Oracle® E-Business Suite

Integrated SOA Gateway User's Guide

Release 12.2

Part No. E20923-20

December 2023

ORACLE®

Contents

5   Working with Composite Services - BPEL

6   Working with Custom Integration Interfaces

Glossary

Index

Send Us Your Comments

Oracle E-Business Suite Integrated SOA Gateway User's Guide, Release 12.2

Part No. E20923-20

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Oracle E-Business Suite Release Online Documentation CD available on My Oracle Support and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: appsdoc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Preface

Intended Audience

Welcome to Release 12.2 of the *Oracle E-Business Suite Integrated SOA Gateway User's Guide.*

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.

- Computer desktop application usage and terminology.

- Oracle E-Business Suite integration interfaces.

- B2B, A2A and BP integrations.

This documentation assumes familiarity with Oracle E-Business Suite. It is written for the technical consultants, implementers and system integration consultants who oversee the functional requirements of these applications and deploy the functionality to their users.

If you have never used Oracle E-Business Suite, we suggest you attend one or more of the Oracle E-Business Suite training classes available through Oracle University.

Documentation Accessibility

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

Access to Oracle Support

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

Structure

1 Introduction to Oracle E-Business Suite Integrated SOA Gateway
2 Understanding Service Enablement
3 Navigating Through Oracle Integration Repository
4 Working with Native Integration Interfaces
5 Working with Composite Services - BPEL
6 Working with Custom Integration Interfaces
Glossary

Related Information Sources

This book is included in the Oracle E-Business Suite Documentation Library. If this guide refers you to other Oracle E-Business Suite documentation, use only the latest Release 12.2 versions of those guides.

Online Documentation

All Oracle E-Business Suite documentation is available online (HTML or PDF).

- **Online Help** - Online help patches (HTML) are available on My Oracle Support.

- **Oracle E-Business Suite Documentation Library** - This library, which is included in the Oracle E-Business Suite software distribution, provides PDF documentation as of the time of each release.

- **Oracle E-Business Suite Documentation Web Library** - This library, available on the Oracle Help Center (https://docs.oracle.com/cd/E26401_01/index.htm), provides the latest updates to Oracle E-Business Suite Release 12.2 documentation. Most documents are available in PDF and HTML formats.

- **Release Notes** - For information about changes in this release, including new features, known issues, and other details, see the release notes for the relevant product, available on My Oracle Support.

- **Oracle Electronic Technical Reference Manual -** The Oracle Electronic Technical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for each Oracle E-Business Suite product. This information helps you convert data from your existing applications and integrate Oracle E-Business Suite data with non-Oracle applications, and write

custom reports for Oracle E-Business Suite products. The Oracle eTRM is available as an application in Oracle E-Business Suite.

Related Guides

You should have the following related books on hand. Depending on the requirements of your particular installation, you may also need additional manuals or guides.

Oracle E-Business Suite Concepts

This book is intended for all those planning to deploy Oracle E-Business Suite Release 12.2, or contemplating significant changes to a configuration. After describing the Oracle E-Business Suite architecture and technology stack, it focuses on strategic topics, giving a broad outline of the actions needed to achieve a particular goal, plus any installation and configuration choices that are available.

Oracle E-Business Suite Electronic Technical Reference Manual User's Guide

This guide describes how to set up and navigate Oracle E-Business Suite Electronic Technical Reference Manual (eTRM) user interface in Oracle E-Business Suite. It also explains how to browse and search the Oracle eTRM repository to locate desired FND and database metadata and objects, and how to view object details, reports, and diagrams.

Oracle E-Business Suite Security Guide

This guide contains information on a comprehensive range of security-related topics, including access control, user management, function security, data security, secure configuration, and auditing. It also describes how Oracle E-Business Suite can be integrated into a single sign-on environment.

Oracle E-Business Suite Integrated SOA Gateway Developer's Guide

This guide describes how integration developers can perform end-to-end service integration activities. These include orchestrating discrete web services into meaningful end-to-end business processes using business process execution language (BPEL), and deploying BPEL processes at runtime.

This guide also explains how to invoke web services using the Service Invocation Framework. This includes defining web service invocation metadata, invoking web services, and testing web service invocation.

Oracle E-Business Suite Integrated SOA Gateway Implementation Guide

This guide explains how integration administrators can manage and administer the web service activities for integration interfaces including native packaged integration interfaces, composite services (BPEL type), and custom integration interfaces. It also describes how to set up and implement Service Invocation Framework to invoke SOAP and REST services from Oracle E-Business Suite, and how to manage web service security, configure logs, and monitor both inbound service invocations using Service Monitor and outbound service invocations through Service Invocation Framework using Service Invocation Monitor.

Oracle Fusion Middleware Oracle E-Business Suite Adapter User's Guide

This book covers the use of Oracle E-Business Suite Adapter (formerly known as Adapter for Oracle Applications in Oracle Fusion Middleware 11g releases) in developing integrations between Oracle E-Business Suite and trading partners.

This book is available in the Oracle Fusion Middleware 12c Documentation Library and Oracle Fusion Middleware 11g Documentation Library.

Oracle E-Business Suite User's Guide

This guide explains how to navigate, enter and query data, and run concurrent requests using the user interface (UI) of Oracle E-Business Suite. It includes information on setting preferences and customizing the UI. In addition, this guide describes accessibility features and keyboard shortcuts for Oracle E-Business Suite.

Oracle e-Commerce Gateway User's Guide

This guide describes the functionality of Oracle e-Commerce Gateway and the necessary setup steps in order for Oracle E-Business Suite to conduct business with trading partners through Electronic Data Interchange (EDI). It also contains how to run extract programs for outbound transactions, import programs for inbound transactions, and the relevant reports.

Oracle iSetup User's Guide

This guide describes how to use Oracle iSetup to migrate data between different instances of the Oracle E-Business Suite and generate reports. It also includes information on configuration, instance mapping, and seeded templates used for data migration.

Oracle Workflow User's Guide

This guide describes how users can view and respond to workflow notifications and monitor the progress of their workflow processes.

Oracle XML Gateway User's Guide

This guide describes Oracle XML Gateway functionality and each component of the Oracle XML Gateway architecture, including Message Designer, Oracle XML Gateway Setup, Execution Engine, Message Queues, and Oracle Transport Agent. It also explains how to use Collaboration History that records all business transactions and messages exchanged with trading partners.

The integrations with Oracle Workflow Business Event System, and the Business-to-Business transactions are also addressed in this guide.

Integration Repository

The Oracle Integration Repository is a compilation of information about the service endpoints exposed by the Oracle E-Business Suite of applications. It provides a complete catalog of Oracle E-Business Suite's business service interfaces. The tool lets users easily discover and deploy the appropriate business service interface for integration with any system, application, or business partner.

The Oracle Integration Repository is shipped as part of the Oracle E-Business Suite. As your instance is patched, the repository is automatically updated with content appropriate for the precise revisions of interfaces in your environment.

Do Not Use Database Tools to Modify Oracle E-Business Suite Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle E-Business Suite data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle E-Business Suite data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle E-Business Suite tables are interrelated, any change you make using an Oracle E-Business Suite form can update many tables at once. But when you modify Oracle E-Business Suite data using anything other than Oracle E-Business Suite, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle E-Business Suite.

When you use Oracle E-Business Suite to modify your data, Oracle E-Business Suite automatically checks that your changes are valid. Oracle E-Business Suite also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

1

Introduction to Oracle E-Business Suite Integrated SOA Gateway

Oracle E-Business Suite Integrated SOA Gateway Overview

Building on top of Oracle Fusion Middleware and service-oriented architecture (SOA) technology, Oracle E-Business Suite Integrated SOA Gateway (ISG) is a complete set of service infrastructure to provide, consume, and administer Oracle E-Business Suite services.

With service enablement feature, integration interfaces published in the Oracle Integration Repository can be transformed into SOAP and REST based web services.

By leveraging Oracle SOA Suite running on Oracle WebLogic Server, Oracle E-Business Suite Integrated SOA Gateway provides greater capabilities and infrastructure for exposing various integration interfaces within Oracle E-Business Suite as SOAP services. SOAP-based services are described in WSDLs and are deployed to Oracle SOA Suite for service consumption.

Unlike SOAP services, REST services, without the dependency on Oracle SOA Suite, are developed with the infrastructure of Oracle E-Business Suite. REST services described in WADLs are directly deployed to an Oracle E-Business Suite WebLogic environment. They can be used for user-driven applications such as Oracle E-Business Suite mobile applications.

Oracle E-Business Suite Integrated SOA Gateway provides service invocation framework to invoke and consume web services provided by other applications.

Major Features

Oracle E-Business Suite Integrated SOA Gateway can do the following:

- Display all Oracle E-Business Suite integration interface definitions through Oracle Integration Repository

- Support custom integration interfaces from Oracle Integration Repository

- Provide service enablement capability (SOAP and REST services) for seeded and custom integration interfaces within Oracle E-Business Suite

- Use the Integration Repository user interface to perform design-time activities such as generate and deploy Oracle E-Business Suite web services

- Support synchronous and asynchronous (callback without acknowledgement only) interaction patterns for SOAP-based services

  > **Note:** In this release, only PL/SQL APIs can be enabled with the support for asynchronous service pattern.

- Support synchronous interaction pattern for REST-based services

  > **Note:** In this release, only PL/SQL APIs, Concurrent Programs, Java Bean Services, Application Module Services, Business Service Objects, Open Interface Tables, and Open Interface Views can be exposed as REST services.

- Support multiple authentication types for inbound service requests in securing web service content

- Enforce function security and role-based access control security to allow only authorized users to process administrative functions

- Provide centralized, user-friendly logging configuration for the services generated through service provider from Oracle E-Business Suite Integrated SOA Gateway

- Audit and monitor Oracle E-Business Suite inbound service operations from Service Monitor

- Audit and monitor outbound service invocations from Oracle E-Business Suite through Service Invocation Monitor

- Leverage Oracle Workflow Business Event System to enable service invocation from Oracle E-Business Suite

Major Components of Oracle E-Business Suite Integrated SOA Gateway

Oracle E-Business Suite Integrated SOA Gateway has the following essential components in enabling Oracle E-Business Suite services. The relationship between these essential components can be illustrated in the following diagram:

Web Service Architecture in Oracle E-Business Suite
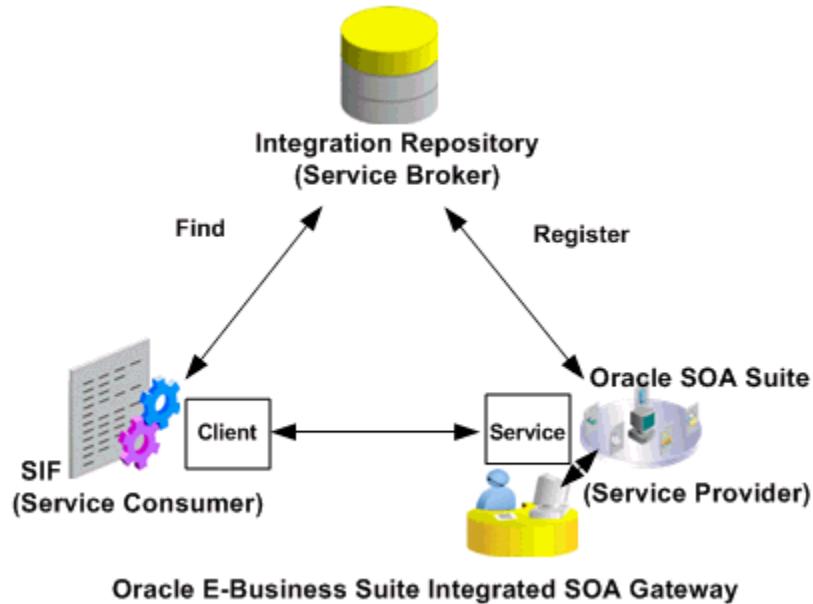
- **Oracle Integration Repository** (Service Broker), an integral part of Oracle E-Business Suite, is the centralized repository that contains numerous interface endpoints exposed by applications within the Oracle E-Business Suite. It supports various interface types including PL/SQL, XML Gateway, Concurrent Program, Business Event, Open Interface Table/View, EDI, Business Service Object (formerly known as Service Beans) and Java APIs.

  Oracle E-Business Suite Integrated SOA Gateway leverages Oracle Integration Repository to provide the capabilities of service generation and deployment, as well as service lifecycle management.

  Application users can browse these interface definitions and services through Oracle Integration Repository and view the interface details. Users who have the Integration Administrator role can perform design-time operations, monitor the services, and view log messages.

- **Service Provider** (previously known as SOA Provider) is the primary engine underlying the web services. It is the engine that performs the actual service generation and deployment behind the scene.

- **Service Invocation Framework** serves as a service consumer to send a request through the invocation of a web service from Oracle E-Business Suite. By leveraging Oracle Workflow Java Business Event System (JBES) and seeded Java rule functions, this framework provides an infrastructure that facilitates SOAP and REST service invocation and consumption from Oracle E-Business Suite.

Additionally, Oracle E-Business Suite Integrated SOA Gateway provides the following

monitoring and auditing tools to help monitor and view SOAP and REST services in and out from Oracle E-Business Suite:

- **Service Monitor**: Audit and monitor inbound Oracle E-Business Suite service operations provided through Oracle E-Business Suite Integrated SOA Gateway.

- **Service Invocation Monitor**: Audit and monitor outbound Oracle E-Business Suite service invocations provided through Service Invocation Framework.

Oracle E-Business Suite Web Service Development Life Cycle

Oracle E-Business Suite Integrated SOA Gateway provides a capability of allowing various users to perform different tasks and to monitor and manage service integration throughout the entire service deployment life cycle.

- At development phase, users who have the Integration Developer role can create custom interfaces, and annotate custom interface's definitions. Users who have the Integration Administrator role can validate and upload annotated custom interfaces to the Integration Repository where all the registered interfaces, regardless of custom or Oracle packaged ones, can be viewed and accessed by all users.

- At design time, users who have the Integration Administrator role can generate SOAP services with desired operation patterns, and deploy them to Oracle SOA Suite by attaching an appropriate security policy. For interfaces that can be exposed as REST services, the administrator can select desired service operations before deploying them to Oracle E-Business Suite.

- At runtime, web service clients send request messages to invoke Oracle E-Business Suite services enabled through the service provider provided through ISG. After authenticating and authorizing the users who request the services, services can be invoked.

  Users who have the Integration Administrator role are responsible for monitoring and managing the entire service deployment life cycle.

2

Understanding Service Enablement

Service Enablement Overview

Oracle E-Business Suite applications are developed through various technologies or written in different forms, such as PL/SQL, Java, Concurrent Programs, and so on. These applications or programs reside either in the Oracle E-Business Suite database or on the application tier. Features such as Business Events System from Oracle Workflow and products such as Oracle XML Gateway for integrating Oracle E-Business Suite with trading partners are also widely used.

To accomplish the goal of integrating Oracle E-Business Suite applications with other systems, these programs written in various formats need to interact with external sources. This is achieved by service enabling or exposing these programs as web services.

A *web service* is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format, such as WSDL or WADL. Other systems interact with the web service in a manner prescribed by its description using SOAP messages for WSDL-based services or using REST messages for WADL-based services. Web services are loosely-coupled, self-describing, reusable software components encapsulating discrete functionality, which are programmatically accessible using standard based protocols.

The process of *service enablement* involves generating service artifacts, such as XSD, to validate XML messages, WSDL or WADL to describe the web service, and deploying them on an application server so that the services are available to clients over the web.

The WSDLs described for SOAP services can be used either to create clients which invoke the deployed SOAP services directly, or use Oracle SOA Suite BPEL component to create a composite application which coordinates the flow of data between various web services to accomplish a business process.

The WADLs described for REST services can be used to create clients which invoke the deployed REST services for mobile applications or UI applications.

The architectural style involving collection of loosely-coupled services that

communicate with each other using standard based technologies is referred as *service-oriented architecture* (SOA).

Common Terms Used in Web Services

Service-oriented architecture is a set of principles and methodologies for designing and developing software in the form of interoperable services. Web services building around SOA-based approach or technologies are reusable, scalable, and platform independent. To better understand the concept of web services, the following common terminologies are explained in this section.

Web Services Discovery

Web services provide access to software systems over the Internet using standard protocols. Therefore, there exists at least a *service provider* that publishes certain services such as computer repair services, and a *service consumer* that uses the services. Web service discovery is the process of finding a suitable web service for a given task.

Simple Object Access Protocol (SOAP)

SOAP is a protocol for exchanging XML-based messages over networks, normally using HTTP/HTTPS.

Web Services Description Language (WSDL)

WSDL is a format for describing a SOAP-based service interface. It is a way to describe services and how they should be bound to specific network addresses.

Representational State Transfer (REST)

REST is an architecture principle in which the web services are viewed as resources and can be uniquely identified by their URLs. The key characteristic of a REST service is the explicit use of HTTP methods (GET, POST, PUT, and DELETE) to denote the invocation of different operations.

The following table lists the interfaces that can be exposed as REST services and their supported HTTP methods:

REST-based Interfaces with Supported HTTP Methods

Interface Type	Supported HTTP Methods
Concurrent Program	POST only
PL/SQL API	POST and GET
Business Service Object	POST and GET

Interface Type	Supported HTTP Methods
Java Bean Service	POST and GET
Application Module Service	POST and GET
Open Interface Table (Inbound)	POST, GET, PUT, and DELETE
Open Interface Table (Outbound)	GET only
Open Interface View	GET only

Please note that for Open Interfaces Tables, the supported HTTP methods are determined by the direction (Inbound or Outbound) of the interfaces.

Web Application Description Language (WADL)

WADL is designed to provide a machine-processable description of HTTP-based web applications. It models the resources provided by a service and the relationships between them.

Web Service Interaction Pattern

Interaction pattern is the way that a web service client can communicate with the service. Interaction pattern can be synchronous and asynchronous.

Web Service Security

Web service security (WS-Security) is a specification to enable applications to conduct secure message exchanges. It provides quality of protection through message integrity, message confidentiality, and single message authentication.

XML (Extensible Markup Language)

XML is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

JSON (JavaScript Object Notation)

JSON is a text-based open standard designed for human-readable data interchange. The JSON format is often used with REST services to transmit structured data between a server and web application, serving as an alternative to XML.

HTTP Clients

Hyper-Text Transfer Protocol (HTTP) is a significant protocol used over the web. HTTP clients are the parties that use and consume the HTTP-based services, such as REST services, provided through HTTP protocol.

Service Enablement in Oracle E-Business Suite

Service enablement is one of the essential features in Oracle E-Business Suite Integrated SOA Gateway. It allows native packaged integration interface definitions written in PL/SQL, Java, and other formats and stored in Oracle Integration Repository to be transformed into web services. This in turn enables all Oracle E-Business Suite services to integrate with other systems over the web.

The following diagram illustrates the high level service enablement process flow within Oracle E-Business Suite:

Service Enablement Functional Process Flow
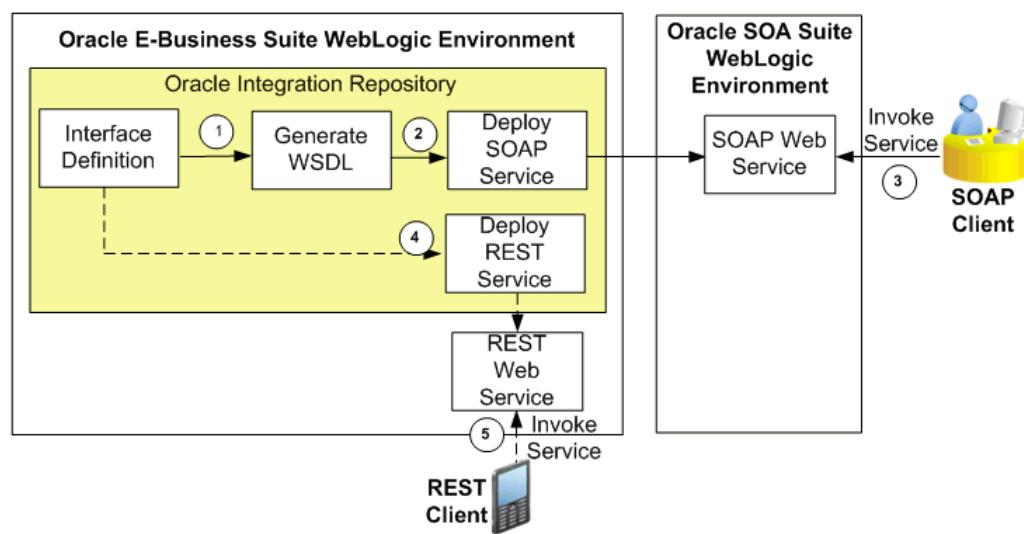
1. An integration administrator or an integration developer transforms the integration interface definitions resided in Oracle Integration Repository into **SOAP**-based services (with web service status "Generated") described in WSDLs.

2. An integration administrator deploys the generated SOAP services as deployed services (with web service status "Deployed"). SOAP services are deployed to an Oracle SOA Suite WebLogic managed server.

3. At runtime, web service clients send inbound requests and invoke Oracle E-Business Suite SOAP services through Oracle SOA Suite.

4. If the selected interfaces can be exposed as **REST** services, an integration administrator can deploy the interfaces as REST services (with web service status "Deployed") described in WADLs. REST services are deployed to an Oracle E-Business Suite managed server.

5. At runtime, REST services which are commonly used for mobile applications can

create or update Oracle E-Business Suite application data.

For more service enablement on SOAP services, see SOAP Service Enablement, page 2-5.

For more service enablement on REST services, see REST Service Enablement, page 2-6.

SOAP Service Enablement

WSDL-based SOAP Services

In a web service environment, SOAP services provide a standard way of structuring XML documents and act as building blocks for web service communication. For example, a service provider receives SOAP requests from web service clients to invoke services and also sends the corresponding SOAP responses to the clients.

At design time, an integration administrator or an integration developer can transform the interface definitions resided in Oracle Integration Repository into SOAP-based services described in WSDLs.

Support with Synchronous and Asynchronous Interaction Patterns

Before service generation, appropriate interaction patterns need to be identified for the desired service operations. Oracle E-Business Suite Integrated SOA Gateway supports both synchronous and asynchronous service processing for SOAP-based services.

- **Synchronous Pattern**: This type of service processing provides an immediate response to a query. The client connection remains open from the time the request is submitted to the server. The client will wait until the server sends back the response message.

- **Asynchronous Pattern**: Unlike the synchronous processing to obtain the result immediately, an asynchronous service may require a significant amount of time to process a request. When the service request processing completes, the response will be sent to the callback address mentioned by the client.

For information on how to select interaction patterns, see Generating SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

SOAP Service Security

To protect application data from unauthorized access, Oracle E-Business Suite integrated SOA Gateway enforces the security rules through subject authentication and authorization. Before service deployment, the administrator must select one desired authentication method.

- To authenticate users who request Oracle E-Business Suite Web services, the SOAP messages must be authenticated using UsernameToken or SAML Token based security. The identified authentication information is embedded in the `wsse:security` Web Security headers.

- To authorize users on specific services or operations, the access permissions must be explicitly given to the users through security grants. Multiple organization access control (MOAC) security rule is also implemented for authorizing interface access related to multiple organizations.

For information on how to specify a desired authentication type, see Deploying and Undeploying SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

Web Services Description Language (WSDL)

After service generation or deployment, users can view the generated or deployed WSDL link for the associated SOAP service.

> **Note:** A deployed WSDL shows the physical location of service endpoint where the service is hosted in soa-infra in this release.

The WSDL URL can be used either to create clients which invoke the deployed SOAP services directly, or use Oracle SOA Suite BPEL component to create a composite application which coordinates the flow of data between various web services to accomplish a business process. At runtime, web service clients send inbound requests and invoke Oracle E-Business Suite SOAP services through Oracle SOA Suite.

For information on how to generate and deploy SOAP services and other administrative tasks, see Administering SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

REST Service Enablement

WADL-based REST Services

REST (Representational State Transfer) is an architecture principle in which the web services are viewed as resources and can be uniquely identified by their URLs. The key characteristic of REST services is the explicit use of HTTP methods to denote the invocation of different operations. This kind of web services serves as a simpler alternative to SOAP services and allows you to access services over the web.

REST messages are supported with the XML and JSON (non-XML data) formats in conjunction with other web-related standards.

Support with Synchronous Interaction Pattern Only

Unlike SOAP services, REST services are all generated with the support of synchronous request-response and request-only interaction patterns only. Asynchronous interaction pattern is not supported for REST services in this release.

REST Service Security

Users who try to invoke Oracle E-Business Suite REST services must be authenticated using HTTP Basic Authentication or Token Based Authentication at the HTTP transport level.

Web Application Description Language (WADL)

If an interface can be exposed as a REST service, the corresponding deployed WADL description can be viewed in a separate window.

WADL is designed to provide a machine-processable description of HTTP-based web applications. It models the resources provided by a service and the relationships between them. WADL is intended to simplify the reuse of web services that are based on the existing HTTP architecture of the web. It is platform and language independent and aims to promote reuse of applications beyond the basic use in a web browser.

The WADL URL can be used to create clients which invoke the deployed REST services.

At runtime, web service clients send inbound REST requests and invoke Oracle E-Business Suite REST services.

For information on how to deploy REST services and other administrative tasks, see Administering REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

3

Navigating Through Oracle Integration Repository

This chapter covers the following topics:

- Oracle Integration Repository Overview
- Getting Started with Oracle Integration Repository
- Discovering and Reviewing Interfaces

Oracle Integration Repository Overview

Oracle E-Business Suite Integrated SOA Gateway is the intrinsic part of Oracle E-Business Suite for service enablement. It provides the capability of invoking web services and allowing web service clients to make use of the services provided from Oracle E-Business Suite.

To accomplish this goal, there must be a centralized location where all service related business interfaces can be stored, and at the same time all application users can browse through these business interfaces knowing what type of services are available for service consumption. Oracle Integration Repository, an integral part of Oracle E-Business Suite, is the repository to serve this purpose.

Oracle Integration Repository contains numerous interface endpoints exposed by applications throughout the entire Oracle E-Business Suite. It is not only an essential component within Oracle E-Business Suite Integrated SOA Gateway, but also provides a complete catalog of integration interfaces within Oracle E-Business Suite. You can use this tool to easily discover and search on interfaces, regardless of custom or Oracle seeded ones.

Integration Interface Types Within Oracle E-Business Suite

Oracle Integration Repository supports various integration interface types categorized as follows:

- Service-enabled interface types

- PL/SQL

- XML Gateway Map (inbound)

- Concurrent Program

    > **Important:** Oracle Integration Repository supports REST service enablement for Open Interface Tables and Views. If a concurrent program is associated with an open interface table or view, this concurrent program can be viewed and displayed under the Open Interface type and can be available as a REST service.

- Open Interface Tables

- Open Interface Views

- Business Service Object (formerly known as Service Beans)

- Java Bean Services

    > **Note:** Java APIs whose methods use parameters of either simple data types or serializable Java Beans are categorized as Java Bean Services, a subtype of Java interface. Such Java APIs can be exposed as REST-based web services.

- Application Module Services

    > **Note:** Application Module Implementation class is a Java class that provides access to business logic governing the OA Framework-based components and pages. Such Java classes are called Application Module Services and are categorized as a subtype of Java interface.

    > Similar to Java Bean Services, Application Module Services can be exposed as REST services only.

- Security Services

    > **Note:** Unlike other service-enabled interfaces, security services are a set of predefined and predeployed REST services from Oracle Application Object Library. This type of services provides security related features for mobile applications.

- Subscription model

  - Business Event

  - XML Gateway Map (outbound)

- Composite services - BPEL

- Non-service enabled public interfaces

  - EDI Interface

  - Java APIs

    Please note that Java APIs, excluding the Application Module Services, Java Bean Services, and Security Services, cannot be exposed as web services.

Detailed information on each interface type, see Interface Types, page 3-16.

Major Features

- A unified repository from which all integration interface types are exposed.

- Any changes in interface definitions and descriptions are automatically reflected with release.

- A powerful user interface lets you easily browse and locate the interfaces by product family, interface type, or comprehensive search feature.

- It displays each interface details including source information, methods within the interface, and web service information if the interface can be service enabled.

- It supports composite services containing a collection of native interfaces.

- It enforces security rules to allow only authorized users to perform administrative tasks.

- It supports custom integration interfaces.

Getting Started with Oracle Integration Repository

Understanding Oracle E-Business Suite Integrated SOA Gateway User Roles

Oracle E-Business Suite Integrated SOA Gateway allows the following three roles to access the Integration Repository user interfaces and perform necessary tasks. Each user role is associated with a specific responsibility by default to access the Integration Repository.

- Integration Analyst - Integration Repository responsibility

- Integration Developer - Integrated SOA Gateway responsibility

- Integration Administrator - Integrated SOA Gateway responsibility

  Please note that the Integration Administrator role is assigned to the SYSADMIN user by default.

Users granted different roles can perform various tasks as described in the following table:

Privileges	Integration Analyst	Integration Developer	Integration Administrator
View Public Interfaces	Yes	Yes	Yes
View Private/Internal Interfaces	No	Yes	Yes
Generate/Regenerate SOAP Web Services	No	Yes	Yes
Deploy/Undeploy SOAP Web Services	No	No	Yes
Activate SOAP Web Services	No	No	Yes
Retire SOAP Web Services	No	No	Yes
Reset SOAP Web Services	No	No	Yes
Deploy/Undeploy REST Web Services	No	No	Yes
Subscribe to Business Events	No	No	Yes
Create Grants	No	No	Yes
View Grants	No	Yes	Yes
Delete Custom Interfaces	No	No	Yes
Download Composite Service	No (Configurable)	Yes	Yes

> **Note:** Oracle E-Business Suite Integrated SOA Gateway leverages the concepts of permissions and permission sets to grant data access privileges or permissions to users through roles. For example, multiple privileges related to administrative functions can be grouped into an administrative permission set and then granted to a user through the Integration Administrator role. That user becomes an integration administrator and has privileges to perform administrative tasks.
>
> Integration analysts by default do not have the privilege to download composite services unless they are granted the download privilege through a permission set. For more information on how to manage security through roles, see Role-Based Access Control (RBAC) Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

For information on SOAP and REST web services and how to perform additional administrative tasks, see Common Information on SOAP Web Services, page 4-6 and Common Information on REST Web Services, page 4-15.

For information on how to manage security grants, see Managing Grants, page 4-31.

Accessing Oracle Integration Repository

In this release, you can access Oracle Integration Repository through either of the following responsibilities depending on your assigned user role:

> **Note:** There are three predefined user roles used in the Oracle E-Business Suite Integrated SOA Gateway: Integration Analyst, Integration Developer, and Integration Administrator. For more information about these roles, see Understanding Oracle E-Business Suite Integrated SOA Gateway User Roles, page 3-3.

- **Integration Analyst Role:** *Integration Repository Responsibility > Integration Repository*

  Only users who are granted the Integration Analyst role can find the Integration Repository responsibility from the navigation menu.

  After selecting the Integration Repository responsibility and the Integration Repository link from the navigation menu, users who have the Integration Analyst role can navigate, search, and view existing integration interface definitions and services from the Integration Repository.

  Please note that when accessing the Integration Repository through this approach, these users do not have the privileges to perform any administrative tasks which are done by the users who have the Integration Administrator role through the Integrated SOA Gateway responsibility.

- **Integration Developer Role or Integration Administrator Role:** *Integrated SOA Gateway Responsibility > Integration Repository*

  Only users who are granted the Integration Developer role or the Integration Administrator role can find the Integrated SOA Gateway responsibility from the navigation menu.

  Similar to accessing the Integration Repository home page through the Integration Repository responsibility, users can search and view existing interface and service details through the repository. Additionally, users who are granted the Integration Administrator role have more privileges to generate and deploy web services and create security grants through the Integration Repository user interface.

  Furthermore, users who have the Integration Administrator role can find the **Administration** link in addition to the **Integration Repository** link from the navigator menu. The **Administration** link lets the administrators perform additional administrative tasks outside the Integration Repository user interface. See: Accessing the Administration Link to Perform Additional Administrative Tasks, page 3-6.

Accessing the Administration Link to Perform Additional Administrative Tasks:

After logging in to Oracle E-Business Suite with the Integrated SOA Gateway responsibility, an integration administrator can find the **Administration** link, in addition to the **Integration Repository** link from the Navigator menu.

This **Administration** link is specifically for the integration administrator to perform additional administrative tasks outside the Integration Repository user interface. Expand the **Administration** link to display the following links:

- **Service Monitor** link: This link lets the administrator access the Service Monitor user interface where the administrator can monitor and audit all REST messages received and sent directly to Oracle E-Business Suite and all SOAP messages passed through Oracle SOA Suite for Oracle E-Business Suite web services.

  For information about how to use Service Monitor, see Monitoring and Managing Inbound Service Invocation Messages Using Service Monitor, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Configuration** link: This link lets the administrator access the centralized Log and Audit configuration user interface where the administrator can configure log settings and auditing information at the service type (SOAP or REST) level of an integration interface.

  For information about log configuration, see Logging for Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Invocation Monitor** link: This link lets the administrator access the Service Invocation Monitor user interface where the administrator can monitor and audit both SOAP and REST request messages received and sent directly from Oracle E-Business Suite through Service Invocation Framework.

For information about how to use Service Invocation Monitor, see Monitoring and Managing Outbound Service Invocation Messages Using Service Invocation Monitor, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

For detailed information on each task performed by the users who have the Integration Administrator role, see *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*. For tasks related to the users who have the Integration Developer role, see *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.

Using Integration Repository

Integration Repository has the following major user interfaces:

- Browse Interface (by default), page 3-7

- Search Interface, page 3-10

- Interface Details Page, page 3-16

To further understand how to use Oracle Integration Repository, the following topics are included in this section:

- Included Interface Types, page 3-16

- Integration Standards, page 3-21

- Searching for a Specific Interface, page 3-10

Discovering and Reviewing Interfaces

This section includes the following topics:

- Browsing the Integration Interfaces, page 3-7

- Searching for an Integration Interface, page 3-10

- Interface Details Page, page 3-16

- Interface Types, page 3-16

- Integration Standards, page 3-21

Browsing the Integration Interfaces

The Browse interface appears by default while accessing the Oracle Integration Repository. You can also access the Browse interface by clicking **Browse** on the search page or any interface information page.

You can browse directly to an appropriate list of interfaces if you know which product family and product you want to integrate with, plus one of the following:

- **Business entity**

  Business entities are objects that either perform business activities or have business activities performed on them. Sales orders, employees, purchase orders, customers, and receipts are all examples of business entities. An interface can be used by multiple business entities, and a business entity can be accessed using multiple interfaces.

- **Interface type**

  Business interface information in Oracle Integration Repository is organized for browsing and searching by interface type, based on the integration technology used.

  For more information, see Interface Types, page 3-16.

- **Integration standard**

  XML Gateway and web service-based interfaces conform to various industry standards.

  For more information, see Integration Standards, page 3-21.

Instead of browsing through the integration endpoints from the repository, you can perform a search to locate your desired interfaces or services. See Searching for an Integration Interface, page 3-10.

Oracle Integration Repository Browse Page

Oracle Integration Repository allows you to browse integration interfaces or services by selecting one of the following views from the **View By** list:

- **Product Family**

- **Interface Type**

- **Standard** (integration standard)

Expand the navigation tree in one of these views to see a list of the available interfaces. To save the list of interfaces in a CSV file, click **Export**.

To review the details of an interface, click the interface name link to access the interface details page.

Browsing by Product Family

The **Product Family** view is organized as follows: **Product Family**, then **Product**, and then **Business Entity**

For example, `Financials`, then `Payables`, and then `Payables Invoice`.

After selecting a business entity (such as 'Payable Invoice'), click a desired integration interface name link from the Interface List table to view the interface details.

Please note that a business entity can include multiple interfaces of different types owned by different products. For example, the business entity 'Supplier Site' can include the following:

- Suppliers Package

- Supplier Sites Open Interface Import

Browsing by Interface Type

The **Interface Type** view is organized as follows: **Interface Type**, then **Product Family**, and then **Product**.

For example, PL/SQL, then Financials, and then Cash Management.

Use this view to see all of the interfaces available for a particular product that use a particular interface type.

Browsing by Standard

The **Standard** view is organized as follows: **Standard and Version**, then **Product Family**, and then **Product**.

For example, OAG7.2, then Financials, then Payables, and then Payables Invoice.

Use this view to browse for a product's XML Gateway maps and web services belonging to the specified standard, for example *W3C* or *OAG 7.2*.

Searching for an Integration Interface

Click **Search** in the Welcome to the Oracle Integration Repository page or any interface details page to access the main Search page.

Oracle Integration Repository Search Page

You can search for interfaces with any combination of the following criteria:

> **Note:** In the Search page the default value 'All' is automatically displayed in the Product Family, Product, and Interface Type fields. This allows a search to be processed appropriately if you do not make further changes from the drop-down selections.
>
> Additionally, the same default value 'All' can be found in the Category, Status, Interface Source, Web Service Status, Standard, and Scope fields while clicking the **Show More Search Options** link.

- **Interface Name**

  This is the interface name displayed in the browse tree of the Integration Repository user interface.

- **Internal Name**, page 4-3

  This is the interface name used internally. It can be PL/SQL package name, the document name, or the Java service interface name.

- **Interface Type**, page 3-16

  Interface definition can be categorized based on the integration technology used, such as PL/SQL or concurrent program related interfaces, when it is displayed or browsed in the repository.

  Interface types supported in Oracle Integration Repository are PL/SQL APIs, Concurrent Programs, XML Gateway messages, Open Interface Tables, Open

Interface Views, Business Service Objects, Business Events, EDI Interfaces, Java, and Composite Services - BPEL.

> **Note:** Java Bean Services, Application Module Services, Java APIs for Forms, and Security Services are a subtype of Java interface.

- **Product Family**

  An Oracle E-Business Suite application family that supplies the interface. Examples of product family can be Application Technology, Financials, and Manufacturing.

- **Product**

  An Oracle E-Business Suite application or component that supplies the interface. Examples of product can be Payables, Cash Management, and Order Management.

- **Business Entity**

  Business entities are objects that either perform business activities or have business activities performed on them. For example, sales orders, account numbers, employees, purchase orders, customers, and receipts are all business entities.

Click **Show More Search Options** to include any of the following additional criteria in your search:

- **Category and Category Value**

  Use these two fields to qualify product-specific features or to categorize a subtype of an interface. Select one of the following available category drop-down values if needed:

  - `All` (default) - This displays all integration interfaces regardless of category and category values.

  - `Extensions` - This category indicates that specific methods of extending the API functionality are provided by certain products. Examples of extensions are the User Hooks provided by Human Resource Management System, and Client Extensions provided by Projects.

    When `Extensions` category is selected, you can select the category value (for example, `HRMS User Hooks provided`).

  - `Interface Subtype` - This category indicates that a subtype of an interface is available.

    For example, 'Java Bean Services', 'Application Module Services', and 'Security Services' are a subtype of Java interface.

    When `Interface Subtype` category is selected, `Java Bean Services`, `Application Module Services`, `Java APIs for Forms`, and `Security`

`Services` are automatically displayed as the list of values for your selection.

- **Interface Source**

  Select one of the following values from the drop-down list:

  - `All` (default) - All integration interfaces will be displayed from the search.

  - `Oracle` - All Oracle native packaged integration interfaces and services are categorized by this interface source type.

  - `Custom` - This indicates annotated custom integration interfaces that are uploaded and published in the Integration Repository.

- **Status**

  Select one of the following values from the drop-down list:

  - `All` (default)

  - `Active`

  - `Deprecated`

  - `Obsolete`

  - `Planned`

  For more information, see Status, page 4-4 in the Common Information table.

- **Web Service Type**

  Use the Web Service Type field to search different types of services supported in Oracle E-Business Suite Integrated SOA Gateway.

  Select one of the following values from the drop-down list:

  - `All` (default) - This displays all interfaces regardless of the interface types whether they can be exposed as SOAP or REST services or not.

  - `REST` - This displays all interfaces that can be exposed as REST services.

  - `SOAP` - This displays all interfaces that can be exposed as SOAP services.

- **Web Service Status**

  Use the Web Service Status field to indicate different state of a web service during the service generation and deployment life cycle. This field is relevant for the interfaces that can be service enabled.

  The list of options displayed in the Web Service Status field depends on the value selected for Web Service Type.

- If `All` is selected as the service type, the available Web Service Status values for selection are `All` and `Deployed`.

- If `REST` is selected as the service type, the available Web Service Status values for selection are `All`, `Deployed`, and `Not Deployed`.

- If `SOAP` is selected as the service type, the available Web Service Status values for selection are `All`, `Generated`, `Not Generated`, `Deployed Active`, and `Deployed Inactive`.

Select one of the following values from the drop-down list:

- `All` (default) - Interfaces are displayed based on the following options:

  - When `All` is selected as the service type, this displays all interfaces regardless of the interface types whether they can be service enabled or not.

  - When `SOAP` is selected as the service type, this displays the interfaces that can be exposed as SOAP services, regardless of the service status.

  - When `REST` is selected as the service type, this displays the interfaces that can be exposed as REST services, regardless of the service status.

- `Not Generated` - When `SOAP` is selected as the service type, this displays all service-enabled interfaces that do not have SOAP services generated.

- `Generated` - When `SOAP` is selected as the service type, this displays all interfaces that have SOAP services generated, but not yet deployed.

- `Deployed Active` - When `SOAP` is selected as the service type, this displays all interfaces that have SOAP services deployed with 'Active' state. These interfaces are ready to be invoked and accept new SOAP requests.

- `Deployed Inactive` - When `SOAP` is selected as the service type, this displays all interfaces that have SOAP services deployed with 'Retire' state. These interfaces cannot accept new SOAP requests until they are activated.

- `Not Deployed` - When `REST` is selected as the service type, this displays all interfaces that do not have REST services deployed.

- `Deployed` - When `REST` is selected as the service type, this displays all interfaces that have REST services deployed.

  When `All` is selected as the service type, this displays all interfaces that have either REST or SOAP web services deployed, including active or inactive SOAP service deployment state.

For more information, see Common Information on Web Services, page 4-6.

- **Scope**

  Select one of the following values from the drop-down list:

  - `All` (default) - This displays all integration interfaces regardless of public, internal, or private interfaces.

  - `Public` - This displays public integration interfaces that can be used by anyone.

  - `Internal To Oracle` - This displays the interfaces that are available for business integration between applications within Oracle E-Business Suite.

    For example, if an interface of this type (Internal to Oracle) belongs to Application Object Library, then that interface can be used by any other applications within Oracle E-Business Suite for process integration in addition to using by the Application Object Library.

    Only users who have the Integration Developer role or the Integration Administrator role can access to this type of interfaces.

  - `Private To Application` - This displays the interfaces that are available for business integration only within the application itself. This type of interfaces can be used only by the applications that the interfaces belong to. For example, if an interface of this type belongs to Purchasing application, then it will not be used by any other applications within Oracle E-Business Suite but Purchasing.

    Only users who have the Integration Developer role or the Integration Administrator role can access to this type of interfaces.

- **Standard and Standard Specification**

  For more information, see Integration Standards, page 3-21.

Viewing the Search Results

After selecting your criteria, click **Go** to launch the search. A list of the available interfaces is displayed in the search result table. The interfaces listed in the table are paginated with a maximum of 10 records in one page. If there are more than 10 records retrieved from the search, click the **Next** icon to view the records listed in the next page.

The search result table lists each interface information including interface name, internal name, product, interface type, source, status, and description. Click the interface name link from the search result table to display the interface details page where you can view the details for the selected interface or service.

To have the search result displayed in your desired order, you can sort the entire set of the search result by clicking the **Up Arrow** or **Down Arrow** icon next to the field you want the result to be sorted against. Initially, the result will be sorted by Interface Name in ascending order. You can optionally change the sorting order by clicking on the **Down Arrow** icon next to the Name field to sort the result in descending order.

To save the list of interfaces to a CSV file, click **Export**. Click **Clear All** to clear all the search fields you entered and start a new search if desired.

Interface Details Page

Once an interface is selected from the browsing tree or the search result, the interface details page is displayed. It contains the following two types of information for a given interface:

- Common information

  This includes a header region with general information, full description of the selected interface, interface source information, as well as procedures and functions or methods contained in the selected interface.

- Web service information (optional)

  If the interface can be service enabled, the web service information is available in the interface details page. It contains interface or service information along with the service status indicating whether the selected interface has an associated service, or whether the service is deployed or not.

Integration administrators can perform administrative tasks for a selected interface in this page, such as generating and deploying a web service, subscribing to a business event, and creating security grants.

For more information about this interface details page, see Common Information in Interface Details, page 4-1.

Interface Types

Business interfaces are organized into *interface types* according to the integration technologies on which they're based.

Based on the natural way of how interfaces are formed or established, Oracle Integration Repository supports the following interface types:

- **Native Interfaces**

  Native integration interfaces or Oracle seeded interfaces are integration endpoints shipped with the Oracle Integration Repository by default. It includes the following interface types:

  - PL/SQL, page 3-17

  - XML Gateway, page 3-18

  - Concurrent Programs, page 3-18

  - Business Events, page 3-18

- Open Interface Tables, page 3-19

- Open Interface Views, page 3-19

- EDI, page 3-19

- Business Service Object (Service Beans), page 3-20

- Java, page 3-20

  Apart from normal Java APIs, Java interface includes the following subcategories:

  - Java Bean Services, page 3-20

  - Application Module Services, page 3-20

  - Security Services, page 3-21

  Please note that in this release Java APIs for Forms are not serviceable interfaces and cannot be exposed as SOAP services. Refer to My Oracle Support Knowledge Document 966982.1 for the suggested alternatives to the existing Java APIs for Forms interfaces.

- Composite Interfaces, page 3-21

  Building upon native interfaces, a composite interface consists of a collection of native interfaces that belong to a specific product or product family available in the Integration Repository.

  The supported composite interface type is Composite - BPEL.

  > **Note:** Oracle Integration Repository supports custom integration interfaces that are created and annotated based on Integration Repository annotation standards. After appropriate validation, these annotated custom source files can be uploaded and displayed along with Oracle interfaces through the Integration Repository browser tree, based on the interface types to which they belong.
  >
  > To easily differentiate custom interfaces from Oracle ones, all custom integration interfaces are categorized with interface source 'Custom' while Oracle interfaces are marked with interface source 'Oracle'. For more information about custom integration interfaces and services, see Working With Custom Integration Interfaces and Services, page 6-1.

PL/SQL Interface

A business interface can be based on a PL/SQL package from which you invoke

procedures and functions appropriate to an integration.

PL/SQL APIs are serviceable interfaces and can be exposed as both SOAP-based and REST-based web services.

For more information about PL/SQL interface type in the Integration Repository, see PL/SQL Information, page 4-40.

XML Gateway Message Map

Oracle XML Gateway comprises a set of services that allows easy integration with Oracle E-Business Suite to support XML messaging. Oracle E-Business Suite utilizes Oracle Workflow Business Event System to support event-based XML message creation and consumption.

Oracle XML Gateway consumes events raised by Oracle E-Business Suite and subscribes to inbound events for processing. It uses the message propagation feature of Oracle Advanced Queuing to integrate with Oracle Transport Agent to deliver messages to and receive messages from business partners. XML Gateway message maps can be used directly, or they can be exposed as web services.

> **Additional Information:** The message map is a file of type .xgm and is created using the XML Gateway Message Designer. Message maps define the data source and data target, any hierarchies between the source and the target, and actions for data transformation and process control.

For more information about XML Gateway map interface type in the Integration Repository, see XML Gateway Map Information, page 4-35.

For more information about Oracle XML Gateway, see the *Oracle XML Gateway User's Guide*.

Concurrent Program

A concurrent program runs as a concurrent process. That is, it processes multiple programs running in the background. Functions performed by concurrent programs are normally data-intensive and long-running, such as posting a journal, and generating an EDI flat file.

For more information about Concurrent Program interface type in the Integration Repository, see Concurrent Program Information, page 4-66.

For more information about how to use concurrent programs, refer to the *Oracle E-Business Suite Setup Guide*.

Business Event

A business event is an occurrence in an internet application that might be significant to other objects in a system or to external agents. An example of a business event can be

the creation of a new sales order or changes to an existing order.

Oracle Workflow uses the Business Event System that leverages the Oracle Advanced Queuing (AQ) infrastructure to communicate and manage business events between systems. When a local event occurs, the event subscribing code is processed in the same transaction as the code that raised the event. Subscription processing can include running custom code on the event information, sending event information to a workflow process, and sending event information to other queues or systems.

For more business event information, see Events, *Oracle Workflow Developer's Guide*.

For more information about business event interface type in the Integration Repository, see Business Event Information, page 4-76.

Open Interface Table

An open interface consists of the interface tables to store data from external sources and concurrent programs, as well as to validate and apply this data into the Oracle E-Business Suite base tables. All open interfaces are implemented using concurrent programs.

Please note that Open Interface Table is often referred as Open Interface.

For more information about Open Interface Table interface type in the Integration Repository, see Open Interface Information, page 4-69.

Open Interface View

Interface views are database objects that make data from Oracle E-Business Suite products available for selection.

For more information about Interface View interface type in the Integration Repository, see Interface View Information, page 4-73.

EDI Message Transaction

Electronic Data Interchange (EDI) is one form of electronic commerce. Interface data files are electronically exchanged between trading partners as messages in a standard format to minimize manual effort. EDI message transactions are supported by *Oracle e-Commerce Gateway.*

Oracle e-Commerce Gateway provides users the ability to conduct business electronically between trading partners based on Electronic Commerce standards and methodology. When used for EDI solutions, Oracle e-Commerce Gateway integrates with EDI translators to provide specific EDI standard formats and versions.

For more information about EDI messages in the Integration Repository, see EDI Message Information, page 4-75.

For more information about Oracle e-Commerce Gateway, see the *Oracle e-Commerce Gateway User's Guide*.

Business Service Object

A business service object, formerly known as Service Bean, is a high-level service component that allows OA Framework or BC4J components to be deployed as web services.

Business service object interfaces provide access to SOA services to facilitate integration between Oracle E-Business Suite and trading partners. They often employ *service data objects* as parameters to pass complex data.

> **Note:** A service data object is not actually an interface type; rather, it is an object used by one or more business service objects or other service data objects to pass data.

For more information about Business Service Object interface type in the Integration Repository, see Business Service Object, page 4-55.

Java

A business interface can be based on a Java class from which you invoke methods that are appropriate to an integration.

Java Bean Services

Java Bean Services are also a subtype of Java interface. This type of Java APIs whose methods must use parameters of either serializable Java Beans or simple data types such as `String`, `Int`, and so forth can be categorized as Java Bean Services. Such Java APIs can be exposed as REST services only.

Similar to the PL/SQL REST services, Java Bean Services have simplified development life cycle - Deploy and Undeploy - and are implemented with the same security mechanism. Java Bean Services can be deployed as REST service operations with the POST and GET HTTP methods in this release.

For annotation guidelines on Java Bean Services, see Annotations for Java Bean Services, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.

Application Module Services

Application Module Implementation class is a Java class that provides access to business logic governing the OA Framework-based components and pages. Such Java classes are called Application Module Services and are categorized as a subtype of Java interface.

Similar to Java Bean Services, Application Module Services can be exposed as REST services only.

For annotation guidelines on Application Module Services, see Annotations for Application Module Services, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.

Security Services

Security services, built on Java, are a set of predefined and predeployed REST services from Oracle Application Object Library. These services including Authentication and Authorization services are developed for mobile applications.

Security services will not require any REST service lifecycle activities such as Deploy or Undeploy. Additionally, they are available to all users.

For more information about Java interface type in the Integration Repository, see Java Information, page 4-44.

Java APIs for Forms interfaces are not serviceable interfaces and cannot be exposed as SOAP services. Refer to My Oracle Support Knowledge Document 966982.1 for the suggested alternatives to the existing Java APIs for Forms interfaces.

Composite Interfaces

A composite interface consists of a collection of native packaged interfaces or services available in the Integration Repository.

Composite interfaces orchestrate the invocation sequence of web services into an end-to-end business process through a web service composition language BPEL (business process execution language).

> **Additional Information:** Composite interfaces can be designed and created in Oracle JDeveloper and Oracle Eclipse. Based on the creation methods, composite services have various composite types such as BPEL, ESB (enterprise service bus), or SCA (service component architecture) types. Composite - BPEL type is the only supported composite interface in this release.

For more information about composite interfaces, see Working with Composite Interfaces, page 5-1.

Integration Standards

Each web service interface conforms to an integration standard, for example OAGIS or RosettaNet. The fully qualified standard includes the name, version, and specification. For example: OAG 7.2 CONFIRMBOD_004. The following standards are observed in Oracle Integration Repository:

- IFX1.2

- OAG6.2

- OAG7.0

- OAG7.1

- OAG7.2

- RosettaNet01.01.00

- RosettaNet01.03.00

- RosettaNet02.02.00

- RosettaNet02.03.00

- UCCnet2.4

- W3C

4

Working with Native Integration Interfaces

This chapter covers the following topics:

- Common Information
- XML Gateway Map Information
- PL/SQL Information
- Java Information
- Business Service Object
- Concurrent Program Information
- Open Interface Information
- Interface View Information
- EDI Message Information
- Business Event Information

Common Information

Each interface details page contains the following two types of information:

- **Interface Detail Information**

  Interface detail information includes a header region with general information, full description of the selected interface, interface source information, as well as methods or procedures and functions contained in the selected interface.

  For more information on interface details, see Common Information on Interface Details, page 4-3.

- **Web Service Information**

  Based on the selected interface, you can view the associated service information if it's available in the interface details page. The service information includes service

status, serviceable operations, service description, and supported service methods for the selected interface.

- For an XML Gateway interface, the associated SOAP service information is shown in the Web Service region of the interface details page.

- For other interfaces that can be exposed as SOAP services, REST services, or both SOAP and REST services, the associated service information is shown in the SOAP Web Service tab, REST Web Service tab, or both tabs.

> **Note:** In this release, only PL/SQL APIs, Concurrent Programs, and Business Service Object interfaces can be exposed as both SOAP and REST services. Java Bean Services, Application Module Services, Open Interface Tables, and Open Interface Views can be exposed as REST services only.

For information on SOAP-based services, see Common Information on SOAP Web Services, page 4-6.

For information on REST-based services, see Common Information on REST Web Services, page 4-15.

> **Important:** For interfaces that can be exposed as SOAP services, if the setup tasks for SOAP services are not performed, when viewing these interfaces through the Integration Repository, you may find a message indicating that Oracle E-Business Suite Integrated SOA Gateway is not configured for SOAP services and refer to My Oracle Support Knowledge Document 1311068.1 for configuration details.

Additionally, the Log Configuration field is displayed with a value (either 'Disabled' or 'Enabled') indicating whether the selected interface has design-time log configured or not for a specific service type (SOAP or REST). By default, 'Disabled' is displayed. Users who have the Integration Administrator role can enable or disable the design-time log for the selected service type (SOAP or REST) of an interface by clicking **Configure**. The Log & Audit Setup Details page is displayed where the administrator can add a new log configuration or update an existing configuration. For more information on log configurations, see Logging for Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

Each interface details page also includes **Search** and **Printable Page** allowing you to perform a search and view the details page of a selected interface in a printable format. See Searching for an Integration Interface, page 3-10.

Common Information on Interface Details

When you click an interface name link from the Interface List page or the Search Interface page, the interface details page appears where you can view the selected interface information.

For information on SOAP and REST web services, see Common Information on SOAP Web Services, page 4-6 and Common Information on REST Web Services, page 4-15.

Common Information in the Interface Details Page

The following fields are common to almost all interface types:

Field	Notes
Internal Name	This is the PL/SQL package name, the document name, or the Java service interface name. **Note:** For Java service interfaces, this is called **Qualified Name,** and includes the full Java package name and the class name.

Field	Notes
Type	Business interfaces are organized into interface types according to the integration technologies on which they're based. Examples of interface types supported in Integration Repository are PL/SQL, XML Gateway, Concurrent Programs, Business Events, Open Interface Tables, Open Interface Views, EDI, Business Service Object (Service Beans), Java, and Composite Service - BPEL. For more information about interface type, see interface type, page 3-16.
Product	An Oracle E-Business Suite application or component that supplies the interface.
Business Entity	Business entities are objects that either perform business activities or have business activities performed on them. For example, sales orders, account numbers, employees, purchase orders, customers, and receipts are all business entities. An interface can be used by multiple business entities, and a business entity can be accessed by multiple interfaces. The Business Entity field lists the business entities accessed by an interface. Click a business entity name to view a list of available interfaces to that entity.
Status	Valid status codes are: • `Active` • `Deprecated` - This interface should not be used, but it will be supported until obsolete. • `Obsolete` - The interface is no longer supported. • `Planned` - This interface will be activated at a future date.
Scope	The scope can be one of the following: • `Public` • `Internal To Oracle` • `Private To Application` For more information, see **Scope** on the Oracle Integration Repository Search page, page 3-15.

Field	Notes
Interface Source	The interface source can be one of the following: • `Oracle`- All Oracle seeded integration interfaces are categorized with this interface source type. • `Custom`- This indicates the interface is a custom integration interface or service.
My Oracle Support	Included for an interface that has a related My Oracle Support Knowledge Document. Click the link to log on to My Oracle Support with a valid user name and password and view the Knowledge Document.
Documentation	Included for an interface that has related online documentation. Click the link to view or download the documentation.
Online Help	Provided for an interface that has related Oracle E-Business Suite online help. Click the link to view online help for the interface.

The following table describes each field contained in the Source Information region:

Field	Notes
Source File	It indicates the source code file for this interface, and its location in the file system.
Source Version	It indicates the version of the source file. The first portion of the number corresponds to the base release version of Oracle E-Business Suite; the second portion is the version of the file. For example, `120.8` is Oracle E-Business Suite 12.0, and `8` indicates that this is the 8th version of the file. **Note:** The version number changes only when it has been worked on by Oracle development. Therefore the version may increment multiple times between releases, or not at all.

Field	Notes
Source Product	It represents the product code of the source product. The source product specifies under which product directory the file resides in the Oracle E-Business Suite file system (also referred to as the *product top*). **Tip:** This field shows the product short name. You can learn the corresponding full product name by choosing the **System Administration** responsibility from the Navigator menu, then selecting **Oracle Applications Manager** >**License Manager** >**Reports** >**Licensed Products**. On the product list that appears, you can filter the results for any product abbreviation (short name) or license status.

Common Information on SOAP Web Services

For service-enabled interfaces, SOAP service information is displayed in the SOAP Web Service tab (or the Web Service region for XML Gateway interface only) no matter if the selected interface is currently exposed as a SOAP service or not.

> **Note:** Web service is defined in a way that the interface forms the service and the methods or functions within the interface are defined as the operations of the service.
>
> By default, an interface is not exposed as a SOAP service. Hence, the service status is 'Not Generated'. An integration administrator may transform the selected interface into a SOAP service. Once the service for the selected interface has been successfully generated, the service status will be changed from 'Not Generated' to 'Generated'.

Example of an Interface Details Page with SOAP Web Service Tab

The following fields are common in the SOAP Web Service tab or the Web Service region to almost all interface types:

Field	Description

Web Service Status
 	
(or SOAP Service Status if it's in the SOAP Web Service tab)
	This field indicates different state of a SOAP service during service generation and deployment life cycle.
	
	• Not Generated: This indicates that the selected interface does not have a SOAP service generated.
	
	• Generated: This indicates that the selected interface has a SOAP service available, but the service has not yet been deployed. Therefore, this generated service is not ready to be invoked. Once a service has been successfully generated, the following buttons appear:
	
	  • **Regenerate**: This button lets you regenerate the service if the interface definition has been changed or the selected interaction pattern information has been modified before service deployment.
	
	  • **Deploy**: This button lets you deploy the generated SOAP service with active deployment state.
	
	  • **Reset**: This button lets you clear up the existing service artifact and change the Web Service Status field from 'Generated' to 'Not Generated'.
	
	• Deployed: This indicates that the selected interface has been deployed to Oracle SOA Suite. A deployed service can have either one of the following states:
	
	  • **Deployed with 'Active' state:** By default, the deployed service is in active state ('Deployed' with 'Active' state). This indicates that the deployed service is ready to be invoked and accept new SOAP requests. Once a service has been successfully deployed with 'Active' state, the following buttons appear:
	
	    • **Retire**: This button lets you disable the active service. The SOAP Service Status field is changed to **Deployed with 'Retired'** state indicating that this deployed service will no longer accept new requests.
	
	    This action also ensures that current running requests for the service are finished. Once a service has been retired ('Deployed' with 'Retired' state), the service is no longer to be invoked.
	
	    • **Undeploy**: This button lets you undeploy the SOAP service from Oracle SOA Suite back to Oracle Integration Repository if necessary.

Field	Description

- **Reset**: This button lets you clear up the existing service artifact and change the SOAP Service Status field from 'Deployed' with 'Active' to 'Not Generated'.

- **Deployed with 'Retired' state:** Once a service has been retired ('Deployed' with 'Retired' state), the service is no longer to be invoked. It cannot accept new SOAP requests until it is activated ('Deployed' with 'Active' state). The following buttons appear for the service with 'Deployed' with 'Retired' state:

  - **Activate**: This button lets you change the retired service back to an active service again.

  - **Undeploy**: This button lets you undeploy the retired service from an Oracle SOA Suite managed server to the repository.

  - **Reset**: This button lets you reset the retired service to its initial state - 'Not Generated' if needed.

In addition to 'Not Generated', 'Generated', 'Deployed' with 'Active' state, and 'Deployed' with 'Retired' state, more intermediate SOAP service statuses can be shown while the service is in the process of performing an action issued by the administrator and transforming to a different state. The following list describes intermediate Web Service Status information:

- Generating: This indicates that the selected interface is in the middle of the process of transforming the interface definition into a SOAP service. After this process completes successfully, 'Generated' is displayed in the Web Service Status field.

- Regenerating: This indicates that the selected interface is in the middle of the process of regenerating the service. After this process is complete, 'Generated' is displayed in the Web Service Status field.

- Deploying: This indicates that the selected interface is in the middle of the process of deploying the service. After this process is complete, 'Deployed' with 'Active' state is displayed in the Web Service Status field.

- Undeploying: This indicates that the selected interface is in the middle of the process of undeploying the service. After this process is complete, 'Generated' state is displayed in the Web Service Status field.

- Resetting: This indicates that the selected interface is in the middle of the process of removing the associated service artifact. After this process is

Field	Description
	complete, 'Not Generated' is displayed in the Web Service Status field.
	• Retiring: This indicates that the selected interface is in the middle of the process of disabling the active state. After this process is complete, 'Deployed' with 'Retired' state is displayed in the Web Service Status field indicating that the service is retired and cannot accept new SOAP requests.
	• Activating: This indicates that the selected interface is in the middle of the process of reverting the retired state back to active again. After this process is complete, 'Deployed' with 'Active' state is displayed in the Web Service Status field indicating that the deployed service is ready to accept new SOAP requests.
View WSDL	This link is displayed after a selected interface has an associated SOAP service available. Click this link letting you review WSDL description for a generated or deployed service. See: Reviewing Web Service WSDL Source, page 4-23.
Service Operations table	This table displays the interface name along with the method names contained in the interface in a table. An integration administrator or an integration developer must select appropriate interaction patterns either at the interface level or at the method level before service generation. After service generation, the selected interaction patterns are displayed for the interface or for specific methods contained in the interface. Click the interface name node to expand and view the interaction pattern selection for all the methods within the interface. Note that this table is still updatable after service generation, but any changes to the table will be applied only after regenerating the service. For more information on each field in the table, see Service Operations Table, page 4-11.

Field	Description
Interaction Pattern	After service generation, this information is displayed which corresponds to the selected interaction patterns in the Service Operations table. For example, if 'Synchronous' is selected for a specific method contained in a PL/SQL interface, and 'Asynchronous' is selected for another method within the interface, then both 'Synchronous' and 'Asynchronous' are shown in this field. **Note:** XML Gateway, Concurrent Program, and Business Service Object interfaces can be service enabled only with synchronous support. Therefore, 'Synchronous' is displayed in the Interaction Pattern field for those interfaces by default if the service is available.
Authentication Type	To secure web service content and authenticate web service operation, before deploying a generated service, an integration administrator must select one desired authentication type for the selected service in the Authentication Type field. • Username Token: This authentication type provides a user name and corresponding password in the security header to authenticate SOAP requests. See: UsernameToken Based Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*. • SAML Token (Sender Vouches): This authentication type is used for SOAP services relying on sending a user name only through SAML Assertion. See: SAML Sender-Vouches Token Based Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*. Only users who have the Integration Administrator role can select and modify the authentication type. For other users, this field is displayed in read-only mode.

Service Operations Table

Oracle E-Business Suite Integrated SOA Gateway supports service generation synchronously, asynchronously, or both synchronously and asynchronously based on the selection made by the administrator or the integration developer in the Service Operations table.

Interaction Pattern Selection

Before service generation, the administrator or the integration developer must select appropriate interaction patterns either at the interface level or at the method level by expanding the interface name node to list all the methods within the interface.

After service generation, the Service Operations table is still updatable, but any changes will be applied only after service regeneration.

> **Note:** In this release, asynchronous SOAP service operation is supported only in PL/SQL interface. Other SOAP service-enabled interface types including XML Gateway, Concurrent Program, and Business Service Object can be exposed as SOAP services with the synchronous support only.

The following table describes each field in the Service Operations table:

Field	Description
Display Name	This is the interface name used externally.
Internal Name	This is the interface name used internally. It can be PL/SQL package name, the document name, or the Java service interface name.
Synchronous	If the 'Synchronous' checkbox is selected for an interface or a specific method, the service or selected operation is generated with the support for synchronous interaction pattern.
	That is when a web service client sends a SOAP request for this service, service is invoked and provides an immediate response to the web service client.
Asynchronous	If the 'Asynchronous' checkbox is selected for an interface or a specific method, the service or selected operation is generated with the support for asynchronous interaction pattern. This type of service invocation may require a significant amount of time to process a request. However, the client that invoked the Oracle E-Business Suite service can continue with other processing in the meantime rather than waiting for the response.
	In this release, asynchronous operation is supported in PL/SQL interface only.
Grant	If the access permission of an operation has been granted to a specific user, user groups, or all users, then the **Grant** icon is available for the operation. Only users who have the Integration Administrator role and the Integration Developer role can find the **Grant** icon and view the grant details.
	The **Grant** icon is shown only in the SOAP Web Service tab.

Performing Administrative Activities for SOAP Web Services

Users who have the Integration Administrator role can perform administrative tasks in the SOAP Web Service tab of an interface that can be exposed as a SOAP service. These tasks include generating, deploying, undeploying, resetting, retiring, and activating

SOAP services by clicking the following buttons in the interface details page:

SOAP Web Service Tab for Performing Administrative Tasks

- **Generate**: This allows an integration administrator or an integration developer to generate a SOAP service.

  > **Note:** Integration developers have the privilege to generate the services, but they do not have privileges to perform other administrative tasks including deploying, undeploying, and managing SOAP services throughout the life cycle.
  >
  > For information about these roles and their associated privileges to administer and view SOAP services, see Accessing Oracle Integration Repository, page 3-5.

  For more service generation information, see Generating SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Deploy**: If the SOAP service has already been successfully generated, the administrator can click **Deploy** in the SOAP Web Service tab (or the Web Service region for XML Gateway interface) to deploy the generated service to Oracle SOA Suite with 'Active' state.

  For more service deployment information, see Deploying and Undeploying SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Retire**: This disables an active deployed service so that the retired service will not accept new requests. Once an active service has been successfully retired, the administrator can activate, undeploy, or reset the retired service if it's needed.

For more information on retiring services, see Retiring SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Activate**: For a retired service in Oracle SOA Suite, the administrator can activate the service. The service can be invoked and accept new SOAP requests again.

  For more information on activating services, see Activating SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Reset**: Once a service has been successfully generated or deployed, the administrator can clear up existing service artifact at any time by clicking **Reset**.

  For more information on resetting services, see Resetting SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Create Grant** (Grants tab): The administrator can create security grants by authorizing access permissions of interface methods to a user, a user group, or all users.

  The grant feature applies to both SOAP and REST service operations if the selected interface can be exposed as both SOAP and REST services. For more information on security grants, see Managing Grants, page 4-31.

- **Configure**: This displays the Log & Audit Setup Details page where the administrator can update an existing configuration or configure a new design-time log for the SOAP service type of the selected interface.

  For more information on log configurations, see Accessing the Logging Configuration User Interface, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **View Log**: This displays the Log & Error Details page where you can view log details.

  For more information on viewing logs recorded at design time, see Viewing Design-Time Logs for SOAP Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  To view logs written in Service Monitor during the invocation of Oracle E-Business Suite services, see Viewing Service Processing Logs, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  > **Additional Information:** In addition to performing service management activities in the Integration Repository tab, integration administrators can perform other administrative tasks in the Administration tab that are outside the Integration Repository user interface. These tasks include:
  >
  > - **Monitoring and Auditing Inbound Service Invocations to Oracle**

E-Business Suite in the Service Monitor Subtab

The Service Monitor subtab allows the administrators to monitor and audit all SOAP messages received and sent from Oracle SOA Suite and REST messages recorded and sent directly to Oracle E-Business Suite using Service Monitor. See: Monitoring and Managing Inbound Service Invocation Messages Using Service Monitor, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Managing Log and Audit Setups in the Configuration Subtab**

  The Configuration subtab allows the administrators to configure log settings and enable service auditing feature at the service type (SOAP or REST) level of an interface. The administrators can easily monitor service activities, track and view log messages, and troubleshoot any issues encountered at each stage of SOAP and REST service development life cycle. See: Logging for Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Monitoring and Auditing Outbound Service Invocations through Service Invocation Framework in the Invocation Monitor Subtab**

  Similar to the service monitoring and auditing features in Service Monitor, the Invocation Monitor subtab allows the administrators to monitor and audit all outbound SOAP and REST service invocations provided through Service Invocation Framework and their associated request and response messages using Service Invocation Monitor. See: Monitoring and Managing Outbound Service Invocation Messages Using Service Invocation Monitor, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

Common Information on REST Web Services

If a selected interface can be exposed as a REST service, you can find the REST Web Service tab included in the interface details page. This interface can be an interface type of PL/SQL, Concurrent Program, Open Interface Table, Open Interface View, Business Service Object, Java Bean Services, or Application Module Services.

Example of an Interface Details Page with REST Web Service Tab

The following fields are common in the REST Web Service tab:

Field	Description
Service Alias	Each REST service should be associated with a unique alias name. Before deploying a REST service, the administrator must enter this field which will be used in service endpoint, WADL, XSDs, and namespaces. Please note the following guidelines when specifying the service alias: • Use simple and meaningful name to represent the service, such as "person", "employee", and so on. • Do not use "rest", "soap", and "webservices" as the alias. • Do not start with a number or a special character, such as #, $, %, _, – and more. • Do not end with a special character. • Characters such as ., _, and – are allowed in a service alias.

Field	Description
REST Service Status	This field indicates different state of a REST service during service life cycle. • Not Deployed: This indicates that the selected interface is not deployed. Click **Deploy** to deploy the REST service. The REST Web Service Status is changed from 'Not Deployed' to 'Deployed'. • Deployed: This indicates that the selected interface is deployed to an Oracle E-Business Suite managed server. Once a REST service has been successfully deployed, the administrator can undeploy the service by clicking **Undeploy**. This action undeploys the REST service from the Oracle E-Business Suite managed server back to Oracle Integration Repository, and at the same time it clears up the existing service artifact. The REST Service Status field is changed from 'Deployed' to 'Not Deployed'. In addition to 'Not Deployed' and 'Deployed' service states, more intermediate service statuses can be shown while the service is in the process of performing an action issued by the administrator and transforming to a different state. The following list describes intermediate service status information: • Deploying: This indicates that the selected interface is in the middle of the process of deploying the service. After this process is complete, 'Deployed' is displayed in the REST Service Status field. • Undeploying: This indicates that the selected interface is in the middle of the process of undeploying and removing the associated service artifact. After this process is complete, 'Not Deployed' is displayed in the REST Service Status field.
View WADL	This link is displayed after the selected interface has an associated REST service deployed. Click this link letting you review WADL description for the deployed REST service. This field appears only when the REST service has been successfully deployed with 'Deployed' status. See: Reviewing Web Service WADL Source, page 4-27.

Field	Description
Verb	The Verb value indicates how the REST service is implemented using an HTTP method.
	For Concurrent Programs, if a selected interface is deployed (with 'Deployed' status) as a REST service, this field appears along with the "POST" HTTP method.
	Note: POST is the only supported HTTP method for Concurrent Programs.
	For PL/SQL APIs, Business Service Objects, Java Bean Services, Application Module Services, Open Interface Tables, and Open Interface Views, the supported verbs are displayed in the Service Operations Table instead.
REST Service Security	To secure REST service content, all REST services are secured by either one of the following security methods:
	• HTTP Basic Authentication: This authentication type is for an HTTP client application to provide a user name and corresponding password when making a REST request that is typically over HTTPS. See: HTTP Basic Authentication, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.
	• Security Token Authentication: This token-based security authenticates a user using a security token obtained by invoking the security Login service. When a user tries to log on to a server, a token (such as Oracle E-Business Suite session ID) may be sent as `cookie` in the HTTP header. This authentication method can be used in multiple consecutive REST invocations. See: Token Based Authentication, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.
	Before deploying an interface as a REST service, an integration administrator must ensure that at least one authentication type is selected for use in authenticating users who invoke the REST services.
Service Operations table	This table displays the list of procedures or functions contained in the selected interface that can be exposed as REST service operations.
	For more information on each field in the table, see Service Operations Table, page 4-18.

Service Operations Table

The Service Operations table displays each method (or procedure or function) contained in the selected interface, and whether it is exposed as a service operation. Users who have the Integration Administrator role can perform administrative tasks including deploying or undeploying services with desired service operations as well as creating or

revoking security grants.

The following table describes each field in the Service Operations table:

Field	Description
Display Name	This is the interface name used externally.
Internal Name	This is the interface name used internally.
Included Operations (Concurrent Programs Only)	This column appears when a selected Concurrent Program is exposed as a REST service. A "check" icon is shown in this column for the method or operation that is exposed as a REST service operation. **Note:** Each concurrent program contains only one method. Once it is deployed as a REST service, the method contained in the concurrent program will be automatically marked with the "check" icon by default and deployed with the POST HTTP method only.
GET (PL/SQL, Business Service Object, Java Bean Services, Application Module Services, and Open Interface Tables and Views Only)	This GET method column appears when a selected interface is an interface type of PL/SQL, Business Service Object, Java Bean Services, Application Module Services, Open Interface Table, or Open Interface View. For PL/SQL APIs, the GET checkbox is enabled only if the selected API is a simple data type. It is disabled if it's a complex data type. For Business Service Object interfaces, the GET checkbox is enabled only if input parameters are of simple data types (String, Number, etc.). It is disabled if input parameters consist of complex data object types (AccountMergeRequest, etc.). For Java Bean Services and Application Module Services, this GET checkbox is preselected if a Java or an Application Module method is annotated with the GET HTTP method. The administrator can uncheck the preselected GET checkbox for the Java or Application Module method if it will not be published with the GET method. However, if it is not annotated with the GET method, unlike the POST method, the GET checkbox becomes inactive or disabled for further selection. For Open Interface Tables with Inbound direction, four HTTP methods (GET, POST, PUT, and DELETE) are all displayed. For Open Interface Tables with Outbound direction and Open Interface Views, only the GET method is displayed in the table.

Field	Description
POST (PL/SQL, Business Service Object, Java Bean Services, Application Module Services, and Open Interface Tables with Inbound direction Only)	This POST method column appears when a selected interface is an interface type of PL/SQL, Business Service Object, Java Bean Services, Application Module Services, or Open Interface Table with Inbound direction. For PL/SQL APIs and Business Service Object interfaces, select the POST checkbox if desired for a method to be deployed as a REST service operation. For Java Bean Services and Application Module Services, if a Java or an Application Module method is annotated with the POST method, similar to the GET method, this POST checkbox is preselected. The administrator can uncheck the preselected checkbox before deploying the service if the Java or Application Module method will not be published with the POST method. If it is not annotated with the POST method, unlike the GET method, the POST checkbox remains active or enabled by default. The administrator can still select the POST checkbox if needed for a method. For Open Interface Table with Inbound direction, select the POST checkbox if desired for an interface table to be deployed as a REST service operation.
PUT (Open Interface Tables with Inbound direction Only)	This HTTP method column appears only when the selected interface is an Open Interface Table with Inbound direction.
DELETE (Open Interface Tables with Inbound direction Only)	This HTTP method column appears only when the selected interface is an Open Interface Table with Inbound direction.
Grant	If the access permission of an operation has been granted to a specific user, user groups, or all users, then the **Grant** icon appears for the operation. Only users who have the Integration Administrator role and the Integration Developer role can find the **Grant** icon and view the grant details.

Performing Administrative Activities for REST Web Services

REST services have a simplified service development life cycle. Users who have the Integration Administrator role can perform the following administrative tasks in the REST Web Service tab of an interface that can be exposed as a REST service:

> **Note:** For information about different user roles and their associated

privileges to administer and view REST services, see Accessing Oracle
Integration Repository, page 3-5.

REST Web Service Tab for Performing Administrative Tasks

Business Service Object : Configuration Information Service

Qualified Name	/oracle/apps/cz/dataservices/ConfigurationInformationService	Status	Active
Interface	oracle.apps.cz.dataservices.ConfigurationInformationService	Scope	Public
Extends	oracle.svc.Service	Interface Source	Oracle
Product	Configurator		

Overview   SOAP Web Service   **REST Web Service**   Grants

* Service Alias   configure   Log Configuration   Enabled   [Configure]   [View Log]
REST Service Status   Not Deployed

Service Operations

• • •

Name	Internal Name	GET	POST	Grant
Configuration Information Service		☐	☐	
get Ui For Item	getUiForItem	☑	☑	

☑TIP To apply any changes in Operation, Undeploy the service.

REST Service Security

Personalize "REST Service Security"

*Authentication Type   ☑ HTTP Basic
                         ☑ Security Token

Tip: Use Login Service to obtain Security Token for given user credentials.

[Deploy]

[Browse]   [Search]   [Printable Page]

- **Deploy**: This allows an integration administrator to deploy the REST service.

  Before deploying the service, an integration administrator must enter a unique
  name in the Service Alias field, select desired service operations in the Service
  Operations table, and ensure that at least one authentication type is selected in the
  REST Service Security region. If the selected interface is an interface type of PL/SQL,
  Business Service Object, Java Bean Services, or Application Module Services, choose
  the desired HTTP method checkboxes for the methods to be exposed as REST
  service operations.

  Once the REST service has been successfully deployed, the service status is changed
  to 'Deployed'. This indicates that this deployed service is ready to be invoked and
  accept new service requests.

  For more REST service deployment information, see Deploying REST Web Services,

Oracle E-Business Suite Integrated SOA Gateway Implementation Guide.

- **Undeploy**: This action not only undeploys the service from an Oracle E-Business Suite managed server to the Integration Repository, but also resets its status to the initial state 'Not Deployed'.

  For more REST service undeployment information, see Undeploying REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Create Grant** (Grants tab): The administrator selects desired service operations before clicking **Create Grant**. The Create Grants page appears where grants can be created for a user, a user group, or all users.

  The administrator selects desired service operations and clicks **Revoke Grant**. This opens the Revoke Grants page where the administrator can remove existing grants from an authorized user, a user group, or all users.

  The grant feature applies to both SOAP and REST services. For more information on security grants, see Managing Grants, page 4-31.

- **Configure**: This displays the Log & Audit Setup Details page where the administrator can update an existing configuration or configure a new design-time log for the REST service type of the selected interface.

  For more information on log configurations, see Accessing the Logging Configuration User Interface, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **View Log**: This displays the Log & Error Details page where you can view log details recorded at the design time for the selected REST service. See Viewing Design-Time Logs for REST Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  To view logs written in Service Monitor during the invocation of Oracle E-Business Suite services, see Viewing Service Processing Logs, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  > **Additional Information:** Integration administrators can perform additional administrative tasks in the Administration tab to manage log settings, monitor inbound service invocation messages recorded through Service Monitor, and monitor outbound service invocations from Oracle E-Business Suite through Service Invocation Monitor. See:
  >
  > - Logging for Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*
  >
  > - Monitoring and Managing Inbound Service Invocation Messages Using Service Monitor, *Oracle E-Business Suite Integrated SOA*

Gateway Implementation Guide

- Monitoring and Managing Outbound Service Invocation Messages Using Service Invocation Monitor, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*

Reviewing Web Service WSDL Sources

Once a SOAP service represented in WSDL has been successfully generated, the SOAP service status is changed from 'Not Generated' to 'Generated'. The WSDL link appears in the SOAP Web Service tab (or the Web Service region for XML Gateway interface) allowing you to view the WSDL description.

Generated WSDL and Deployed WSDL Descriptions

For a generated service before it is deployed, the WSDL description is an abstract definition of the message that has been transmitted. Once the service has been successfully deployed, the WSDL description contains concrete service binding and transport details.

For example, the following elements contain different URL information in the generated and deployed WSDL descriptions:

- Message Schema Location (`schemaLocation`)

  - For a generated service, a temporary schema location is displayed in the generated WSDL description (such as `schemaLocation="http://<hostname>:<port>/ISG-ISG-context-root/isgapp/plsql/pa_cost_plus/APPS_PA_COST_PLUS_GET_BURDEN_AMOUNT.xsd"`).

  - For a deployed service, a physical location of the service endpoint where the service is hosted in `soa-infra` is displayed instead in the deployed WSDL description (such as `schemaLocation="http://<soa_suite_hostname>:<port>/soa-infra/services/default/<jndi_name>_PLSQL_PA_COST_PLUS/PA_COST_PLUS_Service?XSD=xsd/APPS_PA_COST_PLUS_GET_BURDEN_AMOUNT.xsd"`)

- Service Address Location (`soap:address location`)

  - For a generated WSDL, 'Not_Deployed' is shown in the `soap:address location` element (such as `<soap:address location="#NOT_DEPLOYED#"/>`).

  - For a deployed service, a physical location of the service endpoint where the service is hosted in `soa-infra` is displayed in the deployed WSDL description (such as `<soap:address location="http://<soa_suite_hostname>:`

```
<port>/soa-
infra/services/default/<jndi_name>_PLSQL_PA_COST_PLUS/PA_CO
ST_PLUS_Service"/>).
```

For more information on WSDL elements, see Reviewing WSDL Element Details, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.

To view a generated WSDL file:

1. Log in to Oracle E-Business Suite as a user who has the Integration Analyst role. Select the Integration Repository responsibility and the Integration Repository link from the navigation menu.

    The Integration Repository home page appears.

2. Locate your desired interface definition through a search or browse from the interface tree structure within the repository.

3. Click the interface name to open the interface details.

4. In the SOAP Web Service tab (or the Web Service region for XML Gateway interface), click the **View WSDL** link to view the WSDL source code.

    The following sample shows the generated WSDL description with synchronous interaction pattern for the PL/SQL interface PA_COST_PLUS:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<definitions xmlns:tns="http://xmlns.oracle.
com/apps/pa/soaprovider/plsql/pa_cost_plus/"
 xmlns="http://schemas.xmlsoap.org/wsdl/"
 xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
 xmlns:tns1="http://xmlns.oracle.
com/apps/pa/soaprovider/plsql/pa_cost_plus/get_burden_amount/"
name="PA_COST_PLUS"
 targetNamespace="http://xmlns.oracle.
com/apps/pa/soaprovider/plsql/pa_cost_plus/">
<types>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified"
 targetNamespace="http://xmlns.oracle.
com/apps/pa/soaprovider/plsql/pa_cost_plus/get_burden_amount/">
 <include schemaLocation="http://<hostname>:<port>/ISG-ISG-context-
root/isgapp/plsql/pa_cost_plus/APPS_PA_COST_PLUS_GET_BURDEN_AMOUNT.
xsd"/>
</schema>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified"
 targetNamespace="http://xhttp://xmlns.oracle.
com/apps/pa/soaprovider/plsql/pa_cost_plus/">
  <element name="SOAHeader">
   <complexType>
     <sequence>
      <element name="Responsibility" minOccurs="0" type="string" />

      <element name="RespApplication" minOccurs="0" type="string"
/>
      <element name="SecurityGroup" minOccurs="0" type="string" />
      <element name="NLSLanguage" minOccurs="0" type="string" />
      <element name="Org_Id" minOccurs="0" type="string" />
    </sequence>
   </complexType>
  </element>
 </schema>
</type>
<message name="GET_BURDEN_AMOUNT_Input_Msg">
<part name="header" element="tns:SOAHeader" />
<part name="body" element="tns1:InputParameters" />
</message>
<message name="GET_BURDEN_AMOUNT_Output_Msg">
<part name="body" element="tns1:OutputParameters" />
</message>
<portType name="PA_COST_PLUS_PortType">
<operation name="GET_BURDEN_AMOUNT">
<input message="tns:GET_BURDEN_AMOUNT_Input_Msg" />
<output message="tns:GET_BURDEN_AMOUNT_Output_Msg" />
</operation>
</portType>
<binding name="PA_COST_PLUS_Binding" type="tns:
PA_COST_PLUS_PortType">
<soap:binding style="document" transport="http://schemas.xmlsoap.
org/soap/http" />
<operation name="GET_BURDEN_AMOUNT">
<soap:operation soapAction="GET_BURDEN_AMOUNT" />
<input>
<soap:header message="tns:GET_BURDEN_AMOUNT_Input_Msg" part="header"
use="literal" />
<soap:body parts="body" use="literal" />
</input>
<output>
<soap:body use="literal" />
</output>
```

```
        </operation>
        </binding>
        <service name="PA_COST_PLUS_Service">
        <port name="PA_COST_PLUS_Port" binding="tns:PA_COST_PLUS_Binding">
        <soap:address location="#NOT_DEPLOYED#" />
        </port>
        </service>
        </definitions>
```

> **Note:** Values passed in the Responsibility, Responsibility
> Application, Security Group, NLS Language, and Organization ID
> complex types listed under the "SOAHeader" are used to set
> application context during service invocation.
>
> Note that NLS Language and Organization ID are optional values
> to be passed. However, if the invocation of a service is dependent
> on any particular organization, then you must pass the ORG_ID
> element in the "SOAHeader" of that SOAP request.
>
> For more information, see Other Web Service Input Message Parts,
> *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.
>
> The user name and password are defined by the web service
> security policy (such as
> `oracle/wss_username_token_service_policy`). Detailed
> instructions on how to pass the security headers along with the
> SOAP request, see Configuring Web Service Policies, *Oracle E-
> Business Suite Integrated SOA Gateway Developer's Guide*.
>
> For more WSDL element information, see Reviewing WSDL
> Element Details, *Oracle E-Business Suite Integrated SOA Gateway
> Developer's Guide*.

To view a deployed WSDL file:

Once a service has been successfully deployed to an Oracle SOA Suite WebLogic
managed server, the SOAP service status is changed from 'Not Generated' to 'Deployed'
with 'Active' state along with the selected authentication type. Click the **View WSDL**
link to view the deployed WSDL file.

The following sample shows the deployed WSDL description with synchronous
interaction pattern for the same PL/SQL interface PA_COST_PLUS:

```
<?xml version="1.0" encoding="UTF-8" ?>
...
<wsp:Policy xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy"
...
</wsp:Policy>
<types>
<schema xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="
qualified"
 targetNamespace="http://xmlns.oracle.
com/apps/pa/soaprovider/plsql/pa_cost_plus/get_burden_amount/">
<include schemaLocation=http://"<soa_suite_hostname>:<port>/soa-
infra/services/default/<jndi_name>_PLSQL_PA_COST_PLUS/PA_COST_PLUS_Servi
ce?XSD=xsd/APPS_PA_COST_PLUS_GET_BURDEN_AMOUNT.xsd" />
</schema>
<schema xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="
qualified"
...
</schema>
</type>
...
<service name="PA_COST_PLUS_Service">
<port name="PA_COST_PLUS_Port" binding="tns:PA_COST_PLUS_Binding">
<soap:address location="http://"<soa_suite_hostname>:<port>/soa-
infra/services/default/<jndi_name>_PLSQL_PA_COST_PLUS/PA_COST_PLUS_Servi
ce" />
</port>
</service>
</definitions>
```

Reviewing Web Service WADL Sources

Once a REST web service represented in WADL has been successfully deployed, the
REST Service Status field is changed from 'Not Deployed' to 'Deployed'. The WADL
link appears in the REST Web Service tab allowing you to view the WADL description.

For example, the following WADL description is for a PL/SQL API Invoice Creation (
AR_INVOICE_API_PUB) that includes 'CREATE_INVOICE' and
'CREATE_SINGLE_INVOICE' REST service operations:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<application xmlns:tns="http://xmlns.oracle.
com/apps/ar/soaprovider/plsql/rest/ar_invoice_api_pub/" xmlns="http:
//wadl.dev.java.net/2009/02"
xmlns:tns1="http://xmlns.oracle.com/apps/ar/rest/ar/create_invoice/"
name="AR_INVOICE_API_PUB"
targetNamespace="http://xmlns.oracle.
com/apps/ar/soaprovider/plsql/rest/ar_invoice_api_pub/">
    <grammars>
     <include xmlns="http://www.w3.org/2001/XMLSchema" href="http:
//<hostname>:<port>/webservices/rest/Invoice/?
XSD=CREATE_INVOICE_SYNCH_TYPEDEF.xsd" />
   <include xmlns="http://www.w3.org/2001/XMLSchema" href="http:
//<hostname>:<port>/webservices/rest/Invoice/?
XSD=CREATE_SINGLE_INVOICE_SYNCH_TYPEDEF.xsd" />
    </grammars>
 <resources base="http://<hostname>:<port>/webservices/rest/Invoice/">
        ...
 </resources>
</application>
```

> **Note:** The service alias value Invoice entered earlier before service

deployment is now displayed as part of the schema for the service operations - 'CREATE_INVOICE' and 'CREATE_SINGLE_INVOICE'.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<application xmlns:tns=...
...
  <resources base="http://<hostname>:<port>/webservices/rest/Invoice/">
  <resource path="/create_invoice/">
   <method id="CREATE_INVOICE" name="POST">
    <request>
     <representation mediaType="application/xml" type="tns1:
InputParameters" />
     <representation mediaType="application/json" type="tns1:
InputParameters" />
    </request>
    <response>
     <representation mediaType="application/xml" type="tns1:
OutputParameters" />
     <representation mediaType="application/json" type="tns1:
OutputParameters" />
    </response>
   </method>
  </resource>
  <resource path="/create_single_invoice/">
   <method id="CREATE_SINGLE_INVOICE" name="POST">
    <request>
     <representation mediaType="application/xml" type="tns2:
InputParameters" />
     <representation mediaType="application/json" type="tns2:
InputParameters" />
    </request>
    <response>
     <representation mediaType="application/xml" type="tns2:
OutputParameters" />
     <representation mediaType="application/json" type="tns2:
OutputParameters" />
    </response>
   </method>
  </resource>
 </resources>
</application>
```

> **Note:** POST is shown as the method name for two service operations 'CREATE_INVOICE' and 'CREATE_SINGLE_INVOICE'.
>
> Input and output messages can be exchanged in both the XML and JSON formats for both service operations.

If the deployed REST service is an interface type of PL/SQL, Business Service Object, Java Bean Services, or Application Module Services, then both the GET and POST can be shown as the supported methods in the REST service operation. For example, the following WADL description shows many methods contained in the Employee Information service. The getPersonInfo operation is implemented with both the POST and GET HTTP methods.

```xml
<xml version="1.0" encoding="UTF-8">
<application name="EmployeeInfo" targetNamespace="http://xmlns.oracle.
com/apps/per/soaprovider/pojo/employeeinfo/"
 xmlns:tns="http://xmlns.oracle.
com/apps/per/soaprovider/pojo/employeeinfo/"
 xmlns="http://wadl.dev.java.net/2009/02" xmlns:xsd="http://www.w3.
org/2001/XMLSchema"
 xmlns:tns1="http://xmlns.oracle.
com/apps/fnd/rest/empinfo/getallreports/"
 xmlns:tns2="http://xmlns.oracle.
com/apps/fnd/rest/empinfo/getdirectreports/"
 xmlns:tns3="http://xmlns.oracle.
com/apps/fnd/rest/empinfo/getpersoninfo/">

<grammars>
    <include href="http://<hostname>:<port>/webservices/rest/empinfo/?
XSD=getallreports.xsd" xmlns="http://www.w3.org/2001/XMLSchema" />
    <include href="http://<hostname>:<port>/webservices/rest/empinfo/?
XSD=getdirectreports.xsd" xmlns="http://www.w3.org/2001/XMLSchema" />
    <include href="http://<hostname>:<port>/webservices/rest/empinfo/?
XSD=getpersoninfo.xsd" xmlns="http://www.w3.org/2001/XMLSchema" />
</grammars><resources base="http://<hostname>:
<port>/webservices/rest/empinfo/">
  <resource path="/getAllReports/">
 <method id="getAllReports" name="GET">
     <request>
     <param name="ctx_responsibility" type="xsd:string" style="query"
required="false" />
           <param name="ctx_respapplication" type="xsd:string" style="
query" required="false" />
     <param name="ctx_securitygroup" type="xsd:string" style="query"
required="false" />
     <param name="ctx_nlslanguage" type="xsd:string" style="query"
required="false" />
     <param name="ctx_orgid" type="xsd:int" style="query" required="
false" />
    </request>
    <response>
     <representation mediaType="application/xml" type="tns1:
getAllReports_Output" />
     <representation mediaType="application/json" type="tns1:
getAllReports_Output" />
    </response>
   </method>
  </resource>
 <resource path="/getDirectReports/">
  <method id="getDirectReports" name="GET">
    <request>
     <param name="ctx_responsibility" type="xsd:string" style="query"
required="false" />
              <param name="ctx_respapplication" type="xsd:string"
style="query" required="false" />
     <param name="ctx_securitygroup" type="xsd:string" style="query"
required="false" />
     <param name="ctx_nlslanguage" type="xsd:string" style="query"
required="false" />
     <param name="ctx_orgid" type="xsd:int" style="query" required="
false" />
    </request>
     <response>
     <representation mediaType="application/xml" type="tns2:
getDirectReports_Output" />
     <representation mediaType="application/json" type="tns2:
getDirectReports_Output" />
    </response>
   </method>
```

```
       </resource>
 <resource path="="/getPersonInfo/ {personId}/">
  <param name="personId" style="template" required="true" type="xsd:int"
/>
 <method id="getPersonInfo" name="GET">
     <request>
     <param name="ctx_responsibility" type="xsd:string" style="query"
required="false" />
          <param name="ctx_respapplication" type="xsd:string" style="
query" required="false" />
     <param name="ctx_securitygroup" type="xsd:string" style="query"
required="false" />
     <param name="ctx_nlslanguage" type="xsd:string" style="query"
required="false" />
     <param name="ctx_orgid" type="xsd:int" style="query" required="
false" />
    </request>
    <response>
     <representation mediaType="application/xml" type="tns3:
getPersonInfo_Output" />
     <representation mediaType="application/json" type="tns3:
getPersonInfo_Output" />
    </response>
   </method>
  </resource>
 <resource path="/getPersonInfo/">
  <method id="getPersonInfo" name="POST">
    <request>
      <representation mediaType="application/xml" type="tns3:
getPersonInfo_Input" />
     <representation mediaType="application/xml" type="tns3:
getPersonInfo_Output" />
    </request>
     <response>
     <representation mediaType="application/xml" type="tns3:
getPersonInfo_Input" />
     <representation mediaType="application/xml" type="tns3:
getPersonInfo_Output" />     </response>
   </method>
  </resource>
  </resource path>
</application>
```

- {personId} is a path variable, defined using the <param> tag after the <resource> tag and before the <method> tag. Client program should replace the path variable with actual value at runtime. For example, an employee's Id will be passed in an HTTP GET request when the getPersonInfo service operation is invoked. This returns the associated employee's name and his or her manager's name.

  For information on how the path variable can be defined, see Annotations for Java Bean Services, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide* or Annotations for Application Module Services, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.

- For the GET method, application context values, including

```
        Responsibility, Responsibility Application,
        Security Group, NLS Language, and Organization ID
        complex types, are passed as query strings in the RESTHeader
        element.
```

For more information about WADL description, see Reviewing WADL Element Details,
Oracle E-Business Suite Integrated SOA Gateway Developer's Guide.

To view a deployed WADL file:

1. Log in to Oracle E-Business Suite as a user who has the Integration Analyst role.
    Select the Integration Repository responsibility and the Integration Repository link
    from the navigation menu.

    The Integration Repository home page appears.

2. Locate your desired interface definition through a search or browse from the
    interface tree structure within the repository.

3. Click the interface name to open the interface details page.

4. In the REST Web Service tab, click the **View WADL** link to view the WADL source
    code.

Managing Grants

Only integration administrators or users who have the Integration Administrator role
can create security grants by authorizing the access permissions of interface methods
(or procedures and functions) to a user, a user group, or all users. Similarly, the
administrators can revoke the grants from an authorized user, user group, or all users
on a selected method or service.

Managing Grants in the Grants Tab

Interface types that have the security grant feature available are PL/SQL, Concurrent
Program, Business Service Object, Java Bean Services, Application Module Services, and
Open Interface Table and View. You can manage grants for these interfaces using the
Grants tab in the interface details page.

> **Note:** Security grants for XML Gateway interfaces are managed in the
> Trading Partner User Setup Form although XML Gateway interfaces
> can only be exposed as SOAP services. See Managing XML Gateway
> User Security in the Trading Partner User Setup Form, page 4-34.

Interface Details Page with Grants Tab Highlighted

Note that the grant feature applies to the interfaces that can be exposed as both SOAP and REST services. These interfaces are PL/SQL, Concurrent Program, and Business Service Object.

For example, when a user (OPERATIONS) is authorized to have access permission on a PL/SQL API method name called 'Create Credit Request', the user will have the permission to access the associated 'Create Credit Request' service operations of both SOAP and REST service types through the same grant.

Creating Security Grants

To create a grant, select appropriate method name checkboxes in the Grants tab. Click **Create Grant** to open the Create Grants page.

Create Grants Page

In the Create Grants page, select a grantee type and grantee name if it's applicable. Click **Apply**. This creates security grants for the selected methods and the associated SOAP and REST services if available.

Revoking Security Grants

To revoke a grant in the Grants tab, the administrator can perform the action in two ways:

- *Revoking Grants for a Single Procedure and Function*

  Select a desired procedure and function from the Service Operations region first and then click **Revoke Grant**. The Revoke Grants page displays the existing grants details assigned to the selected procedure and function.

  Select one or more existing grants from the table for the selected procedure, and click **Revoke Grant** to revoke the grants.

- *Revoking Commonly Assigned Grants to All Procedures*

  Select more than one procedure and function name that have grants created earlier, and click **Revoke Grant** in the Grants tab. The Revoke Grants page is displayed where the administrator can find existing grants that are commonly assigned to the selected procedures and functions.

  For example, two procedures and functions (such as 'Create Credit Request' and 'Get Application Number') are assigned to the same User (grantee type) 'operations' (grantee name). This common grant User 'operations' is displayed in the second table of the Revoke Grants page.

Revoke Grants Page

The administrator should be able to select the desired common grant(s) (such as User 'operations' in the above example) and click **Revoke Grant**. The specified common grant(s) should be removed for the selected procedures and functions.

For more information about how to manage grants in the Grants tab, see Managing Security Grants for the SOAP and REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

Managing XML Gateway User Security in the Trading Partner User Setup Form

For XML Gateway interfaces, authorizing users to perform XML Gateway inbound transactions with a trading partner is performed in Oracle XML Gateway instead. The administrator needs to:

• Set the "ECX: Enable User Check for Trading Partner" profile option to "Yes" to enable trading partner specific security feature

• Associate users with a trading partner

  Log in to Oracle E-Business Suite as a user who has the XML Gateway responsibility. Navigate to **Setup** and then select **Define Trading Partners** from the navigation menu. In the Define Trading Partner Setup form, click the **User Setup** button to access the Trading Partner User Setup form where the administrator can associate users with a trading partner.

For more information about trading partner user security, refer to Trading Partner

Setup, XML Gateway Setup chapter, *Oracle XML Gateway User's Guide*.

XML Gateway Map Information

The following diagram illustrates the basic structure of the XML Gateway Map information page and its connections to related pages:

XML Gateway Map Basic Structure
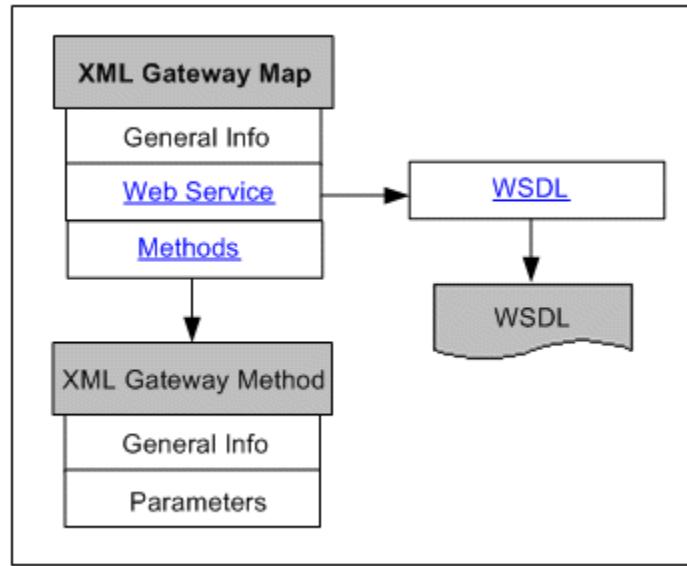
The XML Gateway Map information page includes standard fields shown in the common region of the selected XML Gateway interface along with the Web Service region and Method region.

For XML Gateway interface, it can be either inbound or outbound direction. An inbound XML Gateway map indicates that the interface receives incoming transactions or messages into the Oracle E-Business Suite. An outbound map indicates that the interface sends outgoing transactions or messages to another system. For more common information shown in the interface details page, see Common Information on Interface Details, page 4-3.

XML Gateway Interface Details Page with Show Generic XML Gateway Service Link Highlighted

- **Web Service Region**

  The Web Service region appears only for inbound XML Gateway maps. It contains interface or service information for a given interface.

  This region contains the SOAP service details, including service status, WSDL description, and authentication type information, for the selected XML Gateway map.

  There is no Interaction Pattern table displayed for XML Gateway map. This is because each XML Gateway interface contains only one method, and this interface type can be service enabled only with synchronous operation pattern. If a selected XML Gateway interface is exposed as a web service, the single method contained in the interface must be generated only with the support for synchronous pattern by default.

  For more information on each field in the Web Service region, see Common

Information on SOAP Web Services, page 4-6.

Generic XML Gateway Service Subregion

If your system is upgraded from a previous Oracle E-Business Suite release and if you have been using generic XML Gateway web services, the generic XML Gateway service information can be displayed in the Web Service region for the selected XML Gateway map.

Setup Tasks

To successfully display the generic XML Gateway service and its WSDL URL in the Generic XML Gateway Service subregion, ensure the following tasks are in place:

- The *FND: XML Gateway Map Generic Service* profile value must be set to 'Yes'.

  Use this profile option to display or hide the Generic XML Gateway Service subregion for the selected XML Gateway interface. If it is set to 'Yes', the Generic XML Gateway Services subregion can be displayed within the Web Service region.

- The generic XML Gateway service must be deployed. Otherwise, no WSDL URL is shown.

  Once a generic XML Gateway service has been deployed, the deployed service WSDL URL is populated as the profile value for the *ISG: Generic Service WSDL URL for XMLG* profile option. The WSDL URL is also displayed in the Generic XML Gateway Service subregion.

  If the generic XML Gateway service is not deployed, the profile value will not be shown. Therefore, no WSDL URL is displayed in the subregion and the Web Service Status field is marked as 'Not Deployed'.

For more setup information about generic XML Gateway services, see *Installing Oracle E-Business Suite Integrated SOA Gateway, Release 12.2*, available from My Oracle Support Knowledge Document 1311068.1 for details.

For information on setting profile options, see Setting Profile Options, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

Displaying Generic XML Gateway Services

Click the **Show Generic XML Gateway Service** or **Hide Generic XML Gateway Service** link to display or close the Generic XML Gateway Service subregion for the selected XML Gateway interface.

This subregion contains the following fields:

- **Web Service Status:** This field indicates the current state of the generic XML Gateway service.

  If the setup is not configured for the generic XML Gateway service, the Web Service Status field is displayed as 'Not Deployed'.

- **View Generic WSDL**: Click the **View Generic WSDL** link to display the deployed generic WSDL URL for the selected XML Gateway interface.

  The deployed generic WSDL URL has the following syntax:

  ```
  http://<SOA server host>:<SOA Suite managed server
  port>/soa-infra/services/default/XMLGatewayService!<version
  chosen while deploying>XMLGateway?WSDL
  ```

  - `<SOA Suite managed server port>`: It is the port of the server where the SOA composite is deployed.

  - `<version chosen while deploying>`: At the time of deployment, the deployment version will be asked. The default version value is 1.0.

    For example, `http://<SOA server host>:<SOA Suite managed server port>/soa-infra/services/default/XMLGatewayService!1.0/XMLGateway?WSDL`.

  After the upgrade to Oracle E-Business Suite Release 12.2, the deployed WSDL URL information has been changed from a previous release. Therefore, you may have to replace it with the new WSDL URL and service location or address accordingly in web service clients while invoking the generic XML Gateway

service.

The updated WSDL URL is also populated in the *ISG: Generic Service WSDL URL for XMLG* profile option by default if the setup tasks for generic XML Gateway services are configured properly.

- **Interaction Pattern:** 'Synchronous' is displayed by default in read-only mode.

- **Authentication Type:** 'Username Token' is displayed by default in read-only mode.

Performing Web Service Activities in the Web Service Region

If a web service has been generated successfully, an integration administrator can perform additional administrative tasks including deploying the generated service, regenerating the service, or clearing up the generated service artifact which changes the Web Service Status field from 'Generated' to 'Not Generated'.

If a web service has been successfully deployed as an active service in Oracle SOA Suite, the integration administrator can perform additional lifecycle activities for the deployed service, including undeploying the active service, resetting the deployed service to its initial state - 'Not Generated', retiring the service so that it is no longer to accept new SOAP requests, or activating the retired service so that it can become active again.

For more information on these administrative tasks, see Performing Administrative Activities for SOAP Web Services, page 4-12.

- **Methods Region**

  The XML Gateway method details page appears when you click a method name in the Methods region.

XML Gateway Method Details Page

The Methods region includes a table listing the XML Gateway method parameters, including each parameter's data type and whether the parameter is required or not.

> **Note:** Security grants for XML Gateway interfaces are managed in the Trading Partner User Setup Form although XML Gateway interfaces can be exposed as SOAP services. See Managing XML Gateway User Security in the Trading Partner User Setup Form, page 4-34.

PL/SQL Information

The following diagram illustrates the basic structure of the PL/SQL information page and its connection to the related PL/SQL procedure and function details:

PL/SQL Interface Basic Structure
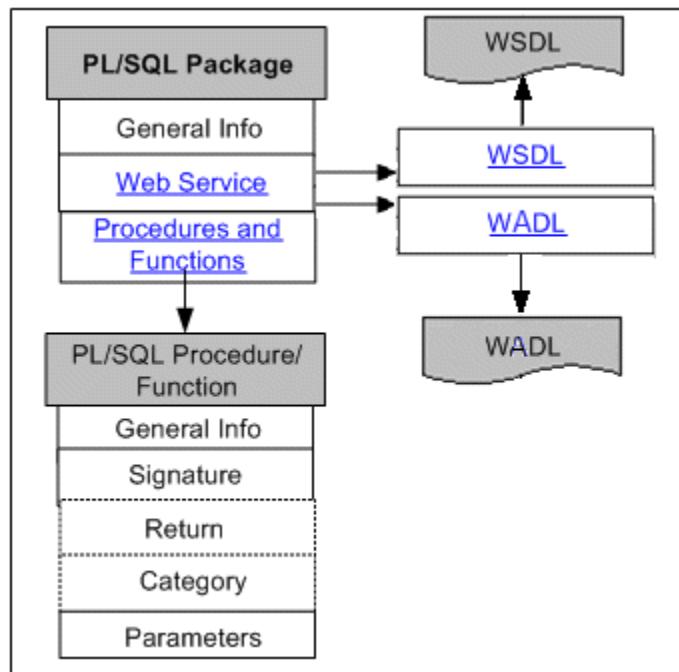
PL/SQL interfaces can be exposed as both SOAP services and REST services. Therefore, the PL/SQL interface details page includes general section of a selected PL/SQL interface, the Overview tab, the SOAP Web Service tab, and the REST Web Service tab.

Users who have the Integration Administrator role can find an additional Grants tab displayed in the interface details page. This tab allows the administrator to create and revoke security grants. For more information on how to manage security grants, see Managing Grants, page 4-31.

For information on the general section, see Common Information, page 4-1.

PL/SQL Interface Details Page

For more information about web services, see Understanding Web Services, page 2-1.

- **Overview Tab**

    This tab displays read-only information about the selected PL/SQL API. It includes full description, interface source information, as well as methods (or procedures and functions) contained in the selected interface.

    For more information on the interface source information, see Common Information on Interface Details, page 4-3.

- **SOAP Web Service Tab**

    This tab contains SOAP service information for a selected PL/SQL interface. This includes service status, WSDL description, interaction pattern, and authentication type information.

    For more information about SOAP service, see Common Information on SOAP Web Services, page 4-6.

For information on viewing WSDL description, see Reviewing Web Service WSDL Source, page 4-23.

If a SOAP service has been successfully generated, the integration administrators can perform additional administrative tasks including deploying the generated service, regenerating the service, or clearing up the generated service artifact.

If a web service has been successfully deployed as an active service in Oracle SOA Suite, the integration administrators can undeploy the active service, reset the deployed service to its initial state - 'Not Generated', retire the service, or activate the retired service so that it can become active again.

For more information on these administrative tasks, see Performing Administrative Activities for SOAP Web Services, page 4-12.

- **REST Web Service Tab**

  This tab contains REST service information for the selected PL/SQL API. This includes service alias, service status, WADL description, verb, service operation, and authentication type information.

  POST and GET are the supported HTTP verbs in this release. For more information about REST service, see Common Information on REST Web Services, page 4-15.

  All REST services are secured by HTTP Basic Authentication or Security Token Authentication at HTTP or HTTPS transport level. For more information on REST service security, see Managing Web Service Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  > **Note:** HTTPS is the recommended secure transport protocol while using the HTTP Basic Authentication security to authenticate user credentials (user name and password).

  If a REST service has been successfully deployed to an Oracle E-Business Suite WebLogic server, the integration administrator can undeploy the service to reset the service to its initial state - 'Not Deployed'.

  For more information on these administrative tasks, see Performing Administrative Activities for REST Web Services, page 4-20.

PL/SQL Method Details Page

The PL/SQL method details page appears when you click a method name in the Overview tab, the SOAP Web Service tab, or the REST Web Service tab.

PL/SQL Method Details Page

This page displays the signature of the selected method, and a table listing the parameters and their attributes.

Java Information

The following diagram illustrates the basic structure of the Java information page and its connection to the related Java method information page.

Java Interface Basic Structure
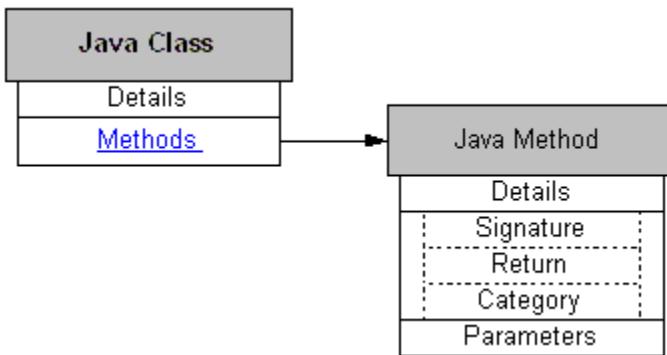
The general section of the Java information page displays common information, page 4-1 for the selected Java class. This page also contains a table listing the class methods, including active status and internal name. Click the Java method name link to access the Java Method Details page, page 4-45.

If you have the Integration Administrator role, the **Grants** tab appears. This lets you grant the access permissions of selected methods to a user, a user group, or all users. For more information on how to create security grants, see Managing Grants, page 4-31.

Java Method Information

The Java method details page appears when you click a method name on the Java interface details page.

The general section of the Java method information page displays common information for the selected method, plus the following additional fields:

- **Interface**

  This Interface field displays the interface that uses this Java method. Click the link to view the interface details.

- **See Also**

  This See Also field displays a related Java method. Click a related Java method name link to view the Java method details.

This page also displays the signature of a selected method, information about the return type, and a table listing the method parameters.

Subtype of Java APIs

Some Java APIs are categorized as a subtype of Java interfaces. To locate those Java APIs, you must perform a search through the combination of Category and Category Value fields.

For information on how to locate these Java APIs through a search, see Searching for an Integration Interface, page 3-10.

If your selected interface belongs to these subtypes of Java APIs, the interface details page may contain web service information if the selected interface is exposed as a service. For the interface details page of these APIs, see:

- Java Bean Services, page 4-46

- Application Module Services, page 4-49

- Security Services, page 4-52

- Java APIs for Forms, page 4-55

Java Bean Services

The following diagram illustrates the basic structure of the Java Bean Services information page and its connection to the related Java method details:

Java Bean Services Basic Structure
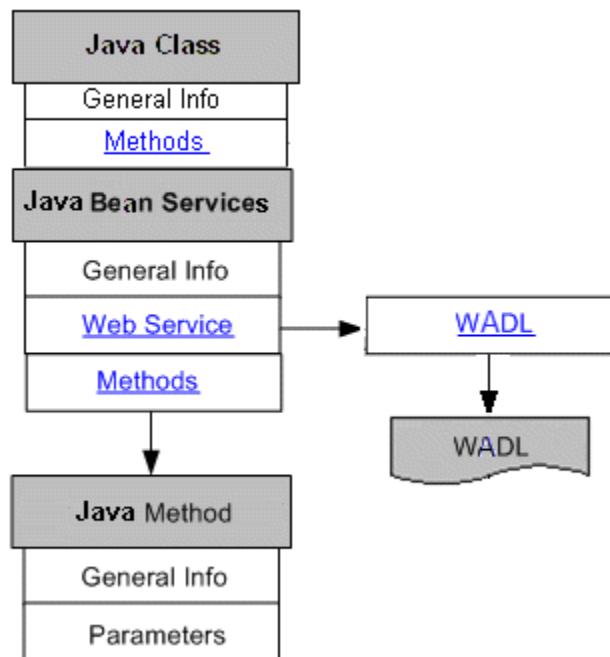
Searching Java Bean Services Interfaces

To easily locate Java Bean Services through the Search page, click **Show More Search Options** to display more search fields.

Enter the following key search values as the search criteria:

- Category: Interface Subtype

- Category Value: Java Bean Services

For information about category and category value fields, see Category and Category Value, page 3-12.

Viewing Java Bean Services

Java Bean Services can be exposed as REST services only. The interface details page contains the Overview tab and the REST Web Service tab. An integration administrator can find an additional Grants tab displayed in the page. This Grants tab allows the integration administrator to create and revoke security grants. For more information on how to manage security grants, see Managing Grants, page 4-31.

For information on the general section, see Common Information, page 4-1.

For more information about web services, see Understanding Web Services, page 2-1.

Java Bean Service Interface Details Page

- **Overview Tab**

  This tab displays read-only information about the selected interface. It includes full description, interface source information, as well as methods contained in the interface.

  For more information on the interface source information, see Common Information on Interface Details, page 4-3.

- **REST Web Service Tab**

  This tab contains REST service information for the selected interface. This includes service alias, service status, WADL description, verb, service operation, and authentication type information.

  For more information about REST service, see Common Information on REST Web Services, page 4-15.

  For Java Bean Services annotation information, see Annotations for Java Bean

Services, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.

Similar to the PL/SQL REST services, all Java Bean Services are secured by HTTP Basic Authentication or Security Token Authentication at HTTP or HTTPS transport level. For more information on REST service security, see Managing Web Service Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

> **Note:** HTTPS is the recommended secure transport protocol while using the HTTP Basic Authentication security to authenticate user credentials (user name and password).

If a REST service has been successfully deployed to an Oracle E-Business Suite WebLogic server, the integration administrator can undeploy the service to reset the service to its initial state - 'Not Deployed'.

For more information on these administrative tasks, see Performing Administrative Activities for REST Web Services, page 4-20.

Application Module Services

The following diagram illustrates the basic structure of the Application Module Services information page and its connection to the related Java method details:

Application Module Services Basic Structure
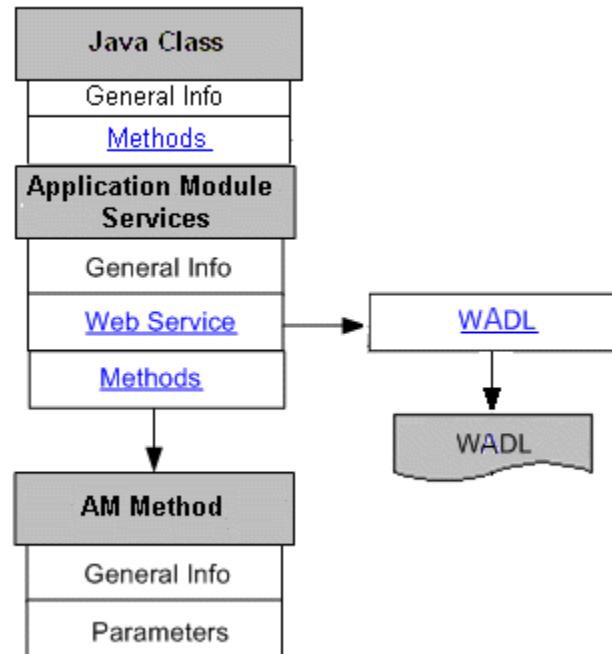
Searching Application Module Services Interfaces

To quickly locate Application Module Services through the Search page, click **Show More Search Options** to display more search fields. Enter the following key search values as the search criteria:

- Category: Interface Subtype

- Category Value: Application Module Services

For information about category and category value fields, see Category and Category Value, page 3-12.

Viewing Application Module Services

Similar to Java Bean Services that can be exposed as REST services only, the interface details page contains the Overview tab and the REST Web Service tab. An integration repository administrator can find an additional Grants tab for managing security grants. For more information on how to manage security grants, see Managing Grants, page 4-31.

For information on the general section, see Common Information, page 4-1.

For more information about web services, see Understanding Web Services, page 2-1.

Application Module Service Interface Details Page

- **Overview Tab**

  This tab displays read-only information about the selected interface. It includes full description, interface source information, as well as methods contained in the interface.

  For more information on the interface source information, see Common Information on Interface Details, page 4-3.

- **REST Web Service Tab**

  This tab contains REST service information for the selected interface. This includes service alias, service status, WADL description, verb, service operation, and authentication type information.

  For more information about REST service, see Common Information on REST Web Services, page 4-15.

For Application Module Services annotation information, see Annotations for Application Module Services, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.

Application Module Services are secured by HTTP Basic Authentication or Security Token Authentication at HTTP or HTTPS transport level. For more information on REST service security, see Managing Web Service Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

> **Note:** HTTPS is the recommended secure transport protocol while using the HTTP Basic Authentication security to authenticate user credentials (user name and password).

If a REST service has been successfully deployed, the administrator can undeploy the service to reset the service to its initial state - 'Not Deployed'.

For more information on these administrative tasks, see Performing Administrative Activities for REST Web Services, page 4-20.

Security Services

Security services are a set of predefined and predeployed REST services from Oracle Application Object Library. These services include Authentication and Authorization services and are used for mobile applications.

For example, Login service validates the user credentials and returns an access token. Logout service invalidates the access token and any associated authentication sessions. These two services are included in the Authentication service that helps session initialization with security or applications context information. Authorization service retrieves the Access Control List which may contain assigned responsibilities, roles, and privileges for all logged-in users.

For more information about these REST security services, refer to the *Oracle E-Business Suite Security Guide*.

Searching and Viewing Security Services

To easily locate security services through the Search page, click **Show More Search Options** and then enter search information in the combination of Category (subtype) and Category Value (Security Services) fields.

Search Page with Criteria for Security Services

For information about category and category value fields, see Category and Category Value, page 3-12.

Viewing Security Service Interface Details

After a search, select a desired security service from the search results table. The interface details page for the selected security service is displayed. It contains some common interface information, REST Web Service region, and Methods region.

Security Service Interface Details Page

For information on the general section, see Common Information, page 4-1.

REST Web Service Region

The REST Web Service region contains the selected security service information.

- **REST Service Status:** 'Deployed' is always displayed for security services because all security services are predeployed REST services.

  Click the **View WADL** link to view the WADL description for the selected security service in a separate window.

- **Verb**: This field displays the Verb value indicating how the REST service is implemented using an HTTP method.

  'POST' is the only method supported in this release.

Please note that security services are pregranted to all Oracle E-Business Suite users which means that all the users can invoke these services.

Methods Region

In the Methods region, click a method name link to open the Java Method Details page for the selected method.

Java APIs for Forms

Java APIs for Forms are XML document-based integration points wrapped in Java classes for processing business logic in Oracle Forms.

Searching and Viewing Java APIs for Forms Interfaces

Similar to other subtype of Java APIs, you can perform a search by clicking **Show More Search Options** to quickly locate the Java APIs for Forms through the combination of Category (Interface Subtype) and Category Value (Java APIs for Forms) fields.

For information about category and category value fields, see Category and Category Value, page 3-12.

To view the interface details, select a desired Java APIs for Forms interface from the search result table. The selected interface details should appear.

Please note that Java APIs for Forms interfaces are not serviceable interfaces and cannot be exposed as SOAP services. Refer to My Oracle Support Knowledge Document 966982.1 for the suggested alternatives to the existing Java APIs for Forms interfaces.

Business Service Object

Business service object (BSO) interface type, formerly known as service bean, provides the access to SOA services and facilitates integration between Oracle E-Business Suite and trading partners. This type of interfaces can be used directly or exposed as SOAP or REST services. BSO interface type often employs *service data objects* as parameters to pass complex data.

A service data objects (SDO) defines a generic API for accessing and manipulating structured data. It is designed to simplify and unify the way in which applications handle data. The SDO API is independent of the actual data source. For example, SDO can be used to access XML data or SQL data. For more information about SDO, see Reviewing Service Data Objects, page 4-62.

> **Note:** A business service object is not actually an interface type, but rather an object used by one or more Java service interfaces or other service data objects to pass data. Oracle Integration Repository includes it on list of interface types, so that you can browse or search for Java service interfaces based on the business service objects that they use.

The following diagram illustrates the basic structure of the business service object interface information page and its connections to related pages:

Business Service Object Interface Basic Structure
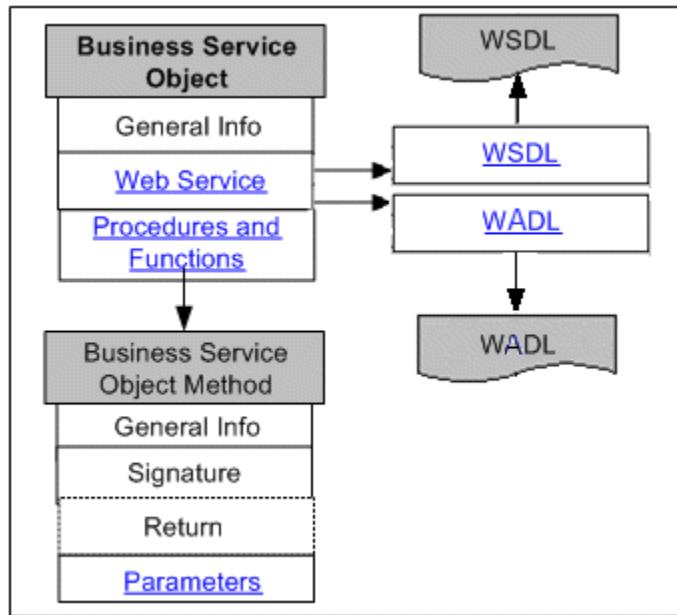
Business Service Object Interface Details Page

This interface type can be exposed as both SOAP services and REST services. Therefore, the interface details page includes general section of a selected business service object interface, the Overview tab, the SOAP Web Service tab, and the REST Web Service tab.

Users who have the Integration Administrator role can find an additional Grants tab displayed in the interface details page. This tab allows the administrator to create and revoke security grants. For more information on how to manage security grants, see Managing Grants, page 4-31.

Business Service Object Interface Details Page

For more information about web services, see Understanding Web Services, page 2-1.

- **Overview Tab**

  This tab displays read-only information about the selected business service object. It includes full description, interface source information, as well as methods contained in the selected business service object.

  For more information on the interface source information, see Common Information on Interface Details, page 4-3.

  For more information about methods, see Business Service Object Interface Method Information, page 4-58.

- **SOAP Web Service Tab**

  This tab contains SOAP service information for a selected business service object. This includes service status, WSDL description and authentication type information.

  For more information about SOAP service, see Common Information on SOAP Web Services, page 4-6.

  For information on viewing WSDL description, see Viewing WSDL Description, page 4-60.

  If a SOAP service has been successfully generated, the integration administrators can perform additional administrative tasks including deploying the generated

service, regenerating the service, or clearing up the generated service artifact.

If a web service has been successfully deployed as an active service in Oracle SOA Suite, the integration administrators can undeploy the active service, reset the deployed service to its initial state - 'Not Generated', retire the service, or activate the retired service so that it can become active again.

For more information on these administrative tasks, see Performing Administrative Activities for SOAP Web Services, page 4-12.

- **REST Web Service Tab**

  This tab contains REST service information for the selected business service object. This includes service alias, service status, WADL description, verb, service operation, and authentication type information.

  Please note that POST and GET HTTP verbs are supported for BSO as REST service operations. For more information about REST service, see Common Information on REST Web Services, page 4-15.

  All REST services are secured by HTTP Basic Authentication or Security Token Authentication at HTTP or HTTPS transport level. For more information on REST service security, see Managing Web Service Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  > **Note:** HTTPS is the recommended secure transport protocol while using the HTTP Basic Authentication security to authenticate user credentials (user name and password).

  If a REST service has been successfully deployed to an Oracle E-Business Suite WebLogic server, the integration administrator can undeploy the service to reset the service to its initial state - 'Not Deployed'.

  For more information on these administrative tasks, see Performing Administrative Activities for REST Web Services, page 4-20.

Business Service Object Interface Method Information

The method details page appears when you click a method name in the Overview tab, the SOAP Web Service tab, or the REST Web Service tab.

Business Service Object Method Details Page

In addition to common information, the general section of the method details page contains a link to the interface that uses this method.

The following regions also appear on the method details page:

- **Signature**

  The region describes the interface method, parameter type, value, and return information.

- **Return**

  If the return type is a service data object, click the link in the **Type** field to access the service data object details page.

- **Parameters**

  If a parameter is a service data object, click the link in the **Type** column to access the service data object details page. See: Reviewing Service Data Object, page 4-62.

  > **Note:** An XML schema describes the structure of an XML document with all input and output message definitions and data types. Click the **XML schema** link that is associated with your selected business service object to view the XML schema document displayed in a separate window.

Integration Repository Service

Based on business service object interface, Integration Repository Service is a service

component in the Integration Repository. It queries Integration Repository data, and provides information about all the interface definitions to facilitate the integration between Oracle E-Business Suite and trading partners.

When you search for Integration Repository Service through the business service object interface type, all business service objects contained in the Integration Repository Service are displayed.

Integration administrators or users who have the Integration Administrator role can create security grants by authorizing each business service object method contained in the service to a user, a user group, or all users. For more information, see Managing Grants, page 4-31.

Use the following steps to locate the Integration Repository Service:

1. Log in to Oracle E-Business Suite as a user who has the Integration Analyst role. Select the Integration Repository responsibility and the Integration Repository link from the navigation menu.

2. Click **Search**.

3. Enter the following information in the Search page:

    *   Product Family: Application Technology

    *   Interface Type: Business Service Object

4. Click **Go** to run the search.

5. Click the **Integration Repository Service** link from the search result table.

    This opens the Business Service Object interface details page.

    See: Business Service Object Interface Details Page, page 4-56.

6. Click a method name link to get to business service object method details.

    See: Business Service Object Method Information, page 4-58.

Viewing WSDL Description

In the SOAP Web Service tab, if the selected business service object has an associated SOAP service available, click the **View WSDL** link to view the WSDL file. The following sample shows the WSDL description for the Integration Repository Service:

```
<?xml version="1.0"?>
<definitions name="IntegrationRepositoryService"
targetNamespace="http://xmlns.oracle.
com/oracle/apps/fnd/rep/ws/IntegrationRepositoryService"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns2="http://xmlns.oracle.com/apps/fnd/ServiceBean"
  xmlns:tns1="http://xmlns.oracle.com/apps/fnd/rep/ws"
  xmlns:tns="http://xmlns.oracle.
com/oracle/apps/fnd/rep/ws/IntegrationRepositoryService">
<types>
<xsd:schema>
  <xsd:import namespace="http://xmlns.oracle.com/apps/fnd/rep/ws"
schemaLocation="http://<hostname>:<port>/ISG-ISG-context-
root/isgapp/apps/bso/ws/IntegrationRepositoryService.xsd"/>
</xsd:schema>
<xsd:schema elementFormDefault="qualified" targetNamespace="http:
//xmlns.oracle.com/apps/fnd/ServiceBean" >
  <xsd:element name="ServiceBean_Header">
    <xsd:complexType>
      <xsd:element name="RESPONSIBILITY_NAME" minOccurs="0" type="xsd:
string"/>
      <xsd:element name="RESPONSIBILITY_APPL_NAME" minOccurs="0" type="
xsd:string"/>
      <xsd:element name="SECURITY_GROUP_NAME" minOccurs="0" type="xsd:
string"/>
      <xsd:element name="NLS_LANGUAGE" minOccurs="0" type="xsd:string"/>
      <xsd:element name="ORG_ID" minOccurs="0" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>
  </xsd:element>
 </xsd:schema>
  . . .
```

> **Note:** Values passed in the Responsibility Name, Responsibility
> Application Name, Security Group, NLS Language, and Organization
> ID elements listed under the ServiceBean_Header are used to set
> application context during service invocation.
>
> Note that NLS Language and Organization ID are optional values to be
> passed. However, if the invocation of a Business Service Object
> interface is dependent on a particular organization, then you must pass
> the ORG_ID element in the ServiceBean_Header of that SOAP request.
>
> For more information, see Other Web Service Input Message Parts,
> *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*.
>
> The user name and password are defined by the web service security
> policy (such as `oracle/wss_username_token_service_policy`).
> Detailed instructions on how to pass the security headers along with
> the SOAP request, see Configuring Web Service Policies, *Oracle E-
> Business Suite Integrated SOA Gateway Developer's Guide*.

If error occurred, the following error message information appears under
<Method>_Response:

```
...
<xsd: complexType name ="
IntegrationRepositoryService_GetServiceDescription_Response">
  ...
  <xsd:sequence>
     <xsd:element name="serviceDescription" type="oans3:
ServiceDescription" minOccurs="0" nillable="true"/>
    <xsd:element name="Message" type="oans1:ServiceMessage" minOccurs="
0" maxOccurs="unbounded"/>

    <xsd:element name="ErrorMessage" type="oans1:ServiceMessage"
minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd: complexType name>
```

> **Note:** The Message and ErrorMessage elements listed under
> <Method>_Response are used for error messages. The Message element
> will appear as warning messages in the SOAP response. It is used to
> display any warning message returned by the API. The ErrorMessage
> element corresponds to OAExceptions that were raised during the
> method invocation. In general, the response for any service method can
> contain any of the following:
>
> • The original output data
>
> • Warning messages if any (in <Message> element)
>
> • OAExceptions raised during the method invocation if any (in
>   <ErrorMessage> element)
>
> For more information about error messages, see Error Handling section,
> Building an OA Framework Application (the Basics) chapter, *Oracle
> Application Framework Developer's Guide*, available from My Oracle
> Support Knowledge Document 1315485.1.

Reviewing Service Data Object

A service data objects (SDO) defines a generic API for accessing and manipulating
structured data. It is designed to simplify and unify the way in which applications
handle data. The SDO API is independent of the actual data source. For example, SDO
can be used to access XML data or SQL data.

The data objects associated with business service objects include *service data objects*
(SDO) and *filter data objects* (FDO).

Service Data Object Information

The service data object details page is accessible directly from the Integration
Repository browse interface through the list of interface types. It is also accessible from
the `getDataList` and `processDataList` method details pages.

The following diagram illustrates the basic structure of the service object information

page and its connections to related pages.

Service Data Object Basic Structure
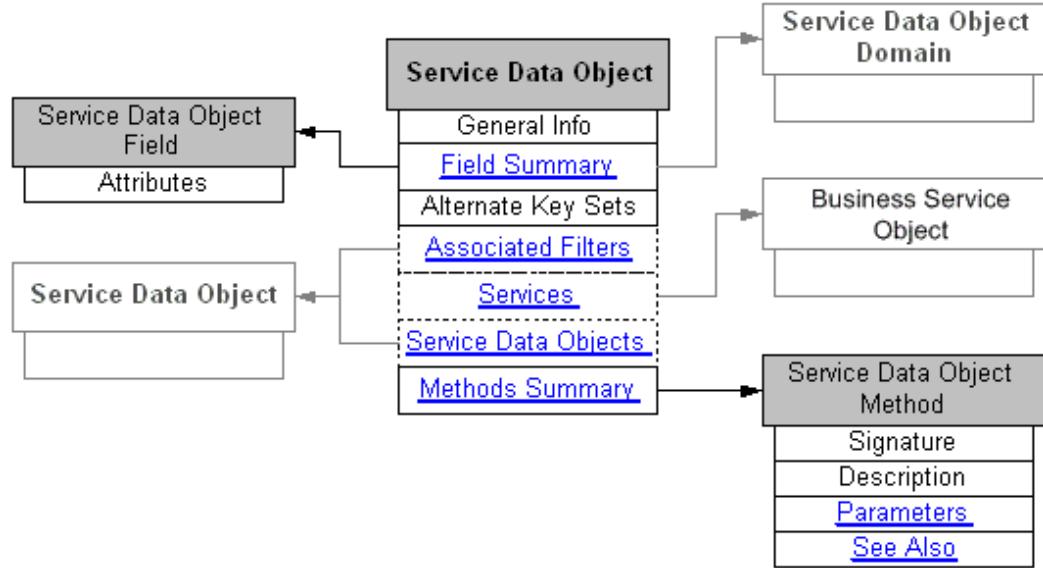
The general section of the service data object page displays common information, plus the data object class name, implementation name, and its associated XML schema.

Click the XML schema name link to view the schema document displayed in a separate window.

> **Note:** An XML schema describes the structure of an XML document with all input and output message definitions and data types.

The following regions also appear on the service object details page:

• **Fields**

  Click a link in the **Name** column to view the field's complete attributes.

  If the field type is a filter, you can click the link in the **Type** column to display the filter data object information page for that filter.

• **Alternate Key Sets**

  These are the key sets that can be used to identify an instance of this data object in the absence of the primary keys. These key sets are evaluated in the sequence specified.

• **Filters**

  The Filters region lists filters that can be used to filter data sources based on this service data object.

For example, data sources based on BaseDataSourceNameDomain service data object can be filtered by BaseDataSourceNameFilter filter data object.

Click the name of a filter (such as BaseDataSourceNameFilter) to display the selected filter data object details page.

- **Services**

  The Services region lists the services that directly use this service data object.

  Click the name of a service to display the information page for a business service object interface that uses this service object.

- **Service Data Objects**

  This Service Data Objects region lists the data objects that directly reference this service data object.

  Click the name of a service data object to display the object details page.

- **Methods**

  Click a link in the **Name** column to display the service object method details page.

Filter Data Object Information

The filer data object details page is accessible only from the `getDataList` method details page.

The following diagram illustrates the basic structure of the filter data object details page and its connections to related pages.

Filter Data Object Basic Structure
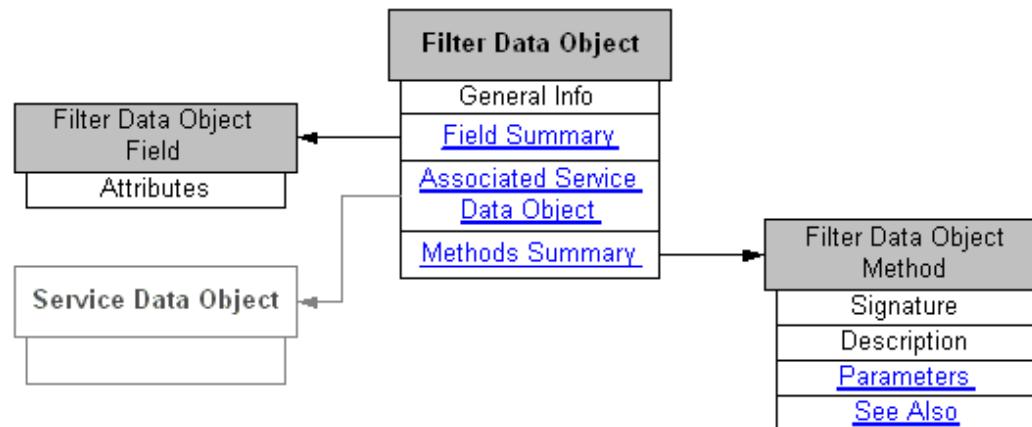
A filter data object is a type of service object. The general section of the filter data object page displays common information, plus the data object class name, implementation name, filter type, and its associated XML schema information.

> **Note:** There are two types of filter:
>
> - Expression Filter: Allows a client program to construct a simple or complex expression, including nested expressions.
>
> - Fixed Filter: Allows a simple list of attributes. This is used when the view object must do custom processing of filter attributes and the client program should not be allowed to build nested and complex filter expressions.
>
> For more information on filters, see *Oracle Application Framework Developer's Guide*, available from My Oracle Support Knowledge Document 1315485.1.

Click the XML schema name link to view the schema document displayed in a separate window.

The following regions also appear on the filter object details page:

- **Fields**

  Click a link in the **Name** column to view the field's complete attributes.

- **Associated Service Data Object**

  This region describes the associated service data object name that can be filtered by this selected filter data object.

  Click the name of a service data object to display the service data object details page.

- **Methods**

  Click a link in the **Name** column to display the object method details page.

Service Data Object Method Information

The type of information provided for filter data object methods is the same as for service data object methods. The data object method details page appears when you click a method name on the service data object details page or the filter data object details page.

In addition to a description, the following regions also appear on the service data object method details page:

- **Signature**

  This region describes the interface method, parameter type, value, and return type information.

- **Parameters**

This region contains descriptions of the parameters that are listed in the Signature region.

- **See Also**

  This region displays related service object methods. Click a related object method link to display the selected service object method details.

Concurrent Program Information

The following diagram illustrates the basic structure of the concurrent program information page.

Concurrent Program Basic Structure
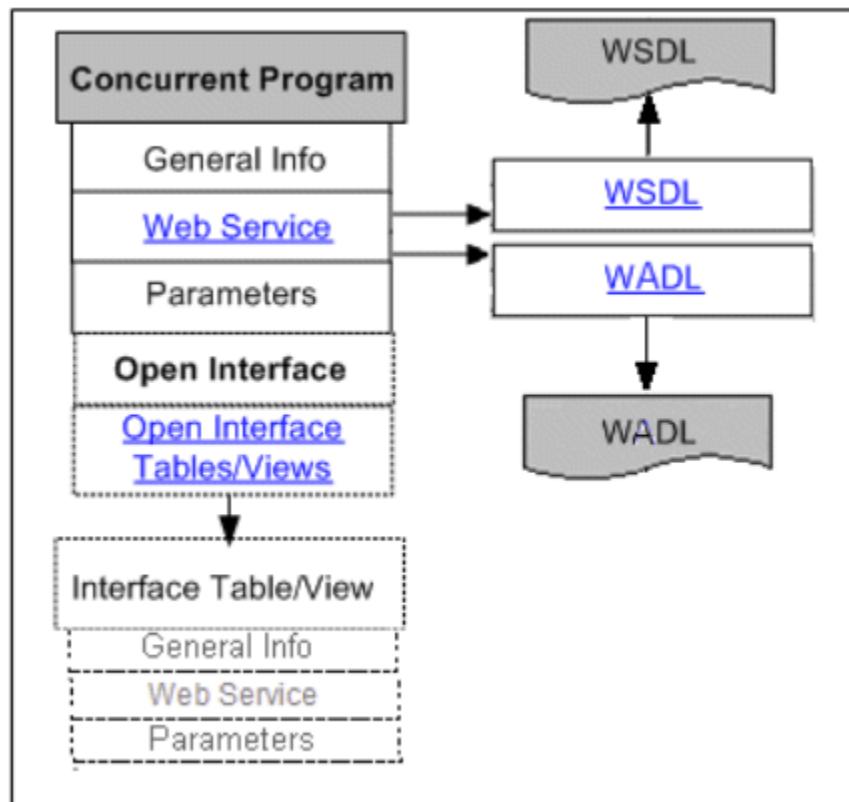
Concurrent programs can be exposed as both SOAP services and REST services. Therefore, the interface details page includes general section of a selected concurrent program, the Overview tab, the SOAP Web Service tab, and the REST Web Service tab.

Users who have the Integration Administrator role can find an additional Grants tab displayed in the interface details page. This tab allows the administrators to create and revoke security grants. For more information on how to manage security grants, see

Managing Grants, page 4-31.

For information on the general section, see Common information, page 4-1.

Concurrent Program Interface Details Page

The concurrent program details page contains the following information:

- **Overview Tab**

  This tab displays read-only information about the selected concurrent program. It includes full description, interface source information, as well as concurrent program specific parameters contained in the selected concurrent program.

  For more information on the interface source information, see Common Information on Interface Details, page 4-3.

- **SOAP Web Service Tab**

  This tab contains SOAP service information for the selected concurrent program.

This includes service status, WSDL description, interaction pattern, and authentication type information.

For more information about SOAP service, see Common Information on SOAP Web Services, page 4-6.

For information on viewing WSDL description, see Reviewing Web Service WSDL Source, page 4-23.

If a SOAP service has been successfully generated, the integration administrators can perform additional administrative tasks including deploying the generated service, regenerating the service, or clearing up the generated service artifact.

For more information on these administrative tasks, see Performing Administrative Activities for SOAP Web Services, page 4-12.

- **REST Web Service Tab**

    This tab contains REST service information for the selected concurrent program. This includes service alias, service status, WADL description, verb, service operation, and authentication type information.

    Note that the only supported HTTP verb for current programs is POST. For more information about REST service, see Common Information on REST Web Services, page 4-15.

    All REST services are secured by HTTP Basic Authentication or Security Token Authentication at HTTP or HTTPS transport level. For more information on REST service security, see Managing Web Service Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

    > **Note:** HTTPS is the recommended secure transport protocol while using the HTTP Basic Authentication security to authenticate user credentials (user name and password).

    If a REST service has been successfully deployed to an Oracle E-Business Suite WebLogic server, the integration administrator can undeploy the service to reset the service to its initial state - 'Not Deployed'.

    For more information on these administrative tasks, see Performing Administrative Activities for REST Web Services, page 4-20.

Method Details Page

The concurrent program method details page appears when you click a method name in the Overview tab, the SOAP Web Service tab, or the REST Web Service tab.

> **Note:** Oracle Integration Repository supports REST service enablement for Open Interface Tables and Views. If a concurrent program is linked to an open interface table or view, this concurrent program can be

viewed and displayed under the Open Interface category and can be available as a REST service.

For more information about the Open Interface integration type, see Open Interface Information, page 4-69.

In the Parameters region, the parameters used in the selected interface are listed in a table. It includes parameter name, parameter type, required or not, displayed or not, and description information.

Open Interface Information

Open Interface integrations are always implemented using concurrent programs; therefore, the Interface Type in the header region of the Open Interface details page is categorized as "Concurrent Program & Open Interface".

This type of interface stores the interface data, including active status, and whether it stores data inbound to Oracle E-Business Suite or outbound to another system.

The following diagram illustrates the basic structure of the Open Interface information and its connection to the related interface table or view information page.

Open Interface Basic Structure

Concurrent Program
General Info
Web Service
Parameters
Open Interface
Open Interface Tables/Views → Interface Table/View
General Info
Web Service → WADL
WADL
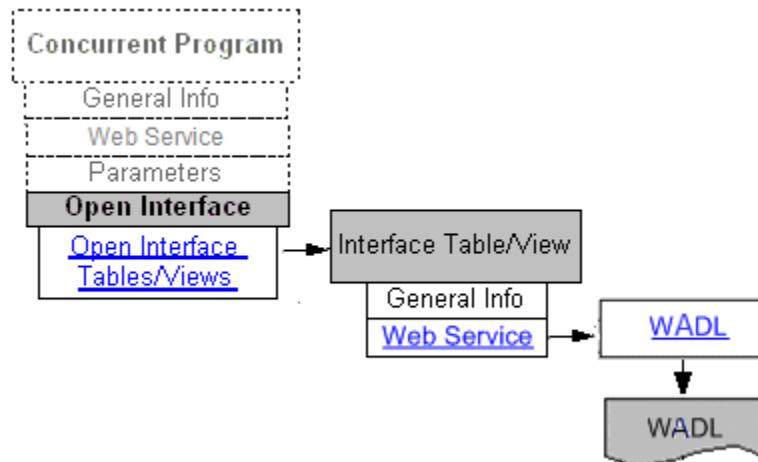
Open Interface integrations can be available as REST based web services. The interface details page includes general section of a selected open interface, the Overview tab and the REST Web Service tab.

Users who have the Integration Administrator role can find an additional Grants tab displayed in the interface details page. This tab allows the administrators to create and revoke security grants. For more information on how to manage security grants, see Managing Grants, page 4-31

For information on the general section, see Common information, page 4-1.

Open Interface Information

The Open Interface details page appears when you click an Open Interface name from the interface list table in the Integration Repository.

Open Interface Details Page

The Open Interface details page contains the following information:

- **Overview Tab**

  This tab displays read-only information about the selected open interface table. It includes full description, interface source information, as well as concurrent program specific parameters.

  For more information on the interface source information, see Common Information on Interface Details, page 4-3.

- **REST Web Service Tab**

  If the selected Open Interface is an open interface table, click the REST Web Service

tab to display the REST service information. This includes service alias, service status, WADL description, verb, service operation, and authentication type information.

Note that the supported HTTP verbs are determined by the direction of an open interface table or view.

- For an open interface table with `Inbound` direction, all four HTTP methods – GET, POST, PUT, and DELETE – can be shown as REST service operations if they were all selected during the service deployment.

- For an open interface table with `Outbound` direction, only the GET HTTP method can be shown.

- For the associated concurrent program (`SUBMIT_CP_<internal name of the associated concurrent program>`, such as `SUBMIT_CP_RAXMTR`) shown as the last entry in the table, only the POST HTTP method can be shown.

For more information about REST service, see Common Information on REST Web Services, page 4-15.

All REST services are secured by HTTP Basic Authentication or Security Token Authentication at HTTP or HTTPS transport level. For more information on REST service security, see Managing Web Service Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

> **Note:** HTTPS is the recommended secure transport protocol while using the HTTP Basic Authentication security to authenticate user credentials (user name and password).

If a REST service has been successfully deployed to an Oracle E-Business Suite WebLogic server, the integration administrator can undeploy the service to reset the service to its initial state - 'Not Deployed'.

For more information on these administrative tasks, see Performing Administrative Activities for REST Web Services, page 4-20.

Open Interface Table Details Page

Click a name link in the Open Interface Tables table to display the selected Open Interface Table details page containing interface table columns and their attributes, such as data type, data length, data precision, data scale, and data requirement.

Open Interface Table Details Page

Interface View Information

When you click the Open Interface view name from the interface list table in the Integration Repository, the Open Interface View details page appears.

Open Interface View Details Page

Similar to the interface details page for open interface table, the open interface view details page contains the following information:

- **Overview Tab**

  This tab displays read-only information about the selected open interface view. It includes full description, interface source information, and column information.

  For more information on the interface source information, see Common Information on Interface Details, page 4-3.

- **REST Web Service Tab**

  This REST Web Service tab contains REST service information for the selected interface view. This includes service alias, service status, WADL description, verb, service operation, and authentication type information.

  Unlike open interface table that can be exposed as a REST service operation with various HTTP methods, GET is the only supported method for open interface view.

  For more information about REST service, see Common Information on REST Web Services, page 4-15.

  All REST services are secured by HTTP Basic Authentication or Security Token Authentication at HTTP or HTTPS transport level. For more information on REST service security, see Managing Web Service Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

> **Note:** HTTPS is the recommended secure transport protocol while using the HTTP Basic Authentication security to authenticate user credentials (user name and password).

If a REST service has been successfully deployed to an Oracle E-Business Suite WebLogic server, the integration administrator can undeploy the service to reset the service to its initial state - 'Not Deployed'.

For more information on these administrative tasks, see Performing Administrative Activities for REST Web Services, page 4-20.

EDI Message Information

Oracle E-Business Suite Integrated SOA Gateway does not support service enablement for EDI messages; therefore, the EDI interface details page contains only general interface information.

EDI interface Details Page

The EDI Message interface details page displays the message direction in the Direction field:

- `Inbound` indicates that the interface is for receiving an incoming transaction or message into Oracle E-Business Suite.

- `Outbound` indicates that the interface is for sending an outgoing transaction or message to another system.

For more information on the general section, see Common Information, page 4-1.

Business Event Information

The general section of the Business Event displays common information, page 4-1 for business event.

Business Event Details Page

Only users who have the Integration Administrator role can perform the administrative tasks:

- Subscribe to an event

  If a selected business event is not subscribed, the **Subscribe** button appears in the event details page for the integration administrator. Clicking the **Subscribe** button lets you subscribe to the selected business event. Internally, an event subscription is automatically created for that event with WF_BPEL_QAGENT as Out Agent. Once the event subscription has been successfully created, a confirmation message appears on the Business Event interface detail page.

  To consume the business event message, you should register to dequeue the event from Advanced Queue WF_BPEL_Q. If a business event is enabled and if there is at least one subscriber registered to listen to the WF_BPEL_Q queue, then the event message will be enqueued in WF_EVENT_T structure to Advanced Queue WF_BPEL_Q.

  For more information on how to dequeue messages, see the *Oracle Streams Advanced Queuing User's Guide*.

- Unsubscribe the event

  The **Unsubscribe** button becomes available in the details page if the selected event has been subscribed. Clicking the **Unsubscribe** button lets you remove or unsubscribe the event.

For more information about subscribing to business events, see Subscribing to Business

Events, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

5

Working with Composite Services - BPEL

This chapter covers the following topics:

- Overview of Composite Services - BPEL
- Viewing Composite Services - BPEL
- Downloading Composite Services - BPEL

Overview of Composite Services - BPEL

Composite services use native interfaces as building blocks to construct the sequence of business flows. Basically, this interface type orchestrates the invocation sequence of web services into a meaningful end-to-end business process through a web service composition language BPEL. For example, use BPEL component in Oracle JDeveloper to integrate the Order to Receipt business process that contains sales order entry, item availability check, pack and ship, and invoice to Accounts Receivable subprocesses handled by various applications. This approach effectively tightens up the control of each individual process and makes the entire business flow more efficiently.

A composite service has its own WSDL definition and service endpoints allowing external web service clients to invoke the services at runtime.

In Oracle SOA Suite 11g and Oracle SOA Suite 12c, BPEL process is managed and deployed together with the associated SOA composite application. In Oracle SOA Suite 10g, it is developed and deployed as a separate component. Integration Repository displays 'Composite Services - BPEL' of Oracle SOA Suite 10g as catalogue in this release.

A Composite Service - BPEL in Oracle JDeveloper
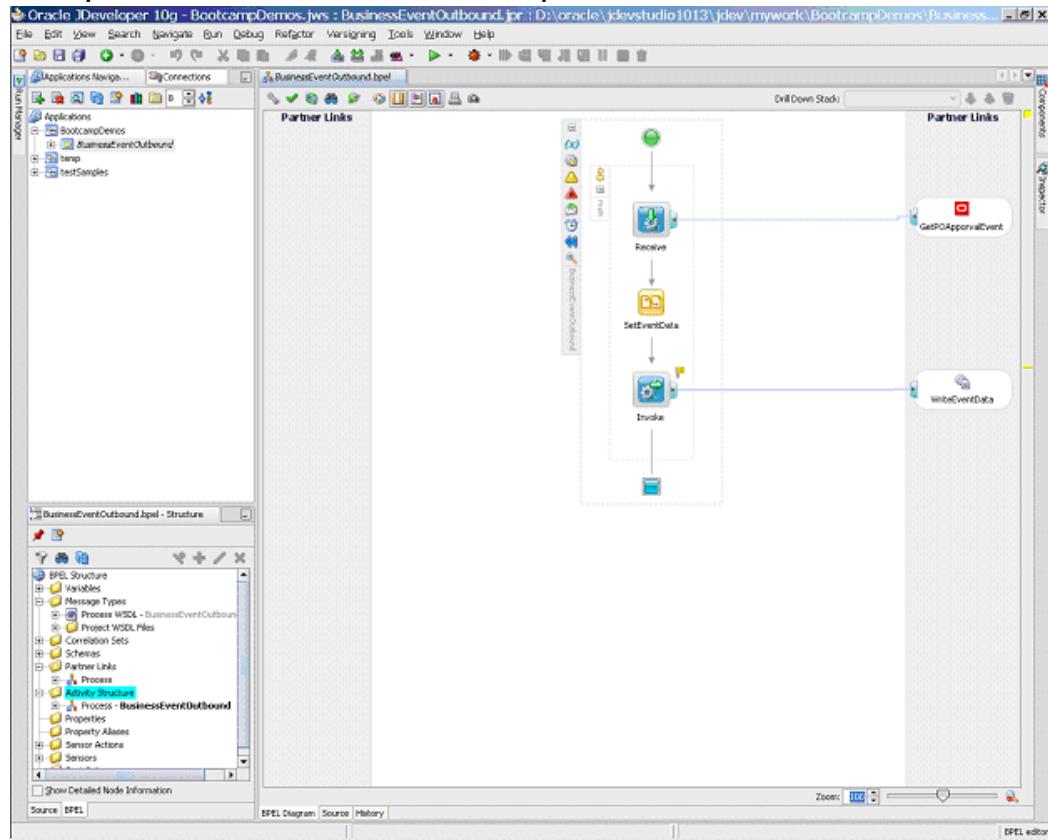
For information on how to create and administer composite services - BPEL, see the *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide* and the *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide* for details.

> **Additional Information:** Since composite interfaces can be designed and created in Oracle JDeveloper and Oracle Eclipse, based on the different creation methods, composite interfaces can have various composite types such as BPEL, ESB (enterprise service bus), or SCA (service component architecture) types. However, composite - BPEL is the only composite interface type supported in this release.

Viewing Composite Services - BPEL

Once annotated custom composite - BPEL definitions are uploaded to the Integration Repository, 'Composite - BPEL' option can be displayed from the repository and available to all users.

You can view composite interfaces by navigating to the Composite - BPEL interface type directly from the Oracle Integration Repository Browser window with View By

'Interface Type' or by performing a search in the Search page.

Click a composite - BPEL interface name link from the navigation tree or the search result to display the composite service - BPEL interface details page where you can view the interface name, description, BPEL file, abstract WSDL, and other annotated information.

The composite service - BPEL interface details page allows you to perform the following tasks in the BPEL Files region:

- View the composite service abstract WSDL file by clicking the **View Abstract WSDL** link.

  See: Reviewing Web Service WSDL Source, page 4-23.

- View the composite - BPEL file by clicking the **View BPEL File** link

  The BPEL code is displayed in a pop-up window containing major BPEL process components and activities included for the selected composite service.

Users who are granted the download composite service privilege can download the service by clicking **Download Service** in the interface details page. This downloads a corresponding composite service project file, such as a BPEL file, to a local directory. See: Downloading Composite Services - BPEL, page 5-4.

For information on how to upload composite - BPEL definitions to the Integration Repository, see Enabling Custom Integration Interface Process Flow, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

To view a composite service - BPEL:

1. Log in to Oracle E-Business Suite as a user who has the Integration Analyst role to access the Integration Repository through the Integration Repository responsibility. Select the Integration Repository link.

2. In the Integration Repository tab, select 'Interface Type' from the View By drop-down list.

3. Expand the Composite - BPEL interface type node to locate your desired composite service.

4. Click the composite - BPEL interface that you want to review to open the Composite Service - BPEL interface details page.

5. Click the **View Abstract WSDL** link to review the WSDL description.

6. Click the **View BPEL File** link to view the BPEL code.

Downloading Composite Services - BPEL

In addition to viewing composite service - BPEL details and reviewing a WSDL file, users who have the download privilege can download a composite - BPEL .JAR file to their local directories by clicking **Download Service** in the composite service details page.

> **Important:** In general, only integration developers and integration administrators can download the composite services - BPEL. However, users who are granted the download privilege through Integration Repository Download Composite Service permission set (FND_REP_DOWNLOAD_PERM_SET) can also perform the download action. Otherwise, **Download Service** may not appear in the details page by default.
>
> For information about how to grant the download composite service privilege, see Role-Based Access Control (RBAC) Security, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

To download a composite service - BPEL:

1. Log in to Oracle E-Business Suite as a user who has the Integration Analyst role and also has the download composite service privilege. Select the Integration Repository responsibility and then choose the Integration Repository link from the navigation menu.

2. In the Integration Repository tab, select 'Interface Type' from the View By drop-down list.

3. Expand the Composite - BPEL interface type node to locate a desired composite service.

4. Click the desired composite service to open the composite service - BPEL interface details page.

5. Click **Download Service** to download the selected composite file to a local directory.

6

Working with Custom Integration Interfaces

This chapter covers the following topics:

- Overview of Custom Integration Interfaces
- Searching Custom Integration Interfaces
- Viewing Custom Interface Details
- Performing Additional Activities for Custom Integration Interfaces

Overview of Custom Integration Interfaces

Oracle E-Business Suite Integrated SOA Gateway supports custom integration interfaces and allows them to be published along with Oracle seeded ones through the Oracle Integration Repository where they can be exposed to all users.

Integration developers create and annotate custom integration interfaces based on the Integration Repository annotation standards. Integration administrators use a standalone design-time tool to validate these annotated source files against the annotation standards. After validation, a loader file is generated and then uploaded to the Integration Repository through backend processing. These custom interfaces are displayed based on the interface types to which they belong and displayed together with Oracle seeded ones from the Integration Repository user interface.

Supported custom interface types include XML Gateway Map, Business Event, PL/SQL, Concurrent Program, Business Service Object, Java APIs, Java Bean Services, Application Module Services, and Composite Service - BPEL type.

> **Note:** Custom interface types of EDI, Open Interface Tables, Interface Views, and Java APIs for Forms interfaces are not supported in this release.
>
> Oracle Integration Repository currently does not support the creation of custom Product Family and custom Business Entity.

To easily distinguish annotated custom interface definitions from Oracle seeded interfaces, the Interface Source 'Custom' is used to categorize those custom integration interfaces in contrast to Interface Source 'Oracle' for Oracle interfaces.

For information on Integration Repository annotation standards, refer to:

• Integration Repository Annotation Standards, *Oracle E-Business Suite Integrated SOA Gateway Developer's Guide*

• *Oracle Application Framework Developer's Guide*, available from My Oracle Support Knowledge Document 1315485.1

The following topics explain more about how to search, view, and administer custom integration interfaces:

• Searching Custom Integration Interfaces, page 6-2

• Viewing Custom Integration Interfaces, page 6-4

• Performing Additional Web Service Activities for Custom Integration Interfaces, page 6-5

Searching Custom Integration Interfaces

Use the following ways to locate custom integration interfaces from the Integration Repository user interface:

• From the Interface List page, select 'Custom' from the Interface Source drop-down list along with a value for the Scope field to restrict the custom integration interfaces display.

Interface List Page with "Custom" Selected as the Interface Source

- From the Search page, click **Show More Search Options** to select 'Custom' from the Interface Source drop-down list along with any interface type, product family, or scope if needed as the search criteria.

Search Page with "Custom" Selected as the Interface Source

For example, select 'Custom' as the Interface Source and 'PL/SQL' as the Interface Type to locate the custom interfaces for PL/SQL type.

To view the custom integration interface details page:

1. Log in to Oracle E-Business Suite as a user who has the Integration Analyst role.

Select the Integration Repository responsibility and then choose the Integration Repository link from the navigation menu. This opens the repository browser.

2. Locate custom integration interface definitions using the following two ways:

   • **From the Interface List page**

     Select the following values:

     • Interface Source: Custom

     • Scope: Select an appropriate value

   • **From the Search page**

     1. Click **Show More Search Options** to open more search options.

     2. Select the following values:

        • Interface Source: Custom

        • Scope: Select an appropriate value

        • Product Family: Select an appropriate value

3. Click **Go** to run the search.

   Custom integration interfaces that match your search criteria are displayed in table format.

4. Click a desired custom interface name from the search result to view the interface details.

   The custom integration interface details page shows 'Custom' as the Interface Source value allowing you to differentiate it from an Oracle seeded one.

Viewing Custom Interface Details

After performing a search on custom integration interfaces, you can view the custom interface details by selecting a custom interface name from the search result table. Custom interface information including name, description, and other annotated information is displayed in the details page.

Note that all custom interface definitions have 'Custom' value displayed in the Interface Source field indicating that they are custom interfaces in contrast to Oracle seeded ones that have the Interface Source 'Oracle'.

Interface Details Page with Interface Source "Custom" Highlighted

Similar to native integration interfaces described in an earlier chapter, you may find additional interface or web service information in this page depending on the interface type you select. For example, if it is a custom PL/SQL API or concurrent program, you can find the Overview tab, SOAP Web Service tab, and REST Web Service tab in the interface details page.

Users who have the Integration Administrator role can perform administrative tasks during the service development life cycle, see Performing Additional Web Service Activities for Custom Integration Interfaces, page 6-5.

Performing Additional Activities for Custom Integration Interfaces

In addition to viewing custom integration interface details, users who have the Integration Administrator role can perform the following administrative tasks:

- **Deleting Custom Integration Interfaces**

  If a custom integration interface is not yet generated or deployed as a web service, users who have the Integration Administrator role can delete the custom interface

by clicking **Delete Interface** in the Overview tab of the custom interface details page.

Once a custom interface has been generated or deployed, it must be reset or undeployed to its initial state before it can be deleted.

See: Deleting Custom Integration Interfaces, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Managing Custom Web Service Lifecycle Activities**

  For Custom Interfaces with Support for SOAP Web Services

  - Managing Security Grants for Custom SOAP Services

    To use custom interfaces and services, users who have the Integration Administrator role need to grant the custom interfaces access privileges to a user, user group, or all users. Additionally, the administrators can revoke the grants that have been created earlier if needed.

    See: Managing Security Grants for SOAP Web Services Only, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  - Generating Custom SOAP Web Services

    Users who have the Integration Administrator role or the Integration Developer role can transform custom integration definitions into SOAP services with desired interaction patterns.

    > **Note:** Integration developers have the privilege to generate web services, but they cannot deploy or undeploy services.

    See: Generating SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  - Deploying Custom SOAP Web Services

    If a custom SOAP service has been successfully generated, users who have the Integration Administrator role can deploy the generated custom SOAP service with an appropriate authentication type.

  - Undeploying Custom Web Services

    If a custom SOAP service has been successfully deployed, the integration administrators can click **Undeploy** to undeploy the custom service that has been deployed earlier.

    See: Deploying and Undeploying SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  - Resetting Custom SOAP Web Services

Once a custom service has been successfully generated or deployed, the integration administrators can clear up existing service artifacts if needed by clicking **Reset**.

See: Resetting SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- Retiring Custom SOAP Web Services

  When a custom service has been deployed to Oracle SOA Suite with active state, the integration administrators can retire the deployed custom service and ensure that all current running requests are completed.

  See: Retiring SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- Activating Custom SOAP Web Services

  For a retired custom SOAP service in Oracle SOA Suite, the integration administrators can activate the retired service by clicking **Activate** in the interface details page. This lets a retired custom service become active again.

  See: Activating SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- Viewing Log Messages

  To effectively troubleshoot any issues or exceptions encountered at each stage of service generation and deployment life cycle, the integration administrators can view design-time logs recorded for a custom SOAP service through the SOAP Web Service tab of the interface details page, and runtime logs through Service Monitor for an interface if the design-time log and runtime log are enabled for the SOAP service type respectively.

  For information on how to view design-time logs for SOAP services, see Viewing Design-Time Logs for SOAP Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  For information on how to view runtime logs, see Viewing Service Processing Logs, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

For detailed information about administrative tasks, see Administering SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

For Custom Interfaces with Support for REST Web Services

> **Note:** Supported interfaces for custom REST services are PL/SQL APIs, Concurrent Programs, Business Service Objects, Java Bean Services, and Application Module Services. Although Open Interface Tables and Views can be exposed as REST services,

custom interfaces of this interface type are not supported.

- Managing Security Grants for Custom REST Services

    Similar to managing the grants for custom SOAP services, to let Oracle E-Business Suite users have the access permissions to the custom interfaces that can be exposed as REST services, users who have the Integration Administrator role need to create new security grants for these custom interfaces and services.

    The administrators can revoke existing grants to remove the access privileges from the users who have been granted the custom interfaces access.

    See: Managing Grants for Interfaces with Support for SOAP and REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- Deploying Custom REST Web Services

    Before deploying a custom interface as a REST service, the administrator must specify service alias for the selected interface in the Service Operations table and ensure that at least one authentication type is selected in the REST Service Security region. If the selected custom interface type is PL/SQL, Business Service Object, Java Bean Services, or Application Module Services, the administrator also needs to specify HTTP verbs for desired methods contained in the selected interface before deployment.

    See: Deploying REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- Undeploying Custom REST Web Services

    If a REST service has been successfully deployed, the integration administrators can click **Undeploy** to undeploy the REST service if necessary.

    See: Undeploying REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- Viewing Log Messages

    Similar to viewing log messages for custom SOAP services, the integration administrators can view log messages recorded specifically for the REST service deployment activities through the REST Web Service tab and the runtime logs through Service Monitor for an interface if the design-time log and runtime log are enabled for the REST service type respectively.

    For information on how to view design-time logs for REST services, see Viewing Design-Time Logs for REST Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

    For information on how to view runtime logs, see Viewing Service Processing Logs, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

For detailed information about administrative tasks, see Administering REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Subscribing to Custom Business Events**

  The integration administrators can subscribe to a custom business event by clicking **Subscribe** in the business event interface details page. This creates subscription for the selected custom event.

  See: Subscribing to Business Events, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Viewing and Downloading Custom Composite Services - BPEL**

  - Viewing Custom Composite Services - BPEL

    To view a custom composite service - BPEL, perform a search from the Search page and click **Show More Search Options** to display more search fields. Enter the following values in the Search page along with product family, scope, or any other values if needed to locate desired custom composite interfaces:

    - Interface Source: Custom

    - Interface Type: Composite

    See: Viewing Composite Services - BPEL, page 5-2.

  - Downloading Custom Composite Services - BPEL

    Similar to downloading composite services, users who have the download privilege can download a custom composite - BPEL .JAR file to their local directories by clicking **Download Service** in the interface details page.

    See: Downloading Composite Services - BPEL, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

Glossary

Agent

A named point of communication within a system.

Agent Listener

A type of service component that processes event messages on inbound agents.

Asynchronous Operation

Unlike the synchronous service processing to obtain the result immediately, an asynchronous operation may require a significant amount of time to process a request.

However, the client that invoked the Oracle E-Business Suite service can continue with other processing in the meantime rather than wait for the response.

Business Event

See Event.

Callback Pattern

Callback pattern is an important communication method in asynchronous services. An asynchronous callback means that a request is made to the service provider and a response (callback) is sent back to the requester when it is ready. This pattern can be used in conjunction with acknowledgement to recognize the receipt of a request sent by a requester.

Concurrent Manager

An Oracle E-Business Suite component that manages the queuing of requests and the operation of concurrent programs.

Event

An occurrence in an internet or intranet application or program that might be significant to other objects in a system or to external agents.

Event Activity

A business event modelled as an activity so that it can be included in a workflow

process.

Event Data

A set of additional details describing an event. The event data can be structured as an XML document. Together, the event name, event key, and event data fully communicate what occurred in the event.

Event Key

A string that uniquely identifies an instance of an event. Together, the event name, event key, and event data fully communicate what occurred in the event.

Event Message

A standard Workflow structure for communicating business events, defined by the datatype `WF_EVENT_T`. The event message contains the event data as well as several header properties, including the event name, event key, addressing attributes, and error information.

Event Subscription

A registration indicating that a particular event is significant to a system and specifying the processing to perform when the triggering event occurs. Subscription processing can include calling custom code, sending the event message to a workflow process, or sending the event message to an agent.

Function

A PL/SQL stored procedure that can define business rules, perform automated tasks within an application, or retrieve application information. The stored procedure accepts standard arguments and returns a completion result.

Integration Repository

Oracle Integration Repository is the key component or user interface for Oracle E-Business Suite Integrated SOA Gateway. This centralized repository stores native packaged integration interface definitions and composite services.

Integration Repository Parser

It is a standalone design-time tool used by the integration administrator to validate annotated custom interface definitions against the annotation standards and generate an Integration Repository loader file (iLDT). This generated iLDT file can be uploaded to Integration Repository where custom interfaces can be exposed to all users.

Interface Type

Integration interfaces are grouped into different interface types.

JSON

JSON (JavaScript Object Notation) is a text-based open standard designed for human-readable data interchange. The JSON format is often used with REST services to transmit structured data between a server and web application, serving as an alternative to XML.

Loose Coupling

Loose coupling describes a resilient relationship between two or more systems or organizations with some kind of exchange relationship. Each end of the transaction makes its requirements explicit and makes few assumptions about the other end.

Lookup Code

An internal name of a value defined in a lookup type.

Lookup Type

A predefined list of values. Each value in a lookup type has an internal and a display name.

Message

The information that is sent by a notification activity. A message must be defined before it can be associated with a notification activity. A message contains a subject, a priority, a body, and possibly one or more message attributes.

Message Attribute

A variable that you define for a particular message to either provide information or prompt for a response when the message is sent in a notification. You can use a predefine item type attribute as a message attribute. Defined as a 'Send' source, a message attribute gets replaced with a runtime value when the message is sent. Defined as a 'Respond' source, a message attribute prompts a user for a response when the message is sent.

Notification

An instance of a message delivered to a user.

Notification Worklist

A web page that you can access to query and respond to workflow notifications.

Operation

An abstract description of an action supported by a service.

Port

A port defines an individual endpoint by specifying a single address for a binding.

Port Type

A port type is a named set of abstract operations and abstract messages involved.

Process

A set of activities that need to be performed to accomplish a business goal.

REST

Representational State Transfer (REST) is an architecture principle in which the web services are viewed as resources and can be uniquely identified by their URLs. The key characteristic of a REST service is the explicit use of HTTP methods (GET, POST, PUT, and DELETE) to denote the invocation of different operations.

SAML Token (Sender-Vouches)

This type of security model authenticates web services relying on sending a user name only through Security Assertion Markup Language (SAML) assertion.

SAML is an XML-based standard for exchanging authentication and authorization data between security domains, that is, between an identity provider and a service provider. SAML Token uses a sender-vouches method to establish the correspondence between a SOAP message and the SAML assertions added to the SOAP message.

See Username Token.

Service

A service is a collection of related endpoints.

Service Component

An instance of a Java program which has been defined according to the Generic Service Component Framework standards so that it can be managed through this framework.

Service Monitor

It is the monitoring and auditing tool in Oracle E-Business Suite allowing administrators to monitor inbound SOAP and REST service invocation messages.

It is known as SOA Monitor in earlier releases.

Service Invocation Monitor

Service Invocation Monitor is a monitoring and auditing tool allowing administrators to monitor outbound SOAP and REST service invocations from Oracle E-Business Suite through Service Invocation Framework.

SOA

Service-oriented Architecture (SOA) is an architecture to achieve loose coupling among interacting software components and enable seamless and standards-based integration

in a heterogeneous IT ecosystem.

SOAP

Simple Object Access Protocol (SOAP) is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment. It uses XML technologies to define an extensible messaging framework providing a message construct that can be exchanged over a variety of underlying protocols.

Subscription

See Event Subscription.

Synchronous Operation

Synchronous operation provides an immediate response to a query. In this situation, the client connection remains open from the time the request is submitted to the server. The client will wait until the server sends back the response message.

Username Token

A type of security model based on user name and password to authenticate SOAP requests at runtime.

See SAML Token (Sender-Vouches).

WADL

Web Application Description Language (WADL) is designed to provide a machine-processable description of HTTP-based web applications. It models the resources provided by a service and the relationships between them.

Web Services

A web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in WSDL. Other systems interact with the web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other web-related standards.

Workflow Engine

The Oracle Workflow component that implements a workflow process definition. The Workflow Engine manages the state of all activities for an item, automatically processes functions and sends notifications, maintains a history of completed activities, and detects error conditions and starts error processes. The Workflow Engine is implemented in server PL/SQL and activated when a call to an engine API is made.

WSDL

Web Services Description Language (WSDL) is an XML format for describing network services as a set of endpoints operating on messages containing either document-

oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint.

WS-Addressing

WS-Addressing is a way of describing the address of the recipient (and sender) of a message, inside the SOAP message itself.

WS-Security

WS-Security defines how to use XML Signature in SOAP to secure message exchanges, as an alternative or extension to using HTTPS to secure the channel.

XML

XML (Extensible Markup Language) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Index