Oracle® Fusion Middleware

Concepts Guide for Oracle Enterprise Repository

11*g* Release 1 (11.1.1.7)

E16581-07

February 2013

Documentation for anyone evaluating or implementing Oracle Enterprise Repository. This guide describes concepts related to governance infrastructure, governed assets, visibility, stakeholder identity, asset types and taxonomies, asset lifecycle, and reporting.

ORACLE®

Oracle Fusion Middleware Concepts Guide for Oracle Enterprise Repository, 11*g* Release 1 (11.1.1.7)

E16581-07

Contents

4 Stakeholder Identity

5 Asset Types and Taxonomies

6 The Asset Lifecycle

7 Reporting

Index

Preface

Oracle Fusion Middleware Concepts Guide for Oracle Enterprise Repository describes the concepts behind the Governance infrastructure and processes.

Audience

This document is intended for:

- Customers who are evaluating or implementing Oracle Enterprise Repository or Oracle's Governance solution
- Oracle Consulting and Oracle Consulting Partners assisting customers with their implementation

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

- Governance Infrastructure
- Governed Assets
- Visibility
- Stakeholder Identity
- Asset Types and Taxonomies
- The Asset Lifecycle
- Reporting

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

For more information, see the following documents in the Oracle Enterprise Repository 11g Release 1 (11.1.1.6.0) documentation set:

- Oracle Enterprise Repository on OTN - The home page for Oracle Enterprise Repository on Oracle Technology Network (OTN) is:

  http://www.oracle.com/technetwork/middleware/repository/overv
  iew/index-100687.html

- SOA Blog - Keep on top of the latest SOA blogs at:

  http://blogs.oracle.com/governance

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.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in This Guide for Release 11.1.1.7

This guide was updated for Release 11.1.1.7. The following table lists sections that are new or changed.

> **Note:** For a list of known issues (release notes), see the "Known Issues for for Oracle SOA Products and Oracle AIA Foundation Pack" at
> http://www.oracle.com/technetwork/middleware/docs/soa-aiafp-knownissuesindex-364630.html.

Sections	Changes Made	February 2013
All Chapters		-
Chapter summaries	Enhanced document and chapter summaries for better web search results.	-
Document structure	Document restructured as a concepts guide.	x

1

Governance Infrastructure

This chapter describes Oracle's service oriented architecture (SOA) governance infrastructure and Oracle's SOA Governance Suite components, such as the Oracle Service Registry (OSR) which is the runtime interface for the repository, Oracle Web Services Manager (OWSM) for security policy management, SOA Suite Runtime MetaData Store (MDS) for design-time tooling, and Oracle's Business Intelligence (BI) Publisher for reporting.

This chapter contains the following sections:

- Section 1.1, "Overview of Governance Infrastructure"
- Section 1.2, "Oracle's Governance Suite"
- Section 1.3, "Design-time Tooling"
- Section 1.4, "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 security policy enforcement (OWSM)
- Service monitoring (Oracle Enterprise Manager SOA Management Pack Enterprise Edition)

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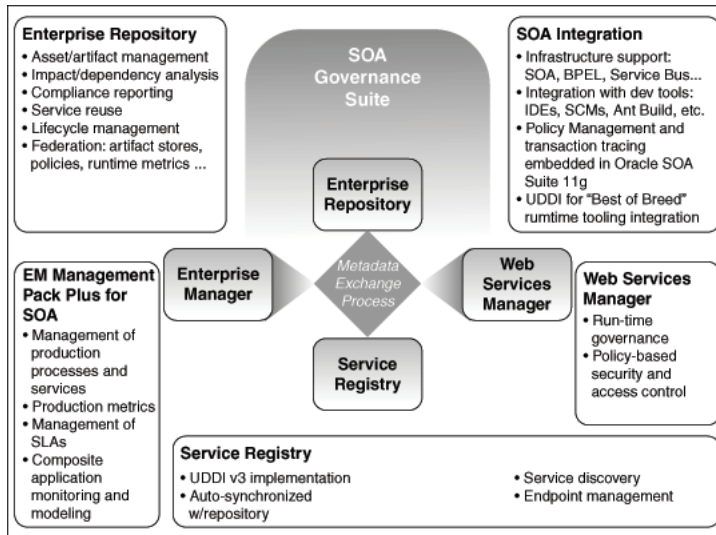
Figure 1–1 illustrates the components of Oracle's SOA Governance Suite.

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.

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 shows the governance infrastructure within Service Oriented Architecture. This guide describes the governance role for each of these tools and how you can install, connect, and configure all the pieces.

Figure 1–2   The Governance Infrastructure
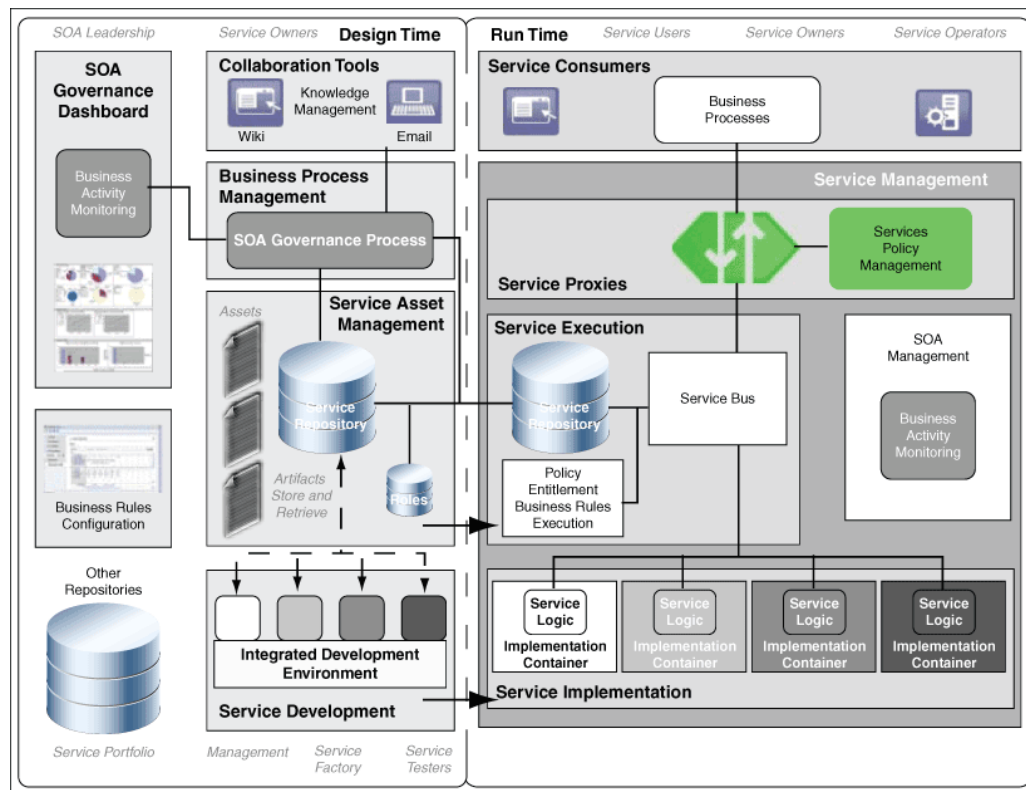
> **Note:**   You need not install all of the governance infrastructure, especially when first starting out. Organizations should select the components most relevant to their use cases.

1.2 Oracle's Governance Suite

This section describes the components in the Governance Suite. It contains the following topics:

- Section 1.2.1, "Oracle Enterprise Repository"
- Section 1.2.2, "Oracle Service Registry"
- Section 1.2.3, "Oracle Web Services Manager"
- Section 1.2.4, "Oracle Enterprise Manager Grid Control"

1.2.1 Oracle Enterprise Repository

Oracle Enterprise Repository provides design-time governance support for the service lifecycle, enabling storage and management of extensible metadata for composites, services, business processes, and other IT-related assets. Oracle Enterprise Repository acts as the central source of SOA information, allowing you to find planned, existing and retired services.

Oracle Enterprise Repository provides vital information to service producers, providers, and consumers who enhance applications or deliver new service capabilities. In addition, portfolio managers and business architects use the repository to understand business capabilities and to assist with strategic planning.

Oracle Enterprise Repository provides role-based links to asset artifact stores and links to design documents, justification documents, test plans, support plans, policies, and so on. Approvers in the lifecycle process can reference and upload documentary evidence to support their approvals, and consumers can reference information to support their service choices. Oracle Enterprise Repository features that help automate service lifecycle governance include:

- An Asset Harvester that automatically populates OER with SOA assets and tracks updates to artifacts automatically.

- Direct integration with IDEs, allowing prescription of assets for new projects and easy browsing and consumption of existing assets.

- Reporting on reuse, compliance, and other portfolio management metrics.

- A built-in process engine with a set of standard but modifiable processes to meet an organization's governance process workflows.

- Built-in automated notification about a selection of lifecycle events, plus discretionary notification of asset subscribers.

- An event engine that can publish repository events to any process service endpoint.

- A bi-directional registry exchange feature 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 synchronized with the IT and business environment that it describes.

In addition, Oracle Enterprise Manager SOA Management Pack Enterprise Edition provides Oracle Enterprise Repository with a summary of runtime performance metrics. These metrics help service providers judge portfolio performance and service consumers judge runtime performance.

Oracle Enterprise Repository also tracks asset usage and provides 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:

- Size Oracle Enterprise Repository

  For more information, see *http://www.oracle.com/technetwork/middleware/repository/overview/oer11gsizingguidelines-130307.pdf*.

- Choose a platform from the 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 use 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 can 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 Oracle Service Registry

Oracle Service Registry is a feature rich UDDI version 3 compliant service registry that provides standards-based interoperability and forms the runtime interface of the Repository. It provides service binding and runtime location transparency, federation of runtime metrics for closed loop governance, and access to an appropriate service version based on the environment. You can look up services published to the registry at runtime 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 Enterprise Manager SOA Management Pack Enterprise
> Edition 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   Service Registry
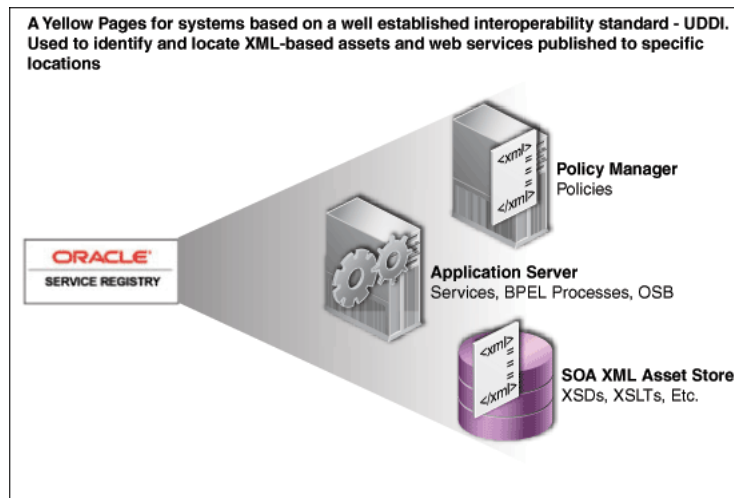
The following resources can help you when installing the Service Registry:

- 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*.

The Oracle Enterprise Repository and the Oracle Service Registry are tightly
integrated. The Oracle Registry Repository Exchange Utility synchronizes Oracle
Enterprise Repository and Oracle Service Registry bi-directionally so metadata can
flow in either direction through the utility.

1.2.3 Oracle Web Services Manager

Oracle Web Services Manager (OWSM) 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 be applied to Web services in a
heterogeneous environment without modification. 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.

OWSM provides end-to-end security between the service consumer and the service
provider by deploying client and server agents within the container itself. The OWSM
Policy Manager provides client and server agents with central policy information,

ensuring that the consumer can apply the correct policies even when the service security measures are updated.

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 Oracle Enterprise Manager Grid Control

Oracle Enterprise Manager SOA Management Pack Enterprise Edition builds upon the core Oracle Enterprise Manager Grid Control (Grid Control) product and provides operational management of complex SOA environments.

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

Figure 1–4   Oracle Enterprise Manager Grid Control
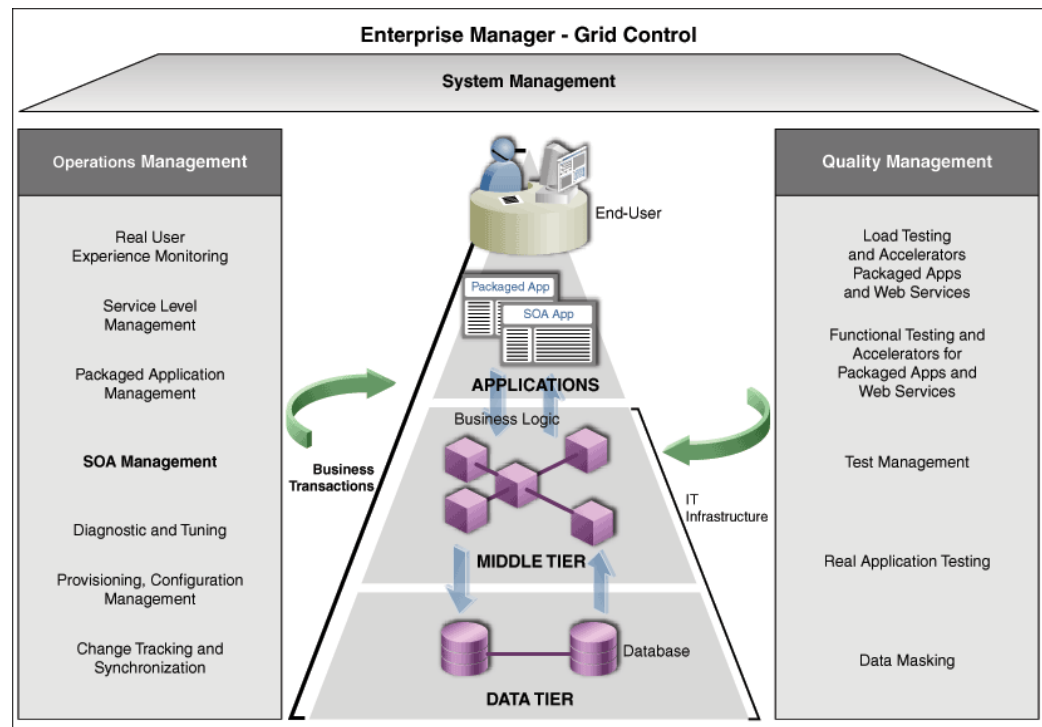
Grid Control also provides dynamic discovery and service-level monitoring of all artifacts deployed within a Java Application Server. Administrators can confirm that what is running in a particular SOA environment is exactly as defined at design time.

Grid Control collects statistics that include latency, invocation counts, and exceptions for each service component. It then aggregates and stores them to provide a rich dashboard of metrics presented across a user selected set of time-periods. You can

establish service-level objectives at each SOA architecture layer. You can set cautionary and violation level thresholds, which when breached over a defined period trigger alerts sent over standard protocols.

1.2.5 Oracle Enterprise Gateway (OEG)

Oracle Enterprise Gateway (OEG) is designed to secure SOA deployments on-premise, across domain boundaries, or in the cloud. It does this by providing an easier way to secure, accelerate, and integrate XML and other types of data. As a result it can significantly lower integration costs, lower costs of ownership, and reduce deployment risks. Oracle Enterprise Gateway also offers rich integration with many identity and access management platforms, and helps streamline regulatory compliance through authentication, authorization, and audit capabilities. OEG offers the following capabilities:

- DMZ-class security and comprehensive threat defense system for SOA and cloud environments

- Ultrafast XML processing, allowing SOA and cloud applications to offload resource-intensive, XML-based operations

- Open and standards-based, certified on Oracle Fusion Middleware and non-Oracle technologies

1.3 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"

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 existing version control systems. When developers need access to the physical artifact (such as an XSD or WSDL file) of an asset, Oracle Enterprise Repository delivers a copy of the code from the version control system.

This benefits your organization in several ways:

- When no behavioral change is required, the organization maintains its existing version control practices.

- When 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 or Source Control Management Systems.

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 or 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 through tight integration with the development environment (IDE). Through the IDE, 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.

Integrating with Supported IDEs:

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

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

- VS. NET

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 in use case documents, architecture documents, test cases, test results, and so on. Such documents are typically stored in version control systems, repositories, and file stores. However, you can create a link from the Enterprise Repository asset to the associated documents so developers can seamlessly access them.

This feature requires no connector or plug-in. 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

Oracle offers OWSM for runtime security policy management and enforcement. In addition, Oracle Enterprise Repository offers native policies to communicate design-time and development requirements to development teams. Policies might include:

- Corporate quality standards and guidelines

- Asset naming conventions

- Performance targets

Policies applied to assets communicate design and development requirements and help administrators enforce 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 acceptable defect density rates.

Oracle Enterprise Repository's native policies can be validated manually by a Subject Matter Expert or programmatically through third-party tooling, such as Parasoft and WebLayers.

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

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 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 stores runtime metadata and configuration information used by Oracle Fusion Middleware, including Oracle SOA Suite. Oracle Enterprise Repository can harvest and display metadata from MDS. Oracle Enterprise Repository 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 repository such as MDS.

The best practice is hosting all shared runtime metadata on a single MDS instance for access by all SOA infrastructures.

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

Because of the number of Web Service testing tools on the market, there is no single out-of-the-box testing tool connector. Instead, use the Repository Extensibility Framework (REX) to integrate with the testing tool of your choice. 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 previous 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 visibility into assets as they move throughout the lifecycle. To harvest information about assets at build time, integrate 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 manages 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 Oracle Enterprise Repository 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 shows an example of an Oracle customer's governance practices.

Figure 1–5 Example Governance Practice
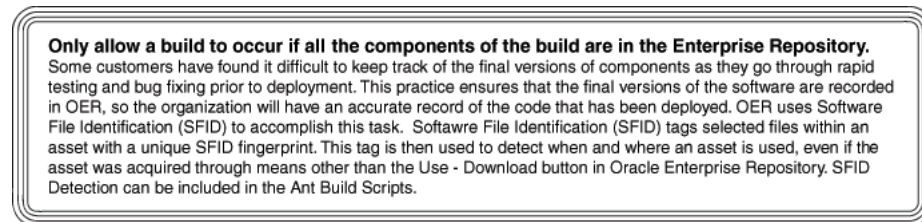
For more information about SFID, see "Configuring Automated Usage Detection" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

1.4.4 Defect Tracking Systems

Defect Tracking Systems track problems with assets after 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. You can 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 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: individual instances of Oracle Enterprise Repository 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*.

2

Governed Assets

This chapter describes best practices for identifying requirements and determining which assets to submit to Oracle Enterprise Repository.

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 acquire assets so that they can be managed and governed. This section provides a careful consideration of which assets to submit.

Oracle recommends that organizations bootstrap the enterprise repository in phases, with sharp focus on the assets that bring the greatest value and benefits.

For example, if your organization's primary focus is application development, 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, and so on)
- Connection pooling services
- Caching services
- Frameworks (STRUTS, security, and so on)
- Validation routines
- Web Services that are already in production that expose application interfaces

Every application contains domain-independent, highly reusable functionality. In addition, organizations typically want to standardize this functionality for consistency across applications. Initial productivity savings for reusing key assets can be impressive. Many organizations save several hundred thousand dollars in months due to reuse, in addition to savings 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 just want a communication mechanism for your project, bootstrap the Enterprise Repository with the information most relevant to the project team. This might include:

- Existing functionality that fulfills the project's functional and non-functional requirements

- A functionality and capabilities overview (the team can manage dependencies and determine what functionality is delivered by whom)

- Standards and best practices

- Open Source licenses and projects

- Commercial Off the Shelf (COTS) dependencies - which commercial packages and libraries are used

- Build labels - check in the repository location of the latest build, test results, and so on using the Ant task

- SCM information (location of the source code)

- Configured platform (for example - WLS configured for use in development environments)

- Web services that provide access to application interfaces needed for the project

2.2 Best Practices

This section describes the Governance Suite components. 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, include product line asset portfolios in the governance program first. If the goal is to establish a standard Software Development Lifecycle Process, include process templates and standards first.

Identify requirements

After identifying the asset portfolios, identify Asset Portfolio Management Team requirements for managing their respective portfolios. For example, Architects may need to package solution sets and monitor use and standards compliance. Service Competency Centers may need to communicate the services that orchestrate a business process. The Governance program distributes asset portfolios to help Portfolio Management Teams.

Structure the program to deliver value

When executives understand the vision, you can gain their sponsorship and structure a program with value for the asset portfolio management teams and their consumers. However, teams cannot be truly motivated to participate unless they can see tangible benefits.

The Bottom Line

Successful governance depends on a supply of assets that help the organization reach its goals. Asset Portfolio Management Teams are the primary sources of assets. The governance program must engage these teams and meet their needs.

For more information about the role of the Asset Portfolio Management team, see Chapter 4, "Stakeholder Identity".

Resources

Oracle offers a whitepaper and a workbook that helps you evaluate the expected ROI from your asset portfolios. This tool can help identify the most valuable assets.

- 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 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*.

Oracle's Unified Method (OUM) provides a wealth of additional governance information. OUM users include Oracle employees, Oracle Partner Network Certified Partners or Certified Advantage Partners, OUM Customer Program clients, and Oracle consulting services clients. 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

3

Visibility

This chapter describes using Oracle Enterprise Repository Solution Packs for managed asset visibility. Oracle Service Registry provides asset visibility when synchronized to the repository using Oracle Registry Repository Exchange Utility.

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 is to make the assets you want to manage visible.

Oracle Enterprise Repository Solution Packs include:

- 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)

Bootstrapping is the initial programmatic loading of assets and asset metadata into the Enterprise repository from sources such as:

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

Organizations can also enter assets manually. Afterward, manual validation ensures that imported assets were harvested properly and allows addition of business-focused metadata.

3.2 Oracle Enterprise Repository Solution Packs

This section describes the Oracle Enterprise Repository solution packs:

- "Base Data Pack"
- "Harvester Solution Pack (Includes Adapter Solution Pack)"

- "AIA Solution Pack"

Base Data Pack

Base Data Pack delivers template assets. These include the business-focused asset metadata that organizations track. The metadata fields have been collected from the best practices of organizations worldwide. The Base Data Pack is automatically installed with Oracle Enterprise Repository. Base Data types can be copied and customized to suit specific organizational assets.

Harvester Solution Pack (Includes Adapter Solution Pack)

The Harvester Solution Pack delivers assets required by the Harvester tool. The Harvester and Harvester Solution Pack provide metadata about an asset's implementation and relationships. The Harvester Solution Pack defines the format into which Oracle Enterprise Repository harvests artifacts.

The Oracle Enterprise Repository Harvester Solution Pack includes asset types for Oracle's Application Adapters and Applications. These asset types allow the harvest and end-to-end visibility of application integrations.

The Oracle Enterprise Repository Harvester tool makes visible assets from 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 how to bootstrap Oracle Enterprise Repository. It contains the following topics:

- Section 3.3.1, "Service Registry"
- Section 3.3.2, "Runtime Environment"
- Section 3.3.3, "Artifacts and Artifact Stores"
- Section 3.3.4, "Spreadsheets and Other Metadata Stores"

3.3.1 Service Registry

Effective SOA Governance requires a synchronized Enterprise Repository and Service Registry.

The Oracle Registry Repository Exchange Utility synchronizes Oracle Enterprise Repository and Oracle Service Registry so metadata can flow in either direction.

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 Runtime Environment

You can bootstrap the Enterprise Repository with assets from 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 harvest standards-based artifacts generated from any tooling. In the runtime environment, you can invoke the Harvester using the command line or 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 Artifacts and Artifact Stores

You can bootstrap the Enterprise Repository with production-grade assets from the organization's version control or file systems. Oracle Enterprise Repository uses the Harvester to access the artifacts in various artifact stores. You can also invoke the Harvester from the command line.

> **Note:** There are limitations on publishing services to OSR when the WSDL is hosted anywhere but on a running endpoint, HTTP, HTTP/S or FTP. Services with artifacts hosted in other types of artifact stores will not publish to OSR/UDDI.

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 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 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, which can bootstrap Oracle Enterprise

Repository from spreadsheets and other metadata stores. This requires customization and coding and is covered as an advanced topic.

For more information about the Enterprise Repository's API, see "Developing Custom Integrations" in *Oracle Fusion Middleware Integration Guide for Oracle Enterprise Repository*.

4

Stakeholder Identity

This chapter describes who stakeholders are, their typical duties, and how to assign roles to them in Oracle Enterprise Repository.

This chapter contains the following sections:

- Section 4.1, "Overview"
- Section 4.2, "Stakeholders"

4.1 Overview

After identifying governed assets, you work with stakeholders to:

- Determine how they can best leverage Oracle Enterprise Repository
- Identify additional metadata required
- Modify taxonomies and categorizations
- Customize their access and asset views

Figure 4–1   Closed Loop Governance Process
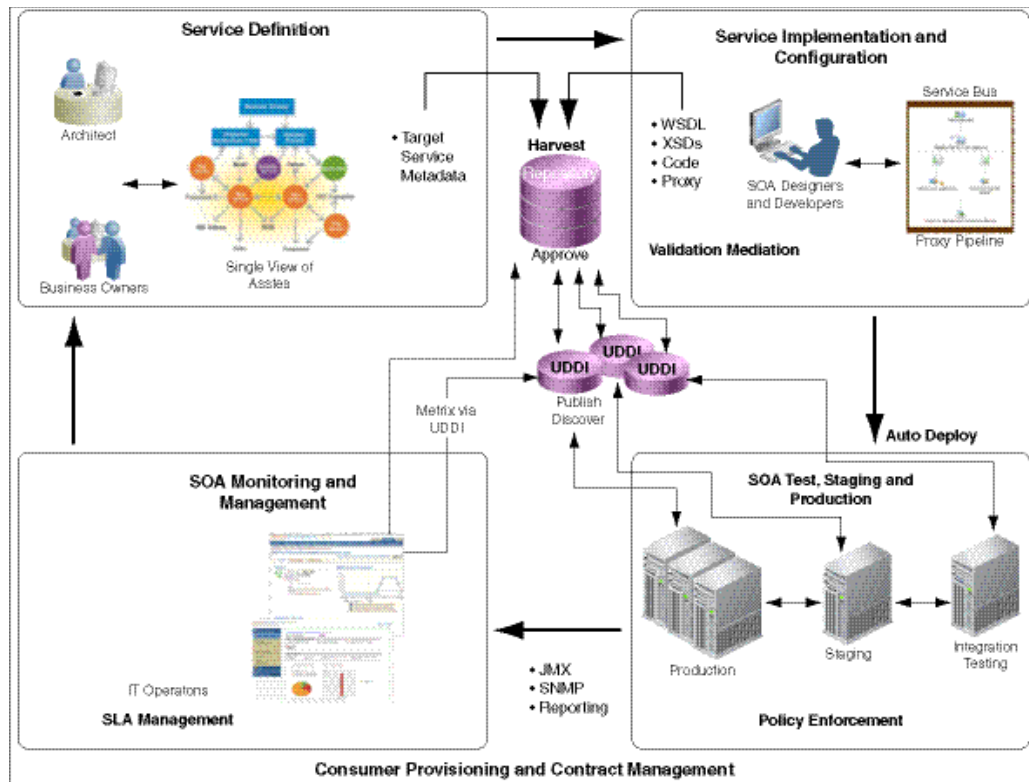
Figure 4–1 shows stakeholder participation in and infrastructure of the closed loop governance process.

- 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*.

- For more information about Governance structures and best-of-breed Governance solutions, see the OFRA Governance material at

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

4.2 Stakeholders

This section describes the Oracle Enterprise Repository stakeholders:

- Section 4.2.1, "Service Competency Center, Software Factory, or 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 or Harvesters)"

- Section 4.2.6, "Registrars"

- Section 4.2.7, "Enterprise Architects"

- Section 4.2.8, "Program Managers"
- 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, or 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, including inception, creation, distribution, usage, and 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 asset portfolio

  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 asset performance by reviewing usage history, developer feedback, runtime performance metrics, policy status, and compliance reports
- Use the Oracle Enterprise Repository Navigator to evaluate the impact asset modification
- 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 software assets for maximum reuse in architecture design (Service Level Agreements 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 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 when creating other assets

- Search the repository for suitable reusable assets for projects

- Integrate reusable assets and make any necessary modifications

Typical Roles Assigned in Oracle Enterprise Repository

User

4.2.5 Developers (as Producers or Harvesters)

Role in Governance

- Create reusable assets from the beginning or reengineer existing assets to be reusable

- Test assets and ensure quality standards

- Maintain assets

- Help 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

- Maintain operational aspects of the repository

- Ensure that work products meet quality, documentation, and other established standards

- 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 Managers

Role in Governance
- Coordinate planning, prioritization, monitoring, and support of the governance program, while aligning it with changing business needs
- Facilitate the definition of the program objectives
- Oversee 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
- Develop 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 use and reuse of applications, components, and services by LOB, product, and value chain

Typical Roles Assigned in Oracle Enterprise Repository
User

4.2.10 Executive Sponsor

Role in Governance

- Represents management interests

- Authorizes, encourages, and supports resource use

- Shows that management supports the program

Typical Roles Assigned in Oracle Enterprise Repository
User

4.2.11 Governance Champion

Role in Governance

- Supports the program through organization-wide evangelism

- Educates others on program concepts, long-term benefits, inhibitors, and purpose

- Sets implementation direction

- Communicates successful efforts throughout the organization

Typical Roles Assigned in Oracle Enterprise Repository
User

5

Asset Types and Taxonomies

This chapter describes asset types, harvested assets, taxonomies, and categorization.

This chapter contains the following sections:

5.1 Overview of Assets and Asset Types

The term **asset** refers to anything 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
- Policies and policy intent

Not all assets are code. An asset can be a process, the IT roadmap, or any number of documents. Flexible asset types support the following:

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

5.1.1 Asset Types

Oracle Enterprise Repository can model any asset type. The Type Manager is a user-friendly tool for modifying the solution pack asset types and defining new ones. The Type Manager provides many element and display types, which you can arrange in a way that makes sense for an Asset Consumer or an Asset Producer or Provider.

5.1.1.1 Asset Type Best Practices

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 are best practices for modifying asset types to meet stakeholder needs.

- 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:** This is especially important 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, because those copies have a different Unique Identifier.

- When working with stakeholders, use templates to identify metadata fields needed for each asset. Stakeholders more easily identify metadata fields when looking at a list. After stakeholders have identified the metadata for each asset, you can develop a prototype in Oracle Enterprise Repository, which stakeholders can validate.

  - Out-of-the-box metadata for each Oracle Enterprise Repository asset type is listed at:
    http://www.oracle.com/technetwork/middleware/repository/overview/asset.xls

    Use this as a template for identifying the metadata needed by stakeholders identified in Chapter 4.2, "Stakeholders".

- Oracle Enterprise Repository offers two views of asset metadata. The Viewer shows the metadata to which all users have read-only access. The Editor is a template-based view available to those who are entering or editing asset information. Organize metadata in the Editor so it is easy for those entering or editing asset information to know which metadata fields to populate.

  - The Editor can have multiple, customizable tabs. We recommend organizing these tabs by user role, so 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 needed at each stage of the asset's lifecycle.

  - Note that Oracle Enterprise Repository's automated workflows use the Tabs to process assets. Maintaining consistency across asset types results in consistent processing by workflows.

5.1.2 Harvested Assets

Oracle Enterprise Repository harvests assets according to an underlying model. For example, Oracle Enterprise Repository harvests a JDeveloper 11g project into an underlying composite asset model. The model determines which assets appear in Oracle Enterprise Repository and 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 represents the underlying model. When the composite is harvested, it is interpreted according to the underlying model. The results are displayed in Figure 5–2.

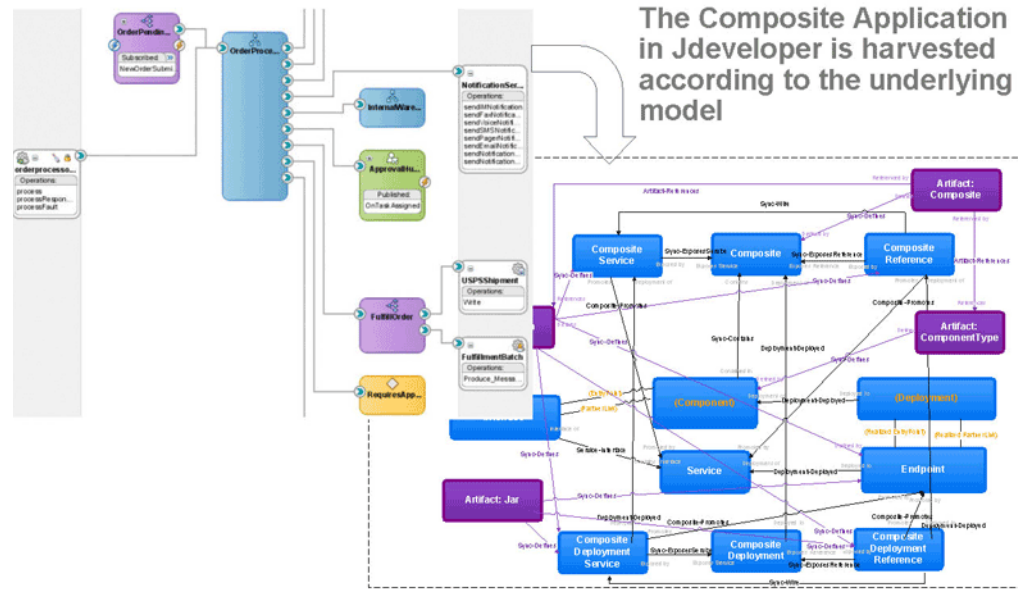**Figure 5–1   JDeveloper Composite and Resulting Oracle Enterprise Repository Composite Model**

Figure 5–2   Composite Harvesting Results

The Solution Packs provide the underlying models that define how harvested assets are displayed in Oracle Enterprise Repository. You can add metadata fields to the Base and solution pack asset types to support stakeholder information requirements.

5.1.3 Categorizations

Categorizations classify and or organize assets and projects within Oracle Enterprise Repository. They appear as asset metadata and provide robust searching capabilities. Both the Browse Tree searches and the Advanced Search capabilities in the Oracle Enterprise Repository Home page use them. Anything hierarchical or taxonomical can be a categorization. Examples include Architecture Reference Models, Asset Lifecycle Stages, Lines of Business, Domains, Business Capabilities, and so on.

Oracle Enterprise Repository ships with many categorizations. These can be customized to reflect concepts relevant to your organization.

- Think about how users search for assets, such as by line of business or technology.

- Think about how teams organize their assets, such as by lifecycle stage or 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*.

6

The Asset Lifecycle

This chapter describes the asset lifecycle stages, including service definition, implementation and configuration, testing, staging and production, and monitoring and management.

This chapter contains the following sections:

- Section 6.1, "Overview"
- Section 6.2, "Asset Lifecycle"

6.1 Overview

After assets are harvested and made visible to stakeholders, the next step is inserting governance into the asset lifecycle process. Figure 6–1 illustrates the asset lifecycle process.

1. First, a capability needed by multiple work groups is defined.

2. The capability is funded and a project is initiated to implement it.

3. The implementation is promoted to pre-production environments, which might include staging and testing.

4. Finally, the implementation is promoted to the production environment, where performance can be monitored and behavior enforced. Oracle Enterprise Repository's automated workflows facilitate this process.

Figure 6–1 Asset Lifecycle Process
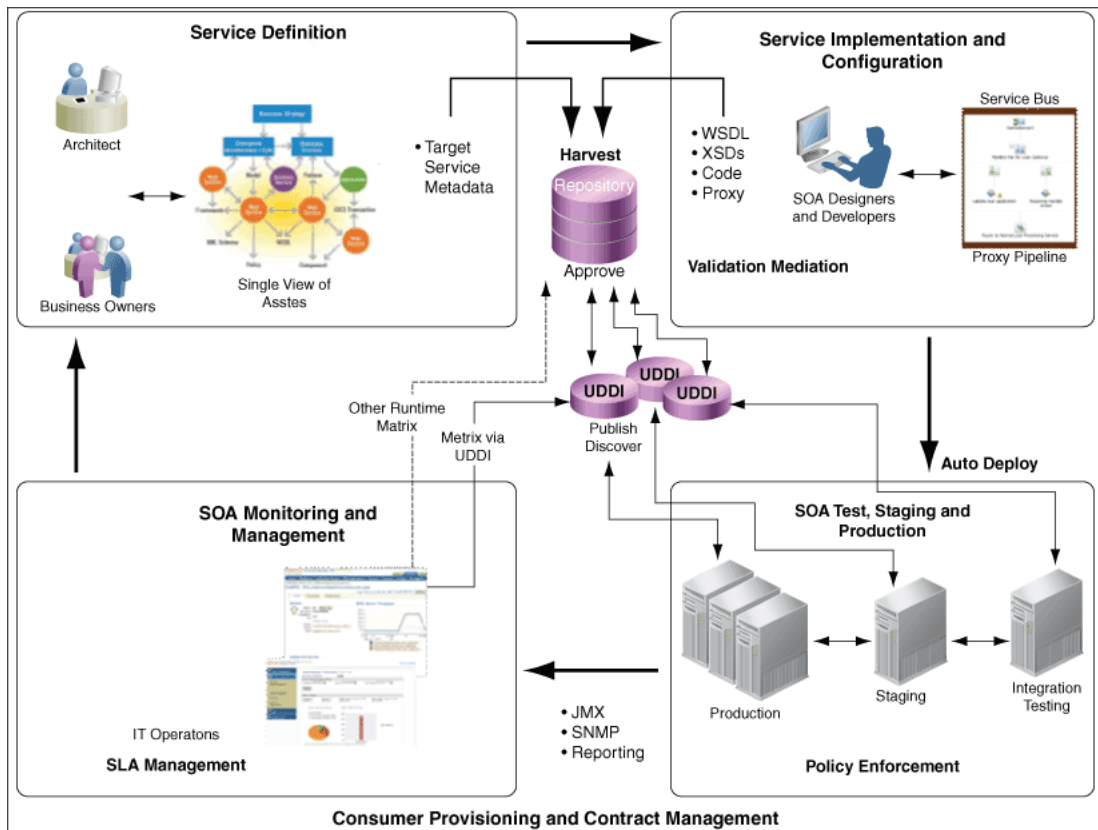
6.2 Asset Lifecycle

This section describes Governance in each asset lifecycle phase. This section contains the following topics:

- Section 6.2.1, "Service 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 Service Definition

When a needed capability is recognized, the Enterprise Repository can make it visible. Capabilities may come from architects working on target architecture or business analysts identifying common business requirements. After these capabilities are visible in the enterprise repository, they can be evaluated and prioritized according to a common business or ROI model. Next, these capabilities can be categorized into stages such as proposed (identified and vetted), planned (in progress in an active project), and released (fully implemented).

You can make a capability visible through the Enterprise Repository in these ways:

- Quick Submission

- Advanced Submission

Information and a workbook are available to help you justify and prioritize your capabilities and 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 planned, projects implement them. SOA Suite developers working in JDeveloper can see and reuse enterprise repository services to complete their projects. Service Bus developers working in Eclipse can see and reuse enterprise repository services 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

After an implementation has been harvested to Oracle Enterprise Repository, the repository's workflows process it according to established governance rules and practices. Oracle Enterprise Repository comes with several preconfigured, customizable workflows.

For more information about Oracle Enterprise Repository's workflows and how to customize them, see "Configuring Oracle Enterprise Repository Workflow" in *Oracle Fusion Middleware Configuration Guide for Oracle Enterprise Repository*.

As an implementation moves through the lifecycle, from testing, through staging, and into production, Oracle Enterprise Repository harvests and makes visible the endpoints for each environment. The Harvester is incorporated into the Ant or WLST deployment script.

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*.

Using the Exchange Utility, Oracle Enterprise Repository can optionally promote the services and harvested endpoints to a Service Registry in each lifecycle environment. Runtime tooling, such as OSB and SOA Suite, can dynamically discover changes to services in the registry. OSB subscribes to the Service Registry's publish/subscribe API, therefore OSB is automatically notified when a change occurs. Alternatively, SOA Suite caches the WSDL and endpoint locations and the UDDI service key at design time. If the WSDL or endpoints become invalid, SOA Suite uses the service key to dynamically discover and update them. Both OER and OSR honor the unique key for each service. Applications can access services by their keys instead of their endpoints, which may change throughout the lifecycle.

- 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.

6.2.5 Monitoring and Management

Both service providers and service consumers need to understand how a service is performing in the runtime environment. Oracle Enterprise Repository 11*g* can import a summary of runtime service performance metrics from Enterprise Manager SOA Management Pack Plus.

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

Some third-party tools can publish runtime metrics to Oracle Service Registry. Bidirectional synchronization between Oracle Enterprise Repository and Oracle Service Registry moves runtime metrics from the registry to the repository.

Note that Oracle Enterprise Manager SOA Management Pack Enterprise Edition 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*.

7

Reporting

This chapter describes reporting concepts in Oracle Enterprise Repository, using Oracle BI Publisher as its reporting engine.

Oracle Enterprise Repository tracks and reports on the design-time use of assets. Usage is tracked in two ways: through the manual asset Use - Download process, and through the IDE.

Oracle Enterprise Repository leverages Oracle BI Publisher as its reporting engine. Oracle Enterprise Repository offers more than twenty preconfigured 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