' xref_table_name, xref_column_name, row_number, value,
is_deleted,
TO_TIMESTAMP('01/01/1970','mm/dd/yyyy') +
umtodsinterval(LAST_MODIFIED/1000,'second') last_modified
from xref_data@aia25
where xref_table_name = 'AIADEMOORDERS'
and IS_DELETED = 'N';

commit;

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

3. Execute the script.
6. Migrating Error Handling

This section discusses the following topics:

- How to Migrate Fault Policies
- How to Migrate Transaction Configurations
- How to Configure AIAConfigurationProperties.xml for Error Handling
- How to Migrate 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

6.1. 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 instructions to update them to AIA Foundation Pack 11gR1 (11.1.1.2.0).

The following example shows how to associate a fault policy defined in a sample fault-policy file with a fault-policy binding.xml file. These two files should be placed within the composite process folder.

**To associate a fault policy defined in a sample fault-policy file with a fault-policy binding.xml file:**

1. Consider a sample fault policy file.

   `SamplesQueryCustomerPartyPortalProvABCSImplFaultPolicy.xml`, with the fault policies defined as shown here:

   ```xml
   <faultPolicies xmlns="http://schemas.oracle.com/bpel/faultpolicy">
   <faultPolicy version="2.4" id="SamplesQueryCustomerPartyPortalProvABCSImplFaultPolicy" . . . .
   . . . . ></faultPolicy>
   ```
The name of the java class is changed from BPELJavaAction to CompositeJavaAction. There is no change in the package structure.

 oracle.apps.aia.core.eh.CompositeJavaAction

Similarly, to suppress notifications if any Process Integration Pack (PIP) used aia-no-action, then the corresponding name of the java class should be changed from BPELJavaNoAction to CompositeJavaNoAction. Again, there is no change in the package structure.

 oracle.apps.aia.core.eh.CompositeJavaNoAction

2. Associate the policies defined in the preceding fault policy file with the level of fault policy binding you are using. We recommend that you set this at the SOA composite level.

3. To do this, modify the template fault-bindings.xml file.

4. In the fault-bindings.xml file, the association performed as follows:

   <faultPolicyBindings version="2.4"
   xmlns=http://schemas.oracle.com/bpel/faultpolicy
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <composite
   faultPolicy="SamplesQueryCustomerPartyPortalProvABCSImplFaultPolicy"/>
   </faultPolicyBindings>


In 11g Mediator, fault policies are similar to BPEL fault policies and the corresponding java class is also the same, CompositeJavaAction, for example.


6.2. How to Migrate Transaction Configurations

6.3. **How to Configure AIAConfigurationProperties.xml for Error Handling**

As in AIA 2.4 and 2.5, the AIAConfigurationProperties.xml file continues to contain settings related to AIA Error Handling, even in AIA Foundation Pack 11gR1 (11.1.1.2.0). The following table provides an overview and instructions on the meaning of the new settings introduced with AIA Foundation Pack 11gR1 (11.1.1.2.0).

<table>
<thead>
<tr>
<th>Setting</th>
<th>Values</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>EH.INVOKE.HWF</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>EH.INVOKE.NOTIFY</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>EH.AGGR.NOTIFY</td>
<td>false</td>
<td>true</td>
</tr>
<tr>
<td>FROM EMAIL.ID</td>
<td>Email address</td>
<td>Sender Email address for error notifications. Note that the email server may deny notifications if the sender email address is not accepted by the email server, for example, if the email domain does not match.</td>
</tr>
<tr>
<td>EH.DEFAULT.ACTOR.ROLE</td>
<td>User Name</td>
<td>Actor user name. Note that this is a user name and no longer a role, as in AIA 2.4 and 2.5.</td>
</tr>
<tr>
<td>EH.DEFAULT.FYI.ROLE</td>
<td>User Name</td>
<td>For your information (FYI) user name. Note that this is a user name and no longer a role, as in AIA 2.4 and 2.5.</td>
</tr>
</tbody>
</table>

6.4. **How to Migrate Error Notification Setup**

To migrate the error notification setup from an AIA 2.4 or 2.5 environment to a new AIA Foundation Pack 11gR1 (11.1.1.2.0) environment, an administrator needs to review all notification records and manually duplicate these entries in the new AIA Foundation Pack 11gR1 (11.1.1.2.0) environment. The Error Notification setup 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 already existed in AIA 2.4 and 2.5. The new attributes relate to error notification throttling and are defaulted 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 *Oracle Application Integration Architecture Foundation Pack: Infrastructure Components and Utilities Guide*, “Using Error Notifications.”
6.5. 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.” At the very least, provide a valid value for the “OutgoingMailServer” property, such as “mail.oracle.com,” and any other settings that are relevant for your environment.

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


6.6. 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,” to which you will assign the email address to which notifications will be sent.

6.7. How to Migrate Custom Error Handlers

Similar to AIA 2.4 and 2.5 Error Handling, AIA Foundation Pack 11gR1 (11.1.1.2.0) 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 custom error handler:

- Implements handleCompositeSystemError as opposed to handleBPELSyserror 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 the following code sample.

For your reference, see this code sample from the Foundation Pack 11gR1 (11.1.1.2.0) AIA Demo implementation. Note the highlighted areas to understand the differences between AIA Foundation Pack 11gR1 (11.1.1.2.0) and 2.4 and 2.5 implementations.

```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.AIAErrorHandlerExtension;
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 componetType) {

        String enrichedFaultMessageConstructed = faultMessageConstructed;

        if (componentType.equals("bpel")) {
            try {
                BPELFaultRecoveryContextImpl bpelCtx = (BPELFaultRecoveryContextImpl) iFaultRecoveryContext;
                CXMessageVariable bpelVar = (CXMessageVariable) bpelCtx.getVariableData("AIADemoProcessCreditChargeAuthorizationReqMsg");
                if (bpelVar == null) {
                    String ebmPayload = bpelVar.toXML();
                }
            } catch (Exception ex) {
                
            }
        }
    }
}
```
6.8. How to Migrate Custom Subscribers

The migration of custom subscribers to the AIA_ERROR_TOPIC that were 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 11gR1 (11.1.1.2.0). Note that these names have changed since AIA 2.4 and 2.5. See the following sample jca file from the AIA Demo to understand the relevant highlighted properties and their correct settings:

```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" UIJmsProvider="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>
```
7. Loading AIA Metadata to Oracle Metadata Services Repository

All schemas, WSDLs, XSLs, DVM and XREF meta information, default fault policies, and 
AIAConfigurationProperties.xml and AIAEHNtification.xml files must be uploaded to the AIA 
Meta Data folder in the SOA partition of the Oracle Metadata Services repository (MDS).

7.1. How to Access 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.

Here is a sample adf-config.xml file:

```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">
<persistence-config>
<metadata-namespaces>
<namespace metadata-store-usage="mstore-usage_2" path="/apps"/>
<namespace metadata-store-usage="mstore-usage_3"
path="/soa/shared"/>
</metadata-namespaces>
<metadata-store-usages>
<metadata-store-usage id="mstore-usage_2">
<metadata-store class="oracle.mds.persistence.stores.db.DBMetadataStore">
(property value="SH_MDS" name="jdbc-userid")
(property value="manager" name="jdbc-password")
(property value="jdbc:oracle:thin:@server.us.oracle.com:1521:orcl"
name="jdbc-url")
(property value="soa-infra" name="partition-name")
</metadata-store>
</metadata-store-usage>
<metadata-store-usage id="mstore-usage_3">
<metadata-store class="oracle.mds.persistence.stores.file.FileMetadataStore">
(property value="D:\jdev\jdeveloper\integration"
name="metadata-path")
(property value="seed" name="partition-name")
</metadata-store>
</metadata-store-usage>
</metadata-store-usages>
</metadata-storage-config>
```
7.2. **How to Move Abstract Service WSDLs into MDS**

The abstract WSDLs of all AIA services are stored in MDS.

**For information** about how MDS is used in AIA and the utility used to move artifacts to MDS, see *Oracle Application Integration Architecture Foundation Pack: Development Guide*, “Building AIA Integration Flows,” Using MDS in AIA.

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

For ABCSs and adapter services, the abstract WSDLs need to be placed in MDS by creating the following folder structure:

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

Possible values for `<Service Type>` are RequesterABCS, ProviderABCS, and AdapterServices. Possible values for `<Version Number>` are V1, V2, and so on.

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 11gR1 (11.1.1.2.0), the abstract WSDL of an ABCS that is being developed should be accessed from MDS. The exceptions to this rule are:
   - The EBS reference WSDLs that define PartnerLink types.
   - Participating application reference WSDLs that define PartnerLink types.
   - Adapter WSDLs that are generated by 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 (Application Business Message [ABM] schemas) should be accessible from MDS. They should not be a part of each ABCS project.

For ABCS, abstract WSDLs are located here in MDS:

\texttt{AIA\_MetaData/AIAComponents/ApplicationConnectorService\_Library/<PRODUCT\_CODE> /<Version Number>/<Service Type>}

Possible values for \texttt{<Service Type>} are RequesterABCS and ProviderABCS.

Possible values for \texttt{<Version Number>} - V1, V2, and so on.

For Composite Business Processes (CBPs) and EBFs, abstract WSDLs are located here in MDS:

\texttt{AIA\_MetaData/AIAComponents\_BusinessProcessService\_Library/<Service Type>}

For example:

\texttt{AIA\_MetaData/AIAComponents\_BusinessProcessService\_Library\_CBP}

\texttt{AIA\_MetaData/AIAComponents\_BusinessProcessService\_Library\_EBF}

---

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

1. **Configure the ComponentType file.**
   
   a. Make changes to the componenttype files to point them to the abstract WSDLs in the MDS.

   The attribute \texttt{ui:wsdlLocation} in the Service element should point to the abstract WSDLs in the MDS.

   For example:

   \begin{verbatim}
   <service
   ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/ApplicationConnectorService\_Library/SampleSEBL/RequesterABCS/SamplesCreateCustomerSiebelReqABCSImpl.wsdl"
   name="SamplesCreateCustomerSiebelReqABCSImpl">
   \end{verbatim}

   b. If the composite has a reference, then the attribute \texttt{ui:wsdlLocation} in the reference element should point to abstract WSDLs in the MDS.

   For example:

   \begin{verbatim}
   <reference name="SamplesCreateCustomerPartyPortalProvider"
   ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/ApplicationConnectorService\_Library/SamplePortal/V1/SamplesCreateCustomerPartyPortalProvider.wsdl">
   </reference>
   \end{verbatim}

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

For example:

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

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

For example:

```
<service
ui:wsdlLocation="oramds:/apps/AIAMetaData/AIAComponents/AIServiceLibrary/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:

```
<wSDL:import
namespace="http://xmlns.oracle.com/EnterpriseServices/Core/CustomerParty/V2"
```

d. Populate attributes of the binding.ws element of the reference services in the composite.xml file. In the composite.xml file, under the element <reference>, the Service Constructor generates empty attributes, port, and location for the element <binding.ws>. Populate the attributes with relevant values as specified in the following list.

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

```
<binding.ws port="[Namespace of the Service as defined in the wsd1]#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]"/>
```

**Note:** The name of the Service is the value of the attribute definition/name in the abstract WSDL.

This follows from the naming conventions of Service name in the ABCS Composite. For example, 'name of the service' is <name of the composite>, which in turn is the value of the attribute ‘name’ of the element ‘definitions’ 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 attribute ‘name’ of the element
```
'definition’ in the WSDL]/[Name of the Porttype element as given in the WSDL]?WSDL

For example:

```xml
<binding.ws
```

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

```xml
<binding.ws
 port="[Namespace of the Service as defined in the wsdl]#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.ws
```

3. Move these WSDLs to MDS.

For more information, see How to Update MDS.

### 7.3. How to Migrate 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 11gR1 (11.1.1.2.0), this file is placed in MDS.

The installation of AIA 11gR1 (11.1.1.2.0) includes a default configuration file and you must manually migrate any system and module settings from the AIA 2.4 or 2.5 environment. See the following procedure for information about how to update AIAConfigurationProperties.xml with custom Service Configuration elements.
Service-related settings are taken care of by the AIA Migration Utility, which migrates them into a file called AIAServiceConfigurationProperties.xml for each migrated service.

The AIA Install Driver handles the merging of 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. Browse to the $AIA_HOME/aia_instances/$INSTANCE_NAME/config folder.
   a. Open AIAConfigurationProperties.xml and add the custom elements under respective Service Configurations.
   b. Open the deployment plan file, UpdateMetaDataDP.xml. Update the file UpdateMetaDataDP.xml by inserting include tags for AIAConfigurationProperties.xml that you want to add to the MDS:
      
      Add the following:
      
      ```xml
      <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. Browse to the $AIA_HOME/aia_instances/$INSTANCE_NAME/bin folder.
3. Source the file aiaenv.sh by executing the following command:
   
   ```bash
   source aiaenv.sh
   ```
4. Browse to $AIA_HOME/Infrastructure/Install/scripts. Execute the script UpdateMetaData.xml by using the following command:
   
   ```bash
   ant -f $AIA_HOME/Infrastructure/Install/scripts/UpdateMetaData.xml
   ```

### 7.4. How to Migrate Custom Extensions to EOL, AOL, and ESL

Any custom extensions to the Enterprise Object Library, Application Object Library, and Enterprise Service 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.
7.5. How to Update MDS

You must repeat this procedure every time a file needs to be added to MDS.

To update SOA-MDS -- apps/AIAMetaData:

1. Browse to the folder at $AIA_HOME/aia_instances/$INSTANCE_NAME/bin.
2. Source the file aiaenv.sh by executing the following command:

   source aiaenv.sh

3. Browse to the folder at $AIA_HOME/aia_instances/$INSTANCE_NAME/config and open the deployment plan file, UpdateMetaDataDP.xml.
4. Update the file UpdateMetaDataDP.xml by inserting include tags for each resource group that you want to add to the MDS:
   a. To upload all the files under "AIAMetaData", add the following:
      <include name ="AIAMetadata/**"/>
   b. To upload the files copied to "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. Browse to AIA_HOME/Infrastructure/Install/scripts. Execute the script UpdateMetaData.xml by typing the command:

   ant -f UpdateMetaData.xml
8. Migrating AIA 2.4 and 2.5 Business Service Repository Artifacts to Oracle Enterprise Repository Artifacts

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

In the AIA Foundation Pack 11gR1 (11.1.1.2.0) release, this functionality is consolidated in Oracle's SOA repository, Oracle Enterprise Repository. Oracle Enterprise Repository brings 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 11gR1 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 11gR1 (11.1.1.2.0) (on Oracle SOA Suite 11g). For instance, AIA does not provide out-of-box integration between AIA 2.4 and 2.5 (and therefore maintaining an ensuing BSR instance) on the 10g SOA Suite and 11g Oracle Enterprise Repository. It does not make sense to remain on AIA 2.4 or 2.5 (and therefore maintaining an ensuing BSR) on the 10g SOA Suite, while operating an 11g Oracle Enterprise Repository.

- For prebuilt AIA 2.4 and 2.5 contents released by Oracle, you need not do anything. The content can be published into Oracle Enterprise Repository via the AIA Solution Pack.
- For custom-built content, you must harvest them into Oracle Enterprise Repository on their own by using the AIA Harvester and following the samples and documents delivered in AIA Foundation Pack 11gR1 (11.1.1.2.0).
  - We assume that custom-build content (EBS.wsdl, EBM.xsd, EBO.xsd, ABCS.wsdl, and composites, for example) were migrated and annotated in accordance with AIA Foundation Pack 11gR1 (11.1.1.2.0) extension and annotation standards.
  - You can harvest your custom-built interfaces dynamically by running the AIA Harvester.


Also note that Oracle Enterprise Repository is about SOA asset lifecycle management, whereas, AIA Foundation Pack 11gR1 (11.1.1.2.0) components, such as the Project Lifecycle Workbench, focus on facilitating development lifecycle management from functional definition and process decomposition to bill-of-material autogeneration 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 the 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 Oracle Enterprise Repository.
9. Cutting Over From a 10g- to an 11g-Based Production Environment

Cutting over from a 10g production environment to an 11g-based environment is a carefully planned exercise. The methodology must be determined by the project team on site, taking into account actual requirements.

Use the following steps as general guidelines for your cutover:

1. Set up an AIA Foundation Pack 11gR1 (11.1.1.2.0) environment.
2. Set up a fresh cross-reference database and AQ JMS queues.
3. Deploy all migrated services to the new 11g production environment.
4. As a planned outage, suspend and stop all events and applications triggering the pushing of messages to the 10g-based AIA 2.4 or 2.5 environment.
5. Allow all messages in flight in the 10g-based AIA 2.4 or 2.5 environment to complete.
6. Modify the applications to divert messages to the new 11g environment based on AIA Foundation Pack 11gR1 (11.1.1.2.0).
7. Move the cross-reference data from the 10g cross-reference database to the 11g cross-reference database.
8. Restart the events and applications triggering the pushing of messages to the 11g-based AIA Foundation Pack 11gR1 (11.1.1.2.0) environment.
10. What You Need to Know About a SOA 11g Upgrade

After upgrading 10g projects using the AIA Migration Utility:

1. Manually change ora:getUserProperty() to ids:getUserProperty() as a part of post-migration steps as this function was deprecated in 10.1.3.4.

2. Verify the properties file name when upgrading projects that use correlation sets in asynchronous services.

For more information, see Oracle Fusion Middleware Upgrade Guide for Oracle SOA Suite, WebCenter, and ADF, “Considerations When Upgrading Oracle SOA Applications,” Verifying the Properties File Name When Upgrading Projects That Use Correlation Sets in an Asynchronous Service.

3. Retire the following artifacts:
   a. Service.xml
   b. System group files
   c. PIP installation scripts
   d. 10g fault policies are not relevant to 11g. You can use and recreate fault policies for 11g.

For more information, see How to Migrate Fault Policies.
Index

abstract service WSDLs
  moving into Oracle Metadata Services repository, 34
adapter services
  migrating, 19
AIA Configuration Properties file
  migrating, 37
AIA Migration Utility
  overview, 15
  using, 15
AIA service artifact
  accessing Oracle Metadata Services repository from, 33
AIAConfigurationProperties.xml
  configuring for error handling, 29
  migrating, 37
AIAServiceConfigurationProperties.xml
  migrating, 37
AOL. See Application Object Library
Application Object Library
  migrating custom extensions to, 38
BSR. See Business Service Repository
Business Service Repository artifacts
  migrating to Oracle Enterprise Repository artifacts, 41
cross-references
  migrating, 25
custom error handlers
  migrating, 30
custom extensions to Application Object Library
  migrating, 38
custom extensions to Enterprise Object Library
  migrating, 38
custom extensions to Enterprise Service Library
  migrating, 38
custom subscribers
  migrating, 32
domain value maps
  migrating, 25
domain value maps
  migrating, 25
e-mail notifications
  configuring Oracle User Messaging Service for, 30
Enterprise Business Services
  migrating, 21
Enterprise Object Library
  migrating custom extensions to, 38
Enterprise Service Library
  migrating custom extensions to, 38
EOL. See Enterprise Object Library
ero-handling
  configuring AIAConfigurationProperties.xml for, 29
  migrating, 27
error notification setup
  migrating, 29
ESL. See Enterprise Service Library
extension services
  migrating, 23
fault policies
  migrating, 27
MDS. See Oracle Metadata Services repository
Index

updating, 39
metadata
loading to Oracle Metadata Services repository, 33
Migration Utility
overview, 15
using, 15
OER. See Oracle Enterprise Repository
Oracle Enterprise Repository artifacts
migration Business Service Repository artifacts to, 41
Oracle Fusion Middleware environment
migration overview, 7
Oracle Metadata Services repository
accessing from the AIA service artifact, 33
loading AIA metadata to, 33
moving abstract service WSDLs into, 34
Oracle User Messaging Service
configuring for email notifications, 30
prerequisites, 11
production environment
cutting over from 10g to 11g, 43
provider Application Business Connector Services
migrating, 20
requester Application Business Connector Services
migrating, 22
service code
migration overview, 8
transaction configurations
migrating, 28
user email addresses
configuring, 30
WSDLs
moving into Oracle Metadata Services repository, 34