Contents

Send Us Your Comments

Preface

1 Introduction to Oracle E-Business Suite Integrated SOA Gateway

   Oracle E-Business Suite Integrated SOA Gateway Overview .............................................. 1-1

2 Understanding Service Enablement

   Service Enablement Overview .......................................................................................... 2-1
   Common Terms Used in the Web Services ........................................................................ 2-2
   Service-Oriented Architecture in Oracle E-Business Suite ............................................. 2-3
   Service Enablement in Oracle E-Business Suite .............................................................. 2-5

3 Navigating Through Oracle Integration Repository

   Oracle Integration Repository Overview ............................................................................ 3-1
   Getting Started ................................................................................................................ 3-3
   Discovering and Reviewing Interfaces ............................................................................. 3-6
      Browsing the Integration Interfaces ............................................................................... 3-6
      Searching for an Integration Interface ........................................................................... 3-8
      Interface Types .............................................................................................................. 3-12
      Integration Standards ................................................................................................... 3-17

4 Working with Native Services and Integration Interfaces

   Common Information ........................................................................................................ 4-1
      Common Information on Interface Details .................................................................... 4-3
      Common Information on SOAP Web Services ............................................................. 4-8
5 Working With Composite Services

Overview of Composite Services .............................................. 5-1
Viewing Composite Services .................................................. 5-2
Downloading Composite Services ........................................... 5-3

6 Working with Custom Integration Interfaces and Services

Overview of Custom Integration Interfaces ............................... 6-1
Searching Custom Integration Interfaces ................................. 6-2
Viewing Custom Integration Interfaces ................................. 6-2
Performing Additional Web Service Activities for Custom Integration Interfaces ...................................... 6-6

Glossary

Index
Send Us Your Comments

Part No. E12064-10

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


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.

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.

See Related Information Sources on page viii for more Oracle E-Business Suite product information.

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 Services and Integration Interfaces
5 Working With Composite Services
6 Working with Custom Integration Interfaces and Services
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 12.1 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.


- **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
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, 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 the installation and configuration choices that may be available.

Oracle E-Business Suite Developer’s Guide

This guide contains the coding standards followed by the Oracle E-Business Suite development staff. It describes the Oracle Application Object Library components needed to implement the Oracle E-Business Suite user interface described in the Oracle E-Business Suite User Interface Standards for Forms-Based Products. It provides information to help you build your custom Oracle Forms Developer forms so that they integrate with Oracle E-Business Suite. In addition, this guide has information for customizations in features such as concurrent programs, flexfields, messages, and logging.

Oracle Fusion Middleware Adapter for Oracle Applications User’s Guide

This guide covers the use of Adapter for Oracle Applications (also known as Oracle E-Business Suite Adapter) in developing integrations between Oracle E-Business Suite and trading partners.

This book covers the use of Oracle E-Business Suite Adapter (formerly known as "Adapter for Oracle Applications" in Oracle Applications Server 10g or Oracle Fusion Middleware 11g releases) in developing integrations between Oracle E-Business Suite and trading partners.

This book is available in the Oracle Application Server 10g Documentation Library and Oracle Fusion Middleware 11g Documentation Library.

Oracle E-Business Suite System Administrator’s Guide Documentation Set

This documentation set provides planning and reference information for the Oracle E-Business Suite System Administrator. Oracle E-Business Suite System Administrator’s Guide - Configuration contains information on system configuration steps, including defining concurrent programs and managers, enabling Oracle Applications Manager features, and setting up printers and online help. Oracle E-Business Suite System Administrator’s Guide - Maintenance provides information for frequent tasks such as monitoring your system with Oracle Applications Manager, administering Oracle E-Business Suite Secure Enterprise Search, managing concurrent managers and reports, using diagnostic utilities including logging, managing profile options, and using alerts. Oracle E-Business Suite System Administrator’s Guide - Security describes User
Management, data security, function security, auditing, and security configurations.

**Oracle E-Business Suite User’s Guide**

This guide explains how to navigate, enter data, query, and run reports using the user interface (UI) of Oracle E-Business Suite. This guide also includes information on setting user profiles, as well as running and reviewing concurrent requests.

**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 repository 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 invoke web services from Oracle E-Business Suite by employing the Oracle Workflow Business Event System, and how to manage web service security, configure logs, and monitor SOAP messages.

**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 Report Manager User’s Guide**

Oracle Report Manager is an online report distribution system that provides a secure and centralized location to produce and manage point-in-time reports. Oracle Report Manager users can be either report producers or report consumers. Use this guide for information on setting up and using Oracle Report Manager.

**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.
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 web services.

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

SOAP-based services are described in WSDLs and are deployed to the application server for service consumption. REST services described in WADLs are 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 contains the following features:

- 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 interaction pattern for both SOAP-based and REST-based web services

  Note: In this release, only PL/SQL APIs, Concurrent Programs, and Business Service Objects 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.

- 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 execute administrative functions

- Provide centralized, user-friendly user interface for logging configuration

- Audit and monitor Oracle E-Business Suite service operations from native SOA Monitor

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

**Business Process Scenario**

Oracle E-Business Suite Integrated SOA Gateway provides a seamless integration between various applications. Take the most common business process such as Order-to-Receipt as an example to further explain how discrete web services can be orchestrated into a standards-based manner and more meaningful end-to-end business flow.
Order-to-Receipt Business Flow Between Applications

The packaged application is used to capture the order. The legacy application is used to fulfill (pick and ship) the order. Oracle E-Business Suite is used to invoice the customer.

- Sales Order Entry: Packaged Application
- Item Availability Check: Legacy Application
- Pick, Pack and Ship: Legacy Application
- Invoicing and A/R: Oracle E-Business Suite

A complete Order-to-Receipt business flow may require to integrate with each of the above applications at different points. With Oracle E-Business Suite Integrated SOA Gateway, the public integration interfaces of E-Business Suite can be exposed as standard web services.

Each individual business process mentioned here managed by packaged application, legacy application, and Oracle E-Business Suite can be orchestrated using Oracle BPEL Process Manager (PM) to streamline the Order-to-Receipt business process.
Oracle E-Business Suite Integrated SOA Gateway Architecture Overview

Oracle E-Business Suite Integrated SOA Gateway employs essential key components that enable service integration at design time and runtime, and ease the service management throughout the entire service integration and deployment life cycle.

The seamless integration between each component forms the Oracle E-Business Suite Integrated SOA Gateway architecture.

The following diagram illustrates the integration architecture flow between each
All the native packaged public integration interfaces are published in the Oracle Integration Repository by default. Integration repository administrators can then transform these native integration interfaces into web services through service generator. Service loader uploads service artifacts to Oracle Integration Repository. Service deployer deploys service artifacts from the Integration Repository to the application server where services can be exposed to customers through service provider.

Service provider identifies and processes inbound SOAP requests from service consumers, reinforces function security and web service security, as well as passes all SOAP request and response messages to SOA Monitor (if the monitoring feature is enabled) for further monitoring SOAP messages to ensure the seamless service invocations throughout the entire service life cycle.

For composite services, system integration developers orchestrate composite services using Oracle JDeveloper. Service loader then uploads these service artifacts to Oracle Integration Repository. Users granted with the Download Composite Service privilege can further download the BPEL files to their local directories. Integration repository developers can open the downloaded BPEL files in Oracle JDeveloper, modify and deploy them if needed. Oracle BPEL Process Manager (BPEL PM) or 3rd party J2EE BPEL PM will then pick up deployed composite services which can be invoked from the Oracle E-Business Suite.

**Note:** Unlike native services that they are deployed directly from the Oracle Integration Repository user interfaces, composite services are
typically not deployed within Oracle E-Business Suite like those of other service enabled interface types. For example, a composite service - BPEL type can be deployed to a BPEL server in Oracle SOA Suite BPEL PM (Process Manager) or a third party BPEL PM in a J2EE environment. This deployed composite service - BPEL project can interact with Oracle E-Business Suite and update the data if necessary.

**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 System Integration Developer role can create custom interfaces, and annotate custom interface's definitions. Users who have the Integration Repository 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 Repository Administrator role can generate SOAP services, and deploy the generated services by attaching an appropriate security policy. For interfaces can be exposed as REST services, the administrators can deploy and undeploy REST services.

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

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

<table>
<thead>
<tr>
<th>Interface Type</th>
<th>Supported HTTP Method(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PL/SQL API</td>
<td>POST only</td>
</tr>
<tr>
<td>Concurrent Program</td>
<td>POST only</td>
</tr>
<tr>
<td>Business Service Object</td>
<td>POST and GET</td>
</tr>
<tr>
<td>Interface Type</td>
<td>Supported HTTP Method(s)</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Java Bean Service</td>
<td>POST and GET</td>
</tr>
<tr>
<td>Application Module Service</td>
<td>POST and GET</td>
</tr>
<tr>
<td>Open Interface Table (Inbound)</td>
<td>POST, GET, PUT, and DELETE</td>
</tr>
<tr>
<td>Open Interface Table (Outbound)</td>
<td>GET only</td>
</tr>
<tr>
<td>Open Interface View</td>
<td>GET only</td>
</tr>
</tbody>
</table>

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 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-Oriented Architecture in Oracle E-Business Suite**

To allow pre-built and reusable business services available to customers and let customers dynamically interact between applications, Oracle E-Business Suite Integrated SOA Gateway, building on the principle of service-oriented architecture,
allows service enablement within the Oracle E-Business Suite. As a result, integration interface definitions that have been used internally within an organization are now Web available.

The following diagram illustrates the essential SOA components in enabling services within Oracle E-Business Suite:

![Service-Oriented Architecture in Oracle E-Business Suite](image)

In this diagram, SOA Provider is the service provider used in enabling services. Oracle Integration Repository plays a role as a service broker. The Web service invocation framework (WSIF) serves as a service consumer to issue a request through the invocation of a Web services from Oracle E-Business Suite.

**SOA Provider**

SOA Provider is an enhanced service provider particularly in supporting additional interface types for service enablement.

*Note:* In Release 12.0, Oracle E-Business Suite is service partially enabled using Web Service Provider to enable XML Gateway Map and Business Service Object (formerly known as Service Bean) interface types. For backward compatibility, Oracle E-Business Suite Integrated SOA Gateway continues to support the Release 12.0 based Web Service Provider service enablement, plus additional interface types using SOA Provider to enable services.

At run time, SOA Provider references integration services and data from Oracle Integration Repository in processing inbound SOAP request messages that invoke Web services and sends the SOAP response out.

**Web Service Invocation Framework**
To invoke integration services from Oracle E-Business Suite, Oracle E-Business Suite Integrated SOA Gateway uses service invocation framework, leveraging Oracle Workflow Java Business Event System (JBES) and a seeded Java rule function, to allow any WSDL-described service to be invoked.


Oracle Integration Repository

Oracle Integration Repository, 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 provides a comprehensive, consistent browsing view of the interface mechanism which lets you easily discover and search on the business interface from the catalog.

Oracle E-Business Suite Integrated SOA Gateway leverages Oracle Integration Repository to provide the capabilities of Web service generation and deployment, as well as service life cycle management. Application users can browse these interface definitions and services through Oracle Integration Repository and view the interface details. Users who have the Integration Repository Administrator role can perform design-time operations and monitor the Web services.

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:
1. An integration repository administrator or a system integration developer transforms the integration interface definitions resided in Oracle Integration Repository into SOAP-based services described in WSDLs.

   See: Reviewing Web Service WSDL Source, page 4-16.

2. An integration repository administrator deploys the SOAP services. SOAP services are deployed to an Oracle E-Business Suite managed server.

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

4. If the selected interfaces can be exposed as REST services, an integration repository administrator can deploy the REST services using a user action called 'Deploy'. REST services described in WADLs are deployed to an Oracle E-Business Suite managed server.

5. At runtime, REST services commonly used for mobile applications can create or update resources in Oracle E-Business Suite.

For more service enablement on SOAP services, see SOAP Service Enablement, page 2-7.

For more service enablement on REST services, see REST Service Enablement, page 2-8.

In addition to transforming interface definitions into web services and deploying them, integration repository administrators can access the SOA Monitor user interface to monitor and manage all SOAP messages in and out from the SOA Provider (if the SOA
monitoring feature is enabled). This allows any operation error if occurred during the message exchanges to be identified and audited. The administrators can search and view SOAP request and response message details, and take necessary actions if needed to expedite the interaction between services and consumers. For more information on how to use SOA Monitor, see Monitoring and Managing SOAP Messages Using SOA Monitor, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

**SOAP Service Enablement**

**WSDL-based SOAP Services**

In a web service environment, SOAP service provides a standard way of structuring XML documents and acts as a building block for web service communication. For example, 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 repository administrator or a system integration developer can transform the interface definitions resided in Oracle Integration Repository into SOAP-based services described in WSDLs.

**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 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 execution 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.

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.

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 service is the explicit use of HTTP methods to denote the invocation of different operations. It is another style of web services that serves as a simpler alternative to SOAP services allowing 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.

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 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.
Navigating Through Oracle Integration Repository

This chapter covers the following topics:

- Oracle Integration Repository Overview
- Getting Started
- 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 the following integration interface types which can be categorized as follows:

- Service enabled
- **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 (Service Beans)**
- **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.

- **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.

- **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**
Navigating Through Oracle Integration Repository

- Business Event
- XML Gateway Map (outbound)
- Composite services - BPEL
- Non-service enabled public interfaces
- EDI Interface

Detailed information on each interface type, see Interface Types, page 3-12.

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 to help you find the data you are looking for from the repository.
- 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 such as generate and deploy Web services.
- It supports custom integration interfaces.

Getting Started

Accessing Oracle Integration Repository

You can access the repository like any other Oracle E-Business Suite application, provided that you are logged in as a user with sufficient permissions. From the Navigator menu, select the Integrated SOA Gateway responsibility, then click the Integration Repository link. Oracle Integration Repository appears.

Oracle E-Business Suite Integrated SOA Gateway allows the following three roles to access the Integration Repository user interfaces and perform necessary tasks:

- System Integration Analyst
- System Integration Developer

- Integration Repository Administrator

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

<table>
<thead>
<tr>
<th>Privileges</th>
<th>System Integration Analyst</th>
<th>System Integration Developer</th>
<th>Integration Repository Administrator</th>
</tr>
</thead>
<tbody>
<tr>
<td>View Public Interfaces</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>View Private/Internal Interfaces</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Generate/Regenerate SOAP Web Services</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Deploy/Undeploy SOAP Web Services</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Deploy/Undeploy REST Web Services</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Subscribe to Business Events</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Create Grants</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>View Grants</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Download Composite Service</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>(Configurable)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**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 Repository Administrator role. That user becomes an integration repository administrator and has privileges to perform administrative tasks.

System 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 4-8 and Common Information on REST Web Services, page 4-11.

For information on how to manage security grants, see Managing Grants, page 4-23.

**Accessing the Administration Link to Perform Additional Administrative Tasks**

After logging in to Oracle E-Business Suite with the Integrated SOA Gateway responsibility, users who have the Integration Repository Administrator role can find the Administration link in addition to the Integration Repository link from the Navigator menu. This Administration link is specifically for the administrator to perform additional administrative tasks outside the Integration Repository user interface. Expand the Administration link to display:

- **SOA Monitor**: This link allows the administrators to access the SOA Monitor user interface where the administrators can monitor and audit all SOAP messages in and out through SOA Provider and view the message details.


- **Log**: This link allows the administrators to access the centralized Log configuration user interface where the administrators can enable and configure log setups.

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

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

**Using Oracle Integration Repository**

Oracle Integration Repository has two main user interfaces: The Browse interface, page 3-6 (the default) and the Search interface, page 3-8.

Following are links to some of the commonly requested information about using Oracle Integration Repository:

- Included interface types, page 3-12

- Integration standards, page 3-17
Discovering and Reviewing Interfaces

Browsing the Integration Interfaces

The Browse interface appears by default when you invoke Oracle Integration Repository. You can also access it by clicking the Browse button 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 3-12.

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

  See: Integration Standards, page 3-17.

  If you don’t have this information, you’ll find it more effective to conduct a search, page 3-8.
You browse the interfaces 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 on the list.

**Browsing by Product Family**

The **Product Family** view is organized as follows: **Product Family > Product > Business Entity**.

For example, **Financials > Payables > Payables Invoice**.

Select a business entity, page 3-6 to view the interfaces that comprise it. Note that a business entity can include multiple interfaces of different types owned by different products. For example, the business entity "Payables Invoice" includes the following:

- Create Credit Card Issue Invoice open interface from Internet Expenses
- Invoice open interface from Payables
- Invoice Notification XML Message from Supply Chain Trading Connector

**Browsing by Interface Type**

The **Interface Type** view is organized as follows: **Interface Type > Product Family > Product**.
For example, Web Service > Financials > 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** > **Product Family** > **Product**.

For example, **OAG7.2** > **Financials** > **Payables** > **Process 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 the **Search** button anywhere in Oracle Integration Repository to access the main Search page.

**Oracle Integration Repository Search Page**

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

**Note:** Before entering search criteria in the Search page, you will find the default value ‘All’ automatically displayed in the Product Family, Product, Interface Source, and Interface Type fields. This allows a search to be executed appropriately if you do not make further selections from the drop-down lists.
Additionally, the same default value ‘All’ can also be found in the Category, Status, 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 4-4
  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 3-12
  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, Concurrent Program, XML Gateway, Open Interface Tables, Open Interface Views, Business Service Object, Business event, EDI Interface, 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**
  Used to qualify product-specific features or to categorize a subtype of an interface. You can 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 with this interface source type.

• **Custom** - This indicates annotated custom integration interfaces. Custom integration interfaces are displayed along with Oracle interfaces from the browser tree.

• **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 Status**
If an integration interface is exposed or generated as a Web service, then the Web service can be further deployed from Oracle Integration Repository to the application server.

Use the Web Service Status field to search by different stages of Web services during the service generation and deployment life cycle.

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

- **All (default)** - This displays all interfaces regardless of the interface types whether they are service enabled or not.
- **Not Generated** - This displays all service-enabled interfaces that do not have Web service generated.
- **Generated** - This displays all interfaces that have Web services generated, but have not yet been deployed.
- **Deployed** - This displays all interfaces that have Web services generated and deployed.

For more information, see Common Information on Web Services, page 4-8.

**Scope**

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

- **All (default)** - All integration interfaces regardless of public, internal, or private interfaces.
- **Public** - These interfaces can be used by anyone.
- **Internal To Oracle** - These interfaces 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.

This type of interface can only be accessed by users who have the System Integration Developer role and the Integration Repository Administrator role.

- **Private To Application** - These interfaces are available for business integration only within the application itself. They will not be used by any other applications outside the application that the interface belongs to.

For example, if an interface with this 'Private to Application' type belongs to Purchasing application, then it will not be used by any other applications within Oracle E-Business Suite but Purchasing.
This type of interface can only be accessed by users who have the System Integration Developer role and the Integration Repository Administrator role.

- **Standard and Standard Specification**
  
  For more information, see Integration Standards, page 3-17.

After selecting your criteria, click **Go** to launch the search and see a list of the available interfaces that meet the criteria. To save the list of interfaces to a CSV file, click **Export**. To review the details of an interface, click the interface name on the list.

### 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 services are formed or established, Oracle Integration Repository supports the following interface types:

- **Native Services**
  
  Native services are native packaged integration interfaces. This type of service includes the following native interfaces:
  
  - PL/SQL
  - XML Gateway
  - Concurrent Programs
  - Business Events
  - Open Interface Tables
  - Open Interface Views
  - EDI
  - Business Service Object (Service Beans)
  
  - Java, page 3-15
    
    Java interface includes the following subcategories:
    
    - Java Bean Services, page 3-15
    - Application Module Services, page 3-16
    - Security Services, page 3-16
Please note that Java APIs for Forms Web services are desupported in Oracle E-Business Suite Release 12.2. If you are planning to use this type of interfaces as Web services, you are advised to use alternate serviceable interfaces, such as PL/SQL and Business Service Objects interfaces, which can be deployed as Web services. Refer to My Oracle Support Knowledge Document 966982.1 for the suggested alternatives to the existing Java APIs for Forms services.

- **Composite Services**

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

The only available composite service type in this release 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 source files can be uploaded and displayed along with Oracle interfaces through the Integration Repository browser tree based on the interface types they belong to.

To easily differentiate them from Oracle interfaces, 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.

**Business Events**

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. The Business Event System consists of an Event Manager and workflow process event activities. The Event Manager lets you register subscriptions to significant events; event activities representing business events within workflow processes let you model complex business flows or logics within workflow processes.

When a local event occurs, the subscribing code is executed in the same transaction as the code that raised the event. Subscription processing can include executing custom code on the event information, sending event information to a workflow process, and sending event information to other queues or systems.

**Additional Information:** Users with Integration Repository
Administrator role can have the privilege to subscribe to a business event in the Business Event Details page. See *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide* for details.

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

**XML Gateway Message Maps**

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

Oracle XML Gateway consumes events raised by the Oracle E-Business Suite and subscribes to inbound events for processing. XML Gateway 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 supports both Business-to-Business (B2B) and Application-to-Application (A2A) initiatives. XML Gateway message maps (or just *XML Gateway 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 the Integration Repository information provided about XML Gateway maps, see XML Gateway Map Information, page 4-27.

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

**PL/SQL Procedures and Functions**

A business interface can be based on a PL/SQL package from which you invoke procedures and functions appropriate to a narrowly defined integration goal.

For the Integration Repository information provided about PL/SQL, see PL/SQL Information, page 4-32.

**Concurrent Programs**

In Oracle E-Business Suite, concurrent processing simultaneously executes programs running in the background with online operations to fully utilize your hardware capacity. A concurrent program runs as a concurrent process and is executed by the Concurrent Manager. Functions performed by concurrent programs are typically data-intensive and long-running, such as posting a journal, populating an interface table, and generating an EDI flat file.
For the Integration Repository information provided about Concurrent programs, see Concurrent Program Information, page 4-59. For more information about concurrent programs, refer to the Oracle E-Business Suite System Administrator’s Guide - Configuration.

Open Interface Tables

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

For the Integration Repository information provided about open interface tables, see Open Interface Information, page 4-62.

Interface Views

Interface views are database objects that make data from Oracle E-Business Suite products available for selection and use by destination applications.

For the Integration Repository information provided about interface views, see Interface View Information, page 4-66.

EDI Message Transactions

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, speed data processing, and ensure accuracy. 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. It is designed with an open and flexible architecture for easy integration with trading partners or EDI translators. When used for EDI solutions, e-Commerce Gateway integrates with EDI translators to provide specific EDI standard formats and versions. Oracle e-Commerce Gateway is a file-based integration layer between Oracle E-Business Suite and any other external application.

For the Integration Repository information provided about EDI messages, see EDI Message Information, page 4-69.

For more information about Oracle e-Commerce Gateway, see Oracle e-Commerce Gateway User’s Guide.

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 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.

Java APIs for Forms Web services are desupported in Oracle E-Business Suite Release 12.2. Refer to My Oracle Support Knowledge Document 966982.1 for the suggested alternatives to the existing Java APIs for Forms services.

**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.

Please note that security services will not require any REST service life cycle activities such as Deploy or Undeploy. Additionally, they are available to all users.

For the Integration Repository information provided about Java, see Java Information, page 4-36.

**Business Service Objects**

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.

It is the tool by which Oracle E-Business Suite employs service oriented architecture (SOA) and Web services to facilitate integration with each other and with third party trading partners.

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. Oracle Integration Repository includes it on lists of interface types, so you can browse or search for business service object interface based on the service data objects that they use.

**Composite Services**

A composite service consists of a collection of native packaged public interfaces or called native services that belong to a specific product or product family and are available in the Integration Repository.

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

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

For more information about composite services, see Working with Composite Services, 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
Working with Native Services and 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

The details page of each integration interface type contains the following two types of information:

- Interface detail information

  This interface detail information includes a header region with general information, a description region, a source region, and an interface methods or procedure and functions region.

Integration repository administrators can perform additional administrative tasks including generating a Web service for a selected interface type if the type has a Web service enabled, subscribing to a business event, and creating security grants for appropriate users.
For more information on interface details, see Common Information on Interface Details, page 4-3.

- Interface Web service information

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

Web service information including the SOAP-based and REST-based services if available for the supported interfaces can be displayed in the interface details page. This service related information such as service status, serviceable operations, service description, and supported service methods can be shown for the selected interface.

**Note:** In this release, only PL/SQL APIs, Concurrent Programs, and Business Service Objects 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 4-8.

For information on REST-based services, see Common Information on REST Web Services, page 4-11.
Each interface information page includes a header region with general information about the interface.

For information on SOAP and REST Web services, see Common Information on SOAP Web Services, page 4-8 and Common Information on REST Web Services, page 4-11.

The following fields are common to almost all interface types:
<table>
<thead>
<tr>
<th>Field</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Name</td>
<td>This is the PL/SQL package name, the document name, or the Java service interface name.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> For Java service interfaces, this is called <strong>Qualified Name</strong>, and includes the full Java package name and the class name.</td>
</tr>
<tr>
<td>Type</td>
<td>Business interfaces are organized into interface types according to the integration technologies on which they’re based.</td>
</tr>
<tr>
<td></td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>For more information about interface type, see interface type, page 3-12.</td>
</tr>
<tr>
<td>Product</td>
<td>The Oracle E-Business Suite product that supplies the interface.</td>
</tr>
<tr>
<td>Business Entity</td>
<td>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 using 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. <strong>Note:</strong> This field does not appear for Java service interfaces.</td>
</tr>
<tr>
<td>Status</td>
<td>Valid status codes are:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Active</strong></td>
</tr>
<tr>
<td></td>
<td>• <strong>Deprecated</strong> - this interface should not be used, but it will be supported until obsolete.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Obsolete</strong> - the interface is no longer supported.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Planned</strong> - This interface will be activated at a future date.</td>
</tr>
</tbody>
</table>
**Field** | **Notes**
---|---
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 3-11.

Interface Source | The only available interface source in Oracle Integration Repository is Oracle native packaged integration interfaces.

Interface Source on the Oracle Integration Repository Search page, page 3-10.

MetaLink | Included for any interface that has a related My Oracle Support (formerly Oracle MetaLink) Knowledge Document. Click the link to log in to My Oracle Support and view the Knowledge Document. A valid user name and password is required to access My Oracle Support.

Documentation | Included for any interface that has related online documentation. Click the link to view or download the documentation.

Online Help | Provided for any interface that has related Oracle E-Business Suite online help. Click the link to view online help for the interface.

Each interface information page also includes a Source Information region that contains the following fields:

**Field** | **Notes**
---|---
Source File | The source code file for this interface, and its location in the file system.
Field Notes

Source Version

The version of the source file. The first portion of the number corresponds to the base release version of Oracle E-Business Suite and the second portion is the version of the file. For example, 12.0.8 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.

Source Product

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 shortname. 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 (shortname) or license status.

Integration repository administrators can find the following buttons available in the interface details page:

- **Generate WSDL:** This generates a Web service WSDL file for a selected interface. If the file is generated successfully, you will find the Web Service region becomes available. **Regenerate WSDL** also appears in the details page allowing you to regenerate the service again if needed.

- **Deploy (Redeploy or Undeploy)**

  If the service has already been generated successfully, the administrators can find **Deploy** available in the Web Service region for the selected integration interface.

  If a service has been deployed, the administrators can find **Redeploy** (or **Undeploy**) available instead. This allows the administrators to redeploy or undeploy the deployed Web service if needed.

  Prior to deploying or redeploying the service to the application server, the administrators must first select at least one authentication type for the generated
service. This allows SOA Provider services the deployment based on the selected type(s).

- **Create Grant**: This allows the administrators to create security grants by authorizing the access permission of a selected interface method or a procedure or function to an appropriate user, a user group or all users.

Oracle Integration Repository also provides a feature to revoke the grants for a particular user for a selected method or service.

For more information on managing function security through security grants, see Managing Grants, page 4-23.

- **View Log**

This allows the administrators to view the logs generated during service generation and deployment. If logging is enabled for specific services or all services at the Site level only, administrators can find **View Log** in the Interface Details page. Click **View Log** to open the Log Details page where you can view log details compiled in the log table. You can also delete and export log messages retrieved in the Log Details page if needed. For more information, see Viewing Generate and Deploy Time Logs, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

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

**Additional Information**: The integration repository administrators (defined by the Integration Repository Administrator role) can find the Administration tab containing the following information displayed next to the Integration Repository tab:

- **SOA Monitor subtab**: This allows the administrators to monitor and audit all SOAP messages in and out through SOA Provider and view the message details.


- **Log subtab**: This allows the administrators to enable and configure log setups.

  For information about log configuration, see Logging for Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*. 
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) of the interface details page.

Web Service Region(s) for XML Gateway Map

XML Gateway maps can be supported by both Web Service Provider in Release 12.0 and by SOA Provider in this release; therefore, for backward compatibility, a profile option FND: XML Gateway Map Service Provider is used to let you select an appropriate service provider in enabling services for XML Gateway Map interface type. Based on your selected profile value, the interface details page can display the 'Web Service - Web Service Provider region' or 'Web Service - SOA Provider region', or displayed both regions at the same time if a service is generated successfully.


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

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Web Service Status (or SOAP Service Status if it's in the SOAP Web Service tab)</td>
<td>This field indicates different state of a SOAP service during service generation and deployment life cycle.</td>
</tr>
<tr>
<td>• 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:</td>
<td></td>
</tr>
<tr>
<td>• Regenerate WSDL: This button lets you regenerate the service if the interface definition has been changed before service deployment.</td>
<td></td>
</tr>
<tr>
<td>• Deploy: This button lets you deploy the generated SOAP service.</td>
<td></td>
</tr>
<tr>
<td>• Deployed: This indicates that the selected interface has been deployed to Oracle Application Server. Once a service has been successfully generated, the following buttons appear:</td>
<td></td>
</tr>
<tr>
<td>• Undeploy: This button lets you undeploy the SOAP service from the application server back to Oracle Integration Repository if necessary.</td>
<td></td>
</tr>
<tr>
<td>• Redeploy: This button lets you update the deployed service with the current system values.</td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>View WSDL</td>
<td>This link is shown after a selected interface has an associated SOAP service available. Click this link letting you review WSDL description for a generated or deployed service.</td>
</tr>
<tr>
<td></td>
<td>See: Reviewing Web Service WSDL Source, page 4-16.</td>
</tr>
<tr>
<td>Service Operations table</td>
<td>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 can generate the selected interface as a SOAP service.</td>
</tr>
<tr>
<td></td>
<td>Note that this table is still updatable after service generation, but any changes to the table will be applied only after regenerating the service.</td>
</tr>
<tr>
<td>Authentication Type</td>
<td>To secure web service content and authenticate web service operation, before deploying a generated service, an integration repository administrator must select one desired authentication type for the selected service in the Authentication Type field.</td>
</tr>
<tr>
<td></td>
<td>• Username Token: This authentication type provides username and password information in the security header to authenticate SOAP requests.</td>
</tr>
<tr>
<td></td>
<td>• SAML Token (Sender Vouches): This authentication type is used for web services relying on sending a username only through SAML Assertion.</td>
</tr>
<tr>
<td></td>
<td>Only users who have the Integration Repository Administrator role can select and modify the authentication type. For other users, this field is displayed in read-only mode.</td>
</tr>
<tr>
<td>Grant</td>
<td>The Grant icon is shown only in the SOAP Web Service tab (not the Web Service region).</td>
</tr>
<tr>
<td></td>
<td>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 Repository Administrator role and the System Integration Developer role can find the Grant icon and view the grant details.</td>
</tr>
</tbody>
</table>

**Performing Administrative Activities for SOAP Web Services**

Users who have the Integration Repository Administrator role can perform administrative tasks. These tasks include generating, deploying, undeploying, and redeploying SOAP services by clicking the following buttons in the interface details.
• **Generate:** This allows the administrators to generate a SOAP service. This transforms a selected integration interface definition into a web service using WSDL. After service generation, the administrators can regenerate or deploy the generated service if it's needed.


• **Deploy:** If the SOAP service has already been successfully generated, the administrators can deploy the generated SOAP service to an Oracle E-Business Suite server.

If the web service is successfully deployed, the administrators can undeploy or redeploy the web service if needed. See Deploying and Undeploying SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

• **Create Grant:** The administrators 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 4-23.

• **View Log:** This displays the Log Details page where the administrators can view and download the log details.

For more information on viewing logs recorded at design time, see Viewing Generate and Deploy Time 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 repository administrators can perform other administrative tasks in the Administration tab that are outside the Integration Repository user interface. These tasks include:

• **Configuring Log Setups**

This allows the administrators to enable and configure log setups for all services or specific services or operations. The administrators can easily monitor system activities, track and view log messages, and troubleshoot any issues encountered at each stage of service development life cycle.
For more information on SOA logging framework and how to enable and configure logs, see Logging for Web Services, Oracle E-Business Suite Integrated SOA Gateway Implementation Guide.

- Monitoring SOAP Messages Using SOA Monitor
  This allows the administrators to access SOA Monitor to monitor and audit all SOAP messages in and out from the SOA Provider (if the SOA auditing feature is enabled).

  For more information on how to use SOA Monitor, see Monitoring and Managing SOAP Messages Using SOA 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. It can be an interface type of PL/SQL, Concurrent Program, Business Service Object, Open Interface Table or View, Java Bean Services, or Application Module Services.

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

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Alias</td>
<td>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.</td>
</tr>
</tbody>
</table>

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.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REST Service Status</td>
<td>This field indicates different state of a REST service during service life cycle.</td>
</tr>
<tr>
<td></td>
<td>• Not Deployed: This indicates that the selected interface is not deployed. Click <strong>Deploy</strong> to deploy the REST service. The REST Web Service Status is changed from 'Not Deployed' to 'Deployed'.</td>
</tr>
<tr>
<td></td>
<td>• Deployed: This indicates that the selected interface is deployed to the application server.</td>
</tr>
<tr>
<td></td>
<td>Once a REST service has been successfully deployed, the administrator can undeploy the service by clicking <strong>Undeploy</strong>. This action undeploys the REST service from the 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'.</td>
</tr>
<tr>
<td></td>
<td>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:</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td></td>
<td>• 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.</td>
</tr>
<tr>
<td>View WADL</td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>This field appears only when the REST service has been successfully deployed with 'Deployed' status.</td>
</tr>
</tbody>
</table>
Field Description

Verb The Verb value indicates how the REST service is implemented using an HTTP method.

For PL/SQL APIs and 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 PL/SQL APIs and Concurrent Programs.

For 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 username and password when making a REST request that is typically over HTTPS.
  

- **Token Based Authentication:** This 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.

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 4-13.

**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 Repository Administrator role can perform administrative tasks including deploying or undeploying services as well as creating or revoking security grants.
The following table describes each field in the Service Operations table:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Name</td>
<td>This is the interface name used externally.</td>
</tr>
<tr>
<td>Internal Name</td>
<td>This is the interface name used internally.</td>
</tr>
<tr>
<td>Included Operations (PL/SQL APIs and Concurrent Programs Only)</td>
<td>This column appears when a selected PL/SQL API or Concurrent Program is exposed as a REST service. All the methods contained in the selected interface are automatically checked by default indicating that they are all exposed as REST service operations. <strong>Note</strong>: All methods in a PL/SQL API are deployed with POST HTTP method. A Concurrent Program contains only single method; therefore, once it is deployed, the method within the interface is automatically deployed with POST as well.</td>
</tr>
<tr>
<td>GET (Business Service Object, Java Bean Services, Application Module Services, and Open Interface Tables and Views Only)</td>
<td>This GET method column appears when a selected interface is an interface type of Business Service Object, Open Interface Table or View, Java Bean Services, or Application Module Services. For Business Service Object interfaces, the GET check box 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 check box is preselected if a Java or an Application Module method is annotated with the GET HTTP method. The administrator can uncheck the preselected GET check box for the Java or Application Module method if it will not be published with the GET method. However, if it is not annotated with GET method, unlike the POST method, the GET check box 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.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>POST</td>
<td>This POST method column appears when a selected interface is an interface type of Business Service Object, Java Bean Services, Application Module Services, or Open Interface Table with Inbound direction. For Business Service Object interfaces, select the POST check box 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 check box is preselected. The administrator can uncheck the preselected check box 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 check box remains active or enabled by default. The administrator can still select the POST check box if needed for a method. For Open Interface Table with Inbound direction, select the POST check box if desired for an interface table to be deployed as a REST service operation.</td>
</tr>
<tr>
<td>PUT</td>
<td>This HTTP method column appears only when the selected interface is an Open Interface Table with Inbound direction.</td>
</tr>
<tr>
<td>DELETE</td>
<td>This HTTP method column appears only when the selected interface is an Open Interface Table with Inbound direction.</td>
</tr>
<tr>
<td>Grant</td>
<td>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 Repository Administrator role and the Integration Repository Developer role can find the Grant icon and view the grant details.</td>
</tr>
</tbody>
</table>

**Performing Administrative Activities for REST Web Services**

REST services have a simplified service development life cycle. Users who have the Integration Repository Administrator role can perform the following administrative tasks in the interface details page:

- **Deploy** (REST Web Service tab): This allows the administrators to deploy the REST service.
For more REST service deployment information, see Deploying REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Undeploy** (REST Web Service tab): This action not only undeploys the service from the 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 administrators select 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 grant feature applies to both SOAP and REST services. For more information on security grants, see Managing Grants, page 4-23.

### Reviewing Web Service WSDL Sources

To make integration interfaces available to customers over a network where customers can dynamically interact between applications, Oracle Integration Repository allows integration repository administrators (or users who have the Integration Repository Administrator role) to generate web services WSDL files.

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.

**Note:** For XML Gateway Map interface type service enabled by Web Service Provider in Release 12.0, you will find a standard and deployed XML Gateway WSDL URL displayed with the 'Web Service Status - Deployed' status in the Web Service - Web Service Provider region.

For more information on service enablement support for XML Gateway Map interface type, see XML Gateway Map Web Service Region, page 4-28.
To view WSDL file:

1. Log in to Oracle E-Business Suite as a user who has the System Integration Analyst role. Select the Integrated SOA Gateway responsibility and the Integration Repository link from the navigation menu.
   
   Locate your desired interface definition through a search or from the navigation tree.

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

3. Click the View WSDL link in the SOAP Web Service tab (or the Web Service region for XML Gateway interface) to view the WSDL source code.

   The following sample shows the WSDL description for the PLSQL Interface: Repair Order:
<definitions name="CSD_REPAIRS_PUB"
    targetNamespace="http://xmlns.oracle.com/apps/csd/soaprovider/plsql/csd_repairs_pub/">
    xmlns="http://xmlns.oracle.com"
    xsi:schemaLocation="http://schemas.xmlsoap.org/soap/soap/"
    xmlns:tns1="http://xmlns.oracle.com/apps/csd/soaprovider/plsql/csd_repairs_pub/create_repair_order/"
<types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="qualified"
        targetNamespace="http://xmlns.oracle.com/apps/csd/soaprovider/plsql/csd_repairs_pub/">
        <include schemaLocation="http://<myhost>:<port>/webservices/SOAProvider/plsql/csd_repairs_pub/APPS_ISG_CREATE_REPAIR_ORDER_CSD_REPAIRS_PUB-24CREATE_REPAIR.xsd"/>
    </schema>
    <schema xmlns="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="qualified"
        <include schemaLocation="http://<myhost>:<port>/webservices/SOAProvider/plsql/csd_repairs_pub/APPS_ISG_UPDATE_RO_STATUS_CSD_REPAIRS_PUB-24UPDATE_RO_STA.xsd"/>
    </schema>
    <schema xmlns="http://www.w3.org/2001/XMLSchema"
        elementFormDefault="qualified"
        targetNamespace="http://xmlns.oracle.com/apps/csd/soaprovider/plsql/csd_repairs_pub/">
        <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>
</types>

Note: Values passed in the Responsibility, Responsibly Application, Security Group, NLS Language, and Organization ID complex types listed under the "SOAHeader" are used to set applications context during service execution.

Note that NLS Language and Organization ID are optional values to be passed. However, if the execution of a service is dependent on any particular organization, then you must pass the ORG_ID element in the "SOAHeader" of that SOAP request.

Note that the user information is defined by the wsseUsername property passed within the security headers. Detailed instructions on how to pass the security headers along with the SOAP request, see Passing Values to Security Headers, 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:

When an integration repository administrator successfully deploys or redeploys a web service, 'Deployed' appears in the SOAP Service Status field along with the deployed WSDL link in the SOAP Web Service tab or the appropriate Web Service region for an XML Gateway interface.

Click the View WSDL link to view the deployed WSDL file.

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
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
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="https://host01.example.com/apps/ar/soaprovider/plsql/rest/ar_invoice_api_pub/Invoice/?XSD=CREATE_INVOICE_SYNCH_TYPEDEF.xsd" />

    <include xmlns="http://www.w3.org/2001/XMLSchema" href="https://host01.example.com:1234/webservices/rest/Invoice/?XSD=CREATE_SINGLE_INVOICE_SYNCH_TYPEDEF.xsd" />
  </grammars>
  <resources base="http://host01.example.com:1234/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://host01.example.com:1234/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’. This is the only HTTP method supported for PL/SQL REST services in this release.

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 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.
<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_resapplication" 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="/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>

• {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 run time. 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 System Integration Analyst role. Select the Integrated SOA Gateway 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 repository administrators (or users who have the Integration Repository Administrator role) can create security grants by authorizing the access permission of a selected interface method or procedure and function to an appropriate user, user group, or all users.

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 4-27.
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 'Change User Name', the user will have the permission to access the associated 'Change User Name' service operations of both SOAP and REST service types through the same grant.

**Creating Security Grants**

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

Note that the grant action applies to both SOAP and REST PL/SQL services.

**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 'Get Profile' and 'Get Profile Value') 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.
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.


**XML Gateway Map Information**

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

**Basic Structure of the XML Gateway Information Page**

The XML Gateway Map information page contains the following information:

- **Web Service Region**
  
  If the XML gateway map is exposed as a Web service, appropriate Web Service region(s) will be available.


- **Methods Region**
  
  The Methods region links to one or more XML Gateway Method information pages.

The general section of the XML Gateway Map displays common information, page 4-1.

The information page or interface details page includes a table listing the XML Gateway methods. Click a method name to access the information page for that method.

**XML Gateway Map Web Service Region**

To support the XML Gateway Map service enabled by Web Service Provider in Release 12.0 and to differentiate the service enablement by SOA Provider in this release, Oracle E-Business Suite Integrated SOA Gateway uses the following profile option to let you select an appropriate service provider in enabling services for XML Gateway Map interface type. Based on the selected profile value, the interface details page displays an appropriate Web Service region or more than one region.

Select one of the following values to define the **FND: XML Gateway Map Service**
Provider profile option:

- **WSP (Web Service Provider)**
  
  This displays the Web Service - Web Service Provider region if Web services are available.

**Web Service - Web Service Provider region**

*XML Gateway Map with Web Service - Web Service Provider Region*

In Release 12.0, XML Gateway Map interface type were deployed by default through Web Service Provider; therefore, you can find a standard XML Gateway WSDL URL displayed in this region with the 'Web Service Status - Deployed' status. http://<hostname>:

<port>/webservices/AppsWSPProvider/oracle/apps/fnd/XMLGateway?wsdl

- **SOAP (SOA Provider)**
  
  This is the default profile value which displays the Web Service - SOA Provider region if Web services are available.

**Web Service - SOA Provider region**

**Note:** The default profile value is set to 'SOAP'. However, if you do not start from this release and your system is upgrading from Release 12.0, you must change the profile value from the default 'SOAP' to 'Both' because Web Service Provider could have already been used in enabling services. To continue having service enabled
using SOA Provider and for backward compatibility, both service providers should be enabled in transforming XML Gateway Map interface definitions into Web services. Otherwise, a fault message appears if it is still set to the default profile value ‘SOAP’ (SOA Provider).

If you start with Rapid Install of Oracle E-Business Suite for this release, the default service provider is SOA Provider (‘SOAP’ profile value). In this situation, Web Service Provider will be disabled and any invocations of generic XML Gateway Web services will return a fault message.

The View WSDL link is available along with appropriate Web Service Status information in the Web Service - SOA Provider region indicating whether the service is generated or deployed. For information on viewing WSDL description, see Reviewing Web Service WSDL Source, page 4-16.

To secure Web service content and authenticate Web service operation, SOA Provider supports multiple authentication types for inbound service requests. In addition to the Web Service Status field and View WSDL link, you can find the Authentication Type field with the following read-only check boxes:

- Username Token: This authentication type provides username and password...
information in the security header for a Web service provider to use in authenticating the SOAP request.

- **SAML Token (Sender Vouches):** This authentication type is used for Web services relying on sending a username only through SAML Assertion.

Before an integration repository administrator deploys a generated service, the administrator must select at least one authentication type for the selected service. See: Deploying and Undeploying Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **Both**

  This displays both the Web Service - Web Service Provider region and Web Service - SOA Provider region in the interface details page if Web services are available.

For more profile option information used in Oracle E-Business Suite Integrated SOA Gateway, see *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

Integration repository administrators (defined by the Integration Repository Administrator role) can also find the following buttons available if the Web service has been generated successfully:

- **Deploy**: Deploys the Web service.
- **Undeploy**: Undeploys a Web service if the service has been deployed.
- **Redeploy**: Redeploys the Web service if needed.

See: Performing Administrative Activities for SOAP Web Services, page 4-9.

**XML Gateway Method Information**

The XML Gateway method information page appears when you click a method name on an XML Gateway Map information page.
The general section of this page displays common information, page 4-1 for the XML Gateway method, plus the following additional field:

**Direction**

- **Inbound** indicates that the interface receives incoming transactions or messages into the Oracle E-Business Suite.
- **Outbound** indicates that the interface sends outgoing transactions or messages to another system.

This page also contains a table listing the XML Gateway method parameters, including each parameter’s data type and whether the parameter is required.

**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/function information page:
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 Repository 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 4-23.

For information on the general section, see Common Information, page 4-1.
Note: 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 4-8.
  
  For information on viewing WSDL description, see Reviewing Web Service WSDL Source, page 4-16.
  
  If a SOAP service has been successfully generated, the administrators can perform
additional administrative tasks including deploying the generated service, or regenerating the service.

For more information on these administrative tasks, see Performing Administrative Activities for SOAP Web Services, page 4-9.

• **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, and service operation information.

  Note that POST is the only HTTP verb supported in this release. For more information about REST service, see Common Information on REST Web Services, page 4-11.

  All REST services are secured by HTTP Basic Authentication or Token Based 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 (username and password).

  If a REST service has been successfully deployed to an Oracle E-Business Suite application server, the administrators 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 4-15.

**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.

Basic Structure of the Java Information Page

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
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 4-23.

**Java Method Details**

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

**Java method information page**

The general section of the Java method information page displays common information, page 4-1 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 3-8.

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 4-38
- Application Module Services, page 4-41
- Security Services, page 4-43
- Java APIs for Forms, page 4-45

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:
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 3-9.

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 repository administrator can find an additional Grants tab displayed in the page. This Grants tab allows the administrator to create and revoke security grants. For more information on how to manage security grants, see Managing Grants, page 4-23.

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

**Note:** For more information about web services, see Understanding Web Services, page 2-1.
• **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, and service operation information.

For more information about REST service, see Common Information on REST Web Services, page 4-11.

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 Token Based 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 (username 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 4-15.

**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:

**Basic Structure of the Application Module Services Information Page**

![Diagram of Application Module Services Information Page]

**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 3-9.

**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 4-23.

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

**Note:** 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, and service operation information.

For more information about REST service, see Common Information on REST Web Services, page 4-11.

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 Token Based 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 (username 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 4-15.

**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 your search criteria in the Category (subtype) and the Category Value (Security Services) fields.

For information about category and category value fields, see Category and Category
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 executing 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 3-9.

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 Web services are desupported in Oracle E-Business Suite Release 12.2. If you are planning to use this type of interfaces as Web services, you are advised to use alternate serviceable interfaces, such as PL/SQL and Business Service Objects interfaces, which can be deployed as Web services. Refer to My Oracle Support Knowledge Document 966982.1 for the suggested alternatives to the existing Java APIs for Forms services.

Business Service Object

Business service object interface type, formerly known as service bean, provides the access to SOA services and facilitates integration between Oracle E-Business Suite and trading partners. They can be used directly, or they can be exposed as SOAP or REST services. They often employ service data objects as parameters to pass complex data.

A service data objects (SDO) defines a generic API for accessing and manipulating structured data as part of a Service Oriented Architecture (SOA). 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 4-52.

Note: A business service object is not actually an interface type; rather, it is an object used by one or more Java service interfaces or other service data objects to pass data. Oracle Integration Repository includes it on lists of interface types, so 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](image)

**Business Service Object Interface Information**

This interface type can be exposed as both SOAP services and REST services. 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.

**Note:** Users who have the Integration Repository 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 4-23.

The general section displays common information for the selected business service object interface, plus interface name, the interface that extends, and XML schema information.

Note that an XML schema is a description of a type of XML document, typically expressed in terms of constraints on the structure and content of documents of that type. It describes all input and output message definition and data type. 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.

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.

The business service object interface information page contains the following information:

- **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 information about each method contained in the selected business service object interface, see Business Service Object Interface Method Information, page 4-48.

- **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 4-8.

For information on viewing WSDL description, see Reviewing Web Service WSDL Source, page 4-16.

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

For more information on these administrative tasks, see Performing Administrative Activities for SOAP Web Services, page 4-9.

* 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, and service operation information.

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 4-11.

All REST services are secured by HTTP Basic Authentication or Token Based 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 (username and password).

If a REST service has been successfully deployed to an Oracle E-Business Suite application server, the administrators 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 4-15.

**Business Service Object Interface Method Information**

The business service object interface method information page appears when you click a method name on the Overview tab, the SOAP Web Service tab, or the REST Web Service tab of the selected business service object interface 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 information page:

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

- **Return**
  If the return type is a service data object, you can click the link in the Type field to access the service data object information 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 4-52.

**Integration Repository Service**

Based on business service object interface, Integration Repository Service is a service component residing in Oracle 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. You can grant the control access of each business service object method to appropriate users.

To access the Integration Repository Service interface, log in to Oracle E-Business Suite as a user who has the System Integration Analyst role and use the following steps to navigate to Integration Repository Service:

1. Select the Integrated SOA Gateway responsibility from the Navigator menu, and click the Integration Repository link that appears.

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 execute the search.

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

   This opens the Business Service Object Interface information page. See: Business Service Object Interface information page, page 4-46.

6. Click a method name link in the Service Operations region to get to business service object method details.


**Viewing WSDL Description**

Click the **View WSDL** link in the SOAP Web Service tab to view the WSDL description generated.

The following sample shows the WSDL description for the Integration Repository Service:
Working with Native Services and Integration Interfaces

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 applications context during service execution.

Please note that NLS Language and Organization ID are optional values to be passed. However, if the execution of a Business Service Object interface is dependent on any particular organization, then you must pass the ORG_ID element in the ServiceBean_Header of that SOAP request.

For more information, see Setting Other Web Service Input Message Parts, Oracle E-Business Suite Integrated SOA Gateway Implementation Guide.

Please note that the user information is defined by the wsseUsername property passed within the security headers. Detailed instructions on how to pass the security headers along with the SOAP request, see Passing Values to Security Headers, Oracle E-Business Suite Integrated SOA Gateway Developer’s Guide.

You might find the following information under <Method>_Response about error
messages if occur:

```xml
<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>
```

**Note:** The Message and ErrorMessage elements listed under <Method>_Response are used for error messages if occur. The Message element will appear as warning messages in the SOAP response. It is used to display any warning messages 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)


### Reviewing Service Data Object

A service data objects (SDO) defines a generic API for accessing and manipulating structured data as part of a Service Oriented Architecture (SOA). 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

This page is accessible from the `getDataList` and `processDataList` method.
information pages. You can also access it directly from the Oracle Integration Repository browse interface, through the list of interface types.

The following diagram illustrates the basic structure of the service data 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, page 4-1, 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 information 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 access 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 access the selected filter data object information page.

- **Services**
  The Services region lists the services that directly use this service data object.
  
  Click the name of a service to access 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 access its information page.

- **Methods**
  Click a link in the **Name** column to access the service object method information page.

**Filter Data Object Information**

This page is accessible only from the **getDataSource** method information page.

The following diagram illustrates the basic structure of the filter data object information page and its connections to related pages.
A filter data object is a type of service object. The general section of the filter data object page displays common information, page 4-1, 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.


Click the XML schema name link to view the schema document displayed in a separate window.
The following regions also appear on the service object information 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 access the service data object information page.

- **Methods**
  Click a link in the **Name** column to access the object method information 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 information page appears when you click a method name on the service data object information page or the filter data object information page.

Service Data Object Method Information Page

In addition to a description, the following regions also appear on the service data object method information 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 access the selected service object method details.
Concurrent Program Information

The following diagram illustrates the basic structure of the concurrent program information page and its connection to the related Open Interface information page:

**Basic Structure of the Concurrent Program Information Page**

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 Repository 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 4-23.

For information on the general section, see Common Information, page 4-1.
Note: For more information about Web services, see Understanding Web Services, page 2-1.

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 a 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 4-8.
For information on viewing WSDL description, see Reviewing Web Service WSDL Source, page 4-16.

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

For more information on these administrative tasks, see Performing Administrative Activities for SOAP Web Services, page 4-9.

• **REST Web Service Tab**

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

  Please note that POST is the only HTTP verb supported in this release. For more information about REST service, see Common Information on REST Web Services, page 4-11.

  All REST services are secured by HTTP Basic Authentication or Token Based 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 (username and password).

  If a REST service has been successfully deployed to an Oracle E-Business Suite application server, the administrators 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 4-15.

**Concurrent Program 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
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.

**Basic Structure of the Open Interface Information Page**

The Open Interface information table lists the open interface tables and views that store the interface data, including active status, and whether it stores data inbound to Oracle E-Business Suite or outbound to another system. Click an interface table name to access the information page for that table.
Open Interface integrations can be available as REST 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 Repository 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 4-23.

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.
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, and service operation information.
  
  Please note that the supported HTTP verbs are determined by the direction of an open interface table.
  
  - 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

### Service Operations

<table>
<thead>
<tr>
<th>Name</th>
<th>Direction</th>
<th>GET</th>
<th>POST</th>
<th>PUT</th>
<th>DELETE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR AUTOINVOICE</td>
<td>Inbound</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAI INTERFACE DEBTS ALL</td>
<td>Inbound</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAI INTERFACE ERRORS ALL</td>
<td>Inbound</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAI INTERFACE LEDS ALL</td>
<td>Inbound</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RAI INTERFACE SALES_ALL</td>
<td>Inbound</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Tip**: To apply any changes in Operation, Undeploy the service.

### REST Service Security

REST Web Service is secured by HTTP Basic Authentication at HTTP Transport level. Send either of the following in "Authorization" header as per HTTP Basic scheme:
- Username/Password
- Security Token.

Tip: Log in to obtain Security Token for given user credentials.
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 4-11.

All REST services are secured by HTTP Basic Authentication or Token Based 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 (username 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 4-15.

**Open Interface Table Details Page**

Click a name link in the Service Operations 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.
Interface View Information

The general section of the interface view information page displays when you browse from the tree nodes.
Open Interface View List

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

View Details: Active Subscriptions View

<table>
<thead>
<tr>
<th>Internal Name</th>
<th>WF_ACTIVE_SUBSCRIPTIONS_V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Interface View</td>
</tr>
<tr>
<td>Product</td>
<td>Workflow</td>
</tr>
<tr>
<td>Business Entity</td>
<td>Business Event</td>
</tr>
</tbody>
</table>

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

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, and service operation 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 4-11.

All REST services are secured by HTTP Basic Authentication or Token Based 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 (username 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 4-15.

**EDI Message Information**

The general section of the EDI message information page displays common information, page 4-1 for the selected EDI message.
The EDI Message information page also 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.

**Business Event Information**

The general section of the Business Event displays common information, page 4-1 for business event.
An integration repository administrator can perform the administrative tasks:

- **Subscribe to an event**

  An administrator can find the **Subscribe** button available in the event details page if the selected event is not subscribed. 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 `WF_BPEL_Q`, 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 and Reference*.

- **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*.
Overview of Composite Services

A composite service consists of a collection of native packaged public interfaces or called native services that belong to a specific product or product family and are available in the Integration Repository.

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

To create a composite service, integration developers need to specify the invocation sequence in Oracle JDeveloper by using the BPEL language. This composite service has its own WSDL definition and endpoint through the creation of a partner link which allows an outbound business event, for example, to be published to the Oracle BPEL Process Manager or to interact with a partner service.
**A Composite Service - BPEL in Oracle JDeveloper**


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

**Viewing Composite Services**

You can view composite services by navigating to the Composite Service interface type directly from the Oracle Integration Repository Browser window or performing a search by selecting Composite Service interface type in the Search page.

By clicking a composite service name link from the navigation tree or search results,
you will find the composite service interface details page where displays composite service name, description, BPEL file, and other annotated information.

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

- View a WSDL file by clicking the URL link
  See: Reviewing Web Service WSDL Source, page 4-16.

- View the composite - BPEL file by clicking the URL link
  You will find the BPEL code displayed in a pop-up window containing major BPEL process components and activities included for the composite service.

Users granted with the Download Composite Service privilege can find additional Download Service in the interface details page. This lets you download a corresponding composite service project file, such as a BPEL file, to your local machine.

See: Downloading Composite Services, page 5-3.

It is important to note that a composite service - BPEL itself consisting of multiple native services is considered as a Web service. Therefore, there is no Generate or Regenerate shown in the composite service details page.

To view a composite service:

1. Log in to Oracle E-Business Suite as a user who has the System Integration Analyst role. Select the Integrated SOA Gateway responsibility. Select the Integration Repository link.

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

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

4. Click the composite service that you want to review to open the Composite Service Interface Details page.

5. Click the WSDL link to review the WSDL description.

6. Click the BPEL link to view the BPEL code.

Downloading Composite Services

In addition to viewing composite service details and reviewing a WSDL file, users with the download privilege can download a composite service BPEL JAR file to your local directory.
Important: In general, only system integration developers and integration repository administrators can download the composite services. However, general users (system integration analysts) who are granted the download privilege, an Integration Repository Download Composite Service permission set FND_REP_DOWNLOAD_PERM_SET, can also perform the download action. Otherwise, users will not find Download Service available in the details page.

For information about how to grant Download Composite Service privilege, see Role-Based Access Control (RBAC) Security, Oracle E-Business Suite Integrated SOA Gateway Implementation Guide.

Composite Details Page with Download Privilege

To download the relevant files aggregated in a .JAR file for a composite service BPEL project, navigate to the composite service details page for a service that you want to download, and then click Download Service to download the file to your local machine.

Note: After downloading the file, system integration developers can unzip the BPEL .JAR file and open the BPEL process in Oracle JDeveloper for further modification on service endpoints if needed. Additionally, the developers can deploy the BPEL process. Since composite services are typically not deployed within Oracle E-Business Suite, a separate BPEL PM (SOA Suite or a third party BPEL PM server) is needed to deploy the BPEL composite services. For example, the developers can deploy it to Oracle BPEL server through Oracle BPEL.
To download a composite service:

1. Log in to Oracle E-Business Suite as a user who has granted the download composite service privilege. Select the Integrated SOA Gateway responsibility from the navigation menu and then 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 Service interface type node to locate your desired composite service.

4. Click the composite service that you want to download it to open the Composite Service Interface Details page.

5. Click **Download Service** to download the selected composite file to your local machine.
Overview of Custom Integration Interfaces

Oracle Integration Repository provides the capability to display annotated custom integration interfaces that are created for native and composite services.

Integration developers create and annotate custom integration interfaces based on the Integration Repository annotation standards. These annotated source files can then be uploaded to the Integration Repository through backend processing.

Since custom interface definitions can be created for various interface types, including custom interface definitions for XML Gateway Map, Business Event, PL/SQL, Concurrent Program, Business Service Object, Java, Java Bean Services, Application Module Services, and Composite Service for BPEL type, these annotated interface definitions are merged into the interface types they belong to and displayed together with Oracle interfaces from the Integration Repository browser window.

**Note:** Please note that custom interface types of EDI, Open Interface Tables, and Open Interface Views 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 interfaces, Oracle Integration Repository provides the following capabilities:

- Ability to restrict display of custom or Oracle interfaces (seeded) in the Interface List page from the navigation tree
- Ability to search on custom or seeded integration interfaces for a product family in the Search page
The interface details page displays an additional value indicating whether this interface is a custom or seeded one.

For Integration Repository annotation standards, see Integration Repository Annotation Standards, Oracle E-Business Suite Integrated SOA Gateway Developer’s Guide.

This section discusses the following topics:

- Searching Custom Integration Interfaces, page 6-2
- Viewing Custom Integration Interfaces, page 6-5
- Performing Additional Web Service Activities for Custom Integration Interfaces, page 6-6

**Searching Custom Integration Interfaces**

You can find custom integration interfaces in the following ways:

- 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.
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 on Oracle E-Business Suite with the Integrated SOA Gateway responsibility. Select the Integration Repository link to open the repository browser.

2. You can locate custom integration interface definitions from 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 execute the search.
   Custom integration interfaces that match your search criteria should be displayed in a table format.

4. Select a custom integration interface name link from the search result to view the interface details.
   Note that the custom integration interface details page shows ‘Custom’ as the Interface Source value allowing you to differentiate it from Oracle seeded interfaces.

Viewing Custom Interface Details

After performing a search on custom integration interfaces either from the Interface List page or from the Search main page, you can view the details page for a selected custom integration interface from the search result.

Select a custom integration interface name link from the search result table, the interface details page appears where you can find the interface name, description, and other annotated information.

Please note that all custom integration interface definitions have ‘Custom’ value in the Interface Source field and this value distinguishes it from a seeded one.

Users who have the Integration Repository Administrator role can perform the following tasks:
   - For the custom interface that can be exposed as SOAP Web service
     - Generate SOAP Web services
     - Deploy (undeploy or redeploy) SOAP Web services
   - For the custom interface that can be exposed as REST Web service
     - Deploy REST Web services
     - Undeploy REST Web services

For more information about these administrative tasks, see Performing Additional Web
Performing Additional Web Service Activities for Custom Integration Interfaces

In addition to viewing custom integration interface details, users with administrator role can perform the following administrative tasks:

- **For Custom Integration Interfaces with Support for SOAP Web Services**
  - Generating Custom SOAP Web Services
    
    Users who have the Integration Repository Administrator role can find **Generate WSDL** (or **Generate** for the custom PL/SQL interface) available in the interface details page. Clicking **Generate WSDL** (or **Generate**) automatically generates the custom SOAP service with corresponding WSDL file. See: Generating SOAP Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

  - Deploying Custom SOAP Web Services
    
    If the custom service has already been generated successfully, users who have the Integration Repository Administrator role can find **Deploy** available for the selected custom integration interface. Prior to deploying the service to the application server, the administrator must first select at least one authentication type for the generated service supported by SOA Provider.

    If the service has been successfully deployed, the Web Service Status field will be updated to 'Deployed'. The **View WSDL** link appears along with the selected authentication type(s) for the deployed custom web service.

- Redeploying or Undeploying Custom SOAP Web Services
  
  The following buttons appear if a custom web service has been successfully deployed:

  - **Redeploy**:
    
    This allows the administrators to redeploy the custom service when needed. If changes are made to the Authentication Type field for the deployed service, the administrator must redeploy the custom web service.

  - **Undeploy**: This allows the administrators to undeploy the custom service that has been deployed earlier.


- Viewing Log Messages
To effectively troubleshoot any issues or exceptions encountered at each stage of service development and deployment life cycle, the administrators can view design-time logs through the Interface Details page and runtime logs through the SOA Monitor user interface for the service or operation if the logging is enabled and configured properly at required logging category level.

For information on how to view design-time logs, see Viewing Generate and Deploy Time Logs, Oracle E-Business Suite Integrated SOA Gateway Implementation Guide.

For information on how to view log messages through SOA Monitor, see Viewing Service Processing Logs, Oracle E-Business Suite Integrated SOA Gateway Implementation Guide.

For detailed information about each administrative task listed here, 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.

- **Deploying Custom REST Web Services**

  Before deploying a custom interface as a REST service, the administrator must specify service alias for the selected interface. If the selected custom interface type is Java Bean Services, Application Module Services, or Business Service Object, the administrator also needs to specify HTTP verbs for the desired methods contained in the selected interface before deployment.

  If the custom REST service has been successfully deployed, the REST Service Status field is updated to 'Deployed' from 'Not Deployed'. This indicates that the active service is ready to accept new service requests.

  For more information on deploying custom REST services, see Deploying REST Web Services, Oracle E-Business Suite Integrated SOA Gateway Implementation Guide.

- **Undeploying Custom REST Web Services**

  If a custom service has been successfully deployed, the administrators can click Undeploy to undeploy the REST service if necessary and at the same time to reset the service status to its initial state - 'Not Deployed'.

  For more information on undeploying REST services, see Undeploying REST

For detailed information about administrative tasks, see Administering REST Web Services, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **For Business Events**
  
The administrators can subscribe to a business event by clicking *Subscribe* in the business event interface details page. This creates subscription for the selected event.

  For information on how to subscribe to an event, see Subscribing to Business Events, *Oracle E-Business Suite Integrated SOA Gateway Implementation Guide*.

- **For Custom Composite Integration Interface**
  
  - **Viewing Custom Composite Services**
    
    You can view custom composite service details including the associated WSDL file for a selected custom composite service.

    To locate a custom composite service, you can perform a search from the Search page. 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 as criteria:
    
    - Interface Source: Custom
    
    - Interface Type: Composite

    can view a custom composite service details, and download the .ZIP file for a composite service if it is available for download.

    For more information on viewing composite services, see Viewing Composite Services, page 5-2.

  - **Downloading Custom Composite Services**
    
    Similar to downloading native packaged 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.

    For more information on how to download a composite service, see Downloading Composite Services, *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.

BPEL
Business Process Execution Language (BPEL) provides a language for the specification of executable and abstract business processes. By doing so, it extends the services interaction model and enables it to support business transactions. BPEL defines an interoperable integration model that should facilitate the expansion of automated process integration in both the intra-corporate and the business-to-business spaces.

Business Event
See Event.

Concurrent Manager
An Oracle E-Business Suite component that manages the queuing of requests and the operation of concurrent programs.

Concurrent Program
A concurrent program is an executable file that performs a specific task, such as posting a journal entry or generating a report.

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.

**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 username 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.

**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.

**Username Token**
A type of security model based on username and password to authenticate SOAP requests at run time.

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 a machine-processable format (specifically 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 executes 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

**A**
- accessing the repository, 3-3

**B**
- browsing the interfaces, 3-6
  - by integration standard, 3-8
  - by interface type, 3-7
  - by product family, 3-7
- business entity, 3-6
- Business Event, 3-13, 4-70
- business interface, 3-1
- business service object, 3-16
  - general information, 4-53
  - Integration Repository Service, 4-49
  - method information, 4-48
  - service data object, 4-52

**C**
- common information
  - interface details, 4-3
  - managing grants, 4-23
  - REST service, 4-11
  - web service, 4-8
    - Reviewing Web Service WADL Sources, 4-19
    - review WSDL, 4-16
- common interface information, 4-1
- Composite service, 3-17
- Composite Services, 5-1
- concurrent program, 3-14, 4-59
- Custom Integration Interfaces, 6-1, 6-5
  - administrative tasks, 6-6
  - searching custom interfaces, 6-2

**D**
- discovering interfaces
  - by browsing, 3-6
  - by searching, 3-8

**E**
- EDI, 3-15, 4-69
  - general information, 4-70

**F**
- filter data object
  - general information, 4-56
  - method information, 4-58

**I**
- Integration Repository Service, 4-49
- integration standard
  - browsing by, 3-6, 3-8
  - searching by, 3-12
- interface information
  - common information, 4-1
  - source information, 4-3
- interface types, 3-1, 3-12
  - browsing by, 3-6, 3-7
- Business Event, 3-13, 4-70
- business service object, 3-16
- Composite service, 3-17
- concurrent program, 3-14, 4-59
EDI, 3-15, 4-69
interface view, 3-15, 4-66
Java, 3-15, 4-36
open interface, 3-15, 4-62
PL/SQL, 3-14, 4-32
service data object, 4-52
XML Gateway, 3-14, 4-27
interface view, 3-15, 4-66

J
Java, 3-15, 4-36
Application Module Services, 4-41
Java APIs for Forms, 4-45
Java Bean Services, 4-38
method information, 4-37
Security Services, 4-43
Java service interface
filter data object information, 4-55

O
open interface, 3-15, 4-62
table information, 4-63
Oracle E-Business Suite Integrated SOA Gateway
Architecture, 1-4
Major Features, 1-1
Overview, 1-1
Oracle Integration Repository, 5-1, 5-2
discovering and reviewing, 3-6
Discovering and Reviewing, 3-6
overview, 6-1

P
PL/SQL, 3-14, 4-32
procedure/function information, 4-35
Web service tab, 4-34, 4-60
product codes, 4-3
product family browsing, 3-7

R
Review Composite Services, 5-2

S
search criteria, 3-8
business entity, 3-6
category, 3-9
integration standard, 3-17
interface type, 3-12
internal name, 4-4
scope, 3-11
status, 3-10
Web service status, 3-10
searching for interfaces, 3-8
security, 3-11
service data object
method information, 4-58
service data object information, 4-52
SOA
common terms, 2-2

U
Understanding Service Enablement
service enablement overview, 2-1
SOA in Oracle E-Business Suite, 2-3, 2-5
using the repository, 3-5

V
viewing custom interface details, 6-5

W
Web service
downloading, 5-3

X
XML Gateway, 3-14, 4-27
general information, 4-28
integration standards, 3-8, 3-17
method information, 4-31
Web Service, 4-28
Web service subtab, 4-28