Oracle® Public Cloud Machine

Using Oracle Integration Cloud Service

Release 16.3.1

E67461-04

March 2017

This guide describes how to create, activate, monitor, and manage integrations in Oracle Integration Cloud Service.

ORACLE

Oracle Public Cloud Machine Using Oracle Integration Cloud Service, Release 16.3.1

E67461-04

Primary Author: Mark Kennedy

Contributing Authors: Larry Hoffman

Contributors: Oracle Integration Cloud Service Development, Product Management, and Quality Assurance teams

Contents

3 Developing Integration Cloud Services

5 Adapter Configuration Reference

Preface

Using Oracle Integration Cloud Service describes how to use Oracle Integration Cloud Service to integrate your applications.

Topics:

Audience

Using Oracle Integration Cloud Service is intended for users who want to create, activate, and monitor application integrations.

Related Resources

For more information, see these Oracle resources:

- *Using the Oracle Mapper*
- *Using the Evernote Adapter*
- *Using the Eventbrite Adapter*
- *Using the Facebook Adapter*
- *Using the FTP Adapter*
- *Using the Gmail Adapter*
- *Using the Google Calendar Adapter*
- *Using the Google Task Adapter*
- *Using the LinkedIn Adapter*
- *Using the MailChimp Adapter*
- *Using the Microsoft Email Adapter*
- *Using the Microsoft Contact Adapter*
- *Using the Microsoft Calendar Adapter*

- *Using the MySQL Adapter*
- *Using the Oracle Commerce Cloud Adapter*
- *Using the Oracle Database Adapter*
- *Using Oracle E-Business Suite Adapter*
- *Using the Oracle Siebel Adapter*
- *Using the SAP Adapter*
- *Using the SurveyMonkey Adapter*
- *Using the Twitter Adapter*
- Oracle Public Cloud Machine documentation in the Oracle Help Center:

  http://docs.oracle.com

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Getting Started with Integration Cloud Service

Review the following topics to learn about how Oracle Integration Cloud Service works. These topics provide information about Oracle Integration Cloud Service concepts and components to help you get started with creating your own integrations.

Topics

About Oracle Integration Cloud Service

Oracle Integration Cloud Service is a complete, secure, but lightweight integration solution that enables you to connect your applications in the cloud. It simplifies connectivity between your applications, and can connect both your applications that live in the cloud and your applications that still live on premises. Oracle Integration Cloud Service provides secure, enterprise-grade connectivity regardless of the applications you are connecting or where they reside.

Oracle Integration Cloud Service provides native connectivity to Oracle Software as a Service (SaaS) applications, such as Oracle Sales Cloud, Oracle RightNow Cloud, and so on. Oracle Integration Cloud Service *adapters* simplify connectivity by handling the underlying complexities of connecting to applications using industry-wide best practices. You only need to create a *connection* that provides minimal connectivity information for each system. Oracle Integration Cloud Service *lookups* map the different codes or terms used by the applications you are integrating to describe similar items (such as country or gender codes). Finally, the visual data mapper enables you to quickly create direct mappings between the trigger and invoke data structures. From the mapper, you can also access lookup tables and use standard XPath functions to map data between your applications.

Once you integrate your applications and activate the integrations to the runtime environment, the dashboard displays information about the running integrations so you can monitor the status and processing statistics for each integration. The

dashboard measures and tracks the performance of your transactions by capturing and reporting key information, such as throughput, the number of messages processed successfully, and the number of messages that failed processing. You can also manage business identifiers that track fields in messages and manage errors by integrations, connections, or specific integration instances.

About Integration Cloud Service Concepts

The following topics describe each of the components required to create an end-to-end integration. Each integration includes connections and mappings. You can also include lookups, which are reusable mappings for the different codes and terms used in your applications to describe the same item. You can also group integrations into packages.

Topics

About Oracle Integration Cloud Service Connections

Connections define information about the instances of each predefined configuration you are integrating. Oracle Integration Cloud Service includes a set of predefined *adapters*, which are the types of applications on which you can base your connections, such as Oracle Sales Cloud, Oracle Eloqua Cloud, Oracle RightNow Cloud, and others. A connection is based on an adapter. A connection includes the additional information required by the adapter to communicate with a specific instance of an application (this can be referred to as metadata or as connection details). For example, to create a connection to a specific RightNow Cloud application instance, you must select the Oracle RightNow adapter and then specify the WSDL URL, security policy, and security credentials to connect to it.

Video

Connection Creation

You can create a connection based on any of the following adapters to which you are subscribed.

- Oracle RightNow Cloud
- Oracle Sales Cloud
- Oracle Messaging Cloud Service
- Oracle Eloqua Cloud
- Oracle HCM Cloud

- Salesforce
- Oracle ERP Cloud
- Oracle CPQ Cloud
- SOAP Adapter
- REST Adapter
- NetSuite Adapter
- Integration Cloud Service Messaging
- Eventbrite (in *Using the Eventbrite Adapter*)
- Evernote (in *Using the Evernote Adapter*)
- Facebook (in *Using the Facebook Adapter*)
- FTP (in *Using the FTP Adapter*)
- Gmail (in *Using the Gmail Adapter*)
- Google Calendar (in *Using the Google Calendar Adapter*)
- Google Task (in *Using the Google Task Adapter*)
- LinkedIn (in *Using the LinkedIn Adapter*)
- Microsoft Calendar (in *Using the Microsoft Calendar Adapter*)
- Microsoft Contact (in *Using the Microsoft Contact Adapter*)
- Microsoft Email (in *Using the Microsoft Email Adapter*)
- MailChimp (in *Using the MailChimp Adapter*)
- MySQL (in *Using the MySQL Adapter*)
- SAP (in *Using the SAP Adapter*)
- Oracle Commerce Cloud (in *Using the Oracle Commerce Adapter*)
- Oracle Database (in *Using the Oracle Database Adapter*)
- Oracle E-Business Suite (in *Using the Oracle E-Business Suite Adapter*)
- Oracle Siebel (in *Using the Oracle Siebel Adapter*)
- SurveyMonkey (in *Using the SurveyMonkey Adapter*)
- Twitter (in *Using the Twitter Adapter*)

Oracle RightNow Cloud

The Oracle RightNow Cloud Adapter enables you to create an integration with an Oracle RightNow Cloud application.

Oracle RightNow Cloud applications enable organizations to combine web, social network, and contact center customer experiences into a unified, cross-channel service solution in Oracle Cloud. Oracle RightNow Cloud provides the following benefits:

- Integrates easily with the Oracle RightNow application's WSDL file to produce a simplified, integration-centric WSDL.
- Generates automatic mapping to the exposed business object or event subscription that you select during adapter configuration:
  - Business object: Represents a self-contained business document that can be acted upon by the integration. An integration can send requests to create a new record for that business object. They can send a request either to update or delete an existing record for a business object. Integrations can also send requests to retrieve information about one or more records representing that business object.
  - Event subscription: Represents an event document to which you subscribe when the Oracle RightNow Cloud Adapter is configured in the trigger (source) direction. The event subscription is raised by the Oracle RightNow application.
- Supports the RightNow Object Query Language (ROQL) to query metadata information when the Oracle RightNow Cloud Adapter is configured in the invoke direction.
- Automatically handles security policy details required to connect to the Oracle RightNow Cloud application.
- Provides standard error handling capabilities.
- Enables you to perform CRUD (create, get, update, and destroy) operations against business objects in the Oracle RightNow Cloud application.

For more information, see Configuring Oracle RightNow Cloud Properties.

Oracle Sales Cloud Adapter

Oracle Sales Cloud Adapter enables modern selling with tools that are easy to deploy and use, completely mobile, packed with powerful analytics, and built for collaborative selling and revenue generation. Oracle Sales Cloud includes a set of features for creating and tracking sales campaigns, developing leads into business opportunities, and pursuing opportunities to generate revenue. Sales accounts, leads, and opportunities can be automatically assigned to territories and sales teams.

The Oracle Sales Cloud Adapter enables you to create an integration with an Oracle Sales Cloud application.

The Oracle Sales Cloud Adapter enables customers to easily integrate their on-premises or SaaS applications with Oracle Sales Cloud without having to know the specific details involved in the integration.

The Oracle Sales Cloud Adapter provides the following benefits:

- Generates a local integration-centric WSDL that is an abstract WSDL. It defines strongly-typed message structures (request and response types) for the selected objects and the name of operations. It provides a simplified user experience in terms of creating data mappings at design time while constructing integrations with Fusion Application services. This WSDL also exposes the generic custom objects as named objects.
- Generates automatic mapping to the exposed business object, event subscription, or Oracle Fusion Applications REST API resource that you select during adapter configuration:

  - Business object: Represents a self-contained business document that can be acted upon by the integration. An integration can send requests to create a new record for that business object. They can send a request either to update or delete an existing record for a business object. Integrations can also send requests to retrieve information about one or more records representing that business object.

  - Event subscription: Represents an event document to which you subscribe. The event subscription is raised by the Oracle Sales Cloud application.

- Enables you to view annotations on Oracle Sales Cloud Adapter elements in the mapper. For information about viewing annotations in the mapper, see About Mappings in *Using the Oracle Mapper*.

- Automatically handles security policy details required to connect to the Oracle Sales Cloud application.

- Provides standard error handling capabilities.

- Enables you to map business objects that have polymorphic data structures.

For more information, see Configuring Oracle Sales Cloud Properties.

Oracle Messaging Cloud Service

The Oracle Messaging Cloud Service Adapter enables you to create an integration with the Oracle Messaging Cloud Service.

This enables you to interact with Oracle Messaging Cloud Service queues and topics hosted in Oracle Public Cloud. You can perform the following tasks against Oracle Messaging Cloud Service queues and topics:

- Send and receive messages to and from queues

- Create durable subscriptions for topics

- Subscribe to and consume messages from topics

- Send messages to topics

For more information, see Configuring Oracle Messaging Cloud Service Properties.

Oracle Eloqua Cloud

The Oracle Eloqua Cloud Adapter enables you to create an integration with an Oracle Eloqua Cloud application.

Oracle Eloqua Cloud provides cloud-based marketing software that delivers marketing automation for industries such as finance, health, media, real estate, sports, entertainment, and so on. Oracle Eloqua Cloud simplifies marketing procedures by delivering leads targeted to new customers. Oracle Eloqua Cloud helps manage and organize marketing segmentation and marketing workflow.

The Oracle Eloqua Cloud Adapter accelerates integrations between customer relationship management (CRM) and Oracle Eloqua Cloud systems. The Oracle Eloqua Cloud Adapter synchronizes accounts, contacts, and custom objects from CRM to Oracle Eloqua Cloud systems.

For more information, see Configuring Oracle Eloqua Cloud Properties.

Oracle HCM Cloud

The Oracle HCM Cloud Adapter enables you to create an integration with Oracle Human Capital Management (HCM) Cloud applications. You select business objects that an integration receives from Oracle HCM Cloud as a request and as a response.

The Oracle HCM Cloud Adapter enables customers to easily integrate their on-premises or SaaS applications with Oracle HCM Cloud without having to know about the specific details involved in the integration.

The Oracle HCM Cloud Adapter provides the following benefits:

- Integrates easily with the Oracle HCM Cloud application's WSDL file to produce a simplified, integration-centric WSDL.
- Generates automatic mapping to the exposed business object or event subscription that you select during adapter configuration:
  - Business object: Represents a self-contained business document that can be acted upon by the integration. An integration can send requests to create a new record for that business object. They can send a request either to update or delete an existing record for a business object. Integrations can also send requests to retrieve information about one or more records representing that business object.
  - Event subscription: Represents an event document to which you subscribe. The event subscription is raised by the Oracle HCM Cloud application.
- Automatically handles security policy details required to connect to the Oracle HCM Cloud application.
- Provides standard error handling capabilities.
- Enables you to map business objects that have polymorphic data structures.

For more information, see Configuring Oracle HCM Cloud Properties.

Salesforce Adapter

The Salesforce third party-created adapter enables you to create an integration with Salesforce CRM applications.

The Salesforce Adapter provides the following benefits:

- Integrates easily with the Salesforce application's WSDL file to produce a simplified, integration-centric WSDL.
- Contacts the Salesforce application to fetch metadata information about business objects.
- Provides invoke (outbound) support for performing the following type of operations against business objects fetched from the Salesforce application:
  - CRUD (create, get, update, and destroy) operations
  - Salesforce Object Query Language (SOQL) or Salesforce Object Search Language (SOSL) query operations
- Provides trigger (source/inbound) messaging support for objects through use of the Salesforce outbound messaging WSDL.

- Provides trigger (source/inbound) callback support.
- Provides metadata caching support. For information, see Refreshing Integration Metadata.

For more information, see Configuring Salesforce Properties.

Oracle ERP Cloud

The Oracle ERP Cloud Adapter enables you to create an integration with Oracle Enterprise Resource Planning (ERP) applications.

Oracle ERP Cloud enables you to streamline your enterprise business processes. Oracle ERP Cloud combines the power of a global business software suite with transactional and reporting capabilities that address financial, procurement, and project requirements across a range of business flows and countries. The complete and integrated suite transforms your organization through the latest social, mobile, and analytic technologies to optimize collaboration, process efficiency, and compliance while increasing insight into the business.

The Oracle ERP Cloud Adapter enables you to easily integrate on-premises or SaaS applications with Oracle ERP Cloud without having to know about the specific details involved in the integration. The Oracle ERP Cloud Adapter provides the following benefits:

- Integrates easily with the Oracle ERP Cloud application's WSDL file to produce a simplified, integration-centric WSDL.
- Generates automatic mapping to the exposed business object or event subscription that you select during adapter configuration:
  - Business object: Represents a self-contained business document that can be acted upon by the integration. An integration can send requests to create a new record for that business object. They can send a request either to update or delete an existing record for a business object. Integrations can also send requests to retrieve information about one or more records representing that business object.
  - Event subscription: Represents an event document to which you subscribe. The event subscription is raised by the Oracle ERP Cloud application.
- Automatically handles security policy details required to connect to the Oracle ERP Cloud application.
- Provides standard error handling capabilities.
- Enables you to map business objects that have polymorphic data structures.

For more information, see Configuring Oracle ERP Cloud Properties.

Oracle CPQ Cloud

The Oracle Configure, Price, and Quote (CPQ) Cloud Adapter enables you to create an integration with an Oracle CPQ application.

The Oracle CPQ Cloud Adapter enables you to convert sales opportunities into revenue by automating the quoting and sales order process with guided selling, dynamic pricing, and a workflow approval process.

Oracle CPQ cloud extends sales automation to include the creation of an optimal quote, which enables sales personnel to configure and price complex products; select

the best options, promotions, and deal terms; and include upsell and renewals, all using automated workflows.

The main use case for Oracle CPQ Cloud is as the trigger (source) in an integration in which Oracle Sales Cloud is the invoke (target). This adapter replicates the point-to-point integration that exists today between Oracle CPQ Cloud and Oracle Sales Cloud. Oracle CPQ Cloud is the trigger (source) of the record application. A synchronize process is triggered as you update and save data in Oracle CPQ Cloud, but it can also be configured by the administrator of the application. The Oracle CPQ Cloud Adapter can also be configured as the invoke (target) in an integration.

Prebuilt integration flows with Oracle CPQ Cloud and Oracle Sales Cloud for quote creation, opportunity import, and quote update are also provided from the Oracle Marketplace.

For more information see Oracle CPQ Cloud and Configuring Oracle CPQ Cloud Properties.

SOAP Adapter

The SOAP Adapter enables you to connect to any SOAP web service.

The SOAP Adapter enables you to expose an Integration Cloud Service integration as a SOAP web service and to invoke an external SOAP web service. When creating a connection with the SOAP Adapter, you can either upload the WSDL file (using the **Upload** button) or provide a URL to the WSDL file. When you invoke the wizard to create a connection, the contents of the specified WSDL file are read and the port type, operation, and service to use are displayed. If your WSDL includes only a single service, port type, and operation, they are automatically selected. If the WSDL includes multiple services, port types, and operations, you can select the ones to use in your integration.

For more information, see Configuring SOAP Adapter Properties.

REST Adapter

The REST Adapter enables you to expose an Oracle Integration Cloud Service integration flow as a REST service and to invoke an external REST application.

The REST Adapter provides the following benefits:

- When configured as a trigger, it acts as a generic inbound REST Adapter for exposing an Integration Cloud Service integration flow as a REST resource. A client can connect to the integration using a REST endpoint.
- When configured as an invoke, it acts as a generic outbound REST Adapter for connecting to any external REST-exposed SaaS application.

The REST Adapter supports the following message exchange patterns:

- Synchronous request and response patterns
- Synchronous one-way request patterns

The REST Adapter supports the following security policies:

- Basic Authentication
- OAuth Client Credentials (two-legged flow)
- OAuth Resource Owner Password Credentials (two-legged flow)

- OAuth Authorization Code Credentials (three-legged flow)

See Configuring the Connection Security Policy for more information about these security policies.

The REST Adapter is not configured from an existing WSDL or WADL file. Instead, you configure the following parameters using the REST endpoint wizard to expose or consume the REST service:

- Relative resource path URI
- HTTP method (actions) to perform
- Template and query parameters
- Request/response message structure

The message in the integration flow is always in XML format. Messages sent to Integration Cloud Service through the REST Adapter in JSON or URL-encoded format are converted to XML. If messages sent from Integration Cloud Service through the REST Adapter are in XML message format, they can be converted to JSON or URL-encoded format, depending on the configured media type.

The REST Adapter provides design-time support for REST metadata catalog-compliant REST APIs. There is also top-level array support in JSON documents.

Note: The REST Adapter does not support multidimensional arrays.

NetSuite Adapter

The NetSuite Adapter enables you to create an integration with a NetSuite application.

NetSuite is a SaaS-based application for business management. The NetSuite platform includes ERP, CRM, PSA, and e-commerce capabilities. To integrate users, NetSuite provides a platform called SuiteCloud that consists of cloud development tools and infrastructure. The SuiteTalk component of the SuiteCloud framework enables integration of NetSuite with other on-premises or cloud solutions.

While SuiteTalk provides the ability to access NetSuite data and business processes through an XML-based API, it requires skills such as Microsoft .NET or Java to build integrations with it. The NetSuite Adapter addresses these requirements by providing a no-coding approach for building integrations with NetSuite. This enables users who are not professional developers to build integrations with NetSuite.

The NetSuite Adapter provides the following features:

- Quickly and easily connect on-premises systems and applications with NetSuite.
- Rapidly integrate with both cloud applications and with existing on-premises business systems.
- Automate the process for discovering NetSuite's web service WSDLs based on the user account.
- Eliminate the need to work with complex polymorphic data objects by elevating NetSuite records and custom objects (records and fields) to strongly-typed objects.
- Display records based on NetSuite's categorization.

- Provide available contextual information about business objects and operations to the developer at design time.
- Provide secured invocation to NetSuite's web services by adhering to the role-based permission structure enforced in NetSuite in a transparent fashion.
- Provide a standard adapter life cycle, controlled runtime environment, and monitoring capabilities.

For more information, see Configuring NetSuite Adapter Properties.

Integration Cloud Service Messaging

Integration Cloud Service Messaging enables you to publish messages to and subscribe to messages from Integration Cloud Service.

You may have business use cases in which you need to synchronize objects between applications. For example:

- Create an object in one application that causes the object to be created in other applications. For example, create a new account in Oracle Sales Cloud Adapter, which causes the creation of an Oracle RightNow organization and an Oracle Eloqua account.
- Enable multiple applications to subscribe to Integration Cloud Service and register for updates.
- Add or remove subscribers without impacting other subscribers or producers.

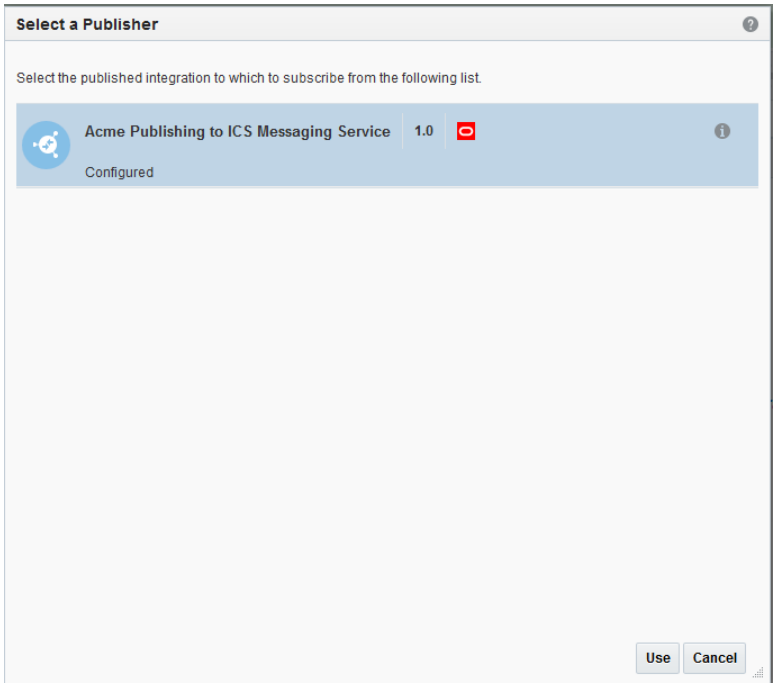
Integration Cloud Service Messaging addresses these business requirements through the creation of two types of integrations: one for publishing to Integration Cloud Service and one for subscribing to Integration Cloud Service.

- You create an integration that enables you to publish messages to Integration Cloud Service by selecting the **Publish to ICS** option in the Create Integration — Select a Pattern dialog. In this integration:
  - Integration Cloud Service is added as an invoke and is automatically configured.
  - You configure a trigger (source) adapter (for example, Oracle RightNow, Oracle Sales Cloud, or another).
  - The message to pass to Integration Cloud Service is opaque, so no request mapper support is provided.
  - No trigger (source) enrichment mapper support is provided.
  - Multiple publishers targeting a single message destination is not supported.

  Note: Modifying the publisher after creating the subscribers can potentially impact the subscribers. For example, if you change the published object, any existing subscriber mappings are impacted.

- You create an integration that enables you to subscribe to messages from Integration Cloud Service by selecting the **Subscribe to ICS** option in the Create Integration — Select a Pattern dialog. In this integration:
  - Integration Cloud Service is added as a trigger (source).

- You are prompted to select the published integration to which to subscribe.



- You configure an invoke adapter to subscribe to and receive messages from Integration Cloud Service.
- Response mapper support is provided between the published object and the subscriber's application object.
- Trigger (source) enrichment mapper support is provided.

For more information, see Creating an Integration to Publish Messages to Integration Cloud Service and Creating an Integration to Subscribe to Integration Cloud Service.

About Oracle Integration Cloud Service Integrations

Integrations are the main ingredient of Oracle Integration Cloud Service. An integration includes at the least a trigger (source) connection (for requests sent to Oracle Integration Cloud Service) and invoke (target) connection (for requests sent from Oracle Integration Cloud Service to the target) and the field mapping between those two connections.

When you create your integrations, you build on the connections you already created by defining how to process the data for the trigger (source) and invoke (target) connections. This can include defining the type of operations to perform on the data, the business objects and fields against which to perform those operations, required schemas, and so on. To make this easier, the most complex configuration tasks are handled by Oracle Integration Cloud Service. Once your trigger (source) and invoke (target) connections are configured, the mappers between the two are enabled so you can define how the information is transferred between the trigger (source) and invoke (target) data structures for both the request and response messages.

Video

Related Topics

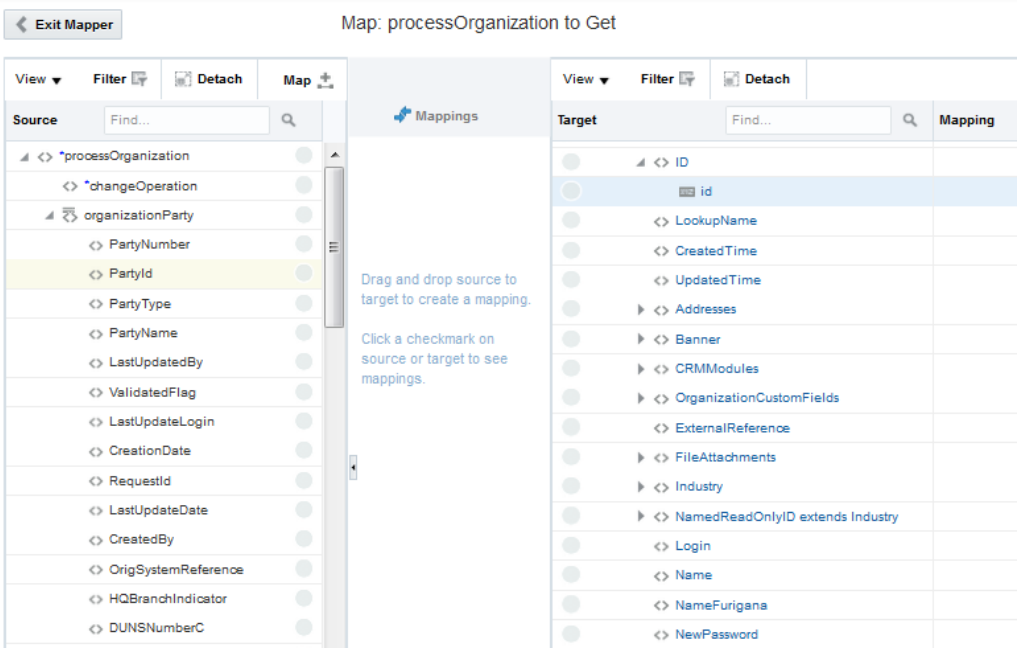
See the following sections for additional information.

- For more information about triggers (sources) and invokes (targets), see Adding a Source Connection and Adding a Target Connection.
- For information about creating a connection, see Creating a Connection.

About Mappings

One of the key tasks to any integration is defining how data is transferred, or *mapped*, between two applications.

In most cases, the messages you want to transfer between the applications in an integration have different data structures. A visual mapper enables you to map fields between applications by dragging source fields onto target fields. When you open the mapper for a request or response message in an integration, the data structures are automatically populated with the information pulled from the source and target connections. You can expand and the load data structure levels on demand to display additional levels. There is no limit on the levels of display.



The maps you create are called transformation maps, and use the eXtensible Stylesheet Language (XSL) to describe the data mappings, which lets you perform complex data manipulation and transformation. A standard set of XPath functions are provided for you to define how data is modified when moving from one application to another. A specialized function is also provided for you to reference lookups directly from the mapper.

The mapper supports both qualified and unqualified schemas (that is, schemas without `elementFormDefault="qualified"`). Elements and attributes with and without namespace prefixes are also supported.

Substitution groups in schemas are supported. You can see all the substitutable elements in a base element in the mapper, and select the one to use.

Mapping Request Data Between Applications

Once you create an integration and have the trigger (source) and invoke (target) in place, you can define how data is mapped between the two data structures.

The mapper appears with the source data structure on the left and the target data structure on the right:

1. Map request data between the source data structure and target data structure.
2. On the toolbar, click **Save**.

When returning from the mapper, the map icon changes color to indicate it is complete. Once you create a mapping in an integration, you can return to the mapping and make any necessary changes to how you mapped your data.

For procedural instructions about mapping request data between applications, see Mapping Data of *Using the Oracle Mapper*.

Mapping Response Data Between Applications

If your integration pattern contains a response, you can map the response.

1. Map response data between the source data structure and target data structure.
2. On the toolbar, click **Save**.

   When returning from the mapper, the map icon changes color to indicate it is complete.

   Once you create a mapping in an integration, you can return to the mapping and make any necessary changes to how you mapped your data.

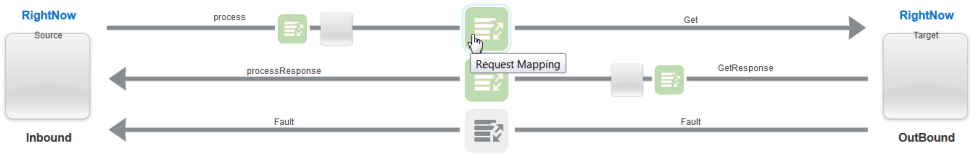
For procedural instructions about mapping response data between applications, see Mapping Data of *Using the Oracle Mapper*.

About Mapping Multiple Sources to a Target

When mapping data between source and target data structures, some integration scenarios enable you to map the fields of multiple source structures to the fields of a single target structure.

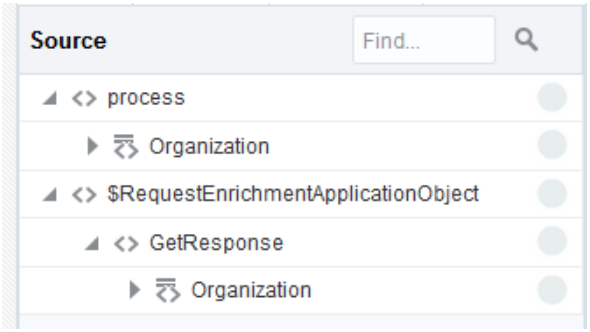
Integration scenarios that include multiple source structure capabilities include the following:

- Integrations in which message enrichment points have been added (for example, a request message enrichment point, a response message enrichment point, or both points). For example, within the context of the following inbound trigger connection to outbound invoke connection, request mappings and request enrichment mappings are both defined.

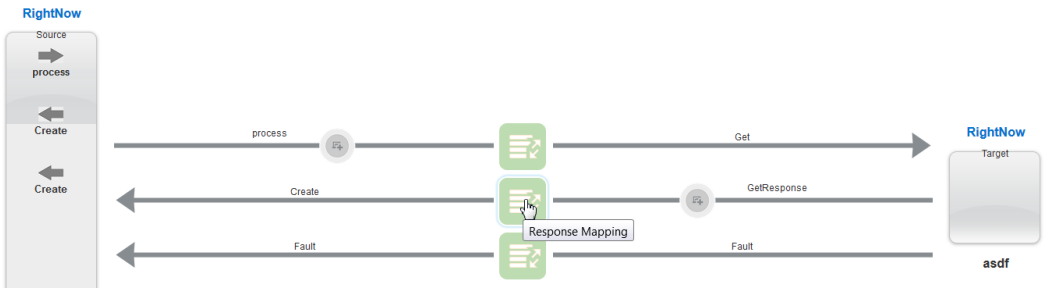


Clicking the **Request Mapping** icon shows that there are two sources available for mapping in the **Source** section. The **process** structure is the primary source. The

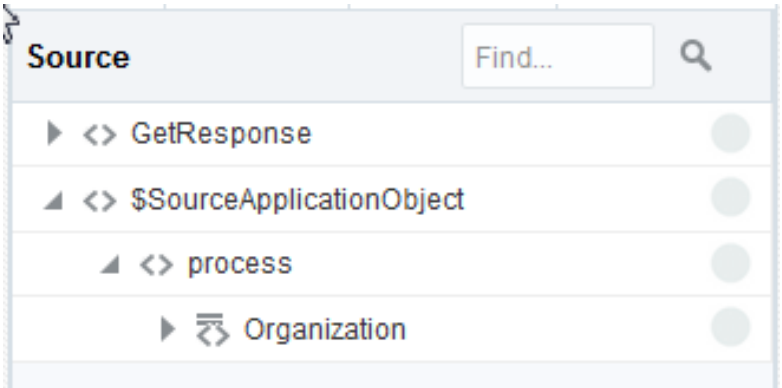
$RequestEnrichmentApplicationObject structure is the secondary source. Secondary sources are treated as variables and identified by the **$** added to the front. The fields of both sources can be mapped to the fields of the target.



- Integration responses with a response mapping between a trigger connection and an invoke connection. For example, within the context of the invoke connection's response back to the trigger connection, there are response mappings.



Clicking the **Response Mapping** icon shows the two sources available for mapping in the **Source** section. The **GetResponse** structure is the primary source. The **$SourceApplicationObject** structure is the secondary source (note the **$**). The fields of both sources can be mapped to the fields of the target.



For more information, see Mapping Multiple Sources to a Target of *Using the Oracle Mapper*.

About Integration Cloud Service Enrichments

You may have business use cases in which you need to enhance data by calling another service *before* sending data to an invoke service or *before* sending data back to a requestor. To address this business requirement, you can optionally add enrichment data sources to the request part, the response part, or both parts of an integration. Enrichments participate in the overall integration flow and can be used in the request and/or response payloads between the trigger and invoke services. Enrichments subscribe to a synchronous request and response message pattern.

Enrichments enable you to:

- Add additional information. For example, your business use case may require you to:
  - Add a stock price
  - Increase on-site quantities of a product
  - Estimate local currency
- Convert data, such as mapping data between account numbers. The ability to map data between the request/response payload and the enrichment source application is a key feature of enrichments.

For information about using enrichments, see Adding Request and Response Enrichments.

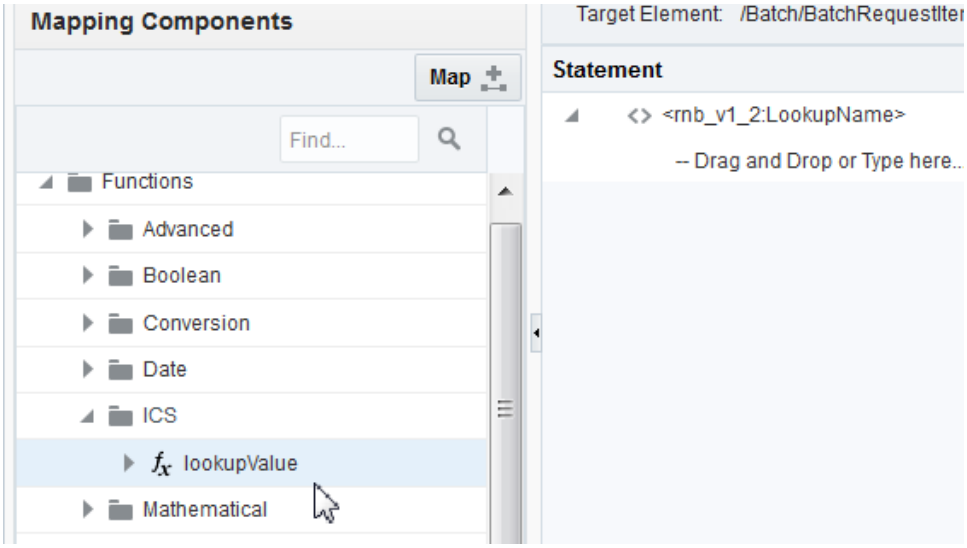
About Integration Cloud Service Lookups

Use lookups in your integrations to create reusable tables that map the different terms used to describe the same item across your applications.

A lookup associates values used by one application for a specific item to the values used by other applications for the same item. For example, one application uses a specific set of codes to describe countries, while another application uses a different set of codes to describe the same countries. Lookups can be used for items such as mapping gender codes, nationality codes, currency codes—any type of information that your applications must share with each other but that they represent differently. You may have several lookups for one integration, depending on the number of fields that require mapping. Lookups are also reusable, and can be used by multiple integrations. Lookups are based on a static definition, meaning you create and populate them during design time, and are not changed by runtime activities. These tables are used for looking up values only.

Lookup Function

Integration Cloud Service provides a lookupValue function that you can call in the mapper to specify when to reference a lookup table. Use this function to look up values at runtime based on information in incoming messages. This way, your integration knows how to map data coming in from one application to data being sent to another application.



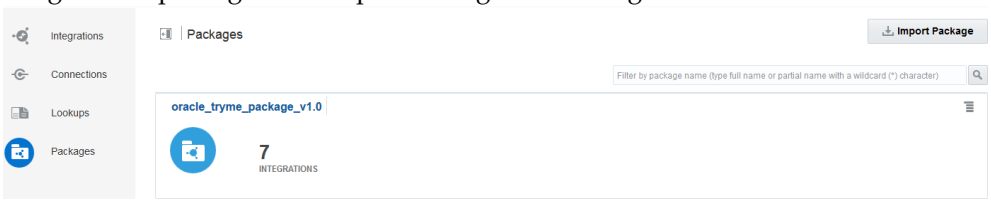
For more information, see Referencing Lookups of *Using the Oracle Mapper*.

Video

About Integration Cloud Service Packages

You can group one or more integrations into a single structure called a package. Packages enable you to easily import and export a group of integrations to and from Integration Cloud Service. You can import packages from the Oracle Marketplace. These packages consist of a series of prebuilt integrations provided by Oracle. You can also import and export packages that consist of integrations that you or other users created. Packages are optional, meaning that integrations do not need to be part of packages. However, for a package to exist, it must include at least one integration. Packages cannot be locked to exclude other users of your Integration Cloud Service instance.

Packages are displayed on the Packages page in Integration Cloud Service. From this page, you can view, delete, import, and export packages. You create packages when you create an integration in the Create Integration dialog. You can also update an integration's package in the Update Integration dialog.



For more information, see Managing Packages and Creating an Integration.

About Agents and Integrations Between On-Premises Applications and Oracle Integration Cloud Service

Oracle Integration Cloud Service provides an agent framework that enables you to create integrations and exchange messages between on-premises applications and Oracle Integration Cloud Service. Message payloads of up to 5 MB are supported through the use of compression, which may bring the payload down to 512 KB in size. The on-premises agent provides multithreading support, which allows for multiple executors to perform downstream message processing.

This type of integration enables you to:

- Access SOAP endpoints
- Access non-SOAP endpoints (such as Oracle E-Business Suite and Oracle Siebel)
- Send requests from a cloud application (for example, send a Create Service Order request from an Oracle RightNow Cloud application) to an on-premises E-Business Suite application

The agent framework consists of the following components:

- SAAS agent: This agent is installed and runs in Oracle Integration Cloud Service and supports communication with on-premises applications. There is one SAAS agent per Oracle Integration Cloud Service environment.
- On-premises agent: This agent is installed and runs in an on-premises environment on the same network as internal systems such as Oracle E-Business Suite, Oracle Siebel, Oracle Database, and others. You download the on-premises agent installer from the Agents page in Oracle Integration Cloud Service to your on-premises environment for installation. Multiple agents can run on a single host. There can be multiple host systems, each running one or more agents, in a cloud/on premises topology. The on-premises agent does not permit any explicit inbound connections. All connections are established from the on-premises environment to Oracle Integration Cloud Service. This functionality means that:
  - No ports are opened on the on-premises system for communication.
  - All communications is secured using SSL.
  - The on-premises agent registers with Oracle Integration Cloud Service over SSL using the provided Oracle Integration Cloud Service credentials.
  - The on-premises agent checks for work by making outbound requests through the firewall.
  - The on-premises agent can use a proxy to access the internet (the same proxy as other internal applications and browsers use).
  - The on-premises agent connections are configured by the agent retrieving the configuration details from Oracle Integration Cloud Service.
  - The on-premises agent processes requests by pulling messages from Oracle Integration Cloud Service across SSL.
  - The on-premises agent posts responses by pushing messages to Oracle Integration Cloud Service across SSL.
  - All communication is initiated by the on-premises agent.
  - No private SOAP-based web services are exposed.
  - No existing J2EE container is required to deploy the on-premises agent.
- Oracle Messaging Cloud Service: This service handles message exchange between Oracle Integration Cloud Service and on-premises environments.
- Outbound adapters: The following adapters can be configured as invoke connections in an integration to support communication with endpoint applications:

- MySQL Database
- Oracle Database
- Oracle E-Business Suite
- SAP
- Siebel
- SOAP

Workflow

The following table describes the workflow for installing and using the on-premises agent in an integration. After completing these tasks, you can invoke the integration.

Task	Documentation
Create an agent group.	Creating an Agent Group
Download and run the on-premises agent installer on your host. During installation setup, you associate the on-premises agent with the agent group.	Downloading and Running the On-Premises Agent Installer
Create an adapter connection in Oracle Integration Cloud Service and associate the connection with the agent group.	Creating a Connection with an Agent Group
Design an integration that uses this connection.	Creating an Integration
Activate the integration.	Activating an Integration

For more information, see Managing Agent Groups and the On-Premises Agent and Monitoring Agents.

Video

About Monitoring

The Integration Cloud Service dashboard displays information about the current state of all your running integrations.

The dashboard gives you a quick view into the performance metrics for all of your active integrations. The graph includes the total number of messages processed, the average processing time, the number of messages with errors, and the success rate. The Integrations tab lists metrics for each running integration, and you can view a graph of the metrics for each integration in the list. The dashboard also provides a view of recent activity for running integrations and the ability to download all activities.

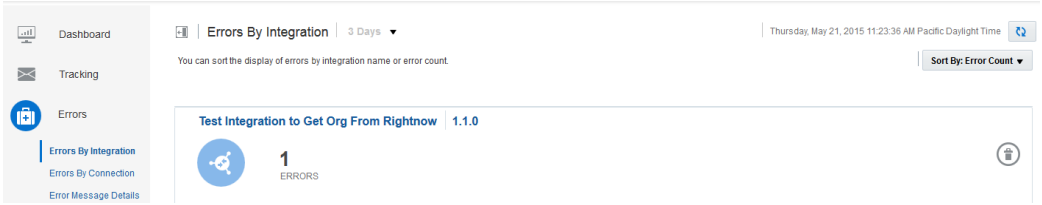
Video

About Error Management

You can manage integration errors from the Errors pages in Integration Cloud Service. The Errors pages display information about individual integration instances and group the errors by integrations and connections over a specified time period.

You can perform the following tasks from the Errors page and its subpages:

- Search for and display errors by integration name or the total error count over a specific time period
- Resubmit errors
- Discard (remove) errors by integration name
- Display errors by connection name or the total error count over a specific time period
- Discard errors by connection name
- Display errors by integration instance identifier, error location, or time of occurrence over a specific time period
- View the instance in which errors occurred
- View the error message
- View and discard errors by instance ID
- View the audit trail and message payload of a failed integration instance
- View the business identifiers of a failed integration instance



Integration Failure Scenarios

Integrations can fail for the following reasons:

- A call to a target system fails because the target system is down (for a short or long time period).
- A source-to-target transformation or target-to-source transformation fails because of faulty XSL coding, an invalid lookup call, or other system issue.
- A target system call results in a business failure for the following possible reasons:
  - Incorrect target application configuration
  - Invalid lookup data
  - Invalid business data
- A call to a trigger system (with a response message) fails because the trigger system is down.

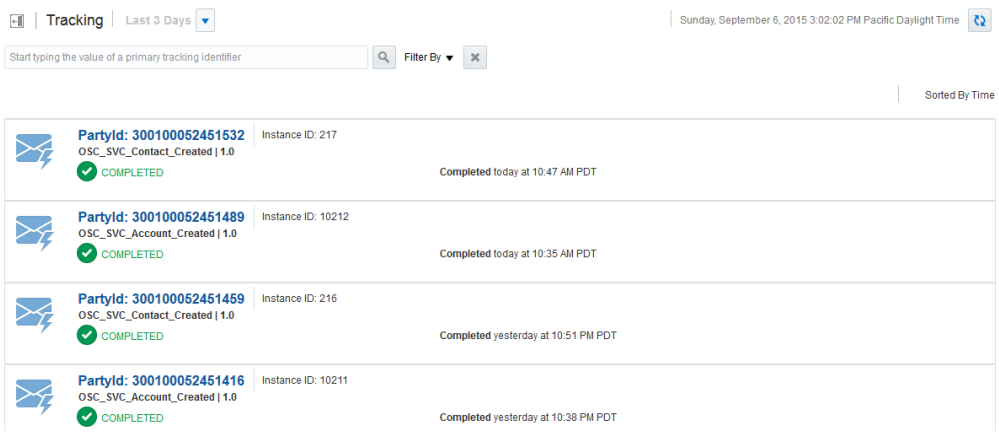
- An enrichment step fails.
- A publish/subscribe scenario fails.
- Any other Integration Cloud Service system failure.

For more information about error management, see Managing Errors.

About Business Identifiers for Tracking Fields in Messages

Business identifiers enable you to track payload fields in messages during runtime. You define up to three business identifiers on payload fields during design time. You designate one field as the primary business identifier field, which enables message fields to be tracked during runtime.

During runtime, the Tracking page displays information about the status of business identifiers and their values in your integrations.



If you have defined business identifiers in integrations that have failed, you can view details on the Errors page.

For more information, see Assigning Business Identifiers for Tracking Fields in Messages, Managing Business Identifiers for Tracking Fields in Messages, and Managing Errors.

Starting Integration Cloud Service

To get started with Integration Cloud Service, you must have a user account already set up. You access Integration Cloud Service through a web browser.

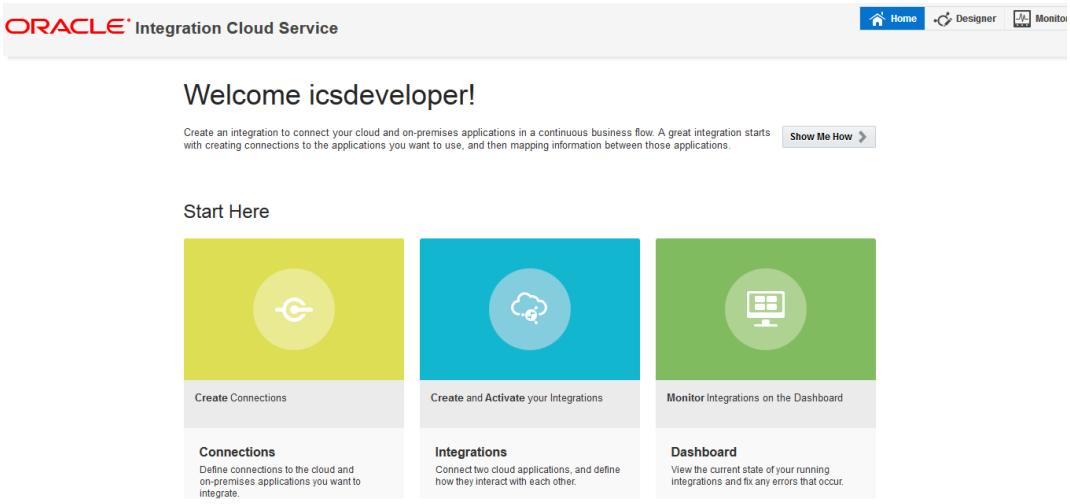
To access Integration Cloud Service:

1. In a web browser, enter the following URL, where *hostname* is the name of the host you received in your email after provisioning Oracle Integration Cloud Service, and *port_number* is the port number on which the server is listening (by default, `7001`):

   `https://hostname:port_number/ics`

2. On the login page, enter your user name and password.

The first page to appear is the home page, which illustrates the typical development workflow and provides links both to the functions of Integration Cloud Service and to sources of additional information about each function.



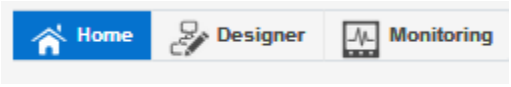
Navigating Integration Cloud Service

Integration Cloud Service provides multiple ways to access the different features you use to create and monitor your integrations.

From the Integration Cloud Service main menu, you can access the home page, the designer portal, and the dashboard.

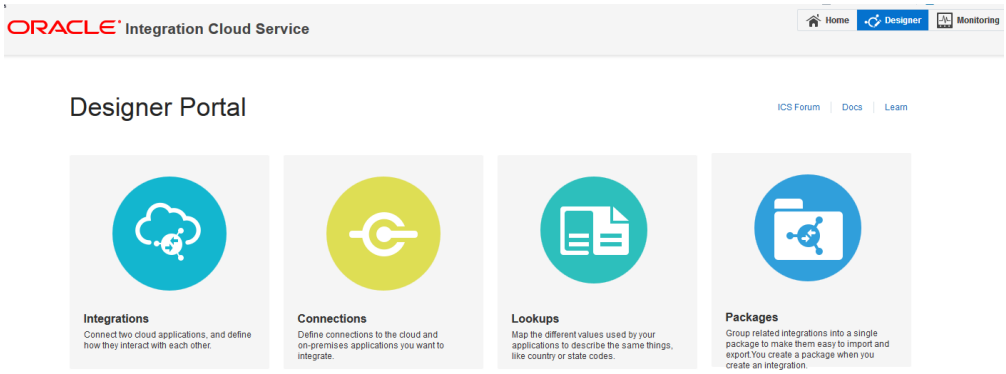
Navigating from the Home Page

The first page you see when you log in to Integration Cloud Service is the home page. This page provides links to more information and demonstrations for each feature. You can also access the Connections page, the Integrations page, the Dashboard page, the Lookups page, and the Packages page from the home page. To return to the home page from any other Integration Cloud Service page, click **Home** in the main menu.



Navigating from the Designer Portal

The designer portal provides links to the primary development features of Integration Cloud Service—integrations, connections, lookups, and packages. To access the designer portal from any other Integration Cloud Service page, click **Designer** in the main menu.



Navigating from the Designer Menu

The main Connections, Integrations, Lookups, and Packages pages list the resources you create and also include a menu that gives you access to the other development features. The menu also provides options to filter the current list of resources by their status.

About Oracle Integration Cloud Service Roles and User Accounts

Oracle Integration Cloud Service uses roles to control access to tasks and resources. A role assigned to a user gives certain privileges to the user.

The following table summarizes the responsibilities of each role in the context of Oracle Integration Cloud Service.

Role	Privileges Provided By This Role
Oracle Integration Cloud Service Users Role	Enables you to access all parts of Oracle Integration Cloud Service to perform the following tasks: • Create, deploy, and monitor integrations. • Upload security certificates. You can assign this role to developers.
Oracle Integration Cloud Service Monitors Role	Enables you to access Oracle Integration Cloud Service to monitor integrations. Only the Monitoring Dashboard is available with this user role. Note the following restrictions with this role: • If you click the **Integrations**, **Connections**, **Lookups**, **Packages**, or **Agents** icons on the home page, a `User is not authorized to perform this action` message is displayed. • The **Administration** tab in the upper right corner that enables you to upload security certificates is not visible.
Oracle Integration Cloud Service Runtime Role	Enables you to access runtime services. This role is assigned to a user and uses SOAP/REST to communicate with services at runtime. This role only has privileges to execute a flow. With this user, you can pass the user name and password for this user to invoke SOAP/REST APIs. A user with this role can log in, but cannot perform any actions in Oracle Integration Cloud Service, and receives a `User is not authorized to perform this action` message on the home page.

2

Setting Up Oracle Integration Cloud Service

You must perform the following tasks to set up Oracle Integration Cloud Service in an Oracle Public Cloud Machine environment.

Topics

Before You Begin Setting Up Oracle Integration Cloud Service

During Oracle Public Cloud Machine installation, Oracle engineers work with you to set up and create a database instance for Oracle Cloud Machine. To provision Oracle Integration Cloud Service, an additional schema must be provisioned in the same database.

Installing an External Database

The external database that is required for Oracle Public Cloud Machine is also used as the database for Oracle Integration Cloud Service (as described in the *Customer Deployment Guide*). The Oracle Public Cloud Machine installation process includes configuring the cloud infrastructure and PaaS services.

For Oracle Integration Cloud Service instance user provisioning, you must create the required schema and tablespace in the database. This database information is needed when you create a new Oracle Integration Cloud Service instance.

Creating the Database Schema and Tablespace

After the Oracle Public Cloud Machine Configuration Utility (ECU) has been executed, you must create the schema and tablespace in the external database.

1. In SQL*Plus, connect to the database AS SYSDBA:

   ```
   CONNECT SYS AS SYSDBA
   ```

2. Create the schema and tablespace in the Oracle database using the following commands. For this example, `ics_pod` is the user created. Enter values appropriate to your environment.

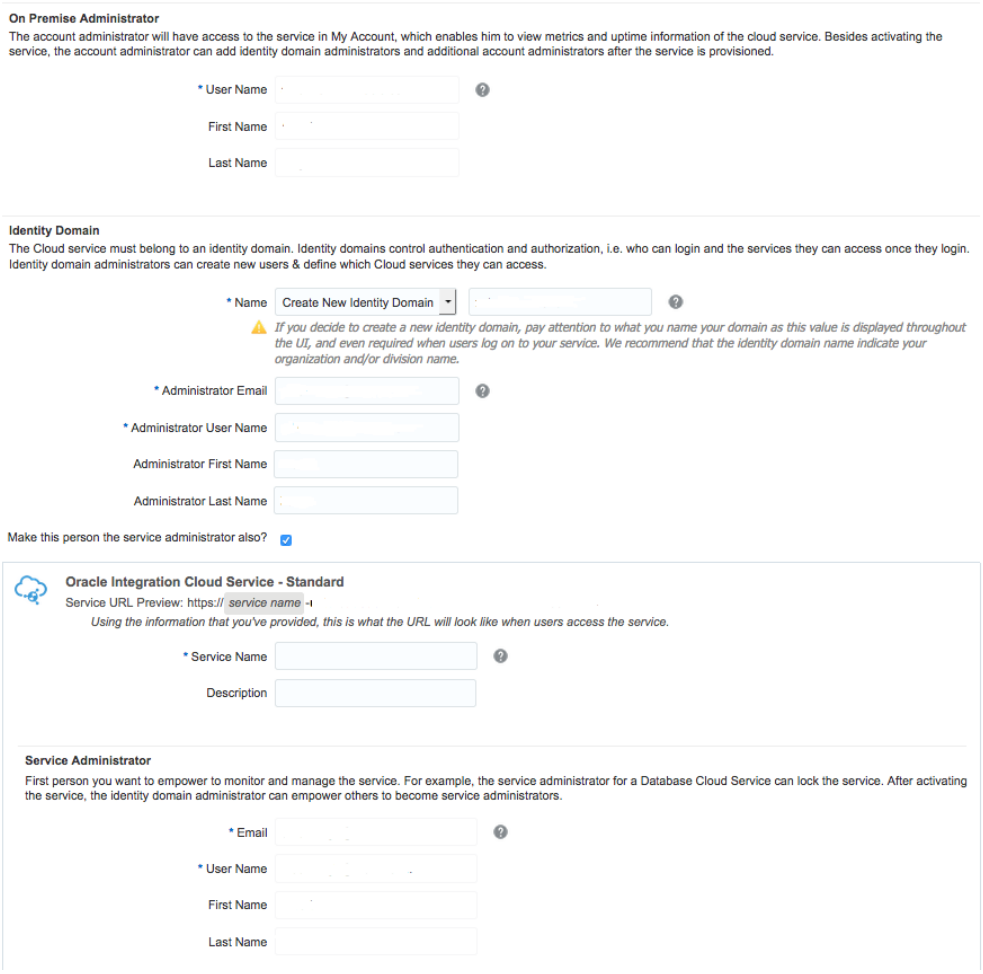
```
create bigfile tablespace ics_pod datafile 'ics_pod.dbf' size 10G;

create user ics_pod identified by password default tablespace ics_pod;
```

Note: The database schema password must be of minimum of 10 characters, and can contain only alphanumeric characters and underscores (_).

Assigning Administrators

When you request a service instance, there are three types of administrators you can assign.



The three administrators are described as follows in order of power within the service instance:

- On Premise Administrator: This user is the account administrator. This administrator has access to the service in My Account, which enables them to view metrics and uptime information about the cloud service. Besides requesting and activating the service, the account administrator can add identity domain administrators and additional account administrators after the service is provisioned. There are two types of on-premise administrators: The default (Cloud Operations) and customer on-premises administrator.

- Identity Domain Administrator: This user is the administrator of the identity domain. The cloud service that the On Premise Administrator creates must belong to an identity domain. Identity domains control authentication and authorization (that is, who can log in and the services they can access once logged in). Identity domain administrators can create new users and define which cloud services they can access. For subsequent instance requests, you can reuse previous domains or create additional domains.
- Service Administrator: This user is the administrator for the actual service. The is the first user that you want to empower to monitor and manage the service. For example, the service administrator for an Oracle Database Cloud Service can lock the service. After activating the service, the identity domain administrator can empower others to become service administrators.

  This administrator can also be the same as the domain administrator.

Managing On-Premises Administrators

On-premises administrators can request for instances, create identity domain administrators and service administrators for the subscribed services, and create and manage other on-premises administrators. Contact the Oracle Cloud Administration for assistance with the workflow described in this section.

Topics

Note:

- If you do not have on-premises administrator privileges, contact the Oracle Cloud Administrator to request that level of access.
- The default on-premises administrator is the role of Oracle Cloud Operations. You must request Oracle Cloud Operations to create a new on-premises administrator through a My Oracle Support (MOS) Change Request.

About On-Premises Administrators

A default on-premises administrator is created when Oracle Public Cloud Machine is installed and configured. The default on-premises administrator is the role of Oracle Cloud Operations. You must request Cloud Operations to create a new on-premises administrator through a MOS Change Request.

The default on-premises administrator can use the Cloud Portal provided by Oracle Public Cloud Machine to add, modify, or delete other Oracle Public Cloud Machine on-premises administrators. An on-premises administrator can request an instance for provisioning a particular service, such as Oracle Integration Cloud Service.

Resetting the Password for an On-Premises Administrator

Use the Cloud Portal to reset passwords for on-premises administrators.

1. Log in to the Cloud Portal as a user with on-premises administrator privileges.
2. Click the **On Premise Administrators** tab.
3. Navigate to the required on-premises administrator entry, click the **Action** icon to the right of the entry, and then click **Reset Password/Unlock Account**.

   The Reset Password /Unlock Account dialog is displayed.
4. Click **Reset**.

The on-premises administrator receives an email with the new password.

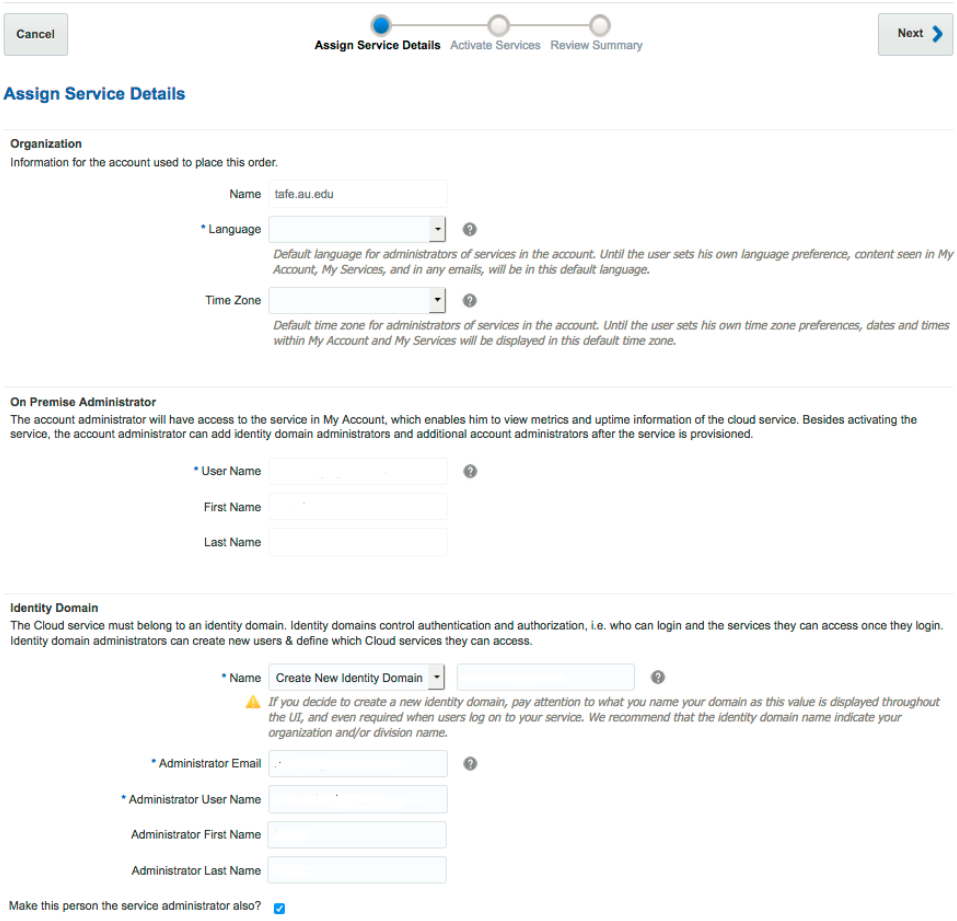
Requesting an Instance

You can create one or more service instances from the My Services dashboard. Only an on-premises administrator can request service instances. Service administrators manage the users in the identity domain. Instance administrators can only view the metrics and the state of service instances.

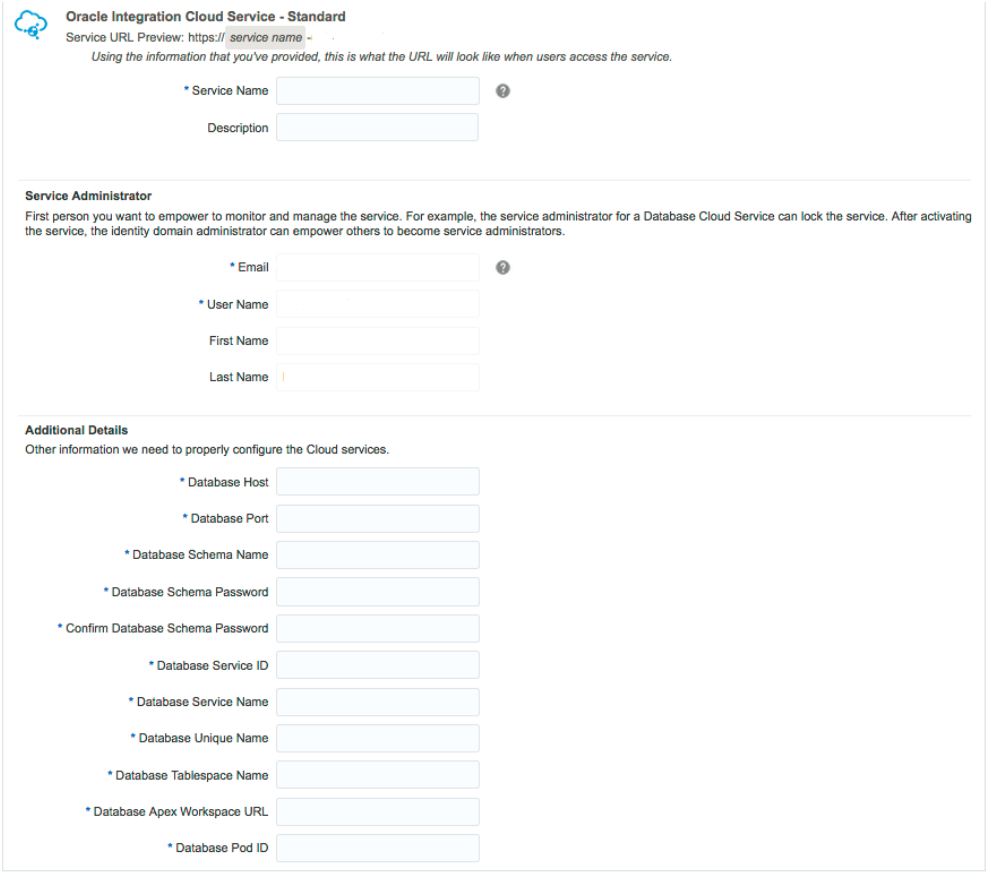
1. Log in to the Cloud Portal as an on-premises administrator.
2. Click **Request Instance** and enter the following details for the instance:
   - Enter the organization name.
   - Select **Oracle Integration Cloud Service** in the **Service Type** list.
3. Click **Request Instance** to initiate the request.

   A message is displayed indicating that the request has been successfully submitted and that additional steps must be performed to activate the request to send the email and password.
4. Click **Done**.
5. Click the **Instance Request** tab for the activation request.
6. Click **Activate** to initiate the instance creation.

   The Instance Creation wizard opens.
7. In the Assign Service Details step, select the **Language** and **Time Zone** values in the **Organization** section.



8. In the **Identity Domain** section, either accept the default generated domain name or provide a new name for the domain.
9. Enter the identity domain administrator values for the **Email**, **User Name**, **First Name**, and **Last Name** fields.
10. Select the option **Use same administrator for services?** to add the identity domain administrator as the service administrator of the domain. Otherwise, you can provide another user name for the service administrator.
11. Enter the **Service Name** and **Description** values for Oracle Integration Cloud Service. The service name is used in the URL of Oracle Integration Cloud Service.



For subsequent instance creations, make an entry of the Integration Cloud Service URL in the DNS.

12. Enter the following external Oracle database details in the **Additional Details** section:

    - For the **Database Host** and **Database Port** fields, use the information that you populated in the configuration file of the external database setup script.

Element	Description
Database Host	The database host that is configured for the external database setup script. In case of a real-applications cluster (RAC) database, the host must be the IP address of the primary host of the RAC database.
Database Port	The database port of the external database that is configured for the external database setup script. By default, the port is `1521`.

Element	Description
Database Schema Name	In capital letters, the database schema that you created for Oracle Integration Cloud Service. For example, if you specified the values with the command executed in Before You Begin Setting Up Oracle Integration Cloud Service, specify `ICS_POD`.
Database Schema Password	The password associated with the schema (for example, the password you specified with the command executed in Before You Begin Setting Up Oracle Integration Cloud Service). **Note**: The Database Schema Password must be of minimum of 10 characters, and can contain alphanumeric characters and the special character underscore (_). The password must have at least one uppercase alphabet, number, and underscore (_).
Database Service ID	The database service ID (SID) that uniquely identifies your database. This field should be equal to the database unique name. You can connect to the database and use the following command to get the database service ID: `select instance_name from v $instance;`
Database Service Name	The name of the database service for Oracle Integration Cloud Service. This field may not be the same as the SID. You can connect to the database and use the following command to get the name of the database service: `select value from v$parameter where name like '%service_name%';`
Database Unique Name	The database instance name. This field should be equal to the SID. You can connect to the database and use the following command to get the instance name: `select instance_name from v $instance;`
Database Tablespace Name	In capital letters, the table that you created for Oracle Integration Cloud Service. For example, if you specified the values with the command executed in Before You Begin Setting Up Oracle Integration Cloud Service, specify `ICS_POD`.

Element	Description
Database Apex Workspace URL	The database host in the following URL format: *DB_Host*:8080/apex/
Database POD ID	Enter ics_pod.

13. Click **Next**

14. Review the service details and click **Activate** to start the instance creation.

Note: Instance creation takes approximately two hours to complete. Upon completion, you receive an email indicating that Oracle Integration Cloud Service is ready.

Accessing the Service Console

All users require the service console URL to access Oracle Integration Cloud Service.

Topics

About the Service Console URL

When requesting an instance, the details of the service (such as the service name) were provided. The service console URL is constructed as follows:

ICS_service_name-ICS_domain.integration.*domain*

For example:

```
testservice-testdomain.integration.example.com
```

The service name and identity domain name were described during the instance request. For more information, see Requesting an Instance.

Setting Up the Host

After the instance is provisioned, add a DNS entry for the new service.

For example:

```
192.0.2.100 testservice-testdomain.integration.example.com
```

As an alternative, you can create an entry in the `hosts` file of the client system from which you access the service console:

OTD_VMIP Service_Console_URL

See Before You Begin Setting Up Oracle Integration Cloud Service for more information about finding the OTD VM IP address.

Accessing the Service Console

You can access the service console and individual services.

Access the service console from the Cloud home page:

`https://external_OTD_VIP_hostname:6900/mycloud/faces/cloudHome.jspx`

You can also directly access a service using the following URL:

`https://service_name-domain_name.integration.us.oracle.com/ics`

Logging in to the Service Console

You must have the following details to log in to the service console:

- Identity domain: The identity domain that was specified during the instance request.
- User name: The email address of the user who requested the instance or the service administrator specified during the instance request.
- Password: A temporary password is sent by email to the user on completion of the instance request.

You are directed to change the password on the first login.

Deleting an Instance

If you need to delete a service instance, contact the Oracle Cloud Administrator. Only the Oracle Cloud Administrator has the permissions to perform this task.

3

Developing Integration Cloud Services

You develop integrations using a simple but robust set of components, including connections to the applications you are integrating, data mappings, and lookups that map similar objects across the applications with which you share data.

Topics

Typical Workflow for Creating Integration Cloud Services

You follow a very simple workflow to develop integrations in Integration Cloud Service. The only prerequisites for creating an integration are that the application connections you need are in place and that any lookups you want to use to map information between applications are created.

This table lists the workflow steps for integrations, and provides links to instructions for each step.

Step	Description	More Information
1	Create the connections for the applications you want to integrate. The connections can be reused in multiple integrations and are typically created by the administrator.	Creating a Connection

Step	Description	More Information
2	(Optional) Create lookups that map the different values used by those applications to identify the same type of object (such as gender codes or country codes).	Creating a Lookup
3	Create the integration. When you do this, you add trigger and invoke connections, and then map the data between the two.	1. Creating an Integration 2. Adding a Source Connection 3. Adding a Target Connection 4. Mapping Data of *Using the Oracle Mapper*
4	Activate the integration.	Activating an Integration
5	Monitor the integration on the dashboard.	Viewing the Dashboard
6	Track payload fields in messages during runtime.	Assigning Business Identifiers and Tracking Business Identifiers in Integrations During Runtime
7	Manage errors at the integration level, connection level, or specific integration instance level.	Managing Errors

Creating Connections

You define connections to the specific cloud applications that you want to integrate. The following topics describe how to define connections:

Topics

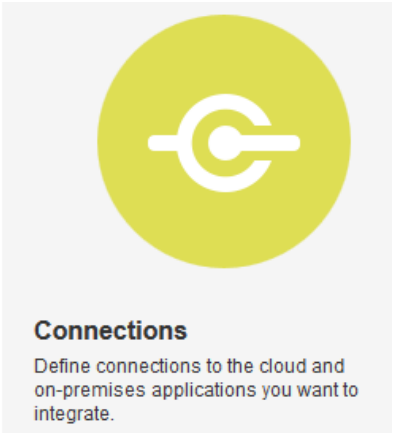
- Creating a Connection
- Adding a Contact Email
- Configuring Connection Properties
- Configuring Connection Security
- Testing the Connection
- Editing a Connection
- Cloning a Connection
- Deleting a Connection
- Refreshing Integration Metadata

Creating a Connection

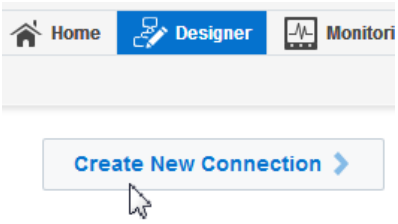
The first step in creating an integration is to create the connections to the applications with which you want to share data.

1. In the Integration Cloud Service toolbar, click **Designer**.

2. On the Designer Portal, click **Connections**.



3. Click **Create New Connection**.

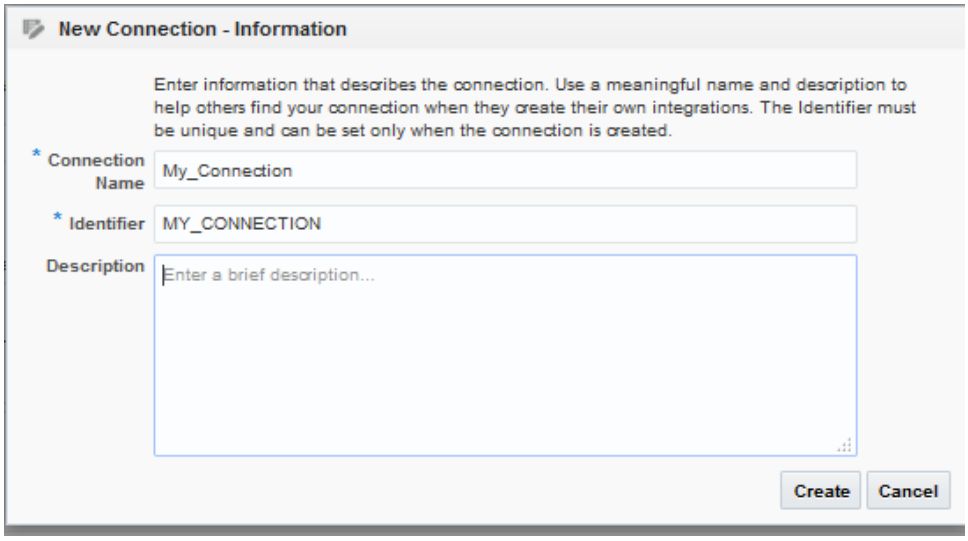


The Create Connection — Select Adapter dialog is displayed.

4. Select an adapter from the dialog. You can also search for the type of adapter to use by entering a partial or full name in the Search field, and clicking **Search**.

The New Connection — Information dialog is displayed.

5. Enter the information to describe the connection. Use a meaningful name to help others find your connection when they begin to create their own integrations.



6. Click **Create**.

   Your connection is created and you are now ready to configure connection details, such as email contact, connection properties, security policies, and connection login credentials.

Adding a Contact Email

From the Connection Administrator section of the connection, you can add a contact email address for notifications.

1. In the **Email Address** field, enter an email address to receive email notifications when problems occur.
2. In the upper right corner, click **Save**.

Configuring Connection Properties

Enter connection information so your application can process requests.

1. Click **Configure Connectivity**.

   The Connection Properties dialog is displayed.

2. See the following sections for information about specifying connection properties.

   - Specifying the Oracle RightNow Cloud WSDL
   - Specifying the Oracle Sales Cloud Adapter Service Catalog Service WSDL or Event Catalog URL
   - Specifying the Oracle Messaging Service Cloud Service Messaging URI
   - Specifying the Oracle Eloqua Cloud Company Name
   - Specifying the Oracle HCM Cloud Service Catalog Service WSDL or Event Catalog URL
   - Specifying the Salesforce Cloud WSDL

- Specifying the Oracle ERP Cloud Service Catalog Service WSDL or Event Catalog URL
- Specifying the SOAP Adapter WSDL
- Specifying the REST Adapter Endpoint
- Specifying the Oracle CPQ Cloud WSDL
- Specifying the NetSuite Adapter WSDL

For the Eventbrite Adapter, see *Using the Eventbrite Adapter*.

For the Evernote Adapter, see *Using the Evernote Adapter*.

For the Facebook Adapter, see *Using the Facebook Adapter*.

For the FTP Adapter, see *Using the FTP Adapter*.

For the Gmail Adapter, see *Using the Gmail Adapter*.

For the Google Calendar Adapter, see *Using the Google Calendar Adapter*.

For the Google Task Adapter, see *Using the Google Task Adapter*.

For the LinkedIn Adapter, see *Using the LinkedIn Adapter*.

For the MailChimp Adapter, see *Using the MailChimp Adapter*.

For the Microsoft Calendar Adapter, see *Using the Microsoft Calendar Adapter*.

For the Microsoft Contact Adapter, see *Using the Microsoft Contact Adapter*.

For the Microsoft Email Adapter, see *Using the Microsoft Email Adapter*.

For the MySQL Adapter, see *Using the MySQL Adapter*.

For the Oracle Commerce Cloud Adapter, see *Using the Oracle Commerce Cloud Adapter*.

For the Oracle Database Adapter, see *Using the Oracle Database Adapter*.

For the Oracle E-Business Suite Adapter, see *Using Oracle E-Business Suite Adapter*.

For the Oracle Siebel Adapter, see *Using the Oracle Siebel Adapter*.

For the SurveyMonkey Adapter, see *Using the SurveyMonkey Adapter*.

For the SAP Adapter, see *Using the SAP Adapter*.

For the Twitter Adapter, see *Using the Twitter Adapter*.

3. Click **OK**.

   You are now ready to configure connection security.

Specifying the Oracle RightNow Cloud WSDL

The following table describes the Oracle RightNow WSDL requirements.

WSDL Requirements	Where Do You Get the WSDL

Only the standard WSDL is supported. The partner WSDL is not supported.	The standard WSDL can include support for both business objects and event subscriptions. This enables you to receive either a business object or an event subscriptions as a request from the Oracle RightNow application. Event subscriptions are supported only if the Oracle RightNow application version is equal to or greater than version 15.5 (May 2015 release). Otherwise, only business objects are visible for selection in the configuration wizard.
	To obtain the standard WSDL:
	1. Collect the following details from your Oracle RightNow Cx Account:
	• Host
	• Interface name
	• User name and password
	2. Open your web browser and enter the following URL to obtain the standard WSDL, replacing the host name and interface details as appropriate.
	`https://host_name/cgi-bin/interface.cfg/services/soap?wsdl`
	For example:
	`https://integration-test.rightnowdemo.com/cgi-bin/integration_ test.cfg/services/soap?wsdl`
	or
	`https://integration-test.rightnowdemo.com/cgi-bin/integration_test.cfg/services/soap?wsdl=typed`
	Permissions for the Public SOAP API must be enabled for the user account to enable use of the Oracle RightNow Connect Web Services for SOAP API. Permissions for the Public SOAP API enable staff members with this profile to access the public SOAP API through account or session authentication.
	The Oracle RightNow adapter does not support a partner WSDL/ generic WSDL. Therefore, the following URL is not supported.
	`https://integration-test.rightnowdemo.com/cgi-bin/integration_test.cfg/services /soap?wsdl=generic`

Specifying the Oracle Sales Cloud Adapter Service Catalog Service WSDL or Event Catalog URL

You must specify a mandatory Oracle Sales Cloud Adapter service catalog service WSDL (for accessing business objects) and optionally an event catalog URL (for accessing event subscriptions) in the Connection Properties dialog.

A mandatory Oracle Sales Cloud Adapter service catalog service WSDL is required to configure the adapter for both inbound and outbound endpoints using either business objects or business services. You can also optionally specify an event catalog URL for accessing and configuring the inbound adapter to use event subscriptions.

Obtaining the Service Catalog Service WSDL

WSDL Requirements	Where Do You Get the WSDL?
The URL must be that of a service catalog service WSDL. The service catalog service is a Fusion Application service that returns a list of external services available for integration. It allows clients to retrieve information about all public Fusion Application service endpoints available for that instance. The service catalog service enables clients to retrieve information about all public Oracle Fusion Application service endpoints available for that instance. The information it returns is specific to the particular cloud instance and also reflects the new services that may have been introduced in patches applied to the instance. This service is used to programmatically discover the SOAP services available on the cloud instance and retrieve the necessary metadata to invoke the SOAP services to manage business objects.	The developer creating an Oracle Sales Cloud connection must work with the Oracle Sales Cloud service administrator to get the concrete WSDL URL for the service catalog service provisioned for the specific SaaS application.

This section describes how to derive the external virtual host and port for a tokenized service catalog service WSDL. The topology information in the Topology Registration setup task contains the external virtual host and port for the domains and applications. The following instructions describe the steps for deriving the values using the service catalog service WSDL URL as an example: `https://atf_server:port/fndAppCoreServices/ServiceCatalogService`.

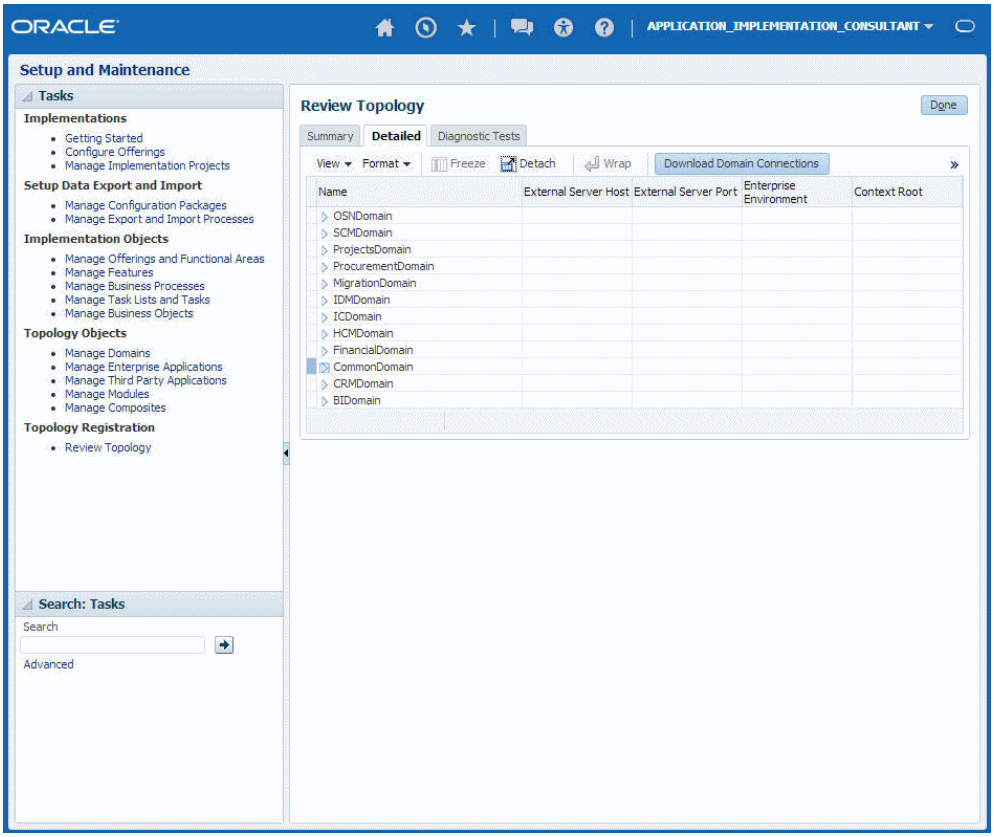
To access the Review Topology page, the ASM_REVIEW_TOPOLOGY_HIERARCHY_PRIV entitlement must be granted to the user's job role. The entitlement is granted to the ASM_APPLICATION_DEPLOYER_DUTY duty role, which is inherited by the duty roles ASM_APPLICATION_DEVELOPER_DUTY and ASM_APPLICATION_ADMIN_DUTY.

If the menu items and tasks described in the following procedure are not available in your cloud instance, your user account is missing the required role. Contact your cloud instance security administrator for assistance.

1. Log in to the cloud instance.
2. Click the **Navigator** icon in the global area in the top part of the window, then chose **Setup and Maintenance** under the **Tools** heading.
3. Select **Review Topology** under the **Topology Registration** section in the **Tasks** regional area on the left side of the window.

4. Click the **Detailed** tab in the middle of the window.

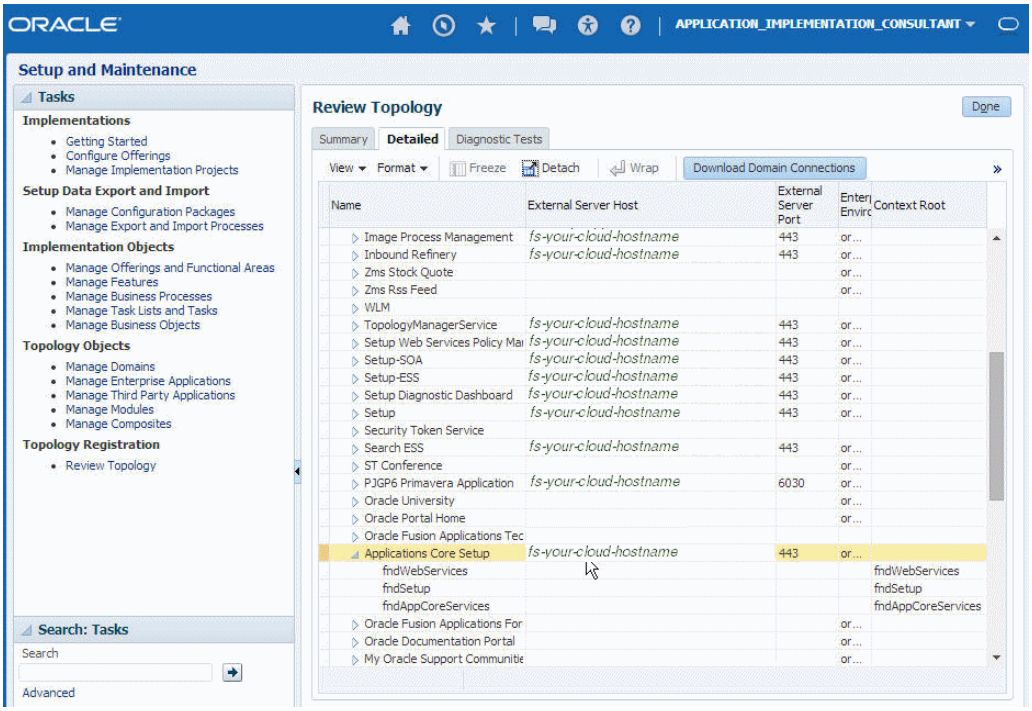
   The tab shows the list of domains configured in the cloud instance.



5. Map the token name for the service path value to the domain name in the Topology Manager:

Token Name in Service Path	Domain Name
atf_server	CommonDomain
crm_server	CRMDomain
fin_server	FinancialDomain
hcm_server	HCMDomain
ic_server	ICDomain
prc_server	ProcurementDomain
prj_server	ProjectsDomain
scm_server	SCMDomain

6. Expand the domain name and select any external virtual host and port for the J2EE applications that are deployed on the domain. In the sample window, the values for this particular instance are **fs-your-cloud-hostname** and **443**, respectively.



7. Replace the *domainName_server*:*PortNumber* with the external virtual host and port identified in the previous step. For example:

   ```
   https://fs-your-cloud-hostname:port/fndAppCoreServices/
   ServiceCatalogService?wsdl
   ```

Obtaining the Event Catalog URL

You must know the customer relationship management (CRM) URL format to access the CRM application user interface. Follow the URL format to determine the event catalog URL. For example, if the CRM URL format is:

```
https://fusxxxx-crm-ext.us.oracle.com/customer/faces/CrmFusionHome
```

Then the event catalog URL is:

```
https://fusxxxx-crm-ext.us.oracle.com/soa-infra
```

Specifying the Oracle Messaging Service Cloud Service Messaging URI

The following table describes the Oracle Messaging Cloud Service messaging URI requirements.

WSDL Requirements	Where Do You Get the WSDL
No restrictions.	Specify the URI of the messaging service to which you subscribed.

Specifying the Oracle Eloqua Cloud Company Name

The following table describes the Oracle Eloqua company name requirements.

Company Name Requirements	Where Do You Get the Company Name
There are no special requirements.	Not applicable.

Specifying the Oracle HCM Cloud Service Catalog Service WSDL or Event Catalog URL

You must specify a mandatory Oracle HCM Cloud service catalog service WSDL (for accessing business objects) and optionally an event catalog URL (for accessing event subscriptions).

Obtaining the Service Cloud Service WSDL

A mandatory Oracle HCM Cloud service catalog service WSDL is required for accessing business objects. You can also optionally specify an event catalog URL for event subscriptions. The service catalog service WSDL is required to configure the adapter for both inbound and outbound endpoints using either business objects or business services. The event catalog URL is required for configuring the inbound adapter for event subscriptions.

WSDL Requirements	Where Do You Get the WSDL
The URL must be that of a service catalog service WSDL. The service catalog service enables clients to retrieve information about all public Oracle Fusion Application service endpoints available for that instance. The information it returns is specific to the particular cloud instance and also reflects the new services that may have been introduced in patches applied to the instance. This service is used to programmatically discover the SOAP services available on the cloud instance and retrieve the necessary metadata to invoke the SOAP services to manage business objects.	The developer creating an Oracle HCM Cloud connection must work with the Oracle HCM Cloud service administrator to get the concrete WSDL URL for the service catalog service provisioned for the specific SaaS application. The concrete WSDL URL must be supplied while creating the connection.

Prerequisites

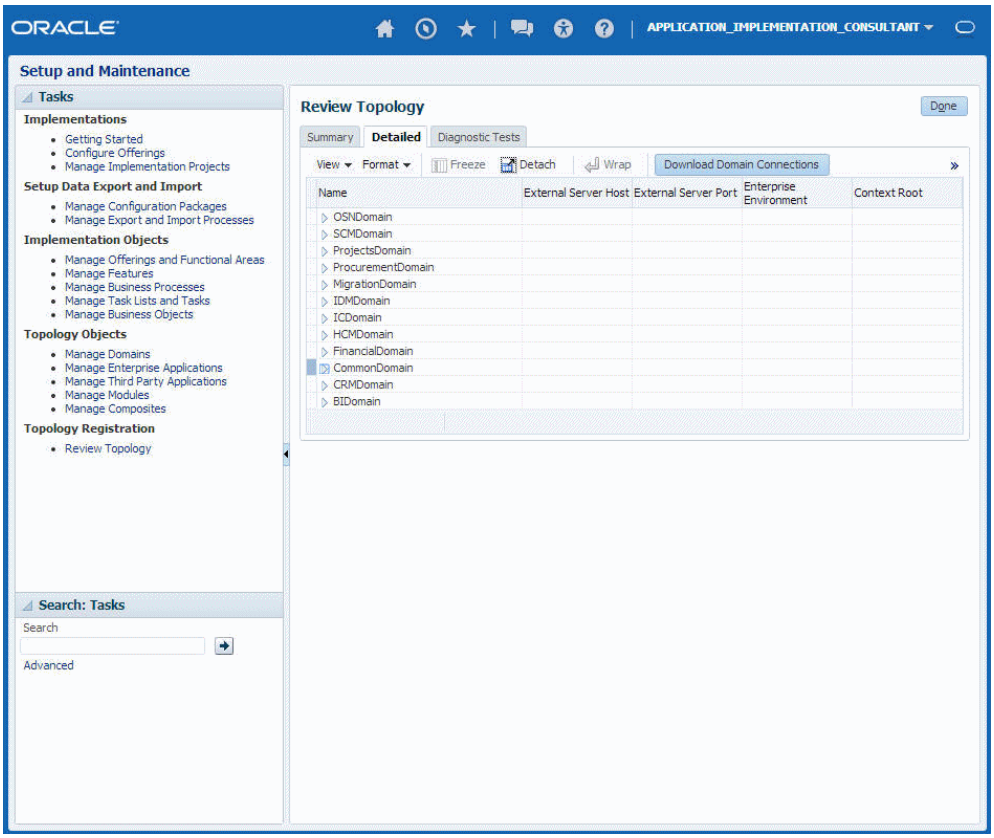
This section describes how to derive the external virtual host and port for a tokenized service WSDL. The topology information in the Topology Registration setup task contains the external virtual host and port for the domains and applications. The following instructions describe the steps for deriving the values using the service catalog service WSDL URL as an example: `https://atf_server:`*`port`*`/fndAppCoreServices/ServiceCatalogService`.

To access the Review Topology page, the ASM_REVIEW_TOPOLOGY_HIERARCHY_PRIV entitlement must be granted to the user's job role. The entitlement is granted to the ASM_APPLICATION_DEPLOYER_DUTY duty role, which is inherited by the duty roles ASM_APPLICATION_DEVELOPER_DUTY and ASM_APPLICATION_ADMIN_DUTY.

If the menu items and tasks described in the following procedure are not available in your cloud instance, your user account is missing the required role. Contact your cloud instance security administrator for assistance.

1. Log in to the cloud instance.
2. Click the **Navigator** icon in the global area in the top part of the window, then chose **Setup and Maintenance** under the **Tools** heading.
3. Select **Review Topology** under the **Topology Registration** section in the **Tasks** regional area on the left side of the window.
4. Click the **Detailed** tab in the middle of the window.

   The tab shows the list of domains configured in the cloud instance.

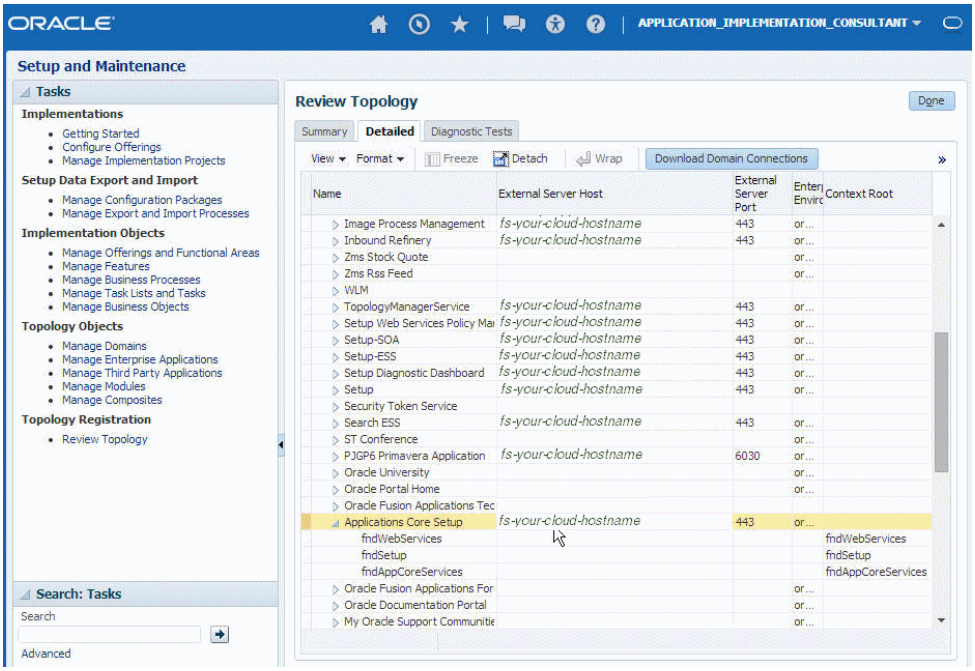


5. Map the token name for the service path value to the domain name in the Topology Manager:

Token Name in Service Path	Domain Name
atf_server	CommonDomain
crm_server	CRMDomain
fin_server	FinancialDomain
hcm_server	HCMDomain
ic_server	ICDomain

Token Name in Service Path	Domain Name
prc_server	ProcurementDomain
prj_server	ProjectsDomain
scm_server	SCMDomain

6. Expand the domain name and select any external virtual host and port for the J2EE applications that are deployed on the domain. In the sample window, the values for this particular instance are **fs-your-cloud-hostname** and **443**, respectively.



7. Replace the *`domainName_server:PortNumber`* with the external virtual host and port identified in the previous step. For example:

   ```
   https://fs-your-cloud-hostname:port/fndAppCoreServices/
   ServiceCatalogService?wsdl
   ```

Obtaining the Event Catalog URL

You must know the CRM URL format to access the CRM application user interface. Follow the URL format to determine the event catalog URL. For example, if the CRM URL format is:

```
https://fusxxxx-crm-ext.us.oracle.com/customer/faces/CrmFusionHome
```

Then the event catalog URL is:

```
https://fusxxxx-crm-ext.us.oracle.com/soa-infra
```

Specifying the Salesforce WSDL

The following table describes the Salesforce WSDL requirements.

WSDL Requirements	Where Do You Get the WSDL
Only enterprise WSDLs are supported.	See the instructions below.

1. Log in to your Enterprise, Unlimited, or Developer Edition Salesforce.com account. Open the Web browser and enter the following URL:

   `www.salesforce.com`

2. Log in to `Salesforce.com` using a valid user name and password.

   You must log in as an administrator or user with the **Modify All Data** permission. Logins are checked to ensure that they are from a known IP address.

3. Under **App Setup**, Expand **Develop** and click **API** to display the WSDL download page.

4. If the organization has managed packages installed in the organization, click **Generate Enterprise WSDL**. Salesforce prompts you to select the version of each installed package to include in the generated WSDL or right-click **Generate Enterprise WSDL** and save it to a local directory.

   In the right-click menu, Internet Explorer users can choose **Save Target As**, while Mozilla Firefox users can choose **Save Link As** to save it to the local directory.

   The Save dialog is displayed.

5. Provide a name for the WSDL file and a location to save the file on your file system, and click **Save**.

Specifying the Oracle ERP Cloud Service Catalog Service WSDL or Event Catalog URL

You specify a mandatory Oracle ERP Cloud service catalog service WSDL (for accessing business objects) and optionally an event catalog URL (for accessing event subscriptions). The service catalog service WSDL is required to configure the adapter for both inbound and outbound endpoints using either business objects or business services. The event catalog URL is required for configuring the inbound adapter for event subscriptions.

Obtaining the Service Cloud Service WSDL

WSDL Requirements	Where Do You Get the WSDL
The URL must be that of a service catalog service WSDL. The service catalog service is a Fusion Application service that returns a list of external services available for integration. It allows clients to retrieve information about all public Fusion Application service endpoints available for that instance. The service catalog service enables clients to retrieve information about all public Oracle Fusion Application service endpoints available for that instance. The information it returns is specific to the particular cloud instance and also reflects the new services that may have been introduced in patches applied to the instance. This service is used to programmatically discover the SOAP services available on the cloud instance and retrieve the necessary metadata to invoke the SOAP services to manage business objects.	The developer creating an Oracle ERP Cloud connection must work with the Oracle ERP Cloud service administrator to get the concrete WSDL URL for the service catalog service provisioned for the specific SaaS application.

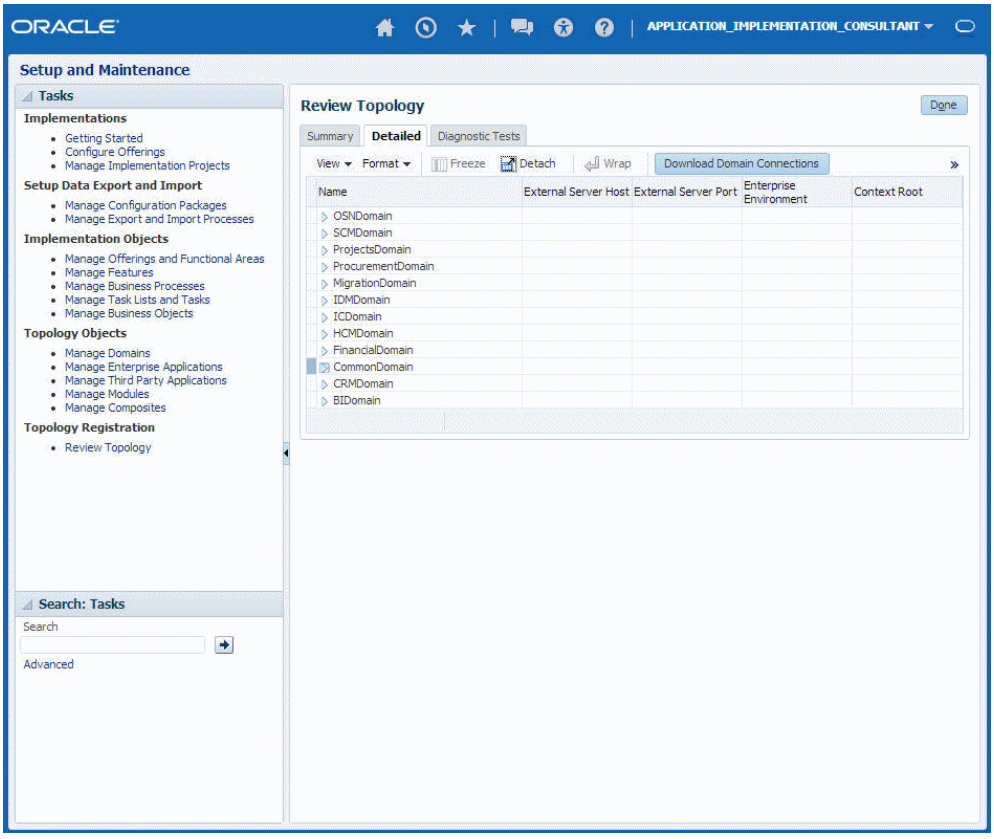
This section describes how to derive the external virtual host and port for a tokenized service WSDL. The topology information in the Topology Registration setup task contains the external virtual host and port for the domains and applications. The following instructions describe the steps for deriving the values using the service catalog service WSDL URL as an example: `https://atf_server:`*`port`*`/fndAppCoreServices/ServiceCatalogService`.

To access the Review Topology page, the ASM_REVIEW_TOPOLOGY_HIERARCHY_PRIV entitlement must be granted to the user's job role. The entitlement is granted to the ASM_APPLICATION_DEPLOYER_DUTY duty role, which is inherited by the duty roles ASM_APPLICATION_DEVELOPER_DUTY and ASM_APPLICATION_ADMIN_DUTY.

If the menu items and tasks described in the following procedure are not available in your cloud instance, your user account is missing the required role. Contact your cloud instance security administrator for assistance.

1. Log in to the cloud instance.
2. Click the **Navigator** icon in the global area in the top part of the window, then chose **Setup and Maintenance** under the **Tools** heading.
3. Select **Review Topology** under the **Topology Registration** section in the **Tasks** regional area on the left side of the window.
4. Click the **Detailed** tab in the middle of the window.

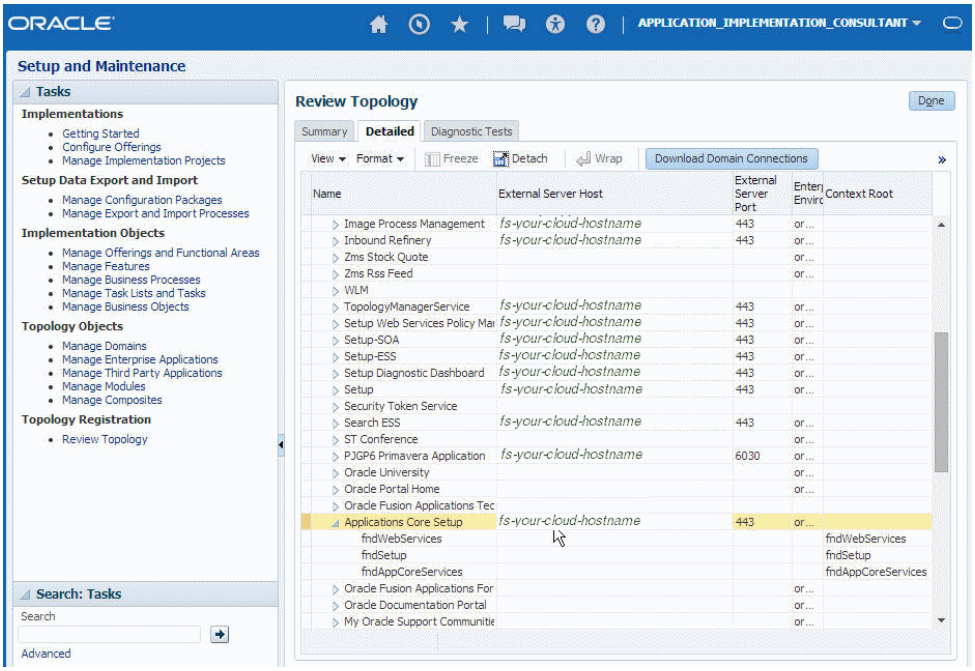
   The tab shows the list of domains configured in the cloud instance.



5. Map the token name for the service path value to the domain name in the Topology Manager:

Token Name in Service Path	Domain Name
atf_server	CommonDomain
crm_server	CRMDomain
fin_server	FinancialDomain
hcm_server	HCMDomain
ic_server	ICDomain
prc_server	ProcurementDomain
prj_server	ProjectsDomain
scm_server	SCMDomain

6. Expand the domain name and select any external virtual host and port for the J2EE applications that are deployed on the domain. In the sample window, the values for this particular instance are **fs-your-cloud-hostname** and **443**, respectively.



7. Replace the *domainName_server:PortNumber* with the external virtual host and port identified in the previous step. For example:

   ```
   https://fs-your-cloud-hostname:port/fndAppCoreServices/
   ServiceCatalogService?wsdl
   ```

Obtaining the Event Catalog URL

You must know the CRM URL format to access the CRM application user interface. Follow the URL format to determine the event catalog URL. For example, if the CRM URL format is:

```
https://fusxxxx-crm-ext.us.oracle.com/customer/faces/CrmFusionHome
```

Then the event catalog URL is:

```
https://fusxxxx-crm-ext.us.oracle.com/soa-infra
```

Specifying the SOAP Adapter WSDL

The following table describes the SOAP Adapter WSDL requirements.

WSDL Requirements	Where Do You Get the WSDL
There is no restriction on the type of WSDL to use.	Any valid WSDL is an acceptable input.

Specifying the REST Adapter Endpoint

The following table describes the REST Adapter endpoint requirements.

Endpoint Requirements	Where Do You Get the Endpoint
Specify an endpoint that adheres to the following format. `https://hostname:443/integration/flowapi/rest/RESTTORESTINTEGRATIONPUT_TEST/v01/`	Not applicable.

Specifying the Oracle CPQ Cloud WSDL

The following table describes the Oracle CPQ Cloud WSDL requirements.

WSDL Requirements	Where Do You Get the WSDL
• The WSDL must be generated by the CPQ Cloud site to integrate with Integration Cloud Service. • Web Services 2.0 must be used to generate the URL needed to generate the WSDL. • The Commerce SOAP server URL endpoint must be used to generate the URL needed to generate the WSDL. • The commerce process to integrate with ICS must be used to generate the URL needed to generate the WSDL.	Integration Cloud Service uses the Oracle CPQ Cloud transaction WSDL to understand the valid data and operations provided by Oracle CPQ Cloud. To access the Oracle CPQ Cloud transaction WSDL: **1.** Log in to the Oracle CPQ Cloud site that you want integrate with Integration Cloud Service. **2.** To open the Admin home page, click **Admin**. The Admin home page appears. **3.** Under **Integration Platform**, click **Web Services**. **4.** For the **Web Service Version**, select **2.0**. **5.** Ensure that the **Commerce** tab is the current tab. **6.** From the **Process Name** list, select the name of the commerce process to integrate with Integration Cloud Service. **7.** In the **SOAP Server URL** field, append `?WSDL` to the end of the value. For example: `https://site_URL/v2_0/receiver/commerce/processVarName/?WSDL`. where: • *`site_URL`* is the base URL of the Oracle CPQ Cloud site. • `processVarName`is the variable name of the selected commerce process. • Optional: To confirm that the URL is correct, open it in a web browser. A page of WSDL should appear. Use the URL you created as needed in Integration Cloud Service to reference the CPQ Cloud transaction WSDL.

Specifying the NetSuite Adapter WSDL

The following table describes the NetSuite Adapter WSDL requirements.

WSDL Requirements	Where Do You Get the WSDL

You must specify the NetSuite WSDL.	Specify the following NetSuite WSDL:
	`https://webservices.netsuite.com/wsdl/`
	`NetSuite_application_version/netsuite.wsdl`
	
	where `NetSuite_application_version` is the version of the NetSuite application. For example:
	`https://webservices.netsuite.com/wsdl/v2014_2_0/netsuite.wsdl`
	
	`https://webservices.netsuite.com/wsdl/v2015_1_0/netsuite.wsdl`
	
	The web services may or may not be hosted at the above location. The adapter can programmatically determine the correct URL for the web services. NetSuite hosts customer accounts in multiple locations. For example:
	• `webservices.netsuite.com`
	• `webservices.na1.netsuite.com`

Configuring Connection Security

Configure security for your connection by selecting the security policy and specifying login credentials.

1. Click **Configure Credentials**.
2. Enter your login credentials.
   a. Select the security policy. The default security policy is displayed first. The security policies available for selection are defined by the adapter you are configuring. For example, the Oracle RightNow Cloud and Oracle Sales Cloud adapters support the username password token policy, the Oracle Eloqua Cloud adapter supports HTTP basic authentication, and the SOAP adapter supports username password token, basic authentication, or not selecting any policy.
   b. If configuring an Eloqua Cloud connection, enter the company property.
   c. Enter a username and password.
   d. Reenter the password a second time.
3. Click **OK**.

   You are now ready to test your connection.

Configuring the Connection Security Policy

The different adapters support a variety of security policies. Some support only a single type, others support multiple types of security policies.

The following table lists the different adapters, the types of security policies they support, and the fields required by those policies.

Adapter Type	Security Policies	Fields
SOAP Adapter	Username Password Token	• Username — The name of a user who has access to the destination web service. • Password • Confirm Password
	Basic Authentication (In the trigger (inbound) direction, HTTP basic authentication over SSL is supported.) If Basic Authentication is required for both trigger and invoke, you must create two separate connections • One for the trigger with the user name and password set to the ICS runtime user name and password. • One for the invoke, with the user name/and password set to the outbound web service.	• Username — The name of a user who has access to the destination web service. • Password • Confirm Password
	No Security Policy	
REST Adapter	Basic Authentication	• Username — The name of a user who has access to the destination web service. • Password • Confirm Password
	OAuth Client Credentials	• Access Token URI — The URL from which to obtain the access token. • Client Id — The client identifier issued to the client during the registration process. • Client Secret — The client secret. • Scope — The scope of the access request. Scopes enable you to specify which type of access you need. Scopes limit access for the OAuth token. They do not grant any additional permission beyond that which the user already possesses. • Auth Request Media Type — The format of the data you want to receive.

Adapter Type	Security Policies	Fields
	OAuth Resource Owner Password Credentials	• Access Token URI — The URL from which to obtain the access token. • Client Id — The client identifier issued to the client during the registration process. • Client Secret — The client secret. • Scope — The scope of the access request. Scopes enable you to specify which type of access you need. Scopes limit access for the OAuth token. They do not grant any additional permission beyond that which the user already possesses. • Auth Request Media Type — The format of the data you want to receive. • Username — The resource owner's user name. • Password — The resource owner's password. • Confirm Password
	OAuth Authorization Code Credentials	• Client Id — The client identifier issued to the client during the registration process. • Client Secret — The client secret. • Authorization Code URI— The URI from which to request the authorization code. • Access Token URI — URI to use for the access token. • Scope — The scope of the access request. Scopes enable you to specify which type of access you need. Scopes limit access for the OAuth token. They do not grant any additional permission beyond that which the user already possesses.

Adapter Type	Security Policies	Fields

	OAuth Custom Three Legged Flow	Complete the fields appropriate to your integration. You must already have created your client application to complete the necessary fields.
		• Authorization Request — The client application URL to which you are redirected when you provide consent. The authorization server sends a callback to Oracle Integration Cloud Service to obtain an access token for storage. When you create your client application, you must register a redirect URI where the client application is listening.
		• Access Token Request — The access token request to use to fetch the access token. Specify the request using CURL syntax. For example:
		`-X POST method -H headers -d string_data access_token_uri? query_parameters`
		• Refresh Token Request — The refresh token request to use to fetch the access token. This request refreshes the access token if it expires. Specify the request using CURL syntax. For example
		`-X POST method -H headers -d string_data access_token_uri? query_parameters`
		• Saccess_token — Use a regular expression (regex) to retrieve the access token.
		• Srefresh_token — Use regex to retrieve the refresh token.
		• Sexpiry — Use regex to identify when the access token expires.
		• Stoken_type — Use regex to identify the access token type.
		• access_token_usage — Specify how to pass the access token to access a protected resource. You can pass the token as a bearer token or as a query parameter. For example:
		`-H Authorization: Bearer ${access_token}`

Adapter Type	Security Policies	Fields
	OAuth Custom Two Legged Flow	Complete the fields appropriate to your integration. You must already have created your client application to complete the necessary fields. • Access Token Request — The access token request to use to fetch the access token. Specify the request using CURL syntax. For example: `-X POST` *`method`* `-H` *`headers`* `-d` *`string_data access_token_uri`*`?` *`query_parameters`* • Refresh Token Request — The refresh token request to use to fetch the access token. This request refreshes the access token if it expires. Specify the request using CURL syntax. For example `-X POST` *`method`* `-H` *`headers`* `-d` *`string_data access_token_uri`*`?` *`query_parameters`* • Sauth_code — Use regex to identify the authorization code. • Saccess_token — Use regex to identify the access token. • Srefresh_token — Use regex to identify the refresh token. • Sexpiry — Use regex to identify when the access token expires. • Stoken_type — Use regex to identify the access token type. • access_token_usage — Specify how to pass the access token to access a protected resource. You can pass the token as a bearer token or as a query parameter. For example: `-H Authorization: Bearer ${access_token}`
	No Security Policy	
Salesforce Adapter	Salesforce Login	• Username — The name of a user who has access to the destination web service. • Password • Confirm Password

Adapter Type	Security Policies	Fields
NetSuite Adapter	NetSuite Authentication	• Email Address — Email address that serves as the user name. • Account • Role — Role-based access control ensures that users can only use data and application functionality that is related to their responsibilities. • Password • Confirm Password
Oracle Eloqua Cloud Adapter	Eloqua HTTP Basic Authentication	• Company — The company name. • Username • Password • Confirm Password
Oracle Sales Cloud Adapter	Username Password Token	• Username • Password • Confirm Password
Oracle HCM Cloud Adapter	Username Password Token	• Username • Password • Confirm Password
Oracle ERP Cloud Adapter	Username Password Token	• Username • Password • Confirm Password
Oracle RightNow Cloud Adapter	Username Password Token	• Username • Password • Confirm Password
Oracle Message Cloud Service	Username Password Token	• Username • Password • Confirm Password
Oracle CPQ Cloud	Username Password Token	• Username • Password • Confirm Password

For the security policies of additional adapters, see the following:

- For the Eventbrite Adapter, see *Using the Eventbrite Adapter*.
- For the Evernote Adapter, see *Using the Evernote Adapter*.
- For the Facebook Adapter, see *Using the Facebook Adapter*.
- For the FTP Adapter, see *Using the FTP Adapter*.
- For the Gmail Adapter, see *Using the Gmail Adapter*.
- For the Google Calendar Adapter, see *Using the Google Calendar Adapter*.

- For the Google Task Adapter, see *Using the Google Task Adapter*.
- For the LinkedIn Adapter, see *Using the LinkedIn Adapter*.
- For the MailChimp Adapter, see *Using the MailChimp Adapter*.
- For the Microsoft Calendar Adapter, see *Using the Microsoft Calendar Adapter*.
- For the Microsoft Contact Adapter, see *Using the Microsoft Contact Adapter*.
- For the Microsoft Email Adapter, see *Using the Microsoft Email Adapter*.
- For the MySQL Adapter, see *Using the MySQL Adapter*.
- For the Oracle Commerce Cloud Adapter, see *Using the Oracle Commerce Cloud Adapter*.
- For the Oracle Database Adapter, see *Using the Oracle Database Adapter*.
- For the Oracle E-Business Suite Adapter, see *Using Oracle E-Business Suite Adapter*.
- For the Oracle Siebel Adapter, see *Using the Oracle Siebel Adapter*.
- For the SurveyMonkey Adapter, see *Using the SurveyMonkey Adapter*.
- For the SAP Adapter, see *Using the SAP Adapter*.
- For the Twitter Adapter, see *Using the Twitter Adapter*.

Testing the Connection

Test your connection to ensure that it is successfully configured.

1. In the upper right corner of the page, click **Test**.

   If successful, the following message is displayed and the progress indicator shows 100%.

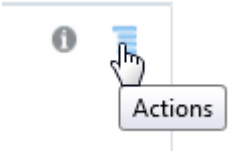
   ```
   The connection test was successful!
   ```

2. If your connection was unsuccessful, an error message is displayed with details. Verify that the configuration details you entered are correct.
3. When complete, click **Save**.

Editing a Connection

You can edit connection settings after creating a new connection.

1. In the Oracle Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. On the Connections page, select **Edit** from the connection **Actions** menu or click the connection name.



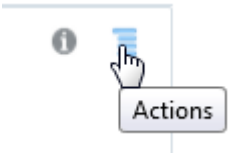
The Connection page is displayed.

4. To edit the notification email contact, change the email address in the **Email Address** field.
5. To edit the connection properties, click **Configure Connectivity**. Note that some connections do not include this button. If your connector does not include a **Configure Connectivity** button, then click the **Configure Credentials** button.

Cloning a Connection

You can clone a copy of an existing connection. It is a quick way to create a new connection.

1. In the Oracle Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. On the Connections page, select **Clone** from the connection **Actions** menu.



The Clone Connection dialog is displayed.

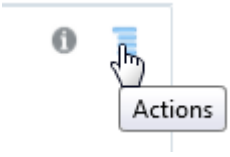
4. Enter the connection information.
5. Click **Clone**.
6. Click **Edit** to configure the credentials of your cloned connection. Cloning a connection does not copy the credentials.

See Editing a Connection for instructions.

Deleting a Connection

You can delete a connection from the connection menu.

1. In the Oracle Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. On the Connections page, click **Delete** from the connection **Actions** menu.



The Delete Connection dialog is displayed if the connection is not used in an integration.

4. Click **Yes** to confirm deletion.

Refreshing Integration Metadata

You can manually refresh the currently-cached metadata available to adapters that have implemented metadata caching. Metadata changes typically relate to customizations of integrations, such as adding custom objects and attributes to integrations. There may also be cases in which integrations have been patched, which results in additional custom objects and attributes being added. This option is similar to clearing the cache in your browser. Without a manual refresh, a staleness check is only performed when you drag a connection into an integration. This is typically sufficient, but in some cases you may know that a refresh is required. For these cases, the **Refresh Metadata** menu option is provided.

To refresh integration metadata:

> **Note:** The **Refresh Metadata** menu option is only available with adapters that have implemented metadata caching.

1. In the Integration Cloud Service toolbar, click **Designer**.

2. In the Designer Portal, click **Connections**.
3. Locate the connection to refresh.
4. From the menu at the right, select **Refresh Metadata**.

A message is displayed indicating that the refresh was successful.

```
Metadata refresh for connection "RightNow" has been initiated successfully.
```

Managing Security Certificates

You can manage security certificates in Oracle Integration Cloud Service.

Topics

- Uploading an SSL Certificate
- Updating or Deleting an SSL Certificate

Uploading an SSL Certificate

Certificates are used to validate outbound SSL connections. If you make an SSL connection in which the root certificate does not exist in Oracle Integration Cloud

Service, an exception is thrown. In that case, you must upload the appropriate certificate. A certificate enables Oracle Integration Cloud Service to connect with external services. If the external endpoint requires a specific certificate, request the certificate and then upload it into Oracle Integration Cloud Service.

To upload a certificate:

1. From the Oracle Integration Cloud Service home page, click the **Administration** tab in the upper right corner.

   All certificates currently uploaded to the trust store are displayed in the Certificates dialog. A navigation panel on the left side of the dialog displays the following details:

   - **All**: Displays all certificates in Oracle Integration Cloud Service.
   - **System**: Displays the certificates automatically included in Oracle Integration Cloud Service. These certificates cannot be deleted.
   - **Uploaded**: Displays the certificates uploaded by individual users. These certificates can be deleted and updated.

2. Click **Upload Certificate** at the top of the page.
3. In the Upload Certificate dialog box, enter a unique identifier for the certificate.

   This is a name you can use to identify the certificate.

4. Click **Browse** to locate the certificate file (`.cer`).
5. Click **Upload**.
6. Click the certificate name to view details such as the subject of the certificate, the issuer of the certificate, the date the certificate was issued, and the date the certificate expires.

Updating or Deleting an SSL Certificate

You can update or delete certificates you uploaded into Oracle Integration Cloud Service. You cannot update or delete system certificates automatically included in Oracle Integration Cloud Service.

To update or delete a certificate:

1. From the Oracle Integration Cloud Service home page, click the **Administration** tab in the upper right corner.
2. Identify the certificate you want to update or delete through either of the following methods:

   a. Scroll through the complete list or filter the display of system-provided or user-uploaded certifications by clicking **System** or **Uploaded** in the navigation tree on the left.

   b. Search by entering a partial or complete certificate name in the **Search** field or filter by selecting an option from the **Filter By** list. From this list, you can filter by **Preinstalled** or **Uploaded**. Search or filter criteria are displayed in the blue banner above the returned list of certificates. To remove search or filter criteria, click the **x** icon in the blue banner or the **x** icon to the right of the **Filter By** list.

3. At the far right of the certificate name, click the **Actions** icon.
4. To update the certificate, click **Update**.
    a. Update the certificate as required, such as updating the certificate name (identifier) and uploading a new certificate. For more information, see Uploading an SSL Certificate.
5. To delete the certificate, click **Delete**.
    a. Click **Yes** when prompted to confirm your selection.

Creating Integrations

Integrations use the connections you created to your applications, and define how information is shared between those applications. You can create new integrations, import integrations, modify or delete integrations, create integrations to publish messages, create integrations to subscribe to messages, and add and remove request and response enrichment triggers. Click one of the following topics for more information.

Topics

Creating an Integration

Creating an integration includes defining the trigger and invoke application connections, and defining how data is mapped between the two applications. The procedure below provides general instructions for creating an integration, with links to more detailed information for certain steps. As you perform each step, the progress indicator changes to let you know how close you are to completing the integration.

If you want to use a lookup table in your data mapping, create the lookup first. See Creating Lookups for instructions.

To create an integration:

1. In the Integration Cloud Service toolbar, click **Designer**.

2. On the Designer Portal, click **Integrations**.
3. On the Integrations page, click **Create New Integration**.

   The Create Integration — Select a Pattern dialog is displayed.
4. Select the type of integration pattern applicable to your business needs. For more information, see Understanding Integration Patterns.

   The New Integration — Information dialog is displayed.
5. Enter the following information:

Field	Description
Integration Name	Provide a meaningful name so that others can understand the integration. You can include English alphabetic characters, numbers, underscores, and dashes in the identifier.
Identifier	Accept the default identifier value. The identifier is the same as the integration name you provided, but in upper case.
Version	Accept the default version number of `01.00.0000`. Or, if you want to change the version number, enter the version using numbers only in this format: `xx.xx.xxxx`. Integrations are uniquely identified by an identifier and version. Note the version format of `xx.yy.zzzz`, where `xx` is the major version and `yy.zzzz` is the minor version. Integrations having the same identifier, but a different major version, can be active at the same time. For example, `INT-A/1.00.0000` and `INT-A/2.00.0000` can be active at the same time. When activating an integration while another integration of the same identifier and same major version is already active, the currently activated integration is deactivated prior to activating the selected integration. For example, if two integrations have the following integration states: • `INT-A/2.00.0000` - Not active • `INT-A/2.10.0000` - Not active Integration `INT-A/2.00.0000` is then activated. • `INT-A/2.00.0000` is now active. • `INT-A/2.10.0000` is not active. Integration `2.10.0000` is then activated. • `INT-A/2.00.0000` is now not active. • `INT-A/2.10.0000` is now active.

Field	Description
Package Name	Enter a new or existing package name in which to place your integration. As you enter the initial letters of an existing package, it is displayed for selection. For more information about packages, see Managing Packages and About Integration Cloud Service Packages.
Description	Provide a meaningful description so that others can understand the integration.

6. Click **Create**.

   The integration designer is displayed with the type of integration pattern you selected in the previous step.

7. Click **Save**.

8. If creating an integration pattern with blank trigger and invoke connections in which to add your own adapters:

   a. Create the trigger connection, as described in Adding a Source Connection.

   b. Create the invoke connection, as described in Adding a Target Connection.

   c. Map data between the two connections, as described in Mapping Data of *Using the Oracle Mapper*.

9. If creating an integration in which to publish to Integration Cloud Service:

   a. Create an integration in which you add a trigger adapter to publish messages to Integration Cloud Service through a predefined Integration Cloud Service Messaging invoke, as described in Creating an Integration to Publish Messages to Integration Cloud Service. No data mapping between the trigger and invoke is permitted.

10. If creating an integration in which to subscribe to Integration Cloud Service:

    a. Create an integration in which you add an invoke adapter to subscribe to messages from Integration Cloud Service through an Integration Cloud Service Messaging trigger, as described in Creating an Integration to Subscribe to Integration Cloud Service.

    b. Map data between the invoke adapter and the Integration Cloud Service Messaging trigger to which to subscribe, as described in Mapping Data of *Using the Oracle Mapper*.

11. When complete, click **Save** and then click **Exit Integration**.

You now see your new integration in the Integrations list ready to be activated. See Activating an Integration for instructions.

Understanding Integration Patterns

You can select from several types of patterns when creating an integration in the Create Integration — Select a Pattern dialog.

Pattern	Description
Map My Data	Create an integration with a blank trigger and invoke in which to add your own adapters.
Publish to ICS	Create an integration in which you add a trigger adapter to publish messages to Integration Cloud Service through a predefined Integration Cloud Service Messaging invoke. No configuration of the invoke subscriber is required. The publisher and subscribers participating in this integration pattern can be activated and deactivated independently of each other. For more information, see Creating an Integration to Publish Messages to Integration Cloud Service.
Subscribe to ICS	Create an integration in which you add an invoke adapter to subscribe to messages from Integration Cloud Service through an Integration Cloud Service Messaging trigger. You are prompted to select the publisher to which to subscribe. You must have already created a publisher to which to subscribe. The publisher does not need to be active, but must already be completely configured. Any business identifiers defined on fields in the published integration are copied to the subscriber. Any changes made to the published integration's business identifiers *after* copying are not reflected in the subscriber. The publisher and subscribers participating in this integration pattern can be activated and deactivated independently of each other. For more information, see Creating an Integration to Subscribe to Integration Cloud Service.

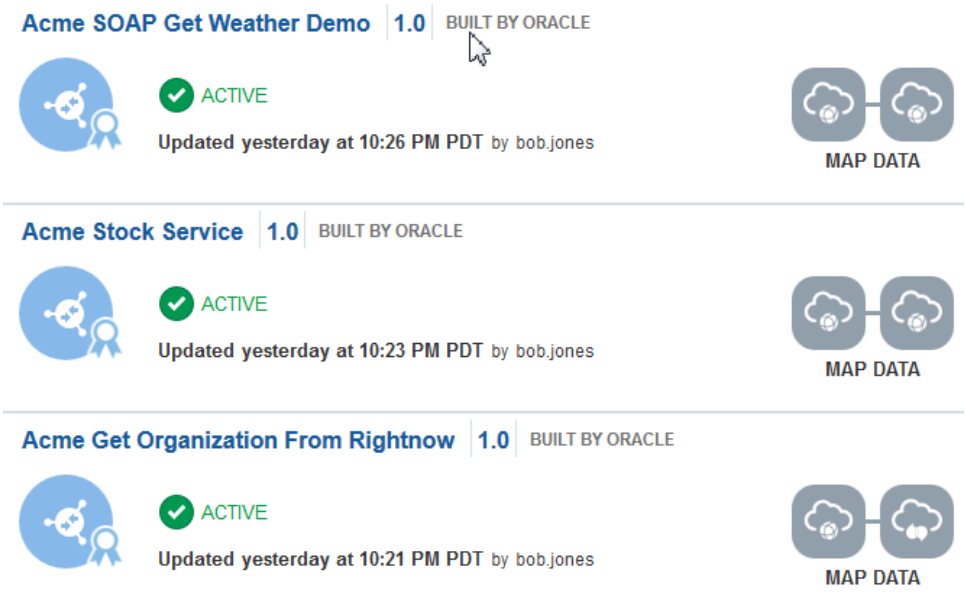
For more information, see Integration Cloud Service Messaging.

Importing a Prebuilt Integration

You can import prebuilt integrations into your Integration Cloud Service environment.

There are two types of prebuilt integrations:

- User-created integrations. These are integrations that you or another user created.
- Oracle-created integrations from the Oracle Marketplace. You import integrations from the Oracle Marketplace as part of a package. These integrations are designated with a **BUILT BY ORACLE** message that is displayed next to the integration name on the Integrations page. You cannot edit these integrations, but you can view their contents, including mappings and business identifiers. You must edit the connections in these integrations to include endpoint credentials relevant to your business requirements. You can also clone these integrations, which enables you to edit the cloned version of the integration.

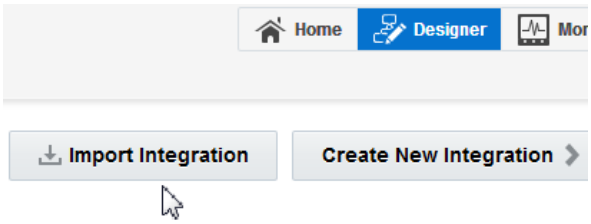


Importing a User-Created Integration

To import a user-created integration:

1. In the Integration Cloud Service toolbar, click **Designer**.

2. On the Designer Portal, click **Integrations**.
3. Above the Integrations list, click **Import Integration**.



4. Click **Browse** to select the file to import. If you are importing a single integration, select the JAR file to import. If you are importing a package of integrations, select the PAR file to import.

Importing a Prebuilt Integration from Oracle Marketplace

To import a prebuilt integration from Oracle Marketplace.

1. In the upper right corner of the page, click **Oracle Marketplace**.
2. The Oracle Marketplace is displayed.

3. Click **Applications**.
4. Browse through the list of applications and select the prebuilt integration package to import.
5. When prompted, select the server to which to upload the prebuilt integration file.

   The prebuilt integration is imported as a package file that is visible on the Packages page in Integration Cloud Service. If you go to the Integrations page, the individual integrations of that imported package file are designated with a **BUILT BY ORACLE** message to the right of the integration name.

You can customize the mappings in the prebuilt integrations imported from Oracle Marketplace. See Adding Customized Mappings to Prebuilt Integrations.

Adding a Trigger (Source) Connection

The trigger (source) connection sends requests to Integration Cloud Service. The information required to connect to the application is already defined in the connection. However, you still must specify certain information, such as the business object and operation to use for the request and how to process the incoming data.

To add a trigger connection:

1. In the Integration Designer, drag a connection from the Connections or Technologies panel on the right to the **Source** area on the canvas.
2. On the Basic Info page, enter a name and optional identifier for this connection. See Basic Info Tab Properties for instructions.
3. Click **Next**.
4. On the next pages of the wizard, enter information for the connection you chose. See the topic below for your connection type to get started with configuring your trigger (source) connection:
   - Configuring Oracle RightNow Cloud Source Request Properties
   - Oracle Sales Cloud Source Request Properties
   - Configuring Oracle Messaging Cloud Service Source Operations Properties
   - Configuring Oracle HCM Cloud Source Request Configuration Properties
   - Configuring Salesforce Trigger Outbound Messaging Properties
   - Configuring Oracle ERP Cloud Trigger Request Properties
   - Configuring Oracle CPQ Trigger Request Properties
   - Configuring SOAP Adapter Trigger Operation Properties
   - Configuring REST Adapter Request Parameters Properties
5. After you configure the connection, the Summary page appears.
6. Review your changes, and click **Done**.

   The connection information appears on the canvas, along with arrows depicting the configured operations.

7. Click **Save**.

To add the Oracle Commerce Cloud Adapter, see *Using the Oracle Commerce Cloud Adapter*.

To add the Oracle E-Business Suite Adapter, see *Using the Oracle E-Business Suite Adapter*.

To add the Oracle Database Adapter, see *Using the Oracle Database Adapter*.

To add the Oracle Siebel Adapter, see *Using the Oracle Siebel Adapter*.

To add the SAP Adapter, see *Using the SAP Adapter*.

Adding an Invoke (Target) Connection

Integration Cloud Service sends requests or information to the invoke (target) connection. The information required to connect to the application is already defined in the connection. However, you still must specify certain information, such as the business object and operation to use for the request and how to process the data.

To add an invoke (target) connection:

1. In the Integration Designer, drag a connection from the Connections or Technologies panel on the right to the **Target** area on the canvas.
2. On the Basic Info page, enter a name and optional identifier for this connection. See Basic Info Tab Properties for instructions.
3. Click **Next** for instructions.
4. On the next pages, enter information for the connection you chose. See the topic below for your connection type:
   - Configuring Oracle RightNow Cloud Target Operation Properties
   - Configuring Oracle Sales Cloud Target Operation Properties
   - Configuring Oracle Messaging Cloud Service Target Operations Properties
   - Configuring Oracle Eloqua Cloud Target Operation Properties
   - Configuring Oracle HCM Cloud Target Operation Properties
   - Configuring Salesforce Target Operation Properties
   - Configuring Oracle ERP Cloud Invoke Operation Properties
   - Configuring Oracle CPQ Invoke Operation Properties
   - Configuring SOAP Adapter Invoke Operation Properties
   - Configuring REST Adapter Request Parameters Properties
   - Configuring NetSuite Adapter Invoke Operation Properties

   To add the Eventbrite Adapter, see *Using the Eventbrite Adapter*.

   To add the Evernote Adapter, see *Using the Evernote Adapter*.

   To add the Facebook Adapter, see *Using the Facebook Adapter*.

   To add the FTP Adapter, see *Using the FTP Adapter*.

To add the Gmail Adapter, see *Using the Gmail Adapter*.

To add the Google Calendar Adapter, see *Using the Google Calendar Adapter*.

To add the Google Task Adapter, see *Using the Google Task Adapter*.

To add the LinkedIn Adapter, see *Using the LinkedIn Adapter*.

To add the MailChimp Adapter, see *Using the MailChimp Adapter*.

To add the Microsoft Calendar Adapter, see *Using the Microsoft Calendar Adapter*.

To add the Microsoft Contact Adapter, see *Using the Microsoft Contact Adapter*.

To add the Microsoft Email Adapter, see *Using the Microsoft Email Adapter*.

To add the MySQL Adapter, see *Using the MySQL Adapter*.

To add the Oracle Commerce Cloud Adapter, see *Using the Oracle Commerce Cloud Adapter*.

To add the Oracle Database Adapter, see *Using the Oracle Database Adapter*.

To add the Oracle E-Business Suite Adapter, see *Using the Oracle E-Business Suite Adapter*.

To add the Oracle Siebel Adapter, see *Using the Oracle Siebel Adapter*.

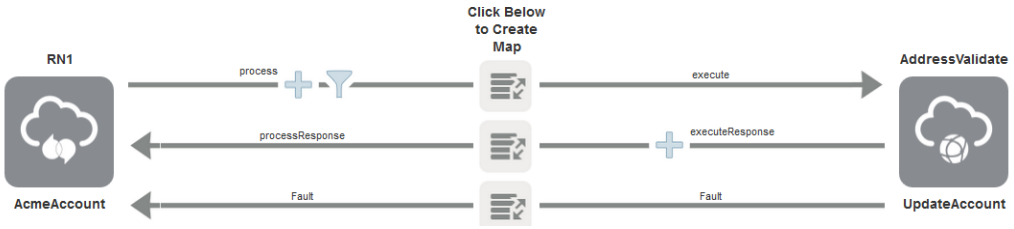
To add the SAP Adapter, see *Using the SAP Adapter*.

To add the SurveyMonkey Adapter, see *Using the SurveyMonkey Adapter*.

To add the Twitter Adapter, see *Using the Twitter Adapter*.

5. After you configure the connection, the Summary page appears.
6. Click **Done**, then click **Save**.

The connection information appears on the canvas, along with arrows depicting the configured operations.



Creating an Integration to Publish Messages to Integration Cloud Service

You can create integrations that enable you to publish messages to Integration Cloud Service. Message publishing is accomplished through use of Integration Cloud Service Messaging.

To create an integration to publish messages to Integration Cloud Service:

Note: Integration Cloud Service Messaging does not support messages larger than 512 KB. Messages larger than 512 KB result in instance failure.

1. Select **Publish To ICS** in the Create Integration — Select a Pattern dialog, as described in Creating an Integration.
2. Complete the fields of the New Integration — Information dialog, as described in Creating an Integration.

   This creates an integration pattern with a predefined Integration Cloud Service Messaging invoke that enables you to publish messages to Integration Cloud Service.
3. In the integration designer, drag an adapter from the Connections panel on the right to the trigger (source) area on the canvas. For this example, an Oracle Sales Cloud Adapter is selected.

   The wizard for the Oracle Sales Cloud Adapter is displayed.
4. On the Basic Info page, enter an endpoint name and optional identifier for this connection. See Configuring Basic Information Properties for instructions.
5. Click **Next**.
6. On the Request page, select a business object (for this example, **Account** is selected), then click **Next**.
7. On the Response page, select **None** as the response type, then click **Next**.
8. On the Summary page, click **Done**.

   The Oracle Sales Cloud Adapter is configured to publish messages to Integration Cloud Service through use of Integration Cloud Service Messaging. Note that there is no request mapper available with this type of integration pattern.

9. Click **Save**, then click **Exit Integration**.

   To subscribe to the message configured in this section, you must now configure Integration Cloud Service to act as a publisher. This enables Integration Cloud Service to publish the messages to which other adapters can then subscribe. For instructions, see Creating an Integration to Subscribe to Integration Cloud Service.

Creating an Integration to Subscribe to Integration Cloud Service

You can create integrations that enable you to subscribe to messages from Integration Cloud Service. Message subscription is accomplished through use of Integration Cloud Service Messaging.

To create an integration to subscribe to Integration Cloud Service:

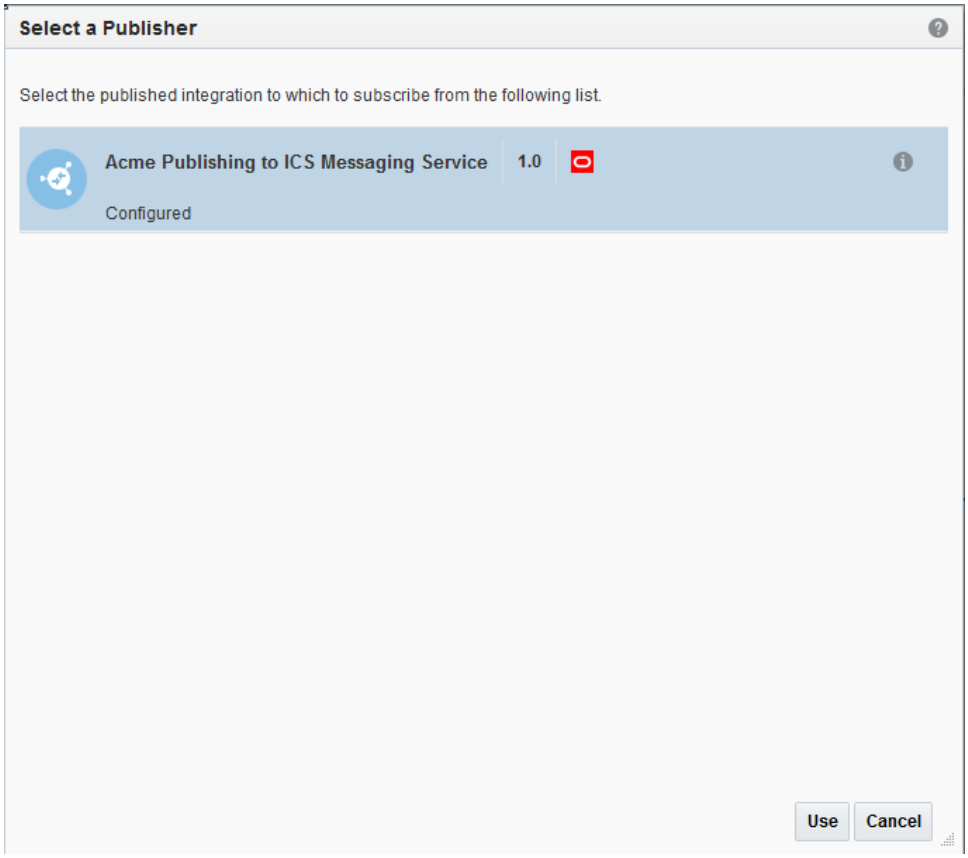
1. Complete the steps in section Creating an Integration to Publish Messages to Integration Cloud Service to first configure Integration Cloud Service as a subscriber to messages from an adapter.

2. Select **Subscribe To ICS** in the Create Integration — Select a Pattern dialog, as described in Creating an Integration.

3. Complete the fields of the New Integration — Information dialog, as described in Creating an Integration. This creates an integration pattern with Integration Cloud Service Messaging that enables you to subscribe to messages from Integration Cloud Service.

   The Select a Publisher dialog is displayed.

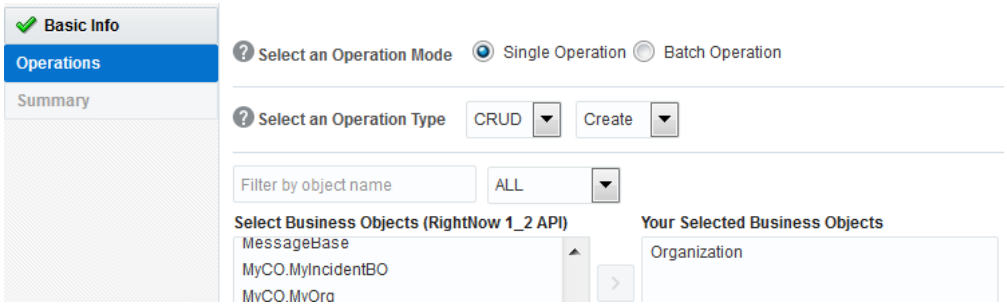
4. Select the integration to which to subscribe, then click **Use**. For an integration to be displayed for selection, you must first configure Integration Cloud Service as a subscriber, as described in Creating an Integration to Publish Messages to Integration Cloud Service. Only integrations that are 100% completed and unlocked are displayed. Integrations that are locked (meaning that they are being edited) are not displayed.



5. Drag an adapter to the invoke (target) area of the integration designer. For this example, an Oracle RightNow Cloud Adapter is added.

6. On the Basic Info page, enter a name and optional identifier for this connection. See Basic Info Tab Properties for instructions.

7. Click **Next**.
8. On the Operations page, select an appropriate operation and business object, then click **Next**. For this example, a CRUD **Create** operation and **Organization** business object are selected.



9. On the Summary page, review your changes, then click **Done**.

   The request mapper is available with this type of integration pattern.

10. Click the **Request Mapping** icon, then click **Create**.
11. Map source fields to the corresponding target fields. For information, see Mapping Data of *Using the Oracle Mapper*.
12. When complete, click **Save**, then click **Exit Mapper**.

    The Oracle RightNow Cloud Adapter is configured to subscribe to messages from Integration Cloud Service through use of Integration Cloud Service Messaging.

13. Click **Save**, then click **Exit Integration**.
14. Activate the publishing integration described in Creating an Integration to Publish Messages to Integration Cloud Service and the subscribing integration described in this section. For instructions, see Activating an Integration.

    The completed publishing and subscription integrations enable you to:

    - Create an object in one application that causes the object to be created in other applications.
    - Enable multiple applications to subscribe to Integration Cloud Service and be registered for updates.
    - Enable additional subscribers to be added or removed without impacting other subscribers or publishers.

Business identifier tracking data is copied when a subscriber is created. If a publishing integration is updated later, you must update the subscribing integration.

For example, assume you create a publishing integration, then create a subscribing integration and select to subscribe to the publishing integration. Select the **Configuration** icon, and note that the tracking attributes of the selected publishing integration are displayed. Assume you then edit the publishing integration and change the operation of the trigger adapter (as an example), save, and exit the canvas. If you then edit the subscribing integration and click the **Configuration** icon, note that the business identifier tracking attributes of the publishing integration that are displayed are those that existed *before* the updates were made. The tracking fields are not updated as per the updated publisher integration. This is the expected behavior.

Adding Request and Response Enrichments

When you create an integration, you also have the option of adding both request and response message enrichment points to the overall integration flow. Enrichments participate in the overall integration flow and can be used in the request and/or response payloads between the trigger and invoke.

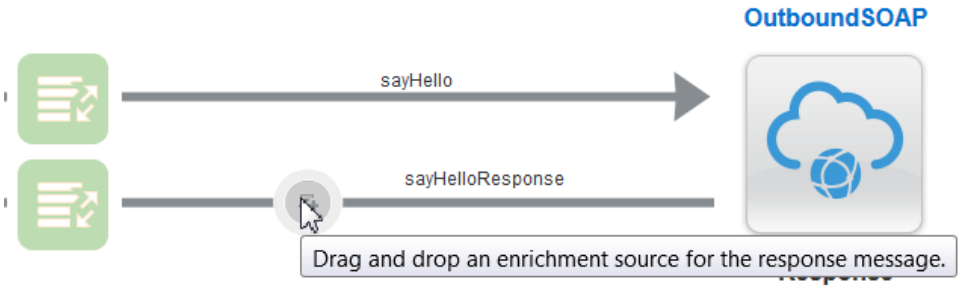
To add request and response enrichments:

1. Design an integration with trigger and invoke connections and request and response mappings. For this example, the integration looks as follows when complete. Note the two enrichment point circles in the design; one appears on the inbound (request) side and the other appears on the outbound (response) side.

The request and response mappings for this example are as follows:

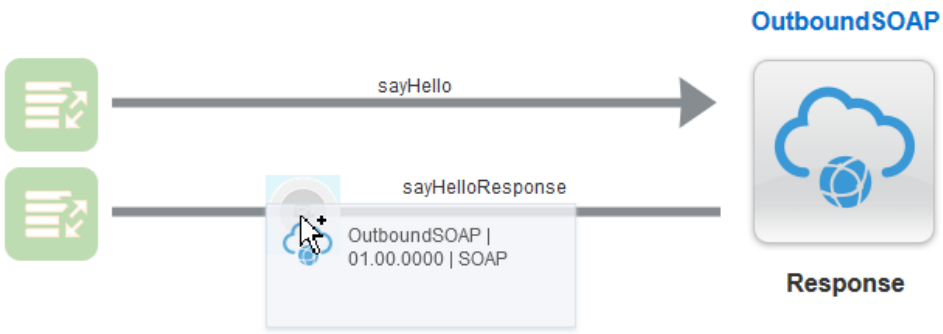
Mapping	Source	Target
Request	HelloRequest/FirstName	sayHello/name
Response	sayHelloResponse/ sayHelloReturn	HelloResponse/Greeting

You are now ready to add enrichments to the integration. For this example, a response message enrichment is added to the **Drag and drop an enrichment source for the response message** area. You can also add request message enrichments on the request (inbound) side.



2. From the Connections panel on the right, drag an adapter to the enrichment area on the response message shown below.

   For this example, a SOAP Adapter is dragged to the **Drag and drop an enrichment source for the response message** area. This action invokes the wizard for configuring the SOAP Adapter.



3. Complete the pages of the wizard to configure the SOAP Adapter, then click **Done**. For this configuration, a different operation for selecting timestamp details is chosen.

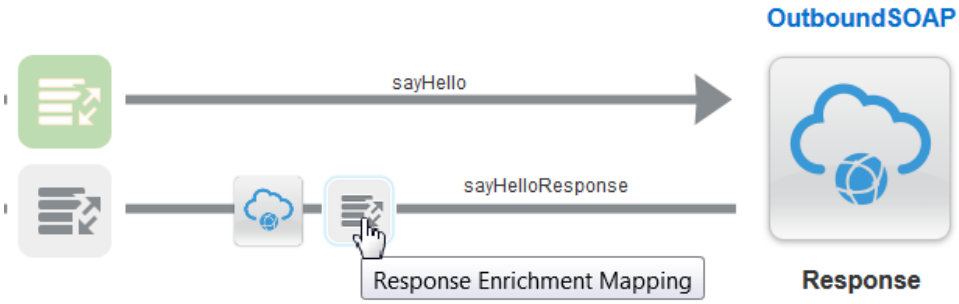
   You are prompted with a dialog to delete any impacted response mappings that you previously configured for the response mapper. The response mapper requires updates because of the enrichment response adapter configuration you just performed.

4. Click **Yes**. You recreate the response mappings later in these steps.

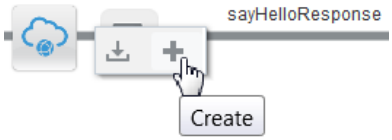
5. Click **Save**.

   A SOAP Adapter icon and response enrichment mapper are added to the response side of the integration. Note that because you deleted the response mappings in the previous step, that icon is no longer shaded in green. This indicates that the response mapper requires configuration.

6. Click the **Response Enrichment Mapping** icon between the trigger and invoke.



7. Click the **Create** icon that is displayed. This invokes the mapper.



8. Map source elements to target elements to include a timestamp with the response, then click **Save** when complete.

   The response enrichment mappings are as follows:

Mapping	Source	Target
Response Enrichment	sayHelloResponse/ sayHelloReturn	visitTimestampReq > reqMsg

   The **Response Mapping** icon is displayed in green, indicating that it has been configured.

9. Click the **Response Mapping** icon to invoke the mapper again. This mapper requires updates because of the enrichment response mapping you performed.

10. Remap the source elements to target elements in the response mapper.

    The response mappings are updated. Note that a different source is now mapped to the original target of HelloResponse/Greeting.

Mapping	Source	Target
Response	$ResponseEnrichmentApplicationObject > visitTimestampResp > respMsg	HelloResponse/Greeting

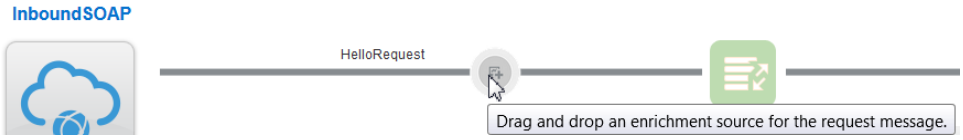
    The **Response Enrichment Mapping** icon is displayed in green, indicating that it has been reconfigured.

11. Click **Save**, then click **Exit Mapper** when complete.

    The integration with response enrichments added to the invoke (target) area looks as follows:

12. Click **Save**, then click **Exit Integration** when complete.

    You are ready to activate the integration. While not demonstrated in this example, you can also configure the enrichment area on the request message shown below by dragging and dropping an adapter to the **Drag and drop an enrichment source for the request message** area. This invokes the adapter configuration wizard.



You can also update existing enrichments at a later time, such as the objects selected in the adapter configuration wizard and the enrichment mappings.

For more information about enrichments, see About Integration Cloud Service Enrichments.

Deleting Request and Response Enrichments

You can delete the request and response message enrichment point mappings added to an integration. After deleting the enrichment point mappings, the integration is returned to its original pre-enrichment state.

To delete request and response enrichments:

1. On the Integration page, select the integration. The integration must not be active.
2. Click the enrichment area on the request message or response message to delete.
3. Select the **Delete** icon that is displayed.

   This deletes the mappings.
4. Click **Yes** when prompted to confirm.

   Click **Save**, then click **Exit Canvas**.

For more information about enrichments, see About Integration Cloud Service Enrichments.

Creating Routing Paths for Two Different Invoke Endpoints in Integrations

You can create an integration in which you define routing paths for two different invoke endpoints. During runtime, the expression filtering logic for the routing paths is evaluated and, based on the results, the path to one of the invoke endpoints is taken. If the filtering logic for neither routing path is satisfied, then neither invoke endpoint is contacted.

The expression logic works as follows:

- You define an expression filter on the first (upper) invoke endpoint.
- You define either an ELSE condition or an expression filter on the second (lower) invoke endpoint.

During runtime, if the expression filtering logic for the first (upper) invoke endpoint evaluates to true, then the path to that invoke endpoint is taken. If the expression evaluates to false, then that invoke endpoint is skipped, and the path to the second (lower) invoke endpoint is taken through either an ELSE condition or an expression filter.

In addition to creating routing paths, you also define request and response (and optionally, enrichment) mappings on both invoke endpoints.

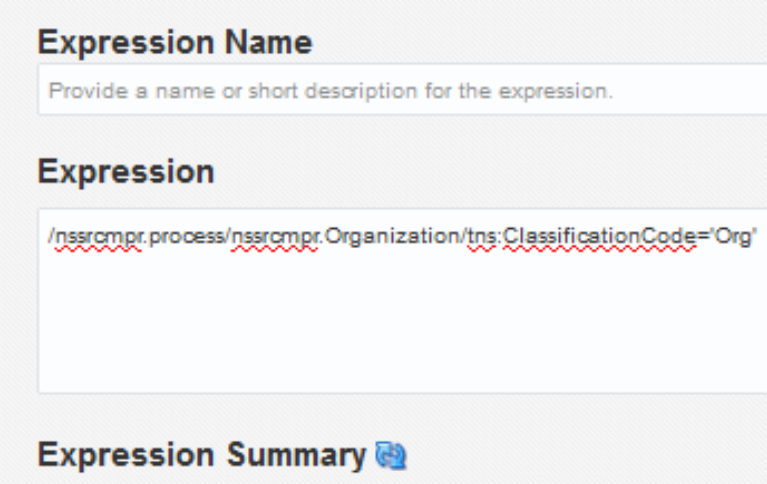
To create routing paths for two different invoke endpoints in integrations:

1. On the Integrations page, select the integration in which to define a routing filter. Ensure that the integration is fully defined with trigger and invoke connections, business identifier tracking, and mappings.
2. Click the **Filter** icon on the trigger side of the integration to create a filtering expression. Routing is created after any defined request enrichment and before the initial request mapping.

3. Click the **Routing** icon in the menu that is displayed.

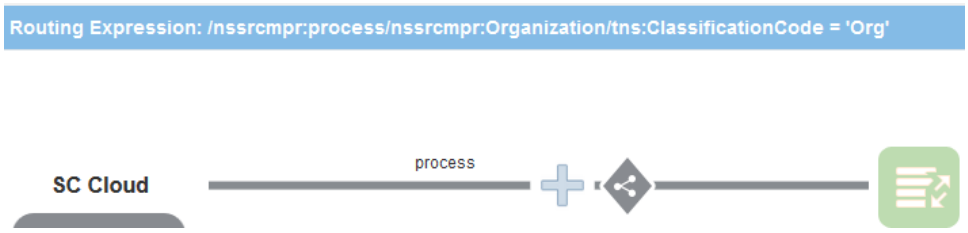
   The Expression Builder is displayed for building routing expressions. The Expression Builder supports multiple source structures. You can create OR expressions using both source structures. You can also name expressions and calculate expression summaries with the **Expression Summary** icon. Elements and attributes with and without namespace prefixes are also supported.
4. Drag an element from the **Source** area to the **Expression** field.
5. Define a value.

   For this example, the **ClassificationCode** element is defined as equal to `Org`. This means that `Org` is retrieved when this expression evaluates to true.
6. If you want to calculate the expression, click the **Expression Summary** icon. This shows the summary of the expression and defines a more user-friendly, readable version of the expression you just created.
7. If that name is not sufficiently user-friendly, copy and paste the expression to the **Expression Name** field for additional editing.

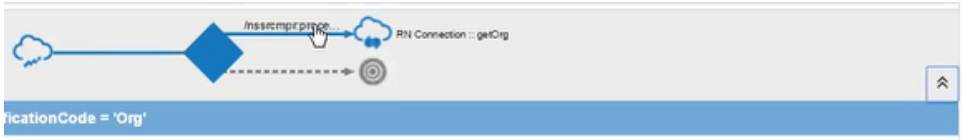


8. Click **Save**, then click **Exit Expression Builder**.

   The defined expression is displayed above the integration. The **Filter** icon has now changed to indicate that an expression is defined.



9. On the right side of the integration, click the **Routing Drawer** icon to display a graphical routing diagram with two potential paths. The first route that you just defined (the upper trigger and invoke) shows the defined expression above the line. The second route (the lower trigger and invoke) is displayed as a dotted line because it is not yet defined.



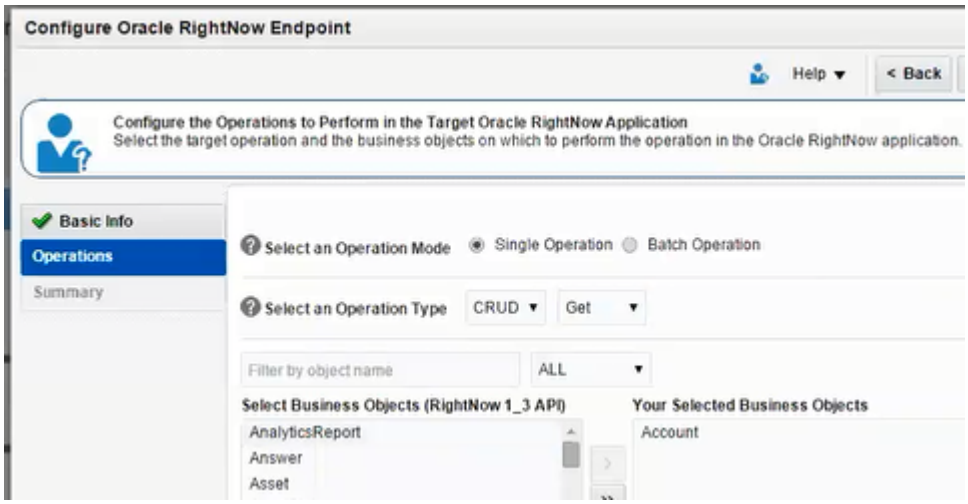
   You can activate the integration now if additional filtering is not required or define an additional routing filter. For this example, a second route is defined.

10. Click the **bull's eye** icon in the lower trigger icon to define routing on the second trigger and invoke route.

This refreshes the integration to display the lower trigger and invoke route in the integration. The trigger side remains as defined for the first route, but the invoke route is undefined.

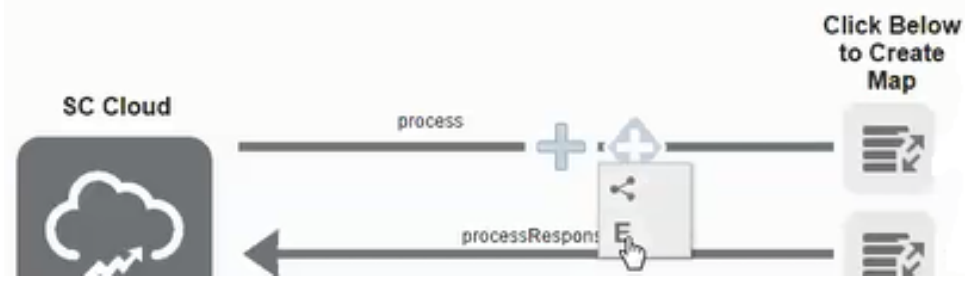
11. Click **Show Palette** to display the list of available connections and technologies.
12. Drag an adapter to the invoke (target) area of the integration (for this example, an Oracle RightNow adapter is added).

    The Adapter Configuration Wizard is invoked.
13. Configure the pages of the wizard for the Oracle RightNow adapter. For this example, the **Get** operation and **Account** business object are selected on the Operations page.



The integration is now defined for the second invoke. You now need to create a filtering expression for the second invoke.

14. Click the **Filter** icon to create a filtering expression.
15. If no additional expression is required, click the **E** icon (to create an ELSE condition).



This defines an ELSE condition for the second trigger and invoke. The ELSE condition is taken if the first route evaluates to false (that is **ClassificationCode** does not equal **Org**). You can toggle back and forth between the two trigger routes

by clicking the adapter icon on the individual line. The line in blue is the currently visible invoke in the integration.

16. If you want to define your own expression filter for the second route instead of using the ELSE condition, perform the following steps:

    a. Click the **Filter** icon.

    b. Select **Clear Expression** to remove the ELSE condition.

    c. Click **Yes** when prompted to confirm.

    d. Click the **Filter** icon again and select the **Edit** icon to invoke the Expression Builder as you did in Step 3.

    e. Define an expression.

    f. Click **Save**, then click **Exit Expression Builder**.

    Request and response mappings must now be defined.

17. Click the **Request Mapper** icon to define the mapping.

    For this example, the following mapping is defined.

Source	Target
process > Organization > Organizationid	Get > Account > ID > id

18. Click the **Response Mapper** icon to define the mapping.

    For this example, the following mapping is defined.

Source	Target
process > GetResponse > Account > ID > LookupName	processResponse > Organization > Name

    Integration design in now 100% complete.

19. Activate the integration.

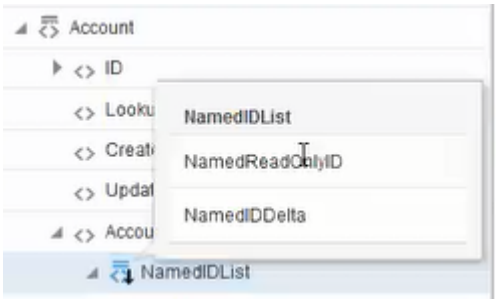
Creating Routing Expression Logic in Both Expression Mode and Condition Mode

You can create XPath expressions for routing conditions in two different user interface modes:

- Expression mode: This mode provides an interface for creating and viewing the entire XPath expression.
- Condition mode: This mode provides an easier-to-read interface to create and view XPath condition expressions. This mode is useful for business analysts who may be less experienced with XPath expressions.

You can toggle between expression mode and condition mode when creating and viewing your expressions. Elements and attributes for which mapping is required are identified by a blue asterisk (*) to the left of their names. You can also place your cursor over elements and attributes to display specific schema details such as the data type, if mapping is required, and so on. When creating an expression, note the following functionality in the tree:

- Three levels of elements are loaded by default in the tree in the **Source** area. When you reach the third level, a **Load more** link is displayed. Click this link to display all the direct children of that element. Only base types are loaded automatically. To load the extended types of the base type, click the base type, which is identified by a unique icon. This invokes a menu of extended types that you can select to load one by one into the tree.



- Elements in the tree in the **Source** area that you have already dragged to an expression are identified by green checkboxes. These elements are displayed even if they are deeper than three levels in the tree.
- You can search for an element that is not yet loaded in the tree by entering the name in the **Find** field and clicking the **Search** icon. This action loads that specific element into the tree.

This section provides an example of building an expression using both modes.

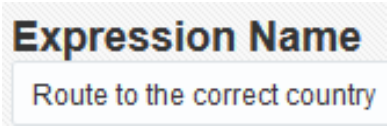
To create routing expressions in both expression mode and condition mode:

1. Click the **Filter** icon on the source side of an integration to create a filtering expression.

2. Click the **Routing** icon in the menu that is displayed.

   The Expression Builder is displayed for building routing expressions. Expression mode is the default mode.

3. In the field immediately below **Expression Name**, optionally enter a short description about the expression you want to build.



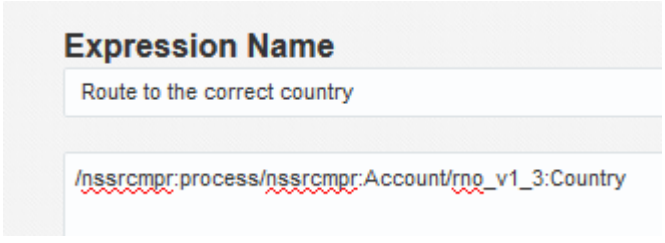
4. Add an element from the **Source** area on the left side to the expression field immediately below the short description field. If needed, you can also add functions from the **Components** section.

   There are two ways to add an element to the expression field:

   a. Drag the element from the **Source** area.

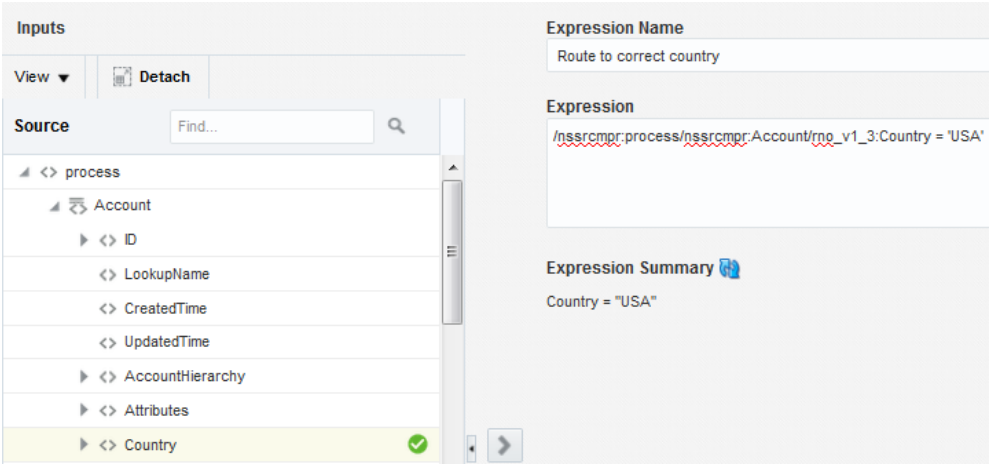
   b. Select the row of the element in the **Source** area, then click the **Move** icon in the middle of the page to move the element.

   The expression for the selected element is displayed in the expression field (for this example, the expression for the **Country** element was added). The selected element is identified by green checkbox in the **Source** area.



5. To the right of the added expression, define an operator and a value within single or double quotes (for this example, = "USA" is defined).

6. Click the **Expression Summary** icon to view a simplified, user-friendly version of the expression.. Easy-to-read output is displayed.



Note:

- To add additional elements to the expression, you can place your cursor in the exact location of the expression, select the row of an element in the **Source** area, and click the **Move** icon. These actions add that element to the exact location of your cursor.
- You can drag an element to the exact location of your cursor in the expression, and the expression of the element is added to the cursor location, and not the location in which you drop the element.
- You can drag an element on top of an existing expression element to replace it.

7. In the upper right corner, click **Condition Mode** to view the expression you created in condition mode. Condition mode provides an easy-to-read interface for creating and viewing your expressions.

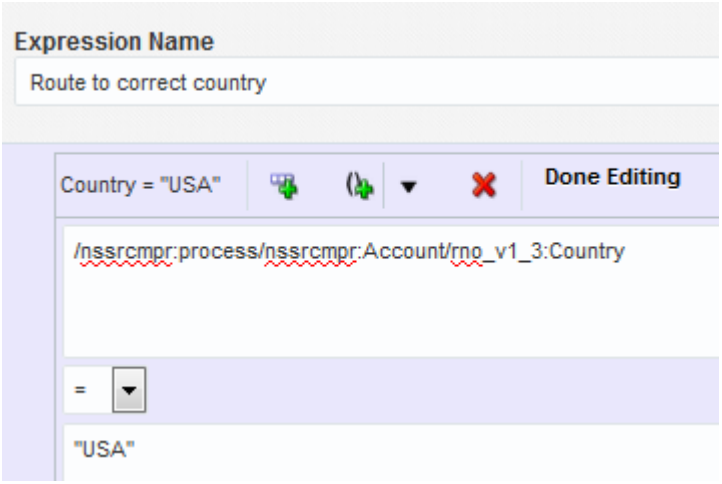
   Note the following details about accessing condition mode:

   - Condition mode can only be accessed if the expression field is empty or completely defined with an expression that returns true or false. If you only partially define an expression (for example, you drag an element to the expression field, but forget to define expression logic and a value such as = "USA"), you receive an error saying that you must provide a valid condition to access condition mode.
   - The **Condition Mode** button toggles to **Expression Mode**.

Note: At any time, you can click **Expression Mode** to view the entire XPath expression.

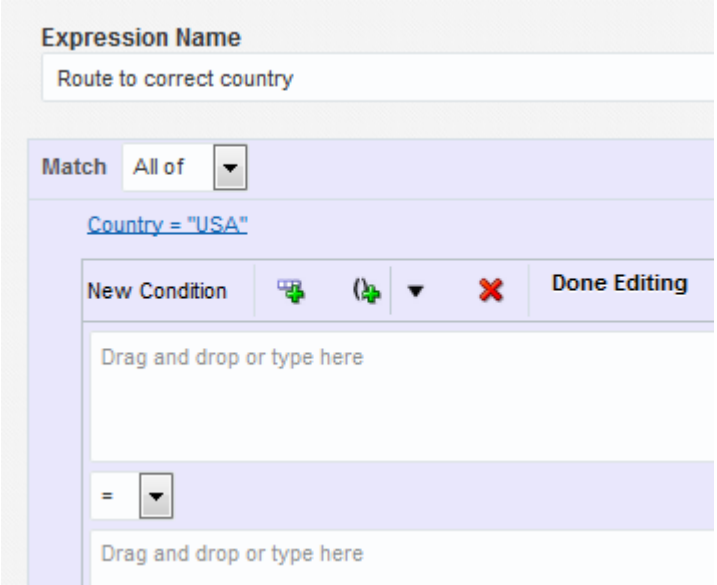
8. Click the expression.

This refreshes the page to display icons for adding additional conditions and conditions groups. Groups enable you to combine multiple conditions into a single logical expression.



9. Click the **Add Condition** icon (first icon) to add additional condition expressions.

   This creates an additional field for entering additional expression logic. The message **Drag and drop or type here** is displayed in this field.



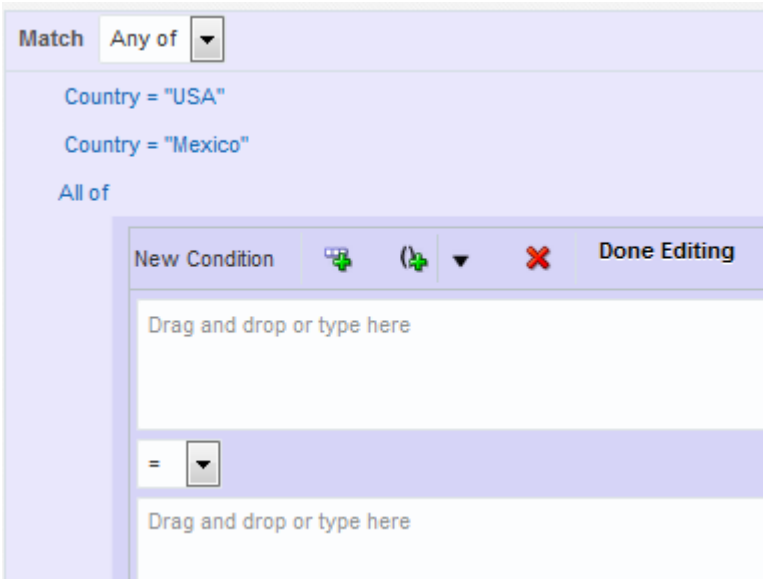
10. Drag an element from the **Source** area to the first **Drag and drop or type here** field (for this example, the **Country** element is again added).

11. Select an operator (for example, **=**, **>**,**!=**, and so on) and enter a value (for this example, `"Mexico"` is added).

12. From the **Match** list, select an option. This list is hidden until at least two conditions are defined.

    - **Any of**: Select if any of the added expressions must be true. This equates to an OR condition in the entire XPath expression shown in expression mode.
    - **All of**: Select if all expressions must be true. This equates to an AND condition in the entire XPath expression shown in expression mode.

13. Select the **Add Group** icon (second icon) to group a series of conditions. This option enables you to build a number of conditions within a single group. The group is identified by the gray outline and the indentation.



14. Add an element from the **Source** area.

    For this example:

    - The **DisplayName** element is added to the first **Drag and drop or type here** field.
    - The not equal operator (**!=**) is selected.
    - The **Country** element is added to the second **Drag and drop or type here** field.

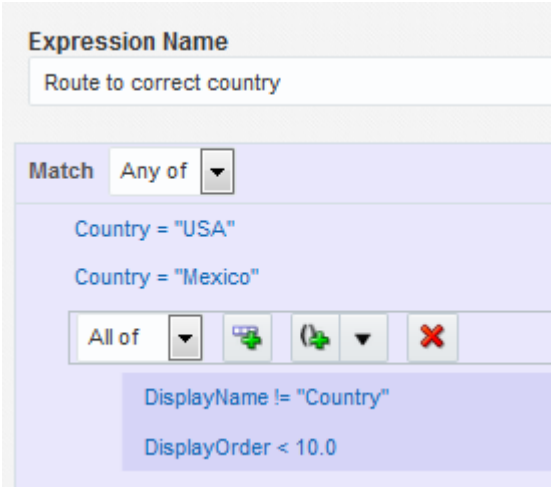
15. Click the **Add Condition** icon (first icon) to add an additional condition expression within the group.

    For this example:

    - The **DisplayOrder** element is added to the first **Drag and drop or type here** field.
    - The less than operator (**<**) is selected.
    - A value of `10` is entered in the second **Drag and drop or type here** field.

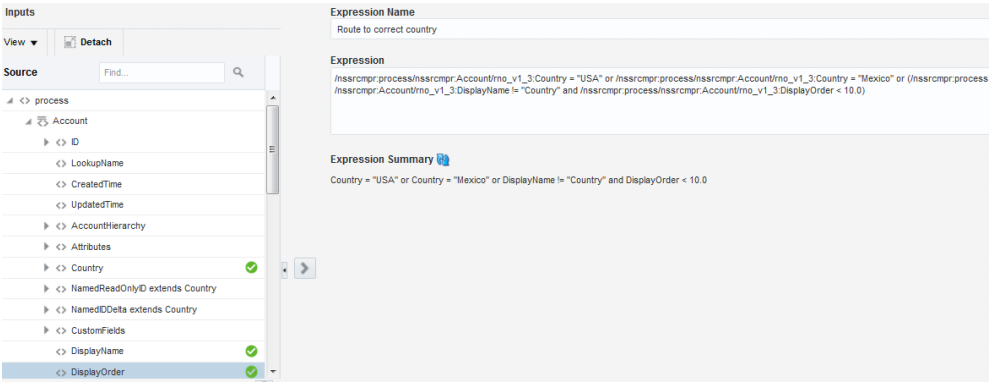
16. Continue building your group condition, as necessary.

    When complete, the expression is displayed. For this example, there are the conditions: if **Country** is **USA** OR **Country** is **Mexico** OR **DisplayName** does not equal country and **DisplayCount** is less than **10**, the integration continues.



17. Click **Expression Mode**.

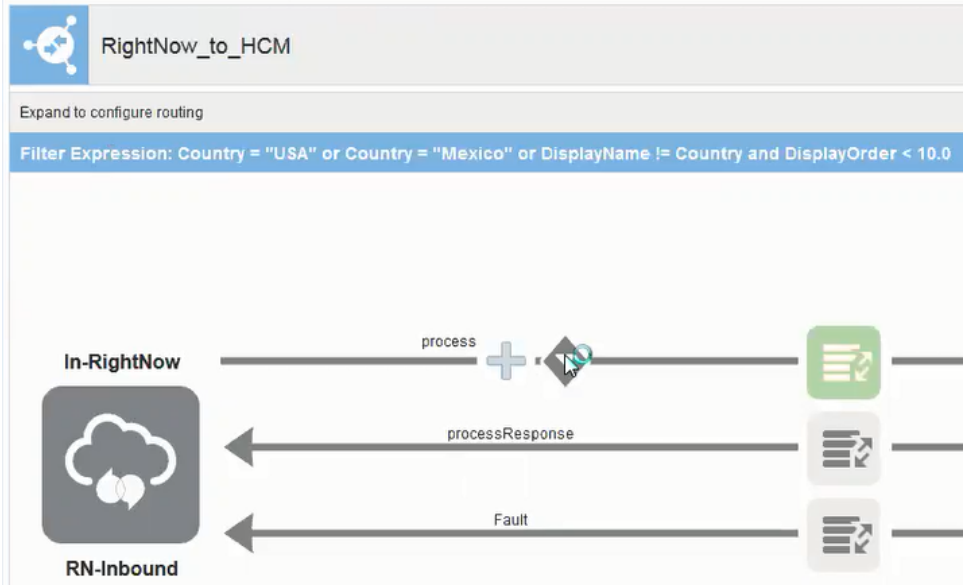
    Note the entire XPath expression and the expression summary at the bottom. The selected elements are displayed (no matter their level of depth in the tree) and identified by green checkboxes in the **Source** area.



18. If you want, you can place your cursor in the XPath expression and edit it as necessary (for example, change `USA` to `Canada`), then click the **Expression Summary** icon to refresh the calculation. If you make an error when editing the XPath expression (for example, forget to add a double quote to a value), an error message is displayed.

19. Click **Save** to view the expression in read-only mode. You can also click **Done Editing** at any time during the creation process to view the expression in read-only mode.

20. Click **Exit Expression Builder** to return to the integration. The user-friendly expression is displayed in the blue banner above the integration.



Deleting Routing Paths

You can delete routing paths that have been created on different target endpoints in an integration.

There are two methods for deleting routing paths:

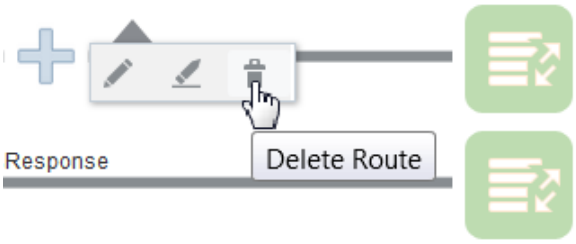
- Delete the routing path and expression filter.
- Delete the endpoint and routing path, but retain the expression filter.

Deleting the Routing Path and Expression Filter

To delete the routing path and expression filter:

1. In the Integrations page, select the integration in which to delete a routing path.
2. Expand the **Routing Drawer** icon to display the diagram of routing paths.
3. Above the integration, select the routing path to delete.

4. Click the **Filter** icon.
5. Select **Delete Route** from the menu that is displayed.



6. Click **Yes** when prompted to confirm.

   This action deletes the routing path, including the expression filter and the request mapping for the selected path. The diagram above the integration shows that the routing path is deleted.

Deleting the Endpoint and Routing Path

To delete the endpoint and routing path:

1. In the integration, click the target endpoint to delete.
2. Click **Delete** in the menu that is displayed.
3. Click **Yes** when prompted to confirm.

   This action deletes the target endpoint and routing path. The diagram above the integration shows that the routing path is deleted. Within the integration, only the expression remains defined in the integration because it is not using anything from the deleted target endpoint.

Mapping Integration Cloud Service Data

Use the mapper to drag fields from the source structure to the target structure to map elements between the two.

Topics

- Creating Mappings
- Modifying Mappings
- Deleting All Mappings
- Mapping Faults
- Adding Customized Mappings to Prebuilt Integrations
- Removing Customized Mappings from Prebuilt Integrations
- Regenerating a WSDL File for Integrations

For information on using the mapper, see Mapping Data of *Using the Oracle Mapper*.

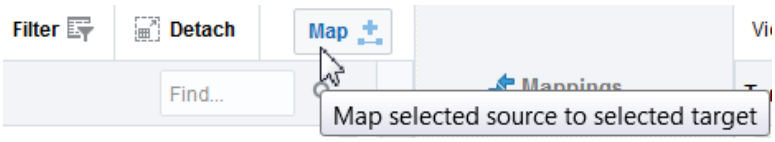
Creating Mappings

You can map fields directly from the source data structure to the target data structure in the mapper.

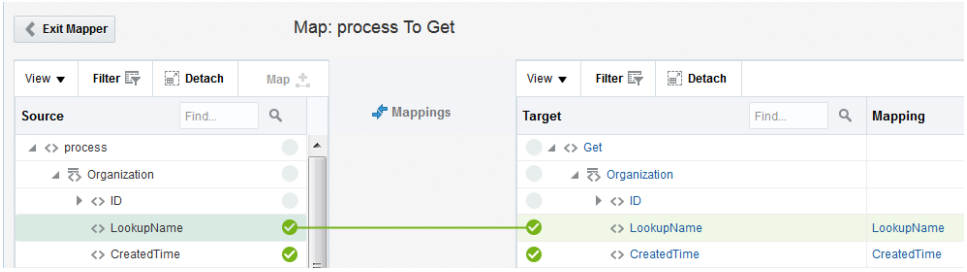
To create mappings:

1. In the middle of the integration, click the **Mapper** icon for the request, response, or fault map to edit.
2. Click **Create**.

3. To map fields directly, perform one of the following steps:
   a. Click a field in the source and drag it to the corresponding field in the target.
   b. Click the source and the target fields, and then click **Map+**.



The name of the source field appears in the target Mapping column, and a green check mark icon appears next to both fields. The most recently mapped fields are connected by a green line. Click the green check mark of other sources and targets to see their current mappings. You cannot drag and drop onto a target that is already mapped.



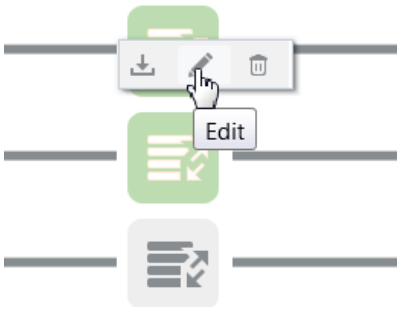
Note: In the mapping summary, the full names of extended elements are not displayed.

Modifying Mappings

Once you create a mapping in an integration, you can return to the mapping and make any necessary changes to how you mapped your data. The integration in which you want to edit the mappings cannot be active.

To modify a data mapping:

1. In the middle of the integration, click the **Mapper** icon for the request, response, or fault map to edit.
2. Click **Edit** to invoke the mapper.



3. Make appropriate updates to the mappings.
4. When complete, click **Save**, then click **Exit Mapper**.

For information on using the mapper, see Mapping Data of *Using the Oracle Mapper*.

Deleting All Mappings

You can delete all mappings in the mapper. This action deletes all source-to-target mappings in the mapper and all mapper statements created in the Mapping Builder.

1. Click the **Mapper** icon in the middle of the integration for the map to delete. For this example, the request mapper is selected, but you can also delete all mappings in another mapper, such as the response mapper, or any request or response enrichment mapping you created.

2. Click **Delete**.

3. Click **Yes** when prompted to confirm.

   The green shading is removed from the mapper, indicating that the mapper is now empty.

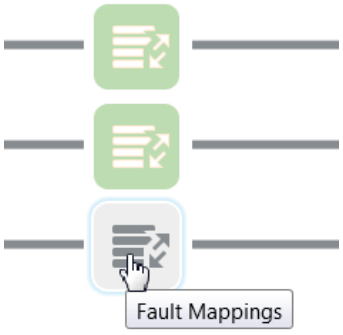
   For information on using the mapper, see Mapping Data of *Using the Oracle Mapper*.

Mapping Faults

You can map portions of a message into the fault message to compose a description that helps you understand the fault.

To map a fault:

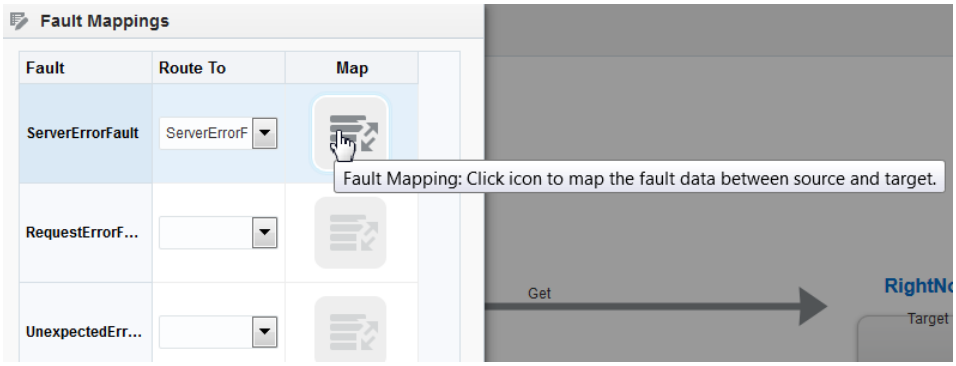
1. Click the **Fault Mappings** icon in an integration.



2. For each fault type, do the following:

   a. Under **Route To**, select the type of fault.

   b. Under **Map**, click the **Mapper** icon of the fault map to perform mapping.



The mapper appears with the source fault data structure on the left and the target fault data structure on the right. When returning from the mapper, the map icon changes color to indicate it is complete.

3. Once you create a mapping in an integration, click **Save** in the toolbar.
4. Return to the mapping to make any necessary changes to how you mapped your data.

   For information on using the mapper, see Mapping Data of *Using the Oracle Mapper*.

Adding Customized Mappings to Prebuilt Integrations

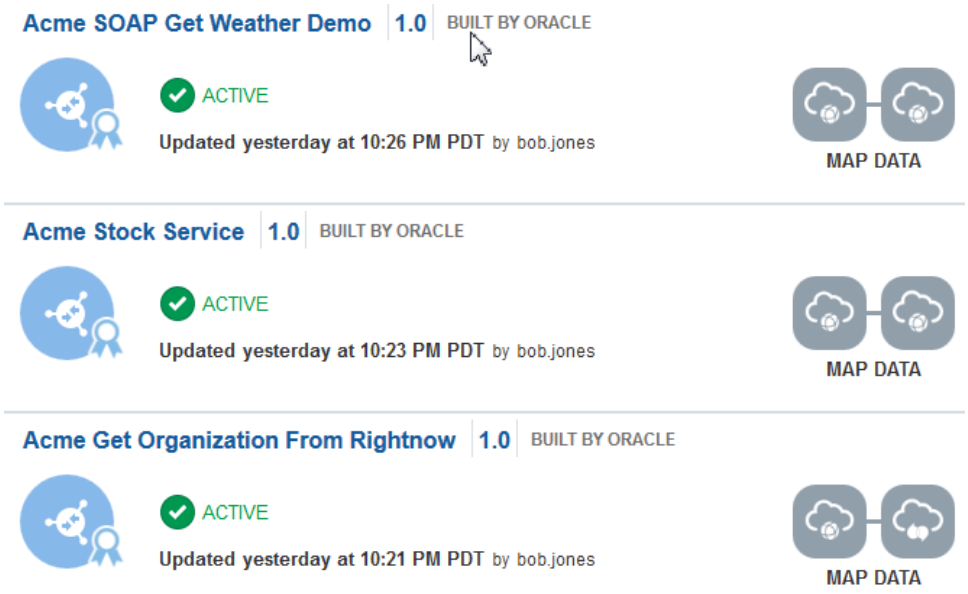
It is a common practice to customize the application endpoints of the prebuilt integrations that you import into Integration Cloud Service from the Oracle Marketplace (for example, adding custom fields). As a result, you must customize the integration mappings to take advantage of these custom fields. Integration Cloud Service enables you to customize the mappings in the prebuilt integrations that you import from the Oracle Marketplace. This action creates a customized mapping layer on top of the base mapping file, which is not modified. You can only add customized mappings to prebuilt integrations imported from the Oracle Marketplace, and not to integrations you or another user created.

To add customized mappings to prebuilt integrations:

1. In the Integration Cloud Service toolbar, click **Designer**.

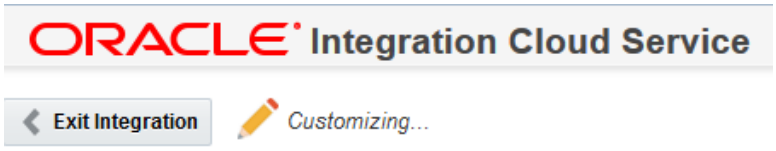
2. Click **Integrations**.

   The Integrations page is displayed.
3. Locate the name of the prebuilt integration to customize. Prebuilt integrations are designated with the words **BUILT BY ORACLE** to the right of the integration name.

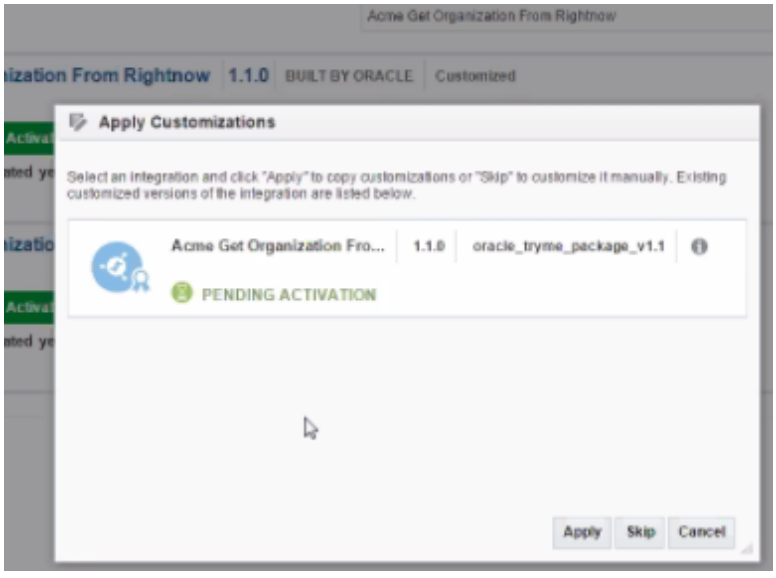


4. From the menu at the far right of the integration name, select **Customize**.

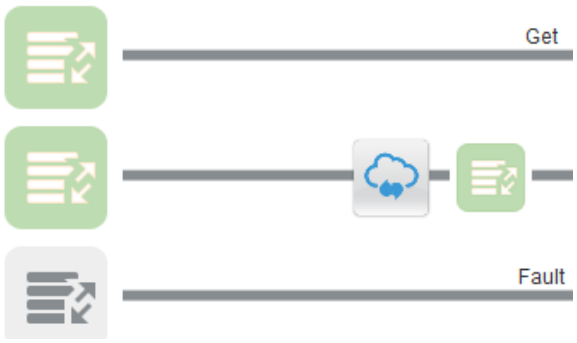
   The message **Customizing...** is displayed above the integration.



   If the existence of more than one customized version of the same prebuilt integration is detected, a dialog is displayed that shows a list of versions from which to copy customizations. You can select a version and click **Apply**, or select **Skip** to bypass the copying of customizations and create your own customizations in the mapper, as described in the steps below.



5. Click the icon for the type of mapping you want to customize. You can customize request, response, fault, enrichment source, and enrichment response mappings.



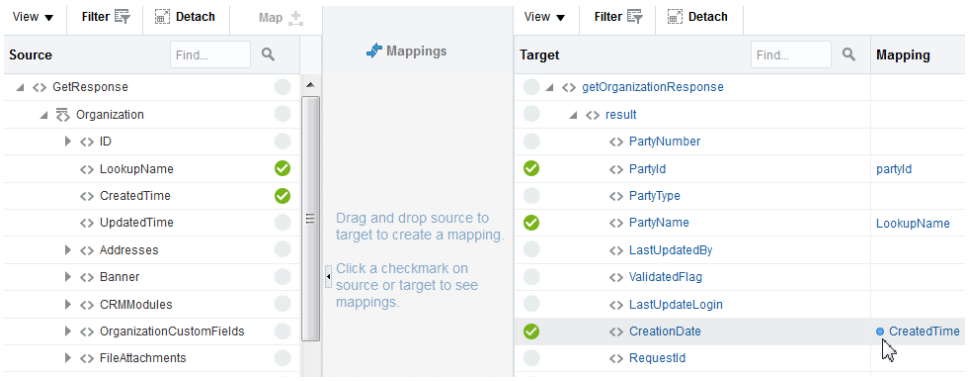
An icon for customizing the selected mapper is displayed.

6. Click **Customize**.

The mapper is displayed in customization mode.

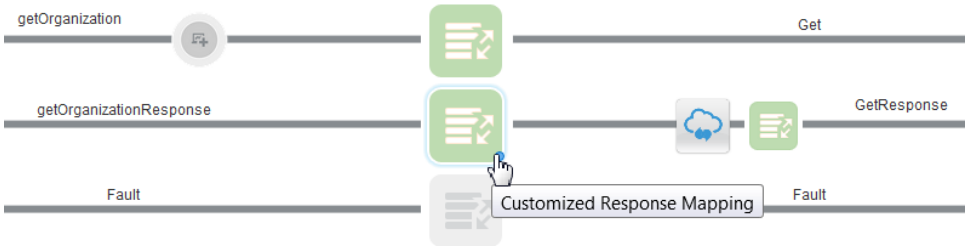
7. Drag and drop source fields to target elements.

Blue dots are added to the left of the mapped target elements in the **Mapping** column to indicate that these are customized mappings. These mappings are added to a customized layer on top of the base mapping file, which is not modified. This dot differentiates the customized mappings from the regular mappings created as part of the prebuilt integration, which are displayed without a blue dot.



8. Click **Save**, then click **Exit Mapper**.

A blue dot with the words **Customized Response Mapping** is displayed in the lower right corner of the icon for the customized mapper (for this example, the response mapper was customized). The other mappers do not have a blue dot because they were not customized (for this example, the request, fault, and request enrichment mappers).



For information on using the mapper, see Mapping Data of *Using the Oracle Mapper*.

Removing Customized Mappings from Prebuilt Integrations

You can remove the customized mappings that you added to prebuilt integrations that you imported from the Oracle Marketplace. You can remove all customized mappings or specific subsets of mappings (for example, request, response, faults, enrichment source, or enrichment response mappings).

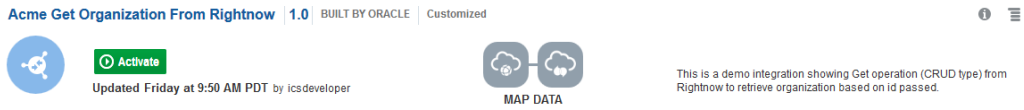
To remove customized mappings from prebuilt integrations:

1. In the Integration Cloud Service toolbar, click **Designer**.

2. Click **Integrations**.

   The Integrations page is displayed.

3. Locate the prebuilt integration in which you want to remove the customized mappings. Prebuilt integrations that have been customized are designated with the words **BUILT BY ORACLE** and **Customized** to the right of the integration name.



4. Click the integration name.

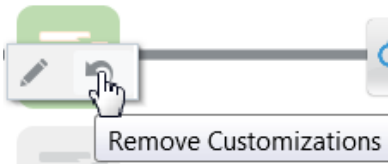
   You can remove all customized mappings added to the integration or specific subsets of mappings (for example, request, response, fault, request enrichment, or response enrichment mappings).

5. To remove all customized mappings from the integration, perform the following step:

a. Click **Remove All Customizations** in the upper right corner.

6. To remove specific subsets of request, response, fault, request enrichment, or response enrichment mappings, perform either of the following steps:

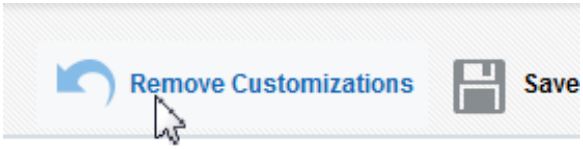
a. Click the mapper icon, then click **Remove Customizations** for the customized mapping to delete (for this example, the customized response mapping is selected).



or

a. Click the mapper icon, then click **Customize** to access the specific mapper.

b. Click **Remove Customizations** in the upper right corner of the mapper page.



7. Click **Yes** when prompted to confirm your selection.

This action removes the specific customized mappings in the integration. Note that the blue dots that previously identified the customized mappings are removed. The existing mappings that are part of the original prebuilt integration are not removed.

Regenerating a WSDL File for Integrations

After you clone an integration, customize a prebuilt integration, or import an existing integration into Integration Cloud Service, you update the connection information (WSDL, username. and password) according to the requirements of your integration environment. If the connection WSDL you specify contains any custom fields or if the connection WSDL is updated with a different version, they are not displayed in the mapper. To get custom fields or updated fields to appear in the mapper, you must regenerate the endpoint in Oracle Integration Cloud Service.

As an example, you may have a scenario in which the WSDLs with one of your connections (for example, a Salesforce connection) change frequently and you must be able to uptake the latest WSDLs into your integrations. By regenerating the WSDL file, the custom fields of imported mappings are not deleted, and are available for editing, as needed. This eliminates the need for remapping source and target elements completely from scratch.

To regenerate a WSDL file for integrations:

Note: There cannot be root level differences between the old and new WSDLs. If there are differences, WSDL regeneration fails.

1. In the Integration Cloud Service toolbar, click **Designer**.

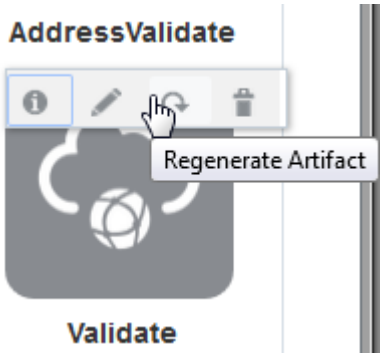
2. Click **Integrations**.

   The Integrations page is displayed.

3. Click the name of the integration in which to regenerate the WSDL. Customized integrations are designated with the words **BUILT BY ORACLE** and **Customized** to the right of the integration name.

   You can regenerate the WSDL for an individual endpoint or the WSDLs for all endpoints in an integration.

4. To regenerate the WSDL for a single endpoint in the integration, click the appropriate source, target, request enrichment, or response enrichment icon.



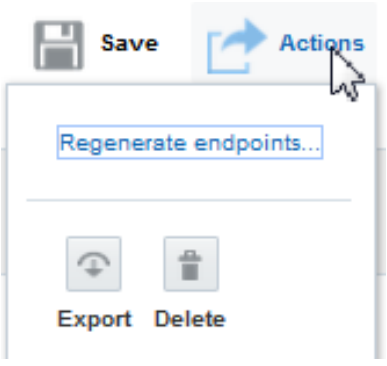
a. Select **Regenerate Artifact**.

You are prompted with a message indicating that WSDL regeneration impacts the mappings in the integration.

b. Click **Yes**.

This regenerates the WSDL and any dependent artifacts so that any custom elements appear during mapping. The imported mappings from any prebuilt integration are not deleted. The maps are validated and any warnings (identified by yellow icons) or errors (identified by red icons) for the impacted maps are displayed. If warnings and errors both exist for a single mapper, only a single error icon is displayed. Icons indicating that this mapper is customized (identified by the blue icons) are displayed at the bottom of the mapper.

5. To regenerate the WSDLs for all endpoints in the integration, select **Actions** at the top of the page.



a. Click **Regenerate endpoints**.

You are prompted with a message indicating that WSDL regeneration impacts the mappings in the integration.

b. Click **Yes**.

This regenerates the WSDLs and any dependent artifacts with the same behavior as described in Step 4.

For information on using the mapper, see Mapping Data of *Using the Oracle Mapper*.

Creating Lookups

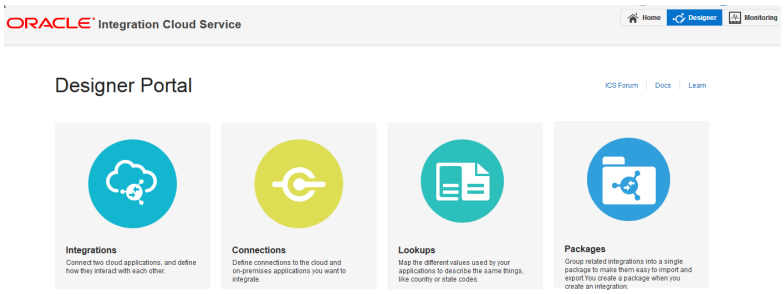
A lookup associates values used by one application for a specific field to the values used by other applications for the same field. This provides the capability to map values across vocabularies or systems. For example, you can map country codes, city codes, currency codes, and so on.

Topics

Creating a Lookup

Create a lookup to map values between applications.

1. From the Designer Portal, select **Lookups**.

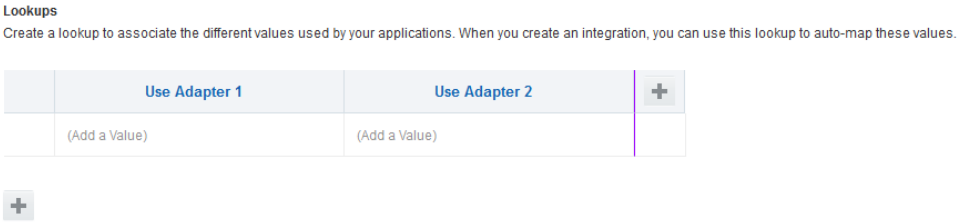


2. Click **Create New Lookup**.

The New Lookup — Information dialog is displayed.

3. Enter a name and optional description for the lookup.
4. Click **Create**.

The Lookup page is displayed.



Adding Adapters to a Lookup

Add adapters to a lookup to map values between connections.

1. Click **Use Adapter 1**.

   The Select Adapter dialog is displayed.

2. Click **Select** to add an adapter.

   The adapter name is now the column heading.

3. Click **Use Adapter 2**.

   The Select Adapter dialog is displayed.

4. Click **Select** to add an adapter.

   The adapter name is now the column heading.

5. Repeat to add more adapters to the lookup.

Adding Values to a Lookup

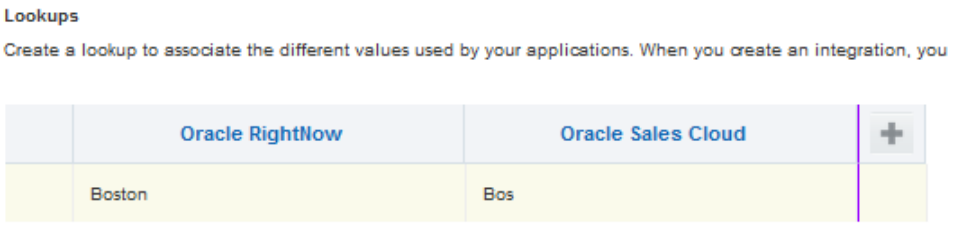
Map values from one adapter to another.

1. In the row for the first adapter, click **Add a Value**.

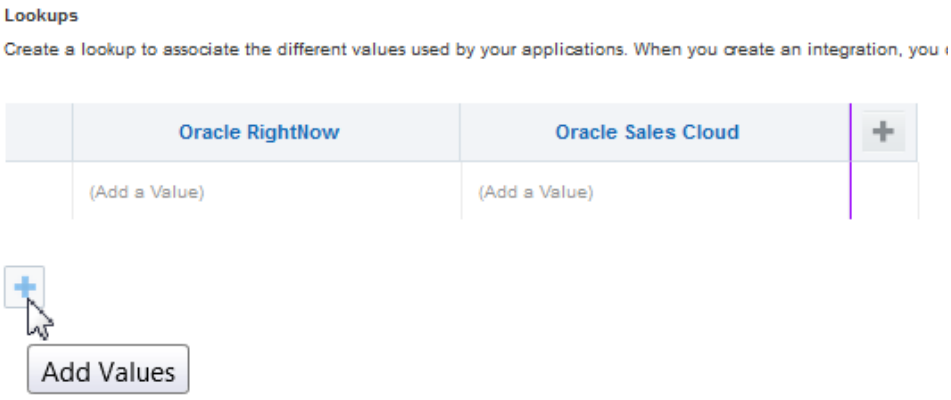
   Enter a value under your first adapter.

2. In the row for the second adapter, click **Add a Value**.

   Enter a value under your second adapter.



3. Click the **Add Values** icon to add another value to the lookup.

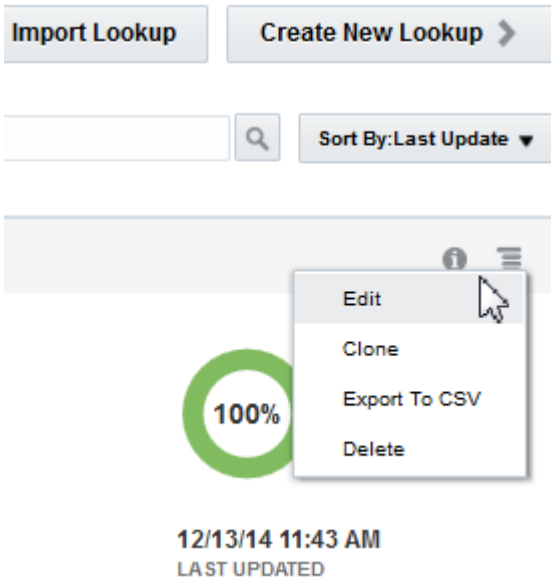


4. Click **Save**, then click **Exit Lookup**.

Cloning a Lookup

You can clone a copy of an existing lookup. It is a quick way to create a new lookup with similar information.

1. On the Lookups page, click **Clone** from the Lookup menu.



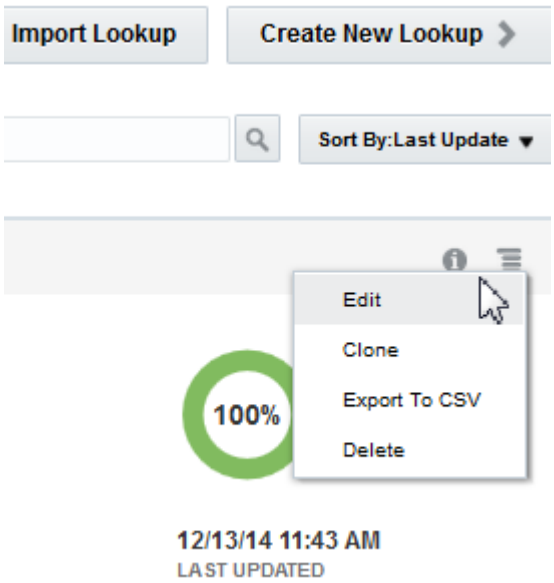
The Clone Lookup dialog is displayed.

2. Enter the lookup information.
3. Click **Clone**.
4. Click **Edit** to further configure your cloned connection.

Deleting a Lookup

You can delete a connection from the Lookup menu.

1. Click **Delete** from the Lookup menu.



   The Delete connection dialog is displayed.

2. Click **Yes** to confirm deletion.

Importing Map Files

Review the following topics to learn how to import map files into Oracle JDeveloper and Integration Cloud Service.

Topics

- Importing a Map File into Oracle JDeveloper
- Importing a Map File into Integration Cloud Service

For information about exporting an integration that includes a map file that you want to edit in Oracle JDeveloper, see Exporting an Integration.

Importing a Map File into Oracle JDeveloper

You can import an Integration Cloud Service archive file into an Oracle Service Bus project in Oracle JDeveloper. The archive file can include a map file that is largely complete in content or a map file that is empty of content. This action enables you to perform advanced XSLT tasks (create variables, use templates, and so on) in Oracle JDeveloper that you cannot perform in the Integration Cloud Service mapper. After you complete these advanced tasks in Oracle JDeveloper, you can save and re-import the map file into Integration Cloud Service.

1. See Exporting an Integration for instructions on exporting an integration that includes the map file you want to edit in Oracle JDeveloper.
2. Create an Oracle Service Bus application with a project in Oracle JDeveloper.

3. In the application navigator, right-click the Oracle Service Bus project and select **Import**.

   The Import dialog is displayed.

4. Select **Service Bus Resources**, and click **OK**.

   The Import Service Bus Resources wizard is displayed.

5. Select **Zipped/Archived Resources**, and click **Next**.

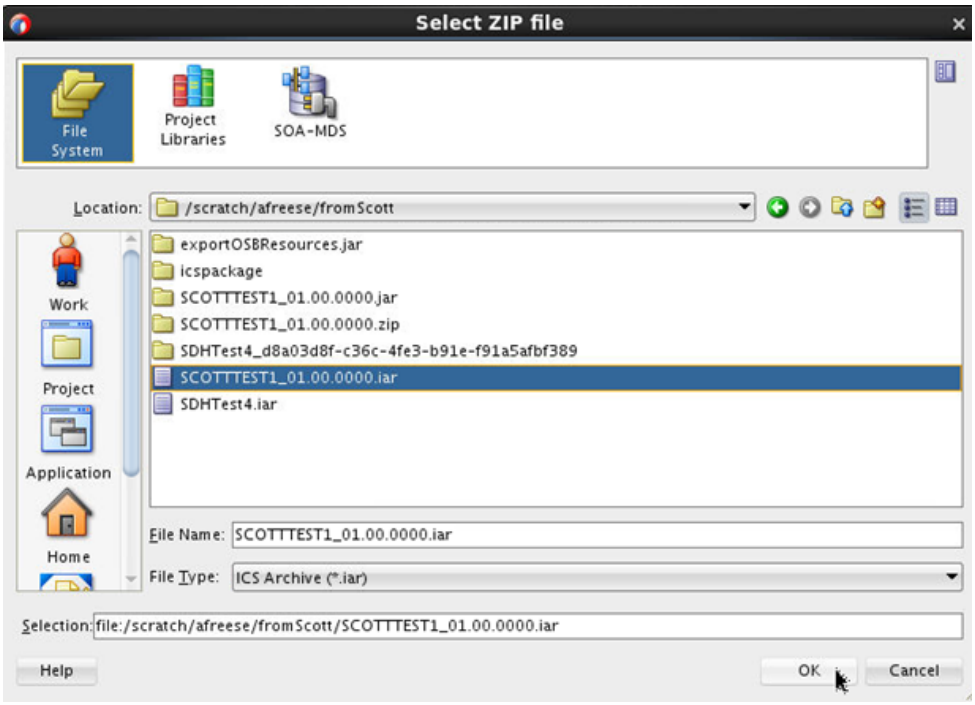
6. Click the **Browse Zip Source** icon to the right of the **Zip Source** field.

   The Select ZIP File dialog is displayed.

7. If using Oracle JDeveloper 12.2.1, perform the following steps:

   a. From the **File Type** menu, select **ICS Archive (*.iar)**.

   b. Browse for and select the Integration Cloud Service IAR archive file that you previously exported.



8. If using Oracle JDeveloper 12.1.3, perform the following steps:

   a. Ensure that you first rename the .iar file extension to .zip.

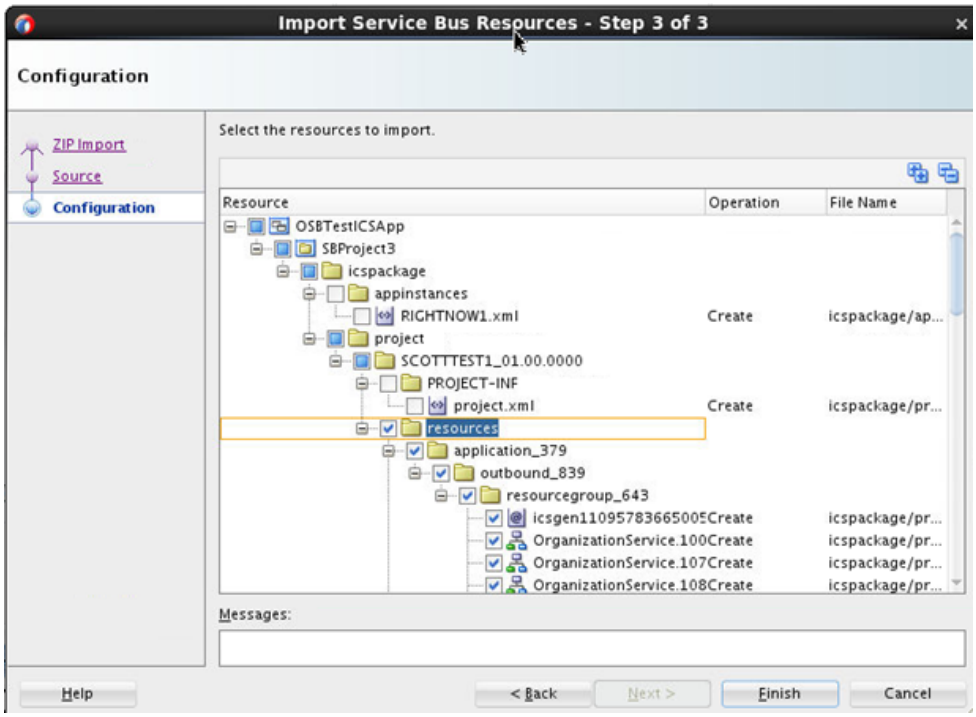
   b. Browse for and select the ZIP file to import.

9. Click **OK**, then click **Next** on the wizard page.

   The contents of the JAR file are displayed and can be selected for import.

10. Select the **resources** folder in which to import the archive file. Note that the entire **Resource** tree is selected by default, including everything above the hierarchy node

that you want to select. Ensure that you deselect the parts above the relevant hierarchy node, then click **Finish**.



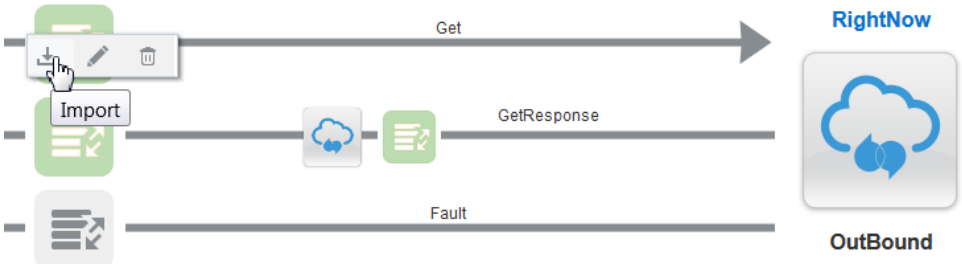
The resources are imported into the Oracle Service Bus project. You can now open the map file for editing with the XSLT Map Editor in Oracle JDeveloper.

Importing a Map File into Integration Cloud Service

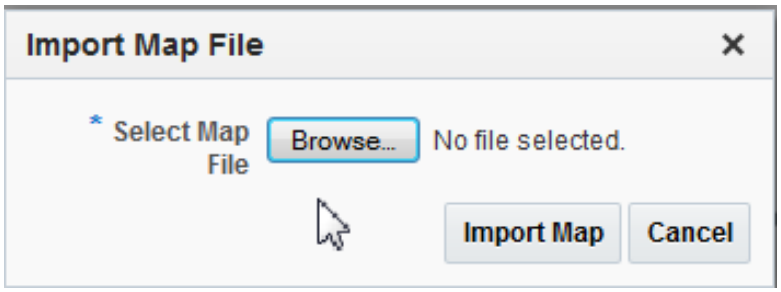
There may be scenarios in which you need to perform an advanced XSLT task (create variables, use templates, and so on) that you cannot perform in the Integration Cloud Service mapper. For these cases, you can export the integration, import the integration into Oracle JDeveloper, perform these advanced tasks in the map file in the XSLT Map Editor in Oracle JDeveloper, and then save and re-import the map file into Integration Cloud Service. The map file must be from an Oracle Service Project in Oracle JDeveloper.

> **Note:** You cannot edit a map file imported into the Integration Cloud Service mapper.

1. Click the **Designer** icon in the upper right corner, then click **Integrations**.
2. Click the specific integration in which to import the map file.
3. Click the mapper icon to display a menu.
4. Click **Import**.



5. Click **Browse** to select the map (.xsl) file. Note that while you exported the entire integration, you do not import the entire integration back into Integration Cloud Service. You only import the map file of the exported integration back into Integration Cloud Service.



Importing and Exporting Components

You can import and export both integrations and lookups to share them between Integration Cloud Service environments.

See the following topics:

- Exporting an Integration
- Importing an Integration
- Exporting a Lookup
- Importing a Lookup

Exporting an Integration

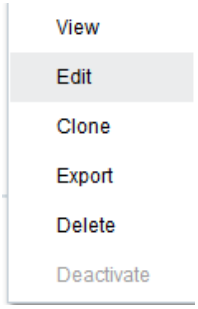
Once you create an integration, you can export that integration as a JAR file for use in other Integration Cloud Service environments or import the integration into Oracle JDeveloper to perform an advanced XSLT mapper task (for example, creating variables or using templates) that you cannot perform in the Integration Cloud Service mapper. After mapper editing in Oracle JDeveloper is complete, the mapper file can then be imported back into Integration Cloud Service. You can export an integration from either the Integration Designer or from the Integrations list.

To export an integration:

1. On the toolbar, click **Designer**.
2. On the Designer Portal, click **Integrations**.

3. Do one of the following:

   - Select the integration in the Integration Designer, click **Actions**, and then select **Export**.
   - Locate the integration to export in the Integrations list on the left side of the page. Click the vertical lines icon to the right in the integration's row, and then select **Export**.



4. In the dialog that appears, select **Save File**, and then click **OK**.
5. Save the file to the location you want.

The file is saved with a name that consists of the identifier plus the version number, and an IAR extension.

For information about importing an exported integration into the XSL Map Editor in Oracle Service Bus, see Importing a Map File into Oracle JDeveloper.

Importing an Integration

You can import integrations that were previously exported as a JAR file from Integration Cloud Service.

To import an integration:

1. On the toolbar, click **Designer**.
2. On the Designer Portal, click **Integrations**.
3. Above the Integrations list, click **Import Integration**.
4. In the Import Integration File dialog, click **Browse** to navigate to and select the file to import.
5. Click **Import**.

   If an integration already exists with the same identifier and version, you must confirm whether to overwrite the existing integration.

The imported integration appears in the Integrations list and you can customize or activate it.

Note: Even though the **Activate** button is enabled after you import an integration, you must first configure your connection endpoints. If you do not, you receive an error when trying to activate the integration. For information about editing connections, see Editing a Connection and Adapter Configuration Reference.

Exporting a Lookup

Once you create a lookup, you can export that lookup for use in other Integration Cloud Service environments. You can export a lookup from either the Lookup Designer or from the Lookups list.

To export a lookup:

1. On the toolbar, click **Designer**.
2. On the Designer Portal, click **Lookups**.
3. Do one of the following:
   - Locate the lookup to export in the Lookups list on the left side of the page. Click the vertical lines icon to the right in the lookup's row, and select **Edit**. In the upper right corner, click **Actions**, and select **Export**.
   - Locate the lookup to export in the Lookups list on the left side of the page. Click the vertical lines icon to the right in the lookup's row, and select **Export to CVS**.
4. In the dialog that appears, select **Save File**, and then click **OK**.
5. Save the file to the location you want.

The file is saved as a CSV file with the same name as the lookup.

Importing a Lookup

You can import lookups that were previously exported from Integration Cloud Service.

The file to import must have the following for the first row, where *table_name* is the name of the table as you want it to appear in Integration Cloud Service. This name cannot contain spaces.

```
DVM,table_name
```

The second row contains the names of the adapters that are being mapped. Use the following case-sensitive IDs for each adapter:

Adapter Name	Identifier to Use in the Import File
Oracle Eloqua Cloud	eloqua
Oracle Sales Cloud	osc
Oracle Messaging Cloud Service	oms
Oracle RightNow Cloud	rightnow

Adapter Name	Identifier to Use in the Import File
Oracle HCM Cloud	hcm
Oracle ERP Cloud	erp
Salesforce Cloud	salesforce

To import a lookup:

1. Locate the CSV file containing the lookup table you want to import.
2. On the toolbar, click **Designer**.
3. On the Designer Portal, click **Lookups**.
4. Above the Integrations list, click **Import Lookup**.
5. In the Import Lookup File dialog, click **Browse** to navigate to and select the CSV file to import.
6. Click **Import**.

   If a lookup already exists with the same identifier and version, you must confirm whether to overwrite the existing lookup.

The imported lookup appears in the Lookups list on the left. You can customize or activate it, if it is ready.

Assigning Business Identifiers for Tracking Fields in Messages

This section describes how to manage business identifiers that enable you to track fields in messages during runtime.

Topics

Assigning Business Identifiers

Business identifiers enable you to track payload fields in messages during runtime. You can specify up to three business identifier fields for tracking during design time. One of these fields must be selected as the primary business identifier field. The primary business identifier enables you to track fields across integration flows during runtime, and is always available. At runtime, the status of business identifiers is visible on the Tracking page and (if integration errors have occurred) the Errors page.

To assign business identifiers:

1. Click the **Designer** icon.

2. Click **Integrations**.

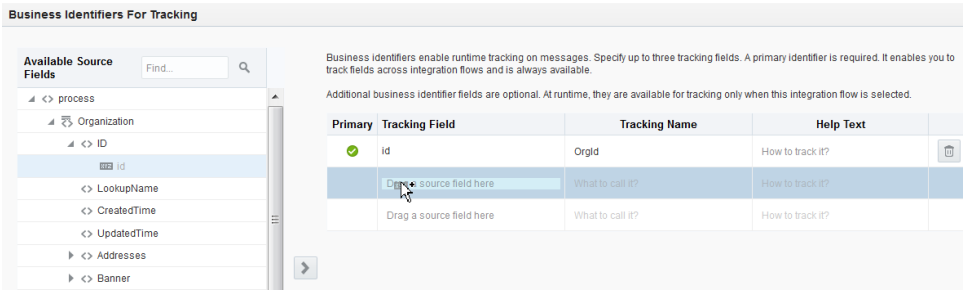
   The Integrations page is displayed.

3. Click the specific integration to which to add business identifiers. You can only add business identifiers to integrations that are *not* active. If an integration is active, you can only view its existing business identifiers.

4. In the upper right corner, click **Configuration**.

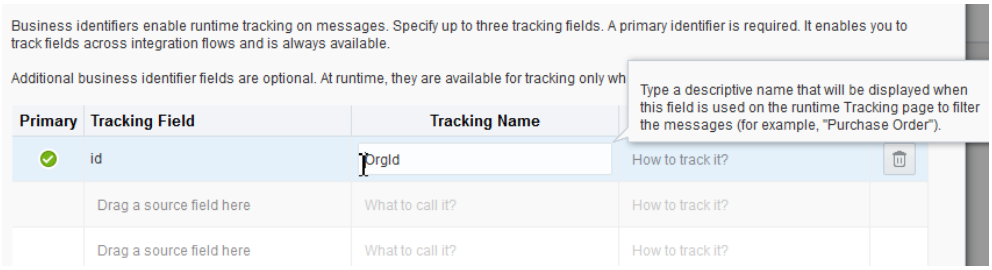
   The Business Identifiers For Tracking dialog is displayed. The source payload for the selected integration is displayed on the left side. You can only assign business identifiers to fields of source payloads. You cannot assign business identifiers to fields of target payloads.

5. From the **Available Source Fields** section, drag the payload field that you want to track to the **Drag a source field here** section.



6. Select the checkbox if you want to make this the primary business identifier. At least one identifier is required.

7. In the **Tracking Name** field, optionally enter a descriptive name to track during runtime (for example, `OrgId`). The name is displayed when this field is used to filter messages on the Tracking page or (if there is an integration error) the Errors page during runtime.



8. In the **Help Text** field, optionally enter instructions to enable users to know what to enter in this field during runtime (for example, `Enter a valid organization number`). These instructions are displayed inside the empty field when it is used on the runtime Tracking page to filter messages .

9. Click **Done**.

Deleting Business Identifiers

You can delete business identifiers that track fields in messages during runtime.

To delete business identifiers:

1. Click the **Designer** icon.
2. Click **Integrations**.

   The Integrations page is displayed.
3. Click the specific integration to which to add a business identifier. You can only add business identifiers to integrations that are *not* active. If an integration is active, you can view, but not edit, the contents of the Business Identifiers for Tracking dialog.
4. In the upper right corner, click **Tracking**.
5. Click **Tracking** in the menu that is displayed.

   The Business Identifiers for Tracking dialog is displayed.
6. At the far right, click the **Delete** icon for the business identifier to delete. If you delete the primary business identifier, select a new one. Without a primary identifier, you cannot track fields across integration flows during runtime on the Tracking page.

Managing Packages

You can group integrations into a package. When you import or export the package to or from Integration Cloud Service, all integrations in that package are imported or exported.

Topics

- Viewing the Integrations in a Package
- Importing a Package
- Exporting a Package
- Updating a Package
- Deleting a Package

When you create an integration, you can also create a package or select an existing package in which to include the integration. For more information, see Creating an Integration. For conceptual information about packages, see About Integration Cloud Service Packages.

Viewing the Integrations in a Package

You can view the integrations included in a package.

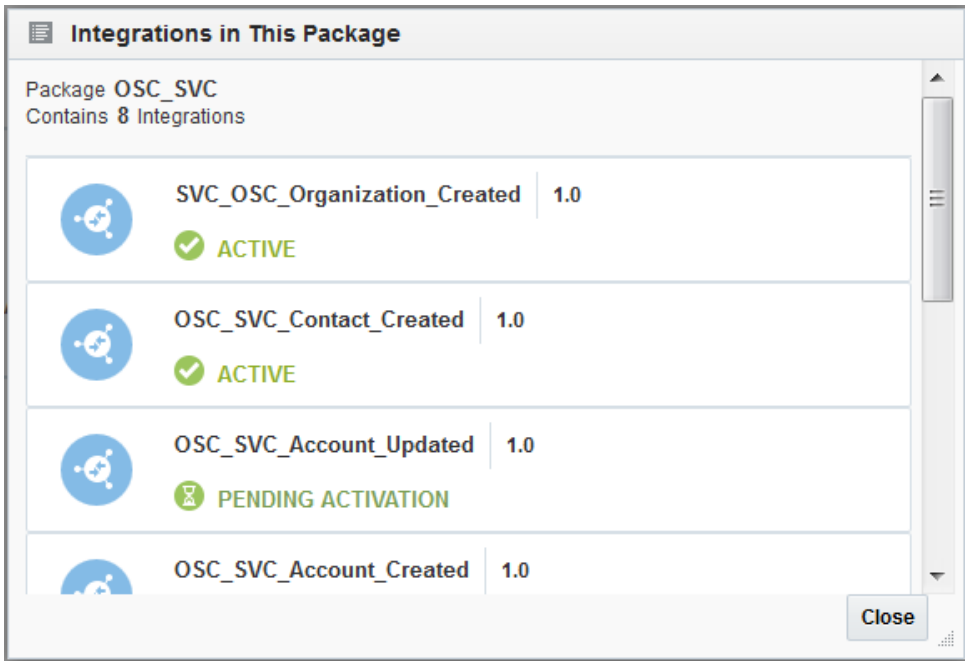
1. In the Integration Cloud Service toolbar, click **Designer**.

2. Click **Packages**.

   The Packages page is displayed. The package names and the number of integrations included in each package are displayed. You can filter the display of packages by entering the full or partial name (using a wildcard value of *) and clicking the **Search** icon. If you have not yet created or imported a package into Integration Cloud Service, this page is empty.

3. Click the name of the package or select **View Integrations** from the menu at the far right.

   The integrations included in that package and their current states are displayed (for example, pending activation or active).



4. In the navigator pane, click **Integrations** to access the Integrations page for viewing these integrations.

Importing a Package

You can import a package of integrations into Integration Cloud Service from the Packages page. The Packages page enables you to import packages that you or other users have created. To import packages that consist of integrations that are prebuilt by Oracle, you must go to Oracle Marketplace.

1. In the Integration Cloud Service toolbar, click **Designer**.

2. Click **Packages**.

3. Click **Import Package**.
4. Browse for and select the packages archive (PAR) file when prompted.
5. Click **Import**.

   The package is added to the list on the Packages page.

For information about importing prebuilt packages from Oracle Marketplace, see Importing a Prebuilt Integration.

Exporting a Package

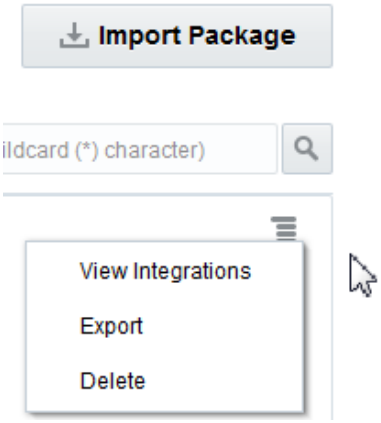
You can export a package of integrations from Integration Cloud Service on the Packages page. This action exports all the integrations included in that package.

1. In the Integration Cloud Service toolbar, click **Designer**.

2. Click **Packages**.

   The Packages page is displayed.

3. From the menu at the far right, select **Export**.



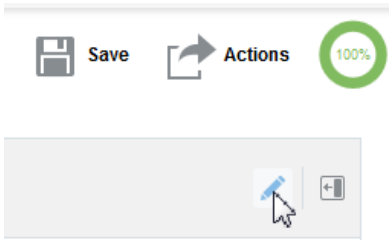
4. Save the package (PAR) file of integrations to a file system location when prompted. The individual integrations inside the PAR file are exported as integration archive (IAR) files.

Updating a Package

You can update the package in which your integration is included. For example, you can create a new package for your integration or move your integration to an existing package.

1. On the Integrations page, click the integration of the package that you want to update. The integration must not be active.

2. Click the **Edit** icon.



The Update Integration dialog is displayed.

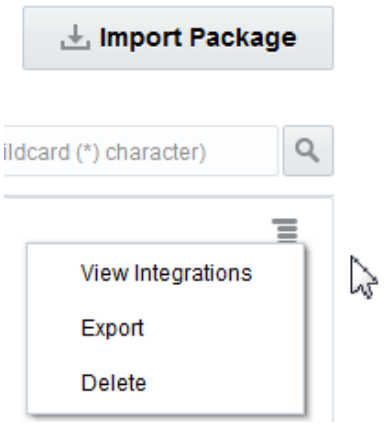
3. In the **Package Name** field, enter a new package name or enter an existing package name (as you type the initial letters, the existing package is displayed) to move your integration to an existing package.
4. Click **OK**.
5. Click **Save**, then click **Exit Integration**.
6. In the navigation pane, click **Packages**.
7. Click the package name you specified in the Update Integration dialog to see your integration.

Deleting a Package

You can delete a package. This action deletes the package and all integrations included in that package.

1. In the Integration Cloud Service toolbar, click **Designer**.

2. Click **Packages**. The Packages page is displayed.
3. From the menu at the far right, select **Delete**.



4. Click **Yes** when prompted to confirm your selection. The package and all of its integrations are deleted.

Managing Agent Groups and the On-Premises Agent

You must create an agent group and install the on-premises agent before you can create an integration in which messages are exchanges between your on-premises applications and Oracle Integration Cloud Service.

Topic

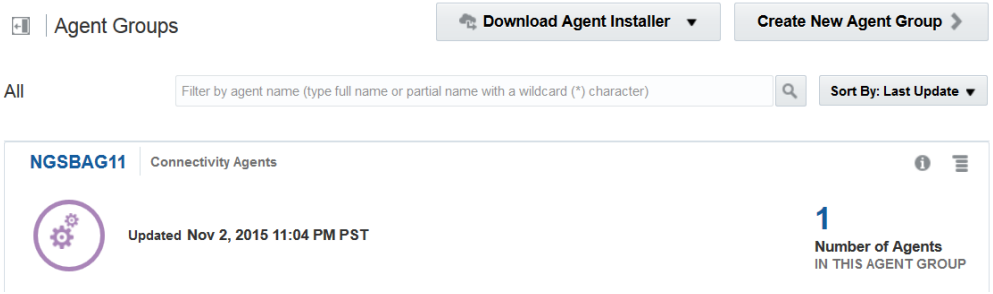
Creating an Agent Group

You must create an agent group in Oracle Integration Cloud Service before you can run the agent installer. When you install the on-premises agent in your environment, you associate the on-premises agent with the agent group identifier. Only one on-premises agent can be associated with an agent group. For a single Oracle Integration Cloud Service instance, you can create up to three agent groups.

To create an agent group:

1. In the Integration Cloud Service toolbar, click **Designer**.

2. On the Designer Portal, click **Agents**.
3. Click **Create New Agent Group**.



The New Agent Group — Information dialog is displayed.

4. Enter the following information:

Field	Description
Agent Group Name	Provide a meaningful name so that others can understand the agent name. The name must be unique among all agent names in the system. The name can consist of the following: • Letters (A-Z, a-z) • Numbers (0-9) • Spaces ( ) • Special characters (_ -) The name must not begin or end with a space and cannot be longer than 50 characters.
Identifier	Accept the default identifier value or change it, if necessary. The identifier is initially the same as the agent group name you provided, but in upper case. When you install the on-premises agent, you must specify the identifier value. For more information, see Downloading and Running the On-Premises Agent Installer.
Agent Type	**Connectivity Agent** is displayed and cannot be edited. The connectivity agent supports integrating with on-premises systems. The agent group references only connectivity agents.
Description	Provide a meaningful description so that others can understand the responsibilities of the agent group.

Downloading and Running the On-Premises Agent Installer

You must download the agent installer from Oracle Integration Cloud Service and run the installer to install the on-premises agent in your local environment. During installation, you associate the agent with the agent group identifier you generated in Oracle Integration Cloud Service. After completing on-premises agent installation, you must apply a patch.

System Requirements

You must satisfy the following prerequisites on your on-premises host before running the agent installer in a production environment:

- Ensure that you have created the agent group. You must specify the agent group identifier when installing the on-premises agent. For information, see Creating an Agent Group.
- Install JRE version 1.7 or higher.
- Set the `JAVA_HOME` property to the location of the JDK installation. For example:

  ```
  JAVA_HOME=/usr/java/jdk1.7.0_79
  ```

- Install the on-premises agent on Linux OEL version 5 or 6 only.

- Provide a minimum of 8 GB memory with 4 GB of heap size dedicated for the agent JVM. If you want to include any other processes on that host besides the on-premises agent, it is strongly recommended that you increase physical memory to greater than 8 GB.
- Set the number of worker threads to a value appropriate to your environment. By default, the on-premises agent is set to 3. This may be sufficient for some low-load environments and most demo environments. However, for high-load environments, the agent instance may need additional tuning. Increase concurrency on the on-premises agent installation host by setting `agentWorkerThreads` to a recommended value of `40` in the `CpiAgent.properties` file. This change requires an agent restart.

  ```
  agentWorkerThreads=40
  ```

  For information about editing this file, see Updating Property Values.
- Do not install the on-premises agent in an environment that includes an existing Oracle WebLogic Server installation.

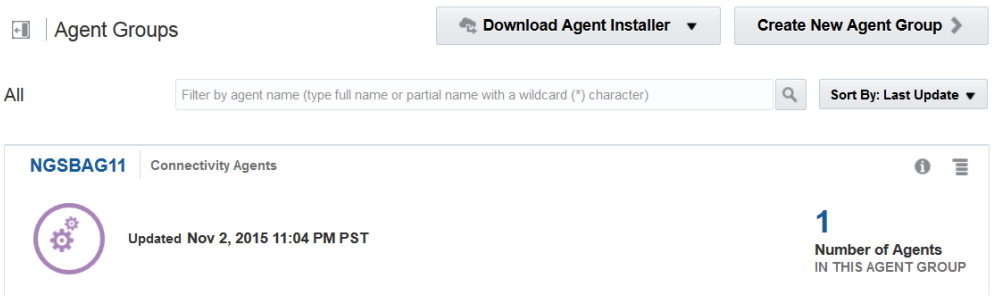
Installation

To download and run the on-premises agent installer:

> **Note:** It is recommended that you retain a copy of the `cloud-connectivity-agent-installer.bsx` agent installer file after completing installation.

1. Log in to Oracle Integration Cloud Service.
2. In the Integration Cloud Service toolbar, click **Designer**.
3. On the Designer Portal, click **Agents**.

   The Agent Groups page is displayed. This page shows any current agent groups and on-premises agents associated with the agent groups.



4. Click **Download Agent Installer**.
5. Select **Connectivity Agent**.

   This selection enables you to integrate Oracle Integration Cloud Service with on-premises environments.
6. Select **Save File** when prompted to save the file to a directory location on your on-premises host.

7. Unzip the downloaded file.
8. Locate the `cloud-connectivity-agent-installer.bsx` agent installer file.
9. Change the file permissions to be executable.

   ```
   chmod +x cloud-connectivity-agent-installer.bsx
   ```

10. Execute the file with the following properties specified. Do *not* install the on-premises agent in a directory path that includes `/tmp`. Note the use of the dash (`-`) when specifying the property value. The output of this command is displayed on-screen. For troubleshooting purposes, you can also redirect the output to an output file.

    ```
    ./cloud-connectivity-agent-installer.bsx –h=https://ICS_host.us.oracle.com:port –
    u=weblogic –p=my_password –ad=agent_group_identifier
    ```

    where:

Parameter	Status	Description	Additional Notes

`-h`	Required	Specifies the Oracle Integration Cloud Service hostname and port. As an example, when installing from your Oracle Integration Cloud Service POD, the host and port you specify are typically the Oracle Integration Cloud Service URL (for example, `https://icsapps.integ.dc4.c1234.oraclecorp.com:443`).	Note the following issues:
			• Do *not* specify `/ics` after the port number.
			• If you forget to specify a port, you receive the following error:
			`Numberformat Exception`
			• If you specify port `80` with the `https` protocol, you can receive the following error:
			`Outbound ProxyHost and ProxyPort as not provided certPath --- /home/oracle/host/downloads/ics_conn_agent_installer_151110.1158/keystore.jks java.net.SocketTimeoutException: Read timed out at java.net.SocketInputStream.socketRead0(Native Method) at java.net.SocketInputStream.socketRead(SocketInputStream.java:116) at java.net.SocketInputStream.read(SocketInputStream.java:170) at java.net.SocketInputStream.read(SocketInputStream.java:141) at sun.security.ssl.InputRecord.readFully(InputRecord.java:465) at sun.security.ssl.InputRecord.read(InputRecord.java:503) . . . . . .`

`-u`	Required	Specifies the Oracle Integration Cloud Service user name.	

Parameter	Status	Description	Additional Notes
`-p`	Required	Specifies the Oracle Integration Cloud Service password.	**Note**: • Some special characters must be properly escaped with a backslash (\) character. For example, if your password includes an exclamation character, you must enter the backslash character before the exclamation character (\!). • If the password contains a dollar sign ($) character (for example, `W$lcome11`), the complete password value must be contained in single quotes (for example, `-p='W$lcome11'`). Otherwise, you receive an `Undefined variable` error when you run the agent installer command.
`-ad`	Required	Specifies the agent group identifier that was generated in the **Identifier** field when you created the agent group in the New Agent Group - Information dialog in Creating an Agent Group. You must create the agent group before you can install the on-premises agent.	
`-ph`	Optional	Specifies the outbound proxy hostname.	If your on-premises host includes a proxy server, you must specify this property. The agent can work with any proxy in the DMZ.
`-pp`	Optional	Specifies the outbound proxy hostname port.	If your on-premises host includes a proxy server, you must specify this property.

During installation, the following tasks are performed:

- All on-premises adapters are registered.
- Oracle WebLogic Server and a Java database are installed.
- The Oracle WebLogic Server and JRF domain is created.
- The on-premises agent is deployed.

Once installation is complete, an agent instance is created for interacting with Oracle Integration Cloud Service. You can verify that the agent instance was created by going to the Agent Groups page and noting that the agent count was increased by one. If you click the number, details about the agent are displayed. For more information, see Viewing and Editing Agent Groups.

Troubleshooting Issues

- If there is a Derby database (DB) already in use on the host on which the agent is being installed, you receive the following error:

```
2016-01-27 20:48:51,714 FINE  [47]
com.oracle.cie.domain.jdbc.DatasourceXBeanAspectHelper -
prop str:
user=dummy;portNumber=1527;databaseName=dummy;create=true;serverName=localhost
url: jdbc:derby://localhost:1527/dummy;ServerName=localhost;databaseName=dummy
2016-01-27 20:48:51,714 FINE  [47]
com.oracle.cie.domain.jdbc.DatasourceXBeanAspectHelper -
Selected DB ID/CAT: DerbyDerby's Driver (Type 4) Versions:Any 2016-01-27
20:48:51,714 FINE
[47] com.oracle.cie.domain.jdbc.DatasourceXBeanAspectHelper - Selected DB
vendor: Derby
2016-01-27 20:48:51,714 FINE  [47]
com.oracle.cie.domain.jdbc.DatasourceXBeanAspectHelper -
adding normal datasource: opss-data-source 2016-01-27 20:48:51,714 FINE  [47]
com.oracle.cie.domain.jdbc.DatasourceXBeanAspectHelper - datasource: opss-data-
source component
name: OPSS Schema
. . .
. . .
```

  Perform the following steps:

  1. Check if the Derby database is running.

     ```
     ps -ef | grep "derby"
     ```

  2. If any processes are displayed, run the following command to terminate them:

     ```
     ps -ef | grep "derby" | awk '{print $2}' | xargs kill -9
     ```

- At the end of on-premises agent installation, the following message is displayed:

```
Agent installation completed Successfully.
Agent Instance creation successful
```

  After this message is displayed, you may receive the following error message that can be safely ignored. On-premises agent installation was successful. This issue occurs during KSS store creation. If KSS store creation is required, you can manually perform this task at a later time.

```
javax.naming.AuthenticationException: User: [...], failed
to be authenticated. [Root exception is java.lang.SecurityException: User:
..., failed to be authenticated.]
Problem invoking WLST - Traceback (innermost last):
  File

"/private/downloads/ICS_1.6.15_Agent/selfextract.TKERY8/agentInstaller/./scrip
ts/import_cert_kss.py", line 45, in ?
File "<iostream>" line 19, in connect
File "<iostream>" line 552, in raiseWLSTException
WLSTException WLSTException: Error occurred while performing connect : User:
..., failed to be authenticated. : User:
..., failed to be authenticated.
Use dumpStack() to view the full stacktrace :
```

You are now ready to create an adapter connection in Oracle Integration Cloud Service that uses the on-premises agent. See ..

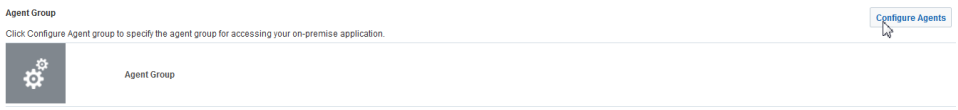
Creating a Connection with an Agent Group

After you have installed the on-premises agent, you can create a connection that uses the agent group and its associated on-premises agent.

To create a connection with an agent group:

1. In the Integration Cloud Service toolbar, click **Designer**.
2. On the Designer Portal, click **Connections**.
3. Click **Create New Connection**.

   For specific details about connection creation, see Creating Connections.
4. Select the adapter to configure as a target endpoint. The following adapters are supported:
   - MySQL
   - Oracle Database
   - Oracle E-Business Suite
   - Oracle Siebel
   - REST
   - SAP
   - SOAP
5. Configure the connection properties and security.
6. In the **Agent Group** section, click **Configure Agents** to select the agent group to associate with the adapter. This enables you to access your on-premises applications.



7. Select the agent group to use with this adapter, and click **Use**.
8. Click **Test**. This test executes the ping command on the on-premises instance when the connection is associated with an agent.
9. Click **Save**, then click **Exit Connection**.
10. Create an integration in which you drag the adapter to the target side for configuration. Only the target side is supported. For this example, an Oracle Siebel adapter is configured as the target connection in the integration.

11. Activate the integration, as described in Activating an Integration.

12. Invoke the integration.

> **Note:** If you receive the following error, a connection time out has occurred. The request may be slow, in which case the request must be executed again. You can also view the agent logs to see what may be causing the request to not be processed.
>
> ```
> CASDK-0005 A connector specific exception was raised by the application.
> oracle.cloud.cpi.omcs.api.CpiOmcsException.
> No response received within response time out window of 60000.
> ```

Viewing and Editing Agent Groups

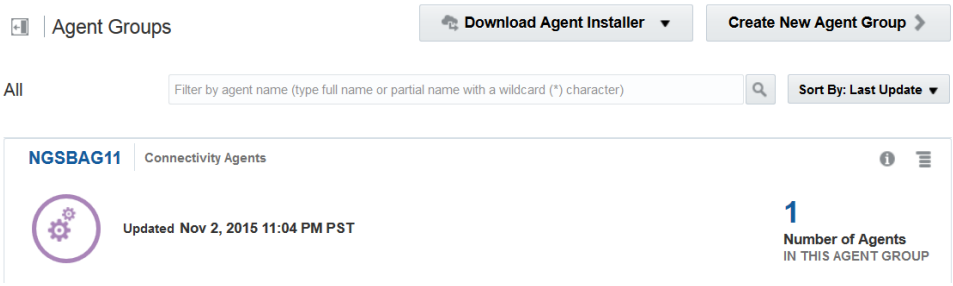
You can view details about agent groups and their associated on-premises agents and edit the agent group name.

To view agent group details.

1. In the Integration Cloud Service toolbar, click **Designer**.

2. On the Designer Portal, click **Agents**.

3. Click the number above the **Number of Agents** label to show the associated on-premises agent.



4. Click **Done** when complete.

5. If you want to edit the agent group name, click the menu icon at the far right.

6. Select **Edit**.
7. Update the name, then click **OK**. Changing the agent group name does not impact the association with the on-premises agent.

Deleting an Agent Group

You can delete an agent group that is not currently associated with a running on-premises agent instance. Deleting an agent group removes the Oracle Messaging Cloud Service queues and topics.

To delete an agent group:

Note: Before deleting an agent group, ensure that you stop the agent on the on-premises host with either of the following commands:

- Stop the Oracle WebLogic Server, which also stops the agent:

  ```
  ./stopWebLogic.sh
  ```

- Kill the running agent at the operating system command prompt:

  ```
  kill -e agent_PID_number
  ```

---

1. In the Integration Cloud Service toolbar, click **Designer**.

2. On the Designer Portal, click **Agents**.
3. Find the agent group to delete.
4. From the menu icon at the far right, select **Delete**.
5. Select **Yes** when prompted to confirm.

Updating Property Values

If you need to change values for properties such as the external Oracle Messaging Cloud Service (OMCS) URI, edit the `CpiAgent.properties` file on your on-premises host.

To update property values:

1. Stop the Oracle WebLogic Server by executing the following command in the `domains/agent-domain/bin` directory. This command also shuts down the on-premises agent.

   ```
   ./stopWebLogic.sh
   ```

2. Open the `agent-domain/agent/config/CpiAgent.properties` file.
3. Change property values, as necessary.

4. Save your changes and exit the file.
5. Restart the Oracle WebLogic Server, which also restarts the agent.

   ```
   ./startWebLogic.sh
   ```

   To troubleshoot errors, you can also redirect the output of `startWeblogic.sh` to a file. For example, `./startWeblogic.sh >& wls.out`.

Scheduling Integration Runs

Oracle Integration Cloud Service scheduling enables you to schedule the running of integrations configured with trigger and invoke FTP adapter connections. You can schedule these integration runs to copy files at a date and time of your choosing. You can also define the frequency of the integration. When you create an integration with trigger and invoke FTP adapter connections, a schedule icon is displayed with the integration entry on the Integrations page.

Topics

For more information about the FTP Adapter see *Using the FTP Adapter*.

Creating an Integration Schedule

You can create a schedule for running integrations in which trigger FTP adapter and invoke FTP adapter connections are defined.

To create an integration schedule:

1. Go to the Integration page.
2. Find the integration in which trigger and invoke FTP adapter connections have been defined.

   These integrations are identified by an **Add Schedule** icon. There are several ways in which to create an integration.

3. If you want to first activate the integration and then create the schedule:

   a. Click **Activate**.

   b. Click **Take me to schedule definition after activation**, then click **Activate**.

4. If you want to first create a schedule for an integration that you activate later:

   a. Click the **Add Schedule** icon or select **Schedule** from the dropdown menu at the far right.

5. Click the **Edit** icon.
6. In the **Schedule Name** field, enter a name for the schedule.
7. In the **Description** field, enter a description for the schedule.
8. When complete, click the **green checkmark** icon to submit your changes. This action does not save the changes.
9. In the **Frequency** section, click the icon to display a dropdown list for selecting the frequency with which to run the integration. As you define one frequency, you can specify additional values by clicking the icon to the right of the **Frequency** section.
    - **Only Once**: This is the default selection. This selection clears all settings except for the **From** field.
    - **Hours and Minutes**: Specify the hours and minutes at which to run the integration.
    - **Days**: Specify the days on which to run the integration.
    - **Weeks**: Specify the weeks during which to run the integration.
    - **Months**: Specify the months during which to run the integration.
10. Click the **green checkmark** icon for each frequency type that you specify.
11. Click **Save** to validate your frequency settings.

    Your selections are validated. If there are any errors, a validation message is displayed in the upper left corner that describes how to resolve the errors.
12. In the **Effective** section, click the time value to the right of **From**.

    A menu is displayed for defining the start date of the schedule.
13. If you want to start the integration run when the schedule is activated:
    a. Click **When schedule starts**.
14. If you want to explicitly set an integration run start date:
    a. Select **Modify start date**.
    b. Click the **Calendar** icon to select the month, year, and day and the hour, minute, and second at which to start the integration run.
    c. Click **OK**.
15. In the **Expire** section, click the link to the right.

A menu is displayed for defining the expiration date.

16. If you want the schedule run to never expire:
    a. Select **Never (repeat indefinitely)**.
17. If you want the integration run to have a fixed expiration date:
    a. Select **Choose expiry date**.
    b. Click the **Calendar** icon to select the month, year, and day and the hour, minute, and second at which to end the integration run.
    c. Click **OK**.
18. Click **Save**.

    If successful, a message is displayed in the upper left corner.

    `Schedule Run: name saved successfully..`
19. Click **Exit Scheduler**.

    The Schedule and Future Runs page is displayed and the Monitoring tab is highlighted in the upper right corner.
20. If you have already started the integration, click **Submit Now** to run the integration or **Start Schedule** to activate the integration schedule.
21. If you have not yet started the integration, return to the Integrations page and click **Activate**.
22. Click the **Add Schedule** icon or select **Schedule** from the dropdown menu at the far right to return to the Schedule and Future Runs page.
23. Click **Submit Now** to run the integration or **Start Schedule** to activate the integration schedule.

Starting and Pausing an Integration Schedule

After you define a schedule, you must activate it. You can also pause (deactivate) a schedule, as needed.

To start and pause an integration schedule:

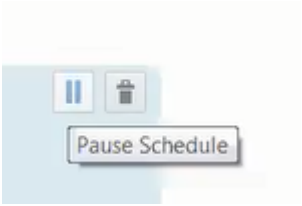
1. Go to the Integration page.
2. Find the integration on which the scheduled run is defined.
3. Click the **Add Schedule** icon or select **Schedule** from the menu at the right. Once a schedule is defined, the tooltip for this icon is changed to **Schedule Defined**.

   The Schedule and Future Runs page is displayed.
4. Click the **Start Schedule** button.

   The following message is displayed at the top of the page: `Schedule is now active.`

Details about the schedule including the frequency and any expiration date are displayed. The **Start Schedule** button is changed to **Pause Schedule**.

5. If you want to pause the schedule run, click **Pause Schedule** at the far right.



6. If you want to resume the schedule run, click **Resume Schedule**. This toggles the button name to **Pause Schedule**.

Viewing Past and Present Integration Runs

You can view the status of past and present scheduled integration runs.

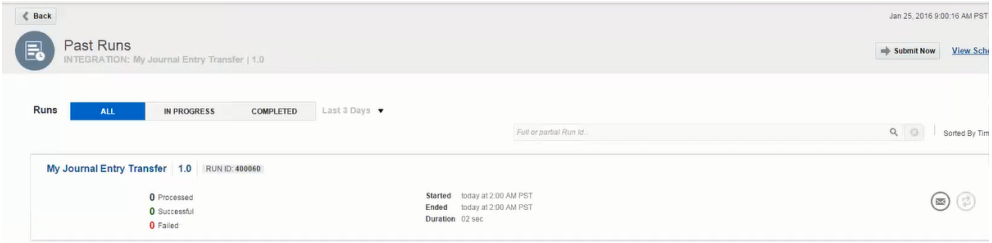
To view past and present integration runs:

1. Go to the Integration page.
2. Find the integration on which the scheduled run is defined.
3. Click the **Add Schedule** icon or select **Schedule** from the menu at the right.

   The Schedule and Future Runs page is displayed.

4. Click **View Past Runs**.

   The Past Runs page is displayed. By default, all integration runs are displayed.



5. Filter the display of integrations:
   a. Click **In Progress** to display all integration runs currently in progress.
   b. Click **Completed** to display all completed integration runs.
   c. Click the dropdown list to filter the display of runs by **1** hour, **6** hours, **1** day, **2** days, or **3** days.
6. If a schedule run has failed (for example, the target FTP server did not have the correct write permissions), click the **Resubmit** icon at the far right to resubmit the schedule run.

This action creates a **RESUBMITTED RUN ID:** ***number*** message to the right of the schedule name and original run ID. If you resubmit again, you end up with the following messages to the right of the schedule name:

- **RUN ID:** ***number***: The run ID for the first resubmission, which failed.
- **RESUBMITTED RUN ID:** ***number***: The run ID for the latest submission.
- **ORIGINAL RUN ID:** ***number***: The run ID for the initial submission.

7. Click the **RESUBMITTED RUN ID:** ***number*** message link to go to the Tracking page.
8. Click the file name of the instance.

   A graphical view of the integration flow is displayed.

   For example, if the resubmission resulted in a failure, details are displayed. For this example, the write portion of the integration failed because of a permissions issue.

9. Select **Actions** > **Audit Trail** to view specific details about the error, such as the target directory not being defined with write permissions.

Viewing Future Runs

You can view the status of future scheduled integration runs.

To view future runs:

1. Go to the Integration page.
2. Find the integration on which the scheduled run is defined.
3. Click the **Add Schedule** icon or select **Schedule** from the menu at the right.

   The Schedule and Future Runs page is displayed.
4. Click **View Past Runs**.
5. Click **View Schedule**.

   Details about the future runs are displayed.

Editing an Integration Schedule

You can edit a schedule for an integration run.

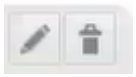
To edit an integration schedule:

1. Go to the Integration page.
2. Find the integration on which the scheduled run is defined.

3. Click the **Add Schedule** icon or select **Schedule** from the menu at the right.

   The Schedule and Future Runs page is displayed.

4. On the far right, click the **Edit** icon.



5. Edit the schedule. For information about the fields you can edit, see Creating an Integration Schedule.

6. Click **Save**.

Deleting an Integration Schedule

You can delete a schedule for an integration run.

To delete an integration schedule:

1. Go to the Integration page.
2. Find the integration on which the scheduler run is defined.
3. Click the **Add Schedule** icon or select **Schedule** from the menu at the right.

   The Schedule and Future Runs page is displayed.

4. Find the schedule in the list that you want to delete.
5. On the far right, click the **Delete** icon.

6. Click **Yes** when prompted to confirm.

Monitoring Integration Runs

You can stop and restart scheduled runs for an integration from the Monitoring page.

To monitor integration runs:

1. On the Integration Cloud Service toolbar, click **Monitoring**.
2. In the navigator, click **Integrations**.

   For integrations that include source and target FTP adapter connections, the calendar icon is displayed. This page shows only active integrations. Integrations can be of any type (that is, with or without the FTP Adapter).

3. If you want to restart scheduled runs of an integration, click **Resume Schedule**.
4. If you want to stop scheduled runs of an integration, click **Pause Schedule**.

4

Administering Integration Cloud Service

The Integration Cloud Service dashboard provides you with the information and tools to monitor and manage your integrations in the runtime environment. Administration tasks can also include working outside the dashboard, such as when you activate or deactivate integrations.

Topics

Monitoring Integration Cloud Services

Use the Integration Cloud Service dashboard to see how your integrations are performing. The dashboard provides multiple views for you to check your running services.

Topics

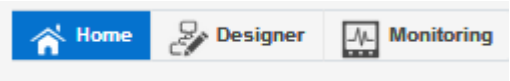
Viewing the Dashboard

Use the Integration Cloud Service dashboard to view information about messages.

You can view information about how your integrations are performing. The main page of the integration shows a snapshot of a state of all running integrations.

To view the dashboard:

1. Access the dashboard through one of the following methods:
   a. On the Integration Cloud Service toolbar, click **Monitoring**.



b. On the Integration Cloud Service Home page, click the **Monitor** diagram.

The dashboard is displayed and shows a snapshot of the current state of your system and each running integration.

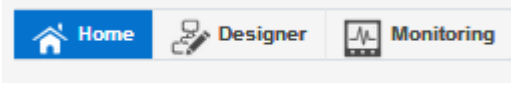
Note: When one of two managed nodes is down, the data on the Dashboard page is removed. A value of **0** is displayed for the fields, including **Total Messages Processed**, **Errors**, and **Success Rate**.

Monitoring Integrations

On the Integration Cloud Service dashboard, you can see how your running integrations are processing messages, such as how many messages have been received and processed, how many successful messages and errors have occurred, and the overall success rate.

To monitor integrations:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



   The dashboard appears, and shows processing information for all your running integrations.

2. In the navigator, click **Integrations**.

Monitoring Activity Streams and Log Files

The Integration Cloud Service dashboard provides a view into the activity stream for your running integrations, such as details about successful responses received from the target application, successful responses sent to the source application, and target application invocations.

To monitor activity streams and log files:

1. On the Integration Cloud Service toolbar, click **Monitoring**.

   The dashboard appears, and shows processing information for all your running integrations.

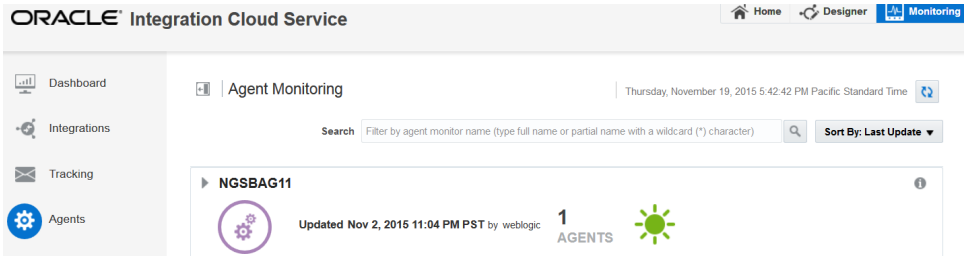
2. Click the **Activity Stream** tab.
3. Click **Refresh Logs** to refresh the view to display the latest activities.
4. Click **Download Logs** to download the `ics-logs.zip` file for offline analysis.

Monitoring Agents

You can monitor the agent groups and their associated on-premises agents in Oracle Integration Cloud Service.

To monitor an agent:

1. In the Integration Cloud Service toolbar, click **Monitoring**.
2. In the navigation panel on the left, click **Agents**.
3. The Agent Monitoring page shows details such as the time at which the agent was last updated and the on-premises agents associated with the agent groups.



4. Click the number above **AGENTS** or click the agent group name to see the on-premises agent associated with this group.
5. At the far right, click the information icon to display details about the agent group such as creation date and the last startup date of the on-premises agent.
6. Click **Sort By** to sort by agent name or the date at which the on-premises agent was last started.

Managing Integrations

When you are ready for your integration to go live, you must activate the integration in Integration Cloud Service. You can also deactivate a running activation if you must make changes to it or if it is no longer needed.

Topics

Activating an Integration

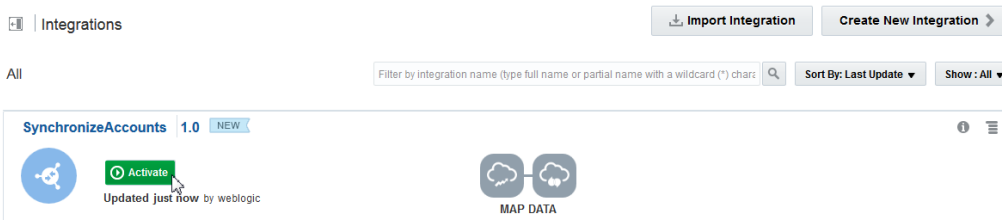
Once you create an integration and the progress indicator shows 100 percent, you can activate that integration to the runtime environment. An integration shows as 100%

and is eligible for activation after you have specified the source connection, the target connection, the data mappings, and the tracking fields.

To activate an integration:

> **Note:** If you activate a new version of an existing integration, tracking instances or logs of the old version are not deleted. However, related artifacts are deleted and redeployment is performed on the back end. Monitoring data is also removed.

1. If you are not already on the Integrations page, click **Designer** on the toolbar, then click **Integrations**.
2. In the Integrations list, locate the integration you want to activate.
3. Click **Activate**.



.

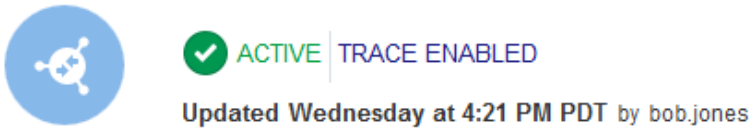
The Confirmation dialog is displayed.

4. If you want to enable detailed tracing, click the **Enable detailed tracing** checkbox.

   When selected, detailed logging information about messages processed by this integration flow at runtime is collected. This can aid in troubleshooting issues. However, this may also impact performance. To disable tracing, you must deactivate the integration, then reactivate it without selecting the **Enable detailed tracing** checkbox.

5. Click **Activate**.

   The status of the integration changes to **ACTIVE** in the list. If you selected to enable tracing, the words **TRACE ENABLED** are displayed next to **ACTIVE**.



   To access the detailing trace logging information:

   a. Click the **Monitoring** tab.
   b. Click the **Activity Stream** tab.
   c. Click **Download Logs**.

6. View active integrations by clicking the integration name or selecting **View** from the menu at the far right of the integration. The active integration is displayed with a message saying **Viewing**.

Note the following details about read-only mode:

- No **Save** button and **Actions** button are displayed.
- There is no Connections Palette for adding adapters.
- You can click through multiple parts of the integration to view configuration details, such viewing the business identifiers under the **Configuration** tab, viewing the source-to-target and target-to-source mappings in the mapper, and viewing the configurations on the pages of the connection wizards, but you cannot modify anything.

Deactivating an Integration

You can deactivate an integration to stop it from processing any new messages. If you want to modify an active integration, you need to deactivate it first.

Deactivation is equivalent to undeployment of a project, which means all existing history, monitoring, and runtime data are lost. Integration Cloud Service does not support the notion of starting and stopping projects. With asynchronous patterns, the queue for the deactivated project is deleted and all messages associated with this queue are also deleted. Therefore, if there are pending requests unprocessed, they are lost after deactivation. The previous version is deactivated and all existing history, monitoring, and runtime data is lost.

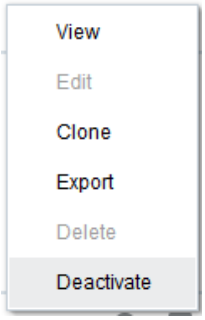
1. In the Integration Cloud Service toolbar, click **Designer**.

   The Designer Portal appears.

2. Click **Integrations**.
3. On the expanded Integrations page, find the integration you want to deactivate.

   To view only active integrations, select **Active** in the Integrations list. You can also filter by integration name or integration type (prebuilt, custom, or developed) to narrow down the list.

4. In the row containing the integration you want to deactivate, click the vertical bars icon to the far right and then click **Deactivate**.



5. Click **Yes** on the dialog that appears.

Modifying an Integration

You can modify an existing integration, including changing a source or target connection, reconfiguring the connection, and updating the data mapping. Changes to the source or target can cause changes to the existing mappings.

If the integration you want to modify is active, deactivate it first. See Deactivating an Integration for instructions.

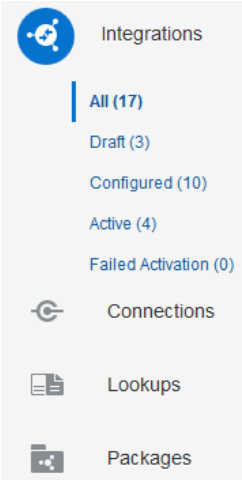
To modify an integration:

1. In the Integration Cloud Service toolbar, click **Designer**.

   The Designer Portal appears.

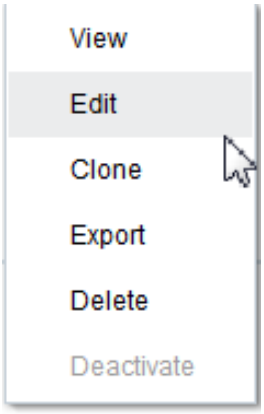
2. Click **Integrations**.

3. On the Integrations page, find the integration you want to modify. You can filter the display of integrations by their current status on the left side of the page.

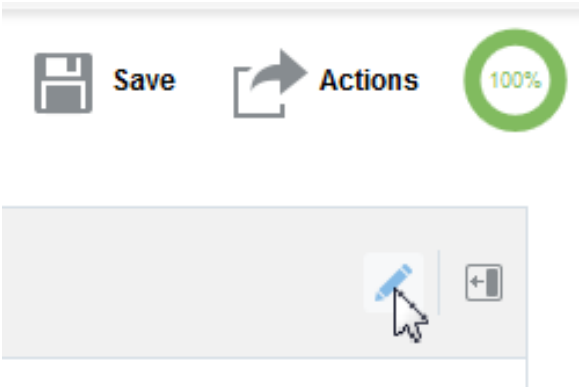


   You can also search by entering a partial or complete integration name in the **Search** field or filter integrations by selecting an option from the **Filter By** list. From this list, you can filter by **Type** (**Custom**, **Developed**, and **Prebuilt**) or **Pattern** (**Map My Data**, **Publish to Integration Cloud Service**, **Subscribe To Integration Cloud Service**, **File Transfer**, **Orchestration**, and others). Search or filter criteria are displayed in the blue banner above the returned list of integrations. To remove search or filter criteria, click the **x** icon in the blue banner or the **x** icon to the right of the **Filter By** list.

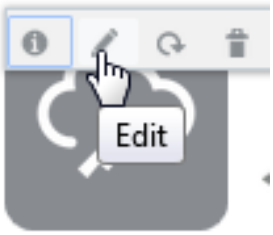
4. On the row that contains the integration you want to change, click the integration name or click the vertical bars icon on the far right and select **Edit**.



5. To modify the name, package, or description, click the **Edit** icon on the right side.

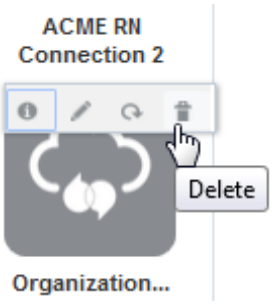


6. To modify the source or target configuration, click the connection on the canvas and click **Edit** on the menu that is displayed.



7. Modify any of the open fields in the wizard that appears. See Connection Configuration Reference for instructions.

8. To assign a new connection as the source or target, click the connection to delete, then click **Delete** on the menu that is displayed.



9. Click the **Show Palette** icon on the right side and drag the new adapter from the Connections or Technologies panel on the right to the connection on the canvas that you want to replace. Configure the new connection. See Connection Configuration Reference for instructions.
10. To modify a data mapping, click the appropriate map icon in the middle of the integration and update the mappings. See Mapping Data of *Using the Oracle Mapper*.
11. When you are done making changes, click **Save** and then click **Exit Integration**.

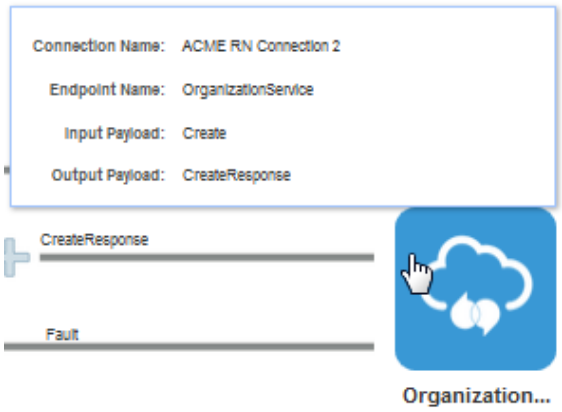
Viewing the Trigger, Invoke, and Enrichment Details of an Integration

You can view the details associated with trigger, invoke, and enrichment endpoints of an integration.

To view the trigger, invoke, and enrichment details of an integration:

1. In the Integration Cloud Service toolbar, click **Designer**.

   The Designer Portal appears.
2. Click **Integrations**.
3. On the Integrations page, find the integration you want to view. You can filter the display of integrations by their current status on the left side of the page.
4. Click the trigger, invoke, or (if configured) enrichment endpoint of the integration.
5. Click the **Details** icon.
6. View the connection name, endpoint name, input payload, and output payload of the integration.



Cloning an Integration

Cloning an integration creates a new copy with identical connections and data mappings. You give the clone a new name, identifier, version number, and package name, but the remaining configuration is the same. You can reconfigure the clone after you create it.

To clone an integration:

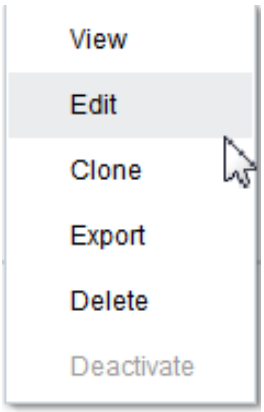
1. In the Integration Cloud Service toolbar, click **Designer**.

   The Designer Portal appears.

2. Click **Integrations**.

3. On the Integrations page, find the integration you want to clone. You can filter the display of integrations by their current status on the left side of the page.

4. In the row containing the integration you want to clone, click the vertical bars icon on the far right and then click **Clone**.



5. In the dialog that appears, enter a name, unique identifier, version number, package name, and an optional description.

You can include English alphabetic characters, numbers, underscores, and dashes in the identifier. Enter the version using numbers only in this format: `xx.xx.xxxx`.

6. Click **Clone**.
7. You can modify the clone in any of the ways described in Modifying an Integration.

Deleting an Integration

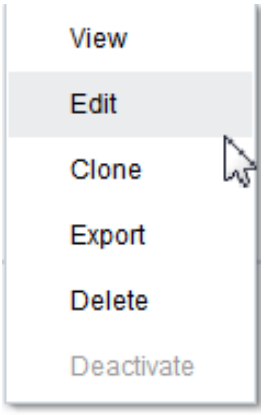
You can delete an integration that is no longer needed.

Make sure the integration you want to delete is not active. To deactivate the integration, see Deactivating an Integration.

To delete an integration:

1. In the Integration Cloud Service toolbar, click **Designer**.

   The Designer Portal appears.
2. Click **Integrations**.
3. On the Integrations page, find the integration you want to delete. You can filter the display of integrations by their current status on the left side of the page.
4. In the row containing the integration you want to delete, click the vertical bars icon on the far right and then click **Delete**.



5. Click **Yes** on the dialog that appears.

Reactivating Integrations After Instance Upgrade to View the Latest Business Identifier Tracking Behavior

After your Integration Cloud Service instance is upgraded to a newer version, you must reactivate your integrations to view the latest and correct business identifier behavior in the Tracking page.

Filtering the Display of Integrations By Type

You can filter the display of integrations by their type (custom, user-developed, or prebuilt).

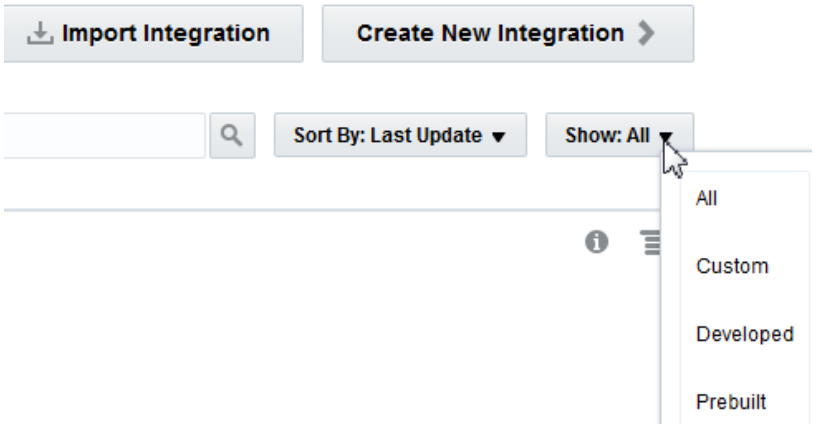
To filter the display of integrations by type:

1. In the Integration Cloud Service toolbar, click **Designer**.

   The Designer Portal appears.

2. Click **Integrations**.

3. Select the **Show** list to filter the display of integrations.



   You can filter integrations by the following types:

   - **All**: Displays all integrations, regardless of their type.
   - **Custom**: Displays prebuilt integrations imported from the Oracle Marketplace that have been customized.
   - **Developed**: Displays integrations created completely from scratch.
   - **Prebuilt** Displays prebuilt integrations imported from the Oracle Marketplace.

4. Select the **Sort By** list to filter the display of integrations by the last update or name.

Changing the Time Zone

You can change the time zone that is displayed in Integration Cloud Service.

To change the time zone:

1. In the upper right corner, click the ***username*** dropdown list, then select **Preferences**.

2. From the **Time Zone Settings** list, select the time zone you want to use.

3. Click **Save**.

4. Go to the Integrations page and note that the time zone is changed in the message below the status of the integration and inside the information icon at the far right.

   The time zone change is also shown in other parts of Integration Cloud Service in which the time is displayed (for example, on the Tracking page).

Managing Errors

You can manage errors from the Errors pages in Integration Cloud Service at the integration level, connection level, or specific integration instance level.

Topics

For more information about error management, see About Error Management.

Managing Errors by Integration

You can manage errors by the integration in which they occurred. Management tasks consist of viewing the total error count for an integration over a specific time period, discarding (removing) all errors for an integration, and viewing more specific error message details by clicking the integration name or the total error count.

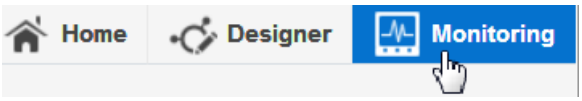
Topics

Viewing Errors By Integration Name Over a Specific Time Period

You can view errors by integration name over a specific time period.

To view errors by integration name or the total number of integration errors that have occurred over a specific time period:

1. On the Integration Cloud Service toolbar, click **Monitoring**.

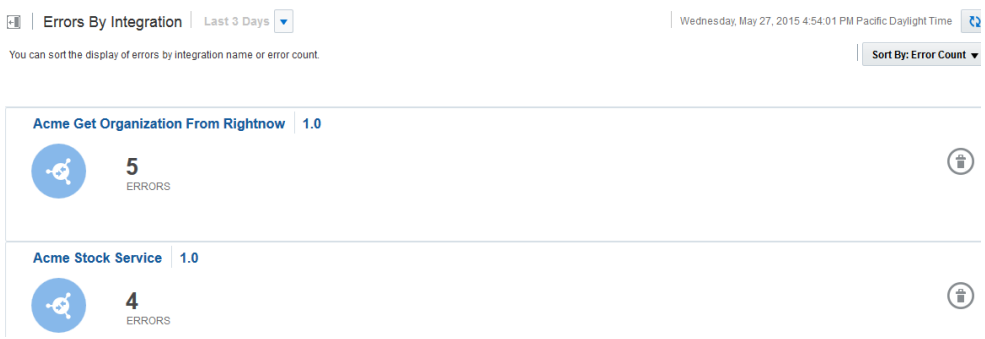


2. In the navigation pane, click **Errors**.

   The Errors By Integration page is displayed. Any integration errors that are displayed by default are those that occurred within the selected time period.

3. From the menu, select the time period for which to display integration errors.

Any errors that occurred during the selected time period are displayed immediately below the menu. Error details consist of the integration name and the total number of errors that occurred in that integration.



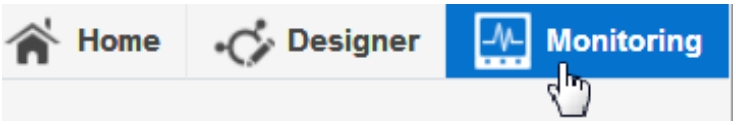
4. Click the integration name or total error count to access the Error Message Details page. This page provides information about the business identifiers defined in the integration, the instance identifier of the integration, the location of the error, the time at which the error occurred, the audit trail, the specific error message, and other information. For more information about the Error Message Details page, see Managing Errors by Integration Instance.

Discarding Errors by Integration Name

You can discard errors based on the integration in which they occurred. A discarded error message is removed from the Errors By Integration page and can be seen in a discarded state on the Tracking page. You cannot perform any further operations on a discarded message, including recovery. After a certain time period, the error message is permanently deleted from the server.

To discard errors by integration name:

1. On the Integration Cloud Service toolbar, click **Monitoring**.

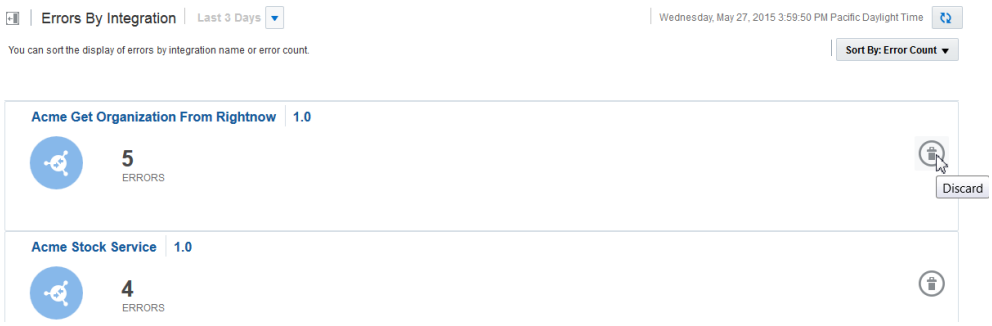


2. In the navigation pane, click **Errors**.

   The Errors By Integration page is displayed.

3. Perform any necessary error filtering by following the instructions in Viewing Errors By Integration Name Over a Specific Time Period.

4. For the integration in which to discard errors, click the **Discard** button at the far right. For synchronous integrations, the **Discard** button is not visible.



5. Click **Yes** when prompted to confirm. This action discards all error messages in all instances of that integration.

Managing Errors by Connection

You can manage errors by the source or target connection in which they occurred. Management tasks consist of viewing the total error count for a connection over a specific time period, discarding all errors for a connection, and viewing more specific error message details by clicking the connection name or the total error count.

Topics

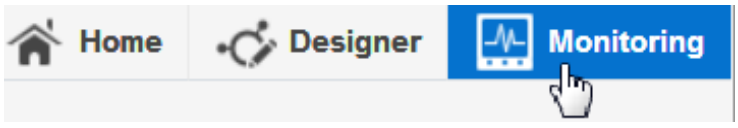
- Viewing Errors By Connection Name or the Total Error Count Over a Specific Time Period
- Discarding Errors by Connection Name

Viewing Errors By Connection Name or the Total Error Count Over a Specific Time Period

You can view errors by connection name or the total number of connection errors that have occurred during a specific time period.

To view errors by connection name or the total number of connection errors that have occurred over a specific time period:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**, then click **Errors By Connection**.

   The Errors By Connection page is displayed. Any connection errors that are displayed by default are those that occurred within the selected time period.

3. From the menu, select the time period for which to display connection errors.

   Any errors that occurred during the specified time period are displayed immediately below the menu. Error details consist of the connection name and the total number of errors that occurred in that connection.

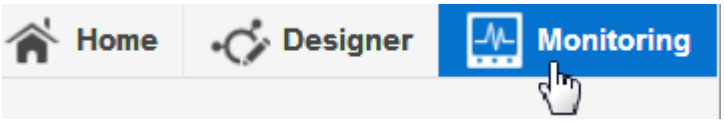
4. Click the connection name or total error count to access the Error Message Details page. This page provides information about the business identifiers defined in the integration, the instance identifier of the integration, the location of the error, the time at which the error occurred, the audit trail, the specific error message, and other information. For more information about the Error Message Details page, see Managing Errors by Integration Instance.

Discarding Errors by Connection Name

You can discard errors based on the connection in which they occurred. A discarded error message is removed from the Errors By Connection page and can be seen in a discarded state on the Tracking page. You cannot perform any further operations on a discarded message, including recovery. After a certain time period, the error message is permanently deleted from the server.

To discard errors by connection name:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**, then click **Errors By Connection**.

   The Errors By Connection page is displayed.

3. Perform any necessary error filtering by following the instructions in Viewing Errors By Connection Name or the Total Error Count Over a Specific Time Period.

4. For the connection in which to delete errors, click the **Discard** button at the far right.

5. Click **Yes** when prompted to confirm. This action discards all error messages in that connection.

Managing Errors by Integration Instance

You can manage errors by the specific integration instance in which they occurred.

Management tasks consist of viewing the business identifiers defined for the integration, the instance identifier of the integration, the location of the error, the time at which the error occurred, the audit trail, and the specific error message. You can also discard failed messages.

Topics

- Viewing Errors by Integration Name, Instance Identifier, Location, or Time of Occurrence Over a Specific Time Period
- Viewing the Integration Instance in Which Errors Occurred
- Discarding Errors by Integration Instance
- Viewing Specific Error Details
- Viewing the Audit Trail of a Failed Integration Instance

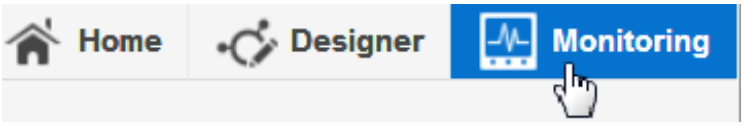
- Viewing Business Identifiers in Failed Integration Instances
- Viewing the Message Payload of a Failed Integration Instance

Viewing Errors by Integration Name, Instance Identifier, Location, or Time of Occurrence Over a Specific Time Period

You can display errors by integration name, instance identifier, error location, or the time of occurrence over a specific time period. This provides you with a more granular view of integration failure details.

To display errors by integration name, instance identifier, error location, or the time of occurrence over a specific time period:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**, then click **Error Message Details**.

   The Error Message Details page is displayed. Any errors that are displayed by default are those that occurred within the selected time period.

3. From the menu, select the time period for which to display error information.

   Any errors that occurred during the specified time period are displayed immediately below the menu.

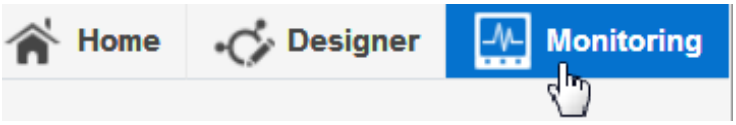
4. Click the primary business identifier name to access a graphical view of the integration instance. This page provides information about the business identifiers defined for an instance, the instance identifier of the integration, the location of the error, the time at which the error occurred, the audit trail, a button for discarding the error, and other information.

Viewing the Integration Instance in Which Errors Occurred

You can view the integration instance in which errors occurred. From the integration instance page, you can perform multiple tasks, including viewing business identifiers in the integration, viewing the audit trail, viewing error messages, and discarding errors.

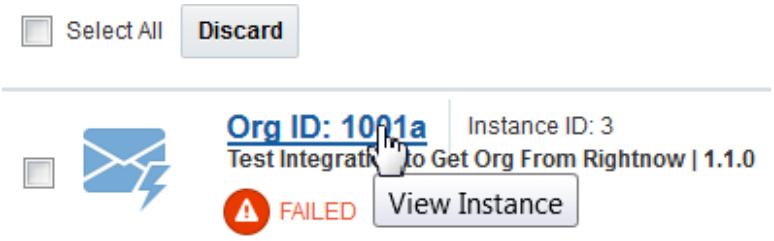
To view the integration instance in which errors occurred:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**, then click **Error Message Details**.

3. Click the business identifier included in the failed integration instance.



The integration is displayed. The direction in which the error occurred is indicated by the color red. The primary business identifier and instance identifier of the integration are displayed above the integration.

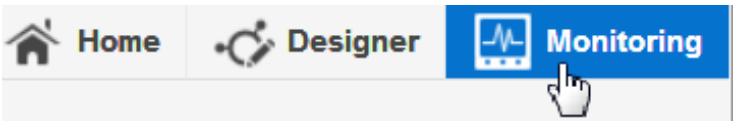
You can perform multiple tasks on this page, including viewing all business identifiers and values in the integration, viewing the audit trail, viewing errors, viewing the payload, and discarding errors.

Discarding Errors by Integration Instance

You can discard errors by integration instance in several locations. A discarded error message is removed from the Errors Message Details page and can be seen in a discarded state on the Tracking page. You cannot perform any further operations on a discarded message, including recovery. After a certain time period, the error message is permanently deleted from the server.

To discard errors by integration instance:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**, then click **Error Message Details**.

   The Error Message Details page is displayed. Any errors that are displayed by default are those that occurred within the selected time period.

3. Select the appropriate method for discarding errors:

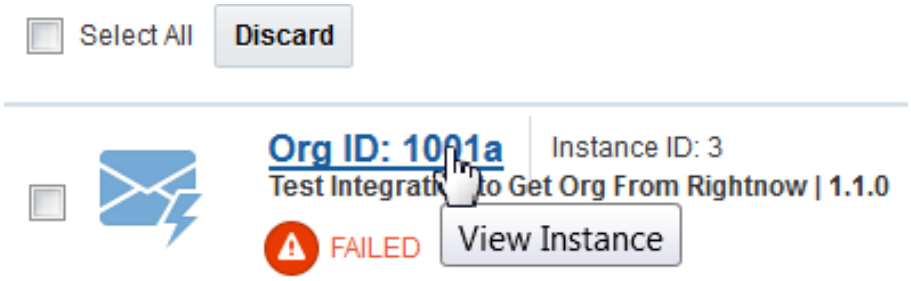
   a. To discard all errors, click **Select All**, then click **Discard**.

   b. To discard selected errors, select the check boxes for the appropriate errors, then click **Discard**.

   c. To discard a single error, select the checkbox for the error, then click **Discard** or simply click **Discard** at the far right of the error to discard.

      You can also view and discard errors on the integration instance page.

4. Click the business identifier included in the failed instance.



The integration instance is displayed.

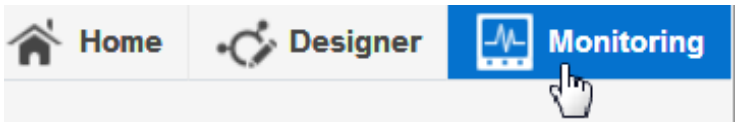
5. To delete the error, click **Actions**, then click **Discard**.
6. Click **Yes** when prompted to confirm you selection.

Viewing Specific Error Details

You can view specific error details by integration instance.

To view specific error details:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**, then click **Error Message Details**.

   The Error Message Details page is displayed. Any errors that are displayed by default are those that occurred within the selected time period.

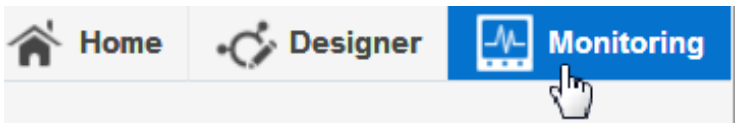
3. To view details about a specific error, click the business identifier included in the failed instance.
4. In the upper right corner, click **View Error**. This action displays the complete error message.

Viewing the Audit Trail of a Failed Integration Instance

You can view the audit trail of a failed integration instance. This enables you to see where an integration error occurred in the message flow.

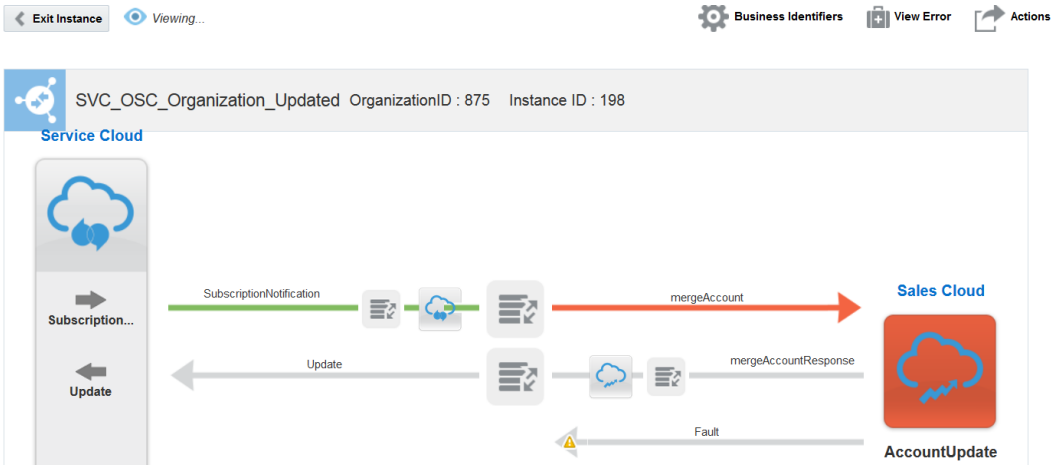
To view the audit trail of a failed integration instance:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



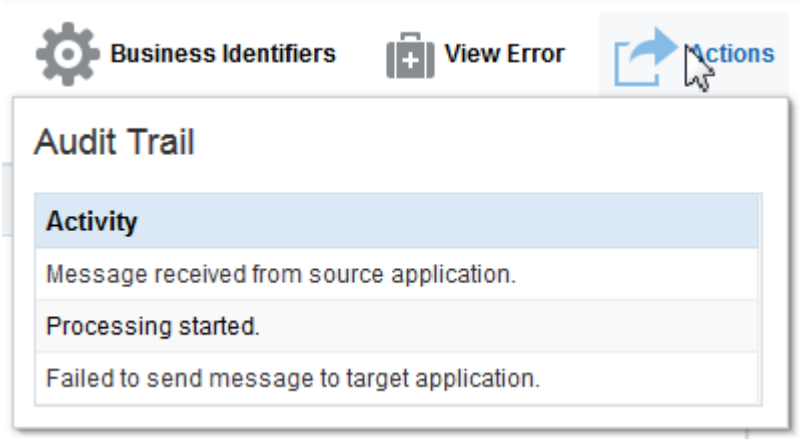
2. In the navigation pane, click **Errors**, then click **Error Message Details**.
3. Click the business identifier included in the instance you want to view.

   The instance integration is displayed. The direction in which the error occurred is indicated by the color red.



4. In the upper right corner, click **Actions**, then click **View Audit Trail**.

   The audit trail shows details about the movement of the message through the integration, including where the failure occurred.

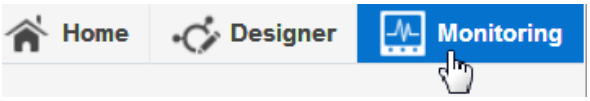


Viewing Business Identifiers in Failed Integration Instances

You can view the business identifiers included in failed integration instances.

To view business identifiers in failed integration instances:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



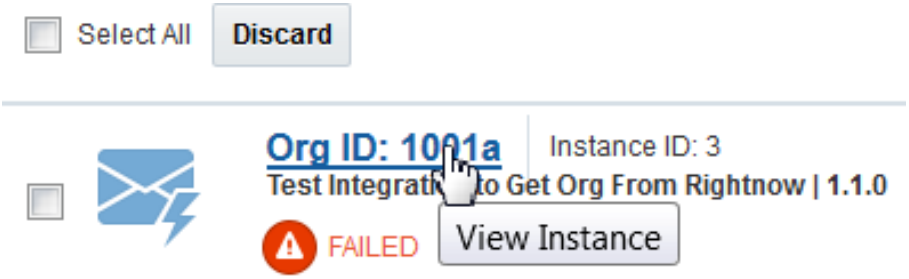
2. In the navigation pane, click **Errors**, then click **Error Message Details**.

   The Error Message Details page is displayed. Any errors that are displayed by default are those that occurred within the selected time period.

3. To search for a specific business identifier, enter the exact business identifier value in the search field, then click **Search**. For example if business identifier **OrgId** has a value of `test2`, enter `test2`.

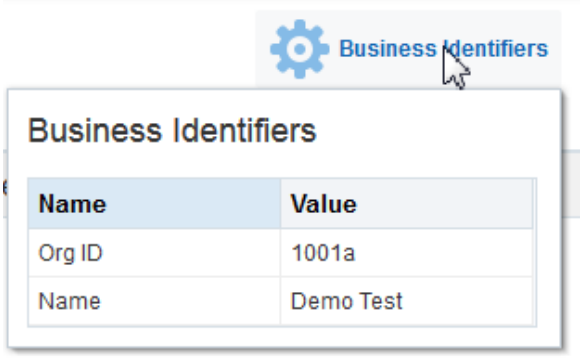
   Any business identifiers with the specified value are displayed.

4. To view details about the business identifiers included in a failed integration instance, click the business identifier.



   The integration instance is displayed.

5. Click **Business Identifiers** to display all the defined business identifiers and values in the integration.



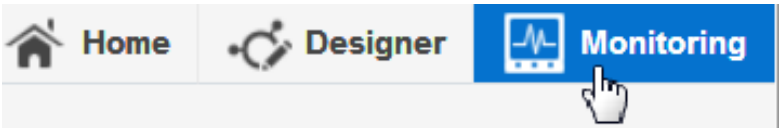
For more information about business identifiers, see Assigning Business Identifiers and Managing Business Identifiers for Tracking Fields in Messages.

Viewing the Message Payload of a Failed Integration Instance

You can view the message payload of a failed integration instance.

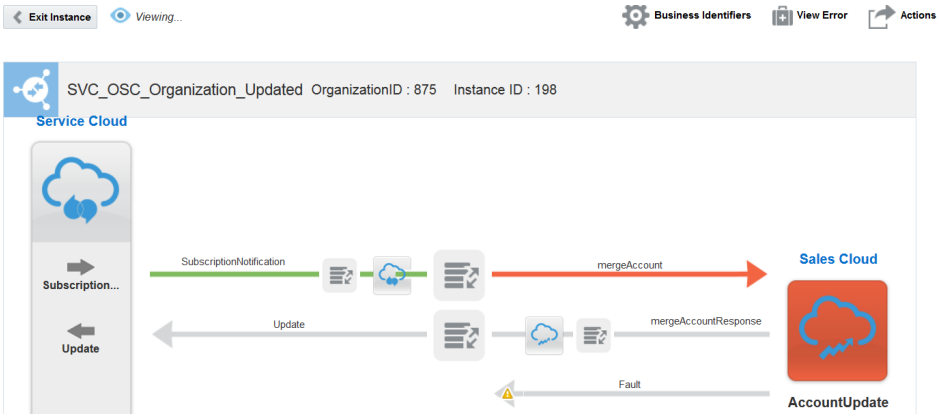
To view the message payload of a failed integration instance:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**, then click **Error Message Details**.
3. Click the business identifier included in the instance you want to view.

   The instance integration is displayed. The direction in which the error occurred is indicated by the color red.



4. In the upper right corner, click **Actions**, then click **View Payload**.

   The audit trail shows the message payload of the integration instance.

Fault Payload

```
<?xml version="1.0" encoding="UTF-16"?>
<Body xmlns="http://schemas.xmlsoap.org/soap/envelope/">
   <nstrgmpr:mergeAccount xmlns=""
      xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
      xmlns:tns="http://xmlns.oracle.com/apps/hcm/employment/absences/absenceCaseServiceV2/"
      xmlns:orafault="http://xmlns.oracle.com/oracleas/schema/oracle-fault-11_0"
      xmlns:types="http://xmlns.oracle.com/apps/hcm/employment/absences/absenceCaseServiceV2/types/"
      xmlns:errors="http://xmlns.oracle.com/adf/svc/errors/"
      xmlns:plnk="http://schemas.xmlsoap.org/ws/2003/05/partner-link/"
      xmlns:nstrgmpr="http://xmlns.oracle.com/cloud/adapter/osc/AccountUpdate_REQUEST/types"
      xmlns:nsmpr0="http://xmlns.oracle.com/cloud/adapter/osc/AccountUpdate_REQUEST"
      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
      xmlns:nsmpr3="http://xmlns.oracle.com/apps/crmCommon/salesParties/accountService/"
      xmlns:nsmpr5="http://www.oracle.com/XSL/Transform/java/com.bea.wli.sb.functions.dvm.DVMFunctions"
xmlns:nsmpr4="http://xmlns.oracle.com/apps/crmCommon/salesParties/commonService/">
      <nstrgmpr:account>
         <nsmpr3:SourceSystem>RNOW</nsmpr3:SourceSystem>
         <nsmpr3:SourceSystemReferenceValue>875</nsmpr3:SourceSystemReferenceValue>
         <nsmpr3:OrganizationName>Act902u1</nsmpr3:OrganizationName>
         <nsmpr3:Type>ZCA_PROSPECT</nsmpr3:Type>
         <nsmpr3:PrimaryAddress>
            <nsmpr4:DeleteFlag>true</nsmpr4:DeleteFlag>
         </nsmpr3:PrimaryAddress>
      </nstrgmpr:account>
```

Resubmitting Failed Messages

You can manually resubmit failed messages. Resubmitting a failed message starts the integration from the beginning.

All faulted instances in asynchronous flows in Integration Cloud Service are recoverable and can be resubmitted. Synchronous flows cannot be resubmitted. You can resubmit errors in the following ways:

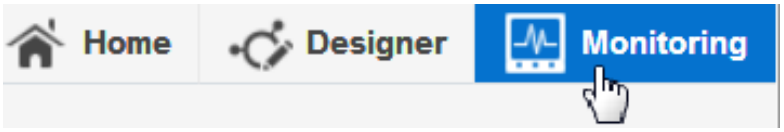
- Single failed message resubmissions
- Bulk failed message resubmissions

Error instances that are resubmitted and successfully resolved are removed from the error list. If an instance is resubmitted and is in progress, a state of `In Progress` is displayed in the list. During this state, additional resubmittals of this error instance are not permitted.

> **Note:** Do *not* discard a message that you want to resubmit. A discarded message cannot be resubmitted.

To resubmit failed messages:

1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Errors**.
3. From the tabs on the left, select the level of message resubmission to perform, then see below for instructions.
    - **Errors By Integration**: For resubmitting all failed messages in an integration.
    - **Errors By Connection**: For resubmitting failed messages in a connection.
    - **Error Message Details**: For resubmitting failed messages in an integration instance.

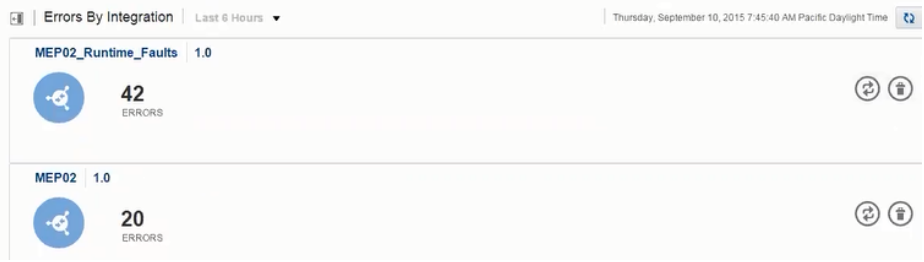
Resubmitting All Failed Messages in an Integration

You can resubmit all failed messages that occurred in an integration.

To resubmit failed messages in an integration:

1. In the navigation pane, click **Errors By Integration**.
2. Find the integration in which to resubmit failed messages.

    At the far right, click the **Resubmit** icon (second from the end) to resubmit the messages. The **Resubmit** icon is not visible for synchronous integrations.



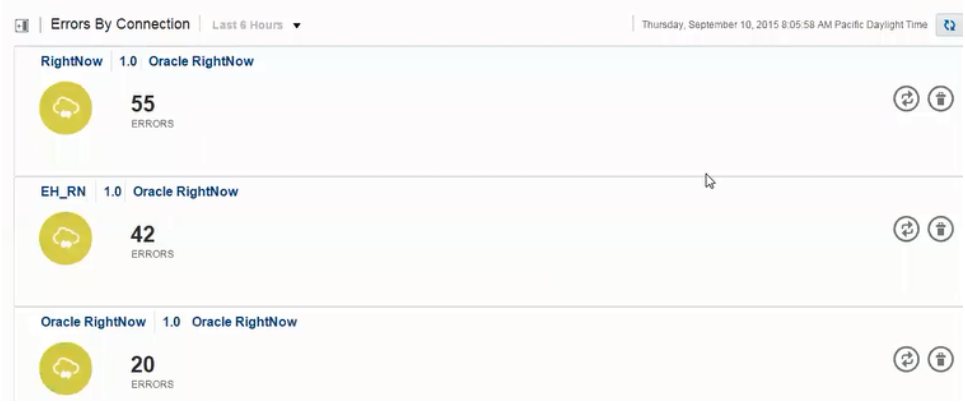
3. Click **Yes** when prompted to confirm.

Resubmitting Failed Messages in a Connection

You can resubmit all failed messages that occurred in a connection.

To resubmit failed messages in a connection:

1. In the navigation pane, click **Errors By Connection**.
2. Find the integration in which to resubmit failed messages.
3. At the far right, click the **Resubmit** icon (second from the end) to resubmit the messages.



4. Click **Yes** when prompted to confirm.

Resubmitting Failed Messages in an Integration Instance

You can resubmit all failed messages that occurred in a specific integration instance.

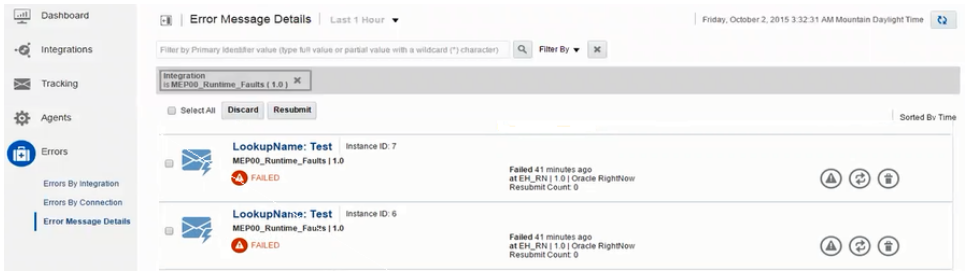
To resubmit failed messages in an integration instance:

1. In the navigation pane, click **Errors**, then click **Error Message Details**.

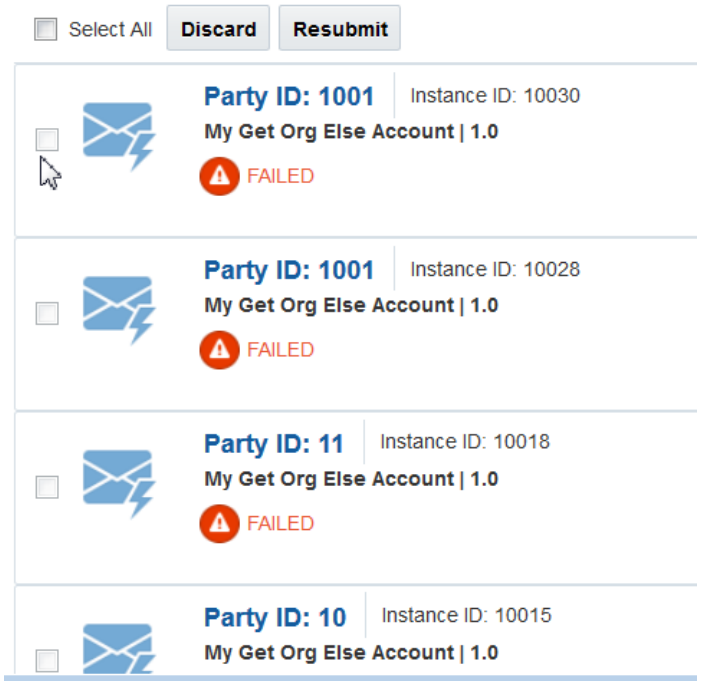
   The Error Message Details page is displayed. Any integration errors that are displayed by default are those that occurred within the selected time period.

2. Select the appropriate method for resubmitting errors.

   a. For single instance resubmissions, optionally click the checkbox, then click the **Resubmit** icon (middle icon at the far right) to resubmit the message.



b. For bulk instance resubmissions, click the appropriate checkboxes, then click **Resubmit** above the integration instances to resubmit the messages.



c. Click **Yes** when prompted to confirm.

Managing Business Identifiers for Tracking Fields in Messages

You can view the status of business identifiers included in integrations on the Tracking page.

Topics

- Tracking Business Identifiers in Integrations During Runtime
- Tracking Business Identifiers in Integrations in Which Routing Paths Are Defined
- Filtering the Display of Business Identifiers in Integrations
- Viewing an Instance Payload

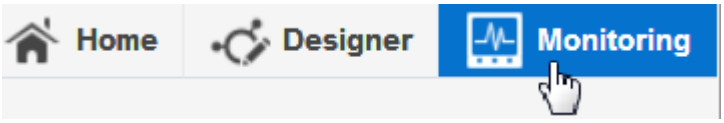
For more information about business identifiers, see Assigning Business Identifiers for Tracking Fields in Messages.

Tracking Business Identifiers in Integrations During Runtime

You can track fields in messages on which you have defined business identifiers on the Tracking page during runtime. These fields are only available for tracking on the Tracking page if you defined a primary business identifier in the Business Identifiers for Tracking dialog during design time.

To track business identifiers in integrations during runtime:

1. On the Integration Cloud Service toolbar, click **Monitoring**.

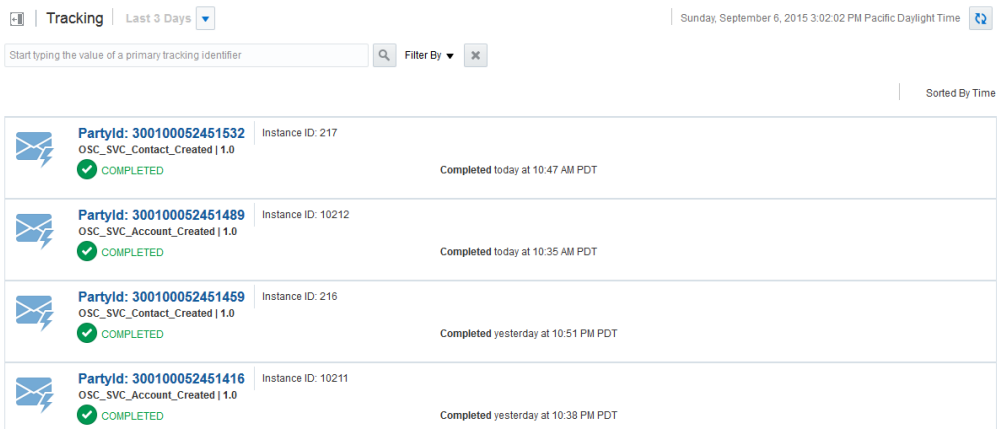


2. In the navigation pane, click **Tracking**.

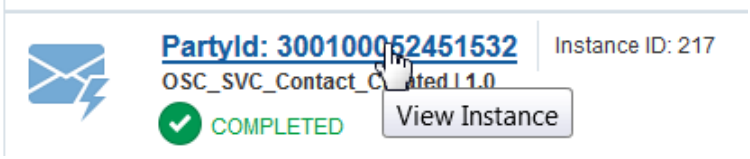
   The Tracking page is displayed.

3. From the menu, select the time period during which to search for business identifiers in messages.

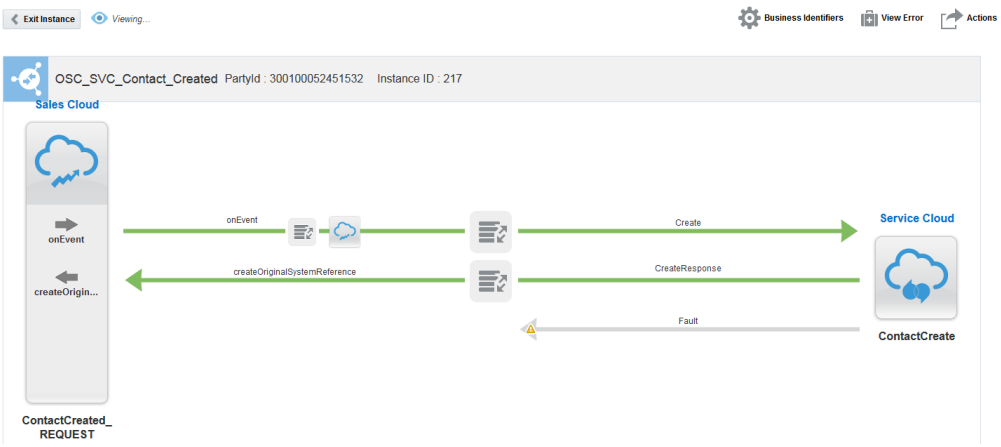
   Results are displayed for any integration on which a primary business identifier is set, including the business identifier and value, the instance identifier of the integration, and the state of the integration (for example, completed, failed, or aborted).



4. Click the business identifier to access a graphical display of the integration instance.



This page provides information about the business identifiers and values defined for the integration, the instance identifier of the integration, any error message, the audit trail, a button for discarding an error, and other information.



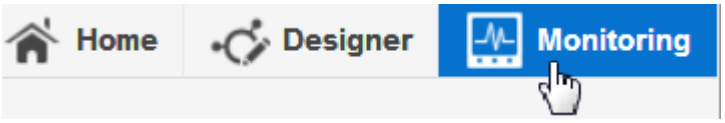
Note: If you enter a primary business identifier in the **Search** field, but do not click the **Search** button, then select a value from the time period dropdown list, note that the instances are filtered considering the string entered in the **Search** field, even though the **Search** button was not clicked. This is the expected behavior and is true for other landing pages in Integration Cloud Service.

Tracking Business Identifiers in Integrations in Which Routing Paths Are Defined

If the integration in which you defined business identifiers also includes definitions for routing paths, you can view the value of the business identifier, the status of the routing path taken based on the business identifier value, the routing expression logic in the blue header above the integration, and the status of the overall integration flow. The status of the routing path taken and the overall integration flow are indicated by color (green indicates success and red indicates failure).

To track business identifiers in integrations in which routing paths are defined:

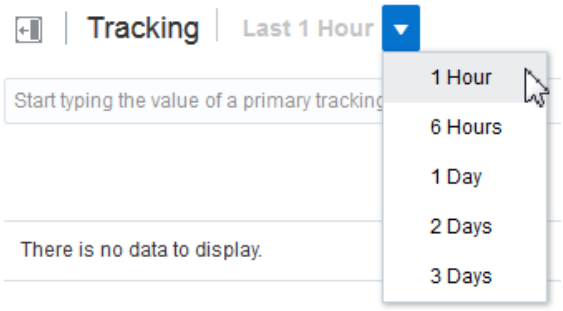
1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Tracking**.

   The Tracking page is displayed.

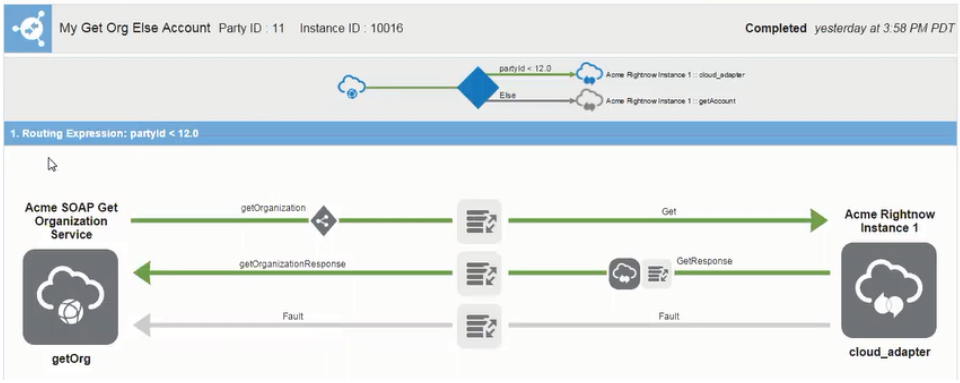
3. From the menu, select the time period during which to search for business identifiers in messages.



4. Click the business identifier to access a graphical display of the integration instance.

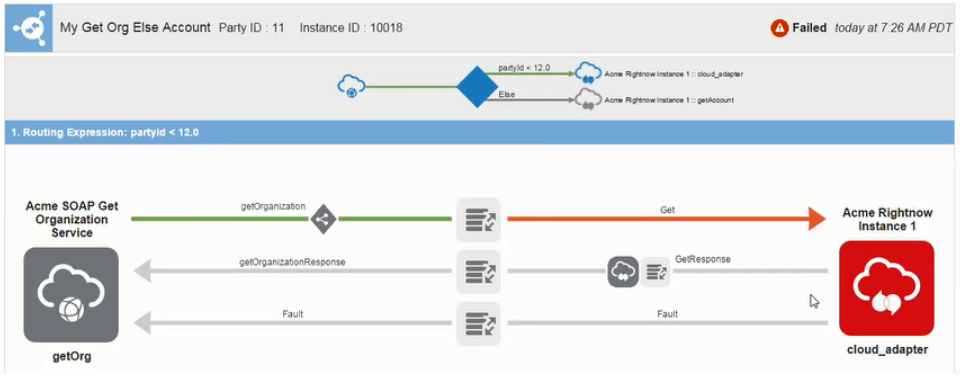
   The page shows (from top to bottom) the value of the business identifier, the status of the routing path taken based on the business identifier value, the routing expression logic in the blue header above the integration, and the status of the overall integration flow. Several examples of what can potentially be displayed are shown below.

   - The following example shows that the business identifier **Party ID** value is **11**. Based on this value, the IF portion of the routing expression logic (**partyid < 12.0**) completed successfully (as indicated by green). Therefore, the ELSE portion was never taken (as indicated by gray). Message delivery in the overall integration flow (request and response parts) also completed successfully (also indicated by green).

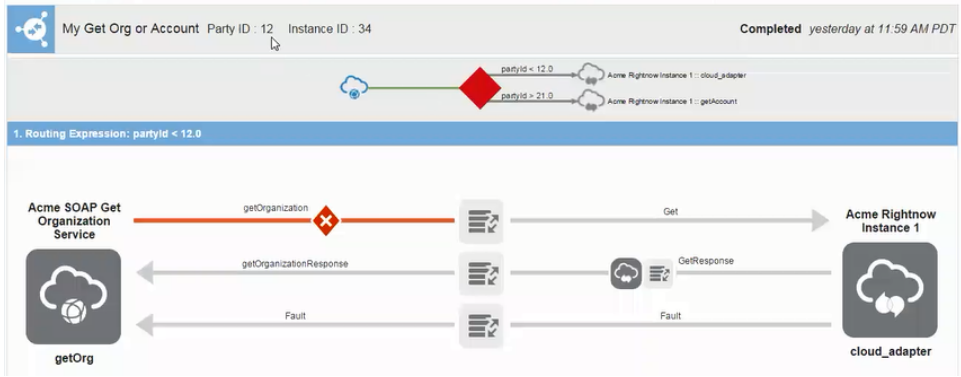


     If the business identifier **Party ID** value had been **12** or greater, the IF portion of the routing expression logic (**partyid < 12.0**) would not have been taken and the ELSE portion would have been taken and displayed in green.

   - The following example shows that the business identifier **Party ID** value is **11**. Based on this value, the IF portion of the routing expression logic (**partyid < 12.0**) completed successfully (as indicated by green). The ELSE path was not taken (as indicated by gray). However, the message delivery in the overall integration flow failed due to an error with the target connection (indicated by red).



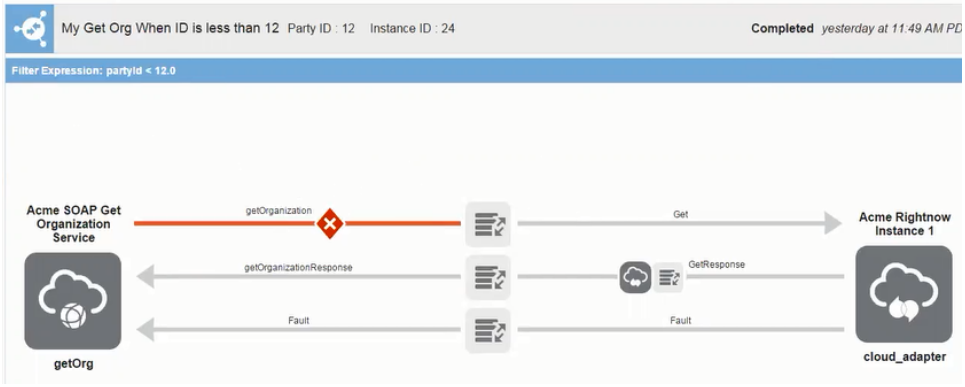
- The following example shows that the business identifier **Party ID** value is **12**. Therefore, the routing expression logic for the IF (**partyid < 12.0**) and ELSE (**partyid >21.0**) portions is not satisfied and neither path is taken. The triangle is displayed in red. Therefore, the request message was never delivered to the target in the overall integration and the integration is displayed as unsuccessful (indicated by red).



If you select **Actions** > **Audit Trail** in the upper right corner, the following message is displayed:

```
Failed to send to target because all routing expressions failed on
evaluation.
```

- The following example shows that the business identifier **Party ID** value is **12**. In this example, only a single routing expression is defined (**partyid < 12.0** in the routing expression logic in the blue header above the integration). Because only a single routing expression is defined, no routing diagram is displayed. The request message is never delivered to the target and the overall integration flow is unsuccessful (indicated by red).

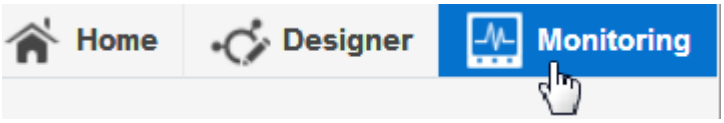


Filtering the Display of Business Identifiers in Integrations

You can filter the display of business identifiers on the Tracking page during runtime.

To filter the display of business identifiers in integrations:

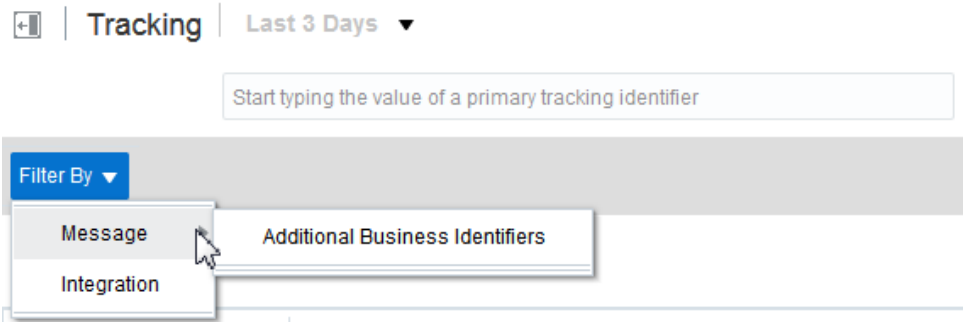
1. On the Integration Cloud Service toolbar, click **Monitoring**.



2. In the navigation pane, click **Tracking**.

   The Tracking page is displayed.

3. Select a method for filtering the display of business identifiers and integrations.



To Filter By...	Perform the Following Steps...
Primary business identifier value	a. In the **Search** field, enter the exact business identifier value, then click **Search** .

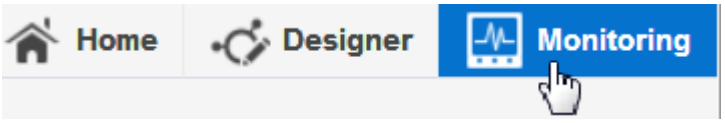
To Filter By...	Perform the Following Steps...
Additional business identifiers Some integrations may have additional business identifiers. Depending on the integration, you may be able to filter messages by additional business identifiers.	a. Select **Filter By** > **Message** > **Additional Business Identifiers**. b. Begin entering the integration name to display names that begin with those letters or select an integration from the dropdown list to see the available tracking fields. c. Click **Set Filter**. The search is performed.
Integrations	a. Select **Filter By** > **Integration**. b. Begin entering the integration name to display names that begin with those letters or select an integration from the dropdown list. c. Click **Set Filter**. The search is performed.

Viewing an Instance Payload

During development and debugging, it is often useful to view an instance message payload.

To view an instance message payload:

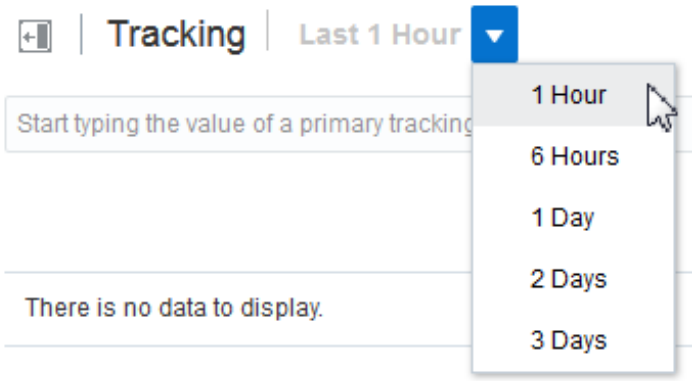
1. On the Integration Cloud Service toolbar, click **Monitoring**.



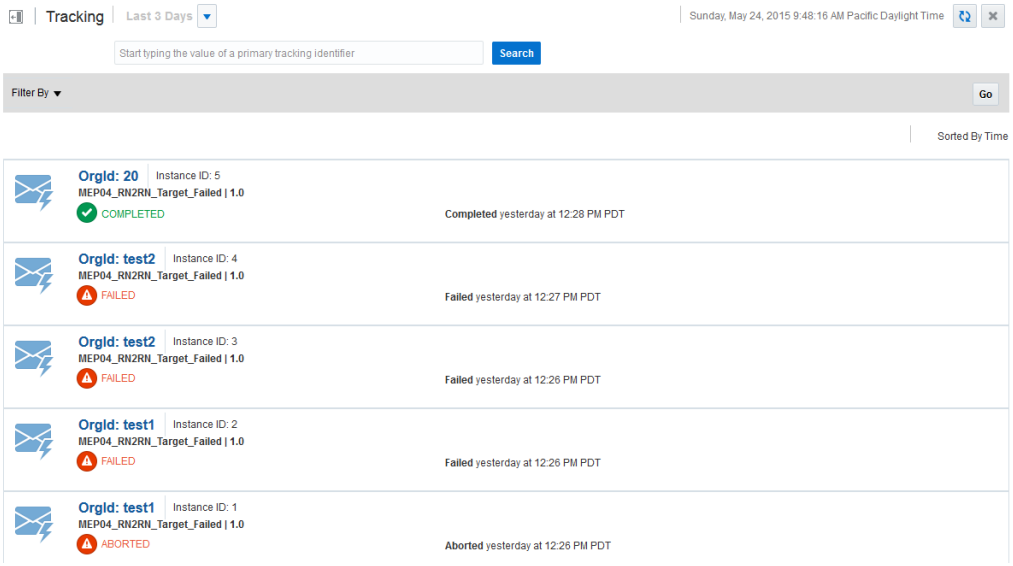
2. In the navigation pane, click **Tracking**.

   The Tracking page is displayed.

3. From the menu, select the time period during which to search for business identifiers in messages.



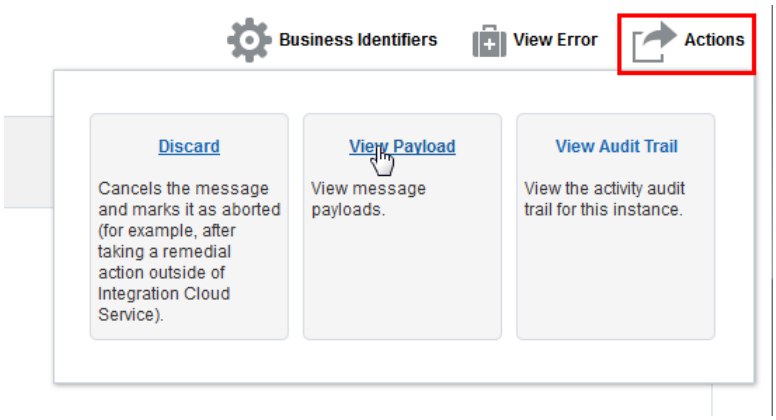
Results are displayed for any integration on which a primary business identifier is set, including the business identifier and value, the instance identifier of the integration, and the state of the integration (for example, completed, failed, or aborted).



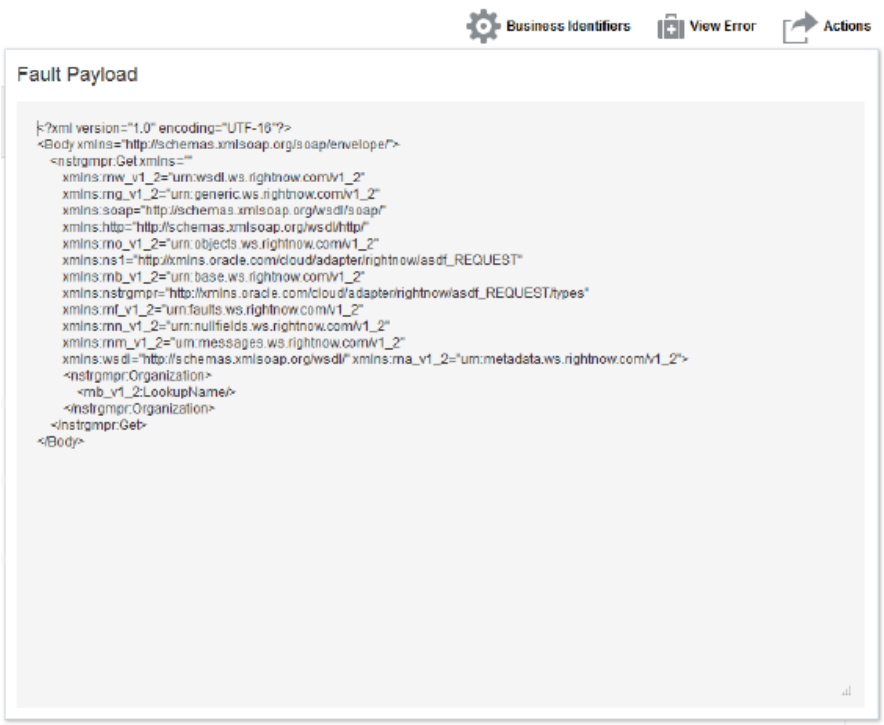
4. Click the business identifier to access a graphical display of the integration instance.

This page provides information about the message payloads, business identifiers and values defined for the integration, the instance identifier of the integration, any error message, the audit trail, the message payload, a button for discarding an error, and other information.

5. Click the **Actions**, then click the **View Payload** link.



The message payload is displayed.



5

Adapter Configuration Reference

When you add an adapter to a trigger (source) or an invoke (target) in an integration, a wizard prompts you to configure how the data is processed for that connection, including the type of operation to perform, the business objects or fields to use, and so on. The properties you can configure vary by each type of adapter. Click any of the topics below to read more about the properties you can configure for each type of adapter.

Topics

Configuring Basic Information Properties

You can enter a name and description on the Basic Info page of each trigger and invoke adapter in your integration.

Topics

What You Can Do from the Basic Info Page

You can specify the following values on the Basic Info page. The Basic Info page is the initial wizard page that is displayed whenever you drag an adapter to the trigger (source) or invoke (target) area supported by your adapter.

- Specify a meaningful name.
- Specify a description of the responsibilities.

What You See on the Basic Info Page

The following table describes the key information on the Basic Info page.

Element	Description
What do you want to call your endpoint?	Provide a meaningful name so that others can understand the responsibilities of this connection. You can include English alphabetic characters, numbers, underscores, and dashes in the name. You cannot include the following: • Blank spaces (for example, `My Inbound Connection`) • Special characters (for example, `#;83&` or `righ(t)now4`) • Multibyte characters
What does this endpoint do?	Enter an optional description of the connection's responsibilities. For example: `This connection receives an inbound request to synchronize account information with the cloud application.`

Where to Go Next

The following table describes where to go after completing the Basic Info page.

See the following section based on the direction of the connection you are creating.

> **Note:** The Oracle REST adapter has its own unique Basic Info page. References to that page are provided in the following table.

If Configuring this Connection...	As a Source...	As a Target...
Oracle RightNow Cloud	Configuring Oracle RightNow Cloud Source Request Properties	Configuring Oracle RightNow Cloud Target Operations Properties

If Configuring this Connection...	As a Source...	As a Target...
Oracle Sales Cloud	Configuring Oracle Sales Cloud Source Request Properties	Configuring Oracle Sales Cloud Target Operations Properties
Oracle HCM Cloud	Configuring Oracle HCM Cloud Source Request Properties	Configuring Oracle HCM Cloud Target Operation Properties
Oracle ERP Cloud	Configuring Oracle ERP Cloud Trigger Request Properties	Configuring Oracle ERP Cloud Invoke Operation Properties
Oracle Eloqua Cloud	N/A	Configuring Oracle Eloqua Cloud Target Operations Properties
Oracle Messaging Cloud Service	Configuring Oracle Messaging Cloud Service Source Operations Properties	Configuring Oracle Messaging Cloud Service Target Operations Properties
Salesforce Cloud	Configuring Salesforce Trigger Outbound Messaging Properties	Configuring Oracle Salesforce Target Operations Properties
Oracle CPQ Cloud	Configuring Oracle CPQ Trigger Request Properties	Configuring Oracle CPQ Invoke Operation Properties
Oracle REST Adapter	Configuring REST Adapter Basic Information Properties	Configuring REST Adapter Basic Information Properties
NetSuite Adapter	N/A	Configuring NetSuite Adapter Invoke Operation Properties
Oracle SOAP Adapter	Configuring SOAP Adapter Trigger Operation Properties	Configuring SOAP Adapter Invoke Operation Properties
Oracle Siebel Adapter	Using the Oracle Siebel Adapter	Using the Oracle Siebel Adapter
Oracle E-Business Suite Adapter	N/A	Using Oracle E-Business Suite Adapter
SAP Adapter	Using the SAP Adapter	Using the SAP Adapter
Google Calendar Adapter	N/A	Using the Google Calendar Adapter
Google Task Adapter	N/A	Using the Google Task Adapter
Oracle Database Adapter	Using the Oracle Database Adapter	Using the Oracle Database Adapter

If Configuring this Connection...	As a Source...	As a Target...
Evernote Adapter	N/A	Using Oracle Evernote Cloud Adapter
Eventbrite Adapter	N/A	Using the Eventbrite Adapter
LinkedIn Adapter	N/A	Using the LinkedIn Adapter
Facebook Adapter	N/A	Using the Facebook Adapter
Gmail Adapter	N/A	Using the Gmail Adapter
Twitter Adapter	N/A	Using the Twitter Adapter
Oracle Commerce Adapter Cloud	Using the Oracle Commerce Cloud Adapter	Using the Oracle Commerce Cloud Adapter
Microsoft Email Adapter	N/A	Using the Microsoft Email Adapter
Microsoft Contact Adapter	N/A	Using the Microsoft Contact Adapter
Microsoft Calendar Adapter	N/A	Using the Microsoft Calendar Adapter
MySQL Adapter	N/A	Using the MySQL Adapter
MailChimp Adapter	N/A	Using the MailChimp Adapter
SurveyMonkey Adapter	N/A	Using the SurveyMonkey Adapter
FTP Adapter	Using the FTP Adapter	Using the FTP Adapter

Configuring Oracle RightNow Cloud Properties

The Oracle RightNow Cloud Adapter enables you to create integrations with an Oracle RightNow Cloud application.

The following sections describe the wizard pages that guide you through configuration of the Oracle RightNow Cloud Adapter as a trigger and invoke in an integration.

Topics

- Reviewing Configuration Values on the Summary Page

For more information about the Oracle RightNow Cloud Adapter, see Oracle RightNow Cloud.

Configuring Oracle RightNow Cloud Trigger Request Properties

Enter the Oracle RightNow Cloud trigger request values for your integration. The values you specify start the integration.

Topics

- What You Can Do from the Oracle RightNow Cloud Source Request Page
- What You See on the Oracle RightNow Cloud Source Request Page

What You Can Do from the Oracle RightNow Cloud Trigger Request Page

You can configure the following request value for Oracle RightNow Cloud.

- Select to receive a business object as a request from the Oracle RightNow application. This selection invokes the integration.
- Select to receive an event subscription as a request from the Oracle RightNow application. This selection invokes the integration. Event subscriptions are supported only if the Oracle RightNow Application version is equal to or greater than release 15.5 (May 2015). Otherwise, only business objects are supported.

> **Note:** The Oracle RightNow application has a limit of 20 subscriptions for every available event subscription. For example, you can build a maximum of 20 integrations, with all subscribing to the Customer Created Event integration and 20 integrations subscribing to the Contact Destroy Event integration. If you create a 21st integration for the same event subscription, this may lead to errors during integration activation.

What You See on the Oracle RightNow Cloud Trigger Request Page

The following table describes the key information on the Oracle RightNow Cloud trigger Request page.

Element	Description
Configure a Request	Select the endpoint configuration option by choosing a business object or event subscription. • **With Business Objects**: Select to display a list of business objects. • **Event Subscription**: Select to display a list of event subscriptions to which to subscribe.
Select a Business Object (is displayed if **With Business Objects** is selected)	Select the business object from the Oracle RightNow Cloud application to receive as a request that starts the integration.

Element	Description
Filter by object name (is displayed if **With Business Objects** is selected)	Enter the initial letters to filter the display of business objects. You can also select a filter type: • **All**: Displays all objects. • **Custom**: Displays objects you created. These business objects are identified by special icons. The naming convention is a combination of the package name and object name joined by a "`.`". For example, if there is a custom object package called `CO` and an object named `PurchaseProduct`, the wizard displays the custom object as `CO.PurchaseProduct`. • **Standard**: Displays business objects delivered as part of the standard Oracle RightNow Cloud application.
Select Event (is displayed if **Event Subscription** is selected)	Select the event subscription from the Oracle RightNow Cloud application. This event is received as a request that starts the integration. **Note**: Only the **Organization** and **Contact** business objects are supported for event subscriptions in this release.
Event Name Filter (is displayed if **Event Subscription** is selected)	Enter the initial letters to filter the display of business events.

Configuring Oracle RightNow Cloud Trigger Response Properties

Enter the Oracle RightNow Cloud trigger response values for your integration.

Topics

What You Can Do from the Oracle RightNow Cloud Trigger Response Page

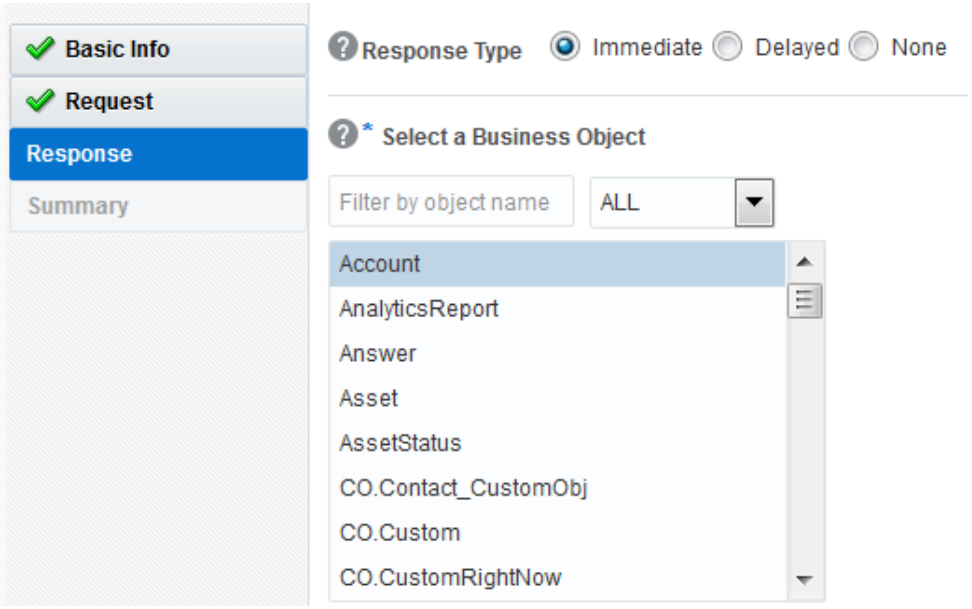
You can configure the operation and business object that comprise the response type for Oracle RightNow Cloud.

- Immediate (synchronous) response: A response business object is immediately returned as output. You select **Immediate** as the response type on the Response page and select the business object as part of the response to the client.
- Delayed (asynchronous) response: A callback service to which to route the callback is exposed. You select **Delayed** as the response type on the Response

page and select the operation and business object that comprise a successful callback response, a failed callback response, or both.

- No response is required: You select **None** on the Response page because a response is not required.

The Response page looks as follows:



What You See on the Oracle RightNow Cloud Trigger Response Page

Select the business object for the integration to send as a response document to the Oracle RightNow Cloud application.

The following types of responses are available.

- Immediate: A synchronous response is required (See Table 5-1 for instructions)
- Delayed: An asynchronous response is required (See Table 5-2 for instructions)
- None: No response is required (See Table 5-3 for instructions)

The following table describes the fields available if an immediate (synchronous) response is required.

Table 5-1 Response Type — Immediate (Synchronous) Response is Required

Element	Description
Response Type	Select **Immediate** for the Oracle RightNow Cloud application to wait until a response is received from the integration. This is also known as the request and response message exchange pattern. This is the default selection.

Table 5-1 (Cont.) Response Type — Immediate (Synchronous) Response is Required

Element	Description
Filter by object name	Enter the initial letters to filter the display of business objects. You can also select a filter type: • **All**: Displays all objects. • **Custom**: Displays objects you created. These business objects are identified by special icons. The naming convention is a combination of the package name and object name joined by a "`.`". For example, if there is a custom object package called `CO` and an object named `PurchaseProduct`, the wizard displays the custom object as `CO.PurchaseProduct`. • **Standard**: Displays business objects delivered as part of the standard Oracle RightNow Cloud application.
Select a Business Object	Select the business object for the integration to send as a response document to the Oracle RightNow Cloud application.

Table 5-2 Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Response Type	Select **Delayed** to configure a successful callback response, a failed callback response, or both. This enables you to configure the operation and business objects that you want the Oracle RightNow Cloud application to process as part of a successful callback response, failed callback response, or both.

Table 5-2 (Cont.) Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Successful Response/Failed Response	Select the type of callback to configure. After configuring one type of callback (for example, successful), you can configure the other type (for example, failed). • **Successful Response**: Select to configure the operation and business objects that you want the Oracle RightNow Cloud application to process as part of a successful callback response sent by the integration. • **Failed Response**: Select to configure the operation and business objects that you want the Oracle RightNow Cloud application to process as part of a failed callback response sent by the integration.
Select an Operation Type	Select the type of create, read, update, and delete (CRUD) operation to perform on the business object. Only CRUD is currently available for selection. CRUD represents functions implemented in relational database applications. Each letter maps to a standard SQL statement, HTTP method, or DDS operation. The following CRUD operations are supported: • **Create** • **Destroy** • **Update**
Filter By object name	Enter the initial letters to filter the display of business objects. You can also select a filter type: • **All**: Displays all objects. • **Custom**: Displays objects you created. These business objects are identified by special icons. The naming convention is a combination of the package name and object name joined by a '.'. For example, if there is a custom object package called `CO` and an object named `PurchaseProduct`, the wizard displays the custom object as `CO.PurchaseProduct`. • **Standard**: Displays business objects delivered as part of the Oracle RightNow Cloud application.

Table 5-2 (Cont.) Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Select Business Objects (RightNow *Version* API)	Select the business objects for the integration to send as a response document to the Oracle RightNow Cloud application.
Your Selected Business Objects	Displays the selected business objects.

The following table describes the fields available if no response is required.

Table 5-3 Response Type — No Response is Required

Element	Description
Response Type	Select **None**.
Select Business Object	If you select **None**, this section is hidden.

Configuring Oracle RightNow Cloud Invoke Operation Properties

Enter the Oracle RightNow Cloud invoke operation values for your integration.

Topics

What You Can Do from the Oracle RightNow Cloud Invoke Operations Page

You can configure the following values for Oracle RightNow Cloud:

- Select to configure a single operation or multiple operations in a batch.
- Select the operation (CRUD or RightNow Object Query Language (ROQL)).
- Select the business objects.
- Enable aspects of server-side processing.

What You See on the Oracle RightNow Cloud Invoke Operations Page

The following table describes the key information on the Oracle RightNow Cloud invoke Operations page.

Element	Description
Select an Operation Mode	Select the operation mode in which to define business objects: • **Single Operation**: Select to configure a single operation. • **Batch Operation**: Select to configure multiple operations in a batch. This enables you to run multiple operations in a defined sequence. Selecting this option refreshes the page to display an option for the following: – **Click to add an operation to the List**: Click to create a list of batch operations and their business objects. The operations are performed in the order in which they appear in the list (from top to bottom). When complete, click **OK**. When you complete invoke Oracle RightNow adapter configuration and click **Next** to access the Summary page, you can perform the following batch operation tasks: – **Edit** icon: Click to edit an operations row in the table or change the order of batch operations. – **Delete** icon: Click to delete a selected operation row in the table.

Element	Description

Select an Operation Type	Select the type of operation to perform on the business objects in an Oracle RightNow application:
	• **CRUD**: Represents the create, read, update, delete, or destroy operations to perform on Oracle RightNow business objects. Each letter maps to a standard SQL statement, HTTP method, or DDS operation. Select the CRUD operation to perform on the business object: **Create**, **Destroy**, **Get**, or **Update**.
	• **ROQL**: (RightNow Object Query Language) enables you to define an ROQL-based query to send as a request to perform in the Oracle RightNow application. If you select this option, the page is refreshed to display a field for entering a query.
	– **ROQL query statement**: Enter a valid ROQL query in the field. For example:
	`SELECT contacts FROM organization WHERE name = 'RightNow'`
	Use the **Find** field to search for an entry in the ROQL query and the **Go to Line** field to go to a specific line in the ROQL query.
	The query can include custom fields and parameters.
	– **Parameter Bindings**: Displays any parameter bindings included in the specified query. For example, `orgId` is a parameter in the following query:
	`SELECT Organization FROM Organization WHERE id = &orgId`
	Enter a query with a parameter and click the **Refresh** icon to the right of **Parameter Bindings**. This displays a text box in which to enter a test value for the parameter.
	– **Test My Query**: Click to validate the query against the Oracle RightNow application. Query results are displayed. If errors occur, you receive results about how to correct the query.

Element	Description
Filter by object name	Enter the initial letters of an object name to display a range of objects. You can also select a filter type: • **All**: Displays all objects. • **Custom**: Displays objects you created. These business objects are identified by special icons. The naming convention is a combination of the package name and object name joined by a '.'. For example, if there is a custom object package called `CO` and an object named `PurchaseProduct`, the wizard displays the custom object as `CO.PurchaseProduct`. • **Standard**: Displays business objects delivered as part of the Oracle RightNow application.
Select Business Objects (RightNow *Version* API)	Select a single business object or multiple business objects from the Oracle RightNow application. The selected operation acts upon these business objects. The RightNow API version that is displayed is based on the Oracle RightNow Cloud application version to which you are connected. When you complete invoke operation configuration, the selected operation and business objects are defined in the integration-centric WSDL file.
Your Selected Business Objects	Displays the selected business objects.

Element	Description
Processing Options	Select to enable aspects of server-side processing. By default, no options are selected. When complete, click **OK**. • **Suppress External Events**: Select to prevent the Oracle RightNow application from processing any external events raised after the completion of create, update, or delete operations. • **Suppress Rules**: Select to prevent business rules from running after the completion of create, update, or delete operations. Business rules are tools for simplifying and automating common business tasks. See the RightNow documentation for more information. • **Suppress Response**: Select to prevent the CRUD **Create** operation from returning a response ID. If this check box is disabled, the **Create** operation returns an ID of the created object. • **Commit After**: Select to group multiple operations in a single transaction. At runtime, when a set of operations in a batch is defined as part of a single operation, the **Commit After** action is sent after the last operation in that transaction boundary. When an operation from the subset of the batch operation fails, it is handled by throwing a fault to the client. This option is only available with batch operations.

Configuring Oracle Sales Cloud Properties

The Oracle Sales Cloud Adapter enables you to create integrations with an Oracle Sales Cloud application.

The following sections describe the wizard pages that guide you through configuration of the Oracle Sales Cloud Adapter as a trigger and invoke in an integration.

Topics

For more information about Oracle Sales Cloud Adapter, see Oracle Sales Cloud.

Configuring Oracle Sales Cloud Trigger Request Properties

Enter the Oracle Sales Cloud connection trigger request values for your integration. The values you specify start the integration.

Topics

What You Can Do from the Oracle Sales Cloud Trigger Request Page

You can select the following trigger request values for the Oracle Sales Cloud application.

Select the specific type to receive as a request from Oracle Sales Cloud. Your ability to select either a business object or event subscription is based on the content of the WSDL file (for business objects) or event catalog URL (for event subscriptions) you specified during Oracle Sales Cloud Adapter configuration.

- Select to receive a business object as a request from Oracle Sales Cloud. This selection invokes the integration.
- Select to receive an event subscription raised by the Oracle Sales Cloud application as a request from Oracle Sales Cloud. This selection invokes the integration.

What You See on the Oracle Sales Cloud Trigger Request Page

The following table describes the key information on the Oracle Sales Cloud Adapter trigger Request page.

Element	Description
Configure a Request	Select the request type appropriate to your integration. The fields that are displayed below are based on the request type that you select. • **With Business Objects**: Select to display a list of business objects. • **With Business Events**: Select to display a list of event subscriptions
Select a Business Object (is displayed if **With Business Objects** is selected)	Select the business object from the Oracle Sales Cloud application to receive as a request that starts the integration.
Business Event For Subscription (is displayed if **With Business Events** is selected)	Select the event subscription from the Oracle Sales Cloud application to which to subscribe. This event is received as a request that starts the integration.

Element	Description
Filter Expr for *Business_Event_Name* (is displayed if **With Business Events** is selected)	Enter an event condition filter expression. A filter expression specifies that the contents (payload or headers) of a message be analyzed before any event subscription is sent. For example, you can apply a filter expression that specifies that an event subscription be sent only if the message includes a customer ID. When the expression logic is satisfied, the event is accepted for delivery to the integration.
Filter by object name or **Filter By Event Name**	Type the initial letters of the name to filter the display of business objects or event subscriptions.

Configuring Oracle Sales Cloud Trigger Response Properties

Enter the Oracle Sales Cloud trigger response values for your integration.

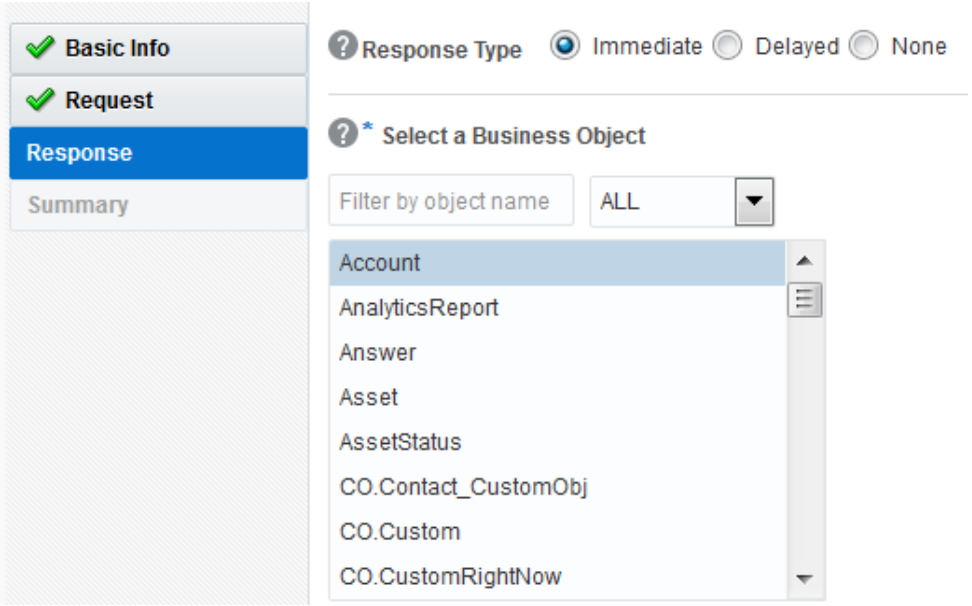
Topics

What You Can Do from the Oracle Sales Cloud Trigger Response Page

You can configure the operation and business object that comprise the response type for Oracle Sales Cloud.

- Immediate (synchronous) response: A response business object is immediately returned as output. You select **Immediate** as the response type on the Response page and select the business object as part of the response to the client.
- Delayed (asynchronous) response: A callback service to which to route the callback is exposed. You select **Delayed** as the response type on the Response page and select the operation and business object that comprise a successful callback response, a failed callback response, or both.
- No response is required: You select **None** on the Response page because a response is not required.

The Response page looks as follows:



What You See on the Oracle Sales Cloud Trigger Response Page

Select the business object for the integration to send as a response document to the Oracle Sales Cloud application.

The following types of responses are available.

- Immediate: A synchronous response is required (See Table 5-4 for instructions)
- Delayed: An asynchronous response is required (See Table 5-5 for instructions)
- None: No response is required (See Table 5-6 for instructions)

The following table describes the fields available if an immediate (synchronous) response is required.

Table 5-4 Response Type — Immediate (Synchronous) Response is Required

Element	Description
Response Type	Select **Immediate** for the Oracle Sales Cloud application to wait until a response is received from the integration. This is also known as the request and response message exchange pattern. This is the default selection.
Filter by object name	Type the initial letters to filter the display of business objects.
Select a Business Object	Select the business object to receive from the Oracle Sales Cloud application as a response. You can filter the display of business objects by typing the initial letters of business objects in the **Filter by object name** field. A description of the selected business object is displayed below this list.

The following table describes the fields available if a delayed (asynchronous) response is required. You can configure a successful response, a failed response, or both.

Table 5-5 Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Response Type	Select **Delayed** to configure a successful callback response, a failed callback response, or both. This enables you to configure the operation and business objects that you want the Oracle Sales Cloud application to process as part of a successful callback response, a failed callback response, or both.
Successful Response/Failed Response	Select the type of callback to configure. After configuring one type of callback (for example, successful), you can configure the other type (for example, failed). • **Successful Response**: Select to configure the operation and business objects that you want the Oracle Sales Cloud application to process as part of a successful callback response sent by the integration. • **Failed Response**: Select to configure the operation and business objects that you want the Oracle Sales Cloud application to process as part of an error callback response sent by the integration.
Select the operation to perform on the business object	Select the operation to perform on the business object.
Life Cycle	Displays the current state of the selected business document. Active indicates the business document is available for use. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Description	Displays a description of the selected business object or service.

The following table describes the fields available if no response is required.

Table 5-6 None — No Response is Required

Element	Description
Response Type	Select **None**.
Select a Business Object	If you select **None**, this section is hidden.

Configuring Oracle Sales Cloud Invoke Operation Properties

Enter the Oracle Sales Cloud invoke operation values for your integration.

Topics

- What You Can Do from the Oracle Sales Cloud Target Operations Page
- What You See on the Oracle Sales Cloud Target Operations Page

What You Can Do from the Oracle Sales Cloud Invoke Operations Page

You can configure the following invoke operations values for Oracle Sales Cloud.

- Browse for and select a business object, service, or Oracle Fusion Applications REST API resource.
- Select the operation to perform on the business object, service, or Oracle Fusion Applications REST API resource.

What You See on the Oracle Sales Cloud Invoke Operations Page

The following table describes the key information on the Oracle Sales Cloud invoke Operations page.

Configuring Oracle Messaging Cloud Service Properties

Oracle Messaging Cloud Service provides a messaging system for applications to communicate reliably with each other, enabling application developers to share information across multiple applications. Oracle Messaging Cloud Service is heavily influenced by the Java Message Service (JMS) API specification, which is a standard messaging interface for sending and receiving messages between enterprise Java applications.

The following sections describe the wizard pages that guide you through configuration of the Oracle Messaging Cloud Service adapter as a trigger and invoke in an integration

Topics

- Configuring Basic Information Properties
- Configuring Oracle Messaging Cloud Service Source Operation Properties
- Configuring Oracle Messaging Cloud Service Source Message Properties
- Configuring Oracle Messaging Cloud Service Target Operation Properties
- Configuring Oracle Messaging Cloud Service Target Messages Properties
- Reviewing Configuration Values on the Summary Page

For more information about Oracle Messaging Cloud Service, see Oracle Messaging Cloud Service.

Configuring Oracle Messaging Cloud Service Trigger Operation Properties

Enter the Oracle Messaging Cloud Service trigger operation values for your integration.

Topics

What You Can Do from the Oracle Messaging Cloud Service Trigger Operations Page

You can specify the following trigger destination and messaging parameter values for the Oracle Messaging Cloud Service.

- Select the JNDI destination name of the queue or topic.
- Specify the message selector filtering logic.
- Select whether to continue receiving messages while offline.
- Specify the durable subscriber identifier (ID).

What You See on the Oracle Messaging Cloud Service Trigger Operations Page

The following table describes the key information on the Oracle Messaging Cloud Service trigger Operations page.

Element	Description
Select Destination	Select the JNDI destination name of the queue or topic to consume the message. In the trigger direction, the connection polls (consumes) messages from a destination.
Destination Name Filter	Enter the initial letters to filter the display of JNDI destination names. You can also select one of the following filtering options. • **All**: Displays all JNDI destination names. • **Queue**: Displays only the queue names. • **Topic**: Displays only the topic names.

Element	Description
Message Selector	Specify filtering logic to receive messages that match certain criteria. Enter an expression between `1` and `255` characters in length. Use SQL92 syntax in this field. The JMS server uses these criteria to filter messages received by this consumer. This works with variables defined in standard JMS headers and user-defined properties. You cannot use variables or elements that are in the payload of the message. For example, you can enter logic such as: • `JMSPriority > 3`. Based on this, messages with a priority greater than `3` are consumed. All other messages are rejected. • `JMSType = 'car' AND color = 'blue' AND weight 2500` • `Country in ('UK', 'US', 'France')`
Continue to receive messages when offline	Select to continue receiving messages while the Oracle Message Cloud Service is offline. If you selected a topic in the **Select Destination** list, this checkbox is enabled.
Subscriber ID	Specify a unique ID to create a durable subscription to a topic on the Messaging Service Cloud. This field is enabled and is mandatory if you selected the **Continue to receive messages when offline** checkbox. This is the identifier of a durable subscription that outlasts a client's connection with a JMS message server. When a durable subscriber is disconnected from the JMS server, the server must store messages that the subscriber misses. When the durable subscriber reconnects, the message server sends it all the unexpired messages that accumulated. The durable subscription only works for JMS topics. For a nondurable subscriber, the message server does not send it all the unexpired messages. You can specify an ID for receiving messages from a topic (multiconsumer queue). Enter a value between `1` and `255` characters in length. When you specify an ID for topics, you receive messages even if you do not currently have an active subscription session. You can access these messages the next time you have an active subscription session.

Configuring Oracle Messaging Cloud Service Trigger Message Properties

Enter the Oracle Messaging Cloud Service trigger message values for your integration.

Topics

What You Can Do from the Oracle Messaging Cloud Service Trigger Messages Page

You can specify the following trigger schema messaging parameter values for the Oracle Messaging Cloud Service.

- Indicate if the schema file to pass is an opaque message.
- Specify the schema XSD file location.
- Select the schema element.

What You See on the Oracle Messaging Cloud Service Trigger Messages Page

The following table describes the key information on the Oracle Messaging Cloud Service trigger Messages page.

Element	Description
Specify a definition for the message	Deselect if the schema to pass is an opaque message (for example, a GIF or PNG file). Deselecting this check box disables the fields below.
Provide a location to the XSD that describes your message	Click **Browse** to select the XSD file (for example, `/net/myhost/scratch/omsc/expense.xsd`). When selected, the XSD file contents are displayed. The Oracle Messaging Cloud Service connection requires complete XSDs that are self-resolvable in the Integration Cloud Service environment. Any schema file (XSD) that explicitly performs an include or import of any other child schema must be self-resolvable. If not, the schemas (XSDs) are not consumed by the Oracle Messaging Cloud Service connection.
Select the schema element to use in this integration	Select the schema element to use in this integration. The elements available are based on the selected XSD file.

Configuring Oracle Messaging Cloud Service Invoke Operation Properties

Enter the Oracle Messaging Cloud Service invoke operation values for your integration.

Topics

What You Can Do from the Oracle Messaging Cloud Service Invoke Operations Page

You can specify the following invoke destination and messaging parameter values for the Oracle Messaging Cloud Service.

- Select the JNDI destination name of the queue or topic.
- Specify the life span for the message.

What You See on the Oracle Messaging Cloud Service Invoke Operations Page

The following table describes the key information on the Oracle Messaging Cloud Service invoke Operations page.

Element	Description
Select Destination	Select the JNDI destination name of the queue or topic to produce the message. In the invoke direction, the connection sends (produces) messages to a destination.
Destination Name Filter	Enter the initial letters to filter the display of JNDI destination names. You can also select one of the following filtering options: • **All**: Displays all JNDI destination names. • **Queue**: Displays only the queue names. • **Topic**: Displays only the topic names.
Remove Message after *0* seconds	Specify the life span of the message. If no subscribers consume the message in the given life span, the message is not delivered. The maximum time a message can live in Oracle Messaging Cloud Service is 14 days. The time-to-live can be set to a value less than 14 days for any given message. When a message reaches the defined time-to-live value, it is permanently deleted.

Configuring Oracle Messaging Cloud Service Invoke Message Properties

Enter the Oracle Messaging Cloud Service invoke message values for your integration.

Topics

What You Can Do from the Oracle Messaging Cloud Service Invoke Messages Page

You can specify the following invoke schema messaging parameter values for the Oracle Messaging Cloud Service.

- Indicate if the schema file to pass is an opaque message.
- Specify a definition for the message.
- Specify the schema XSD file location.
- Select the schema element.

What You See on the Oracle Messaging Cloud Service Invoke Messages Page

The following table describes the key information on the Oracle Messaging Cloud Service invoke Messages page.

Element	Description
Specify a definition for the message	Deselect if the schema to pass is an opaque message (for example, a GIF or PNG file). Deselecting this check box disables the fields below.
Provide a location to the XSD that describes your message	Click **Browse** to select the XSD file (for example, `/net/myhost/scratch/omsc/costs.xsd`). When selected, the contents are displayed. The Oracle Messaging Cloud Service connection requires complete XSDs that are self-resolvable in the Integration Cloud Service environment. Any schema file (XSD) that explicitly performs an include or import of any other child schema must be self-resolvable. If not, the schemas (XSDs) are not consumed by the Oracle Messaging Cloud Service connection.
Select the schema element to use in this integration	Select the schema element to use in this integration. The elements available are based on the selected XSD file.

Configuring Oracle Eloqua Cloud Properties

The following sections describe the wizard pages that guide you through configuration of the Oracle Eloqua Cloud Adapter as an invoke in an integration. The Oracle Eloqua Cloud Adapter cannot be configured as a trigger in an integration.

Topic

- Configuring Basic Information Properties
- Configuring Oracle Eloqua Cloud Target Operation Properties
- Reviewing Configuration Values on the Summary Page

For more information about Oracle Eloqua Cloud Adapter, see Oracle Eloqua Cloud.

Configuring Oracle Eloqua Cloud Invoke Operation Properties

Enter the Oracle Eloqua Cloud invoke operation values for your integration.

Topics

- What You Can Do from the Oracle Eloqua Cloud Target Operations Tab
- What You See on the Oracle Eloqua Cloud Target Operations Tab

What You Can Do from the Oracle Eloqua Cloud Invoke Operations Page

You can specify the following invoke operation values for the Oracle Eloqua Cloud:

- Business object
- Unique identifier fields
- Field used by the Oracle Eloqua Cloud application to match the data sent by this endpoint to the existing data in the application
- Time interval between data synchronizations

What You See on the Oracle Eloqua Cloud Invoke Operations Page

The following table describes the key information on the Oracle Eloqua Cloud invoke Operations page.

Element	Description
Cloud Operation	Displays the operation to perform. The currently supported operation is **import**. This means that the Oracle Eloqua invoke imports data into the connected Oracle Eloqua Cloud application.

Element	Description
Select the Business Object	Select the business object for the Oracle Eloqua invoke to import into the Oracle Eloqua Cloud application. This enables you to synchronize the business object data from a source system (such as Oracle RightNow Cloud or Oracle Sales Cloud) to the targeted Oracle Eloqua Cloud application. • **Accounts** • **Contacts** • Other custom objects associated with the configured Oracle Eloqua Cloud connection
Filter by Field Name	Type the initial letters of the field to filter the display of field names.
Select the Fields	Select the fields of the business object that you want to send to the Oracle Eloqua Cloud application as part of the import operation.
Your Selected Fields	Displays the selected fields.
Uniquely Match on Field	Select the field that the Oracle Eloqua Cloud application uses to match your data to the existing data in the Oracle Eloqua application. Select a field that is likely to be unique to avoid updating the wrong record. Do not use a large text field because this can potentially produce errors.
Auto-Synch Time Interval	Select the time interval between data synchronizations. The Oracle Eloqua Cloud connection uses this interval to synchronize data with the Oracle Eloqua Cloud application.

Configuring Oracle HCM Cloud Properties

The Oracle HCM Cloud Adapter enables you to create integrations with an Oracle HCM Cloud application.

The following sections describe the wizard pages that guide you through configuration of the Oracle HCM Cloud Adapter as a trigger and invoke in an integration.

Topics

- Configuring Oracle HCM Cloud Target Operation Properties
- Reviewing Configuration Values on the Summary Page

> **Note:** If a selected business object or service has a life cycle value of deprecated, a warning message is displayed. If the business object or service captures information about a replacement business object or service, then it may also be displayed. This information comes from the annotation element in the service's WSDL.

For more information about Oracle HCM Cloud Adapter, see Oracle HCM Cloud.

Configuring Oracle HCM Cloud Trigger Request Properties

Enter the Oracle HCM Cloud trigger request values for your integration. The values you specify start the integration.

Topics

- What You Can Do from the Oracle HCM Cloud Source Request Page
- What You See on the Oracle HCM Cloud Source Request Page

What You Can Do from the Oracle HCM Cloud Source Request Page

You can select the following source request values for Oracle HCM Cloud.

Select the specific type to receive as a request from Oracle HCM Cloud:

- Select to receive either a business object as a request or an event subscription raised by the Oracle HCM Cloud application as a request from Oracle HCM Cloud. This selection invokes the integration. Your ability to select either a business object or event subscription is based on the content of the WSDL file (for business objects) or event catalog URL (for event subscriptions) you specified during Oracle HCM Cloud Adapter configuration.

What You See on the Oracle HCM Cloud Source Request Page

The following table describes the key information on the Oracle HCM Cloud source Request page.

Element	Description
Configure a Request	Select the request type appropriate to your integration. The fields that are displayed below are based on the request type that you select. • **With Business Objects**: Select to display a list of business objects. • **With Business Events**: Select to display a list of event subscriptions.
Select a Business Object (is displayed if **With Business Objects** is selected)	Select the business object from the Oracle HCM Cloud application to receive as a request that starts the integration.

Element	Description
Business Event For Subscription (is displayed if **With Business Events** is selected)	Select the event subscription from the Oracle HCM Cloud application to which to subscribe. This event is received as a request that starts the integration.
Filter Expr for *Business_Event_Name* (is displayed if **With Business Events** is selected)	Enter an event condition filter expression. A filter expression specifies that the contents (payload or headers) of a message be analyzed before any event is sent. For example, you can apply a filter expression that specifies that an event be sent only if the message includes a customer ID. When the expression logic is satisfied, the event is accepted for delivery to the integration.
Filter by object name or **Filter By Event Name**	Type the initial letters of the name to filter the display of business objects or event subscriptions.

Configuring Oracle HCM Cloud Trigger Response Properties

Enter the Oracle HCM Cloud trigger response values for your integration.

Topics

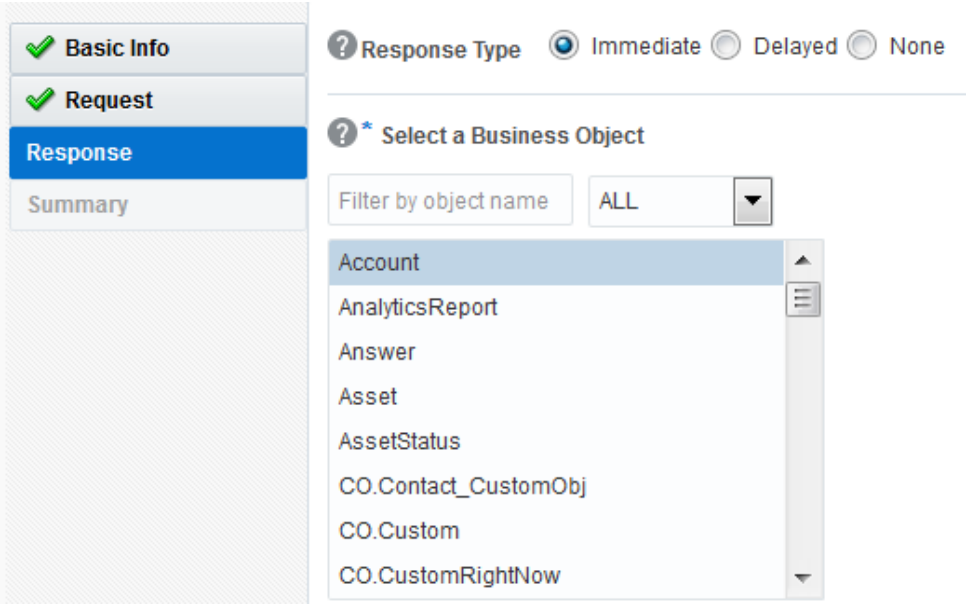
- What You Can Do from the Oracle HCM Cloud Source Response Page
- What You See on the Oracle HCM Cloud Source Response Page

What You Can Do from the Oracle HCM Cloud Trigger Response Page

You can configure the operation and business object that comprise the response type for Oracle HCM Cloud.

- Immediate (synchronous) response: A response business object is immediately returned as output. You select **Immediate** as the response type on the Response page and select the business object as part of the response to the client.
- Delayed (asynchronous) response: A callback service to which to route the callback is exposed. You select **Delayed** as the response type on the Response page and select the operation and business object that comprise a successful callback response, a failed callback response, or both.
- No response is required: You select **None** on the Response page because a response is not required.

The Response page looks as follows:



What You See on the Oracle HCM Cloud Trigger Response Page

Select the business object for the integration to send as a response document to the Oracle HCM Cloud application.

The following types of responses are available.

- Immediate: A synchronous response is required (See Table 5-7 for instructions)
- Delayed: An asynchronous response is required (See Table 5-8 for instructions)
- None: No response is required (See Table 5-9 for instructions)

The following table describes the fields available if an immediate (synchronous) response is required.

Table 5-7 Response Type — Immediate (Synchronous) Response is Required

Element	Description
Response Type	Select **Immediate** for the Oracle HCM Cloud application to wait until a response is received from the integration. This is also known as the request and response message exchange pattern. This is the default selection.
Filter by object name	Type the initial letters to filter the display of business objects.
Select a Business Object	Select the business object to receive from the Oracle HCM Cloud application as a response. A description of the selected business object is displayed below this list.
Name	Displays the name of the selected business object.

Table 5-7 (Cont.) Response Type — Immediate (Synchronous) Response is Required

Element	Description
Description	Displays the description of the selected business object.

The following table describes the fields available if a delayed (asynchronous) callback response is required. You can configure a successful callback response, a failed callback response, or both.

Table 5-8 Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Response Type	Select **Delayed** to configure a successful callback response, a failed callback response, or both. This enables you to configure the operation and business objects that you want the Oracle HCM Cloud application to process as part of a successful callback response, failed callback response, or both.
Successful Response/Failed Response	Select the type of callback to configure. After configuring one type of callback (for example, successful), you can configure the other type (for example, failed). • **Successful Response**: Select to configure the operation and business objects that you want the Oracle HCM Cloud application to process as part of a successful callback response sent by the integration. • **Failed Response**: Select to configure the operation and business objects that you want the Oracle HCM Cloud application to process as part of a failed callback response sent by the integration.
Select the operation to perform on the business object	Select the operation to perform on the business object.
Life Cycle	Displays the current state of the selected business document. Active indicates the business document is available for use. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Name	Displays the name of the selected business object.

Table 5-8 (Cont.) Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Description	Displays the description of the selected business object.

The following table describes the fields available if no response is required.

Table 5-9 None — No Response is Required

Element	Description
Response Type	Select **None**.
Select a Business Object	If you select **None**, this section is hidden.

Configuring Oracle HCM Cloud Invoke Operation Properties

Enter the Oracle HCM Cloud invoke operation values for your integration.

Topics

- What You Can Do from the Oracle HCM Cloud Target Operations Page
- What You See on the Oracle HCM Cloud Target Operations Page

What You Can Do from the Oracle HCM Cloud Invoke Operations Page

You can configure the following invoke operation values for the Oracle HCM Cloud.

- Select the business object or service.
- Select the operation to perform on the selected business object or service.

What You See on the Oracle HCM Cloud Invoke Operations Page

The following table describes the key information on the Oracle HCM Cloud invoke Operations page.

Element	Description
Browse by	Select to browse by business object or service. There is a one-to-one correspondence between the business object and service. The service acts on the business document. • **Business Objects**: Select to browse a list of available business objects. • **Services**: Select to browse a list of available services.
Filter by object name (displayed if **Business Objects** is selected)	Type the initial letters to filter the display of business objects.

Element	Description
Select a Business Object (displayed if **Business Objects** is selected)	Select the business object to use.
Filter by service (displayed if **Services** is selected)	Type the initial letters to filter the display of services.
Select a Service (displayed if **Services** is selected)	Select the service to use.
Select the operation to perform on the business object or service	Select the operation to perform on the selected business object or service.
Life Cycle	Displays the state of the selected business object or service. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Name	Displays the name of the selected business object or service.
Description	Displays the description of the selected business object or service.

Configuring Salesforce Properties

The Salesforce Adapter enables you to create integrations with a Salesforce application.

The following sections describe the wizard pages that guide you through configuration of the Salesforce adapter as a trigger and invoke in an integration. Postconfiguration tasks are also provided.

Topic

- Understanding Salesforce Cloud Constraints
- Configuring Basic Information Properties
- Configuring Salesforce Trigger Outbound Messaging Properties
- Configuring Salesforce Trigger Response Properties
- Configuring Salesforce Trigger Callback Response Properties
- Configuring Salesforce Target Operations Properties
- Configuring Salesforce Target Headers Properties
- Reviewing Configuration Values on the Summary Page
- Performing Salesforce Postconfiguration Tasks

For more information about Salesforce Cloud, see Salesforce Cloud.

Understanding Salesforce Cloud Constraints

You must be aware of the following constraints before configuring the Salesforce Adapter.

- The Salesforce Adapter uses the SalesForceDotCom (SFDC) API for all activities. Therefore, it is subject to any Salesforce API limitations. The limitations are defined in the Salesforce Limits Quick Reference Guide.
- Not all the Push Topic queries are supported by Salesforce. See Supported Push Topic Queries and Unsupported PushTopic Queries.
- Client applications must adhere to Salesforce's SOAP API support policy and backward compatibility terms. These terms are available at SFDC SOAP API Support Policy.

Configuring Salesforce Trigger Outbound Messaging Properties

Enter the Salesforce trigger outbound messaging values for your integration.

Topics

What You Can Do from the Salesforce Cloud Trigger Outbound Messaging Properties Page

You can configure the trigger outbound messaging WSDL for the Salesforce Cloud adapter.

This process consists of several steps:

- The outbound message consists of a workflow, approval, or milestone action that sends your specified information to your specified endpoint. You configure outbound messaging in the Salesforce setup menu. Afterward, you configure the endpoint.

  To create a workflow rule:

  1. Log in to your Salesforce account and go to **Setup**.
  2. Under the **App Setup** menu, expand **Create**, followed by **Workflow & Approvals**.
  3. Select a workflow rule or approval process as per your integration requirement.
  4. Click **Create New**, provide the required information in the following wizards, and click **Save**.
     a. For the workflow rule, click **Edit** under the **Workflow Action** menu followed by **Add Workflow Action**, and then **New Outbound Message**.

b. For the approval process, click **Add New** (you can select for one or more actions including **Submission**, **Approval**, **Rejection**, and **Recall**) followed by **New Outbound Message**.

Outbound messaging WSDLs associated with approval processes or entitlement processes are also supported and consumed by the adapter.

- Generate the Salesforce outbound messaging WSDL at www.salesforce.com, then select the invoke outbound messaging WSDL you created to receive outbound message notifications from the Salesforce application. For instructions, see What You See on the Salesforce Cloud Trigger Outbound Messaging Properties.

What You See on the Salesforce Cloud Trigger Outbound Messaging Properties

The following table describes the key information on the trigger Outbound Messaging page.

Element	Description
Select the Outbound Messaging WSDL	Generate and then select the invoke Salesforce outbound messaging WSDL to receive outbound message notifications from the Salesforce application. **Note**: You must first create a workflow rule as described in What You Can Do from the Salesforce Cloud Trigger Outbound Messaging Properties Page. Outbound messaging WSDLs associated with approval processes or entitlement processes are also supported and consumed by the adapter. To generate and then select the invoke Salesforce outbound messaging WSDL: 1. Log in to your Salesforce account and go to **Setup** > **Outbound Messages**. 2. Select the required object, and click **Next**. 3. Enter other required details (in the **Endpoint URL** field, enter a dummy URL), and click **Save**. 4. Click **Generate WSDL** to download the WSDL. 5. Return to this wizard page and browse for the generated WSDL. 6. Activate the integration and copy the endpoint URL from the integration information icon. 7. Go to the **Outbound Messaging** section at www.salesforce.com and replace the dummy URL you entered in Step 3 with the real endpoint URL.

Configuring Salesforce Trigger Response Properties

Enter the Salesforce trigger response values for your integration.

Topics

What You Can Do from the Salesforce Trigger Response Page

You can configure the response parameters for the Salesforce Adapter.

You can select the type of callback response to send as a response document from the integration flow to the Salesforce Cloud application.

- Configure to send no callback response.
- Configure the operation and business objects to use for a successful callback response.
- Configure the operation and business objects for a callback response for a failed integration flow.

What You See on the Salesforce Cloud Trigger Response Page

The following table describes the key information on the trigger Response page.

Element	Description
Send a response	Deselect if no callback response is required.
Configuration a Successful Callback Response	Select **Configure** to configure the operation and business objects to use for a successful callback response.
Configuration a Failure Callback Response	Select **Configure** to configure the operation and business objects for a callback response for a failed integration flow. The option to configure a failure callback response gets enabled only after the configuration of successful callback response.
Edit	Click to edit the operation or business objects of a successful or failed callback response. This button is displayed after you configure a successful callback response, a failed callback response, or both.
Header	Click to configure the header properties for the selected operation. The headers available for configuration are based on the type of operation you selected. This button is displayed after you configure a successful callback response, a failed callback response, or both.

Element	Description
Reset	Click to reset the operation and header configuration to the default values. This button is displayed after you configure a successful callback response, a failed callback response, or both.

Configuring Salesforce Trigger Callback Response Properties

Enter the Salesforce trigger callback response values for your integration.

Topics

What You Can Do from the Salesforce Trigger Callback Response Page

You can configure the callback response parameters for the Salesforce Adapter.

- Configure the operation and business objects to use for a successful callback response.
- Configure the operation and business objects for a callback response for a failed integration flow.

What You See on the Salesforce Cloud Trigger Callback Response Page

The following table describes the key information on the trigger Callback Response page.

Element	Description
Select an Operation Type	Select the type of operation to perform on the business objects in a Salesforce Cloud application: • **CORE**: Displays the following selections: **ConvertLead**, **Merge**, **Undelete**, or **Upsert**. • **CRUD**: Represents the create, read, update, delete, or destroy operations to perform on Salesforce Cloud business objects. Each letter maps to a standard SQL statement, HTTP method, or DDS operation. Select the CRUD operation to perform on the business object: **Create**, **Delete**, or **Update**. • **MISC**: Represents the set of specialized task operations to perform in the Salesforce Cloud application.

Element	Description
Filter by object name	Enter the initial letters of an object name to display a range of objects. You can also enter an asterisk (*) after the query in the search field (for example, to search for all objects starting with `Acc`, enter `Acc*`). You can also select a filter type: • **All**: Displays all objects. • **Custom**: Displays objects you created. The naming convention is a combination of the object name appended with `_c`. • **Standard**: Displays business objects delivered as part of the Salesforce Cloud application.
Select Business Objects (Salesforce API *version*)	Select a single business object or multiple business objects from the Salesforce Cloud application. The selected operation acts upon these business objects. When you complete invoke operation configuration, the selected operation and business objects are defined in the integration-centric WSDL file.
Your Selected Business Objects	Displays the business objects you selected.

Configuring Salesforce Invoke Operation Properties

Enter the Salesforce invoke operation values for your integration.

Topics

- What You Can Do from the Salesforce Target Operations Page
- What You See on the Salesforce Target Operation Page

What You Can Do from the Salesforce Cloud Invoke Operations Page

You can configure the following invoke operations values for Salesforce Cloud.

- Select either of the following operation types:
  1. CRUD
  2. Salesforce Object Query Language (SOQL) or Salesforce Object Search Language (SOSL) query
- Select the business objects.
- Specify the SOQL/SOSL query.

What You See on the Salesforce Invoke Operations Page

The following table describes the key information on the Salesforce Cloud invoke Operations page.

Element	Description

Select an Operation Type	Select the type of operation to perform:
	• **CORE** : Represents all core operations supported by the Salesforce application.
	• **CRUD**: Represents the create, read, update, delete, or destroy operation to perform on Salesforce business objects. Each letter maps to a standard SQL statement, HTTP method, or DDS operation. Select the CRUD operation to perform: **Create**, **Delete**, **Retrieve**, or **Update**.
	• **MISC**: Represents specialized task operations (such as fetching user information associated with the current session) in the Salesforce application.
	• **SOSL/SOQL**: Select to enter a Salesforce Object Query Language (SOQL) or Salesforce Object Search Language (SOSL) query to send as a request to the Salesforce application. The following operations are available:
	– **query**: Executes a query against specific criteria and returns data matching that criteria. Only records not deleted from your Salesforce application account are returned.
	– **queryAll**: Returns the same data as the **query** operation, along with deleted records present in the recycle bin.
	– **search**: Returns records from the Salesforce application. You can specify binding parameters to dynamically provide a search string as input to your search operation.
	If you select this option, the page is refreshed to display a field for entering an SOQL or SOSL query to send for validation:
	– **Query Statement**: Enter a valid query statement. SOQL statements evaluate to a list of sObjects, a single sObject, or an integer for `count` method queries. The following examples are provided:
	`"SELECT Id FROM Contact WHERE Name LIKE 'A%' AND MailingCity = 'California'"`
	`SELECT COUNT() FROM Contact`
	SOSL statements evaluate to a list of sObjects, where each list contains the search results for a particular sObject type. For example:
	`"SELECT a.name, a.id, a.accountNumber, c.name from Contact c, c.Account"`
	– **Binding Parameters**: Displays any parameters included in the query. For example, `orgId` is a parameter in the following query:
	`SELECT a.name, a.id, a.accountNumber, c.name from Contact c, c.Account a WHERE a.name = "&orgId"`
	This query displays a binding parameters text box in which to enter a test value for `orgId`.
	– **Test My Query**. Click to validate the query against the Salesforce application. Query results are displayed. If errors occur, you receive results about how to correct the query.

Element	Description
Filter By Object Name	Type the initial letters to filter the display of business objects. You can also select a filter type: • **All**: Displays all objects. • **Custom**: Displays objects you created. Custom business objects are appended with "_c." • **Standard**: Business objects delivered as part of the Salesforce application.
Select Business Objects	Select a single or multiple business objects to include in the operation. You can select up to ten objects for one operation.

Configuring Salesforce Invoke Header Properties

Enter the Salesforce invoke header values for your integration.

Topics

- What You Can Do from the Salesforce Target Headers Page
- What You See on the Salesforce Target Headers Page

What You Can Do from the Salesforce Cloud Invoke Headers Page

You can configure the invoke header properties for Salesforce Cloud.

What You See on the Salesforce Invoke Headers Page

The following table describes the key information on the Salesforce Cloud invoke Headers page.

The headers available for configuration are based on the operation you selected on the invoke Operations page. There are two types of headers:

- Request headers are sent with the request message to the Salesforce application.
- Response headers are received with the response message sent from the Salesforce application.

For more information about these header properties, visit www.salesforce.com and specify the specific name of the property in the search utility.

Element	Description
AllOrNoneHeader (request header)	Specifies the transactional behavior for Salesforce application operations. If you set **AllorNone** to **true**, the call to the Salesforce application is committed only if it completes without any errors. Otherwise, it is rolled back. The default behavior is to commit partial records without any error.

Element	Description
AllowFieldTruncationHeader (request header)	Specifies the truncation behavior for the following fields (each are string data types): • `anyType` • `email` • `picklist` • `encryptedstring` • `textarea` • `mulitpicklist` • `phone` • `string` Set **allowFieldTruncation** to one of the following values: • **True**: If you enter a value of 25 characters in a field of 20 characters, the first 20 records are inserted into the field and the transaction is successful. • **False**: If you enter a value of 25 characters in a field of 20 characters, an error is thrown and the transaction does not commit.
AssignmentRuleHeader (request header)	Specifies the assignment rule to use when creating or updating an account, case, or lead. The assignment rule can be active or inactive. The ID is retrieved by querying the `AssignmentRule` object. If the ID is specified, you do not need to specify the **useDefaultRule** value. • **assignmentRuleId**: The ID of the assignment rule to use. The ID is not validated by the Salesforce Cloud application, whether or not it exists. Validation occurs during runtime. • **useDefaultRule**: If set to **true**, the default (active) assignment rule is used. If set to **false**, the default (active) assignment rule is not used.

Element	Description
EmailHeader (request header)	Specifies whether or not to send a notification email. You can set the following properties: • **triggerAutoResponseEmail** – **true**: Triggers automatic response rules for leads and cases. – **false**: Automatic response rules for leads and cases are not triggered. • **triggerOtherEmail** – **true**: The email is triggered outside the organization. – **false**: The email is not triggered outside the organization. • **triggerUserEmail** – **true**: The email is triggered and sent to users in the organization. This email is triggered by a number of events such as adding comments to a case or updating a task. – **false**: The email is not triggered and sent to users in the organization.
DebuggingHeader (request header)	Specify the debugging log level. The following log levels are supported: • **NONE** (least verbose) • **DEBUGONLY** • **DB** • **PROFILING** • **CALLOUT** • **DETAIL** (most verbose)
MruHeader (request header)	The Salesforce application shows the most recently used (MRU) items. In API version 7.0 or later, the list is not updated by itself. Use **MruHeader** to update the list. Using this header can negatively impact performance. Set **updateMru** to one of the following values: • **true**: The list of MRU items is updated in the Salesforce application. • **false**: The list of most recently used items is not updated in the Salesforce application.
PackageVersionHeader (request header)	Specifies the package version for any installed package. The package version identifies the components in a package. The package version follows the format `majorNumber.minorNumber.patchNumber` (for example, `3.4.5`, where `3` refers to `majorNumber`, `4` refers to `minorNumber`, and `5` refers to `patchNumber`) .

Element	Description
QueryOptions (request header)	Specifies the batch size for queries. The default value is `500`, the minimum value is `200`, and the maximum value is `2000`.
DebuggingInfo (response header)	This information is only returned if the **debugLevel** request header is sent with the request payload to the Salesforce application.
LimitInfoHeader (response header)	Provides information about the limitations of API calls on a per-day basis for the organization. • **current**: The number of calls already used in the organization. • **Limit**: The organization's limit for the specified limit type. • **Type**: The limit information type specified in the header `API REQUESTS` (contains limit information about API calls for the organization).

Performing Salesforce Postconfiguration Tasks

After activating your integration, you must update the outbound message for the Salesforce adapter to send messages to Oracle Integration Cloud Service. This section describes how to activate a workflow rule.

1. Open the Salesforce application.
2. Scroll down and click **Workflow Rules.**
3. In the **Workflow Rules** panel, click the workflow rule.
4. Scroll down to the **Immediate Workflow Actions** section and click the outbound message.
5. In the **Outbound Message** panel, click **Edit**.
6. In the **Edit Outbound Message** panel, enter the endpoint URL from the **Integration Details** icon for the integration.
7. In the **Edit Outbound Message** panel, click **Save**.

   The **Outbound Message** panel is displayed.
8. In the **Outbound Message** panel, scroll down and find the **Workflow Rules Using This Outbound Message** section.
9. Click the workflow link.

   The **Workflow Rule** panel is displayed.
10. In the **Workflow Rule** panel, click **Activate**.

Your workflow is activated. The Salesforce application starts sending messages to the integration endpoint URL created when you activated the integration.

Configuring Oracle ERP Cloud Properties

The Oracle ERP Cloud Adapter enables you to create integrations with an Oracle ERP Cloud application.

The following sections describe the wizard pages that guide you through configuration of the Oracle ERP Cloud Adapter as a trigger and invoke in an integration.

Topics

- Configuring Basic Information Properties
- Configuring Oracle ERP Cloud Source Request Properties
- Configuring Oracle ERP Cloud Source Response Properties
- Configuring Oracle ERP Cloud Target Operation Properties
- Reviewing Configuration Values on the Summary Page

For more information about Oracle ERP Cloud Adapter, see Configuring Oracle ERP Cloud Properties.

Configuring Oracle ERP Cloud Trigger Request Properties

Enter the Oracle ERP Cloud connection trigger request values for your integration. The values you specify start the integration.

Topics

- What You Can Do from the Oracle ERP Cloud Trigger Request Page
- What You See on the Oracle ERP Cloud Trigger Request Page

What You Can Do from the Oracle ERP Cloud Trigger Request Page

You can select the following trigger request values for the Oracle ERP Cloud application.

Select the specific type to receive as a request from Oracle ERP Cloud. Your ability to select either a business object or event subscription is based on the content of the WSDL file (for business objects) or event catalog URL (for event subscriptions) you specified during Oracle ERP Cloud Adapter configuration.

- Select to receive a business object as a request from the Oracle ERP Cloud application. This selection invokes the integration.
- Select to receive an event subscription raised by the Oracle ERP Cloud application as a request from Oracle ERP Cloud. This selection invokes the integration.

What You See on the Oracle ERP Cloud Trigger Request Page

The following table describes the key information on the Oracle ERP Cloud trigger Request page.

Element	Description
Configure a Request	Select the request type appropriate to your integration. The fields that are displayed below are based on the request type that you select. • **With Business Objects**: Select to display a list of business objects. • **With Business Events**: Select to display a list of business events.
Select a Business Object (is displayed if **With Business Objects** is selected)	Select the business object from the Oracle ERP Cloud application to receive as a request that starts the integration.
Business Event For Subscription (is displayed if **With Business Events** is selected)	Select the event subscription from the Oracle ERP Cloud application. This event is received as a request that starts the integration.
Filter Expr for *Business_Event_Name* (is displayed if **With Business Events** is selected)	Enter an event condition filter expression. A filter expression specifies that the contents (payload or headers) of a message be analyzed before any event is sent. For example, you can apply a filter expression that specifies that an event be sent only if the message includes a customer ID. When the expression logic is satisfied, the event is accepted for delivery to the integration.
Filter by object name or **Filter By Event Name**	Type the initial letters of the name to filter the display of business objects or event subscriptions.

Configuring Oracle ERP Cloud Trigger Response Properties

Enter the Oracle ERP Cloud trigger response values for your integration.

Topics

What You Can Do from the Oracle ERP Cloud Source Response Page

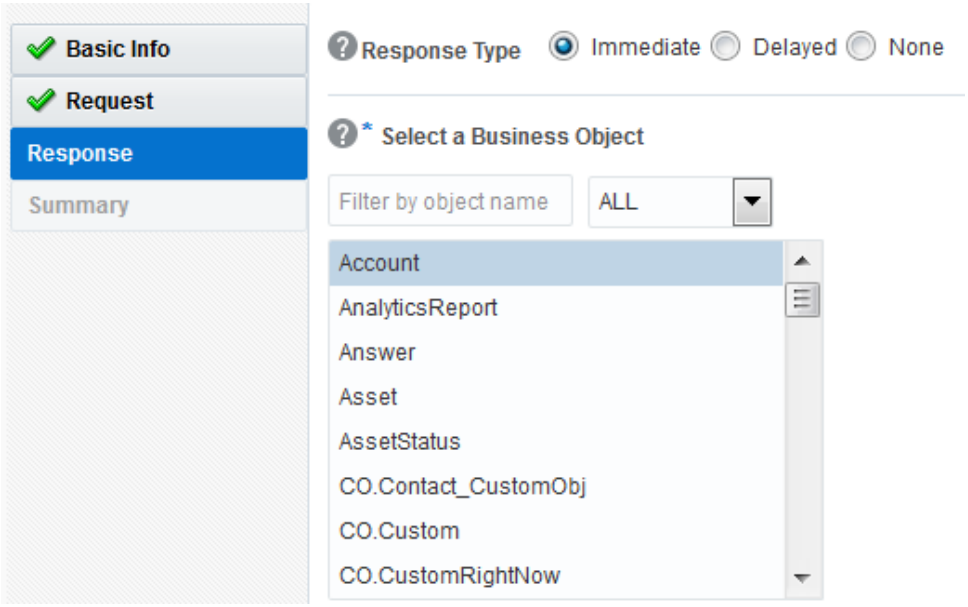
You can configure the operation and business object that comprise the response type for the Oracle ERP Cloud application.

- Immediate (synchronous) response: A response business object is immediately returned as output. You select **Immediate** as the response type on the Response page and select the business object as part of the response to the client.
- Delayed (asynchronous) response: A callback service to which to route the callback is exposed. You select **Delayed** as the response type on the Response

page and select the operation and business object that comprise a successful callback response, a failed callback response, or both.

- No response is required: You select **None** on the Response page because a response is not required.

The Response page looks as follows:



What You See on the Oracle ERP Cloud Trigger Response Page

Select the business object for the integration to send as a response document to the Oracle ERP Cloud application. The following types of responses are available.

Select the type of response appropriate to your business requirements:

- Immediate: A synchronous response is required (See Table 5-10 for instructions)
- Delayed: An asynchronous response is required (See Table 5-11 for instructions)
- None: No response is required (See Table 5-12 for instructions)

The following table describes the fields available if an immediate (synchronous) response is required.

Table 5-10 Response Type — Immediate (Synchronous) Response is Required

Element	Description
Immediate Response	Select **Immediate** for the Oracle ERP Cloud application to wait until a response is received from the integration. This is also known as the request and response message exchange pattern. This is the default selection.
Filter by object name	Enter the initial letters to filter the display of business objects.

Table 5-10 (Cont.) Response Type — Immediate (Synchronous) Response is Required

Element	Description
Select a Business Object	Select the business object to receive from the Oracle ERP Cloud application as a response. A description of the selected business object is displayed below this list.
Name	Displays the name of the selected business object.
Description	Displays the description of the selected business object.

The following table describes the fields available if a delayed (asynchronous) callback response is required. You can configure a successful callback response, a failed callback response, or both.

Table 5-11 Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Response Type	Select **Delayed** to configure a successful callback response, a failed callback response, or both. This enables you to configure the operation and business objects that you want the Oracle ERP Cloud application to process as part of a successful callback response, failed callback response, or both.
Successful Response/Failed Response	Select the type of callback to configure. After configuring one type of callback (for example, successful), you can configure the other type (for example, failed). • **Successful Response**: Select to configure the operation and business objects that you want the Oracle ERP Cloud application to process as part of a successful callback response sent by the integration. • **Failed Response**: Select to configure the operation and business objects that you want the Oracle ERP Cloud application to process as part of a failed callback response sent by the integration.
Select the operation to perform on the business object	Select the operation to perform on the business object.

Table 5-11 (Cont.) Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Life Cycle	Displays the current state of the selected business document. Active indicates the business document is available for use. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Name	Displays the name of the selected business object.
Description	Displays the description of the selected business object.

The following table describes the fields available if no response is required.

Table 5-12 Response Type — None

Element	Description
Response Type	Select **None**.
Select a Business Object	If you select **None**, this section is hidden.

Configuring Oracle ERP Cloud Invoke Operation Properties

Enter the Oracle ERP Cloud invoke operation values for your integration.

Topics

- What You Can Do from the Oracle ERP Cloud Invoke Operations Page
- What You See on the Oracle ERP Cloud Invoke Operations Page

What You Can Do from the Oracle ERP Cloud Invoke Operations Page

You can configure the following invoke operation values for the Oracle ERP Cloud.

- Select the business object or service.
- Select the operation to perform on the selected business object or service.

What You See on the Oracle ERP Cloud Invoke Operations Page

The following table describes the key information on the Oracle ERP Cloud invoke Operations page.

Element	Description
Browse by	Select to browse by business object or service. There is a one-to-one correspondence between the business object and service. The service acts on the business document. • **Business Objects**: Select to browse a list of available business objects. • **Services**: Select to browse a list of available services.
Filter by object name (displayed if **Business Objects** is selected)	Type the initial letters to filter the display of business objects.
Select a Business Object (displayed if **Business Objects** is selected)	Select the business object to use.
Filter by service (displayed if **Services** is selected)	Type the initial letters to filter the display of services.
Select a Service (displayed if **Services** is selected)	Select the service to use.
Select the operation to perform on the business object or service	Select the operation to perform on the selected business object or service.
Life Cycle	Displays the state of the selected business object or service. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Name	Displays the name of the selected business object or service.
Description	Displays the description of the selected business object or service.

Configuring Oracle CPQ Cloud Properties

The Oracle CPQ Cloud Adapter enables you to create integrations with an Oracle CPQ Cloud application.

The following sections describe the wizard pages that guide you through configuration of the Oracle CPQ Cloud Adapter as a trigger and invoke in an integration.

Topics

- Configuring Oracle CPQ Target Operation Properties
- Reviewing Configuration Values on the Summary Page

For more information about Oracle Sales Cloud, see Oracle CPQ Cloud.

Configuring Oracle CPQ Trigger Request Properties

View the Oracle CPQ Cloud trigger request values for your integration.

Topics

- What You Can Do from the Oracle CPQ Trigger Request Page
- What You See on the Oracle CPQ Trigger Request Page

What You Can Do from the Oracle CPQ Trigger Request Page

You can view the Transaction business object to receive from the Oracle CPQ application as a request document to start the integration flow.

What You See on the Oracle CPQ Trigger Request Page

The following table describes the key information on the trigger Oracle CPQ Request page.

Element	Description
Business Object	View the transaction business object. This object is from the CPQ commerce process. This is the business object that you receive from the Oracle CPQ application as a request document to start this integration flow. This business object is automatically selected based on the content of the WSDL file you specified when creating the Oracle CPQ connection.

Configuring Oracle CPQ Trigger Response Properties

View the Oracle CPQ Cloud trigger response values for your integration.

Topics

- What You Can Do from the Oracle CPQ Trigger Response Page
- What You See on the Oracle CPQ Trigger Response Page

What You Can Do from the Oracle CPQ Trigger Response Page

You can view the trigger response properties for Oracle CPQ Cloud.

- The response business object sent from the integration flow to the Oracle CPQ application
- The synchronous response selection for the response type
- The transaction business object used in this connection

What You See on the Oracle CPQ Trigger Response Page

The following table describes the key information on the trigger Oracle CPQ Response page.

Element	Description
Send a Response	Indicates that a response business object is sent from the integration flow to the Oracle CPQ application. This option is automatically configured and cannot be changed.
Response Type	Indicates that the business object is sent back synchronously to the source application. This option is automatically configured and cannot be changed.
Business Object	Displays the Transaction business object. This object is from a CPQ Commerce process. This business object is automatically selected based on the content of the WSDL file you specified when creating the Oracle CPQ connection.

Configuring Oracle CPQ Invoke Operation Properties

View and configure the Oracle CPQ Cloud invoke operation values for your integration.

Topics

- What You Can Do from the Oracle CPQ Invoke Operations Page
- What You See on the Oracle CPQ Invoke Operations Page

What You Can Do from the Oracle CPQ Invoke Operations Page

You can view and configure the invoke operation properties for Oracle CPQ Cloud.

- View the Oracle CPQ API version being used.
- Select the type of operation for the connection to perform.
- View the transaction business object used in this connection.

What You See on the Oracle CPQ Invoke Operations Page

The following table describes the key information on the invoke Oracle CPQ Operations page.

Element	Description
CPQ API Version 2	Displays Commerce.

Element	Description
Select an Operation	Select an operation. These operations come from the WSDL you specified when creating the connection. • **Add a Transaction**: Adds a new item to an existing transaction performing the Add from Catalog action. The input parameters include the process, the document, the action on the document, and the items to be added. • **Create Transaction**: Supports the creation of a commerce Transaction without line items and transactions with nonconfigurable line items from a specified process. For all transactions required to be created with configurable line items, the Configuration SOAP API must be invoked. This action returns the transaction ID. • **Export File Attachments**: Exports a file attachment using one of two methods to stream the data through SOAP: – Inline base64 content in a SOAP message – Binary stream with MIME containers through an MTOM transmission These methods read and write multiple attributes at once per transaction. This API can only be used by full-access users with the Modify Users permission. There are two modes available for use: – Content: Retrieves the content of the attached fields. – Metadata: Retrieves the file information or metadata for the referenced fields. • **Get Transaction**: Returns the complete Transaction XML content for the given Transaction ID. • **Import File Attachments**: Imports a file attachment using one of two methods to stream the data through SOAP: – Inline base64 content in a SOAP message – Binary stream with MIME containers through an MTOM transmission These methods read and write multiple attributes at once per transaction. This API can only be used by full-access users with the Modify Users permission. There are two modes available for use: – Update: Attaches a file or set of files to the transaction – Delete: Removes a file from the file attachment attribute • **Remove from Transaction**: Removes an item from an existing transaction by performing the Remove Line Items action. The input parameters include the document ID, document number, process name, and document name. • **Update Transaction**: Updates an existing transaction by performing the Modify and Auto-fill actions. The Create Document action is not supported.
Business Objects	Displays the Transaction business object.

Configuring SOAP Adapter Properties

The SOAP Adapter enables you to create integrations with SOAP endpoints.

The following sections describe the wizard pages that guide you through configuration of the SOAP Adapter as a trigger and invoke in an integration.

Topics

For more information about the SOAP Adapter, see SOAP Adapter.

Configuring SOAP Adapter Trigger Operation Properties

Enter the port type and operation for the SOAP Adapter.

Topics

What You Can Do from the Oracle SOAP Adapter Trigger Operations Page

You can configure the following trigger operations for the Oracle SOAP Adapter. If your WSDL includes only a single service, port type, and operation, they are automatically selected. If the WSDL includes multiple services and port types, then select the ones to use in your integration.

- Select the port type
- Select the operation

Based on the selected values, other objects such as the request object, response object, and fault object may also be automatically displayed.

What You See on the Oracle SOAP Adapter Trigger Operations Page

The following table describes the key information on the Oracle SOAP Adapter trigger Operations page.

Element	Description
Select the Port Type	Select the port type (if your WSDL includes multiple port types).
Select the Operation	Select the operation (if your WSDL includes multiple operations).

Configuring SOAP Adapter Invoke Operation Properties

Enter the port, operation, and service for the SOAP Adapter.

Topics

What You Can Do from the SOAP Adapter Invoke Operations Page

You can configure the following invoke operations for the SOAP Adapter. If the WSDL file you specified during SOAP Adapter connectivity configuration includes only a single service, port type, or operation, they are automatically selected for use. If the WSDL included multiple services, port types, or operations, then select the ones to use in this integration.

- Select the service.
- Select the port.
- Select the operation.

Based on the selected values, other objects such as the request object, response object, and fault object may also be automatically displayed.

What You See on the SOAP Adapter Invoke Operations Page

The following table describes the key information on the SOAP Adapter invoke Operations page.

Element	Description
Selected Service	Select the service (if your WSDL includes multiple services).
Select the Port	Select the port (if your WSDL includes multiple ports).
Select the Operation	Select the operation (if your WSDL includes multiple operations).

Configuring REST Adapter Properties

The REST Adapter enables you to create integrations with an REST endpoint.

The following sections describe the wizard pages that guide you through configuration of the REST Adapter as a trigger and invoke connection in an integration.

Topics

For more information about the REST adapter, see REST Adapter.

Configuring REST Adapter Basic Information Properties

Enter the REST Adapter user name, description, relative resource URI, and endpoint action. You can also select to add query and template parameters or configure a request and/or response for the endpoint.

Topics

What You Can Do from the REST Adapter Basic Info Page

You can specify the following values on the trigger (source) or invoke (target) REST Adapter Basic Info page. The REST Adapter Basic Info page is the initial wizard page that is displayed when you drag a REST Adapter to the trigger (source) or invoke (target) area.

- Specify a meaningful name.
- Add a description of endpoint responsibilities.
- Specify the relative resource URI of the endpoint.
- Select the HTTP action for the endpoint to perform.
- Add endpoint parameters, configure the endpoint request payload, or configure the endpoint to receive the response. Based on the HTTP action selected, you can configure multiple options.
- Select to configure standard and custom HTTP request and response headers.
- Select to configure cross origin resource sharing (CORS).

What You See on the REST Adapter Basic Info Page

The following table describes the key information on the REST Adapter Basic Info page.

Element	Description
What do you want to call your endpoint?	Provide a meaningful name so that others can understand the connection. For example, if you are creating a source Oracle REST connection, you may want to name it `ExposeFlowAsRESTResource`. You can include English alphabetic characters, numbers, underscores, and dashes in the name. You cannot include the following: • Blank spaces (for example, `My REST Connection`) • Special characters (for example, `#;83& or res(t)4)` • Multibyte characters

Element	Description
What does this endpoint do?	Enter an optional description of the connection's responsibilities (for example,`This inbound REST connection exposes this integration flow as a REST resource`).
What is the endpoint's relative resource URI?	Specify the relative path associated with the resource. The path can contain template parameters specified with curly braces (for example, `{order-id}`). A resource is any source of specific information that can be addressed. The resource path follows a fixed, prefixed URL appended with the specified relative path. By default, the URL is prefixed with the following path: `http://`*`host`*`:`*`port`*`/integration/flowapi/rest/`*`INTEGRATION_NAME`* For example, if the integration name is `ExposeFlowAsRESTResource`, the URL becomes: `http://`*`host`*`:port/integration/flowapi/rest/EXPOSEFLOWASRESTRESOURCE` You can override the URL, except for the fixed part at the beginning: *`host`*`:`*`port`*`/integrations`
What action does the endpoint perform?	Select a single HTTP action (method) for the endpoint to perform: • **GET**: Retrieves (reads) information (for example, makes queries). If you select this option, you cannot configure a request payload for this endpoint. • **PUT**: Updates information. • **POST**: Creates information. • **DELETE**: Deletes information. If you select this option, you cannot configure a request payload for this endpoint.

Element	Description
Based on your selections, you can add parameters or configure a request and/or response for this endpoint	Select the options that you want to configure: • **Add and review parameters for this endpoint**: Click to specify the query parameters and view the template request parameters created as part of the resource URI for this endpoint. If you select this option and click **Next**, the Request Parameters page is displayed. • **Configure a request payload for this endpoint**: Click to configure the request payload for this endpoint, including specifying the schema location and payload type with which you want the endpoint to reply. You can also select this option if you want to include an attachment with the inbound request. If you select this option and click **Next**, the Request page is displayed. • **Configure this endpoint to receive the response**: Click to configure the response payload for this endpoint, including specifying the schema location and payload type that you want the endpoint to receive. If you select this option and click **Next**, the Response page is displayed.

Configuring REST Adapter Request Parameters Properties

Enter the REST Adapter request parameters for this endpoint.

Topics

What You Can Do from the REST Adapter Request Parameters Page

You can view and configure the following request parameters on the trigger or invoke REST Adapter Request Parameters page.

- Specify query parameters.
- View the template parameters in the relative resource URI.

What You See on the REST Adapter Request Parameters Page

The following table describes the key information on the REST Adapter Request Parameters page.

Element	Description
Resource URI	Displays the endpoint relative resource URI entered on the Basic Info page.
Specify Query Parameters	Specify query parameters for the REST endpoint. Click the **Add** icon to display a row for entering the parameter name and selecting its data type. For example, specify `state` and select a data type of **string**. Click the **Delete** icon to delete a selected row.
Template Parameters	Displays the template parameters in the relative resource URI. Template parameters are based on details you specified on the Basic Info page and cannot be edited. Template parameters must be defined as part of a path with curly braces around them. For example, the URL `default/customers/{cust-id}/{ship-id` has `cust-id` and `ship-id` template parameters. You can change the data type for the parameters. **Note**: Both query and template parameter values are displayed in the mapper through use of an XML element. For query, the XML element name is `query_parameters` For template, the XML element name is `template_parameters`.

Configuring REST Adapter Request Properties

Enter the REST Adapter request payload details for the endpoint.

Topics

- What You Can Do from the REST Adapter Request Page
- What You See on the REST Adapter Request Page

What You Can Do from the REST Adapter Request Page

You can configure the following request payload details on the trigger or invoke REST Adapter Request page.

- Select to include attachments with inbound requests
- Specify the schema or JSON sample request payload file location
- Select the type of payload with which you want the endpoint to reply

What You See on the REST Adapter Request Page

The following table describes the key information on the REST Adapter Request page.

Element	Description
Select the attachment processing options	Select the multipart attachment type to include with the inbound request. • **Accept attachments from request**: Select for the REST endpoint to process attachments from the inbound multipart request. This selection refreshes the page to display the **Select the type of payload that you want the endpoint to receive** field at the bottom of the page. • **Request is HTML form**: Select for the REST endpoint to accept to configure an HTML form. You must first select the **Accept attachments from request** option before you can select this option. This selection assumes that the media type is multipart/form-data.
Select the request payload file	Select the type of request payload file to use. The request payload body must be defined by the XSD element that defines the structure of this representation. • **XML Schema** • **JSON Sample** (Creates a sample JSON file.)
Schema Location	Click **Browse** to select the request schema file to use.
Element	Select the element that defines the payload structure. This field is not displayed until you import the request payload file. Once you browse for and select the schema or JSON sample file, the schema is displayed automatically. It also displays a combination box that selects the root element by default.
Select the type of payload with which you want the endpoint to reply	Select the payload type with which you want the request payload to reply. • **None**: Select if no payload type is required. • **XML**: Displays the payload in XML format. • **JSON**: Displays the payload in JavaScript Object Notation (JSON) format. • **URL-encoded**: Displays the payload in URL-encoded format. • **Other Media Type**: Select to display the payload in another format (for example, `application/oracle.cloud+json`). You can only specify the media types that end with `+json` or `+xml`. The following media types are supported implicitly and cannot be configured. At runtime, the request media type is in the form of an `http Content-Type` header. The expected response media type is specified through an `Accept` header. Any service can be accessed through either of these media types. – Application/XML – Application/JSON

Element	Description
Select the type of payload that you want the endpoint to receive (this field is displayed if you selected an option in **Select the attachment processing options** field)	Select the multipart attachment type for the endpoint to receive: • **multipart/mixed**: Send an XML or JSON payload type with an attachment. For example, send a PDF document for review as a link in an email. • **multipart/form-data**: Send an XML or JSON payload type with an attachment. For example, you create an HTML form to upload and send an image. In the HTML form, the `method` is defined as `post` and the `enctype` (encoding type) is defined as `multipart/form-data`.

Configuring REST Adapter Response Properties

Enter the REST Adapter response payload details for the endpoint.

Topics

What You Can Do from the REST Adapter Response Page

You can configure the following response payload details on the trigger or invoke REST Adapter Response page.

- Specify the schema or JSON sample response payload file location.
- Select the type of payload for the endpoint to receive.

What You See on the REST Adapter Response Page

The following table describes the key information on the REST Adapter Response page.

Element	Description
Select the request payload file	Select the type of response payload file to use. The response payload body must be defined by the XSD element that defines the structure of this representation. • **XML Schema** • **JSON Sample** (Creates a sample JSON file.)
Schema Location	Click **Browse** to select the response schema file to use.
Element	Select the element that defines the payload structure. This field is not displayed until you import the response payload file. Once you browse for and select the schema file, it displays a combination box that selects the root element by default.

Element	Description
Select the type of payload you want the endpoint to receive	Select the payload type you want the endpoint to receive. • **None**: Select if no payload type is required. • **XML**: Displays the payload in XML format. • **JSON**: Displays the payload in JavaScript Object Notation (JSON) format. • **URL-encoded**: Displays the payload in URL-encoded format. • **Other Media Type**: Select to display the payload in another format (for example, `application/oracle.cloud+json`). You can only specify media types that end with `+json` or `+xml`. The following media types are supported implicitly and cannot be configured. At runtime, the request media type is in the form of an `http Content-Type` header. The expected response media type is specified through an `Accept` header. Any service can be accessed through either of these media types. – Application/XML – Application/JSON

Configuring NetSuite Adapter Properties

The NetSuite Adapter enables you to create integrations with a NetSuite cloud application.

The following sections describe the wizard pages that guide you through configuration of the NetSuite Adapter as an invoke in an integration. The NetSuite Adapter cannot be configured as a trigger in an integration.

Topic

- Configuring Basic Information Properties
- Configuring NetSuite Adapter Invoke Operation Properties
- Reviewing Configuration Values on the Summary Page

For more information about the NetSuite Adapter, see NetSuite Adapter.

Configuring NetSuite Adapter Invoke Operation Properties

Enter the NetSuite Adapter invoke operation values for your integration.

Topics

- What You Can Do from the NetSuite Adapter Invoke Operations Page
- What You See on the NetSuite Adapter Invoke Operations Page

What You Can Do from the NetSuite Adapter Invoke Operations Page

You can configure the following invoke operation values for the NetSuite Adapter.

- Select the operation to perform.
- Select the business object on which to perform the operation.

What You See on the NetSuite Adapter Invoke Operations Page

The following table describes the key information on the invoke Operations page.

Element	Description
Select Processing Options	*This feature is not currently available.*
Select Operation Type	Select the type of operation to perform on the business object: • **Basic** – **Add**: Adds a record into the system. – **Delete**: Deletes a record from the system. – **Get**: Queries the system for a record. – **Update**: Updates an existing record in the system. • **Miscellaneous** Select a specific operation. If you select **Miscellaneous**, you do not need to select a business object (the **Select Business Objects** table is not visible). – **GetBudgetExchangeRate**: Gets and filters data related to the budget exchange rate. – **GetConsolidatedExchangeRate**: Gets and filters all data related to the consolidated exchange rate. – **GetCurrencyRate**: Gets and filters the current currency rate. – **GetItemAvailability**: Retrieves the inventory availability for a given list of items. – **GetPostingTransactionSummary**: Retrieves a summary of the actual data in an account.
Filter by object name	Type the initial letters to filter the display of business objects.
Select Business Objects	Select the business object to use.
Your Selected Business Objects	Displays the selected business objects.
Processing Options	Select this link to enable and disable certain aspects of NetSuite cloud application server-side processing when performing an operation. • **Treat Warning As Error**: If selected, the endpoint treats all warning messages that are displayed by the NetSuite cloud application as errors. • **Ignore Read Only Fields**: If selected, the endpoint ignores read-only fields during any requests.

Configuring Trigger Integration Cloud Service Messaging

Configure trigger Integration Cloud Service Messaging for your integration. This dialog is displayed when you add Integration Cloud Service Messaging as a trigger to an integration.

Topics

- What You Can Do from the Trigger Oracle Integration Cloud Service Messaging Page
- What You See on the Trigger Oracle Integration Cloud Service Messaging Page

For more information, see Integration Cloud Service Messaging, Creating an Integration to Publish Messages to Integration Cloud Service, and Creating an Integration to Subscribe to Integration Cloud Service.

What You Can Do from the Trigger Oracle Integration Cloud Service Messaging Page

You can configure the trigger Oracle Integration Cloud Service Messaging. This enables you to subscribe to messages from Integration Cloud Service. Message subscription is accomplished through use of Integration Cloud Service Messaging.

Select the published integration to which to subscribe.

What You See on the Trigger Oracle Integration Cloud Service Messaging Page

The following table describes the key information on the trigger Integration Cloud Service Messaging page.

Element	Description
Select a Publisher	Select the published integration to which to subscribe. You must have already created a publisher to which to subscribe. The publisher does not need to be active, but must already be completely configured. After selecting a published integration, you perform the following tasks: • Add an invoke adapter to the integration to subscribe to the published integration. • Perform source-to-target mapping between the published integration and the invoke adapter. • Optionally configure source enrichments between the published integration and the invoke adapter.

Reviewing Configuration Values on the Summary Page

You can review the specified adapter configuration values on the Summary page.

Topics

- What You Can Do from the Summary Page
- What You See on the Summary Page

What You Can Do from the Summary Page

You can review trigger (source) or invoke (target) configuration details from the Summary page. The Summary page is the final wizard page for each adapter after you have completed your configuration.

- View the configuration details you defined for the trigger (source) or invoke (target) adapter. For example, if you have defined an inbound trigger (source) adapter with a request business object and immediate response business object, specific details about this configuration are displayed on the Summary page.
- Click **Done** if you want to save your configuration details.
- Click a specific tab in the left panel or click **Back** to access a specific page to update your configuration definitions.
- Click **Cancel** to cancel your configuration details.

What You See on the Summary Page

The following table describes the key information on the Summary page.

Element	Description
Summary	Displays a summary of the trigger (source) or invoke (target) configuration values you defined on previous pages of the wizard. The information that is displayed can vary by adapter. For some adapters, the selected business objects and operation name are displayed. For adapters for which a generated XSD file is provided, click the XSD link to view a read-only version of the file. To return to a previous page to update any values, click the appropriate tab in the left panel or click **Back**.