

An Oracle White Paper
Updated - February 2011

AIA 11gR1 Installation and Deployment – Flexibilities, Topologies, Architecture and Tools

Introduction	1
Scope of this document	3
AIA Installer – Overview of Installation and Deployment Modes	4
AIA Installer - Installation and Deployment Topologies	6
Database Schema Options	7
SOA Server Selection.....	15
Deploying on a Remote Server	15
Deploying on a SOA Cluster	16
Distributed Deployment	20
AIA Deployment Architecture.....	22
Challenges in Multiple Service deployment in a SOA landscape .	22
How AIA Deployment Architecture addresses these challenges ..	23
Deploying custom AIA content - Methodology and tooling	25
What to Deploy (BOM and Deployment Plan).....	25
Where to deploy (AIAInstallProperties.xml).....	28
How to deploy (AIA Installation Driver)	29
Conclusion	31

Introduction

SOA adoption is the buzzword today in every organization's IT roadmap. As organizations scale the SOA maturity model from initial adoption to building agile and optimized enterprise business services, it is crucial to unify and optimize different stages of the SOA lifecycle in order to gather momentum in the right direction.

Oracle Application Integration Architecture (AIA) is a fully SOA-based product for Enterprise Application Integration providing you all vital ingredients, end-to-end, for a successful SOA Adoption.

Oracle Application Integration Architecture (AIA) Foundation Pack (FP) provides the architecture blueprint and standards-based canonical business objects to design true SOA Integrations. AIA FP also provides the framework for rapidly implementing, testing, deploying, monitoring and governing business integrations allowing you to jumpstart your SOA Initiative or to attain higher levels of maturity for your existing SOA adoption.

While traditionally much focus is restricted to phases of the SOA lifecycle concerning design, implementation and testing of services, AIA Foundation Pack also considers in an inclusive manner, how these services are effectively deployed, operated, monitored and governed in a manner that provides true value to business.

This paper will lay focus on the AIA Installer, the Deployment Tools and Architecture that AIA FP provides, that will enable you to deploy services and other content in an easy, controlled and repeatable manner.

AIA FP is an installable product and can be deployed against Oracle SOA Suite deployments. AIA Installer ensures that all Oracle-delivered AIA FP content, tools etc described earlier is installed, configured and ready to use at the end of its run.

Although AIA FP does not deliver the actual services that implement any specific business integration, it is important to understand the options and flexibilities provided during the FP Installation and determine the appropriate topology (single silo-ed, clustered, distributed etc.), because it consequently determines the way integration content (like services, JMS etc) will later be deployed and operated on top of the FP layer. This paper will initially discuss that.

This Integration content itself could be Oracle AIA Process Integration Packs - PIPs (Oracle delivered pre-built, extensible and deployable integration content meant for specific business processes between specific Enterprise applications) or custom-built Integration content built by you for integrating your different Organization's Application assets.

The paper will later take you through the Architecture and tools that you can use to create your own deployment plans, for custom services built by your Organization using Foundation Pack. While Oracle delivered PIPs are out of the box content and can be automatically deployed and configured ready to run, custom services built by your Organization on top of Foundation Pack must also be deployed/configured with relatively the same ease as Oracle PIPs.

To this effect AIA FP provides a compelling approach. AIA FP exposes all tools and methodologies to build your own deployment plan and execute them against your target sever with absolute ease and configurability.

The architecture also accounts for the fact that service design, implementation and maintenance is a continuous activity and hence deployment and upgrade of services need to happen periodically, rapidly and consistently across multiple environments in a controlled manner.

Scope of this document

This paper can be broadly split into **three** parts.

The **first** part intends to provide an overview of the flexibilities that AIA 11gR1 Installer offers. The paper also provides an insight into when and how you could use these flexibilities to achieve the deployment topology that best suits your organization.

The **second** part will focus on the AIA deployment architecture and the associated tools that allow you to easily construct your own deployment for the AIA content that you created or customized.

Lastly, the paper provides you a brief overview on migration strategies that you could employ for your AIA environments.

Note: This document is not intended to be a step-by-step Installation or Implementation guide. It is only to provide you an idea of the options available through the Installer and the associated tooling and how these flexibilities can be leveraged to achieve the desired results.

For actual steps please refer to the AIA Foundation Pack Installation Guide (latest version can be obtained from <http://support.oracle.com>). Also refer to the known issues/release notes at oracle.com, to check for deviations if any.

AIA Installer – Overview of Installation and Deployment Modes

AIA Installer provides a wizard driven installation approach for installing AIA Foundation pack and other AIA products. The Installer prompts users to select products to be installed and deployed and to input server details and required configuration details. The user has flexibility in providing the server details (SOA Server and database) and in configuring the installation as per the topology requirements.

There are 5 different modes in which the AIA Installer based installation/deployment can be performed (modes are conceptual and not represented on the Installer). This section provides an overview of all modes. Subsequent chapters in this guide will elaborate on specific steps required to achieve these modes.

1. Basic Installation and deployment mode

This mode represents the basic single server installation of AIA. If this is the first time you are installing AIA FP, you might want to begin with Basic Installation mode.

For this installation mode you would need a WebLogic Server with Admin Server and SOA Server configured in the same server.

You will be installing AIA (creating AIA_HOME directory) on this server and deploying the artifacts to the SOA Server

For a basic installation you will typically configure all AIA database schemas on the same database server although you can still choose different database connections (which will be elaborated in later sections)

This installation mode will deliver a fully functional single server installation of AIA.

2. Cluster Deployment mode

This Installation mode is to deploy AIA on an existing SOA Suite Cluster to derive high-availability, effective workload distribution, increased throughput, performance etc.

SOA Suite Cluster has to be successfully setup according to the SOA Suite 11g Enterprise Deployment Guide (and tested with samples) before proceeding with AIA Installation on the cluster.

The Installer will automatically detect the presence of a SOA Cluster and prompt the user with fields for appropriate additional information.

When providing database information, users can choose to connect to RAC databases.

At the end of the deployment, AIA products would be deployed to all nodes of the SOA Cluster. Additional nodes can be added to/removed from the SOA Cluster any time without the need of additional configuration.

3. Remote Deployment mode

This Installation mode allows users to deploy AIA to a SOA server different from where the installation was launched

In this case the AIA_HOME will be created on the server where the installer is launched but the actual deployment can be on a different physical server.

This mode thus allows users to use a single AIA_HOME and target deployments to several SOA servers.

The Installer user experience will be almost the same as that of basic install.

4. AIA Files only installation mode

This installation mode does no AIA configuration or AIA deployment. Choosing this option on the installer will just deliver all AIA artifacts to the directories of the physical server.

This mode is typically used to browse Oracle AIA delivered content like EBOs and process models even before AIA FP Installation. This mode doesn't require the presence of Weblogic server and so users can download AIA content to even their personal computers.

5. Silent Installation mode

This installation mode is to be used when you want to bypass the Graphical User Interface of the Installer and instead provide the necessary input for the installer using a response file. The response file is a text file containing the variables and the corresponding values required for installation. AIA installer ships standard response file templates and also allows users to easily generate their own response file templates using the Installer.

This allows the installation to be performed without manual intervention and is most useful when you need multiple installations of AIA with the same or similar parameters on multiple servers.

Besides providing these modes at a higher level, Installer also provides several options when providing the actual server information (where you plan to deploy) thereby allowing you to achieve varied topologies. These flexibilities and topologies will be described next.

AIA Installer - Installation and Deployment Topologies

Using different modes of the AIA Installation, several deployment topologies can be achieved by leveraging the flexibilities of the installer and installation tools

The topologies arise through variations in

- Database Schema details
- SOA Server selection

The flexibilities that each of these present will be explained followed by some sample Installation topologies.

In the remainder of this document you will notice two terminologies being used – AIA Installation and AIA Deployment. Although used interchangeably at times, there is a distinct difference.

AIA Installation refers to the delivery of all AIA content (AIA_HOME) to the server whereas AIA Deployment refers to the deployment of AIA products (like AIA foundation pack or AIA demo or PIPs) or deploying specific AIA artifacts to specified servers.

For every physical Weblogic Server there can be one installation of AIA. This installation is identified by an AIA_HOME, which is an ORACLE_HOME created as a part of the AIA Installation. The AIA_HOME is where all the AIA content is delivered as files. An AIA_HOME should be maintained as the source of truth and identifies the availability of an Installation during patching, upgrades etc.

Database Schema Options

AIA Installer provides you multiple options while configuring the database schemas used by AIA. The schema types that you can configure include

- AIA Schema (used by AIA Error Handling, CAVS, System Setup)
- Cross Reference Schema
- AIA Lifecycle Schema (used by AIA Project Lifecycle Workbench)
- JMS Schema

Each of these schema types is AIA specific and so has to be configured as a part of AIA Installation. Using AIA Installer for each schema type, you can provide the details of the actual schema to be created or you can even specify already existing schemas to which you want to connect to (either created through another AIA installation or created manually through SQL scripts).

AIA Installer provides a mechanism to configure each schema type one at a time or more than one schema type at the same time. When performing this, Installer provides the option to provide the same physical database server for all AIA schemas or optionally each schema can be configured to be on a different database server.

The following figure shows the above options available through the installer. Later chapters will give detailed information on how to complete information on this screen.

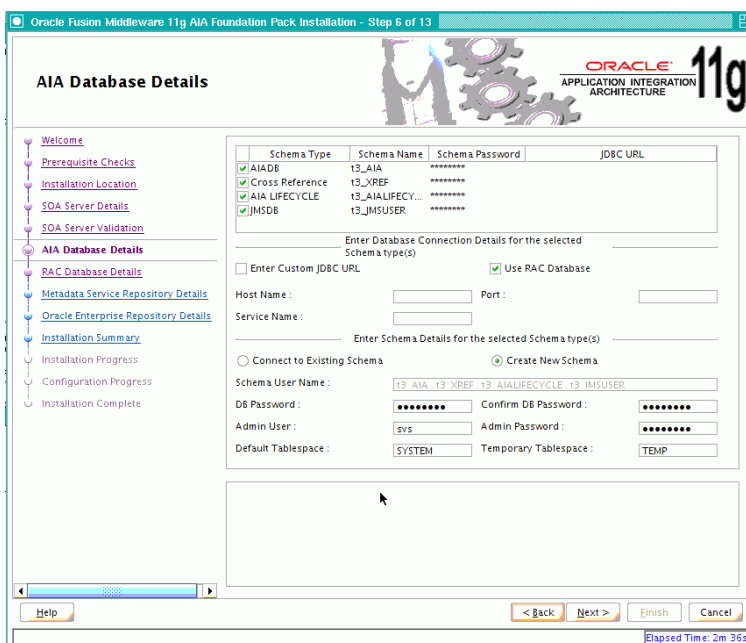


Figure 1: Sample AIA Database Details Screen on the AIA Installer

The figure below is an illustration of how AIA installations can be created using the above screen to utilize database schemas running on single or multiple physical servers or any other combination in between.

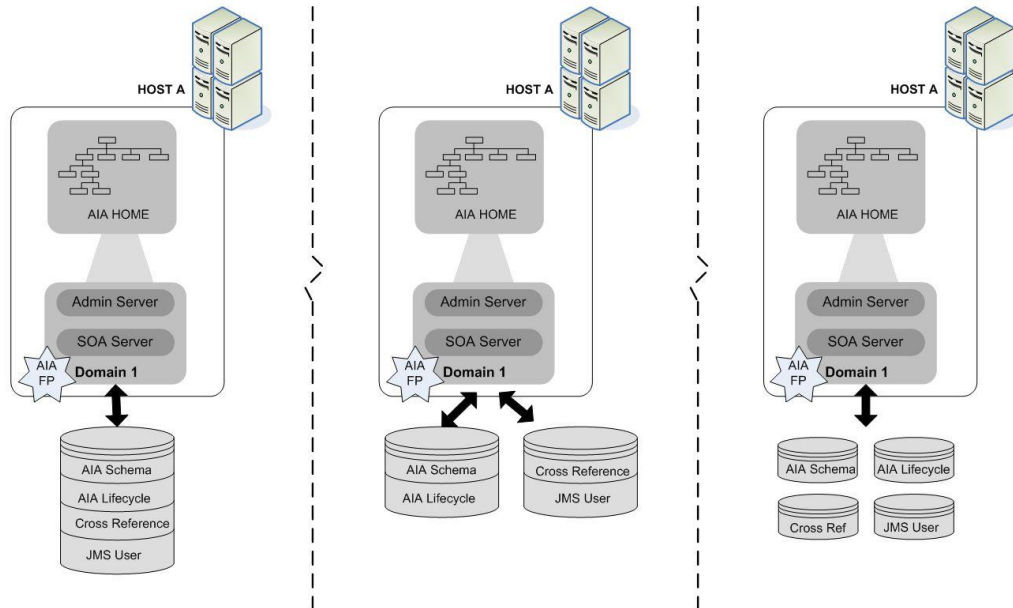


Figure 2: Possibilities that arise based on single database server or multiple servers

The above flexibility opens up several topology options from a database perspective. We will discuss a few common scenarios.

1. Multiple AIA installations sharing the same AIA Lifecycle database -

Typically every AIA installation has its own set of AIA schemas however it is not an uncommon scenario for AIA installations to share the same physical schema.

One example is AIA lifecycle schema used by AIA Project Lifecycle Workbench. The Workbench is for orchestrating the development-time activities and so a common Lifecycle Workbench Database schema is desirable. This topology is often useful in a distributed development environment; you may have different teams working on different AIA installations, yet they share a same AIA Project Lifecycle Workbench, which guide them through a common release/development cycle.

In figure 3, the Foundation Pack installations corresponding to AIA1 and AIA2 can each have its own AIALifecycle Schema viz. AIA1_LIFECYCLE and AIA2_LIFECYCLE. Alternatively when creating AIA2 Installation, it can be configured

to 'Connect to' AIA1_LIFECYCLE using the AIA Database Details screen presented above, instead of creating a new schema for itself.

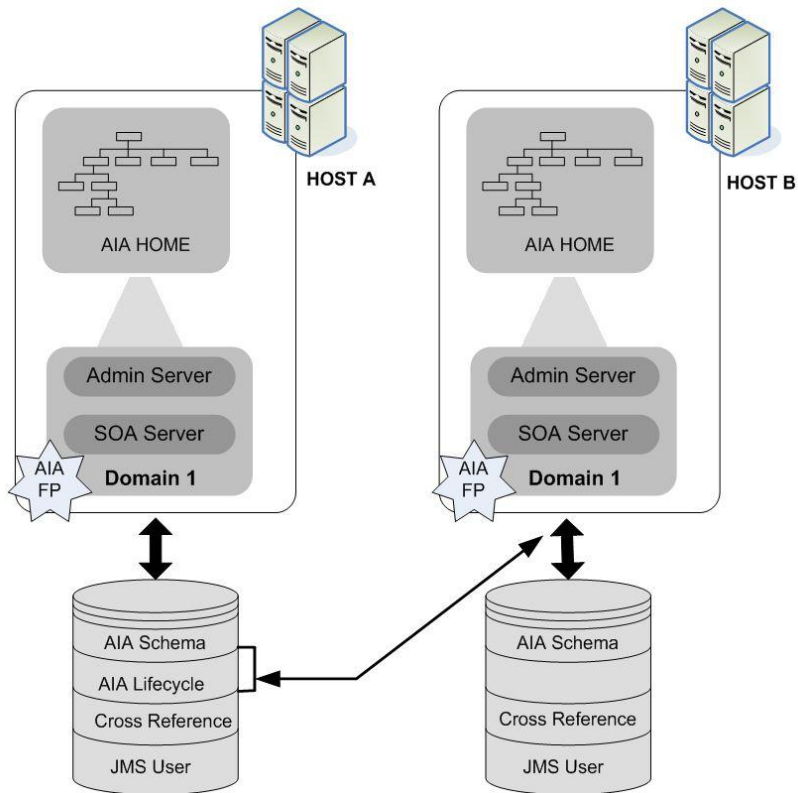


Figure 3: Multiple installations sharing the same AIA Lifecycle Database

To do this, while running the installer the second time to install AIA2 follow these steps

- a. Choose the AIA Lifecycle schema checkbox alone in the table region of the database details screen.
- b. Click inside the cell corresponding to 'AIA Lifecycle' Row and 'Schema Name' column.
- c. Now this cell will be editable. Change the name to AIA1_AIALIFECYCLE. The default value would be AIA2_AIALIFECYCLE. This is because the AIA

Instance name provided in the first screen of the installation is appended to the schema name automatically.

- d. In the region below enter the database connection details.
- e. In the Enter Schema Details region, choose 'connect to existing schema' and provide the password of the AIA1_AIALIFECYCLE schema that was created during the earlier run of the installer when creating AIA1 Instance. This schema is now configured to reuse an existing schema
- f. Now uncheck the AIA Lifecycle row in the table and choose all other schema types.
- g. Provide connectivity information, desired password and SYS user credentials at one go to create other schemas.

Note: In this scenario only AIA Lifecycle schema is shown as shared. Other schema types are shown to have different physical schemas one for each AIA Installation. However they can also be shared. AIA Lifecycle Schema is quoted only an example, as it is a common scenario.

Relationship to OER

Oracle Enterprise Repository (OER) is Oracle's SOA repository that provides governance to all types of 11g SOA assets. AIA leverages OER to achieve SOA visibility to the pre-built and custom-built SOA assets across the AIA's development lifecycle. While the source of truth is the AIA installation (AIA_HOME or the linked source control for development time) and AIA deployment (AIA Instances during runtime), OER can capture and reflect the truth in a governance perspective (available throughout the organization) by way of harvesting across several AIA prescribed harvesting points in the development lifecycle.

OER is about SOA asset lifecycle management, whereas, AIA Project Lifecycle Workbench mainly facilitates development lifecycle management from functional definition and process decomposition to bill-of-material auto-generation and deployment plan construction.

It must be noted that OER is an optional component during AIA installation and it is possible to operate AIA (including the Lifecycle Workbench) without OER.

However, users often find it is convenient to manage and reuse the many types of SOA artifacts produced throughout AIA development lifecycle in SOA Repository. If they choose to do so, they may operate the AIA Project Lifecycle Workbench in concert with OER. In such cases, users may deploy the AIALifecycle schema alongside OER Database schema.

Typically to promote reuse, visibility, and SOA values in an organization, OER is centralized. In a similar manner, AIA highly recommends centralizing AIALifecycle database to be shared across different AIA implementations.

2. Multiple AIA Installations using the same XREF (Cross Reference) schema

This scenario is similar to the previous one with only one main difference. The cross-reference database is not only shared but is also physically present in a different database server.

Cross-reference information is specific to cross application communication and is independent of the specific integration flow (or PIP) that uses it. For example a cross-reference for storing Order ID reference between a specific Siebel installation and specific E-Business Suite installation is applicable for all business integrations built between these applications that use Order ID. Hence AIA1 and AIA2 in figure 4 below would want to refer to the same cross-reference information.

Another reason why this would be performed is because cross-reference data typically grows rapidly and hence it is desirable to provision a different database server for it.

To achieve this topology, when creating AIA1, all other schemas are configured such that they are in one database whereas for XREF alone a different database is mentioned. When creating AIA2, it can be configured to 'Connect to' this XREF using the Database details screen, instead of creating a new schema for XREF.

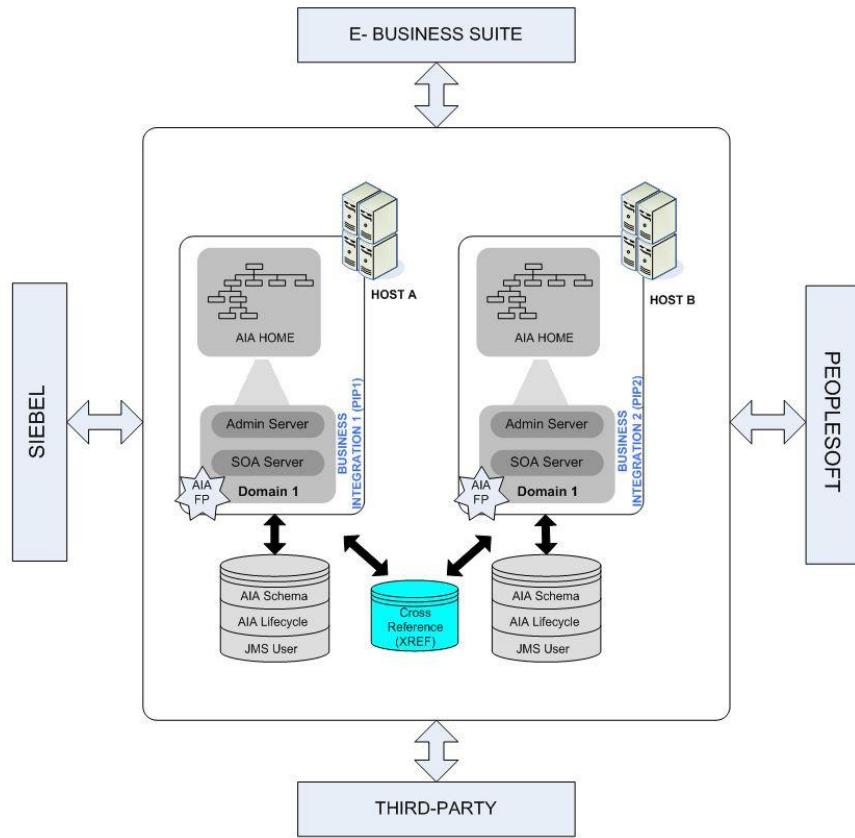


Figure 4 Centralized cross reference database

3. Configuring one JMS schema per JMS entity

For volume and performance reasons it is sometimes required to have multiple physical schemas for the same schema type serve a particular AIA Installation. An example would be JMS schemas. Multiple JMS schemas may exist, say, one for each entity.

This configuration is not directly supported when using the installer but can be achieved when performing a manual deployment. This will be further described in later sections of this paper.

4. Using a RAC database

Oracle RAC databases can be used (typically in production environments) to achieve high availability, failover, reliability, performance and scalability.

It is natural therefore to choose to use RAC databases where there is high volume data and heavy runtime usage, for example XREF database.

Typically AIA artifacts delivered on fusion middleware tech-stack are agnostic of the database type (single node or RAC). Using a RAC is therefore an install time option. Hence while configuring AIA schemas using the AIA Installer it is required to provide RAC database server details when creating or connecting to database servers.

Note: This document is **not** intended to be a step-by-step Installation or Implementation guide. It is only to provide an idea of the options available through the Installer and how these flexibilities can be leveraged.

For actual steps please refer to the AIA Foundation Pack Installation Guide (latest version can be obtained from <http://support.oracle.com>). Also refer to the known issues/release notes at oracle.com, to check for deviations if any.

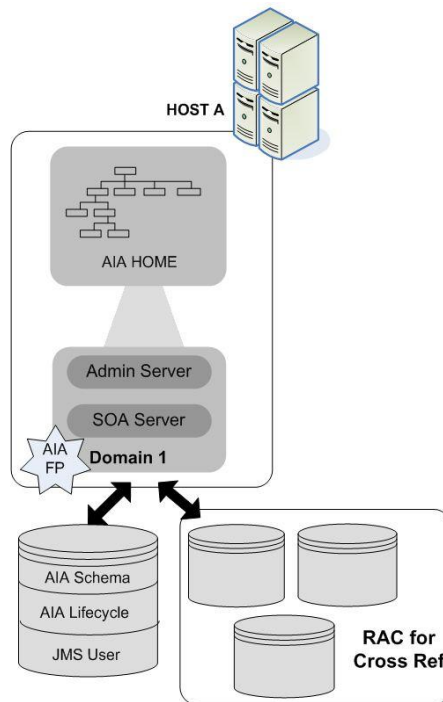


Figure 5: Usage of RAC for Cross Reference database

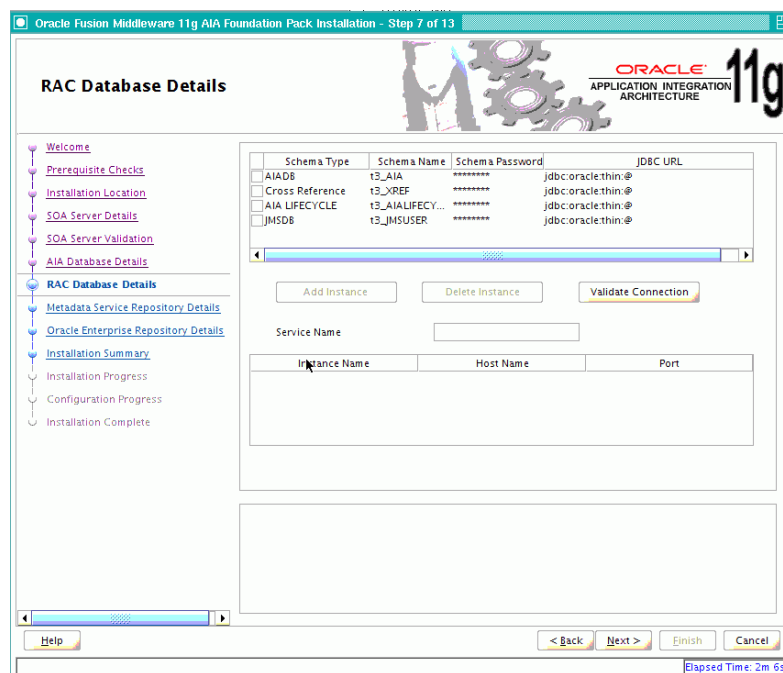


Figure 6: Establishing RAC connections for AIA schemas

5. Requirement to create database schemas with advanced database options

When creating new schemas, the Installer allows users to choose the schema name, password and also the default and temporary table space on which these schemas have to be created.

In addition to these options it might be desirable that AIA Schemas and other DB artifacts be created in a specific manner based on organization standards and advanced database options.

In this case DBAs are free to create AIA schemas and the underlying artifacts using the SQL scripts shipped with the AIA Installation bundle under Disk1/misc/sql of your AIA Installation CD

These SQL scripts have to be executed for the desired schemas before launching the Installer. There is one SQL script per schema for creation (to be executed using the SYS user) and one script per schema for populating the schema (to be executed as the user that was just created). It is not required to create all schemas manually using this process. Once the creation of schemas and the underlying database artifacts is complete, launch the installer and in the database details screen, choose the 'connect to' option to connect to the schema that was created. For other schema types for which the SQL scripts were not executed manually choose the 'Create New Schema' option on the Installer.

SOA Server Selection

Selection of the SOA Server on which deployments can be done opens up many topologies.

Deploying on a Remote Server

Apart from choosing the SOA Server on which the installation is being done, Remote SOA servers can also be chosen for deployment. This has been explained in the earlier chapter

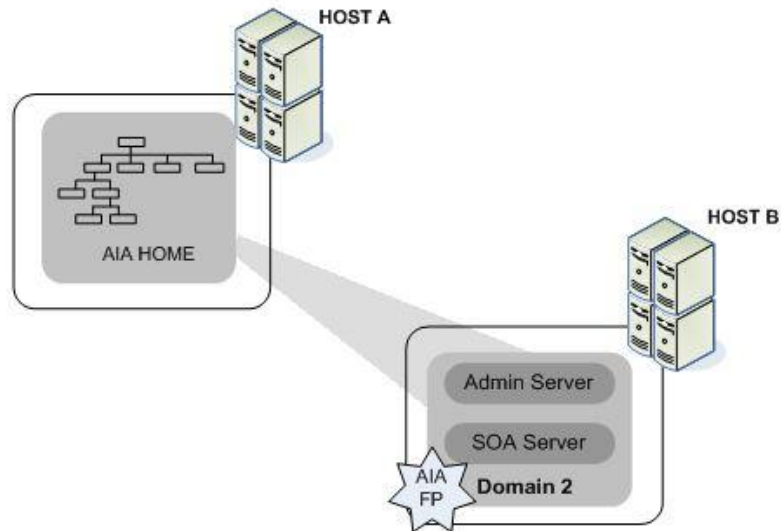


Figure 7: Depiction of remote Installation topology of Foundation Pack

In this case, it is not necessary to have the SOA Server deployed on the server where AIA Installer is launched. For example in the diagram shown above, it is only necessary to have the latest JDeveloper installed on HOSTA, where the installer is being launched.

The following figure depicts the regular SOA Server details page and shows how Remote Server option can be used.

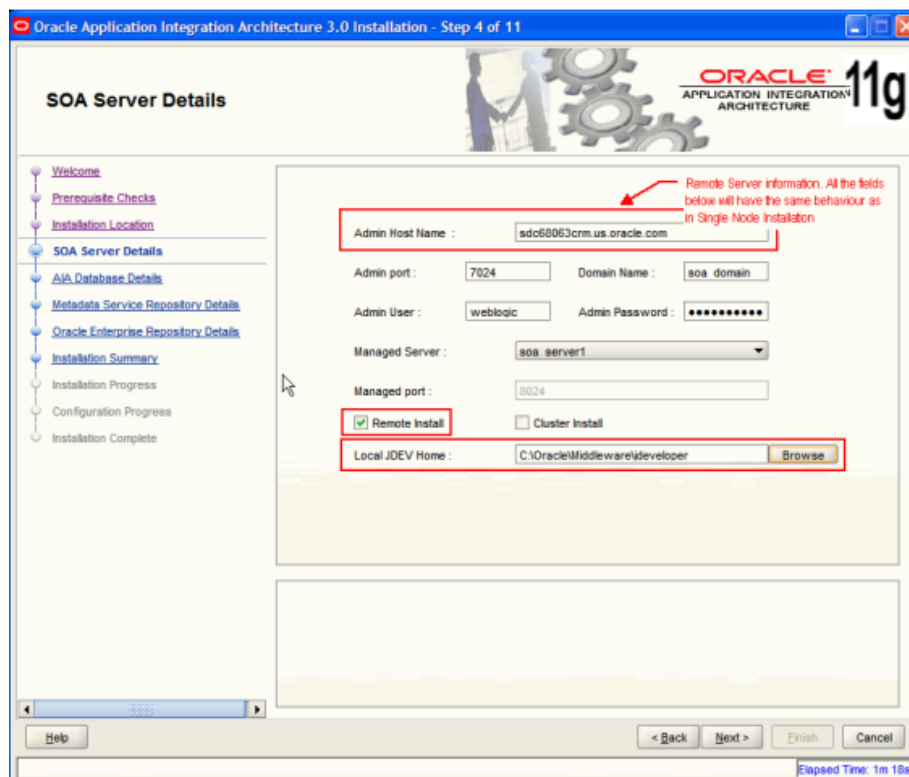


Figure 8: SOA Server details page depicting Remote Install use case

Deploying on a SOA Cluster

AIA can also be deployed to a **SOA Cluster**. This was briefly explained earlier. A successful SOA Cluster setup is an absolute pre-requisite for AIA deployment.

Please follow Oracle Fusion Middleware – [Enterprise Deployment Guide for Oracle SOA Suite 11g](#) to set up the SOA Cluster.

After performing the SOA Cluster setup as per instructions in the above guide, ensure that the SOA cluster has been well set up by accessing appropriate URLs (like EM Console) from all hosts participating in the cluster, by deploying sample artifacts shipped along with SOA Suite bundle and by performing standard failover tests on the SOA Cluster.

AIA Installer allows you to deploy AIA directly to a SOA cluster. Cluster nodes can be present in the same physical server under the same domain or can belong to another physical server. While the former only provides increased throughput, having each node in a different server provides high-availability and is the recommended topology as per Enterprise deployment Guide. Refer chapter 1 of the Enterprise deployment Guide.

Deployment of AIA to the cluster can be achieved in one go using the AIA Installer.. The figure below is how AIA artifacts relate to a sample SOA Cluster topology as defined in SOA Enterprise Deployment Guide.

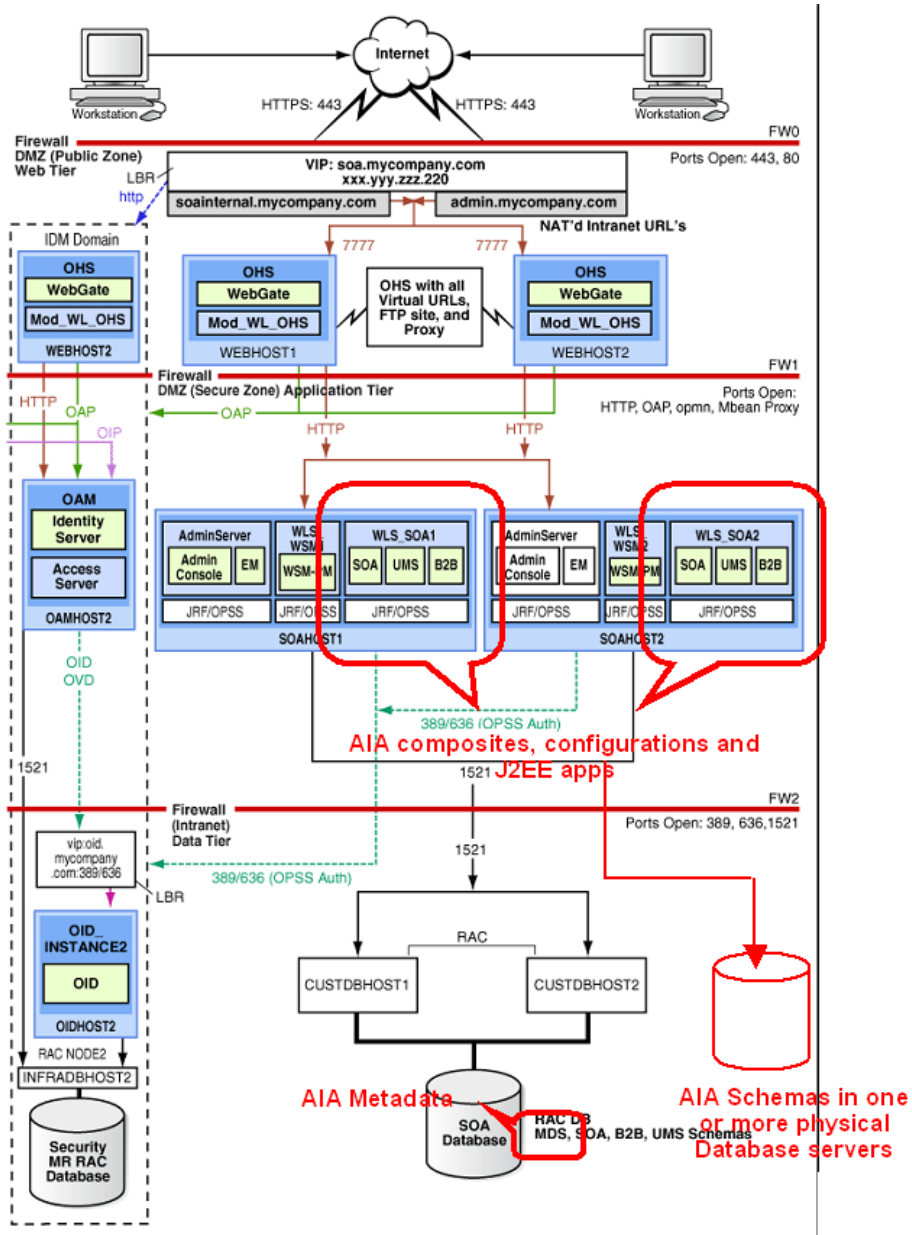


Figure 9a: AIA deployment on SOA Cluster

Below is how AIA Installation and deployment concepts that we've seen so far relate to a SOA Cluster

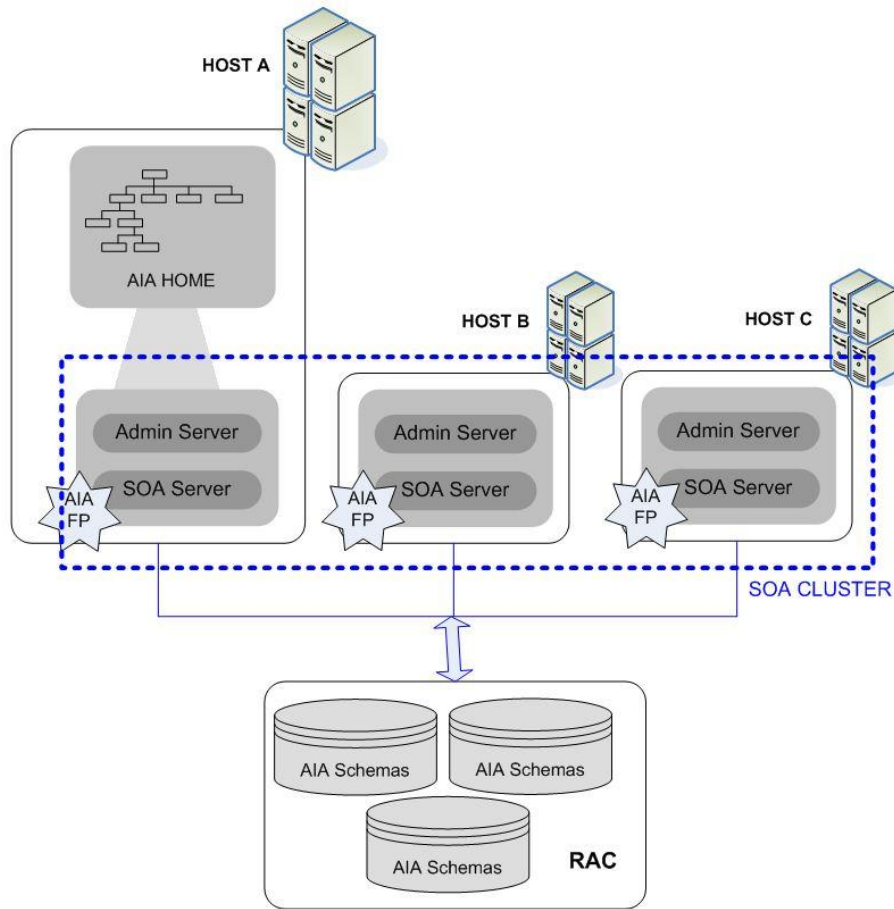


Figure 9b: AIA deployment on SOA Cluster

In general RAC database is recommended for such enterprise deployments. A RAC database can be specified using the option in the Database details screen explained earlier in this document.

When providing the Server details on the installer, the cluster name automatically appears in the list of available managed servers. On choosing the same, additional information viz. proxy URL and proxy host is also gathered.

Oracle Fusion Middleware 11g AIA Foundation Pack Installation - Step 4 of 12

SOA Server Details

Welcome
 Prerequisite Checks
 Installation Location
SOA Server Details
 SOA Server Validation
 AIA Database Details
 Metadata Service Repository Details
 Oracle Enterprise Repository Details
 Installation Summary
 Installation Progress
 Configuration Progress
 Installation Complete

Admin Host Name :

Admin port : Domain Name :

Admin User : Admin Password :

Managed Server : Cluster Name appears automatically

Managed port :

Remote Install Cluster Install

Local JDEV Home :

Cluster Proxy Host :

Proxy Port :

Elapsed Time: 1m 38s

Figure 10: SOA Server details page depicting Cluster deployment use case

Note: This document is not intended to provide step-by-step instructions on how to deploy to a SOA Cluster. Refer to the AIA 11gR1 Installation Guide for detailed instructions.

Adding new nodes to the cluster

Additional nodes can be added easily to an existing SOA cluster and there are almost NO extra steps required to configure AIA on those node.

i.e., the following are some of the components deployed as a part of AIA and how they are automatically made available in the new node

- AIA Composites – Due to common MDS for all nodes of the cluster these composites are instantly available in the new node
- AIA configurations, EOL etc – These are also stored in MDS and so they are centrally referenced
- AIA XREF and other database artifacts – The new node will have automatically be configured to have the same data sources as the other nodes and because AIA always uses data sources, this connectivity is also established by default

Essentially, AIA cluster can scale almost instantly.

As a combination remote-cluster installation is also possible i.e. in figure 9b, instead of the installer being launched on Host A, it could have been any other Host X.

Distributed Deployment

It is also possible to deploy AIA in a distributed manner. This means the same AIA can be deployed to more than one non-clustered SOA Server in a different domain on the same physical server or even a remote server. This is useful when wanting to maintain a single AIA instance having one or more business process integrations (or PIPs) but there is a need to distribute the artifacts to different servers based on business or performance requirements.

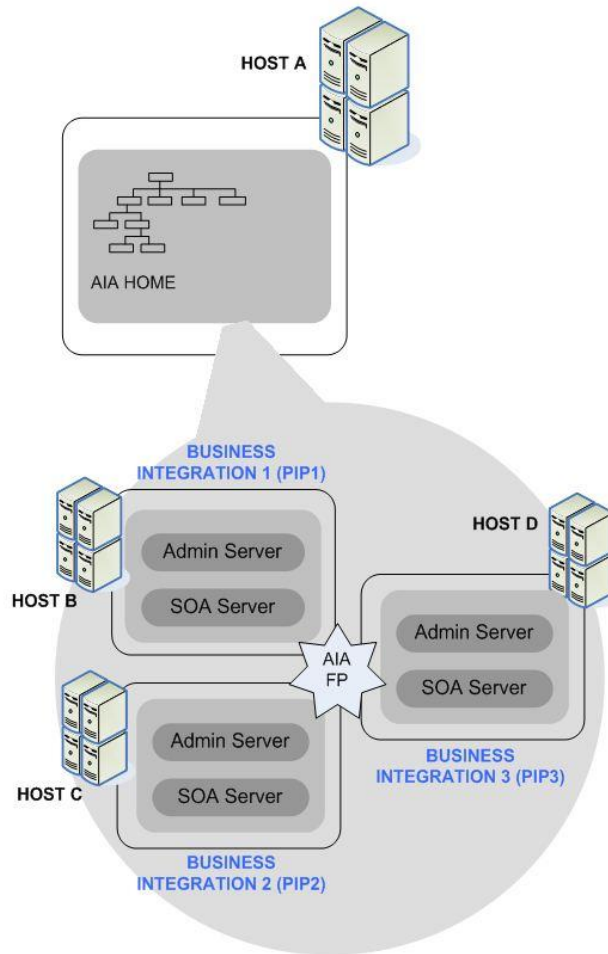


Figure 11: Distribution of the same AIA Instance to multiple SOA Servers

In the figure above instead of PIP1 and PIP2, it can also be the same PIP being distributed. For example in an Order2Bill business process integration, all 'Order' related business flows can be on host B and all 'customer' related flows can be on host C and all 'billing' related can be on host D etc. This allows provisioning each of host B and host C in a different manner to suit the performance and throughput requirement of each business activity that it hosts.

Note: Although PIP terminology is used here, it can instead be a mere grouping of custom created processes.

It is not possible to achieve this topology directly using the current version of the Installer. This topology has to be achieved using some manual steps, which will be detailed in the manual deployment section of this guide.

Note that this whitepaper gives many possibilities not all of them are tested options

AIA Deployment Architecture

Until now we have discussed deployment topologies, strategies and flexibilities while installing and deploying Oracle-delivered AIA products using the AIA Installer.

More often than not, it is also required to customize Oracle delivered content (like customizing a PIP) and/or completely build new AIA content using Foundation Pack, like building out your specific Integration for your assets.

Now the biggest challenge is how to go about deploying content in an automated and repeatable fashion.

The good news is AIA deployment Architecture bears this in mind. AIA FP exposes all the tools and utilities required so that you can construct your deployment of custom content and deploy them in the same manner as how Oracle does.

AIA treats deployment as a part of a continuous SOA lifecycle and several stakeholders contribute at different stages to derive the deployment content. Yet AIA Deployment architecture provides loose coupling so that these stakeholders are not bound to work together and in a rigid fashion – a true SOA principle.

Challenges in Multiple Service deployment in a SOA landscape

Some of the fundamental challenges faced during deployment of multiple services in an Enterprise SOA landscape are as follows.

- Typically in an organization developers that implement a service belong to a different team and possess different skill sets as compared to those that deploy these services say, administrators or Architects. So it is not obvious to the person that deploys what are the artifacts that this service is dependent on. It is for these two sets of people to interact and communicate through some channel and the end result of this is invariably a piece of deployment code that needs to be ‘implemented’ and ‘maintained’ in a continuous manner.
- In addition to being a cumbersome and manual process, such implementations become a bottleneck. Deployments need to be easy and rapid. Development is a continuous process and should be a natural extension of the development lifecycle.
- Not all services developed (and the associated configurations) will be deployed together. Different processes implementing a different business process may be deployed to a different physical server. For example, some processes may have to process high – volume and critical data and may be long running therefore requiring higher memory and performance when compared to others.

- Different organizations have different deployment topology requirements. Some of them might require their deployments to be on a cluster, some of them want to maintain multiple physical database servers etc. The key is to be able to easily achieve the desired topology

How AIA Deployment Architecture addresses these challenges

The fundamental principle is to make the deployment modular.

This is addressed by separating the deployment into three different aspects.

- What to deploy
- Where to deploy
- How to deploy

What to deploy:

This is defined using a “Deployment Plan”. A deployment plan is an auto generated xml file that will be composed of services selected for deployment (by the individual performing the deployment), along with the necessary configurations and dependant artifacts.

For AIA prepackaged products this file is pre-generated and bundled along with the product. When AIA products are deployed using the AIA Installer (to be detailed later), there is a deployment plan for each product.

Where to deploy:

This is defined in an xml file called AIAInstallProperties.xml. This file can contain multiple name value pairs for targets. These names are referenced in the generated deployment plan.

For AIA prepackaged products, the AIA Installer provides the UI to collect values for the necessary fields and the values are stored in the same xml file. Otherwise a template file is made available that can be modified manually to suit requirements.

How to deploy:

How to deploy is taken care of by the AIA Installation Driver. This is an AIA Foundation Pack provided tool and takes in as input the deployment plan and the AIAInstallProperties.xml and deploys each artifact in the deployment plan to the servers specified in the AIAInstallProperties.xml

The tool itself is a library of deployment enablers and is written using ant and python code. Using the AIA Installation Driver, the administrator (or any person that does deployment) can be oblivious to the actual deployment intricacies. This tool handles all complexities.

For AIA prepackaged products, on clicking the “Install” button on the last screen of the Installer this is the tool that is launched. For custom deployment, this tool will be invoked using command line.

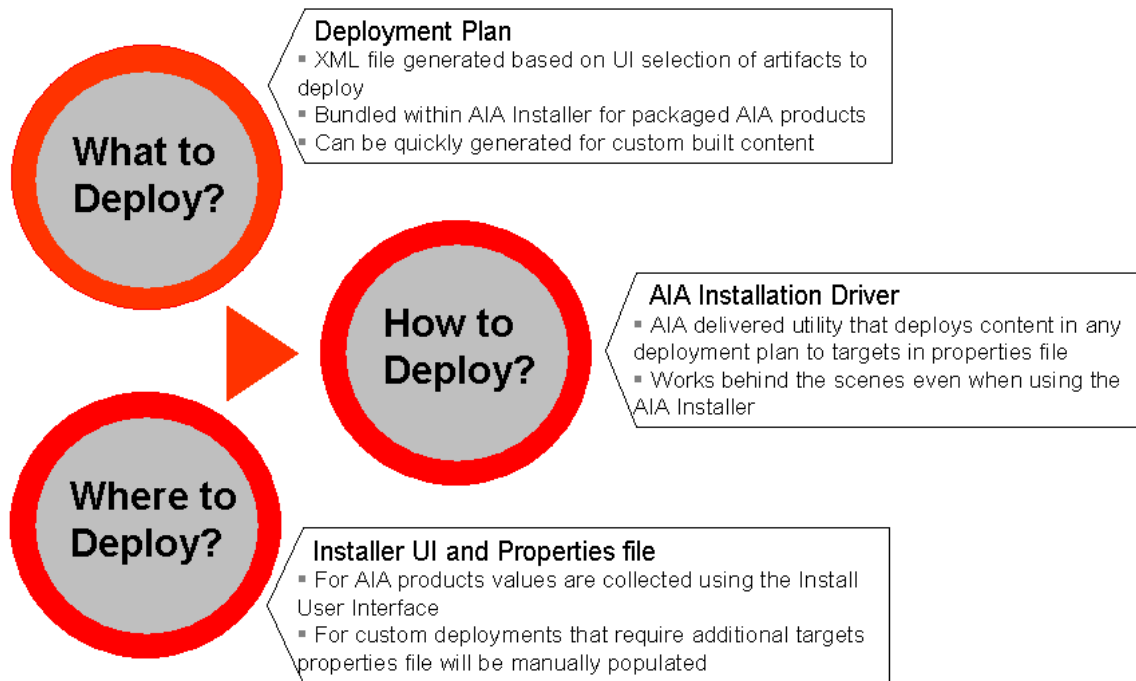


Figure 13: Modular AIA Deployment Architecture

Deploying custom AIA content - Methodology and tooling

This section will provide more details on each part of the deployment architecture defined above along with the associated tooling.

What to Deploy (BOM and Deployment Plan)

Like mentioned above, the 'what to deploy' is defined as a Deployment Plan. We will focus on how this Deployment plan is derived using AIA tools and subsequently the structure of the Deployment plan. The steps at high level to arrive at a deployment plan are

1. Annotating the services with deployment Information
2. Harvesting developed services
3. Creating Bill of Materials (BOM)
4. Generating Deployment Plan using the Bill of Materials (BOM)

AIA Project Lifecycle Workbench application is delivered with Foundation Pack (Refer to AIA documentation for more information on the same). This application is used to define and sustain an AIA project. It allows business users to easily define the artifacts that make up a given business process integration from the functional perspective down to providing reference to the individual services that implement it. This information is automatically fed as input to developers that implement the service.

Each developer when implementing a service also annotates the service with deployment specific information onto pre-generated annotation tags. At the end of the service construction, the developers harvest information back to the lifecycle workbench and in parallel (and optionally) to OER (Oracle Enterprise Repository)

Once the elements of this project are thus defined, implemented and harvested, the application provides an ability to produce a Bill of materials for this project (as an xml file). This activity is done visually and deployment users can pick and choose the services they want to deploy at that point. The following diagram gives a representation of the same.

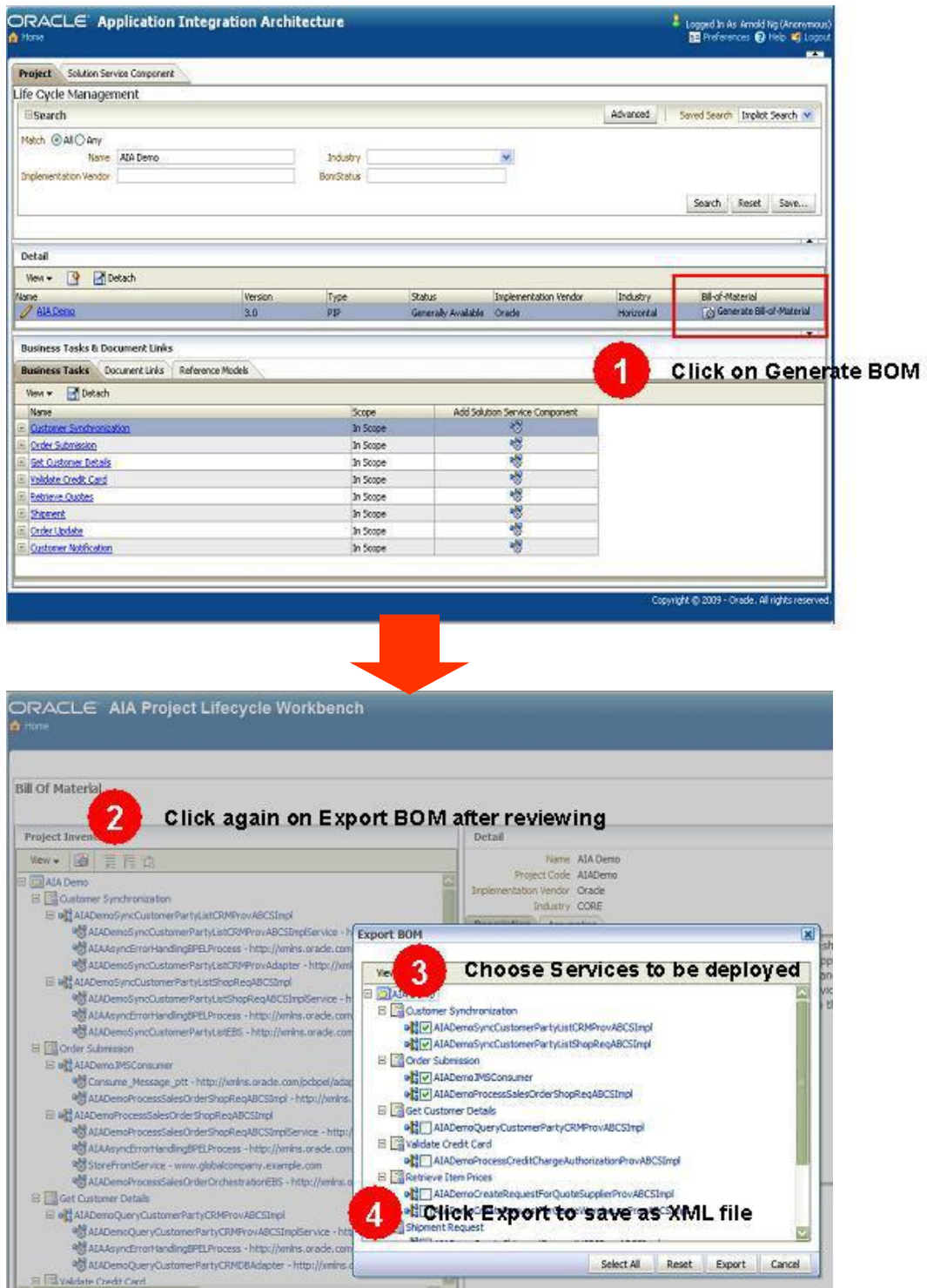


Figure 14: Bill of Material generation

This bill of material (BOM) generated, not only has the information seen on the lifecycle application but is also enriched with content from harvested annotations (for those artifacts selected). So if a composite is chosen in the BOM for deployment, then all the schemas, queues etc used by this composite are automatically available in the BOM (although the user just chose the composite) by virtue of the fact that the developer had added AIA prescribed annotations to the composite regarding the same.

This way, the individual that actually prepares the ‘what to deploy’ need not know about the internals of the content itself, which is useful because the content itself may be coming from several individuals from several development organizations.

Once the bill of materials is generated, it is required to deploy the content in the same. Bill of materials is a plain xml dump of the details of the selected artifacts and their details. So from this BOM, a **deployment plan** is generated that re-organizes all the content into a bunch of configurations and deployment activities.

The deployment plan can be generated using a command line utility. This deployment plan is now the source of truth for deployment for deploying those content that were selected in the AIA Lifecycle Application.

```

- <DeploymentPlan component="FP" version="3.0">
- <PreConfiguration>
  <!-- Add custom content -->
</PreConfiguration>
- <Configurations>
  <DataSource name="fodDS" jndiLocation="jdbc/fodDS" action="create" resourceTargetIdentifier="fod" wiserver="fp" />
  <DataSource name="SOADemo" jndiLocation="jdbc/SOADemo" action="create" resourceTargetIdentifier="fod" wiserver="fp" />
  <UpdateMetadata wiserver="fp">
    <fileset dir="{AIA_HOME}/AIAMetaData">
      <include name="AIAComponents/ApplicationObjectLibrary/AIADemo/**" />
    </fileset>
  </UpdateMetadata>
  <DbAdapter connection-instance-jndi="eis/DB/fodDS" datasource-jndi="jdbc/fodDS" action="create" wiserver="fp" />
  <DbAdapter connection-instance-jndi="eis/DB/SOADemo" datasource-jndi="jdbc/SOADemo" action="create" wiserver="fp" />
  <JMSResource jmsResourceType="ConnectionFactory" jmsResourceName="jms/aia/AIADemoBatchQueueCF" jmsresourceModule="AIAJMSModule" jmsSubDeploymentName="AIASubDeployment" action="create" wiserver="fp" />
  <JMSAdapter connection-instance-jndi="eis/aia/AIADemoBatch" connection-fac-Location="jms/aia/jms/aia/AIADemoBatch" wiserver="fp" />
</Configurations>
- <PostConfiguration>
  <!-- Add custom content -->
</PostConfiguration>
- <PreDeployment>
  <!-- Add custom content -->
</PreDeployment>
- <Deployments>
  <Composite compositeName="AIADemoProcessSalesOrderCBP" compositedir="/slot/ems3470/oracle/AIAHOME_102" revision="1.0" wiserver="fp" action="deploy" />
  <Composite compositeName="AIADemoSyncCustomerPartyReqDBAdapter" compositedir="/slot/ems3470/oracle/AIAHOME_1023/samples/AIADemo/Adapters/AIADemoSyncCustomerPartyReqDBAdapter" action="deploy" />
  <Composite compositeName="AIADemoSyncCustomerPartyListCRMProvDBAdapter" compositedir="/slot/ems3470/oracle/AIAHOME_1023/samples/AIADemo/Adapters/AIADemoSyncCustomerPartyListCRMProvDBAdapter" action="deploy" />
  <Composite compositeName="AIADemoCreateShipmentRequestBatchProvABCSImpl" compositedir="/slot/ems3470/oracle/AIAHOME_1023/samples/AIADemo/ProviderABCS/AIADemoCreateShipmentRequestBatchProvABCSImpl" action="deploy" />
</Deployments>
- <PostDeployment>
  <!-- Add custom content -->
</PostDeployment>
</DeploymentPlan>

```

Figure 15: Sample Deployment Plan

Note the tags configurations and Deployments. The content under the deployments tag define those services that were selected. The content under configurations tag defines those configurations that are necessary for the services to operate successfully. It is important to see that the person that is performing the deployment is not aware of these configurations. They need not be. The information is derived automatically.

Deployment plan is usually generated but is also extensible to allow users add more content that wasn't captured using the AIA lifecycle workbench application. Users can add more tags to the deployments or configurations or even post or pre configuration and post or pre deployment actions. For example a custom J2EE application may be required or some kind of server setup may be required prior to deployment. As long as the deployment plan adheres to the schema it will be ready for deployment

Where to deploy (AIAInstallProperties.xml)

Where to deploy is defined by the xml file called AIAInstallProperties.xml This file contains key/value pairs storing information of all target servers (like Weblogic servers, database servers etc.) where the services have to be deployed.

The values here can be entered manually and incase of installing pre-packaged AIA products this information is filled in through screens of the AIA Installer. The following is a sample.

```

<?xml version="1.0" encoding="UTF-8" ?>
- <properties>
+ <aiainstance>
+ <oeer>
- <fp>
  <version>3.0.0.0.0</version>
- <server>
  <adminhostname>sd68021crm.us.oracle.com</adminhostname>
  <adminport>7071</adminport>
  <domainname>soa_domain</domainname>
  <username>weblogic</username>
  <password>[C@568c24</password>
  <servertype>Server</servertype>
  <soaserver>soa_server1</soaserver>
  <!-- In case of cluster the below would be the Proxy URL and proxy Port -->
  <soaserverhostname>sd68021crm.us.oracle.com</soaserverhostname>
  <soaserverport>8071</soaserverport>
</server>
- <db>
- <aia>
  <jdbc-url>jdbc:oracle:thin:@sd68021crm.us.oracle.com:1591:ovm21crm</jdbc-url>
  <username>AIA1023_AIA</username>
  <password>[C@957096</password>
  <createschema>true</createschema>
  <sysusername>sys</sysusername>
  <syspassword>[C@112905b</syspassword>
  <role>SYSDBA</role>
  <defaulttablespace>SYSTEM</defaulttablespace>
  <temptablespace>TEMP</temptablespace>
</aia>
+ <mds>
+ <sr>
+ <jms>
+ <xref>
+ <aialifecycle>
</db>
</fp>

```

Figure 16: Sample AIAInstallProperties.xml

This is a hierarchical structure and allows users to add more tags as long as it adheres to the specific schema.

Notice the highlighted tags. These are references used in the Deployment plan.

For example notice in the Deployment plan above that the 'wlsserver' is assigned to the value 'fp'. This 'fp' corresponds to the details of the Weblogic server defined in the AIAInstallProperties above.

Similarly there are multiple database schemas used by AIA. If there is a JMS to be configured the deployment plan will contain some information about the JMS and have the 'target' as 'jms'

The value for jms is defined under 'db' (database) tag and under the 'jms' tag of the AIAInstallProperties.xml file as shown above. This means all JMS will be configured on this database schema. Now if it is desired that a separate schema be created for a specific JMS for load or other reasons, then the deployment plan can be modified by changing the resourcetargetidentifier to say 'jms_new' and adding an additional xml snippet under db tag where the details of 'jms_new' can be entered manually

How to deploy (AIA Installation Driver)

AIA Installation Driver (AID) is responsible for deploying the content defined in the deployment plan to the target servers defined in AIAInstallProperties.xml. The installation driver is a command line tool that can interpret line by line of the deployment plan and deploys it to the specified target server. For example, in the above deployment plan the AIA Installation Driver encounters a data source and some information about it. The AID knows how to deploy a data source. It also knows what to do in case an error occurs during deployment and has additional intelligence to deploy to remote servers, clusters etc.

Just like how the content to be deployed was completely generated just based on UI selection of artifacts making the details of the deployment plan agnostic to an end user, the actual deployment activity is also performed using the tool AID that lets the end user be not bothered about how to deploy a specific content whether it is a queue, composite etc.

AID is essentially a library of ant tasks and python scripts along with logical orchestration, exception handling etc., which can interpret the xml tags in the Deployment Plan. Most of the common deployment tasks are covered by AID. In case there are additional deployment types that are not covered by the AID, say for example JCA adapter, then AID can be extended to define a central target that can configure a JCA adapter and it can act on all Deployment plans that have tags for JCA Adapters.

Summary of overall AIA lifecycle

The diagram below tries to depict how the deployment content is derived and how it is tied to the rest of the service lifecycle. It also depicts actors at each stage.

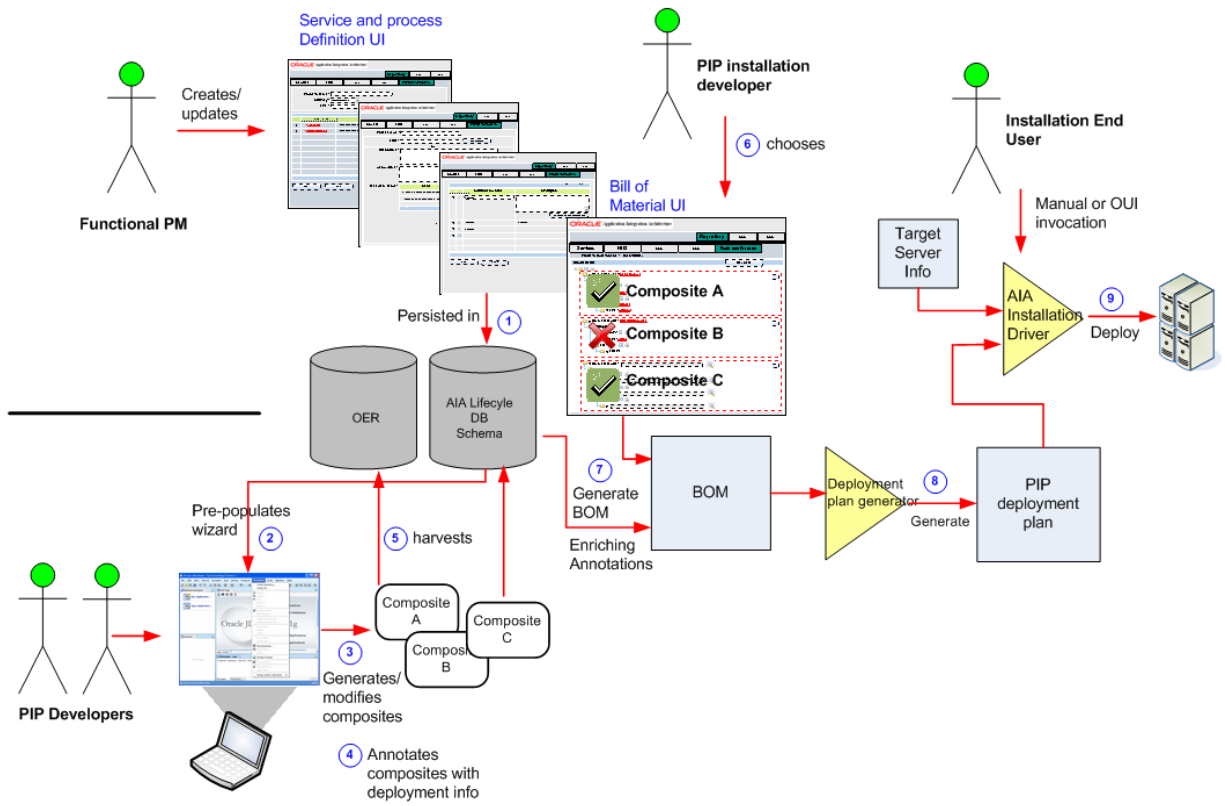


Figure 17: Diagram that shows the overall AIA lifecycle with different actors and how deployment plan is constructed and deployed to target servers

Conclusion

AIA provides you the necessary tools, content and architecture to jump-start and sustain your SOA initiative at all stages of the lifecycle.

As much as well-developed services matter, the deployment, operation and maintenance of these services and other artifacts is crucial to the success of the SOA implementation.

This paper provided you an insight into how AIA does it in an easy, automated and controlled manner. While several topologies, flexibilities and strategies were discussed, you should choose the one or extrapolate what was presented so that you attain what best suits your Organization.



AIA 11gR1 Installation and Deployment –
Flexibilities, Topologies, Architecture and Tools
February 2011

Author: Arvind Srinivasamoorthy

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2009, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.