

Oracle® Fusion Middleware

Quick Start Guide for Oracle Enterprise Repository

11g Release 1 (11.1.1.5.0)

E16581-04

April 2011

Oracle Fusion Middleware Quick Start Guide for Oracle Enterprise Repository, 11g Release 1 (11.1.1.5.0)

E16581-04

Copyright © 2008, 2011, Oracle and/or its affiliates. All rights reserved.

Primary Author: Vimmika Dinesh

Contributing Author: Dave Berry, Deb Ayers, Jason Xie, Jayesh Shah, John Yoder, Sharon Fay, Manoramya Chandramohan

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

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

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

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

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Related Documents	vi
Conventions	vi
1 Install the Governance Infrastructure	
1.1 Overview of Governance Infrastructure	1-1
1.2 Oracle's Governance Suite	1-3
1.2.1 Enterprise Repository	1-3
1.2.2 Service Registries	1-5
1.2.3 Runtime Policy Validation Tooling	1-6
1.2.4 Service Management and Monitoring	1-7
1.3 Configure the Governance Infrastructure to Interoperate with Design-time Tooling	1-8
1.3.1 Version Control Systems	1-8
1.3.2 Integrated Development Environments (IDE)	1-9
1.3.3 Document Repositories and File Stores	1-9
1.3.4 Design-time Policy Validation Tooling	1-10
1.4 Configure the Governance Infrastructure to Interoperate with Runtime Tooling	1-11
1.4.1 SOA Suite Runtime MetaData Store (MDS)	1-11
1.4.2 Testing Tools	1-11
1.4.3 Build Tools/Scripts	1-12
1.4.4 Defect Tracking Systems	1-12
1.4.5 Federation	1-13
2 Identify Assets to be Governed	
2.1 Overview	2-1
2.2 Best Practices	2-2
3 Visibility	
3.1 Overview	3-1
3.2 Oracle Enterprise Repository Solution Packs	3-1
3.3 Bootstrapping the Enterprise Repository	3-2
3.3.1 Bootstrapping the Enterprise Repository from a Service Registry	3-3
3.3.2 Bootstrapping the Enterprise Repository from the Runtime Environment	3-3

3.3.3	Bootstrapping the Enterprise Repository from Artifacts and Artifact Stores.....	3-3
3.3.4	Bootstrapping the Enterprise Repository from Excel spreadsheets and Other Metadata Stores	3-3

4 Identify Stakeholders

4.1	Overview	4-1
4.2	Stakeholders.....	4-2
4.2.1	Service Competency Center/ Software Factory/ Portfolio Management Team.....	4-3
4.2.2	Business Analysts	4-3
4.2.3	Project Architects	4-4
4.2.4	Developers (as Consumers).....	4-4
4.2.5	Developers (as Producers/Harvesters)	4-4
4.2.6	Registrars	4-5
4.2.7	Enterprise Architects.....	4-5
4.2.8	Program Manager	4-5
4.2.9	CIO.....	4-6
4.2.10	Executive Sponsor.....	4-6
4.2.11	Governance Champion	4-6

5 Modify Asset Types and Taxonomies

6 Govern the Asset Lifecycle

6.1	Overview	6-1
6.2	Asset Lifecycle	6-2
6.2.1	Definition	6-2
6.2.2	Implementation and Configuration.....	6-3
6.2.3	Testing, Staging, and Production	6-3
6.2.4	Consumer Provisioning and Contract Management.....	6-4
6.2.5	Monitoring and Management	6-4

7 Report and Promote

Index

Preface

Oracle Fusion Middleware Quick Start Guide for Oracle Enterprise Repository describes how to get your Governance infrastructure and processes, up and running very quickly, so that you can start realizing immediate value.

Audience

This document is intended for:

- Customers who are evaluating or implementing Oracle Enterprise Repository or Oracle's Governance solution
- Oracle Governance Pillar Solution Consultants assisting customers with their Proof of Concept
- Oracle Consulting as well as Oracle Consulting Partners assisting customers with their implementation

Oracle Fusion Middleware Quick Start Guide for Oracle Enterprise Repository describes the following steps and also points you to the resources that will help you get started with these:

- Install SOA Governance Infrastructure
- Identify Assets to be Governed
- Visibility
- Identify Stakeholders
- Modify Asset Types and Taxonomies
- Govern the Asset Lifecycle
- Report and Promote

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/support/contact.html> or visit <http://www.oracle.com/accessibility/support.html> if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Enterprise Repository 11g Release 1 (11.1.1.5.0) documentation set:

- *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Repository*
- *Oracle Fusion Middleware Upgrade Guide for Oracle Enterprise Repository*
- *Oracle Fusion Middleware User's Guide for Oracle Enterprise Repository*
- *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*
- *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*
- Oracle's Unified Method (OUM)

A wealth of additional Governance information can be found within Oracle's Unified Method (OUM). OUM can be used by Oracle employees, Oracle Partner Network Certified Partners or Certified Advantage Partners, and Clients who either participate in the OUM Customer Program or are engaged on projects where Oracle provides consulting services. OUM is a web-deployed toolkit for planning, executing and controlling software development and implementation projects.

For more information about OUM, see the OUM FAQ at

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

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.

Convention	Meaning
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Install the Governance Infrastructure

The first step in getting started with Governance is to install the Governance Infrastructure. This chapter describes the components of a Governance Infrastructure and also provides pointers to download and install the governance infrastructure components.

This chapter contains the following sections:

- [Section 1.1, "Overview of Governance Infrastructure"](#)
- [Section 1.2, "Oracle's Governance Suite"](#)
- [Section 1.3, "Configure the Governance Infrastructure to Interoperate with Design-time Tooling"](#)
- [Section 1.4, "Configure the Governance Infrastructure to Interoperate with Runtime Tooling"](#)

1.1 Overview of Governance Infrastructure

The Governance Infrastructure includes the following components of Oracle's SOA Governance Suite:

- Oracle Enterprise Repository (OER)
- Oracle Service Registry (OSR)
- Runtime policy validation tooling (OWSM)
- Service monitoring (Enterprise Manager Management Pack Plus for SOA)

Figure 1–1 Components of Oracle's SOA Governance Suite

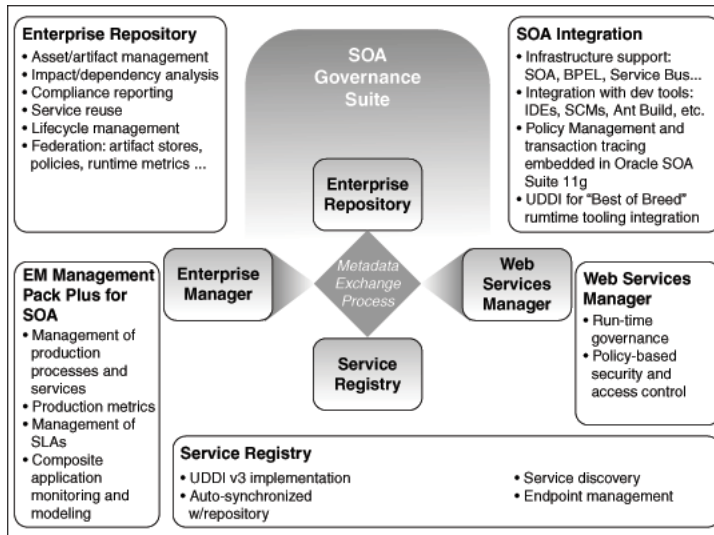


Figure 1–1 illustrates the components of Oracle’s SOA Governance Suite.

Oracle's Governance Suite is a loose bundle meaning that Oracle does not require you to purchase all of the components. As a matter of fact, some of the Governance components are also included in Oracle's SOA Suite, so the loose bundling allows you to purchase and install only the pieces that you need.

The Governance Infrastructure can interoperate with the following design-time tooling:

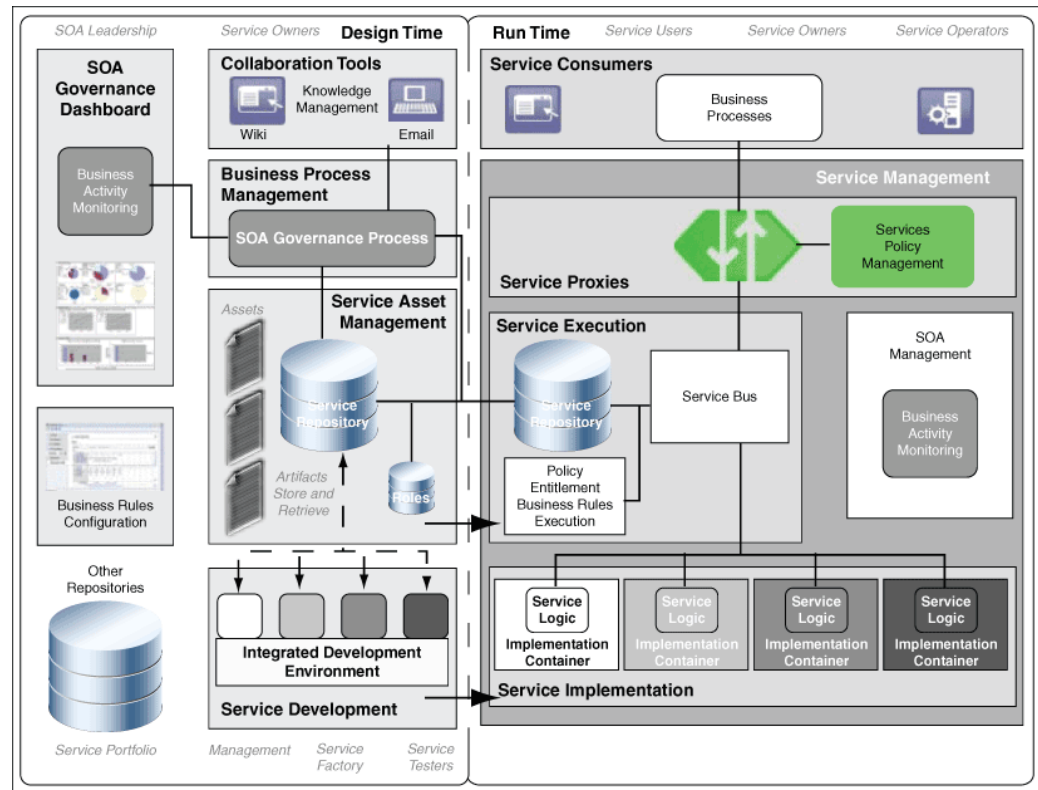
- Version Control Systems
- IDEs
- Document repositories and File Stores
- Design-time policy validation tooling

The Governance Infrastructure can also interoperate with the following runtime tooling:

- MetaData Store (MDS)
- Testing tools
- Build Tools/Scripts
- Defect tracking systems

Figure 1–2 provides an example of the use of Governance Infrastructure within a Service Oriented Architecture. This guide allows you to investigate the governance role for each of these tools and identify how you can install, connect and configure all the pieces.

Figure 1–2 The Governance Infrastructure



Note: It is not necessary to install all of the Governance Infrastructure, especially when first starting out. Organizations should select the components that are most relevant for their use cases.

For more information about the description of the Governance components, see [Section 1.2, "Oracle's Governance Suite"](#).

For additional information about the tooling versions supported by Oracle Enterprise Repository, see the Governance Ecosystem Interoperability Matrix at

<http://www.oracle.com/technology/products/soa/repository/index.html>

1.2 Oracle's Governance Suite

This section provides descriptions of the components in the Governance Suite. It contains the following topics:

- [Section 1.2.1, "Enterprise Repository"](#)
- [Section 1.2.2, "Service Registries"](#)
- [Section 1.2.3, "Runtime Policy Validation Tooling"](#)
- [Section 1.2.4, "Service Management and Monitoring"](#)

1.2.1 Enterprise Repository

Oracle Enterprise Repository provides design-time governance in support of the service lifecycle, delivering key capabilities for the storage and management of an

extensible set of metadata for any number of composites, services, business processes, and other IT-related assets. Oracle Enterprise Repository acts as the central source of SOA information, providing all actors in the service lifecycle with a human-centric discovery environment for planned, existing and retired services.

Oracle Enterprise Repository provides vital information to portfolio managers, business architects and the likes about the business capabilities and assists with strategic planning. Oracle Enterprise Repository also provides role-based links to the artifact stores of the assets that it describes as well as links to design documents, justification documents, test plans, support plans, policies, and other forms of documentation. This ensure that approvers in the lifecycle process can reference and upload documentary evidence in support of their approvals and that consumers can reference information to assist in the decision to use a particular service. Oracle Enterprise Repository is packed with features that help to automate the governance of the service lifecycle including:

- An event engine that can publish repository events to any process service endpoint.
- A built-in process engine with a set of standard processes that can be modified (using Oracle's BPM Studio product) to meet organizations' governance process workflows.
- A bi-directional registry exchange utility that keeps information synchronized between the enterprise repository and the service registry.
- A rich SOAP-based API that allows organizations to programmatically update the repository, keeping it in sync with the IT and business environment that it describes.

In addition, Oracle's Enterprise Manager Management Pack Plus for SOA provides Oracle Enterprise Repository with a summary of runtime performance metrics. These metrics help service providers view the performance of their service portfolios, and allow service consumers to judge whether a service's runtime performance is adequate for their needs.

Oracle Enterprise Repository also tracks asset usage and provides a set of robust reports that help organizations determine the ROI on asset reuse. Oracle Enterprise Repository's interface integrates with JDeveloper and other integrated development environments, and with source code management tools, to promote developer adoption.

The following resources can help you when installing the Enterprise Repository:

- Oracle Enterprise Repository Sizing Guidelines.
For more information, see <http://www.oracle.com/technetwork/middleware/repository/overview/oer11gsizingguidelines-130307.pdf>.
- Oracle Enterprise Repository Supported Platforms
For more information, see <http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html>.
- Download the Oracle Enterprise Repository from the Oracle Enterprise Repository page at <http://www.oracle.com/technology/products/soa/repository/index.html>
- Install the Oracle Enterprise Repository
For more information, see *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Repository*.

- Upgrade from previous Oracle Enterprise Repository versions
For more information, see *Oracle Fusion Middleware Upgrade Guide for Oracle Enterprise Repository*.
- Run Oracle Enterprise Repository in a clustered environment For more information, see "Install Oracle Enterprise Repository into a Clustered Environment" in *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Repository*.

Oracle Enterprise Repository can also leverage the Organization's common authentication mechanisms. Oracle Enterprise Repository supports:

- LDAP/ Active Directory
For more information, see "LDAP/ Active Directory" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.
- eTrust Siteminder
For more information, see "eTrust Single Sign-On" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.
- Container-managed authentication
For more information, see "Container Managed Setup" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

Oracle Enterprise Repository uses Oracle's Business Process Management (BPM) Tool to run the out-of-the-box automated workflows. While organizations are free to use the workflow engine of their choice, the Oracle Enterprise Repository includes a limited use license of Oracle BPM. For more information about Oracle BPM, see

- <http://www.oracle.com/us/technologies/bpm/index.html> to download Oracle BPM.
- http://download.oracle.com/docs/cd/E13154_01/bpm/docs65/installguide/index.html to install Oracle BPM.

Oracle Enterprise Repository uses Oracle's Business Intelligence (BI) Publisher as its reporting engine. The Oracle Enterprise Repository includes a limited use license of Oracle BI Publisher. For more information about Oracle BI Publisher, see

- <http://www.oracle.com/technetwork/middleware/bi-publisher/downloads/index.html> to download Oracle BI Publisher.
- http://download.oracle.com/docs/html/B25825_01/toc.htm to install Oracle BI Publisher.

1.2.2 Service Registries

Oracle Service Registry is a feature rich UDDI version 3 compliant service registry that provides standards based interoperability and forms the run time interface of the Repository. It provides service binding and run time location transparency, federation of run time metrics for closed loop governance and allows access to an appropriate service version based on the environment. Services published to the registry can be looked up at run time for dynamic service location. For more information, see <http://www.oracle.com/technology/products/soa/registry/index.html>.

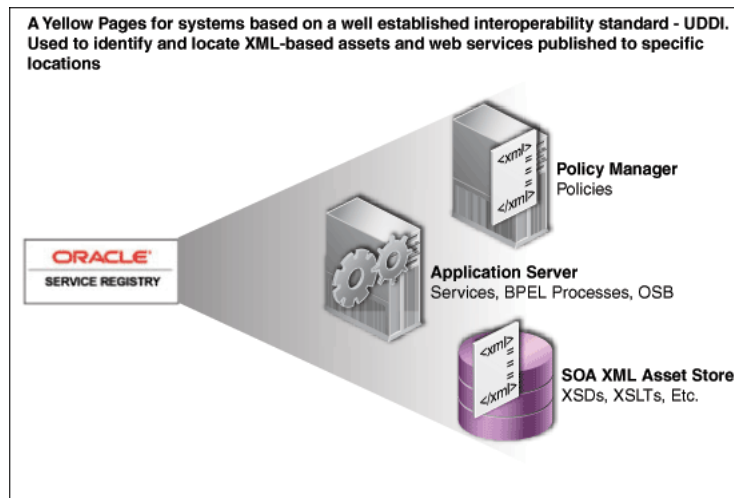
Oracle Service Registry also serves as an integration point for runtime tooling

- Oracle Service Bus subscribes to new or modified assets
- Composite applications can discover updated endpoints and WSDL locations
- Runtime monitoring tooling, such as Amberpoint, can publish metrics to Oracle Service Registry, which are then propagated back to Oracle Enterprise Repository.

Note: Oracle's Enterprise Manager Management Pack Plus for SOA provides runtime metrics directly to the Enterprise Repository.

Registries can also be federated - system configurations with a single logical repository, and multiple registries (one for each major environment or stage in the lifecycle) are common, as shown in [Figure 1-3](#).

Figure 1-3 Single Repository and Multiple Registries



The following resources can help you when installing the Service Registry:

- For information about downloading the Oracle Service Registry, see <http://www.oracle.com/technologies/soa/service-registry.html>.
- For information about installing the Oracle Service Registry into each major environment or stage in the lifecycle, see <http://download.oracle.com/otndocs/tech/soa/OSR11gR1ProductDocumentation.pdf>.
- For information about running Oracle Service Registry in a clustered environment, see http://www.oracle.com/technology/tech/soa/uddi/osr_cluster_config.pdf.

There is tight integration between the Oracle Enterprise Repository and the Oracle Service Registry. The Oracle Registry Repository Exchange Utility synchronizes Oracle Enterprise Repository and Oracle Service Registry bi-directionally so that the metadata from either of these products can flow in either direction through the utility.

1.2.3 Runtime Policy Validation Tooling

Oracle Web Services Manager (OWSM) is a security policy manager for Web services that allows IT management to centrally define security policies that govern Web services operations (such as access policy, logging policy, and load balancing). Those policies can then be applied to Web services in a heterogeneous environment without requiring modification to those services. In addition, Oracle Web Services Manager collects and monitors statistics to ensure quality of service, uptime, and security and displays them in a Web dashboard.

A key OWSM feature is the ability to provide end-to-end security between the service consumer and the service provider by deploying client and server agents within the container itself. Both the client agent and the server agent obtain central policy

information from the OWSM Policy Manager, thereby ensuring that the consumer can apply the correct policies, even when the security measures are updated on the called service.

The Oracle Web Services Manager is part of Oracle's SOA Suite. If you have installed Oracle SOA Suite, then OWSM is also installed.

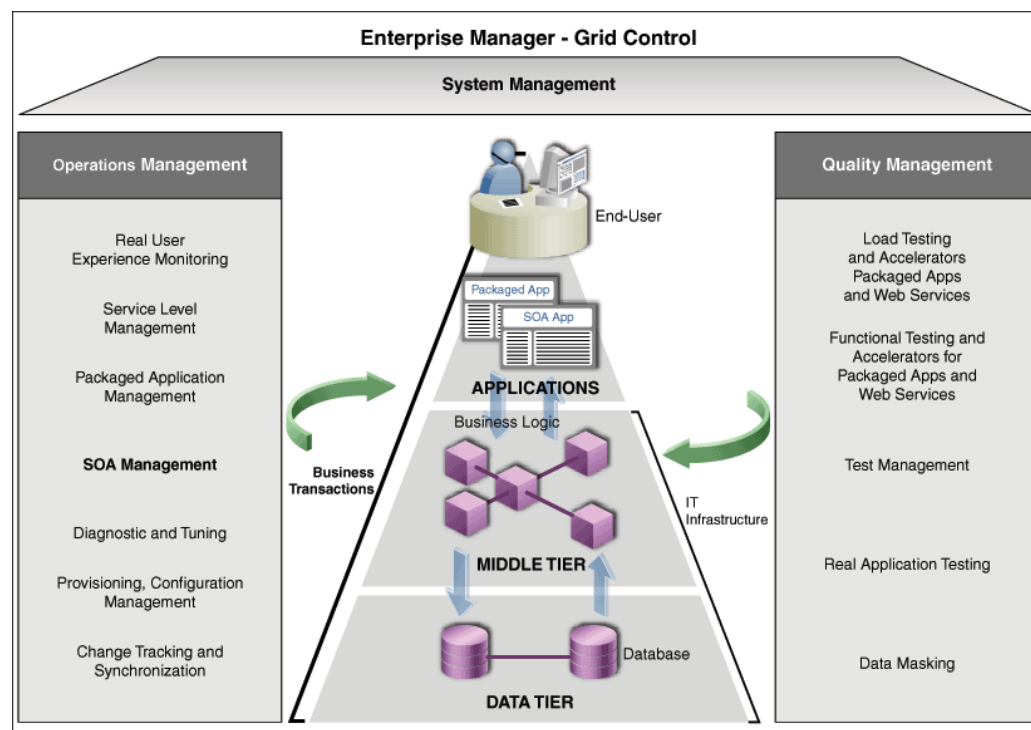
- For more information about downloading the Oracle Web Services Manager, see <http://www.oracle.com/technetwork/testcontent/index-085253.html>.
- For more information about installing the Oracle Web Services Manager, http://download.oracle.com/docs/cd/E12524_01/web.1013/e12576/toc.htm.
- For more information about configuring the Oracle Web Services Manager, http://download.oracle.com/docs/cd/E12839_01/web.1111/b32511/toc.htm.

1.2.4 Service Management and Monitoring

Oracle's Enterprise Manager Management Pack Plus for SOA builds upon the core Enterprise Manager Grid Control product and provides operational management of complex SOA environments.

Enterprise Manager Grid Control is a Web-based system for central management of Oracle products, host systems, and applications, as shown in Figure 1–4. It is architected to allow management of various Oracle products through packs and plug-ins. The Management Pack Plus for SOA provides discovery and configuration management of deployed SOA artifacts.

Figure 1–4 Enterprise Manager Grid Control



Enterprise Manager Grid Control also provides dynamic discovery and service level monitoring of all artifacts deployed within a Java Application Server. This gives administrators the ability to confirm what is running in a particular SOA environment is exactly as defined at design time.

Enterprise Manager Grid Control collects statistics that include latency, invocation counts and exceptions for each of the service components. It then aggregates and stores them over various aggregation periods to provide a rich dashboard of metrics presented across a user selected set of time-periods. Service-level objectives can be established at each layer in the SOA architecture and both cautionary and violation level thresholds can be set, which when breached over a defined period can trigger alerts, delivered via a number of standard protocols.

1.3 Configure the Governance Infrastructure to Interoperate with Design-time Tooling

You can configure the Governance infrastructure to interoperate with design-time tooling. This section contains the following topics:

- [Section 1.3.1, "Version Control Systems"](#)
- [Section 1.3.2, "Integrated Development Environments \(IDE\)"](#)
- [Section 1.3.3, "Document Repositories and File Stores"](#)
- [Section 1.3.4, "Design-time Policy Validation Tooling"](#)

1.3.1 Version Control Systems

Oracle Enterprise Repository is not a Version Control system; it is a metadata management system. However, Oracle Enterprise Repository can be tightly integrated with your existing version control systems. When developers find an asset in the Enterprise Repository, and need access to the physical artifact (such as an XSD or WSDL file), Oracle Enterprise Repository serves as a broker and deliver a copy of the code from the version control system to the developer.

This benefits your organization in several ways:

- In a scenario where no behavioral change required, the organization maintains its existing version control practices. Thereby, there are no behavioral changes required.
- In a scenario where Oracle Enterprise Repository serves as a single broker for all SCMs, the developers do not need access to multiple version control systems.

You can also connect to several Version Control Systems/Source Control Management Systems.

- Version Control Systems that provide URI access to content
 - List the SCMs included in this category
 - Download the connector
 - Access the documentation

For more information about connectors, see "Oracle Enterprise Repository Connectors" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

The supported Version Control Systems/Source Control Management Systems are as follows:

- ClearCase
 - Download the connector
 - Access the documentation

For more information about ClearCase, see "ClearCase Integration" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

- Harvest-HTTP
 - Download the connector
 - Access the documentation

For more information about Harvest-HTTP, see "Harvest-HTTP Repository Host Integration" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

- Serena ChangeMan
 - Download the connector
 - Access the documentation

For more information about ChangeMan, see "Serena ChangeMan Integration" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

1.3.2 Integrated Development Environments (IDE)

Oracle Enterprise Repository makes governance as invisible as possible to developers. One way this is accomplished is through tight integration with the development environment (IDE). Through the development environment, developers can search for relevant assets, evaluate asset metadata, and select assets that they would like to reuse. Oracle Enterprise Repository can then harvest their completed work, automatically generating assets and relationships based on the harvested artifacts.

Integrating with Supported IDEs:

- JDeveloper (new in Oracle Enterprise Repository 11g)
 - Download the plugin
 - Access the documentation

For more information about Oracle JDeveloper, see "Configuring Oracle JDeveloper" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

- Eclipse
 - Download the plugin
 - Access the documentation

For more information about Eclipse, see "Configuring Eclipse" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

- VS .NET
 - Download the plugin
 - Access the documentation

For more information about VS .NET, see "Configuring VS .NET" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

1.3.3 Document Repositories and File Stores

Sometimes developers need additional information to understand an asset's functions and behaviors. This information might be captured in use case documents,

architecture documents, test cases, test results, etc. Such documents are typically stored in version control systems, repositories, and file stores. However, you'll want to create a link from the asset in the Enterprise Repository to the associated documents, so that developers can seamlessly access this information.

This functionality is available natively within Oracle Enterprise Repository and requires no connector or plugin. For more information, see "Configure Artifact Stores" in *Oracle Fusion Middleware User's Guide for Oracle Enterprise Repository*.

1.3.4 Design-time Policy Validation Tooling

Oracle offers OWSM for runtime security policy management and enforcement. However, there are times when it is important to communicate design-time and development requirements to development teams as policies. Oracle Enterprise Repository offers native policies to serve this purpose. Policies might include:

- Corporate quality standards and guidelines
- Asset naming conventions
- Performance targets

Policies are applied to assets in order to communicate asset requirements that must be considered during design and development, and to provide administrators with the means to enforce and monitor asset compliance with governance, architecture, and other organizational standards. For example, a policy might articulate corporate quality standards, identifying the platforms that an asset should run on, and identifying acceptable defect density rates.

Oracle Enterprise Repository's native policies can be manually validated by a Subject Matter Expert, or they may be validated programmatically through third party tooling.

To use Oracle Enterprise Repository's native policies:

- Activate the Policy Management Feature
- Create a Policy Type
- Add the Policy Assertion Element to the Policy Type
- Create a Policy
- Add the Applied Asset Policies Element to a Type
- Apply a Policy to an Asset
- View Policy Status
- Validate Policy Assertions

Oracle Enterprise Repository's native policies can be validated by third party tooling, such as Parasoft and WebLayers.

For more information about configuring Oracle Enterprise Repository's native policies, see "Configuring Policy Management" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

For more information about validating Oracle Enterprise Repository native policies using third-party tooling, see <http://www.oracle.com/technetwork/middleware/repository/>.

1.4 Configure the Governance Infrastructure to Interoperate with Runtime Tooling

You can configure the Governance infrastructure to interoperate with runtime tooling. This section contains the following topics:

- [Section 1.4.1, "SOA Suite Runtime MetaData Store \(MDS\)"](#)
- [Section 1.4.2, "Testing Tools"](#)
- [Section 1.4.3, "Build Tools/Scripts"](#)
- [Section 1.4.4, "Defect Tracking Systems"](#)
- [Section 1.4.5, "Federation"](#)

1.4.1 SOA Suite Runtime MetaData Store (MDS)

MDS is a runtime metadata storage system that is used by Oracle's Fusion Middleware including Oracle SOA Suite to store runtime metadata and configuration information. Oracle Enterprise Repository can harvest metadata from MDS. The runtime and design-time metadata can be viewed in Oracle Enterprise Repository, which provides Oracle SOA Suite with visibility into the service endpoints. Oracle Enterprise Repository maintains references to externally hosted artifacts that reside in an HTTP server, Web based Distributed Authoring and Versioning (WEBDAV), or in a repository such as MDS.

The best practice is to host all shared runtime metadata on a single MDS instance so that the metadata can be accessed by all SOA infrastructures. This is also the instance from which you would harvest metadata into Oracle Enterprise Repository.

For more information about harvesting assets and metadata from MDS, see "Configuring and Using Automated Harvesting in Design-time and Runtime Environments" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

1.4.2 Testing Tools

A wide selection of Web Service testing tools is available. These tools are extremely valuable in that they allow organizations to test their Web Services early and often. Because of the number of tools on the market, there is no single out of the box testing tool connector. Instead, integration with the testing tool of your choice can be accomplished using the Repository Extensibility Framework (REX). REX is a Web Services API for programmatic integration into Oracle Enterprise Repository. It is based on accepted industry standards, and designed with a focus on interoperability and platform independence. REX uses Remote Procedure Call (RPC) Web Services described by the Web Services Description Language (WSDL v1.1). This allows clients to interact with Oracle Enterprise Repository using any platform and any implementation language that supports Web Services.

When integrating with testing tools, the following approach is recommended:

- When a Web Service asset in Oracle Enterprise Repository reaches a particular lifecycle stage, an event is triggered
- The event initiates retrieval of the Web Service code from the location indicated in Oracle Enterprise Repository.
- The testing tool runs the appropriate tests for the lifecycle stage.

- The test results are posted to Oracle Enterprise Repository as metadata or as a document link attached to the Web Service asset.

To view the REX API calls that can be used to execute the above sequence, see "Repository Extensibility Framework" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

1.4.3 Build Tools/Scripts

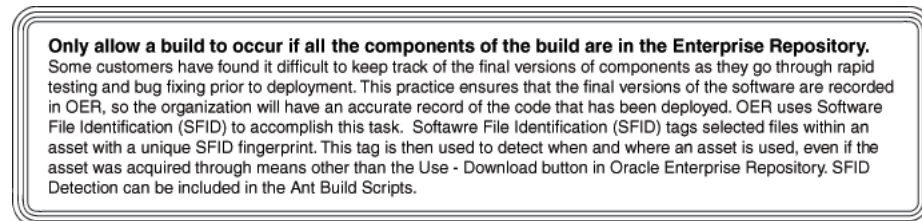
One of the goals of governance is to ensure that the organization has visibility into assets as they move throughout the lifecycle. Therefore, it is important to harvest information about the organization's assets at build time. This can be done by integrating the Oracle Enterprise Repository Harvester into build tools and scripts, including Ant and the WebLogic Scripting Tool (WLST). WLST is a command-line scripting interface that system administrators can use to manage WebLogic Server instances. WLST supports Oracle Service Bus and SOA Suite.

For more information about WLST, see *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

You can invoke harvester from WLST to ensure that all deployment information is stored in Oracle Enterprise Repository at deployment time. For more information about embedding the Harvester into Build Tools and Scripts, see "Configuring and Using Automated Harvesting in Design-time and Runtime Environments" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

Governance practices can also be enforced at build time. [Figure 1–5](#) illustrates a sample example of one of an Oracle customer's governance practices.

Figure 1–5 Example Governance Practice



For more information about SFID, see "Configuring Automated Usage Detection" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

1.4.4 Defect Tracking Systems

Defect Tracking Systems are useful in tracking problems with assets once they are deployed to the runtime environment. When a developer considers whether to reuse an asset, it is helpful to view all of the defects associated with that asset.

Oracle Enterprise Repository can integrate with defect tracking systems, and has an adapter that supports integration with ClearQuest. Integrating ClearQuest with your system enables you to use a ClearQuest store in Oracle Enterprise Repository. For more information about integration with ClearQuest, see "ClearQuest Integration" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

1.4.5 Federation

Oracle Enterprise Repository can be logically federated: the repository's Role Based Access controls can be used to ensure that different groups across the organization have access to their assets, and that everyone can see enterprise-wide assets.

Oracle Enterprise Repository does not have any out-of-the-box support for physical federation: there are individual instances of Oracle Enterprise Repository to serve the needs of individual groups. While it is possible to build a federated structure using Oracle Enterprise Repository, Oracle does not recommend it.

For more information about setting up your Oracle Enterprise Repository instance to support logical federation, see "Advanced Role-based Access Control" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

Identify Assets to be Governed

The second step in getting started with Governance is to identify the assets that need to be governed.

This chapter contains the following sections:

- [Section 2.1, "Overview"](#)
- [Section 2.2, "Best Practices"](#)

2.1 Overview

Oracle Enterprise Repository provides automated tooling to get assets into the Enterprise Repository so that they can be managed and governed. This section provides a careful consideration of which assets to be submitted to the repository.

While an iterative approach to governance is recommended, it is important to demonstrate the value of governance as quickly as possible. Oracle recommends that organizations bootstrap the enterprise repository in phases, with sharp focus on getting visibility into the assets that will bring the greatest value and benefits.

For example, if your organization's primary focus is application development, you may want to start bootstrapping the Enterprise Repository with domain-independent enterprise components and services that have high reuse potential, such as:

- Logging services
- Error handling services
- Authentication services
- Online data capture and storage routines
- Data access services (that provide customer information, account information, inventory information, etc)
- Connection pooling services
- Caching services
- Frameworks (STRUTS, security, etc)
- Validation routines

Approximately 20% of every application is composed of domain-independent functionality (Poulin, 1996) , so this type of functionality is highly reusable across the entire organization. In addition, organizations typically want to standardize this type of functionality so that there is consistency across applications. The initial productivity savings for the few key assets mentioned above can be quite impressive. Many

organizations realize several hundred thousand dollars in savings in a matter of months, in addition to the value derived through standardization.

Organizations focused on application integration may want to start bootstrapping the enterprise repository with:

- Commonly used data access objects
- XML schemas
- Messaging hubs and common message formats
- Services that access information in the legacy systems
- Existing business processes with links to the underlying systems and services that orchestrate them.

If you are not thinking Governance at this point, but just want to use the Enterprise Repository as a communication mechanism for your project, bootstrap the enterprise repository with the information that is most relevant to the project team. This might include:

- Existing functionality that fulfills the project's functional and/or non-functional requirements
- An overview of the functionality/capabilities to be delivered by the team (this allows the team to determine what pieces of functionality will be delivered by whom, and to manage dependencies throughout the project)
- Standards and best practices
- Open Source licenses and projects
- Commercial Off the Shelf (COTS) dependencies - which commercial packages/libraries are used
- Build labels - check in the repository location of the latest build, test results, etc. using the Ant task.
- SCM information (location of the source code)
- Configured platform (for example - WLS configured for use in development environments)

2.2 Best Practices

This section provides descriptions of the components in the Governance Suite. It contains the following topics:

Understand which portfolios will help the organization reach its goals

Identify the organizational goals for governance, and align these goals with the supporting asset portfolios. For example, if the organization is focused on bringing new products to market more quickly, product line asset portfolios should be among the first included in the governance program. If the goal is to establish a standard Software Development Lifecycle Process in order to bring new products to market more quickly, process templates and process standards should be among the first asset portfolios.

Identify requirements

After identifying the asset portfolios, it's time to identify Asset Portfolio Management Team requirements for managing their respective portfolios. For example, Architects may need to package and track the use of standard solution sets, and monitor

compliance to standards. Service Competency Centers may need to communicate the services that orchestrate a business process. The Governance program serves as a channel of distribution for asset portfolios and help Portfolio Management Teams better manage their portfolios.

Structure the program to deliver value

We know that people don't do anything without motivation. Motivation can take the form of a stick (penalties for not completing the task) or a carrot (benefits for completing the task). There are advantages to leveraging both. A clear understanding of the program vision from executives puts you in a better position to gain their sponsorship and to structure a program that will have value for the Asset Portfolio Management Teams and their consumers. But it's not enough to communicate the benefits - benefits must be demonstrated. Teams will not be truly motivated to participate unless they can see tangible benefits.

The Bottom Line

Governance will not be successful unless there is a supply of assets of sufficient value to help the organization reach its goals. Asset Portfolio Management Teams are the primary sources of asset supply. These teams must be engaged and their needs must be factored into the structure of the governance program.

For more information about the role of the Asset Portfolio Management team, see [Chapter 4, "Identify Stakeholders"](#).

Resources

Oracle offers a whitepaper and a workbook that will help you evaluate the expected ROI from your asset portfolios. You can use this tool as one mechanism for identifying assets of the greatest value to the organization.

- Whitepaper: Determining the ROI of SOA through Reuse
- Workbook: Determining the ROI of SOA through Reuse

For more information about determining the ROI of SOA through reuse, see <http://www.oracle.com/technetwork/middleware/repository/overview>.

- Additional resources are available that can help you determine the total cost of ownership for your application integrations:
 - Whitepaper: Building the Business Case for Application Integration Architecture (AIA)
 - Workbook: Application Integration Architecture Total Cost of Ownership

For more information about how to build the business case for AIA, see <http://www.oracle.com/technetwork/middleware/repository/overview>.

A wealth of additional Governance information can be found within Oracle's Unified Method (OUM). OUM can be used by Oracle employees, Oracle Partner Network Certified Partners or Certified Advantage Partners, and Clients who either participate in the OUM Customer Program or are engaged on projects where Oracle provides consulting services. OUM is a web-deployed toolkit for planning, executing and controlling software development and implementation projects.

For more information about OUM, see the OUM FAQ at

<http://www.oracle.com/us/products/consulting/resource-library/oracle-unified-method1-069271.pdf>

The third step in getting started with Governance is to install Solution Packs and Harvest Targeted Assets into Oracle Enterprise Repository.

This chapter contains the following sections:

- [Section 3.1, "Overview"](#)
- [Section 3.2, "Oracle Enterprise Repository Solution Packs"](#)
- [Section 3.3, "Bootstrapping the Enterprise Repository"](#)

3.1 Overview

An organization cannot govern what it cannot see. Therefore, the first step of any Governance initiative is to get visibility into the assets that you want to manage.

To that end, a number of Oracle Enterprise Repository Solution Packs are available:

- Base Data Pack
- Harvester Solution Pack (includes Adapter Solution Pack)
- AIA Solution Pack (available with the purchase of Oracle's Application Integration Architecture Foundation Pack and PIPs)

Existing assets can also be harvested into the enterprise repository. The process through which organizations initially gather assets into the Enterprise Repository is called "bootstrapping". Bootstrapping uses a programmatic approach to loading assets and asset metadata into the Enterprise repository. The Enterprise Repository can be bootstrapped from a number of sources:

- A service registry
- The runtime environment
- Artifacts and artifact stores
- Excel spreadsheets and other metadata stores

Organizations might also choose to use a manual approach for entering the assets in the portfolio. In addition, manual validation after harvesting ensures that the imported assets were harvested properly, and additional, business-focused metadata can be added.

3.2 Oracle Enterprise Repository Solution Packs

This section describes the various Oracle Enterprise Repository solution packs available:

- ["Base Data Pack"](#)
- ["Harvester Solution Pack \(Includes Adapter Solution Pack\)"](#)
- ["AIA Solution Pack"](#)

Base Data Pack

The asset types delivered as part of Base Data Pack are templates. They include the business-focused asset metadata that organizations track. The metadata fields in these asset types have been collected from the best practices of organizations worldwide. The Base Data Pack is automatically installed with a new installation of Oracle Enterprise Repository. Base Data types can be copied and customized to suit specific organizational assets.

Harvester Solution Pack (Includes Adapter Solution Pack)

The asset types delivered as part of the Harvester Solution Pack are required by the Harvester tool. The Harvester and Harvester Solution Pack provide metadata about the implementation of an asset and the assets's relationships. Oracle Enterprise Repository harvests artifacts into a pre-specified format defined by the Harvester Solution Pack.

The Oracle Enterprise Repository Harvester Solution Pack includes asset types for Oracle's Application Adapters and Applications. These asset types allow an organization to harvest application integrations into Oracle Enterprise Repository and have end-to-end visibility into the integrations between two or more applications.

The Oracle Enterprise Repository Harvester tool provides visibility into the assets generated by many of Oracle's products, including Oracle SOA Suite, Oracle Service Bus, Oracle Enterprise Manager, Oracle BPEL PM, UDDI Registries, and others. The Oracle Enterprise Repository Harvester tool also allows organizations to harvest any standards-based WSDL, BPEL, XSD, and XSLT artifacts generated by third-party tooling. The harvesting process requires the Harvester Solution Pack.

The Harvester is available in 11.1.1.x.x-OER-Harvester-Solution-Pack.zip, which is bundled with the Oracle Enterprise Repository installation in the following .zip file:

```
<ORACLE_
HOME>/repository111/core/tools/solutions/11.1.x.x.0-OER-Harvester-Solution-Pack.zip
```

For more information, see *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

AIA Solution Pack

For more information about the AIA Solution Pack, see the "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository" section in *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*.

3.3 Bootstrapping the Enterprise Repository

This section describes the bootstrapping of Oracle Enterprise Repository. It contains the following topics:

- [Section 3.3.1, "Bootstrapping the Enterprise Repository from a Service Registry"](#)
- [Section 3.3.2, "Bootstrapping the Enterprise Repository from the Runtime Environment"](#)

- [Section 3.3.3, "Bootstrapping the Enterprise Repository from Artifacts and Artifact Stores"](#)
- [Section 3.3.4, "Bootstrapping the Enterprise Repository from Excel spreadsheets and Other Metadata Stores"](#)

3.3.1 Bootstrapping the Enterprise Repository from a Service Registry

Since effective SOA Governance requires an Enterprise Repository working in combination with a Service Registry, it is necessary to synchronize the two tools.

The Oracle Registry Repository Exchange Utility synchronizes Oracle Enterprise Repository and Oracle Service Registry bi-directionally so that the metadata from each of these products can flow in either direction through the utility.

To bootstrap the Oracle Enterprise Repository from a Service Registry, follow these steps:

- Configure the Oracle Registry Repository Exchange Utility
- Understand how Metadata is Synchronized between Oracle Service Registry and Oracle Enterprise Repository
- Review Known Issues

For more information about Oracle Registry Repository Exchange Utility, see http://download.oracle.com/docs/cd/E14571_01/doc.1111/e16580/oereu.htm#CACEFFIH.

3.3.2 Bootstrapping the Enterprise Repository from the Runtime Environment

The Enterprise Repository can be bootstrapped from assets that reside in the organization's runtime containers. Oracle Enterprise Repository uses the Harvester tool to access deployed artifacts. The Harvester reads standards-based files including BPEL, WSDL, XSD and XSLT files, as assets generated by Oracle SOA Suite and Oracle Service Bus. The Harvester automatically creates assets, populates asset metadata, and generates relationship links based on the information in the artifact files. Note that the Harvester is not restricted to Oracle products - it can be used to harvest standards-based artifacts generated from any tooling. In the runtime environment, the Harvester can be used from the command line or it can be invoked using an Ant task.

For more information about harvesting from runtime containers, see "Runtime Harvesting Details" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

3.3.3 Bootstrapping the Enterprise Repository from Artifacts and Artifact Stores

The Enterprise Repository can be bootstrapped from production-grade assets that reside in the organization's version control systems or file systems. Oracle Enterprise Repository uses the Harvester to access the artifacts in various artifact stores. The Harvester can be invoked from the command line to harvest from artifact stores.

For more information about harvesting production-grade artifacts from artifact stores, see "Configuring the Harvester for the Command Line" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

3.3.4 Bootstrapping the Enterprise Repository from Excel spreadsheets and Other Metadata Stores

Organizations just beginning to explore Governance practices might track asset metadata and dependencies in a spreadsheet, on a wiki, or through some other type of

tool. At a certain point, however, this becomes unmanageable and it then makes sense to move the information to an Enterprise Repository.

The Oracle Enterprise Repository includes a Web Services based API called the Repository Extensibility (REX) Framework that can be used to bootstrap Oracle Enterprise Repository from spreadsheets and other metadata stores. This requires customization and coding, and is covered as an advanced topic.

If you are interested in finding out more about the Enterprise Repository's API, see "Developing Custom Integrations" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

Identify Stakeholders

The fourth step in getting started with Governance is to identify initial Governance stakeholders.

This chapter contains the following sections:

- [Section 4.1, "Overview"](#)
- [Section 4.2, "Stakeholders"](#)

4.1 Overview

Now that you have identified the assets that should be governed, it is time to work with the stakeholders to:

- Determine how they can best leverage the Enterprise Repository
- Identify additional metadata required by stakeholders
- Modify taxonomies/categorizations
- Customize/tailor each stakeholder's access to and views of assets

Figure 4–1 Closed Loop Governance Process

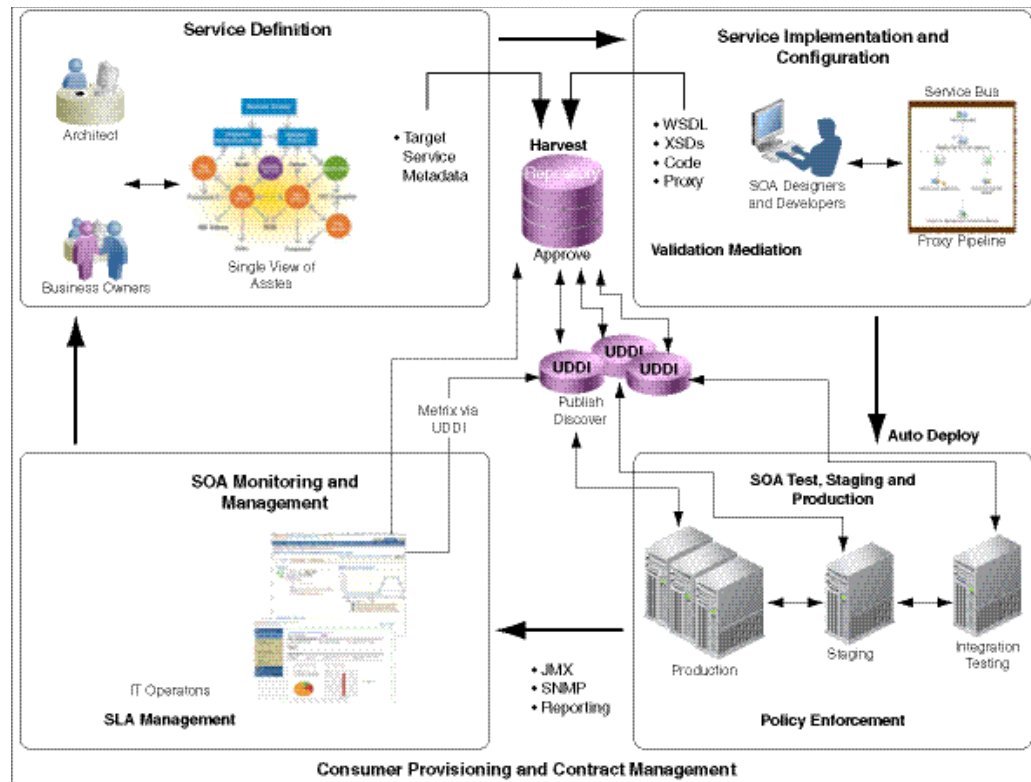


Figure 4–1 provides an overview of some of the stakeholders that participate in the closed loop governance process. This process is supported by the governance infrastructure.

The following description of common stakeholders in the closed-loop governance process, and the typical Roles assigned to them in the Oracle Enterprise Repository.

- For information about establishing Users in Oracle Enterprise Repository, see "Users" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.
- For information about assigning Roles to Users, see "Roles" and "Access Settings" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.
- Oracle has a perspective on some of the characteristics of best-of-breed Governance solutions. For more information about Governance structures, see the OFRA Governance material at

<http://www.oracle.com/technology/products/soa/repository/index.html>

4.2 Stakeholders

This section describes the available Oracle Enterprise Repository stakeholders:

- Section 4.2.1, "Service Competency Center/ Software Factory/ Portfolio Management Team"
- Section 4.2.2, "Business Analysts"
- Section 4.2.3, "Project Architects"
- Section 4.2.4, "Developers (as Consumers)"

- Section 4.2.5, "Developers (as Producers/Harvesters)"
- Section 4.2.6, "Registrars"
- Section 4.2.7, "Enterprise Architects"
- Section 4.2.8, "Program Manager"
- Section 4.2.9, "CIO"
- Section 4.2.10, "Executive Sponsor"
- Section 4.2.11, "Governance Champion"

4.2.1 Service Competency Center/ Software Factory/ Portfolio Management Team

Role in Governance

- Identify common needs across the community
- Identify assets and levels of investment needed to fulfill those needs
- Determine the expected return on asset investments
- Manage the asset lifecycle, from initial inception, creation, distribution, usage all the way through to retirement
- Define common functionality in the Enterprise Repository
 - Create an asset in Oracle Enterprise Repository
 - Populate definitional metadata
 - Attach relevant documentation
 - Identify a target availability date
- Establish Governance workflows for their portfolio of assets

For more information about workflows, see "Configuring Oracle Enterprise Repository Workflow" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.
- Participate in the Governance approval process for assets in their portfolio.
- Evaluate the performance of their assets by reviewing usage history, developer feedback, runtime performance metrics, policy status, and compliance reports
- Use the Oracle Enterprise Repository Navigator to evaluate the impact should they change or modify their assets
- Create new versions of existing assets, and retire old versions

Typical Roles Assigned in Oracle Enterprise Repository

Registrar

4.2.2 Business Analysts

Role in Governance

- Specify global business requirements
- Reuse existing business processes
- Structure new business processes using existing capabilities and services

- Drive revenue by extending business processes to trading partners

Typical Roles Assigned in Oracle Enterprise Repository

Advanced Submitter

4.2.3 Project Architects

Role in Governance

- Analyze and search software assets to design a project architecture that maximizes appropriate reuse (SLA for services, maturity of components, design document)
- Create and use applications, services and component versions
- Notify stakeholders of changes to applications, services and components

Typical Roles Assigned in Oracle Enterprise Repository

Advanced Submitter

4.2.4 Developers (as Consumers)

Role in Governance

- Analyze and search software assets for specific reuse (features, dependencies, how to guides, test harness)
- Use applications, services and component versions
- Notify stakeholders of changes to applications, services and components
- Use reusable assets in the creation of other assets
- Search repository for suitable reusable assets for projects
- Integrate reusable assets, with additional responsibility for engineering or modifications made necessary by the integration

Typical Roles Assigned in Oracle Enterprise Repository

User

4.2.5 Developers (as Producers/Harvesters)

Role in Governance

- Creates reusable assets from scratch and/or reengineers existing assets to be reusable
- May be responsible for testing assets and ensuring quality standards are met
- May be responsible for maintaining assets and helping users understand asset functionality, features, limitations, and applicability

Typical Roles Assigned in Oracle Enterprise Repository

Advanced Submitter

4.2.6 Registrars

Role in Governance

- Assess, certify, classify, add, and delete assets from the repository
- May be responsible for maintaining the operational aspects of the repository
- Ensures that work products meet quality, documentation, and other established standards
- May communicate new work product additions or deletions to all constituents
- Match consumer engineer needs with producer engineer capabilities and availability

Typical Roles Assigned in Oracle Enterprise Repository

Registrar

4.2.7 Enterprise Architects

Role in Governance

- Track progress toward the target architecture
- Communicate progress to the executives
- Communicate architectural standards to delivery teams
- Monitor compliance

Typical Roles Assigned in Oracle Enterprise Repository

Registrar

4.2.8 Program Manager

Role in Governance

- Coordinates planning, prioritization, monitoring, and support of the governance program, while aligning it with changing business needs
- Facilitates the definition of the program objectives
- Oversee all aspects of the enterprise initiative
- Establish consensus and promote the program goals and objectives, including schedules, budget, key milestones, and deliverables
- Co-develop, review, and integrate the supporting project plans
- Development of communication plans, risk management plans, and skill gap analysis
- Provide regular updates to senior management, customer representatives and program participants
- Establish metrics and reporting requirements

Typical Roles Assigned in Oracle Enterprise Repository

Registrar Administrator

4.2.9 CIO

Role in Governance

Analyze and search reuse and the contributions of applications, components, services by LOB, product, value chain

Typical Roles Assigned in Oracle Enterprise Repository

User

4.2.10 Executive Sponsor

Role in Governance

An individual or group representing management and its interests

- Authorizes, encourages and supports the use of resources
- Provides visibility that management supports the program

Typical Roles Assigned in Oracle Enterprise Repository

User

4.2.11 Governance Champion

Role in Governance

- Program support and evangelism throughout every level of the organization
- Educates others on the program concepts, its long-term benefits, its inhibitors, and the motivation behind its implementation
- Sets direction for the effort and implementation
- Communicates successful efforts throughout the organization

Typical Roles Assigned in Oracle Enterprise Repository

User

Modify Asset Types and Taxonomies

The fifth step in getting started with Governance is to review and modify asset types & taxonomies to meet stakeholder needs.

The term "asset" refers to anything that is stored in Oracle Enterprise Repository. This can include - but is not limited to- business processes, services, patterns, frameworks, applications, components and data services. Different asset types include:

- Service-related assets
- Process assets
- Knowledge assets
- Governance assets

An asset is more than just code; in fact it may not be code at all. An asset could be a process, the IT roadmap, or any number of documents. Flexible asset types support the following:

- Component-based development
- Enterprise integration
- Service-oriented architecture
- Architecture, standards and best practices deployment

One of Oracle Enterprise Repository's unique features is the ability to model any asset type. The Type Manager provides a user-friendly tool for modifying the asset types supplied with the solution packs, changing their look and feel in order to define new asset types. The Type Manager provides a wide range of element/display types and the ability to arrange these types in a way that makes sense for the organization, from an Asset Consumer and Asset Creator/Editor perspective.

Users assigned to the "Registrar Administrator" role in Oracle Enterprise Repository can create new asset types, modify asset type metadata, create new categories, and new tabs and add any number of new elements to each tab.

The following best practices are helpful when modifying asset types to meet stakeholder needs.

1. Retain a copy of the original metadata
 - Make copy of the original asset type.
 - Rename the copy by adding a prefix of "z" and make the copy inactive.
 - Make desired changes to the original.

Note: It is important to follow this best practice, particularly if you are automatically harvesting assets. Oracle Enterprise Repository harvests the assets into a pre-defined model. The model leverages the Unique Identifiers associated with the out-of-the-box asset types. The harvester does not populate copies of the original assets, as those copies have a different Unique Identifier.

2. Use templates when working with stakeholders to identify the metadata fields that they are needed for each asset. We have found that it is easier for stakeholders to look at a list of metadata to initially identify the metadata fields. After stakeholders have identified the metadata that they would like to see on each asset, a prototype can be developed in Oracle Enterprise Repository, and the prototype can be validated and revised by stakeholders.
 - The following link provides a list of out-of-the-box metadata for each Oracle Enterprise Repository asset type:
<http://www.oracle.com/technetwork/middleware/repository/overview/asset.xls>
Use this as a template for identifying the metadata that is needed by the stakeholders identified in [Chapter 4, "Identify Stakeholders"](#).
3. Oracle Enterprise Repository offers two views of asset metadata. The Viewer identifies the metadata that is visible to all users of the system. The Editor is a template-based view available to those who are entering or editing asset information. Organize metadata in the Editor in a way that makes it easy for those who are entering or editing asset information to know which metadata fields they need to populate.
 - The Editor can have multiple, customizable tabs. We recommend organizing these tabs by user role, such that users can quickly identify the asset metadata they need to provide.
 - As an alternative, you can organize tabs according to SDLC state. This clarifies the metadata that is needed at each stage of the asset's lifecycle.
 - Note that Oracle Enterprise Repository's automated workflows use the Tabs to process assets. This means that you should maintain consistency across your asset types so that there is consistency in the way that they are processed by the workflows.

Oracle Enterprise Repository harvests assets according to an underlying model. For example, when harvesting from JDeveloper 11g, Oracle Enterprise Repository harvests the project into an underlying composite asset model. The model determines which assets appear in Oracle Enterprise Repository, as well as the relationships that appear between the assets. As discussed in [Chapter 3, "Visibility"](#), the harvested assets display the implementation details.

[Figure 5-1](#) shows one of the composite assets that ships with SOA Suite. The diagram on the right is a representation of the underlying model. When the composite is harvested, it is interpreted according to the underlying model. The results are displayed in [Figure 5-2](#).

Figure 5-1 JDeveloper Composite and Resulting Oracle Enterprise Repository Composite Model

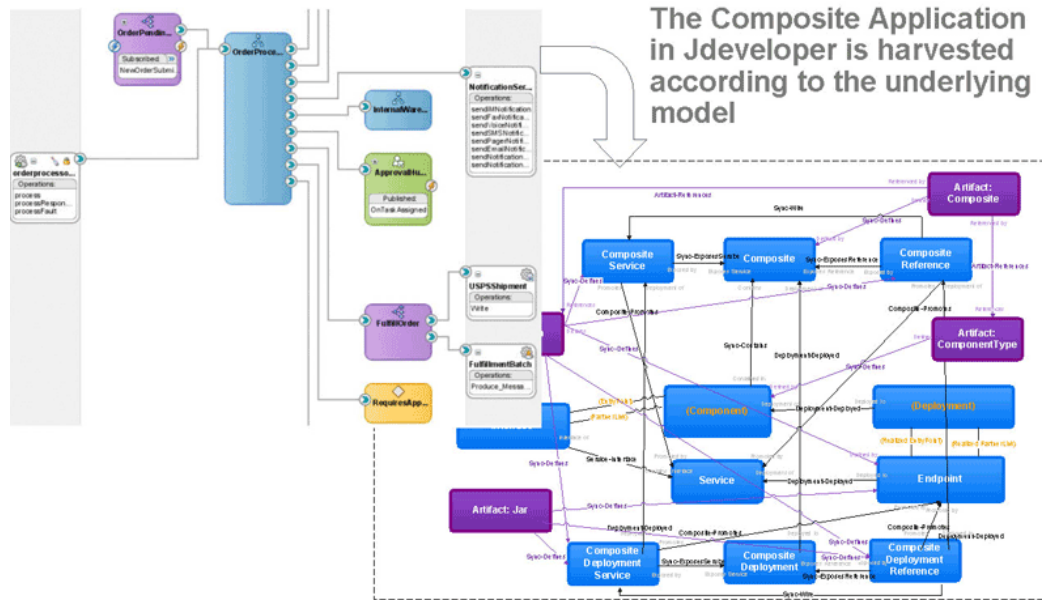


Figure 5-2 Composite Harvesting Results

OER displays a summary of the composite in the Asset Detail page as well as in the navigator. Both summaries can be navigated.

The Solution Packs provide the underlying models that define how harvested assets are displayed in Oracle Enterprise Repository. The Oracle Enterprise Repository Base asset types and the asset types in each of the solution packs can be extended to include additional metadata fields needed to support the stakeholder information requirements.

Categorizations are a unique way of classifying or categorizing assets and projects within Oracle Enterprise Repository. They provide the ability to not only appear as metadata within an asset but also provide very robust searching capabilities. These are demonstrated both in the Browse Tree searches and Advanced Search capabilities in the Oracle Enterprise Repository Home page. Categorizations can be represented as anything that is hierarchical or taxonomical; examples include Architecture Reference Models, Asset Lifecycle Stages, Lines of Business, Domains, Business Capabilities, etc.

Oracle Enterprise Repository ships with a number of categorizations. These can be customized to reflect the concepts that are relevant to your organization.

- Think about how users would search for assets, for example, line of business, technology
- Think how teams organize their assets, for example, lifecycle stage, business domain

For more information about customizing taxonomies, see sections "Configuring Categorizations" and "Tab Elements" in *Oracle Fusion Middleware User's Guide for Oracle Enterprise Repository* and "Configuring Oracle Enterprise Repository Categorizations in the UDDI Mappings File" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

Govern the Asset Lifecycle

The sixth step in getting started with Governance is to govern the lifecycle of your assets.

This chapter contains the following sections:

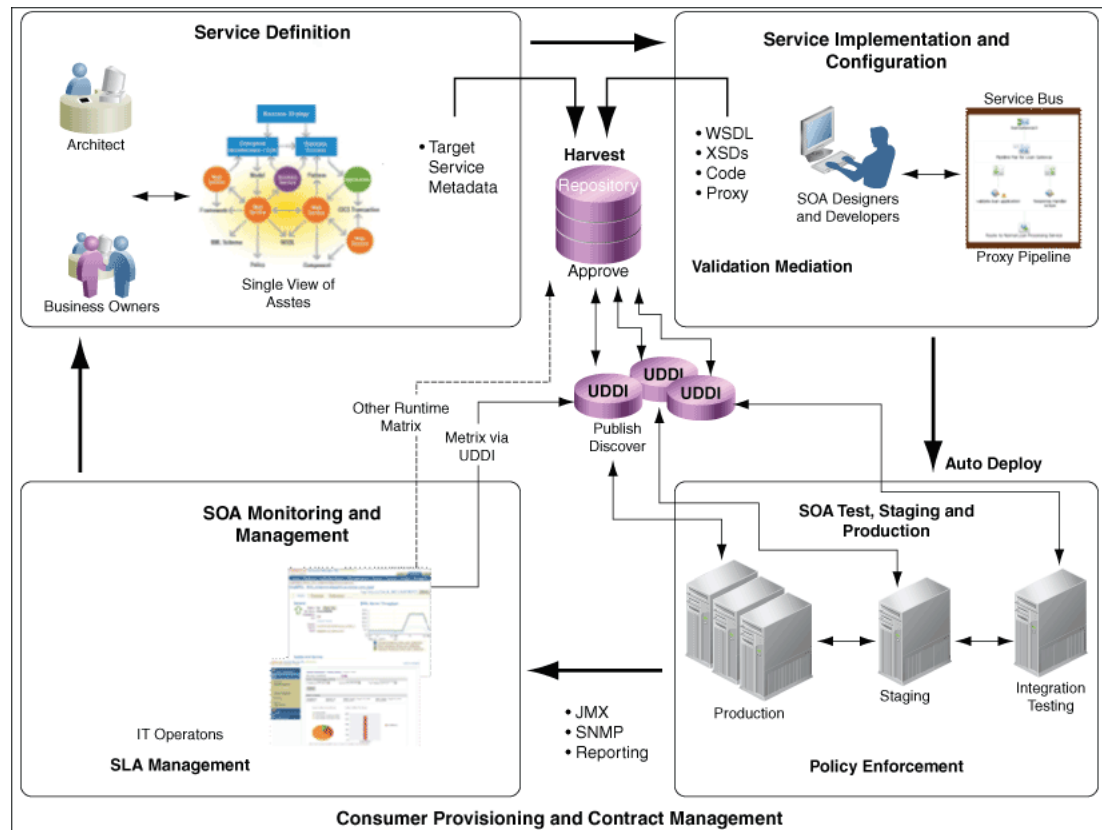
- [Section 6.1, "Overview"](#)
- [Section 6.2, "Asset Lifecycle"](#)

6.1 Overview

Once an organization has visibility into existing assets that they want to govern, the organization then should focus on inserting governance into the asset lifecycle process. [Figure 6-1](#) illustrates the asset lifecycle process.

The process starts with the definition of a common capability that is needed by multiple groups across the enterprise. Once the capability is funded, a project is initiated to implement and/or configure the capability. After a capability is developed, it gets promoted from development to pre-production environments, which might include staging and testing. Finally, the capability is promoted to the production environment, where its performance can be monitored and measured, and its behavior can be enforced. This process is facilitated by Oracle Enterprise Repository's automated workflows.

Figure 6–1 Asset Lifecycle Process



6.2 Asset Lifecycle

This section describes the role of Governance in each of the asset lifecycle phases. This section contains the following topics:

- [Section 6.2.1, "Definition"](#)
- [Section 6.2.2, "Implementation and Configuration"](#)
- [Section 6.2.3, "Testing, Staging, and Production"](#)
- [Section 6.2.4, "Consumer Provisioning and Contract Management"](#)
- [Section 6.2.5, "Monitoring and Management"](#)

6.2.1 Definition

When the organization recognizes that a capability is required, it should be made visible through the Enterprise Repository. These capabilities may come from architects who are working on the target architecture, or from business analysts who identify common requirements for their lines of business. Once in the enterprise repository, they can be evaluated and prioritized according to a common business or ROI model. Once evaluated and prioritized, these capabilities can be categorized into stages, such as proposed (meaning the capability has been identified and vetted), funded (there is an active project in the process of delivering the capability), and complete (the capability has been fully implemented).

There are several ways to make a capability visible through the Enterprise Repository:

- Quick Submission

- Advanced Submission

There is also information and a workbook available that will help you place a value on your capabilities. This can be used to help you justify and prioritize your asset investments:

- Whitepaper: Determining the ROI of SOA through Reuse
- Workbook: Determining the ROI of SOA through Reuse

For more information about how to determine the ROI of SOA through Reuse, see <http://www.oracle.com/technetwork/middleware/repository/overview/index.html>.

6.2.2 Implementation and Configuration

As capabilities are funded, projects are established to implement the capability. SOA Suite developers working in JDeveloper can see and reuse services from the enterprise repository to complete their projects. Service Bus developers working in Eclipse can see and reuse services from the enterprise repository to complete their projects. Developers can also submit their completed implementations directly to the Enterprise Repository. The Enterprise repository also supports VS .Net development.

- For information about reusing assets through JDeveloper, Eclipse, and VS .Net, see "Configuring Your IDE to Support Integration with Oracle Enterprise Repository" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.
- For information about harvesting assets from JDeveloper, Eclipse, and VS .Net, see *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

6.2.3 Testing, Staging, and Production

Once an implementation has been harvested to Oracle Enterprise Repository, the repository's workflows process the implementation according to the governance rules and practices established by the organization. Oracle Enterprise Repository comes with several pre-configured workflows that can be customized to support the Governance practices of your organization.

For more information about Oracle Enterprise Repository's workflows, as well as the procedures for customizing the workflows, see "Configuring Oracle Enterprise Repository Workflow" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

As the implementation moves throughout the lifecycle, from testing, through staging, and into production, the endpoints for each environment are harvested into Oracle Enterprise Repository. This is accomplished by incorporating the Harvester into the Ant or WLST deployment script. After harvesting, Oracle Enterprise Repository provides visibility into a service with multiple endpoints, one for each environment.

For information about harvesting endpoints from the runtime environment using deployment scripts, see "Configuring and Using Automated Harvesting in Design-time and Runtime Environments" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

Oracle Enterprise Repository promotes the services and harvested endpoints to a Service Registry in each lifecycle environment. Oracle Enterprise Repository uses a tool called the Exchange Utility to perform the promotion. Once services are in the service registry, runtime tooling, such as OSB and SOA Suite, can dynamically discover changes in the runtime environment. OSB subscribes to the Service Registry's publish/subscribe API, such that OSB is automatically notified when a change occurs. Alternately, SOA Suite caches the WSDL and endpoint locations, as well as the UDDI

service key, at design-time. If the WSDL or endpoints become invalid at any point in the lifecycle, SOA Suite will use the service key to dynamically discover and update the WSDL location and endpoint information.

- For more information about configuring the Exchange Utility to promote assets from Oracle Enterprise Repository to Oracle Service Registry, see "Configuring Oracle Registry Repository Exchange Utility" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.
- For more information about Oracle Service Bus's dynamic Discovery capabilities, see the "UDDI" section in *Oracle Fusion Middleware Administrator's Guide for Oracle Service Bus*.
- For more information about the dynamic discovery capabilities of Oracle SOA Suite, see *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite* at http://download.oracle.com/docs/cd/E15523_01/integration.1111/e10224.pdf.

6.2.4 Consumer Provisioning and Contract Management

Some organizations may not want to automatically expose their production service endpoints to all consumers. Consumers may first be required to negotiate a service level agreement with the service provider. Oracle Enterprise Repository can facilitate this negotiation process.

For more information about the contract negotiation process, see <http://download.oracle.com/otndocs/tech/soa/uddi/OracleSOAGovBootcampConsumerMgtLab.zip>.

6.2.5 Monitoring and Management

It is important to both service providers and service consumers to understand how a service is performing in the runtime environment. In Oracle Enterprise Repository 11g, a summary of runtime service performance metrics can be brought into Oracle Enterprise Repository from Enterprise Manager SOA Management Pack Plus.

For information on configuring interoperability between Oracle Enterprise Repository and Enterprise Manager Management Pack Plus for SOA, see *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

Runtime metrics can also be brought into Oracle Enterprise Repository from Amberpoint, and similar tools, which publish service metrics to Oracle Service Registry. Bidirectional synchronization between Oracle Enterprise Repository and Oracle Service Registry is used to promote the runtime metrics from the registry to the repository.

Note that Oracle's Enterprise Manager Management Pack Plus for SOA provides runtime metrics directly to the Enterprise Repository.

For additional information about promoting metrics from Oracle Service Registry to Oracle Enterprise Repository, see "Metric Publishing" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

Report and Promote

The seventh step in getting started with Governance is to report on and promote your savings.

Oracle Enterprise Repository tracks and reports on the design-time use of assets. Usage is tracked through the manual asset Use - Download process within Oracle Enterprise Repository, and when usage occurs through the IDE.

Oracle Enterprise Repository leverages Oracle BI Publisher as its reporting engine. Oracle Enterprise Repository offers more than twenty pre-configured reports. Customers can also generate their own reports using the reporting tool of choice.

For more information about Oracle Enterprise Repository's reporting capabilities and metrics, see *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

Index

A

AIA Solution Pack, 3-2
Asset Lifecycle, 6-2
 see Definition
 see Implementation and Configuration
 see Overview
 see Testing, Staging, and Production
Asset Lifecycle Process, 6-2

B

Base Data Pack, 3-2
Best Practices, 2-2
Bootstrapping the Enterprise Repository, 3-2
Bootstrapping the Enterprise Repository from a Service Registry, 3-3
Bootstrapping the Enterprise Repository from Artifacts and Artifact Stores, 3-3
Bootstrapping the Enterprise Repository from Excel spreadsheets and Other Metadata Stores, 3-3
Bootstrapping the Enterprise Repository from the Runtime Environment, 3-3
Build Tools/Scripts, 1-12
Business Analysts, 4-3

C

Categorizations, 5-3
CIO, 4-6
 see Role in Governance
 see Typical Roles Assigned in Oracle Enterprise Repository
Closed Loop Governance Process, 4-2
Components of Oracle's SOA Governance Suite, 1-2
Composite Harvesting Results, 5-3
Configure the Governance Infrastructure to Interoperate with Design-time Tooling, 1-8
Configure the Governance Infrastructure to Interoperate with Runtime Tooling, 1-11
Consumer Provisioning and Contract Management, 6-4

D

Defect Tracking Systems, 1-12
Definition, 6-2

Design-time Policy Validation Tooling, 1-10
Developers (as Consumers), 4-4
 see Role in Governance
 see Typical Roles Assigned in Oracle Enterprise Repository
Developers (as Producers/Harvesters), 4-4
 see Role in Governance
 see Typical Roles Assigned in Oracle Enterprise Repository
Document Repositories and File Stores, 1-9

E

Enterprise Architects, 4-5
 see Role in Governance
 see Typical Roles Assigned in Oracle Enterprise Repository
Enterprise Manager Grid Control, 1-7
Enterprise Repository, 1-3
Example Governance Practice, 1-12
Executive Sponsor, 4-6
 see Typical Roles Assigned in Oracle Enterprise Repository
Executive Support
 see Role in Governance

F

Federation, 1-13

G

Govern the Asset Lifecycle, 6-1
Governance Champion, 4-6
 see Role in Governance
 see Typical Roles Assigned in Oracle Enterprise Repository
Governance Infrastructure, 1-1

H

Harvester Solution Pack (Includes Adapter Solution Pack), 3-2

I

Identify Assets to be Governed, 2-1
*see*Overview
Identify Business Analysts
*see*Role in Governance
*see*Typical Roles Assigned in Oracle Enterprise
Repository
Identify Stakeholders, 4-1
*see*Overview
Implementation and Configuration, 6-3
Install the Governance Infrastructure, 1-1
Integrated Development Environments (IDE), 1-9

J

JDeveloper Composite and Resulting Oracle
Enterprise Repository Composite Model, 5-3

M

Modify Asset Types and Taxonomies, 5-1
Monitoring and Management, 6-4

O

Oracle Enterprise Repository Solution Packs, 3-1
Oracle's Governance Suite, 1-2, 1-3
Oracle's Unified Method (OUM), vi
Oracle's Governance Suite
*see*Enterprise Repository
*see*Service Registries
Overview, 2-1, 3-1, 4-1, 6-1

P

Program Manager, 4-5
*see*Role in Governance
*see*Typical Roles Assigned in Oracle Enterprise
Repository
Project Architects, 4-4
*see*Role in Governance
*see*Typical Roles Assigned in Oracle Enterprise
Repository

R

Registrars, 4-5
*see*Role in Governance
*see*Typical Roles Assigned in Oracle Enterprise
Repository
Report and Promote, 7-1
Role in Governance, 4-3, 4-4, 4-5, 4-6
Runtime Policy Validation Tooling, 1-6

S

Service Competency Center/ Software Factory/
Portfolio Management Team, 4-3
Service Management and Monitoring, 1-7
Service Registries, 1-5

Single Repository and Multiple Registries, 1-6
SOA Suite Runtime MetaData Store (MDS), 1-11
Stakeholders, 4-2

T

Testing Tools, 1-11
Testing, Staging, and Production, 6-3
The Governance Infrastructure, 1-3
Typical Roles Assigned in Oracle Enterprise
Repository, 4-4, 4-5, 4-6

V

Version Control Systems, 1-8
Visibility, 3-1
*see*Overview